Universitatea Politehnica București

Facultatea de Automatică și Calculatoare

Catedra de Calculatoare

Personalized information retrieval

Students:

Iulia Baluta

Alexandru Gorgoi

Mihnea Donciu

Madalina Ionita

Contents

- **Introduction**
  - Learning user's preferences
  - Methods of personalization
  - Evaluation
  - Problems
- Personalized Search
- Collaborative Search
- Personalized Social Search
- Task-based personalized search

Why personalized search?

- A typical search engine returns the same result per query for all users
- User queries are in general very short and provide an incomplete specification of individual users' information needs
- E.g.: "canon book" can refer to religion, photography, literature or music
- Studies showed that more than 80% of people would prefer personalized search results to more general results

Gathering users' preferences

- Modeling the users' preferences and interests
  - tracking and aggregating users' interaction with the system
- Types:
  - users' previous queries
  - click-through analysis
  - eye-tracking during the search session
  - relationship to other users
- Creating an user profile

Search Personalization Techniques

- Personalized query expansion
  - adding new terms to the query
  - reweighting the original query terms based on the user profile
- Adjusting search results according with users' interests:
  - Re-ranking
  - Filtering

Evaluation Techniques

- User explicit feedback – user studies
- Users' implicit feedback
- Bookmark-based evaluation
  - For personalized social search
  - User feedback through rating, tagging, commenting, etc.

Personalized problems

- User profiling violates user privacy
- Previous interactions might not be consistent with current needs – different context
- The benefits of personalization vary across queries

Next ?

- Introduction
- **Personalized Search**
  - **Strategies to improve retrieval effectiveness**
  - Ontology-based personalized search
  - Query expansion using desktop data
  - Query expansion using gaze-based feedback
- Collaborative Search
- Personalized Social Search
- Task-based personalized search

Strategy to improve retrieval effectiveness (1)

- Model and gather user's search history
- Construct a user profile based on the history and a general profile based on the Open Document Project category hierarchy
- Deduce appropriate categories for each user query based on the user and general profiles
- Improve Web search effectiveness by using these categories as a context for each query

Strategy to improve retrieval effectiveness (2)

- User search history:
  - queries + relevant documents + categories
  - Tree model of search records: root = query, with one or mode categories as children, each category is parent node of the corresponding documents
- User profile:
  - Set of categories of terms with associated weights
- Matrix representation:
  - *DT* (Document-Term - a) and *DC* (Document-Category -b)
  - *M* (user profile matrix - c) is constructed from DT and DC

Strategy to improve retrieval effectiveness (3)

Doc\Term	apple	recipe	pudding	football	soccer	fifa
D1	1	0	0	0	0	0
D2	0.58	0.58	0.58	0	0	0
D3	0	0	0	1	0	0
D4	0	0	0	0.58	0.58	0.58

(a)

Doc\Category	COOKING	SOCCER
D1	1	0
D2	1	0
D3	0	1
D4	0	1

(b)

Cate\Term	apple	recipe	pudding	football	soccer	fifa
COOKING	1	0.37	0.37	0	0	0
SOCCER	0	0	0	1	0.37	0.37

(c)

- Category hierarchy: general knowledge of our system extracted from *ODP*
- Inference of user's search intention:
  - interests may change in time
  - the most recent search records are considered

User profile learning algorithms

- Computing matrix *M* from *DT* and *DC*
- Linear Least Square Fit (*LLSF* or *pLLSF*): Singular Value Decomposition of DT:

$$M = DC^T * U * \sum^+ * V^T$$

- Rocchio based:
  - batch (*bRocchio*): $M(i,j) = \frac{1}{N_i} \sum_{k=1}^{m} DT(k,j) * DC(k,i)$
  - adaptive (*aRocchio*):

$$M(i,j)^t = \frac{N_i^{t-1}}{N_i^t} M(i,j)^{t-1} + \frac{1}{N_i^t} \sum_k DT(k,j) * DC(k,i)$$

