4. Personalization
Outline

4.1. Objectives
4.2. Concerns
4.3. Potential
4.4. Link Analysis
4.5. Query Expansion
4.6. Retrieval Model
4.7. Re-Ranking
1. Objectives

- Our focus will be on web search; personalization also affects other applications (e.g., recommender systems, advertising)

- Personalization can serve **different objectives** in web search
  - **disambiguate** the query based on user profile (e.g., jaguar)
  - **adapt** query results to the user profile or abilities (e.g., reading level)
  - **localize** results based on the user location (e.g., uds, coffee shop)
Search results can be personalized using **different data sources**

- **Feedback** (e.g., about relevance of search results)
- **Traits** (e.g., age, gender, income level, education level, religion)
- **Social profiles** (e.g., likes on facebook, tweets)
- **Behavior** (e.g., short/long-time browsing, search, and click histories)
- **Desktop** (e.g., office documents, e-mail)
Search results can be personalized in different locations [12]

- **Server:** the search engine knows the user profile and personalizes the search result according to it

- **Client:** only the client knows the user profile and personalizes the generic result from the search engine according to it

- **Client-Server Cooperation:** the client knows the user profile and reveals parts of it to the search engine to personalize the result
Search results can be personalized in different locations [12]

- **Server**: the search engine knows the user profile and personalizes the search result according to it
- **Client**: only the client knows the user profile and personalizes the generic result from the search engine according to it
- **Client-Server Cooperation**: the client knows the user profile and reveals parts of it to the search engine to personalize the result
Search results can be personalized in **different locations** [12]

- **Server**: the search engine knows the user profile and personalizes the search result according to it
- **Client**: only the client knows the user profile and personalizes the generic result from the search engine according to it
- **Client-Server Cooperation**: the client knows the user profile and reveals parts of it to the search engine to personalize the result
Search results can be personalized in **different locations** [12]

- **Server**: the search engine knows the user profile and personalizes the search result according to it
- **Client**: only the client knows the user profile and personalizes the generic result from the search engine according to it
- **Client-Server Cooperation**: the client knows the user profile and reveals parts of it to the search engine to personalize the result
Search results can be personalized in different locations [12]

- **Server**: the search engine knows the user profile and personalizes the search result according to it
- **Client**: only the client knows the user profile and personalizes the generic result from the search engine according to it

**Client-Server Cooperation**: the client knows the user profile and reveals parts of it to the search engine to personalize the result
Search results can be personalized using different methods

- **Link analysis:** by computing a *user-specific static score* for each web page, reflecting its importance relative to the user profile.

- **Query expansion:** by *augmenting the query* with terms from the user profile to disambiguate it and inform the search engine.

- **Retrieval model:** by *directly considering* the user profile when deciding which documents to return as results and how to order them.

- **Re-ranking:** by considering the *generic results* returned by the search engine and *re-ranking* them considering the user profile.
2. Concerns

- Personalization of search results requires **data about the user**
  - **personal traits** (e.g., gender, age, income level)
  - search, click, or browsing histories

- **Privacy is a concern** in the post-Snowden era

- Personalization of search results **can affect users and society**
  - by not exposing users to **views different from their own**
  - by only showing results fitting the **user’s interests, location, intellect**

- **Filter bubble** is a concern regarding the effects of personalization
Shen et al. [10] study the **tension** between **privacy preservation** and **personalization** and define four levels of privacy protection:

- **Level 1: Pseudo Identity**
  (user identity is replaced by an identifier in the search system)

- **Level 2: Group Identity**
  (multiple users share a single user identifier in the search system)

- **Level 3: No Identity**
  (search system does not know the user identity)

- **Level 4: No Personal Information**
  (search system does not know any personal information)
How Much Do They Know?

"You have zero privacy anyway. Get over it."

(Scott McNealy, former CEO of Sun Microsystems)

- Bi et al. [1] examine to what extent a user’s demographics can be inferred purely based on the search queries she issues.

- myPersonality.org data provides the Facebook likes of millions of anonymous users together with their demographic profiles.

- Open Directory Project (DMOZ.org) as common representation for liked entities on Facebook and queries issued by users.
How Much Do They Know?

