Internet Mathematics Vol. 6, No. 3: 399–435

Criteria for Cluster-Based Personalized Search

Hyun Chul Lee and Allan Borodin

Abstract. We study personalized web-ranking algorithms based on the existence of document clusterings. Motivated by topic-sensitive page ranking [Haveliwala 03], we develop and implement an efficient "local-cluster" algorithm by extending the web search algorithm of [Achilioptas et al. 01]. We propose some formal criteria for evaluating such personalized ranking algorithms and provide some preliminary experiments in support of our analysis. Both theoretically and experimentally, our algorithm differs significantly from Topic-Sensitive PageRank.

1. Introduction

Due to the size of the current Web and the diversity of user groups using it, the current algorithmic search engines are not completely ideal for dealing with queries generated by a large number of users with different interests and preferences. For instance, it is possible that some users might input the query "Star Wars" with their main topic of interest being "movie" and therefore expecting pages about the popular movie as results of their query. On the other hand, others might input the query "Star Wars" with their main topic of interest being "politics" and therefore expecting pages about proposals for deployment of a missile defense system. (Of course, in this example, the user could easily disambiguate the query by adding, say, "movie" or "missile" to the query terms.)

1542-7951/09 $0.50 per page

To expedite simple searches as well as to try to accommodate more complex searches, *web search-personalization* has recently gained significant attention for handling queries produced by diverse users with very different search intentions. The goal of web-search personalization is to allow the user to expedite web search according to personal search preference or context.

There is no general consensus on exactly what web search personalization means, and moreover, there have been no general criteria for evaluating personalized search algorithms. The goal of this paper is to propose a framework that is general enough to cover many real application scenarios and yet is amenable to analysis with respect to correctness in the spirit of [Achilioptas et al. 01] and with respect to stability properties in the spirit of [Ng et al. 01] and [Lee and Borodin 03] (see also [Borodin et al. 05, Donato et al. 05]). We achieve this goal by assuming that the targeted web service has an underlying cluster structure. Given a set of clusters over the intended documents in which we want to perform personalized search, our framework assumes that a user's preference is represented as a preference vector over these clusters. A user's preference over clusters can be collected either online or offline using various techniques [Qiu and Cho 06, Chirita et al. 05, Teevan et al. 05, Ferragina and Gulli 05].

We do not address how to collect the user's search preferences, but we simply assume that such preferences (possibly with respect to various search features) are already available and can be translated into search preferences over given cluster structures of targeted documents.

We define a class of personalized search algorithms called "local-cluster" algorithms that compute each page's ranking with respect to each cluster containing the page rather than with respect to every cluster. We propose a specific local-cluster algorithm by extending the approach taken in [Achilioptas et al. 01]. Our proposed local-cluster algorithm considers linkage structure and content generation of cluster structures to produce a ranking of the underlying clusters with respect to a user's given search query and preference. The rank of each document is then obtained through the relation of the given document with respect to its relevant clusters and the respective preference of these clusters.

Our algorithm is particularly suitable for equipping already existing web services with a personalized search capability without affecting the original ranking system.

Our framework allows us to propose a set of evaluation criteria for personalized search algorithms. We observe that Topic-Sensitive PageRank [Haveliwala 03], which is probably the best-known personalized search algorithm in the literature, is not a local-cluster algorithm and does not satisfy some of the criteria that we propose. In contrast, we show that our local-cluster algorithm satisfies the suggested properties.

Our main contributions are the following:

- We define a personalized search algorithm that provides a more practical implementation of the web search model and algorithm proposed in [Achilioptas et al. 01].
- We propose some formal criteria for evaluating personalized search algorithms and then compare our proposed algorithm and the Topic-Sensitive PageRank algorithm based on such formal criteria.
- We experimentally evaluate the performance of our proposed algorithm against that of the Topic-Sensitive PageRank algorithm.

2. Motivation

We believe that our assumption that the web service to be personalized admits cluster structures is well justified. For example, we mention the following:

- *Human-generated web directories*: In web sites like Yahoo! and Open Directory Project (DMOZ), web pages are classified into human-edited categories (possibly machine-generated as well) and then organized in a taxonomy. In order to personalize such systems, we can simply take the leaf nodes in any pruning of the taxonomy tree as our clusters.
- *Geographically sensitive search engines*: Sites like Yahoo! Local, Google Local, and Citysearch classify reviews, web pages, and business information of local businesses into different categories and locations (e.g., city level). Therefore, in this particular case, a cluster would correspond to a set of data items or web pages related to the specific geographic location (e.g., web pages about restaurants in Houston, Texas).

We note that the same corpus can admit several cluster structures using different features. For instance, web documents can be clustered according to features such as topic (Yahoo!, DMOZ, Topix, Gnews, About) whether commercially or educationally oriented (Yahoo! Mindset), domain type, language, etc. Our framework allows incorporating various search features into web-search personalization because it works at the abstract level of clustering structures.

3. Preliminaries

Let G_N (or simply G) be a web-page collection (with content and hyperlinks) of node size N, and let q denote a query string represented as a term vector.

Let $\mathcal{C} = \mathcal{C}(G) = \{C_1, \ldots, C_m\}$ be a clustering (not necessarily a partition) for G (i.e., each $x \in G$ is in $C_{i_1} \cap \cdots \cap C_{i_r}$ for some $i_1, \ldots, i_r$). For simplicity we will assume that there is a single clustering of the data but that there are a number of ways that we can extend the development when there are several clusterings.

We define a *cluster-sensitive page-ranking algorithm* μ as a function with values in $[0, 1]$ where $\mu(C_j, x, q)$ will denote the ranking value of page x relative to cluster C_j with respect to query q.[1]

We define a user's preference as a $[0, 1]$-valued function P, where $P(C_j, q)$ denotes the preference of the user for cluster C_j (with respect to query q). We call $(G, \mathcal{C}, \mu, P, q)$ an *instance of personalized search*, that is, a personalized search scenario where there exist a user having a search preference function P over a clustering $\mathcal{C}(G)$, a query q, and a cluster-sensitive page-ranking function μ. Note that either μ or P can be query-independent.

Definition 3.1. Let $(G, \mathcal{C}, \mu, P, q)$ be an instance of personalized search. A *personalized search ranking* (PSR) is a function that maps G_N to an N-dimensional real vector by composing μ and P through a function F; that is,

$$\mathrm{PSR}(x) = F(\mu(C_1, x, q), \ldots, \mu(C_m, x, q), P(C_1, q), \ldots, P(C_m, q)).$$

For instance, F might be defined as a weighted sum of μ and P values.

4. Previous Algorithms

4.1. Modifying the PageRank Algorithm

Due to the popularity of the PageRank algorithm [Brin and Page 98], the first generation of personalized web search algorithms were based on the original PageRank algorithm by manipulating the *teleportation factor* of the PageRank algorithm. In the PageRank algorithm, the rank of a page is determined by the stationary distribution of a modified uniform random walk on the Web graph. Namely, with some small probability $\epsilon > 0$, a user at page i uniformly jumps to a random page, and otherwise with probability $(1 - \epsilon)$ jumps uniformly to one of its neighboring pages.[2] That is, the transition probability matrix is given by $P_\epsilon^A = \epsilon \cdot U + (1 - \epsilon) \cdot A$, where $U = ev^T$ is the teleportation factor matrix with $e = (1, 1, \ldots, 1)$ and v a uniform probability vector defined by $v_i = 1/N$, and $A = (a_{ij})$ with $a_{ij} = 1/\mathrm{outdeg}(i)$ if (i, j) is an edge and zero otherwise. The

[1] Our definition allows and even assumes a ranking value for a page x relative to C_j even if $x \notin C_j$. Most content-based ranking algorithms provide such a ranking, and if not, we can then assume that x has rank value 0.

