Available online at www.sciencedirect.com
SciVerse ScienceDirect
ELSEVIER
Procedia Computer Science 19 (2013) 428 – 436
Procedia Computer Science

The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013)

A formal model of information retrieval based on user sensitivities

Mehdi Adda

Mathematics, Computer Science and Engineering Department, University of Quebec At Rimouski (UQAR), Quebec, Canada

Abstract

Search engines are a very important web applications used by millions of users around the world on a daily basis to search the Web. Finding relevant information in this growing space is challenging, and is complicated by the diversity and needs of the community of Web users. Indeed, the Web is one, but the needs of users are multiple and different. Thus, information relevancy is not only related to the formulated query, but also to the user who is formulating this query. For example, user sensitivities may enhance information relevancy. In this paper, we are proposing to derive a formal model of user sensitivities integration into search engines.

Selection and peer-review under responsibility of Elhadi M. Shakshuki

Keywords: Web, Search Engines, Information Retrieval, User sensitivities

1. Introduction

Search engines aim to map a user query to a subset of a collection of documents that better match it. Furthermore, the millions of people around the world that are accessing the Web on a daily basis, have different needs, goals, and expectations. They are from different ideological, cultural, social, religious, and linguistic backgrounds. Hereafter, we refer to those aspects as user sensitivities. We believe that the Web has to be presented and navigated differently depending on user sensitivities. However, popular search engines in the best cases offer users limited sensitivity-based personalization features, and in the worst cases they offer the same interface/view of the Web for all Web users. It is noteworthy that some popular search engines, such as Google[1] and Bing[2], offer a feature that is mostly limited to adult content filtering with generally three levels (strict, moderated, disabled).

In this paper, we extend, through a structured methodology, the preliminary ideas presented in [1] to integrate user sensitivities into search engines by formalizing the underlying domain. The proposed formalism describes the different entities and concepts related to the domain and is based on set theory

Email address: mehdi_adda@uqar.ca (Mehdi Adda)
[1]Google: http://www.google.com
[2]Bing: http://www.bing.com

Selection and peer-review under responsibility of Elhadi M. Shakshuki
doi:10.1016/j.procs.2013.06.058

and first order logic (Sections 2, 3). Section 4 discusses related work and a conclusion of our work is presented in Section 5.

2. Architecture overview

A search engine's main mission is fulfilled by automatically and periodically performing actions such as: (1) exploring the web and download pages of interest; (2) process the downloaded pages and insert them into an index; (3) return index entries that best match user queries.

Now, the question is: How can sensitivity data be integrated to a search engine composed of the main modules listed above? User sensitivities may be integrated to search engines in different ways and at different phases of the search process. For an existing search engine with a rigid indexing schema, integration may be performed at query time. However, this method may introduce overhead during query execution due to the extra processing layer required by this approach. Another alternative consists of integrating sensitivity data to the index and extending the query processor in order to interrogate the index based on user sensitivities (See Figure 1). In this paper we have opted for the second approach as it involves a natural integration of sensitivity data into search engines. This approach is incarnated by two modules: i) *SensiCalc* and ii) *SensiAugment* that are respectively responsible for the integration of sensitivity data to the index and the enhancement of user queries with sensitivity data. Besides the sensitivity modules, the architecture overview below depicts three processes that are similar to those of classic search engines: crawling, indexing, and searching.

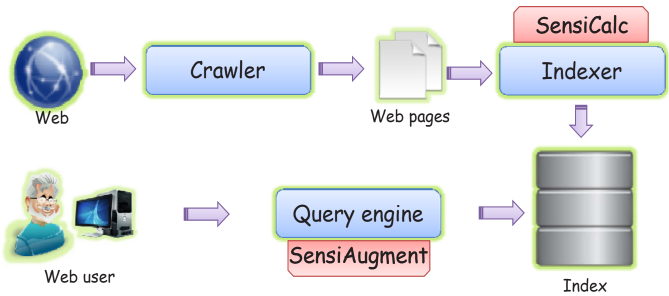

Fig. 1: Architecture overview of a Sensitivity-based Search Engine

3. Conceptual Framework

The conceptual framework of sensitivity data integration is based on a set of languages we propose to describe the different components of the sensitivity model. When presenting those languages, we generally conform to the following presentation schema: for each language, abstract concepts are presented first, followed by the concrete data that is related to those concepts. After that, we identify the relationships that are inherent in the abstract and concrete concepts such as instantiation, interpretation, subsumption, and generality order.