- Bing users as probability distributions over ODP topics
- Probability distributions over ODP topics for traits from Facebook

**Results**: 

- **AUC (Area Under receiver operating characteristic Curve)**
  - **0.803** for predicting gender based on queries issued
  - **0.735** for predicting age based on queries issued
Filter Bubble

- Eli Pariser [9] coined the notion “filter bubble”, observing that personalization traps users by increasingly exposing them to content that is in line with what they know or believe.

- **Examples:**
  - Query “egypt” brings up only tourism-related results, but none related to political situation.
  - Query “bp” brings up stock-related results, but none related to oil spill.

[TED talk]
Hannak et al. [4] conducted a study with 200 Google users to measure the **degree of personalization** and identify **personal features** with an impact on search results.

- **120 queries** from Google Zeitgeist and WebMD (tech, news, etc.)
- **200 users** from 43 different U.S. states recruited via **Mechanical Turk**
- **scripted issuing of queries** through **HTTP proxy**

**Observations:**

- **extensive personalization** (at lower ranks)
- **most personalized queries** related to companies/stores (localization)

### Table 1: Top 10 Most/Least Personalized

<table>
<thead>
<tr>
<th>Most Personalized</th>
<th>Least Personalized</th>
</tr>
</thead>
<tbody>
<tr>
<td>gap</td>
<td>what is gout</td>
</tr>
<tr>
<td>hollister</td>
<td>dance with dragons</td>
</tr>
<tr>
<td>hgtv</td>
<td>what is lupus</td>
</tr>
<tr>
<td>boomerang</td>
<td>gila monster facts</td>
</tr>
<tr>
<td>home depot</td>
<td>what is gluten</td>
</tr>
<tr>
<td>greece</td>
<td>ipad 2</td>
</tr>
<tr>
<td>pottery barn</td>
<td>cheri daniels</td>
</tr>
<tr>
<td>human rights</td>
<td>psoriatic arthritis</td>
</tr>
<tr>
<td>h2o</td>
<td>keurig coffee maker</td>
</tr>
<tr>
<td>nike</td>
<td>maytag refrigerator</td>
</tr>
</tbody>
</table>

### Figure 4: Usage of Google services

- **Reader**
- **Google Reader**
- **Calendar**
- **Drive**
- **Gmail**
- **Android**
- **YouTube**

Is the Filter Bubble Real?
Is the Filter Bubble Real?

To identify personal features that impact search results, Hannak et al. [4] created different Google profiles and compared results:

- logged in / not logged in / cookies cleared (little impact)
- browser user-agent (no impact)
- geolocation from IP address (big impact)
- gender (no impact)
- search history (no impact)
- click history (no impact)
- browsing history (no impact)
3. Potential

- **Question**: How much can be gained, in terms of retrieval performance, by personalizing web search results?

- Teevan et al. [11] estimate the potential for personalization (in terms of nDCG) using three kinds of data sources:
  - **explicit relevance feedback** from 125 users on 699 queries (gain value {0, 1, 2} derived from graded relevance judgment)
  - **desktop data** of 59 users as **implicit feedback** on 822 queries (gain value [0, 1] based on cosine similarity to desktop)
  - **click logs** of 1.5 M users as **implicit feedback** on 2.4 M queries (gain value {0, 1} based on whether user clicked on result)
Potential for Personalization

- Given feedback from an individual user, we can determine the optimal result for her and how much worse the web result is.
Given **feedback from an individual user**, we can determine the **optimal result** for her and **how much worse** the web result is.

<table>
<thead>
<tr>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>d₂</td>
</tr>
<tr>
<td>d₁</td>
</tr>
<tr>
<td>d₄</td>
</tr>
<tr>
<td>d₃</td>
</tr>
<tr>
<td>d₅</td>
</tr>
</tbody>
</table>
Given **feedback from an individual user**, we can determine the **optimal result** for her and **how much worse** the web result is.

<table>
<thead>
<tr>
<th>Result (d_i)</th>
<th>Feedback (d_j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_2)</td>
<td>0</td>
</tr>
<tr>
<td>(d_1)</td>
<td>2</td>
</tr>
<tr>
<td>(d_4)</td>
<td>0</td>
</tr>
<tr>
<td>(d_3)</td>
<td>1</td>
</tr>
<tr>
<td>(d_5)</td>
<td>1</td>
</tr>
</tbody>
</table>
Given **feedback from an individual user**, we can determine the **optimal result** for her and **how much worse** the web result is.