[2] When page i has no hyperlinks (i.e., $\mathrm{outdeg}(i) = 0$), it is customary to let $\epsilon = 1$.

first generation of personalized web-search algorithms introduced some bias, reflecting the user's search preference, by using nonuniform probabilities on the teleportation factor (i.e., controlling v). Among these, we have Topic-Sensitive PageRank [Haveliwala 03], Modular PageRank [Jeh and Widom 03], and BlockRank [Kamvar et al. 03]. In this paper, we restrict analysis to Topic-Sensitive PageRank.

4.1.1. Topic-Sensitive PageRank. One of the first proposed personalized search-ranking algorithms was *Topic-Sensitive PageRank* [Haveliwala 03]. It computes a topic-sensitive ranking (i.e., cluster-sensitive in our terminology) by constraining the uniform jumping factor of a random surfer to each cluster. That is, when computing the PageRank vector with respect to cluster C_j of pages, we use the personalization vector v^j, where

$$v_i^j = \begin{cases} \frac{1}{|T_j|} & \text{if } i \in C_j, \\ 0 & \text{if } i \notin C_j. \end{cases}$$

More precisely, the page-rank vector with respect to the cluster C_j is then computed as the solution to $\mathrm{TR}(C_j) := (1-\epsilon) \cdot A^T \cdot \mathrm{TR}(C_j) + \epsilon \cdot v^j$. We note that if there exists $y \in C_j$ with a link to x, then $\mathrm{TR}(x, C_j) \neq 0$ whether or not $x \in C_j$. During query time, the cluster-sensitive ranking is combined with a user's search preference. Given query q, using (for example) a multinomial naive-Bayes classifier, we compute the class probabilities for each of the clusters, conditioned on q. Let q_i be the ith term in the query q. Then given the query q, we compute for each C_j the following:

$$\Pr(C_j \mid q) = \frac{\Pr(C_j) \cdot \Pr(q \mid C_j)}{\Pr(q)} \propto \Pr(C_j) \cdot \prod_i \Pr(q_i \mid C_j). \tag{4.1}$$

Then $\Pr(q_i \mid C_j)$ is easily computed from the class term vector D_j. The quantity $\Pr(C_j)$ is not as straightforward. In the original Topic-Sensitive PageRank, $\Pr(C_j)$ is chosen to be uniform. Certainly, more advanced techniques can be used to better estimate $\Pr(C_j)$. To compute the final rank, we retrieve all documents containing all query terms using a text index. The final query-sensitive ranking of each of these pages is given as follows: For page x, we compute the final importance score $\mathrm{TSPR}(x, q)$ as $\mathrm{TSPR}(x, q) = \sum_{C_j \in \mathcal{C}} \Pr(C_j \mid q) \cdot \mathrm{TR}(x, C_j)$. Then the Topic-Sensitive PageRank algorithm is a personalized search-ranking algorithm with $\mu(C_j, x, q) = \mathrm{TR}(x, C_j)$ and $P(C_j, q) = \Pr(C_j \mid q)$.

4.2. Other Personalized Systems

In [Aktas et al. 04], the Topic-Sensitive PageRank algorithm is employed at the level of URL features such as Internet domain names. In [Chirita et al. 05],

the authors extend the Modular PageRank algorithm [Jeh and Widom 03]. In [Chirita et al. 05], rather than using an arduous process for collecting the user's profile as in Modular PageRank, the user's bookmarks are used to derive the user profile. The algorithm augments the pages obtained in this way by finding their related pages using the Modified PageRank and HITS algorithms.

Most content-based web-search personalization methods are based on the idea of reranking the returned pages in the collection using the content of pages (represented as snippet, title, full content, etc.) with respect to the user profile. Some content-analysis-based personalization methods consider how to collect user profiles as part of their personalization frameworks. In [Liu et al. 02], the authors propose a technique for web-search personalization that maps a user query to a set of categories that represent the user's search intention. A user profile and a general profile are learned from the user's search history and a category hierarchy respectively. Later, these two profiles are combined to map a user query into a set of categories. In [Chirita et al. 05], the authors propose a way of performing web search using the ODP (Open Directory Project) metadata. First, the user has to specify a search preference by selecting a set of topics (hierarchical) from the ODP taxonomy. Then, at run time, the web pages returned by the ordinary search engine can be resorted according to the distance between the URL of a page and the user profile. In [Sun et al. 05] is proposed an approach called CubeSVD (motivated by HOSVD = High-Order Singular Value Decomposition) that focuses on utilizing click-through data to personalize the web search. Note that the click-through data are highly sparse, containing relations among user, query, and clicked web page.

5. A Generative Model and Our Local-Cluster Algorithm

We first define local-cluster algorithms and show how such algorithms can be derived from an existing document ranking. We then present a generative model by modifying the model of [Achilioptas et al. 01] so that clusters will now play the role of documents. This generative model motivates our local-cluster PSP algorithm and also allows us to formulate a correctness result for the PSP algorithm analogous to the correctness result of [Achilioptas et al. 01]. We note that like the SP algorithm, PSP is defined without any reference to the generative model.

5.1. Local-Cluster Algorithms Using an Existing Document Ranking

For a given clustering $\mathcal{C}$, let $\mathrm{CS}(x) = \{C_j \in \mathcal{C} \mid x \in C_j\}$. Given an instance $(G, \mathcal{C}, \mu, P, q)$ of personalized search, a *local-cluster algorithm* is a personalized

search ranking such that $\mathrm{PSR}(x, q)$ depends only on clusters $C_i \in \mathrm{CS}(x)$. A *linear* (personalized search) algorithm is given by $\mathrm{PSR}(x, q) = \sum_j P(C_j, q) \cdot \mu(C_j, x, q)$. A linear local search algorithm is therefore one that satisfies

$$\begin{aligned}\mathrm{PSR}(x, q) &= F(\mu(C_1, x, q), \ldots, \mu(C_m, i, q), P(C_1, q), \ldots P(C_m, q)) \\ &= \sum_{C_j \in \mathrm{CS}(x)} P(C_j, q) \cdot \mu(C_j, x, q).\end{aligned}$$

We note that TSPR is a linear algorithm but not a local algorithm. Our algorithm personalizes existing web services utilizing existing ranking algorithms. Our framework assumes that there is a generic page ranking $R(x, q)$ for ranking page x given query q. Using an algorithm CR to compute the ranking for clusters, we compute the cluster-sensitive ranking $\mu(C_i, x, q)$ as

$$\mu(C_i, x, q) = R(x, q) \cdot \mathrm{CR}(C_i, q),$$

where $\mathrm{CR}(C_j, q)$ refers to the ranking of cluster C_j with respect to query q. In Section 5.3, we will define a specific cluster-ranking algorithm and then define our Personalized SP algorithm as $\mathrm{PSP}(x, q) = \sum_{C_j \in \mathrm{CS}(x)} P(C_j, q) \cdot \mu(C_j, x, q)$. We note that PSP is a linear local-cluster algorithm.

5.2. The Generative Model

Our personalized search algorithm for computing a cluster-sensitive page ranking is based on a linear model capturing correlations between cluster content, cluster linkage, and user preference. Our model borrows heavily from the Latent Semantic Analysis (LSA) of [Deerwester et al. 90], which captures term-usage information based on a (low-dimensional) linear model, and the SP algorithm of [Achilioptas et al. 01], which captures correlations among three components (i.e., links, page content, user query) of web search in terms of proximity in a shared latent semantic space. The algorithm for ranking clusters is the direct analogy of the SP algorithm, where now clusters play the role of pages. That is, we will be interested in the aggregation of links between clusters and the term content of clusters. We modify the generative model of [Achilioptas et al. 01] so as to apply to clusters.

Let $\{C_1, \ldots, C_m\}$ be a clustering for the targeted corpus. Now following [Deerwester et al. 90] and [Achilioptas et al. 01], we assume that there exists a set of k unknown (latent) basic concepts whose combinations represent every topic of the web. Given such a set of k concepts, a *topic* is a k-dimensional vector describing the contribution of each of the basic concepts to this topic.

