Implicit User Modelling Using Hybrid Meta-heuristics

Pavel Krömer, Václav Snášel, Jan Platoš
FEECS, Dept. of Computer Science
VŠB – Technical University of Ostrava
17. listopadu 15, 708 33 Ostrava–Poruba
Czech Republic
{pavel.kromer.fei, vaclav.snasel,jan.platos}@vsb.cz

Ajith Abraham
Department of Computer Science
Norwegian University of Science and Technology, Trondheim,
Norway
ajith.abraham@ieee.org

Abstract

The requirements imposed on information retrieval systems are increasing steadily. The vast number of documents in today's large databases and especially on World Wide Web causes notable problems when searching for concrete information. It is difficult to find satisfactory information that accurately matches user information needs even if it is present in the database. One of the key elements when searching the web is proper formulation of user queries. Search effectiveness can be seen as the accuracy of matching user information needs against the retrieved information. Personalized search applications can notably contribute to the improvement of web search effectiveness. In this paper, we investigate two user modelling and search optimization techniques based on genetic algorithms and ant colony optimization.

1. Introduction

The requirements imposed on search applications are increasing steadily. The amount of available data is growing and user demands as well. The search application should provide the users with accurate, sensible responses to their requests. Unified consensual approach to search requirements of all inquirers becomes with growing amount of data and documents on the WWW inefficient to satisfy needs of large number of individuals desiring to retrieve particular information from Internet. Personalized approach to the needs of each user is general trend in state-of-the-art web applications including search engines. Personalization, based on stored knowledge of users' general needs, area(s) of interest, usual behaviour, long and short term context and search practices can be evaluated when improving web search applications, no matter if they are standalone search engines or more advanced meta-search systems lying on the top of individual search applications.

In this paper, we propose click-through data based document relevance estimation method for creating user profiles. Moreover, we present exploitation of such user profiles for efficient improvement of search effectiveness.

2. Web Search Personalization

In order to offer personalized search services, user profiles or models are needed and user modelling becomes an important task of advanced search engines. A proper user model provides accurate and sufficient information on user that can be exploited in the optimization phase. User profiling is non-trivial branch of information retrieval under investigation of several groups worldwide introducing multiple methods.

2.1. User modelling

An individual user profile (IUP), containing stored knowledge about system user, could be utilized to improve search results through personalization. Search engine with user profiling could exploit user-specific acquirements to retrieve documents satisfying search queries with respect to individual user, her or his general needs, preferences, abilities, history, knowledge and current context.

The profile consists usually of keywords (*simple profile*) or it could include personal data (*extended user profile*). Advanced user profiles contain rather than set of keywords a list of queries characterizing the users' preferences.

Explicit profiles, defined by users themselves, are rather imprecise and not flexible enough. Instead, various techniques for implicit creation and maintenance of user profiles are being investigated [1].

2.2. Click-through data

Among the most promising methods, personalization techniques based on click-through data analysis attract attention [2, 3]. Click-through data recorded during web search activities might be seen as triplet (q, R, C) consisting of query q, ordered set of retrieved documents R and set of clicks C denoting documents user picked from the set of retrieved documents R, introducing individual search preferences [2].

The appeal of click-through data analysis for user profiling is based on several facts. It is omnipresent during web browsing – click-through data is present in the web browsing activities always. The clicks are needed by the very essential structure of html documents and the WWW.

Click-through data is implicit – user clicks are almost necessary to browse the web. Click-through data gathering must not be seen as an additional disturbing or obstructing activity. The clicks (or alternative link-following actions) are necessary to work with web. Additionally, click-through data has relevance feedback potential. The users click on links that he or she feels as relevant to his or her needs. Mostly, these links relevant by belief are really relevant to previous request although the essential information contained in click-through data is still under investigation.

Click-through is up-to-date and with appropriate analysis, the data gathered for sufficient time period could provide information on both, users long time interests and needs and immediate contemporary context. Finally, click-through data stored in query logs can be used for many methods of information retrieval improvement, including offline techniques. Summarizing, most users click on rather relevant results and we should benefit from a large quantity of query logs. Experiments show that about 82% of the queries are in fact related to the topics of the clicked Web pages [4].

On the other hand, there are known issues with click-through data [2, 3]: it is usually noisy and rather incomplete piece of evidence about user's relevance assessments. It is sparse since user clicks can cover only very small portion of WWW document space.

