# INFSCI 2480 Adaptive Information Systems Personalized Web Search

#### Peter Brusilovsky

http://www.sis.pitt.edu/~peterb/2480-012

# Why Search Personalization?

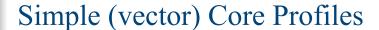
- R. Larsen: With the growth of DL even a good query can return not just tens, but thousands of "relevant" documents<sup>1</sup>
- Personalization is an attempt to find most relevant documents using information about user's goals, knowledge, preferences, navigation history, etc.
- Larsen, R.L. Relaxing Assumptions . . . Stretching the Vision: A Modest View of Some Technical Issues. D-Lib Magazine, 3, April (1997), available online at http://www.dlib.org/dlib/april97/04larsen.html



- Common term for user models in IR/IF
- A user's profile is a collection of information about the user of the system.
- This information is used to get the user to more relevant information
- Views on user profiles in IR community
  - Classic (Korfhage) a reference point
  - Modern simple form of a user model

#### Core vs. Extended User Profile

- Core profile
  - contains information related to the user search goals and interests
- Extended profile
  - contains information related to the user as a person in order to understand or model the use that a person will make with the information retrieved



- Primitive profile (any model)
  - A set of search terms (0-1 vector)
- For Boolean model of IR
  - A Boolean query
- For vector model of IR (dominated)
  - A set of terms with their weights (vector)
  - An overlay (set of weights) over a simple domain model that is just a list of terms that could be of interest to the users

#### Who Maintains the Profile?

- Profile is provided and maintained by the user/administrator
  - Sometimes the only choice
- The system constructs and updates the profile (automatic personalization)
- Collaborative user and system
  - User creates, system maintains
  - User can influence and edit
  - Does it help or not?

### Adaptive Search

#### Goals:

 Present documents (pages) that are most suitable for the individual user

#### Methods:

- Employ user profiles representing shortterm and/or long-term interests (Korfhage)
- Rank and present search results taking both user query and user profile into account

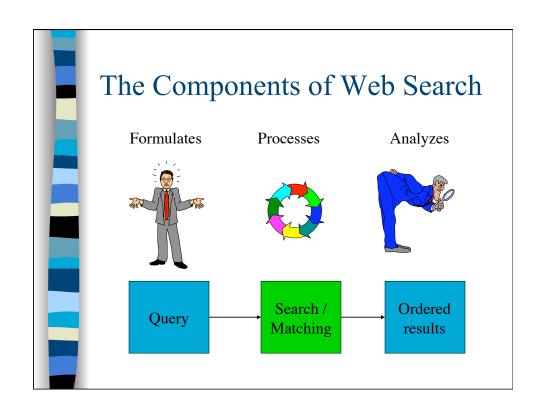
#### Personalized Search: Benefits

#### Resolving ambiguity

- The profile provides a context to the query in order to reduce ambiguity.
- Example: The profile of interests will allow to distinguish what the user asked about "Berkeley" ("Pirates", "Jaguar") really wants

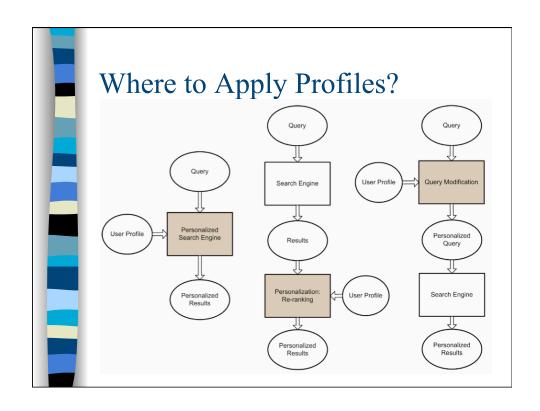
#### Revealing hidden treasures

- The profile allows to bring to surface most relevant documents, which could be hidden beyond top results page
- Example: Owner of *iPhone* searches for *Google Android*.
   Pages referring to both would be most interesting