<table>
<thead>
<tr>
<th>Result</th>
<th>Feedback</th>
<th>Optimal Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_2$</td>
<td>0</td>
<td>$d_1$</td>
</tr>
<tr>
<td>$d_1$</td>
<td>2</td>
<td>$d_3$</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0</td>
<td>$d_5$</td>
</tr>
<tr>
<td>$d_3$</td>
<td>1</td>
<td>$d_2$</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1</td>
<td>$d_4$</td>
</tr>
</tbody>
</table>
Potential for Personalization

Given feedback from an individual user, we can determine the optimal result for her and how much worse the web result is.

<table>
<thead>
<tr>
<th>Result</th>
<th>Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td>d₂</td>
<td>0</td>
</tr>
<tr>
<td>d₁</td>
<td>2</td>
</tr>
<tr>
<td>d₄</td>
<td>0</td>
</tr>
<tr>
<td>d₃</td>
<td>1</td>
</tr>
<tr>
<td>d₅</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Optimal Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>d₁</td>
</tr>
<tr>
<td>d₃</td>
</tr>
<tr>
<td>d₅</td>
</tr>
<tr>
<td>d₂</td>
</tr>
<tr>
<td>d₄</td>
</tr>
</tbody>
</table>

nDCG: 1.0
Potential for Personalization

Given **feedback from an individual user**, we can determine the **optimal result** for her and **how much worse** the web result is.

<table>
<thead>
<tr>
<th>Result</th>
<th>Feedback</th>
<th>Optimal Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_2$</td>
<td>0</td>
<td>$d_1$</td>
</tr>
<tr>
<td>$d_1$</td>
<td>2</td>
<td>$d_3$</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0</td>
<td>$d_5$</td>
</tr>
<tr>
<td>$d_3$</td>
<td>1</td>
<td>$d_2$</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1</td>
<td>$d_4$</td>
</tr>
</tbody>
</table>

nDCG: 0.79

nDCG: 1.0
Given **feedback from an individual user**, we can determine the **optimal result** for her and **how much worse** the web result is.

<table>
<thead>
<tr>
<th>Result</th>
<th>Feedback</th>
<th>Optimal Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_2)</td>
<td>0</td>
<td>(d_1)</td>
</tr>
<tr>
<td>(d_1)</td>
<td>2</td>
<td>(d_3)</td>
</tr>
<tr>
<td>(d_4)</td>
<td>0</td>
<td>(d_5)</td>
</tr>
<tr>
<td>(d_3)</td>
<td>1</td>
<td>(d_2)</td>
</tr>
<tr>
<td>(d_5)</td>
<td>1</td>
<td>(d_4)</td>
</tr>
</tbody>
</table>

nDCG: 0.79  

Potential for Personalization  
nDCG: 1.0
Explicit relevance feedback
- Personalized result (nDCG 1.0)
- Result for group of six (nDCG 0.85)
- Web result (nDCG 0.58)

Potential for personalization
- smallest for click logs (behavior)
- largest for desktop data (content)
Potential for Personalization

- Mei and Church [7] make use of information theory to estimate how hard web search is and how much personalization helps

  Query (e.g., fb), URL (e.g., http://www.fb.com), IP (e.g., 139.19.54.9)

- Data: Click log from the Microsoft Live search engine (now: Bing)
  - 18 months (until July 2007)
  - 193 M unique IP addresses (users)
  - 637 M unique queries
  - 585 M unique URLs
Entropy measures the degree of uncertainty of a random variable \( X \), thereby characterizing the size of the search space.

\[
H(X) = - \sum_x P(x) \log P(x)
\]

**Example**: Dice with six faces having uniform probability

\[
H(D) \approx 2.58 \quad \text{Size of search space: 6}
\]

**Example**: Dice with six faces; 1 has probability 0.8; others 0.04

\[
H(D) \approx 1.19 \quad \text{Size of search space: 2.28}
\]
Conditional Entropy

- **Conditional entropy** measures the **remaining uncertainty** of a random variable $X$ given the value of another random variable $Y$

$$H(X|Y) = H(X,Y) - H(Y)$$

- **Example**: Dice with six colored faces having uniform probability

$$\begin{array}{c}
1 & 2 & 3 & 4 & 5 & 6 \\
\end{array}$$

Consider $N = \{\text{even, odd}\}$ and $C = \{\text{black, white}\}$

$$H(N) = 1 \quad H(C) \approx 0.92$$

$$H(N,C) \approx 1.46 \quad H(N|C) \approx 0.54$$
How Hard is Web Search?