5.2.1. Authority and Hub Values for Clusters. We extend the notion of a page's authority and hub values as introduced in [Kleinberg 99] and utilized in [Achilioptas et al. 01]. Two vectors are associated with a web page x:

- There is a k-tuple $A(x) \in [0,1]^k$ reflecting the topic on which x is an authority. The cth entry in $A(x)$ expresses the degree to which x concerns the concept c. This topic vector captures the content on which this page is an authority.
- The second vector associated with x is a k-tuple $H(x) \in [0,1]^k$ reflecting the topic on which x is a hub.

Based on this notion of page hub and authority values, we introduce the concept of cluster hub and authority values. With each cluster $C_j \in \mathcal{C}$, we associate two vectors:

- The first vector associated with C_j is a k-tuple $\widetilde{A}^{(j)}$ that represents the cumulative authority value that is accumulated in cluster C_j with respect to each concept. We define $\widetilde{A}^{(j)}$ as $\widetilde{A}^{(j)}(c) = \sum_{x \in C_j} A(x,c)$, where $A(x,c)$ is document x's authority value with respect to the concept c.
- The second vector associated with C_j is a k-tuple $\widetilde{H}^{(j)}$ representing the cumulative hub value accumulated in cluster C_j with respect to each concept. We define $\widetilde{H}^{(j)}$ as $\widetilde{H}^{(j)}(c) = \sum_{x \in C_j} H(x,c)$, where $H(x,c)$ is document x's hub value with respect to concept c.

5.2.2. Link Generation over Clusters. In what follows, we assume that all random variables have bounded range. Given clusters C_p and $C_r \in \mathcal{C}$, our model assumes that the total number of links from pages in C_p to pages in C_r is a random variable with expected value equal to the inner product $\langle \widetilde{H}^{(p)}, \widetilde{A}^{(r)} \rangle$. Note that the intuition is the same as in the link-generation model for two arbitrary documents [Achilioptas et al. 01]. The more closely aligned the hub topic of the pages in C_p is with the authority topic of the pages in C_r, the more likely it is that there will be a link from a document in C_p to a document in C_r. Therefore, the link-generation model among different clusters is described in terms of an $m \times m$ matrix $\widetilde{W} = \widetilde{H} \cdot \widetilde{A}^T$, where the pth row of $\widetilde{H}$ is $(\widetilde{H}^{(p)})^T$ and the rth row of $\widetilde{A}$ is $(\widetilde{A}^{(r)})^T$. Each entry (p,r) of $\widetilde{W}$ represents the expected number of links from C_p to C_r.

Let $\widehat{W}$ be the actual link structure of documents for the targeted corpus and let Z be the $n \times m$ indicator matrix Z whose (x,j) entry indicates whether page x is a page in cluster j. The assumption is that the actual number of links $\overline{W} = Z^T \widehat{W} Z$ from C_p to C_r is an instantiation of the link-generation model for clusters.

5.2.3. Term-Content Generation over Clusters. Once again, our term-content-generation model heavily borrows from that introduced in [Achilioptas et al. 01]. We assume that there are l terms and that the term distributions over clusters are given by the following two distributions:

- The first distribution expresses the expected number of occurrences of terms as authoritative terms within all documents. More precisely, we assume a k-tuple $\widetilde{S}_A^{(u)}$ whose cth entry describes the expected number of occurrences of the term u in the set of *all pure authority documents* in the concept c that are not hubs on anything.

- The second distribution expresses the expected number of occurrences of terms as hub terms within all documents. More precisely, we assume a k-tuple $\widetilde{S}_H^{(u)}$ whose cth entry describes the expected number of occurrences of the term u in the set of *all pure hub documents* in the concept c that are not authorities on anything.

The above distributions can be expressed in terms of two matrices, namely $\widetilde{S}_A$, the $l \times k$ matrix whose rows are indexed by terms, where row u is the vector $(\widetilde{S}_A^{(u)})^T$, and $\widetilde{S}_H$ is the $l \times k$ matrix whose rows are indexed by terms such that row u is the vector $\left(\widetilde{S}_H^{(u)}\right)^T$. Our model assumes that terms within cluster C_p having authority value $\widetilde{A}^{(p)}$ and hub value $\widetilde{H}^{(p)}$ are generated from a distribution of bounded range where the expected number of occurrences of term u is $\left\langle \widetilde{A}^{(p)}, \widetilde{S}_A^{(u)} \right\rangle + \left\langle \widetilde{H}^{(p)}, \widetilde{S}_H^{(u)} \right\rangle$. We describe the term-generation model of clusters with an $m \times l$ matrix $\widetilde{S} = \widetilde{H} \cdot \widetilde{S}_H^T + \widetilde{A} \cdot \widetilde{S}_A^T$, where again m is the number of underlying clusters and l is the total number of possible terms.

The (j, u) entry in $\widetilde{S}$ represents the expected number of occurrences of term u within all documents in cluster j. Let $\widehat{S}$ be the actual term-document matrix of all documents in the targeted corpus. Analogous to the previous link-generation model of clusters, we assume that $\overline{S} = Z^T \widehat{S}$ is an instantiation of the term-generation model of clusters described by $\widetilde{S}$.

5.2.4. Preference Vector. As discussed in Section 1, we assume that the user provides a search preference having in mind certain clusters (types of documents in which he is interested). If the user exactly knows what the given clusters are, then she might directly express her search preference over these clusters. However, such explicit preferences will not generally be available. Instead, we consider a more general scenario in which the user expresses his search interests through a set of keywords (terms). More precisely, the model for the user search preference is as follows:

1. The user expresses his search preference by providing a vector p over terms whose uth entry indicates his degree of preference over the term u. Note that this preference for terms could be made relative to a query or can be query-independent.

2. Given the vector p, the preference vector over clusters is obtained as $p^T \cdot \overline{S}^T$.

5.2.5. User Query. The user has in mind some topic on which she wants to find the most authoritative cluster of documents on the topic when she performs the search. The terms that the user presents to the search engine should be the terms that a perfect hub on this topic would use, and then these terms would potentially lead to the discovery of the most authoritative cluster of documents on the set of topics closely related to these terms. The query-generation process in our model is given as follows:

- The user chooses the k-tuple $\overline{v}$ describing the topic he wishes to search for in terms of the underlying k concepts.
- The user computes the vector $\tilde{q}^T = \overline{v}^T \widetilde{S}_H^T$, where the uth entry of $\tilde{q}$ reflects the expected number of occurrences of the term u in queries on the user's topic.
- The user then decides whether to include term u among her search terms by sampling from a distribution with expectation $\tilde{q}[u]$. We denote the instantiation of the random process by $\overline{q}[u]$.

The input to the search engine consists of the terms with nonzero coordinates in the vector $\overline{q}$.

5.3. Our PSP Algorithm

Given this generative model that incorporates link structure, content generation, user preference, and query, we can rank clusters of documents using a spectral method. While the basic idea and analysis for our algorithm follow from [Achlioptas et al. 01], our PSP algorithm is different from the original SP algorithm in one substantial aspect: *In contrast to the original SP algorithm, which works at the document level, our algorithm works at the cluster level, making our algorithm computationally more attractive and consequently more practical.*[3] More specifically, in our algorithm, the SVD computations of the $\overline{M}$, $\overline{W}$, and $\overline{S}$ matrices are relatively inexpensive, since the size of these matrices depends on the number of clusters rather than the number of documents.

[3]To the best of our knowledge, the SP algorithm has never been implemented. Ignoring any personalization aspects (i.e., setting the preference P to be a constant function), the cluster framework provides a significant computational benefit.

We need some additional notation. For two matrices A and B with an equal number of rows, let $[A \mid B]$ denote the matrix whose rows are the concatenations of the rows of A and B. Let $\sigma_i(A)$ denote the ith-largest singular value of a matrix A and let $r_i(A) = \sigma_1(A)/\sigma_i(A) \geq 1$ denote the ratio between the primary singular value and the ith singular value.