3.1. Setting and notation

In our setting, we refer to a domain as collection of values of the same type/family. For instance, we may represent the domain of an attribute age as a collection of numbers that belong to the interval [1..115]. Unlike a domain that may be finite, a universe is infinite by definition. The basic domain types we are using are: i) the domain of strings denoted by Str, and ii) the domain of alphanumerical values denoted by Alp. The universe of all domains is represented by UD. An attribute is a symbolic concept to which values, drawn from a domain that belongs to UD, may be assigned. The domain of an attribute

X is denoted by $[X]$. For instance, an attribute name takes its values from the domain Str and is denoted $[name] = Str$.

3.2. Sensitivity description language: classes and objects

The data language is composed of sensitivity objects and sensitivity classes (See Figure 2). The latter represent families of the former (sensitivity objects).

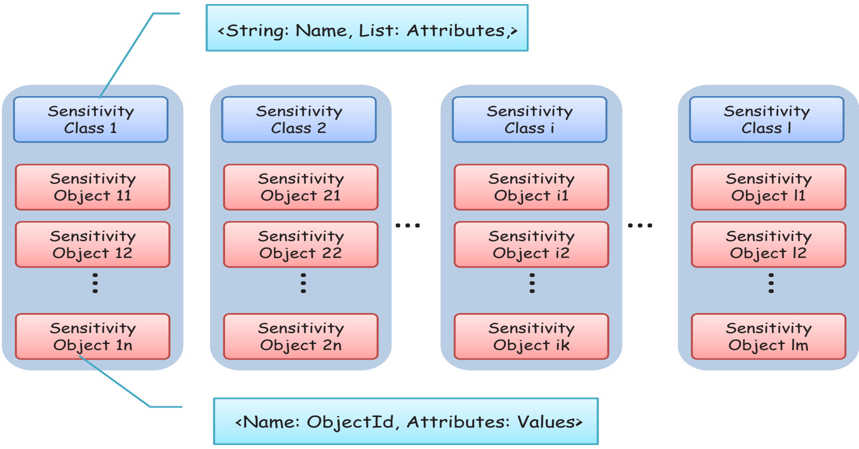

Fig. 2: Conceptual Framework: Sensitivity Classes and Objects

A sensitivity class is the central concept upon which our model rests. A class is represented by a name and a list of attributes. For example, the sensitivity class *violence* may be represented by the name "Violence" and an attribute that represents the degree of violence (See Figure 3). This attribute takes its values from the interval [0..9], where each value represents a degree of violence: 0 for absence of violence and 9 for extreme violence (See Definition 3.1).

Definition 3.1. ***Sensitivity class*** *A sensitivity class s is a couple $(name, Atts)$ where $name \in Str$ and $Atts = \langle Att_1, Att_2, .., Att_n \rangle$ such that:*

$$\forall i \in [1..n],\ \exists\ d \in U_D : [Att_i] = d$$

For the sake of simplicity, the name and the list of attributes of a class s will be respectively referred to as $s.name$ and $s.Atts$. Also, an attribute at position i will be referred to as $Atts[i]$.

The universe of all sensitivity classes is denoted by USC.

Fig. 3: The class *Violence* has one attribute named *Degree* defined on the interval [0..3]

A sensitivity object is composed of a unique *Id* and a list of values. Sensitivity objects are formally defined as follows:

Definition 3.2. ***Sensitivity object*** *A sensitivity object o is a couple (id, Vls) where $id \in Alp$ and $Vls = \langle v_1, v_2, .., v_n \rangle$ such that:*

$$\forall i \in [1..n],\ \exists\ d \in U_D : v_i \in d$$

For the sake of simplicity, the identifier and the list of values of an object o will be respectively referred to as o.id and o.Vls. The universe of all sensitivity objects is denoted by USO.

Following the example that illustrates sensitivity classes, the couple $(id_1, < 0 >)$ represents a sensitivity object. When this object is associated to a web page, it indicates that this page is classified as not containing violent content. This mapping is done by the *SensiCalc* module.

3.3. Sensitivity class/object relationships: interpretation and generality relationship

The interpretation of a sensitivity class $s \in USC$ is a subset of sensitivity objects from USO. The interpretation relationship is based on two elements: i) the domain of interpretation (USO), and ii) an interpretation function which associates to each class a set of elements of the interpretation domain (See Definition 3.4). This function is based on the instantiation relationship between a sensitivity class and a sensitivity object (See Definition 3.3). Furthermore, we define a generality order relationship among sensitivity classes that follow instantiations.