- *kNN*: k-Nearest Neighbour computes similarity between the query and each category for the k most similar documents of DT: $Sim(q, c_j) = \sum_{d_i \in kNN} Cos(q, d_i) * DC(i,j)$

Mapping queries to categories

- Using user profile only (baseline)
- Using general profile only (baseline)
- Combining the 2 profiles (3 methods)
- Experimental results:
  - Accuracy: pLLSF, kNN and bRocchio were the best
  - Performance: the 3 combining methods outperform the 2 baselines
  - Adaptivity: accuracy increases with then size of input data, going to 100% for all training data
- 99% of the time of processing a query is spent to retrieve documents and extract lists of documents from the result pages, 1% of the time is spent to map the queries to specific categories and to merge the lists into a final list of docs -> the algorithms are very efficient

Next ?

- Introduction
- **Personalized Search**
  - Strategies to improve retrieval effectiveness
  - **Ontology-based personalized search**
  - Query expansion using desktop data
  - Query expansion using gaze-based feedback
- Collaborative Search
- Personalized Social Search
- Task-based personalized search

Ontology Based Personalized Search

- A user profile is created over time by analyzing surfed pages
  - associating the content with the length of the document and the time that was spent on it
- Ontology node – browsing hierarchy node – a set of documents (content)
- Documents/superdocuments are represented as weighted keyword vectors using the vector space model

User profiles

- Hierarchically structured, generated automatically, dynamic
- The files in a web browser's cache folder are periodically characterized
- The strength of match is combined with the length of the page and the time spent on that page
- User profile convergence – the number of nodes with non zero interest values coverges over time

Approaches

- Re-ranking – applying a function to the ranking numbers returned by a search engine referring to the user profiles
- Eleven point precision average evaluates ranking performance in terms of *recall* and *precision*
- *Filtering* - comparing the documents to a list of keywords that describe a user or a set of documents that the user previously judged relevant or irrelevant
- Query expansion – expanding the query with the user's interests (very difficult)

Results

- Cache folders analyzed
- Re-ranking – performance increases of up to 8%
- The length of a surfed page can be neglected when the interest in a page is inferred (time spent on that page matters more)

Next ?

- Introduction
- **Personalized Search**
  - Strategies to improve retrieval effectiveness
  - Ontology-based personalized search
  - **Query expansion using desktop data**
  - Query expansion using gaze-based feedback
- Collaborative Search
- Personalized Social Search
- Task-based personalized search

Personalized Query Expansion for the Web

- Improve Web search queries by expanding them with terms collected from each user's Personalized Information Repository(PIR)
- Generating additional query keywords by analizing user data at increasing granularity levels
- Adapting query expansion to various features of each query

Personal Information Repository

- PIR is also referred to as “Desktop”
- User’s personal collection of text documents, emails, cached web pages etc.
- All profile information is stored locally, which gives total privacy
- Web queries will be expanded with keywords extracted from user’s PIR

Query expansion using Desktop data

- Algorithms:
  - *Expanding with Local Desktop Analysis* related to expansion keywords from the PIR best hits, with 3 granularity levels: Term and Document Frequency (TF/DF); Lexical Compounds (LC); Sentence Selection (SS)
  - *Expanding with Global Desktop Analysis* relies on information from all the personal Desktop, with 2 techniques: Term Co-occurrence Statistics; Thesaurus Based Expansion
- Experiments:
  - Google vs TF/DF, LC, SS, TC, WordNet expansion
  - TF and LC produced improvements over regular search
  - All were better than Google on ambiguous queries

Introducing adaptivity

- An optimal personalized query expansion algorithm should adapt to the user's particularities
- Adaptivity factors:
  - *Query clarity*, with metrics of: length, scope, divergence
  - *Query formulation process* by adding terms
- Experiments showed that the adaptive algorithms performed as least as well as Google
- For random queries, results are worse than for the static techniques

Next?