Click-through data collecting can be done on the top of current search systems and services. There could be a server based solution, observing user click behaviour from some central point like web application used as proxy for access to search services or client based solution tracking user clicks from i.e. web browser. The web application is limited by its scope and as soon as the user leaves the application, the clicks are almost unrecordable. The client application is limited by the abilities of user workstations; the accommodation of such application must not be disturbing, i.e. it must not consume too much processor time, memory or disc space.

3. Implicit relevance based user model

In this section, we provide description of implicit web search user model.

3.1. Analysis of web information retrieval process

Information retrieval provides means for discovery of information satisfying user needs in large amounts of data. The World Wide Web can be seen as large collection of hypertext, plaintext and multimedia documents. Web search services, provided by Google, Yahoo etc, are IR systems implementing certain IR technique developed and tuned for efficient performance over web document space. Currently, majority of web search services, including the biggest providers, offer consensual search not aware of individual inquirers.

Common search session proceeds as follows:

1. An information need comes into existence
2. User formulates search expression in query language of chosen web IR system
3. The search request is being processed (computer level IR tasks are performed). Search engine presents search results.
4. User picks some of the presented results (human level IR tasks are performed)

From the previous, we can see that the whole web IR process consist of two levels of IR tasks – computer level IR task performed by search engine and human level IR performed by user over the presented results. Result set shown by the search engine in response to user query is the basis for higher level IR decision by the user. Let us discuss the common structure of search results. The ordered result set consists of triplets (u, n, s) where u is URL of the document, n is name of the document and s is short textual description or resume of the document. Therefore, human level IR decision can be seen as IR over collection of textual documents where the content of the document is its summary s.

3.2. Document relevance estimation based on click-through data

Document relevance estimation model based on click-through data consists of recorded clicks committed by particular user. Each click c is captured a triplet (u, d, t), where $u \in U$ is particular user from the set of all users U, $d \in D$ is the clicked document and t is timestamp, describing moment in which the click was committed. $D = \{(u, n, s)\}$ is set of all documents known to the application. Consider $c_t : D \times U \rightarrow R^n$ as a set of timestamps describing clicks issued by particular user on certain document. For each document and user, the relevance $r : D \times U \rightarrow R$ is estimated by (1).

$$r(u,d) = \sum_{t \in c_t(u,d)} f'(t) \qquad (1)$$

The function f enumerates the contribution of click issued at time t to relevance estimate of the document and t' denotes age of the click. The contribution function used in this paper is reversed asymmetric sigmoid as defined in (2).

$$f(t) = 1 - \frac{1}{\left(1 + e^{\frac{t + c \cdot \ln\left(2^{\frac{1}{d}} - 1\right) - b}{c}}\right)^{d}} \qquad (2)$$

Asymmetric reverse sigmoid as defined in (2) is highly customizable function. The parameter b denotes centre of the transition, c and d are used for enumeration of transition width w as specified in (3).

$$w = \left| c \cdot \ln\left(4^{\frac{1}{d}} - 1\right) - c \cdot \ln\left(4^{\frac{1}{d}} \cdot 3^{\frac{-1}{d}} - 1\right) \right| \qquad (3)$$

Figure 1 illustrates the reverse asymmetric sigmoid with b=5, c= -2 and d=10. The x axis unit is day and the transition width is 3.256. The scale of x axis in presented work is subject of further customization and it is variable parameter for different deployment cases of presented method.

Additionally, user model contains recorded recent user queries to be exploited later during query optimization process as an initial population for optimizing genetic algorithm.

4. Evolutionary query optimization

User profiling can be used for search optimization. The knowledge about user must be reasonably exploited to improve the level of search. As the most promising, we see the possibility to build upon contemporary search services that are mature and advanced in indexing the web and performing consensual information retrieval tasks. User profiling and search optimization might leverage existing search services including meta-search engines solving another issue of web querying, the imperfect coverage of web data by standalone search engines.

Search improvement based on evolutionary query optimization is profitable in more ways. First, evolutionary algorithms and especially genetic programming are capable to evolve complex symbolic structures like computer programs [5], mathematical equations or for our purposes search queries [6, 7]. Second, the optimization, if properly designed, can be executed on the top of existing search environment in such way that it will exploit only information from user profile and responses from search engine. The queries evolved over just a small portion of search space (the space of all web documents) will be applicable on whole document space (they will be again evaluated using the underlying search engine).