Definition 3.3. ***Sensitivity instantiation relationship*** *Let s be a sensitivity class from U_{SC} and o a sensitivity object from U_{SO}, o is instance of s, denoted by $o \lhd s$, if and only if there exists an injective function $\Psi : [1..|s|] \rightarrow [1..|o|]$ such that: $\forall k \in [1..|s|],\ o.Vls[\Psi(k)] \in [s.Atts[k]]$.*

For example, in Figure 4, sensitivity class $s = (Horror, \langle Degree : [0..9]\rangle)$ is instantiated by $o_1 = (id_1, \langle 4\rangle)$, but not $o_2 = (id_2, \langle 15\rangle)$ as 15 does not belong to the domain of the Degree attribute of s ($15 \notin [0..9]$).

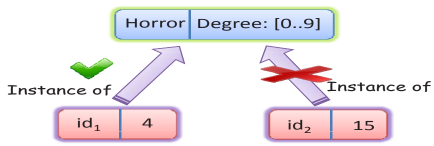

Fig. 4: The class $(Horror, \langle Degree : [0..9]\rangle)$ is instantiated by the object $(id_l, \langle 4\rangle)$. However, the object $(id_2, \langle 15\rangle)$ is not an instance of this class

Based on the instantiation definition above, the formalization of the interpretation relationship is depicted below in Definition 3.4.

Definition 3.4. ***Interpretation relationship*** *The interpretation of the class $s \in U_{SC}$, denoted by $[s]_{sc}$, is a set of sensitivity objects o such that $o \lhd s$, which means:*

$$\forall s \in U_{SC} : [s]_{sc} = \{o \in U_{SO} | o \lhd s\}$$

The class of the objet o is denoted by $\lceil o \rceil$.

The *generality* order on U_{SC} follows class instantiations as a class s_1, is superclass of another class s_2 whenever every sensitivity object that instantiates s_2 also instantiates s_1. For example, in Figure 5 where four sensitivity classes are presented, the class $(Adult, \langle Degree : [0..9]\rangle)$ is a generalization of the classes $(Violence, \langle Degree : [0..3]\rangle)$ and $(Horror, \langle Degree : [0..9]\rangle)$ as, according to common sense, each instance of the two latter classes is also an instance of the first class. However, it is not related to the sensitivity class $(MeatFood, \langle Degree : [0..1]\rangle)$ as instances of this class are not necessarily instances of $(Adult, \langle Degree : [0..9]\rangle)$.

Formally speaking, we have the following.

Definition 3.5. ***Generality relationship*** *Let s_1 and s_2 be two classes from U_{SC}. The class s_1 is more general than s_2, denoted by $s_2 \leq s_1$, if and only if:*

$$[s_2]_{sc} \subseteq [s_1]_{sc}$$

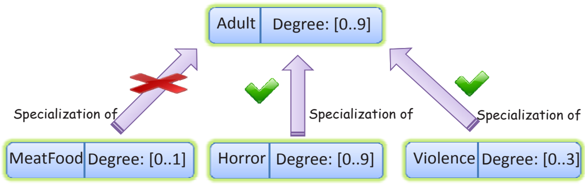

Fig. 5: The sensitivity class $(Adult, \langle Degree : [0..9]\rangle)$ is a generalization of $(Violence, \langle Degree : [0..3]\rangle)$ and $(Horror, \langle Degree : [0..9]\rangle)$ but not hierarchically related to the sensitivity class $(MeatFood, \langle Degree : [0..1]\rangle)$

As the domain of interpretation of each class is only partially available, it is impractical to use the above definition that is based on the instantiation relationship to test the generality relationship between two classes. An alternative computation mechanism based on class structural tests is designed. Hence, we propose a class relationship, called sensitivity class subsumption which relies on the structure of sensitivity classes to replace the instantiation test. Similarly to instantiation, subsumption reflects the existence of a substructure of the more specific class that maps to the more general one (See Definition 3.6).