- Introduction
- **Personalized Search**
  - Strategies to improve retrieval effectiveness
  - Ontology-based personalized search
  - Query expansion using desktop data
  - **Query expansion using gaze-based feedback**
- Collaborative Search
- Personalized Social Search
- Task-based personalized search

Query Expansion Using Gaze-Based Feedback on the Subdocument Level

- Incorporating eye tracker information – keep track of document parts the user read in some way
- Fixation (200ms) and saccades – document metadata
- Identifying the precise query context by analyzing at what document parts the user looked immediately before issuing the query

Extracting query-expansion terms

- Methods:
  - Baseline - uses TF×IDF on the entire document and extracts the highest scoring terms.
  - *Gaze-Filter* - applies the score calculation of the baseline method (TF×IDF) only on gaze-annotated document parts
  - Gaze-Length-Filter - ignores all not gaze-annotated document parts and calculates an *interest* score for every viewed term *t*
  - Reading-Speed

Results

- Considering additional information like reading speed and coherence has a great deal of impact
- Additional information like reading speed and coherence does not seem to have high impact
- Compared to other methods for relevance feedback on the subdocument level, gaze-based feedback seems to be sufficiently precise, even though eye trackers are expensive

Next ?

- Introduction
- Personalized Search
- **Collaborative Search**
- Personalized Social Search
- Task-based personalized search

Collaborative Search

- Passive collaboration
  - Collaborative filtering – data from similar people can be used to personalize search
  - Use click-through data from users in the same "search comunity" to enhance search results
- Collaborative searching
  - Users search „together" - for a common task

Discovering and Using Groups to Improve Personalized Search

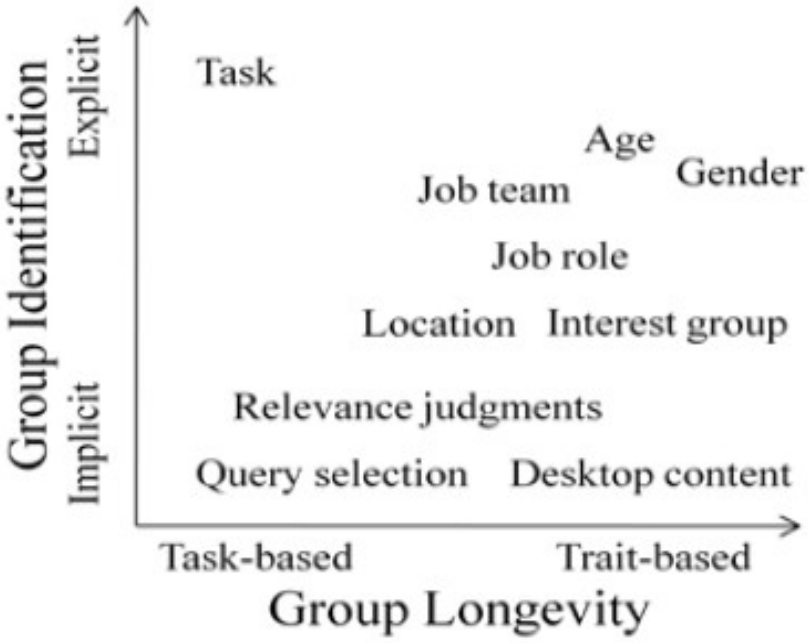
- Augmenting an individual profile by using data from other people
- 2 grouping dimensions
  - Group longevity
  - How explicitly the group is formed
- Algorithm for groupization
  - Aggregates personalization scores from different group members

Group types

- Trait-based groups
  - Shared interests
  - Occupation
  - Geography
  - Demographics
- Explicit identification
  - Explicit task-based

collaboration

  - Self-reported informatio
- Implicit identification
  - Similar desktop indices
  - Similar queries or relevance judgements



User study

- Both task-based groups and trait-based groups
- Data collection
  - Pre-cached search results listed randomly
  - Measure relevance of the result for the query: highly relevant, relevant, not relevant
- Query selection
  - Pre-generated queries
  - Email questionnaire – ask group members for query generation and have other groups measure the relevance of the results