Using standard notation for the singular value decomposition (SVD) of matrix $B \in \mathbb{R}^{n \times m}$, we have $B = U\Sigma V^T$, where U is a matrix of dimension $n \times \text{rank}(B)$ whose columns are orthonormal, Σ is a diagonal matrix of dimension $\text{rank}(B) \times \text{rank}(B)$, and V^T is a matrix of dimension $\text{rank}(B) \times m$ whose rows are orthonormal. The (i, i) entry of Σ is $\sigma_i(B)$. The cluster-ranking algorithm preprocesses the entire corpus of documents, independent of the query.

Preprocessing Step.

1. Let $\overline{M} = [\overline{W}^T \mid \overline{S}]$. $\overline{M} \in \mathbb{R}^{m \times (m+l)}$ (m is the number of clusters and l is the number of terms). Compute the SVD of the matrix as
$$\overline{M}^* = U_{\overline{M}} \Sigma_{\overline{M}} V_{\overline{M}}^T.$$
2. Choose the largest index r such that the difference $|\sigma_r(\overline{M}^*) - \sigma_{r+1}(\overline{M}^*)|$ is sufficiently large (we require $\omega(\sqrt{(m+l)})$). Let $\overline{M}_r^* = (U_{\overline{M}})_r (\Sigma_{\overline{M}})_r (V_{\overline{M}}^T)_r$ be the rank-r SVD approximation to $\overline{M}$.
3. Compute the SVD of the matrix $\overline{W}$ as $\overline{W}^* = U_{\overline{W}} \Sigma_{\overline{W}} V_{\overline{W}}^T$.
4. Choose the largest index t such that the difference $|\sigma_t(\overline{W}^*) - \sigma_{t+1}(\overline{W}^*)|$ is sufficiently large (we require $\omega(\sqrt{(t)})$). Let $\overline{W}_t^* = (U_{\overline{W}})_t (\Sigma_{\overline{W}})_t (V_{\overline{W}}^T)_t$ be the rank-t SVD approximation to $\overline{W}$.
5. Compute the SVD of the matrix $\overline{S}$ as $\overline{S}^* = U_{\overline{S}} \Sigma_{\overline{S}} V_{\overline{S}}^t$.
6. Choose the largest index o such that the difference $|\sigma_o(\overline{S}^*) - \sigma_{o+1}(\overline{S}^*)|$ is sufficiently large (we require $\omega(\sqrt{(o)})$). Let $\overline{S}_o^* = (U_{\overline{S}})_o (\Sigma_{\overline{S}})_o (V_{\overline{S}}^T)_o$ be the rank-o SVD approximation to $\overline{S}$.

Query Step. Once a query vector $\overline{q}^T \in \mathbb{R}^l$ is presented, let $\overline{q'}^T = [0^m \mid \overline{q}^T] \in R^{m+l}$. The user's preference for cluster C_j is the jth component of $P = p^T \overline{S_o^*}^T$, where p is the user's preference vector over terms.[4] Then we compute the cluster authority vector $v(q)^T = \overline{q'}^T \overline{M}_r^{*-1} \overline{W}_t^*$, where $\overline{M}_r^{*-1} = (V_{\overline{M}}^T)_r (\Sigma_{\overline{M}})_r^{-1} (U_{\overline{M}})_r$ is the pseudoinverse of $\overline{M}_r$.

[4] We note that if p were query-dependent, then it would be more appropriate to write $P(C_j, q)$. In our experiments we will use preference vectors that are independent of the query and hence it is more informative to use $P(C_j)$. When the preference vectors are independent of the query, we can compute P in the preprocessing step.

Final Ranking. Once we have computed the ranking for clusters, we proceed with the actual computation of cluster-sensitive page ranking. Let $\upsilon(C_j, q)$ denote the authority value of cluster C_j for query q as computed in the previous section. The cluster-sensitive page rank for page x with respect to cluster C_j is computed as

$$\mu(x, C_j, q) = \begin{cases} R(x, q) \cdot \upsilon(C_j, q) & \text{if } x \in C_j, \\ 0 & \text{otherwise,} \end{cases}$$

where again $R(x, q)$ is the generic rank of page x with respect to query q.

The final personalized rank for page x is computed as

$$\mathrm{PSP}(x, q) = \sum_{C_i \in \mathrm{CS}(x)} P(C_i) \cdot R(x, q) \cdot \upsilon(C_i, q).$$

For a d-dimensional vector $V = (v_1, \ldots, v_d)$, we let $\mathrm{diag}(V)$ denote the $d \times d$ diagonal matrix whose (i, i) entry is v_i. Then in matrix form we express the PSP algorithm as $\mathrm{PSP}(q) = \mathrm{diag}(R)Z\,\mathrm{diag}(P)\upsilon$.

6. Personalized Search Criteria

We present a series of results comparing the Topic-Sensitive PageRank algorithm and our PSP algorithm with respect to a set of personalized search algorithm criteria that we propose.

Our criteria are all of the form "small changes in the input imply small changes in the computed ranking." We believe that such criteria have immediate practical relevance as well as theoretical interest. Since our ranking of documents produces real authority values in $[0, 1]$, one natural approach is to study the effect of small continuous changes in the input information as in the rank-stability studies [Borodin et al. 05, Donato et al. 05, Lee and Borodin 03, Ng et al. 01].

One basic property shared by both Topic-Sensitive PageRank and our PSP algorithm is continuity.

Theorem 6.1. *Both TSPR and our PSP ranking algorithms are continuous; i.e., small changes in any μ value or preference value will result in a small change in the ranking value of all pages.*

Proof. The continuity of Topic-Sensitive PageRank and PSP easily follows from the way these algorithms produce the final ranking. Both algorithms linearly combine μ and P to produce the final ranking. That is, for both algorithms the final rank vector $\mathrm{FR}(q)$ with respect to query q can be written as $\mathrm{FR}(q) =$

$\Gamma(q)\cdot P(q)$, where $\Gamma(q)$ is an $n\times m$ matrix whose (x,j)th entry denotes $\mu(C_j,x,q)$, and $P(q)$ denotes the cluster-preference vector. We first prove the continuity of algorithms with respect to the cluster-preference vector. Given $\epsilon > 0$, we have $\|\Gamma(q)\cdot P(q) - \Gamma(q)\cdot \tilde{P}(q)\|_2 \leq \|\Gamma(q)\|_F \|P(q) - \tilde{P}(q)\|_2 < \epsilon$. Therefore, $\delta = \frac{\epsilon}{m}$ would be sufficient for achieving the continuity of algorithms with respect to the cluster-preference vector. The continuity with respect to μ can be proved in a similar fashion. □

Our first distinguishing criterion is a rather minimal *monotonicity property* that we claim any personalized search should satisfy. Namely, since a (cluster-based) personalized ranking function depends on the ranking of pages within their relevant clusters as well as the preference of clusters, when these rankings for a page and cluster preferences are increased, we expect that the personalized rating can only improve. More precisely, we have the following definition:

Definition 6.2. Let $(G,\mathcal{C},\mu,P,q)$ and $(G,\mathcal{C},\mu,\tilde{P},q)$ be two instances of personalized search. Let χ and ψ be the set of ranked pages produced by $(G,\mathcal{C},\mu,P,q)$ and $(G,\mathcal{C},\mu,\tilde{P},q)$ respectively. Suppose that $x \in \chi$, $y \in \psi$ share the same set of clusters (i.e., $\mathrm{CS}(x) = \mathrm{CS}(y)$), and suppose that $\mu(C_j,x,q) \leq \mu(C_j,y,q)$ and $P(C_j,q) \leq \tilde{P}(C_j,q)$ hold for every C_j that they share. We say that a personalized ranking algorithm is *monotone* if $\mathrm{PSR}(x) \leq \widetilde{\mathrm{PSR}}(y)$ for every such $x \in \chi$ and $y \in \psi$.