Definition 3.6. ***Sensitivity class subsumption*** *Let s_1 and s_2 be two sensitivity classes from U_{SC}. The class s_2 subsumes s_1, denoted by $s_1 \sqsubseteq_{sc} s_2$, if there exists an injective function $\psi : [1..|s_2.Atts|] \rightarrow [1..|s_1.Atts|]$ such that $\forall i \in [1..|s_2.Atts|]$ the following conditions are satisfied.*

- *The attribute $s_1.Atts[\psi(i)]$ is semantically equivalent to $s_2.Atts[i]$, and*
- $[s_1.Atts[\psi(i)]] \subseteq [s_2.Atts[i]]$.

Figure 6 illustrates sensitivity class subsumption. It has to be noted that for the classes $s_1 = (Horror, \langle Degree : [0..5], Importance : [0..9]\rangle)$ and $s_2 = (Adult, \langle Degree : [0..5], Importance : [0..9]\rangle)$, the conditions of Definition 3.6 are satisfied, and hence conclude that s_2 subsumes s_1 via $\psi = \{(1, 1), (2, 2)\}$, but not $s_3 = (Violence, \langle Degree : [0..5]\rangle)$ since no attribute can be associated to the attribute $Importance$.

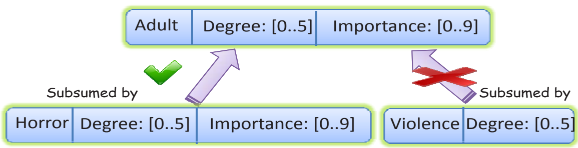

Fig. 6: The sensitivity class $(Horror, \langle Degree : [0..5], Importance : [0..9]\rangle)$ is subsumed by $(Adult, \langle Degree : [0..5], Importance : [O..9]\rangle)$. However, $(Violence, \langle Degree : [0..5]\rangle)$ is not subsumed by $(Adult, \langle Degree : [0..5], Importance : [0..9]\rangle)$

Remark 3.7. *In the remainder of this paper, we assume that our working model is based on sensitive classes, such as s, that are composed of the attribute Degree, referred to as s.deg, and it takes its values from the interval* $[0..9]$*. The first value, corresponds to the minimum value of the interval, indicates that this sensitivity is not important or present. Also, the last value that corresponds to the maximum value of that interval indicates that the sensitivity is very important. As we will see it later in this paper, the sensitivity degree is used to compute the global sensitivity of Web pages.*

3.4. Sensitivity profiles description language: schemas and instances

As shown in Figure 7, a sensibility profile schema is a collection of non-redundant sensitivity classes. For instance, $p = ((DarkHumour, \langle Degree : [0..9]\rangle), (Sex, \langle Degree : [0..9]\rangle))$ is a sensitivity profile schema which is composed of two sensitivity classes $s_1 = (DarkHumour, \langle Degree : [0..9]\rangle)$ and $s_2 = (Sex, \langle Degree : [0..9]\rangle)$. More rigorously, the sensibility profile schema is depicted below in Definition 3.8.

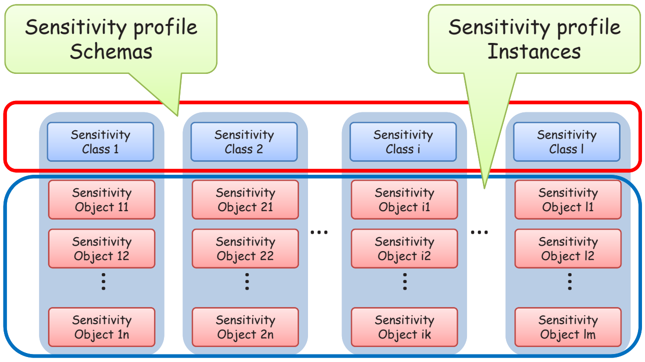

Fig. 7: Conceptual Framework: Sensitivity Profile Schemas and Sensitivity Profile Instances

Definition 3.8. ***Sensitivity profile schema*** *A sensitivity profile schema spc is a collection* $(s_1, s_2, .., s_n)$ *such that:*

- $\forall i \in [1..n] : s_i \in U_{SC}$;
- $\forall (i, j) \in ([1..n])^2 : s_i \neq s_j$.

For the sake of simplicity, each element of spc at position i will be referred to as $spc[i]$. *Also, the universe of all sensitivity profile schemas will be denoted by* U_{SPC}.