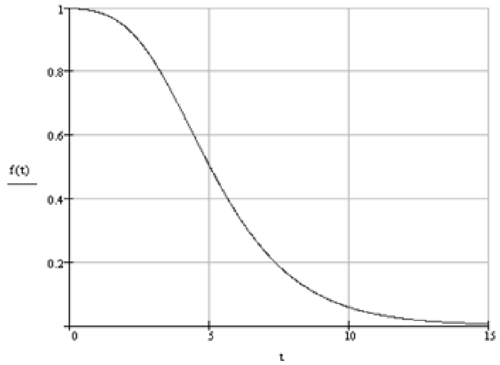

Figure 1. Click contribution according to its age using reverse asymmetric sigmoid

In this section, an approach exploiting information from user profiles to optimize search queries by artificial evolution will be discussed.

4.1. Evolutionary query optimization exploiting user model

Recently, a genetic programming based technique of search query optimization has been introduced [6, 7]. It has been shown that genetic programming is capable to optimize user queries towards users area of interest described by a collection of documents annotated by the means of relevance. An IR model of the document collection was constructed and the search tasks performed, evaluating the method in laboratory experiments. The drawback of the technique was missing method of relevance assessment to real world document documents; the relevance assignment was part of the experiments. In this paper, we propose the exploitation of optimization method from [6, 7] in the web search environment.

In previous sections, we have identified set of documents retrieved by the search engine in response to users query as an ordered collection of records in the form (u, n, s). To support human level IR task, user query was evolved over an IR model describing the collection of retrieved documents taking the document summary s as content of the document. The IR model was created by the means of extended Boolean IR model featuring document representation as fuzzy set of index terms and Boolean search queries [8, 9, 10].

Document model was created using TFIDFT indexing formula by Salton [11] and exploiting Porters stemming algorithm [12] while removing common English words with poor distinctive meaning found in English stop-list[1]. For each of such modelled documents, the relevance was estimated using the method introduced in section 3.2. As initial population for evolutionary query optimization, the last 100 user queries were taken. If there were less than 100 user queries, the missing ones were generated randomly using terms captured by user profile. For the purpose of genetic programming, the queries were encoded into tree-like chromosomes corresponding to their derivation (parse) trees according to Boolean query language grammar. The evolution was executed for 200 generations. For detailed setup of other genetic parameters used for query optimization see [6, 7]. Following section describes in detail performed experiments.

4.2. Evolutionary query optimization experiments

To evaluate proposed user modelling method and search optimization technique, a set of experiments comparing search experience in different cases with and without query optimization support was designed and performed. Number of emitted queries, average click rate and length of mouse trajectory created before retrieving satisfactory information were traced as objective measure of search task.

Intentionally, user queries were during experiments constructed from simple to more complex. In order to create initial user profile for optimized search, the participants performed common search activities focused on evolutionary algorithms and optimization techniques. The resulting profile snapshot contained 1044 terms in 120 documents and 25 queries such as:

"genetic" AND "algorithm"
"genetic" AND "operator"
"dynamic" AND "optimization"
"dynamic" AND "optimization" AND "task" AND NOT "dbm"

We have performed three experiments aiming to recognize the effect of evolutionary query optimization on users search experience. In the first two search sessions, questions from topics covered by the user profile (EA, optimization) were submitted to the system. The last search was unrelated to user profile themes. The experiments are summarized in Table 1, where *NO* denotes non-optimized search and *O* denotes optimized search.

Table 1. Summary of evolutionary query optimization experiments.

Search type Criterion	*Experiment 1*		*Experiment 2*		*Experiment 3*	
	NO	*O*	*NO*	*O*	*NO*	*O*
No. of queries	4	1	5	2	4	4
No. of clicks	55	7	22	14	92	104
Mouse trajectory [m]	12	1,8	4,6	2,2	14,4	17,2

As expected, when the searched information was covered by the user profile, the optimized query improved the search results (less queries, clicks and mouse movements were needed to get the information). When searching information not contained in the profile, the optimization brings no benefits.

5. Ant Colony Optimization for Personalized Search

Ant Colony Optimization (ACO) [13, 14] is a popular meta-heuristics based on certain behavioral patterns of foraging ants. Ants have shown ability to find optimal paths between their nest and source of food. The intelligent path-finding activity is based on indirect communication consisting of modification of their environment (stigmery). Ants travel randomly to find food and when returning to their nest, they lay down pheromones. When other foraging ants encounter a pheromone trail, they are likely to follow it. The more ants travel on the same trail, the more intensive is the pheromone trace and the more attractive is it for other ants.