Given a click log, one can now estimate **how hard search** is as

\[ H(\text{URL} | \text{Query}) \]

Mei and Church [7] observe the following (conditional) entropies

\[ H(\text{URL} | \text{Query}) \approx 3.5 \]

\[ H(\text{URL, Query}) \approx 26.41 \quad H(\text{Query}) \approx 22.94 \]
How Much does Personalization Help?

- Assuming that IPs correspond to individuals, we can estimate how much easier search becomes once the IP is known.

  \[ H(\text{URL}|\text{Query}, IP) \approx 1.26 \]

  \[ H(\text{URL}, \text{Query}, IP) \approx 31.67 \quad H(\text{Query}, IP) \approx 30.41 \]

- Personalization reduces the size of the search space from about 11.31 to 2.39 (reflecting how many results users typically have to inspect).
4. Link Analysis

- Search results can be personalized by computing a **user-specific static score** for every web page that reflects its **importance** relative to the user profile.

- **Recap**: PageRank (as part of the original Google search engine) operates on the **web graph** $G(V, E)$ consisting of **web pages** ($V$) and **hyperlinks** ($E$)

  $r(v) = (1 - \epsilon) \sum_{(u,v) \in E} \frac{r(u)}{\text{out}(u)} + \frac{\epsilon}{|V|}$

- PageRank models a **random surfer** who **follows random hyperlink** with probability $(1 - \epsilon)$ and **jumps to random web page** with probability $\epsilon$
PageRank scores correspond to the stationary state probabilities of an ergodic Markov chain with transition probability matrix $P$

$$P = (1 - \epsilon) T + \epsilon J$$

with matrix $T$ capturing hyperlink following as

$$T_{ij} = \begin{cases} 
1/\text{out}(i) & : (i, j) \in E \\
0 & : \text{otherwise}
\end{cases}$$

and matrix $J$ capturing random jumps as

$$J = [1 \ldots 1]^T \times j$$

with random jump vector $j$ as

$$j = [1/|V| \ldots 1/|V|]$$
Power-Iteration Method

- **Power-iteration method** to compute PageRank vectors
  - initialize
  - repeat
  - until convergence

\[
\pi^{(0)} = \left[\frac{1}{|V|} \ldots \frac{1}{|V|}\right]
\]

\[
\pi^{(i)} = \pi^{(i-1)} \times P
\]

\[
|\pi^{(i)} - \pi^{(i-1)}| < \delta
\]
Personalized PageRank

- Haveliwala [5] proposed a **topic-specific variant of PageRank** that performs **random jumps only to on-topic web pages**

- Let $C \subseteq V$ be the web pages belonging to **topic** $C$ (e.g., Sports), the random jump vector $j$ is defined as

$$j_i = \begin{cases} 
1/|C| & : \ i \in C \\
0 & : \ otherwise 
\end{cases}$$

- Web pages “closer” to on-topic web pages in $C$ are favored

- **Personalized PageRank** considers a set of **user-specific favorite web pages** $F$ as random jump targets
Personalized PageRank

Computing and storing personalized PageRank scores for large numbers of users and/or web pages is prohibitive.


Let $j$ and $j'$ be two random jump vectors and $\pi$ and $\pi'$ be the two corresponding PageRank vectors, then

$$ (\alpha \pi + \beta \pi') = (\alpha \pi + \beta \pi') \times ((1 - \epsilon) T + \epsilon [1 \ldots 1]^T \times (\alpha j + \beta j')) $$

One can thus select a small set of basis vectors, compute the corresponding PageRank vectors, and obtain user-specific PageRank scores as a linear combination of them.
Chirita et al. [2] personalize search results by augmenting the query with **terms selected from the user’s desktop**

**Local Desktop Analysis** issues the query locally against the user’s desktop search engine and extracts terms from top-$k$ pseudo-relevant documents, e.g., based on

- **term frequency** ($tf$) or **document frequency** ($df$) (but not: $tf.idf$)
- **dispersion analysis** (most frequent compounds: adjective? noun+)
Query Expansion

- **Global Desktop Analysis** precomputes term co-occurrence scores by analyzing documents from the user’s desktop.