A sensitivity profile object is a collection of sensitivity objects that may be used to describe the sensitivity of a given content, or to be part of a user sensitivity profile (See Definition 3.9). It has to be noted that in a user profile, instances of the same sensitivity class are not allowed. In fact, objects from the same class may introduce confusion/contradiction to a sensitivity profile. For instance, if we consider $o_1 = (id_1, \langle 2\rangle)$ and $o_2 = (id_2, \langle 8\rangle)$ two instances of the sensitivity class $c = (Horror, \langle degree : [0..9]\rangle)$, putting o_1 and o_2 in the same profile is contradictory; while the first object states that the Horror sensitivity degree is low (2 on a scale of 10), the second object indicates that the same sensitivity is high (8 on a scale of 10).

Definition 3.9. ***Sensitivity profile object*** *A sensitivity profile object spo is a set of elements such that:*

- $\forall o \in spo : o \in U_{SO}$;
- $\forall (o_1, o_2) \in (spo)^2 : \lceil o_1 \rceil \neq \lceil o_2 \rceil$.

An element of sp at position i is referred to as $sp[i]$, *and the universe of all sensitivity profile objects is denoted by* U_{SPO}.

3.5. Sensitivity profile schema/object relationships: interpretation and generality relationship

A sensitivity profile schema $spc \in U_{SPC}$ is interpreted by a subset of U_{SPO}. The interpretation is based on a domain of interpretation (U_{SPO}), and an interpretation function which associates a set of profile objects to each profile schema. This function is based on the instantiation relationship between a sensitivity profile schema and a sensitivity profile object presented below.

Definition 3.10. ***Sensitivity profile schema instantiation relationship*** *Let spc be a sensitivity profile schema from* U_{SPC} *and spo a sensitivity profile object from* U_{SPO}*. The object spo is instance of spc, denoted by* $spo \lhd_p spc$*, if and only if there exists an injective function* $\pi : [1..|spc|] \rightarrow [1..|spo|]$ *such that:*

$$\forall k \in [1..|spc|] : \; spc[k] = \lceil spo[\pi(k)] \rceil$$

Now, the sensitivity profile interpretation relationship may be defined as follows:

Definition 3.11. ***Sensitivity profile interpretation relationship*** *The interpretation of the class* $spc \in U_{SPC}$*, denoted by* $[spc]_{spc}$*, is a set of sensitivity profile objects spo such that* $spo \lhd_p spc$*, which means:*

$$\forall spc \in U_{SPC} : [spc]_{spc} = \{spo \in U_{SPO} | spo \lhd_p spc\}$$

The profile schema of the profile objet spo is denoted by $\lceil spo \rceil_p$*.*

The *profile schema generality* order on U_{SPC} follows profile schema instantiations as a schema sps_1 is super-profile of another one, spc_2, whenever every profile object that instantiates spc_2 also instantiates spc_1. Formally, this is depicted as follows.

Definition 3.12. ***Profile schema generality relationship*** *Let* spc_1 *and* spc_2 *be two schemas from* U_{SPC}*;* spc_1 *is more general than* spc_2*, denoted by* $spc_2 \leq_p spc_1$*, if and only if:*

$$[spc_2]_{spc} \subseteq [spc_1]_{spc}$$

Similarly to the impracticability of the sensitivity class generality relationship, the profile schema generality relationship is impractical. In fact, the interpretations of profile schemas are only partially available. Thus, we designed a relationship called the *sensitivity profile schema subsumption* which is based on structural tests on profile schemas to compute the generality relationship (See Definition 3.13).

Definition 3.13. ***Sensitivity profile schema subsumption*** *Let* spc_1 *and* spc_2 *be two sensitivity profile schemas from* U_{SPC}*.* spc_2 *subsumes* spc_1*, denoted by* $spc_1 \sqsubseteq_{spc} spc_2$*, if there exists an injective function* $\psi_p : [1..|spc_2|] \rightarrow [1..|spc_1|]$ *such that:*

$$\forall i \in [1..|spc_2|], \; spc_1[\psi_p(i)] \sqsubseteq_{spc} spc_2[i]$$

3.6. User modeling

In our system, a user is represented by an identifier, a collection of sensitivity profile objects, and an active profile object. A user may have different sensitivity profiles, but only one is active at the same time. Formally speaking, a user model is described as follows:

Definition 3.14. ***User model*** *A user u from the universe of all users* U_U *is represented by* $u = (id, spos, spov)$ *such that:*

- $id \in Alp$;
- $spos = \{spo | spo \in U_{SPO}\}$;
- $spov \in spos$.