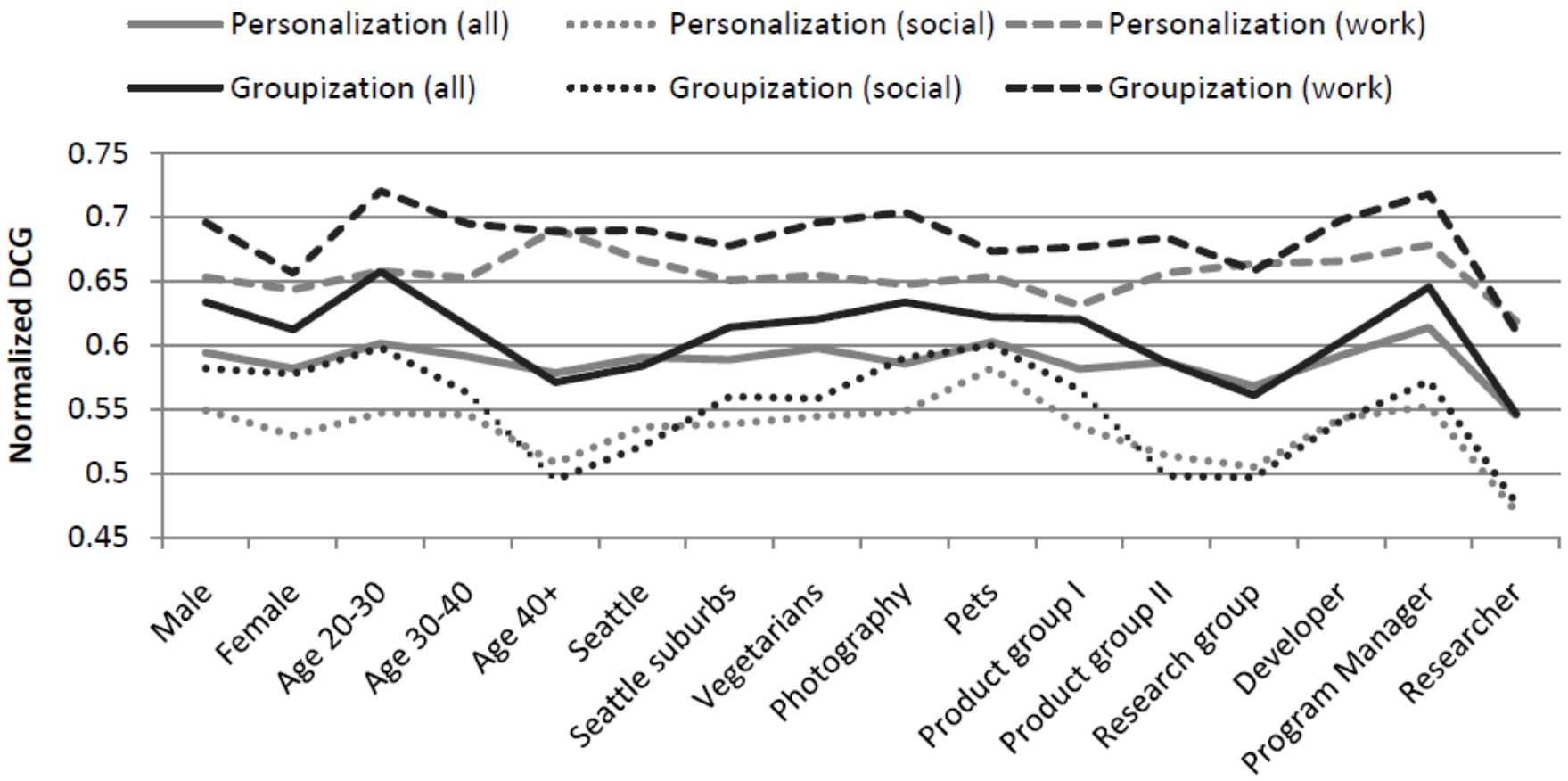
Variations within groups

- Query selection
  - Correlation to group membership
  - People with similar queries do not have similar relevance judgments
- User profile
  - Term vectors
  - Index similarity – for common queries
    - Useful to identify membership in explicit groups
    - Not correlated to relevance judgments
- Relevance judgments
  - Would members of a group benefit from common ranking?
  - Best for task-based groups

Groupization

- On top of an existing Web search personalization system
- Groupized score = sum of personalized score of each member
- Improved results for some explicit groups: task-based, interest-based and occupational
- Best for group related queries
- Might be a challenge to apply it to implicitly identified groups

Groupization performance



Next ?