# Where to Apply Profiles?

- The user profile can be applied in several ways:
  - To modify the query itself (pre-processing)
  - To change the usual way of retrieval
  - To process results of a query (postprocessing)
  - To present document snippets
  - Special case: adaptation for meta-search

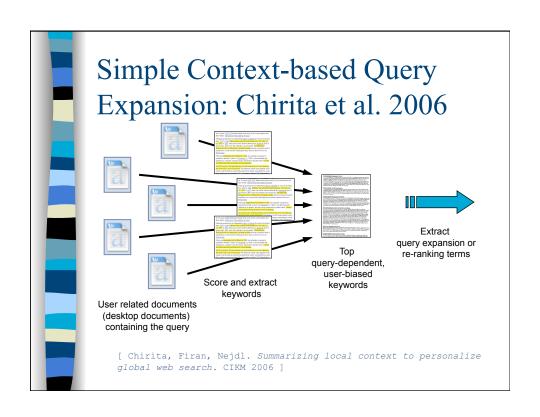


# **Examples of Systems**

- Pre-process with QE Koutrika, Mobasher, Chirita
- Pre-process with RF : SmartGuide
- Post-process with annotations: Syskill & Webert
- Post-process with re-ranking: Syskill & Webert, WIFS, YourNews, TaskSieve
- Adaptive Snippets: TaskSieve

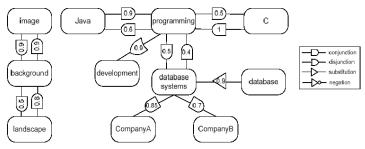
# Pre-Process: Query Expansion

- User profile is applied to add terms to the query
  - Popular terms could be added to introduce context
  - Similar terms could be added to resolve indexer-user mismatch
  - Related terms could be added to resolve ambiguity
  - Works with any IR model or search engine





- Advanced relevance network for query expansion
- java -> java and programming -> java and (programming or development)



A Unified User-Profile Framework for Query Disambiguation and Personalization Georgia Koutrika and Yannis Ioannidis, http://adiret.cs.uni-magdeburg.de/pia2005/Proceedings.htm

#### Pre-Process: Relevance Feedback

- In this case the profile is used to "move" the query vector (vector model only)
- Imagine that:
  - the documents,
  - the query
  - the user profile

are represented by the same set of weighted index terms



- The query  $q=q_1, q_2, \dots q_n$
- The profile  $p=p_1, p_2, \dots p_n$
- The query modified by the user profile will be something like that:

modified 
$$q_i = Kp_i + (1-K)q_i$$
  $i=1,2,...n$ 

# Pre-process: Linear Transformation

modified  $q_i = Kp_i + (1-K)q_i$ 

■ In this case we add the terms of the profile to the query ones, weighted by *K* 

for K=0 modified  $q_i=q_i$  the query is unmodified for K=1 modified  $q_i=p_i$  the query is substituted by the profile

#### Piecewise Linear Transformation

- if the term appears in the query and in the profile then the linear transformation is applied
- if the term appears in the query but not in the profile is left unmodified or diminished slightly
- if the term appears in the profile but not in the query it is not introduced, or introduced with a weight lower than in the profile.

# Example: SmartGuide

- Access to the CIS-like information
- User has a long-term interests profile and current queries
- Information is searched using a combination of both
- Profile is initiated from a stereotype and kept updated
- Increased user satisfaction, decreased navigation overhead

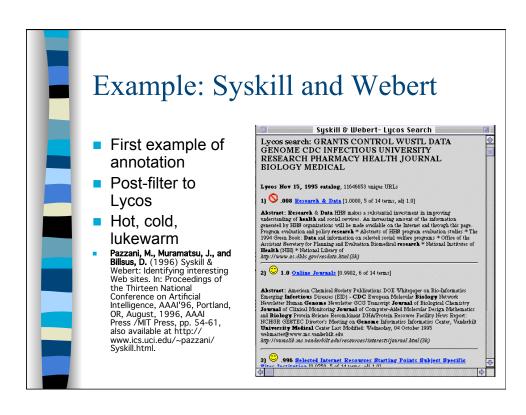
Gates, K. F., Lawhead, P. B., and Wilkins, D. E. (1998) Toward an adaptive WWW: a case study in customized hypermedia. *New Review of Multimedia and Hypermedia* **4**, 89-113.