3.7. Content representation

In our model, the content refers to the content of Web pages, a.k.a web documents, and is represented by a collection of tokens (words, sentences, etc.) and a sensitivity profile object (See Definition 3.15).

Definition 3.15. ***Web document*** *A web document c is a couple* (tos, spo) *where:*

- $tos = \{to | to \in (Str \cup Alp)\}$;
- $spo \in U_{SPO}$.

The universe of all web documents is represented by U_C*.*

4. Related work

Generally speaking, a search engine may adopt different strategies: a direct search approach, navigational search, faceted search, category search, cluster search, or a personalized search [2]. While direct search offers the simplicity of a text box to users, navigational search offers users the possibility to refine search results based on the values of content attributes. Faceted search is a navigational search approach that is based on the organization of the content into multiple independent facets [3, 4, 5]. Category search is an approach that may complement direct, faceted, or navigational search [6, 7]. The principle of this approach is to search only a subset of the available corpus (search engine content). For instance, to complement faceted search, it only requires the availability of a feature that offers to users the possibility to search within one or more facets [4]. Cluster search is similar to category search and the only difference is that clustering happens at search time while categorization is generally performed when the content is indexed [8, 9, 10]. Personalized Web search approaches are intended to improve the pertinence of the retrieved results based on user profiles and/or past search experience [11, 12, 13]. Profile-based personalized search is generally seen as a system of ”get all or nothing”. Users are required to complete forms and respond to many questions [2, 14]. Studies showed that it is not efficient and users are usually not interested in filling in forms [15, 16]. Usage-based personalization is an alternative to profile-based personalization where profiles are constructed from the usage data and not from user information that is explicitly entered [17, 18, 16]. However, this approach has its own limitations such as the problems of user and session identification [19], click interpretation [20], etc. For example, if general information, such as date of birth, address, work experience, etc., is required from users, a personalized search engine may try to map the provided information with returned results in order to be filtered, or even enrich queries before their execution. We call this issue the guess gap problem. This problem consists of trying to interpret user information to figure out how it may be applied to personalize search results.

5. Concluding remarks and perspectives

In this paper, we addressed the problem of modeling information retrieval based on user sensitivities. The model we propose is based on set theory and first order logic and it captures the key elements of user sensitivity-based information retrieval system. This formalism sets the background foundations necessary to reason, verify and extend sensitivity-based systems.

There are many possible future extensions of this work. It is interesting to explore the integration of a sensitivity-based recommender. It is also possible to explore how (dynamic) faceted search may be combined with sensitivity-based search in order to enhance user experience and offer another way to navigate through search results.

References

[1] M. Adda, Building blocks of a sensitivity-based search engine, Third International Conference on Next Generation Networks and Services (2011) 1 –5doi:10.1109/NGNS.2011.6142534.

[2] D. Zhu, H. Dreher, Improving web search by categorization, clustering, and personalization, in: Proceedings of the 4th international conference on Advanced Data Mining and Applications, ADMA ’08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 659–666. doi:http://dx.doi.org/10.1007/978-3-540-88192-6_69.
URL http://dx.doi.org/10.1007/978-3-540-88192-6_69

[3] J. Koren, Y. Zhang, X. Liu, Personalized interactive faceted search, in: Proceeding of the 17th international conference on World Wide Web, WWW ’08, ACM, New York, NY, USA, 2008, pp. 477–486. doi:http://doi.acm.org/10.1145/1367497.1367562.
URL http://doi.acm.org/10.1145/1367497.1367562

[4] D. Tunkelang, Dynamic category sets: An approach for faceted search, 2006.

[5] K.-P. Yee, K. Swearingen, K. Li, M. Hearst, Faceted metadata for image search and browsing, in: Proceedings of the SIGCHI conference on Human factors in computing systems, CHI ’03, ACM, New York, NY, USA, 2003, pp. 401–408. doi:http://doi.acm.org/10.1145/642611.642681.
URL http://doi.acm.org/10.1145/642611.642681

[6] M.-K. Shan, M.-F. Chiang, F.-F. Kuo, Relevance feedback for category search in music retrieval based on semantic concept learning, Multimedia Tools Appl. 39 (2008) 243–262. doi:10.1007/s11042-008-0201-8.
URL http://portal.acm.org/citation.cfm?id=1388468.1388472