Emulation of ants' behavior can be used as probabilistic computational technique for solving complex problems which can be reduced to finding optimal paths [13, 14]. An artificial ant k placed in vertex i moves to node j with probability p_{ij}^k:

$$p_{ij}^k = \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{l \in N_i^k} \tau_{il}^\alpha \eta_{il}^\beta}, \text{ if } j \in N_i^k \qquad (4)$$

where N_i^k represents the neighborhood of ant k in node i (i.e. nodes that are available to move on), τ_{ij} represents amount of pheromones placed on arc a_{ij} and η_{ij} corresponds to a-priori information reflecting the cost of passing arc a_{ij}. After the ants finish their movement forward, they return to the nest with food. The amount of collected food L^k (i.e. solution quality) is used to specify the amount of pheromones $\Delta\tau^k$ to be placed by ant k on each arc on the trail that led to the food source:

$$\Delta\tau^k = \frac{1}{L^k} \qquad (5)$$
$$\tau_{ij} = \tau_{ij} + \Delta\tau^k$$

[1] http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/

After all ants finish one round of their movement, the pheromones evaporate (i.e. the amount of pheromones on each arc is reduced):

$$\tau_{ij} = (1-\rho)\tau_{ij} \qquad (6)$$

The coefficientsα, β and ρ are general parameters of the algorithm.

5.1. ACO approach to search optimization

In personalized search, the objective document-query similarity (i.e. the estimate of document relevance with respect to query in current IR model), usually based on some term statistics, has to be supplemented by subjective relevance. The results of search process have to be constructed so that they reflect both, query-term similarity and individual relevance.

In order to apply ACO to search optimization, let us define personalized search as path-finding graph task. A collection of N documents with assigned relevance can be seen as fully connected graph $G = (V, A)$ where $V = \{d_1, d_2, \ldots, d_N\}$ and $A = \{a_{ij}\}$ for all $i,j \in \{1,\ldots,N\}$ subject to $i \neq j$. The vertices (relevance weighted documents) are connected with arcs. The goal of search process is to retrieve a set of k documents that are relevant to user query.

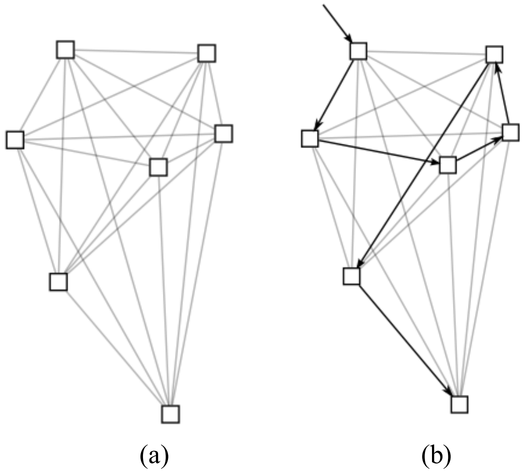

Figure 2. Document collection as fully connected graph (a) and search as path in document graph (b)

The process of retrieving a set of k documents in response to a query can be modeled as a search for optimal path between k nodes in a fully connected graph. The sum of relevance weight of the vertices in the search path represents the quality of the solution. The quality of the solution defines the amount of pheromones that will be placed on the arcs representing the search path. The a-priori information η_{ij} corresponds to document-query similarity between query q associated with particular ant in node i and document d_j corresponding to vertex j. The Ant Colony Optimization for personalized search (ACOps) algorithm can be for every search session (i.e. for every submitted query q) summarized as follows:

1. Place k ants onto the document graph G randomly according to the estimated relevance of documents.
2. Let every ant k move n steps forward (i.e. retrieve n documents) with probability of transition from d_i to d_j specified by p_{ij} from (4). Evaluate the solution quality L^k of path discovered by every ant.
3. Evaporate pheromones according to (6)
4. Update pheromone trails according to (5)
5. Repeat 1-4 several times. At the end, pick best valued path, order documents by relevance and document-query similarity and present the results to the user.

The algorithm considers both, objective document query similarity expressed by a-priori information η_{ij} and individual user preferences expressed by relevance estimates (L^k) and captured in pheromone trials τ_{ij}.

5.2. ACOps experiments

We have conducted a set of experiments to evaluate the ability of ACOps to retrieve relevant documents. The same data as in evolutionary query optimization experiment were used as a sandbox for ant ACO search optimization. Each document in the collection has known relevance and its retrieval status value to any query can be evaluated. The goal of the experiment was to investigate whether, and under what conditions, can the ACOps discover more relevant documents in the document collection. The experiment consisted in submission of query to the system and ACOps optimization of the results. 10 documents were retrieved.