- **Cosine similarity**
  \[
  \text{score}(a, b) = \frac{df(a \land b)}{\sqrt{df(a) \cdot df(b)}}
  \]

- **Mutual information**
  \[
  \text{score}(a, b) = \log \frac{|D| \cdot df(a \land b)}{df(a) \cdot df(b)}
  \]

- **Expansion terms** for a query \( q \) are then determined as those having the highest aggregated score:
  \[
  \text{agg\_score}(e) = \prod_{v \in q} \text{score}(v, e)
  \]

- Experiments show significant improvement over baseline (Google) for ambiguous queries; but deterioration for clear queries.
6. Retrieval Model

- Xue et al. [12] devise a language modeling approach to personalize results based on what users have viewed.

- Let $V_{i,t}$ be documents that user $i$ has viewed at time $t$, and let $nw$ denote the current time period (e.g., day).

- **Short-term profile** for user $i$ is estimated based on what the user has viewed within the last time period:

\[
P \left[ v \mid \theta_{i}^{st} \right] = \frac{\sum_{d \in V_{i,nw}} tf(v, d)}{\sum_{d \in V_{i,nw}} |d|}
\]
User Model

- **Long-term profile** for user $i$ is estimated based on what the user has viewed **within the last $h$ time periods**

$$
P \left[ v \mid \theta^l_{i} \right] = \frac{\sum_{t=1}^{h} \sum_{d \in V_{i,nw-t}} tf(v,d) \cdot e^{-\rho t}}{\sum_{t=1}^{h} \sum_{d \in V_{i,nw-t}} |d| \cdot e^{-\rho t}}$$

applying **exponential temporal decay** to give **lower weight** to what has been **viewed longer ago**

- **User language model** is then estimated as

$$
P \left[ v \mid \theta_i \right] = \beta \ P \left[ w \mid \theta^{st}_{i} \right] + (1 - \beta) \ P \left[ w \mid \theta^{lt}_{i} \right]$$
Global Model

- **Global language model** for all users is obtained as

\[ P[v | \theta_g] = \frac{1}{|U|} \sum_{i \in U} P[v | \theta_i] \]

with \( U \) as the set of **all users**
Group Model

- Users are grouped into clusters $c_1, \ldots, c_k$ based on the similarity of their user language models (e.g., using $k$-means with KLD).

- **Cluster language model** for cluster $c$ is estimated as

  $$P[v | \theta_c] = \frac{1}{|c|} \sum_{i \in c} P[v | \theta_i]$$

- For query $q$ issued by user $i$ identify a single cluster $c$ as

  $$\arg\min_c (\zeta \ KL(\theta_i \| \theta_c) + (1 - \zeta) \ KL(\theta_q \| \theta_c))$$

  and parameter $\zeta$ controlling fit of cluster to user and/or query.
Combining the Models

- **Combined language model** to rank documents is estimated as

\[
P[v | \theta] = \lambda P[v | \theta_q] + (1 - \lambda) \left[ \gamma P[v | \theta_i] + (1 - \gamma) \left[ \eta P[v | \theta_c] + (1 - \eta) P[v | \theta_g] \right] \right]
\]

with **smoothing parameters** \(\lambda, \gamma, \eta\) controlling the influence of the **query, user, group, and global model**

- Experiments based on **click-through data** from 1,000 users of MSN search engine (now: Bing) and 50/50 split of queries

### Table III. Performance on Test70 Using Different Personalized Schemas for Web Pages Ranking

<table>
<thead>
<tr>
<th>Model</th>
<th>NDCG1</th>
<th>NDCG5</th>
<th>NDCG10</th>
<th>NDCG20</th>
<th>NDCG30</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q)</td>
<td>0.422</td>
<td>0.434</td>
<td>0.441</td>
<td>0.416</td>
<td>0.384</td>
</tr>
<tr>
<td>(q + i)</td>
<td>0.664</td>
<td>0.655</td>
<td>0.613</td>
<td>0.535</td>
<td>0.467</td>
</tr>
<tr>
<td>(q + c)</td>
<td>0.724</td>
<td>0.674</td>
<td>0.635</td>
<td>0.515</td>
<td>0.438</td>
</tr>
<tr>
<td>(q + g)</td>
<td>0.672</td>
<td>0.667</td>
<td>0.626</td>
<td>0.546</td>
<td>0.497</td>
</tr>
<tr>
<td>(q + i + g)</td>
<td>0.707</td>
<td>0.674</td>
<td>0.641</td>
<td>0.556</td>
<td>0.474</td>
</tr>
<tr>
<td>(q + i + c)</td>
<td>0.712</td>
<td>0.675</td>
<td>0.64</td>
<td><strong>0.557</strong></td>
<td>0.474</td>
</tr>
<tr>
<td>(q + i + c + g)</td>
<td><strong>0.724</strong></td>
<td>0.683</td>
<td><strong>0.644</strong></td>
<td>0.555</td>
<td><strong>0.499</strong></td>
</tr>
</tbody>
</table>