- Introduction
- Personalized Search
- Collaborative Search
- **Personalized Social Search**
- Task-based personalized search

Social search

- Using data from Web 2.0 applications to improve search engines
- Relationships between entities in a social network such as users, documents and tags can be used to expand the knowledge about user's preferences
- Rich feedback from the user – by tagging, rating and commenting

Personalized Social Search based on User's Social Network

- Re-ranking based on relation to individuals in the social network
- 3 types of social networks
  - Familiarity-based network - people related to the user through explicit familiarity connection
  - Similarity-based network - people “similar” to the user as reflected by their social activity
  - Overall network – both relationship types
- Comparison with Topic-based personalization – based on user's related terms, aggregated from several social applications

Personalized social search

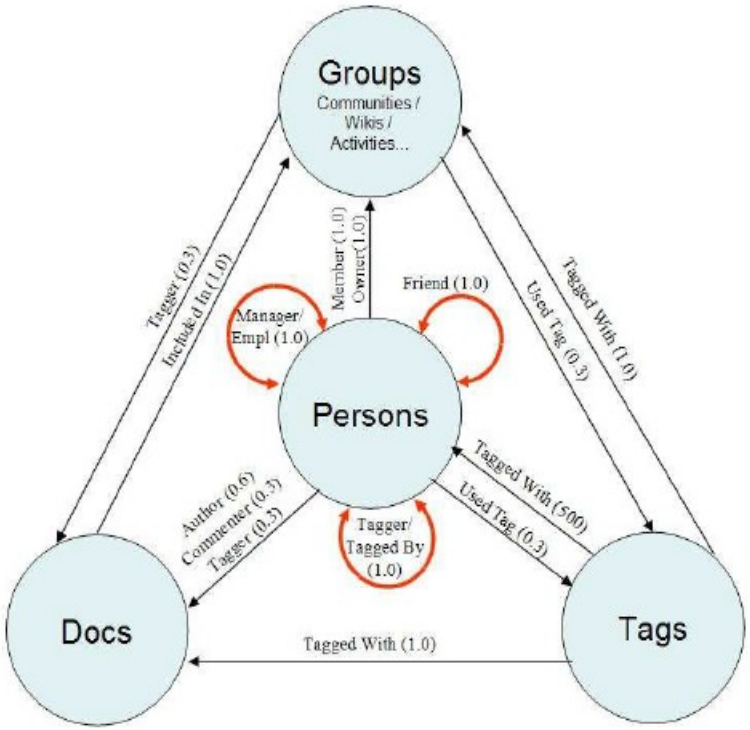
- Social search – search over “social” data gathered from Web 2.0 applications
  - Social bookmarking systems
  - Wikis
  - Blocks
  - Forums
  - Social Network Sites (SNS)
- User profile derived from user feedback (bookmarking, rating, commenting) – a very good indicator or user's interests
- Obtain user's preferences as inferred from user's related people → re-rank results

Social Network and Discovery - SaND

- An aggregation tool for information discovery and analysis over social data
- Leverages complex relationships between content, people and tags
- Builds an entity-entity

relationship matrix

  - Direct relations
  - Indirect relations



Search personalization

- A user profile is constructed on the fly at login
- For a user u SaND retrieves
  - N(u) – the ranked list of users related to u
  - T(u) – the ranked list of related terms
- User profile: P(u) = (N(u), T(u))
- Search results are re-ranked as follows

$$S_p(q,e|P(u)) = \alpha S_{np}(q,e) + (1-\alpha)[\beta \sum_{v \in N(u)} w(u,v) \cdot w(v,e) + (1-\beta) \sum_{t \in T(u)} w(u,t) \cdot w(t,e)]$$