We now introduce the idea of "locality." The idea behind locality is that (small) discrete changes in the cluster preferences should have only a minimal impact on the ranking of pages. The notion of locality justifies our use of the terminology "local-cluster algorithm." A *perturbation* ∂_α *of size* α changes a cluster-preference vector P to a new preference vector $\tilde{P} = \partial_\alpha(P)$ such that P and $\tilde{P}$ differ in at most α components. Let $\widetilde{\mathrm{PSR}}$ denote the new personalized ranking vector produced under the new search preference vector $\tilde{P}$.

Definition 6.3. Let $(G,\mathcal{C},\mu,P,q)$ and $(G,\mathcal{C},\mu,\tilde{P},q)$ be the original personalized search instance and its perturbed personalized search instance respectively. Let $\mathrm{AC}(\partial_\alpha)$, the active clusters, be the set of clusters that are affected by the perturbation ∂_α (i.e., $P(C_j,q) \neq \tilde{P}(C_j,q)$ for every cluster C_j in $\mathrm{AC}(\partial_\alpha)$). We say that a personalized ranking algorithm is *local* if for every $x,y \notin \mathrm{AC}(\partial_\alpha)$,

$$\mathrm{PSR}(x,q) \leq \mathrm{PSR}(y,q) \Leftrightarrow \widetilde{\mathrm{PSR}}(x,q) \leq \widetilde{\mathrm{PSR}}(y,q),$$

where PSR refers to the original personalized ranking vector, while $\widetilde{\mathrm{PSR}}$ refers to the personalized ranking vector after the perturbation.

Theorem 6.4. *The Topic-Sensitive PageRank algorithm is not monotone and not local.*

Proof. *Nonmonotonicity of TSPR.* Suppose that G is a graph that consists of four points $\{x_1, x_2, x_3, x_4\}$. Let $C = \{C_1, C_2, C_3\}$ be a clustering of G such that $x_1, x_2 \in C_1$, $x_3 \in C_2$, and $x_4 \in C_3$, and let there be links (edges) $x_3 \to x_1$, $x_4 \to x_1$, and $x_4 \to x_2$. In addition, we assume $\epsilon \geq 0.25$. We have $\mathrm{TR}(x_1, C_1) = \frac{\epsilon}{2} + (1-\epsilon)$, $\mathrm{TR}(x_2, C_1) = \frac{\epsilon}{2} + (1-\epsilon)$, $\mathrm{TR}(x_1, C_2) = (1-\epsilon)\epsilon$, $\mathrm{TR}(x_1, C_3) = (1-\epsilon)\frac{\epsilon}{2}$, and $\mathrm{TR}(x_2, C_2) = 0$, $\mathrm{TR}(x_2, C_3) = (1-\epsilon)\frac{\epsilon}{2}$. Moreover, we assume $P(C_1, q) = 2/5$, $P(C_2, q) = 1$, $P(C_3, q) = 1$, $\tilde{P}(C_1, q) = 3/5$, $\tilde{P}(C_2, q) = 1$, and $\tilde{P}(C_3, q) = 1$. Therefore, all assumptions of monotonicity are satisfied with respect to x_1 and x_2. However, we have

$$\begin{aligned}
&\mathrm{TSPR}\,(x_1)\\
&\quad = P\,(C_1, q)\,\mathrm{TR}\,(x_1, C_1) + P\,(C_2, q)\,\mathrm{TR}\,(x_1, C_2) + P\,(C_3, q)\cdot \mathrm{TR}\,(x_1, C_3)\\
&\quad = \frac{2}{5}\left(\frac{\epsilon}{2} + (1-\epsilon)\right) + (1-\epsilon)\,\epsilon + (1-\epsilon)\,\frac{\epsilon}{2}\\
&\quad > \frac{3}{5}\left(\frac{\epsilon}{2} + (1-\epsilon)\right) + (1-\epsilon)\,\frac{\epsilon}{2}\\
&\quad = \tilde{P}\,(C_1, q)\,\mathrm{TR}\,(x_2, C_1) + \tilde{P}\,(C_2, q)\,\mathrm{TR}\,(x_2, C_2) + \tilde{P}\,(C_3, q)\cdot \mathrm{TR}\,(x_2, C_3)\\
&\quad = \widetilde{\mathrm{TSPR}}\,(x_2)\,.
\end{aligned}$$

Nonlocality of TSPR. In particular, we show that a small perturbation in preference values can have considerably large impact on the overall ranking. Let $G = C_1 \cup C_2 \cup C_3 \cup C_4$, $|C_1| = |C_2| = N - \beta$, and $|C_3| = |C_4| = \beta$, where β is a fixed constant. Every page in $C_3 \cup C_4$ points to every page in $C_1 \cup C_2$. One can verify that for each $x \in C_1$ and $y \in C_2$ we have $\mathrm{TSPR}(x, C_1) = \mathrm{TSPR}(y, C_2)$, and similarly we have $\mathrm{TSPR}(x, C_2) = \mathrm{TSPR}(y, C_1)$. Furthermore, $\mathrm{TSPR}(x, C_3) = \mathrm{TSPR}(x, C_4) = \mathrm{TSPR}(y, C_3) = \mathrm{TSPR}(y, C_4)$. Now suppose that the original cluster preferences are altered from $P(C_1, q) = P(C_2, q)$, $P(C_3, q) < P(C_4, q)$ to $\tilde{P}(C_1, q) = \tilde{P}(C_2, q)$, $\tilde{P}(C_3, q) > \tilde{P}(C_4, q)$. From the original cluster preferences, we will have $\mathrm{TSPR}(x) < \mathrm{TSPR}(y)$ for $x \in C_1$, $y \in C_2$. On the other hand, from the modified cluster preferences, we will have $\widetilde{\mathrm{TSPR}}(x) > \widetilde{\mathrm{TSPR}}(y)$ for $x \in C_1$, $y \in C_2$. That is, nonlocality is proven. □

In contrast we show that our PSP algorithm does enjoy the monotone and local properties.

Theorem 6.5. *Any linear local-cluster algorithm (and hence PSP) is monotone and local.*

Proof. *Monotonicity.* Since by the assumption, for every $C_j \in \mathrm{CS}(x) = \mathrm{CS}(y)$, we have $P(C_j, q) \leq \tilde{P}(\tilde{C}_j, q)$ and $\mu(C_j, x, q) \leq \mu(C_j, y, q)$, we will have

$$\mathrm{PSR}(x) = \sum_{C_j \in \mathrm{CS}(x)} P(C_j, q)\mu(C_j, x, q) \leq \sum_{C_j \in \mathrm{CS}(y)} P(C_j, q)\mu(C_j, y, q) = \widetilde{\mathrm{PSR}}(x).$$

Locality. This easily follows from the fact that the ranking produced by local-cluster algorithms is based only on those clusters containing the point to be ranked. Therefore, the original ranking for points not in the set $\mathrm{AC}(\delta_\alpha)$ of affected clusters is unaffected by the perturbation. □

We next consider a notion of stability (with respect to cluster movement) in the spirit of [Ng et al. 01, Lee and Borodin 03]. Our definition reflects the extent to which small changes in the clustering can change the resulting rankings. We consider the following page-movement changes to the clusters:

- A *migration* $\mathrm{migr}(x, C_i, C_j)$ moves page x from cluster C_i to cluster C_j.
- A *replication* $\mathrm{repl}(x, C_i, C_j)$ adds page x to cluster C_j (assuming x was not already in C_j) while keeping x in C_i.
- A *deletion* $\mathrm{del}(x, C_j)$ is the deletion of page x from cluster C_j (assuming that there exists a cluster C_i in which x is still present).

We define the *size* of these three page-movement operations to be $\mu(C_i, x, q) + \mu(C_j, x, q)$ for migration/replication, and $\mu(C_j, x, q)$ for deletion. We measure the size of a collection M of page movements to be the sum of the individual page-movement costs. Our definition of stability then is that the resulting ranking does not change significantly when the clustering is changed by page movements of small size.