# Post-Processing

- The user profile is used to organize the results of the retrieval process
  - present to the user the most interesting documents
  - Filter out irrelevant documents
- Extended profile can be used effectively
- In this case the use of the profile adds an extra step to processing
- Similar to classic information filtering problem
- Typical way for adaptive Web IR

#### Post-Filter: Annotations

- The result could be relevant to the user in several aspects. Fusing this relevance with query relevance is error prone and leads to a loss of data
- Results are ranked by the query relevance, but annotated with visual cues reflecting other kinds of relevance
  - User interests Syskill and Webert, group interests KnowledgeSea



# Post-Filter: Re-Ranking Re-ranking is a typical approach for post-filtering Each document is rated according to its relevance (similarity) to the user or group profile This rating is fused with the relevance rating returned by the search engine The results are ranked by fused rating User model: WIFS, group model: I-Spy

# Example: WIFS (Micarelli)

- Adaptive post-filter to AltaVista search engine
- Maintains an advanced stereotypebased user model (Humos subsystem)
- User model is updated by watching the user
- The model is used to filter and re-order the links returned by AltaVista

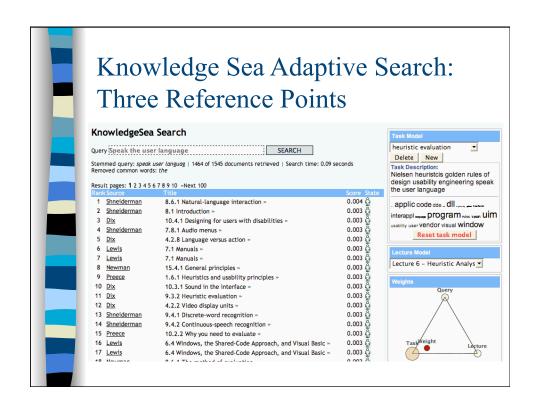




# TaskSieve: Adaptive Snippets The goal of a usual snippet is to show query relevance. TaskSieve applies adaptive snippets to show profile relevance as well Selects top 3 relevant sentences combining query relevance and task relevance to sentences Applies color coding by query/profile ZBN20001113.0400.0019 ★★★★ (0.80) Salzburg, Austria, Burg introduce the state governol passengers, of which 12 people escape, and the rest lovers who he said that as a driver were killed in fire, ... Tunnel fire extinguished, Austria rescue personn the train to find the bodies of victims.

### Knowledge Sea Adaptive Search: Three Reference Points

- Adaptive search in TaskSieve uses linear combinations of two ranks - query-based and profile-based to calculate the final rank
- Query and profile are two reference points for ranking
- What if there are three reference points?
- Knowledge Sea Search allows to do useradaptive ranking with three reference points: Query, task profile, and lecture





- Query and Profile are considered as Separate Reference Points
- In this case documents are retrieved if they are "near" the query or the profile.
- For the following slides, let's assume that the similarity is measured by distance

where D is the document and Q is the query

# Separate Reference Points

- We have different way to integrate query and profile as separate reference points:
  - Disjunctive model of query-profile integration
  - Conjunctive model of query-profile integration
  - Ellipsoidal model
  - Cassini oval model

# Disjunctive Model

We will take the document if the following condition is satisfied:

$$\min(||D,Q||,||D,P||) < d$$

The D document should be "near" the query Q or the profile P

# Conjunctive Model

We will take the document if the following condition is satisfied:

$$\max(||D,Q||,||D,P||) < d$$

- The D document should be "near" the query Q and the profile P
- In this case if the profile and the query have little in common very few documents are retrieved