- $S_p(q,e|P(u))$- the personalized score of entity e to query q, given the profile of user u
- $S_{np}(q,e)$ - the non personalized SaND score of e to q
- w(u,v) – the relationship strength of user/term v to u

Evaluation

- Bookmark-based evaluation vs user survey
- Both studies suggest that all personalization methods outperform non-personalized search
- Off-line study
  - Similarity SN and Topic-based personalization with no SN outperform Familiarity SN and Overall SN
- User survey
  - Overall SN outperforms all other personalized searches
  - All SN-based strategies significantly outperform Topic-based personalization

Next ?

- Introduction
- Personalized Search
- Collaborative Search
- Personalized Social Search
- **Task-based personalized search**

Task-aware search personalization

- User interests change over time
- Users can have various tasks within a short timespan
- History-based personalization may impede a user's desire of discovering new topics
- A proposed solution
  - Use history-based personalization only when relevant
  - Hierarchical clustering of the user's profile → a history of user's tasks
  - Tasks range from very specific, short-term tasks to general interests

Task-based personalization

- Obtain candidate query facets representing the different aspects a query might span → obtained by clustering query results
- Retrieve top-k tasks most similar to the user query from the user's profile
- Include a task representing the current active session
- Represent query facets and tasks by a unigram language model

Task-based personalization(2)

- Determine the task/facet pair with the lowest Kullback-Leibler (KL) divergence
- If KL divergence for that pair is larger than a threshold *a* → previously unexplored task
- Update the query representation with terms that best discriminate the chosen query facet, while being most similar to the chosen task
- Re-rank the original result based on the KL divergence between their title/query representation and the new query representation

Task Language Model

- $P(w|T) = a\ P(w|Q) + (1-a)\ P(w|B)$
  - B – average of individual browsed documents' language models
  - Q – uniform mixtures of the task's query chains
- Query chain – a weighted sum of its constituent queries – later queries are more important
- A single query - a mixture of its query terms
- Last visited documents in a search session are given a higher weight in the clickstream language model

Results

- User survey
- NDCG (Normalized Discounted Cumulative Gain) - for measuring the ranking quality
- Generating query facets by
  - Human labels
  - Automatic hierarchical clustering
- Selective personalization outperforms both original Google ranking and the enforced personalization

The Roles of Task Stage and Task Type

- Multi-session work tasks are often complex and consist of multiple sub-tasks (parallel or independent)
- Factors of task type and task stage can be helpful in predicting the usefulness of a document
- Eg: at the beginning of their tasks, it is less likely to start initial queries by introducing all the search terms, more synonyms and parallel terms

Metrics

- Dwell time – time between opening a document and switching or closing it
- Display time – time between opening a document and closing it
- Decision time – equivalent to the first dwell time
- Dwell time was able to predict usefulness in both tasks combined, and for both the parallel and dependent tasks
- Display time can predict usefulness in parallel and dependent tasks separately, but not in both combined
- Decision time can only predict usefulness in the dependent task

Results

- Total display time and total dwell time can be used for personalizing search for subsequent sessions in multi-session tasks
- Task stage and task type information does not necessarily need to be considered.
- Decision time can be used for personalizing search for an ongoing session as well as for subsequent sessions

Current Trends in Personalized Information Retrieval

- Contextual IR
- Collaborative filtering
- Using agents and information scent
- Combine Adaptive Hypermedia and IR approaches to deliver personalised information seeking and access
- A unified method of evaluating the personalized IR systems

Conclusions

- Search personalization significantly improves the relevance of the results
- The main challenge of personalizing search consists in accurately building an user profile
- There are three main approaches in search personalization : re-ranking, filtering and query expansion
- Creating context-aware and adaptive user profiles is currently the main topic of research