We recall that each cluster is a set of pages and its induced subgraph, induced from the graph on all pages. We will assume that the μ ranking algorithm is a stable algorithm in the sense of [Ng et al. 01, Lee and Borodin 03]. Roughly speaking, locality of a μ ranking algorithm means that there will be a relatively small change in the ranking vector if we add or delete links to a web graph. Namely, the change in the ranking vector will be proportional to the ranking values of the pages adjacent to the new or removed edges.

Definition 6.6. Let $(G, \mathcal{C}, \mu, P, q)$ and $(G, \mathcal{C}, \mu, \tilde{P}, q)$ be a personalized search instance. A personalized ranking function PSR is *cluster-movement stable* if for every set of page movements M there is a β, independent of G, such that

$$\| \mathrm{PSR} - \widetilde{\mathrm{PSR}} \|_2 \leq \beta \cdot \mathrm{size}(M),$$

where PSR refers to the original personalized ranking vector, while $\widetilde{\text{PSR}}$ refers to the personalized ranking vector produced when the set of page movements M has been applied to a given personalized search instance.

Theorem 6.7. *The Topic-Sensitive PageRank algorithm is not cluster-movement stable.*

Proof. We exhibit a counterexample to show that Topic-Sensitive PageRank is not cluster-movement stable. Let G be a graph that consists of $n+1$ points and three clusters C_1, C_2, and C_3 such that C_1 contains x_0, C_2 contains $\{x_2, \ldots, x_n\}$, and C_3 contains all points. We have $x_0 \to x_1$, $x_n \to x_1$, and $x_k \to x_{k+1}$ for every $1 < k < (n-1)$. Furthermore, suppose that $P(C_1, q) = 1$, $P(C_2, q) = 0$, and $P(C_3, q) = 0$.[5] One can verify that $\text{TSPR}(x_0) = \text{TR}(x_0, C_1) = \epsilon$, $\text{TSPR}(x_n) = \text{TR}(x_n, C_1) = \delta$, and $\text{TSPR}(x_m) = \text{TR}(x_m, C_1) = (1-\epsilon)^m(\epsilon + \delta)$, where $\epsilon \leq 1$ and $\delta = \frac{\epsilon(1-\epsilon)^n}{1-(1-\epsilon)^n}$ for every $1 \leq m < (n-1)$.

On the other hand, one can easily see that $\text{TR}(x_i, C_2) = 1/n$ for every $1 \leq i \leq n$. Now we delete x_1 from C_2. One can see that $\widetilde{\text{TSPR}}(x_0, C_1) = \widetilde{\text{TR}}(x_0, C_1) = 1$ and $\widetilde{\text{TSPR}}(x_i, C_1) = \widetilde{\text{TR}}(x_i, C_1) = 0$ for every $1 \leq i \leq n$. We have

$$\begin{aligned}
\frac{\|\,\text{TSPR} - \widetilde{\text{TSPR}}\|_2}{\text{size}(\text{del}(x_1, C_2))} &= n \cdot \sqrt{((\epsilon-1)^2 + \sum_{i=1}^{n-1}((1-\epsilon)^i(\epsilon+\delta))^2} \\
&= n\sqrt{((\epsilon-1)^2 + (\frac{1-(1-\epsilon)^{2n}}{1-(1-\epsilon)^2} - 1)(\epsilon+\delta)^2)} \\
&\geq n(1-\epsilon),
\end{aligned}$$

which is unbounded with respect to n. □

Theorem 6.8. *The PSP algorithm is cluster-movement stable.*

Proof. It is sufficient to consider only replication and deletion, since migration $\text{migr}(x_a, C_i, C_j)$ can be seen as a sequential application of $\text{repl}(x_a, C_i, C_j)$ followed by $\text{del}(x_a, C_i)$. Furthermore, we present the proof only for replication, since the proof for deletion is similar. Let $\text{diag}(R)Z\,\text{diag}(P)v$ be the ranking before the page movement. Let $\tilde{Z} = Z + E$ be the new matrix representing the page's membership in a cluster, where E is given as $E_{a,j} = 1$ for $\text{repl}(x_a, C_i, C_j)$, while the rest of entries are all zero. Let $\text{diag}(R)\tilde{Z}\,\text{diag}(P)\tilde{v}$ be the ranking after

[5] Since C_3 contains x_0, it is not true that $P(C_3, q) = 0$, but when n is sufficiently large, $P(C_3, q) \approx 0$. Therefore, we assume that $P(C_3, q) = 0$ for the sake of simplicity.

the page movement. We will show that

$$\frac{\| \operatorname{diag}(R) Z \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \tilde{\upsilon} \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_1}$$

is bounded by a constant, where λ is the projection of υ over the affected clusters (e.g., $\lambda_{fd} = 1$ for $f = a$, $d = i, j$, and $\lambda_{fd} = 0$ otherwise for $\text{repl}(x_a, C_i, C_j)$).

We have

$$\begin{aligned}
&\frac{\| \operatorname{diag}(R) Z \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \tilde{\upsilon} \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_1} \\
&\quad \leq \frac{1}{\| \operatorname{diag}(R) \lambda \upsilon \|_2} \big\| \operatorname{diag}(R) Z \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon \\
&\qquad\qquad + \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \tilde{\upsilon} \big\|_2 \\
&\quad \leq \frac{\| \operatorname{diag}(R) Z \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2} \\
&\qquad + \frac{\| \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \tilde{\upsilon} \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2}.
\end{aligned}$$

The first term,

$$\frac{\| \operatorname{diag}(R) Z \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2} = \frac{\| \operatorname{diag}(R) E \operatorname{diag}(P) \upsilon \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2},$$

is bounded as follows. We have

$$\begin{aligned}
\| \operatorname{diag}(R) E \operatorname{diag}(P) \upsilon \|_2 &= \sqrt{R(x_a)^2 P(C_i)^2 \upsilon_j^2} \leq \sqrt{R(x_a)^2 \upsilon_i^2 + R(x_a)^2 \upsilon_j^2} \\
&= \| \operatorname{diag}(R) \lambda \upsilon \|_2.
\end{aligned}$$

Therefore,

$$\frac{\| \operatorname{diag}(R) E \operatorname{diag}(P) \upsilon \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2} \leq 1.$$

The second term,

$$\frac{\| \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \upsilon - \operatorname{diag}(R) \tilde{Z} \operatorname{diag}(P) \tilde{\upsilon} \|_2}{\| \operatorname{diag}(R) \lambda \upsilon \|_2},$$

is bounded as follows. One should note that

$$\| \operatorname{diag}(R) \lambda \upsilon \|_2 \geq \frac{1}{\sqrt{2}} \| \operatorname{diag}(R) \lambda \|_F \| \tau \|_2 \| \upsilon \|_2,$$

where τ is the smallest possible cluster-ranking value (i.e., for a cluster having one page with no links). Therefore, we have

$$\frac{\| \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\upsilon - \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\tilde{\upsilon}\|_2}{\| \operatorname{diag}(R)\lambda\upsilon\|_2} \leq \sqrt{2}\frac{\| \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\|_F \|\upsilon - \tilde{\upsilon}\|_2}{\| \operatorname{diag}(R)\lambda\|_F \|\tau\|_2 \|\upsilon\|_2}.$$

But one can observe that

$$\begin{aligned}\frac{\| \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\|_F}{\| \operatorname{diag}(R)\lambda\|_F} &\leq \frac{\sqrt{\sum_{x_i} \sum_{x_i \in C_j} R^2(x_i, q) P(C_j, x, q)^2}}{\sqrt{R(x_a, q)^2}} \\ &\leq \sqrt{\frac{\sum_{x_i} R^2(x_i, q) \sum_{x_i \in C_j} 1}{R^2(x_a, q)}} \\ &\leq \sqrt{\frac{\sum_{x_i} R^2(x_i, q) m}{R^2(x_a, q)}} \\ &\leq \sqrt{2m}.\end{aligned}$$