### Table IV. Performance on Test30 Using Different Personalized Schemas for Web Pages Ranking

<table>
<thead>
<tr>
<th>Model</th>
<th>NDCG1</th>
<th>NDCG5</th>
<th>NDCG10</th>
<th>NDCG20</th>
<th>NDCG30</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q)</td>
<td>0.462</td>
<td>0.416</td>
<td>0.409</td>
<td>0.391</td>
<td>0.375</td>
</tr>
<tr>
<td>(q + i)</td>
<td>0.622</td>
<td>0.619</td>
<td>0.587</td>
<td>0.513</td>
<td>0.463</td>
</tr>
<tr>
<td>(q + c)</td>
<td>0.663</td>
<td>0.621</td>
<td>0.573</td>
<td>0.509</td>
<td>0.451</td>
</tr>
<tr>
<td>(q + g)</td>
<td>0.654</td>
<td>0.6</td>
<td>0.562</td>
<td>0.515</td>
<td>0.45</td>
</tr>
<tr>
<td>(q + i + g)</td>
<td>0.663</td>
<td>0.617</td>
<td>0.577</td>
<td>0.507</td>
<td>0.451</td>
</tr>
<tr>
<td>(q + i + c)</td>
<td>0.663</td>
<td>0.621</td>
<td>0.582</td>
<td>0.509</td>
<td>0.448</td>
</tr>
<tr>
<td>(q + i + c + g)</td>
<td>0.673</td>
<td>0.625</td>
<td>0.592</td>
<td>0.52</td>
<td>0.472</td>
</tr>
</tbody>
</table>
7. Re-Ranking

Matthijs and Radlinski [7] develop a browser plug-in that builds a (local) user profile which is then used to re-rank Google search results based on the information in their snippets.

User profile based on viewed web pages includes:

- unigrams from full-text (body) and title
- unigrams from meta-data fields (description and keywords)
- extracted keywords and noun phrases

For each term $v$ in the user profile, a tf.idf weight $w_{\text{tf.idf}}(v)$ is estimated with a document frequency from Google.
Re-Ranking

- Given a query, the **search results returned by Google are re-ranked** taking into account the following factors
  - **matching score** between **search result title** and **user profile**
    \[
    \text{score}_M(r) = \prod_{v \in \text{title}(r)} \log \frac{w_{tfidf}(v) + 1}{\sum_{v'} w_{tfidf}(v')}
    \]
  - **original rank** in Google result (logarithmically damped)
    \[
    \text{score}_R(r) = \frac{1}{1 + \log(\text{rank}(r))}
    \]
  - **number of previous visits** to the URL
    \[
    \text{score}_V(r) = (1 + \alpha \cdot \text{visits}(r))
    \]
    with tunable parameter \( \alpha \)
Re-Ranking

- Re-ranking Google top-50 results based on
  \[ \text{score}_M(r) \times \text{score}_R(r) \times \text{score}_V(r) \]
  improved nDCG from 0.502 to 0.573 (14%) in a user study with six users and 72 queries

- While relatively simple the approach yields a significant improvement (p = 0.042) and can be implemented locally (i.e., without disclosing personal information)
Summary

- Search results are personalized to **resolve ambiguity**, **localize** them, or **adapt them** to the user’s traits or interests.
- Personalization can be achieved by leveraging **different data sources** including users traits, social media profiles, desktop.
- **Privacy and filter bubble effects** are **serious concerns** regarding personalized search – with differing opinions.
- **Potential impact of personalization** can be assessed through user studies or by observing their behavior at large scale.
- Personalization of search results can be achieved using **different methods** including link analysis, retrieval models, and re-ranking.
References


References


[12] **G.-R. Xue, J. Han, Y. Yu:** *User Language Models for Collaborative Personalized Search*, ACM TOIS 27(2), 2009