[7] G. P. Nguyen, M. Worring, Scenario optimization for interactive category search, in: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval, MIR '05, ACM, New York, NY, USA, 2005, pp. 159–166. doi:http://doi.acm.org/10.1145/1101826.1101853.
URL http://doi.acm.org/10.1145/1101826.1101853

[8] M. A. Hearst, Clustering versus faceted categories for information exploration, Commun. ACM 49 (2006) 59–61. doi:http://doi.acm.org/10.1145/1121949.1121983.
URL http://doi.acm.org/10.1145/1121949.1121983

[9] P. Blunsom, K. Kocik, J. R. Curran, Question classification with log-linear models, in: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '06, ACM, New York, NY, USA, 2006, pp. 615–616. doi:http://doi.acm.org/10.1145/1148170.1148282.
URL http://doi.acm.org/10.1145/1148170.1148282

[10] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, J. Ma, Learning to cluster web search results, in: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '04, ACM, New York, NY, USA, 2004, pp. 210–217. doi:http://doi.acm.org/10.1145/1008992.1009030.
URL http://doi.acm.org/10.1145/1008992.1009030

[11] Y.-A. Chen, Semantics-based customization of information retrieval on the world wide web, Ph.D. thesis, Los Angeles, CA, USA, aAI3233848 (2006).

[12] W. Fan, M. D. Gordon, P. Pathak, Personalization of search engine services for effective retrieval and knowledge management, in: Proceedings of the twenty first international conference on Information systems, ICIS '00, Association for Information Systems, Atlanta, GA, USA, 2000, pp. 20–34.
URL http://portal.acm.org/citation.cfm?id=359640.359720

[13] F. Liu, C. Yu, W. Meng, Personalized web search for improving retrieval effectiveness, IEEE Trans. on Knowl. and Data Eng. 16 (2004) 28–40. doi:http://dx.doi.org/10.1109/TKDE.2004.1264820.
URL http://dx.doi.org/10.1109/TKDE.2004.1264820

[14] P. De Meo, G. Quattrone, D. Ursino, A query expansion and user profile enrichment approach to improve the performance of recommender systems operating on a folksonomy, User Modeling and User-Adapted Interaction 20 (2010) 41–86. doi:http://dx.doi.org/10.1007/s11257-010-9072-6.
URL http://dx.doi.org/10.1007/s11257-010-9072-6

[15] S. Gauch, M. Speretta, A. Chandramouli, A. Micarelli, User profiles for personalized information access, in: P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web, Vol. 4321 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, pp. 54–89, 10.1007/978-3-540-72079-9_2.
URL http://dx.doi.org/10.1007/978-3-540-72079-9_2

[16] K. Sugiyama, K. Hatano, M. Yoshikawa, Adaptive web search based on user profile constructed without any effort from users, in: Proceedings of the 13th international conference on World Wide Web, WWW '04, ACM, New York, NY, USA, 2004, pp. 675–684. doi:http://doi.acm.org/10.1145/988672.988764.
URL http://doi.acm.org/10.1145/988672.988764

[17] M. Eirinaki, M. Vazirgiannis, Usage-based pagerank for web personalization, in: Proceedings of the Fifth IEEE International Conference on Data Mining, ICDM '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 130–137. doi:http://dx.doi.org/10.1109/ICDM.2005.148.
URL http://dx.doi.org/10.1109/ICDM.2005.148

[18] E. Yilmaz, M. Shokouhi, N. Craswell, S. Robertson, Expected browsing utility for web search evaluation, in: Proceedings of the 19th ACM international conference on Information and knowledge management, CIKM '10, ACM, New York, NY, USA, 2010, pp. 1561–1564. doi:http://doi.acm.org/10.1145/1871437.1871672.
URL http://doi.acm.org/10.1145/1871437.1871672

[19] Z. Chen, A. W.-C. Fu, F. C.-H. Tong, Optimal algorithms for finding user access sessions from very large web logs, World Wide Web 6 (2003) 259–279. doi:10.1023/A:1024606901978.
URL http://portal.acm.org/citation.cfm?id=850668.850689

[20] H. Cheng, E. Cantú-Paz, Personalized click prediction in sponsored search, in: Proceedings of the third ACM international conference on Web search and data mining, WSDM '10, ACM, New York, NY, USA, 2010, pp. 351–360. doi:http://doi.acm.org/10.1145/1718487.1718531.
URL http://doi.acm.org/10.1145/1718487.1718531