Moreover, we have

$$\frac{\|\upsilon - \tilde{\upsilon}\|_2}{\|\tau\|_2 \|\upsilon\|_2} \leq \frac{1}{\tau}\left(1 + \frac{\|\tilde{\upsilon}\|_2}{\|\upsilon\|_2}\right) \leq \frac{2}{\tau}.$$

Therefore,

$$\frac{\| \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\upsilon - \operatorname{diag}(R)\tilde{Z} \operatorname{diag}(P)\tilde{\upsilon}\|_2}{\| \operatorname{diag}(R)\lambda\upsilon\|_2} \leq \frac{2\sqrt{2m}}{\tau}.$$

□

7. Experiments

As a proof of concept, we implemented the PSP algorithm and the Topic-Sensitive PageRank algorithm for comparison. In Section 7.1, we consider the retrieval effectiveness of our PSP algorithm versus that of the Topic-Sensitive PageRank algorithm. In Section 7.2, we briefly discuss experiments regarding monotonicity and locality. A more complete reporting of experimental results can be found at http://www.cs.toronto.edu/~leehyun/cbps_experiment.html.

As a source of data, we used the Open Directory Project (ODP)[6] data, which is the largest and most comprehensive human-edited directory in the Web. We

[6]See http://www.dmoz.com.

first obtained a list of pages and their respective categories from the ODP site. Next, we fetched all pages in the list, and parsed each downloaded page to extract its pure text and links (without nepotistic links). We treat the set of categories in the ODP that are at distance two from the root category (i.e., the "Top" category) as the cluster set for our algorithms. In this way, we constructed 549 categories (or clusters) in total. The categorization of pages using these categories did not constitute a partition, since some pages (5.4% of ODP data) belong to more than one category.

7.1. Comparison of Algorithms

To produce rankings, we first retrieved all the pages that contained all terms in a query, and then computed rankings taking into account the specified categories (as explained below). The PSP algorithm assumes that there is already an underlying page ranking for the given web service. Since we were not aware of the ranking used by the ODP search, we simply used pure PageRank as the generic page ranking for our PSP algorithm. Topic-Sensitive PageRank was implemented as described in Section 4.1.1. We used the same $\alpha = 0.25$ value as that used in [Haveliwala 03].

We devised 20 sample queries and their respective search preferences (in terms of categories) as shown in Table 1. The "preferred" categories were chosen as follows: for each query in Table 1, we chose a random subset of the categories given for the top-ranked pages returned by the ODP search. For the Topic-Sensitive PageRank algorithm, we did not use the approach for automatically discovering the search preference (See (4.1)) from a given query, since we found that the most probable categories discovered in this way were heavily biased toward "news"-related categories. Instead, we computed both Topic-Sensitive PageRank and PSP rankings by equally weighting all categories listed in Table 1.

The evaluation of ranking results was done by three individuals, two with computer-science degrees and one with an engineering degree, all with extensive web search experience.

We used the precision over the top ten (p@10) as the evaluation measure using the methodology employed in [Tsaparas 04]. That is, for each query we merged the top ten results returned by both algorithms into a single list. Without any prior knowledge about what algorithm was used to produce the corresponding result, each person was asked to carefully evaluate each page from the list as "relevant" if in their judgment the corresponding page should be treated as a relevant page with respect to the given query and one of the specified categories, or *nonrelevant* otherwise. In Table 2, we summarize the evaluation results, where the presented precision value is the average of all three precision values.

Query Used	Categories	Query Used	Categories
middle east	Society/Issues News/Current Events Recreation/Travel	northern light	Science/Astronomy Kids and Teens/School Time Science/Software
popular blog	Arts/Weblogs Arts/Chats and Forums News/Weblogs	jaguar	Recreation/Autos Sports/Football Science/Biology
planning	Home/Personal Finance Shopping/Weddings Recreation/Parties	star wars	Arts/Movies Games/Video Games Recreation/Models
common tricks	Home/Do It Yourself Arts/Writers Resources Games/Video Games	technique	Science/Methods and Techniques Arts/Visual Arts Shopping/Crafts
integration	Computers/Software Health/Alternative Society/Issues	strong man	Sports/Strength Sports World/Deutch Recreation/Drugs
chaos	Science/Math Society/Religion and Spirituality Games/Video Games	vision	Health/Senses Computers/Artificial Intelligence Business/Consumer Goods
proverb	Society/Folklore Reference /Quotations Home/Homemaking	conservative	Society/Politics Society/Religion and Spirituality News/Analysis and Opinion
english	Arts/Education Kids and Teens/ Society/Ethnicity School Time	graphic design	Business/Publishing and Printing Computers/Graphics Arts/Graphic Design
fishing expedition	Recreation/Camps Recreation /Outdoors Sports/Adventure Racing	liberal	Society/Politics Society/Religion and Spirituality News/Analysis and Opinion
war	Society/History Games/Board Games Reference/Museums	environment	Business/Energy and Environment Science/Environment Arts/Genres

Table 1. Sample queries and the preferred categories for search used in our experiments.

These evaluation results suggest that our PSP algorithm outperforms the Topic-Sensitive PageRank algorithm. We also report on the actual produced results in *experimental result 1* of our web page.

To gain further insight, we analyzed the distribution of categories associated with each produced ranking. An ideal personalized search algorithm should retrieve pages in clusters representing the user's specified categories as the top-

Query	PSP	TSPR	Query	PSP	TSPR
middle east	0.76	0.8	northern lights	0.7	0.8
popular blog	0.93	0.7	jaguar	0.96	0.46
planning	0.96	0.56	star wars	0.6	0.66
common tricks	0.66	0.9	technique	0.96	0.7
integration	0.6	0.16	strong man	0.9	0.86
chaos	0.56	0.56	vision	0.43	0
proverb	0.9	0.83	conservative	0.86	0.76
english	0.8	0.26	graphic design	1	0.73
fishing expedition	0.86	0.66	liberal	0.76	0.73
war	0.83	0.16	environment	0.93	0.5

Average PSP = 0.80, Average TSPR = 0.59

Table 2. Top ten precision scores for PSP and Topic-Sensitive PageRank.

ranked pages. Therefore, in the list of the top 100 pages associated with each query, we computed how many pages were associated with those categories specified in each search preference. Each page p in the list of the top 100 pages was counted as $1/|\,\mathrm{nc}(p)|$, where $\mathrm{nc}(p)$ is the total number of categories associated with page p. We report on these results in Table 3.

The results here exclude four queries (strong man, popular blog, common tricks, and vision) that did not retrieve a sufficient number of relevant pages in their lists of top 100 pages. Note that with the $1/|\,\mathrm{nc}(p)|$ scoring, the total sum of all three preferred categories for each query was always less than 100, since several pages pertain to more than one category. For several queries in our web page, one can observe that each algorithm's favored category is substantially different. For instance, for the query "star wars," the PSP algorithm prefers the "Games/Video Games" category, while Topic-Sensitive PageRank prefers the "Recreation/Models" category. Furthermore, for the queries "liberal," "conservative," "technique," "english," and "planning," the PSP algorithm and the Topic-Sensitive PageRank algorithm have very different views on what the most important context associated with "liberal," "conservative," "technique," "english," and "planning" is. One should also observe that when there is a highly dominant query context (e.g., "Society/Issues" category for "integration," and "Arts/Graphic Design" for "graphic design") over other query contexts, then for both algorithms the rankings are dominated by this strongly dominant category with PSP being somewhat more focused on the dominant category. Finally, in averaging over all queries, 86.38% of pages in the PSP list of the top 100 pages were found to be in the specified preferred categories, while for Topic-Sensitive PageRank, 69.05% of pages in the list of the top 100 pages were found to be in the specified preferred categories.