The submitted queries were:

1. *“genetic" AND "algorithm”*
2. *“adaptive” AND “optimization”*
3. *“weather" AND "daylight" AND "Ostrava”*

The ACOps experiments showed excellent ability of Ant Colony Optimization algorithm to retrieve relevant documents from data base. It is able to recognize and prioritize relevant documents over similar documents. The increase of relevance of the result set was rapid (see Table 2, Experiment 1). ACOps failed in retrieving novell information from the data basis (Experiment 2 and Experiment 3).

Table 2. Summary of ACOps experiments.

Query Metric	*Experiment1*	*Experiment 2*	*Experiment 3*
Avg. similarity	0,25	0	0
∑ relevance	0,00459	0,00115	0
∑ rel. with ACOps	1,00344	3,29088	3,14
Available relevance	1,01034	0	0

Indeed, this is not a surprising result. The algorithm was designed to mine hidden relevance-based relationships among documents in user portfolio. In case we want to retrieve completely new information, which is not covered by the user profile, ACOps brings no benefit to inquirer.

6. Conclusions and future work

This paper presents novel approach to web inquirers modelling for personalized search. Proposed system was implemented and experimentally evaluated. The performed experiments show that evolutionary query optimization is able to improve (speed up) search in the areas covered by the user profile. It is not able to improve search process when aiming to totally new area of interest, however in real-life deployment; the application would be able to learn from every submitted query while in presented set of experiments the profile was constant.

Next investigated search improvement method, the ant colony optimization for personalized search, has proven good ability to mine relevant documents from user profile. It can be used to prefer truly relevant documents over statistically similar documents and it can help in getting right and relevant answers in response to vague and imprecise questions. It is not designed to discover new information (such as evolutionary query optimization) but to reveal implicit relationships between user, queries and documents in the profile.

References

[1] O. Cordón, F. de Moya, and C. Zarco, "Fuzzy logic and multiobjective evolutionary algorithms as soft computing tools for persistent query learning in text retrieval environments," in *IEEE International Conference on Fuzzy Systems 2004*, (Budapest, Hungary), pp. 571–576, 2004.

[2] T. Joachims, "Optimizing search engines using clickthrough data," in *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, ACM, 2002.

[3] G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan, "Optimizing web search using web clickthrough data," in *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, (New York, NY, USA), pp. 118–126, ACM Press, 2004.

[4] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," in *Readings in Fuzzy Sets for Intelligent Systems* (D. Dubois, H. Prade, and R. R. Yager, eds.), pp. 80–87, San Mateo, CA: Kaufmann, 1993.

[5] J. Koza, "*Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems*," Technical Report STAN-CS-90-1314, Dept. of Computer Science, Stanford University, 1990.

[6] D. Húsek, V. Snášel, R. Neruda, S. S. J. Owais, and P. Krömer, "Boolean queries optimization by genetic programming," *WSEAS Transactions on Information Science and Applications*, vol. 3, no. 1, pp. 15–20, 2006.

[7] S. Owais, P. Kromer, V. Snasel, D. Husek, and R. Neruda, "Implementing GP on optimizing both boolean and extended boolean queries in IR and fuzzy IR systems with respect to the users profiles," in *Proceedings of the 2006 IEEE Congress on Evolutionary Computation* (G. G. Yen, L. Wang, P. Bonissone, and S. M. Lucas, eds.), (Vancouver, BC, Canada), pp. 5648–5654, IEEE Computer Society, 6-21 July 2006.

[8] G. Bordogna and G. Pasi, "Modeling vagueness in information retrieval," pp. 207–241, 2001.

[9] D. H. Kraft, F. E. Petry, B. P. Buckles, and T. Sadasivan, "Genetic Algorithms for Query Optimization in Information Retrieval: Relevance Feedback," in *Genetic Algorithms and Fuzzy Logic Systems* (E. Sanchez, T. Shibata, and L. Zadeh, eds.), (Singapore), World Scientific, 1997.

[10] F. Crestani and G. Pasi, "Soft information retrieval: Applications of fuzzy set theory and neural networks," in *Neuro-Fuzzy Techniques for Intelligent Information Systems* (N. Kasabov and R. Kozma, eds.), pp. 287–315, Heidelberg, DE: Springer Verlag, 1999.

[11] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. pp. 513–523, 1988.

[12] M. F. Porter, *An algorithm for suffix stripping*, pp. 313–316. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997.

[13] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Cambridge, MA: MIT Press, 2004.

[14] A. Abraham, H. Guo, and H. Liu, "Swarm intelligence: Foundations, perspectives and applications," in *Swarm Intelligent Systems* (N. Nedjah and L. de Macedo Mourelle, eds.), vol. 26 of *Studies in Computational Intelligence*, pp. 3–25, Springer, 2006.