Query	Category	PSP	TSPR
Middle East	Society/Issues	51.17	6.17
	News/Current Events	3.67	14.17
	Recreation/Travel	31.50	19.50
Liberal	Society/Politics	48.33	26.67
	News/Analysis and Opinion	4.50	49.50
	News/Religion and Spirituality	36.83	1.00
Planning	Home/Personal Finance	80.75	3.0
	Shopping/Wedding	13.00	68.00
	Recreation/Parties	4.5	11.50
Chaos	Science/Math	11.33	49.91
	Society/Religion and Spirituality	28	3
	Games/Video Games	57	30.00
Integration	Computers/Software	0.0	0.0
	Health/Alternative	0.0	0.0
	Society/Issues	88.92	54.08
English	Arts/Education	17.83	43.66
	Kids and Teens/School Time	35.16	8.33
	Society/Ethnicity	23.5	5.66
Fishing Expedition	Recreation/Camps	62.33	62.33
	Sports/Adventure Racing	22.83	22.83
	Recreation/Outdoors	0.50	0.50
War	Society/History	65.75	15.41
	Games/Board Games	0.5	0.5
	Reference/Museums	8.25	9.08
Northern Lights	Science/Astronomy	62.33	62.33
	Kids and Teens/School Time	22.83	22.83
	Science/Software	0.50	0.50
Jaguar	Recreation/Autos	53.83	68
	Sports/Football	15.5	15.5
	Science/Biology	26	0
Star Wars	Arts/Movies	22.83	11.33
	Games/Video Games	69.83	32.33
	Recreation/Models	0.0	0.0
Technique	Science/Methods and Techniques	2	17
	Arts/Visual Arts	36.5	6.5
	Shopping/Crafts	55.5	42.5
Conservative	Society/Politics	53.17	20.33
	News/Analysis and Opinion	8.0	56.00
	News/Religion and Spirituality	30.00	1.00
Graphic Design	Business/Publishing and Printing	0.5	0.5
	Computers/Graphics	0.0	0.0
	Arts/Graphic Design	94	92.5
Environment	Business/Energy and Environment	3.16	3.3
	Environment	74.41	26.25
	Arts/Genres	1	17.5

Table 3. Distribution of the preferred categories in the top 100 pages.

We personally considered a number of queries altering the preferred categories. For "integration," we considered the single category "Science/Math," and the

precision over the top 20 was 0.5 for PSP and 0.35 for TSPR. (Both algorithms suffered from uses of the term "integration" not applying to the calculus operation.) For the "star wars" query, we added the category "Society/Politics" to our preferred categories. We note that the ODP search does not return any pages within this category. Looking at the top 100 pages returned, PSP returned 3 pages in society/politics not relevant to our query, while TSPR returned 33 nonrelevant pages in this category. We also considered the query "middle east" (using the single category "Recreation/Travel"), the query "conservative" (using the single category "News/Religion and Spirituality"), and the query "jaguar" (using the single category "Sports/Football") with regard to precision over the top ten and observed that PSP performed qualitatively much better than TSPR. We report on these results in the *experimental result 3* of our web page.

We further compared the PSP and TSPR rankings using a variant of the Kendall tau similarity measure [Haveliwala 03, Fagin et al. 03]. Consider two partially ordered rankings σ_1 and σ_2, each of length n, and let U be the union of the elements in σ_1 and σ_2. Let σ_1' be the extension of σ_1, where σ_1' contains the pages in $\sigma_2 - \sigma_1$ appearing after all the URLs in σ_1. We do the analogous σ_2' extension of σ_2. Using the measure

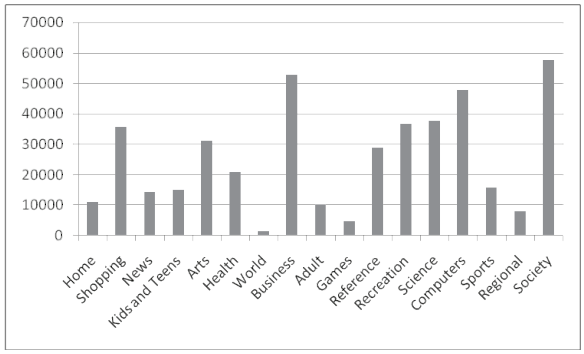
$$\mathrm{KTSim}(\sigma_1, \sigma_2) = \frac{|\{(u,v) : \sigma_1', \sigma_2' \text{ agree on order of } (u,v), u \neq v\}|}{|U\|U-1|},$$

we computed the pairwise similarity between the PSP and TSPR rankings with respect to each query. Averaging over all queries, the KTSim value for the top 100 pages is 0.58, while the average KTSim value for the top 20 pages is 0.43, indicating a substantial difference in the rankings.

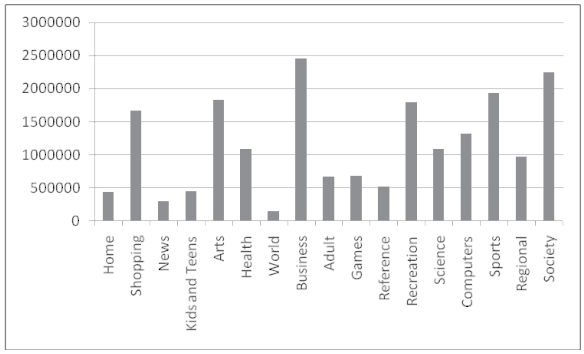
7.2. Monotonicity and Locality

How is the absence of monotonicity (as shown in Theorem 6.4) reflected in the ODP data? We searched our ODP data set and randomly selected 19,640,761 pairs[7] of sites (x, y) that share precisely one common cluster $C_{I(x,y)}$; i.e., $\mathrm{CS}(x) \cap \mathrm{CS}(y) = \{C_{I(x,y)}\}$ and $\mathrm{TR}(x, C_{I(x,y)}) < \mathrm{TR}(y, C_{I(x,y)})$. We computed the final $\mathrm{TSPR}(x)$ and $\mathrm{TSPR}(y)$ by uniformly weighting all 549 categories (or clusters). We found that 431,116 (approximately 2%) of these pairs violated monotonicity; that is, the ranking in $C_{I(x,y)}$ was opposite to the ranking produced by TSPR without favoring any particular category (or clusters). This would lead to a violation of monotonicity in the sense of Theorem 6.4 if, for example, we generated a query using the intersection of common terms in x and y. We report on the distribution of pairs violating monotonicity in Figure 1.

[7]The pairs were selected in such way that they were reasonably distributed with respect to the common cluster.



Distribution (with respect to the top level 17 categories of ODP) of pair pages violating monotonicity



Distribution (with respect to the top level 17 categories of ODP) of original pair pages

Figure 1. Monotonicity results.

We also conducted a study on how sensitive the algorithms are to change in search preferences. We argued that such sensitivity is theoretically captured by the notion of locality in Section 6, and showed that the PSP algorithm is robust to the change in search preferences, while Topic-Sensitive PageRank is not. Our experimental evidence indicates that the Topic-Sensitive PageRank algorithm is somewhat more sensitive to the change in search preferences. For each query we randomly chose seven equally weighted categories so as to define a fixed preference vector.

Let Δ_α^N refer to the class of perturbations induced by deleting some set of α categories. To compare the personalized ranking vectors produced under different perturbations, we again use the above KTSim measure [Haveliwala 03, Fagin et al. 03]. In particular, we varied α as 1, 3, and 5 and for each fixed α and

α	PSP	Topic-Sensitive PageRank
1	0.91	0.92
3	0.77	0.69
5	0.79	0.66

Table 4. Average KTSim values of rankings under different perturbation sizes across all queries.

for five random $\partial_i \in \Delta_\alpha^N$, we computed the resulting rankings and then all $\binom{5}{2}$ pairs of (KTSim) values considering the top 100 pages. We report on the average pairwise similarity across all queries for each fixed α in Table 4.[8]

8. The Correctness of the PSP Algorithm

The following theorem formalizes the correctness of the PSP algorithm with respect to the generative model formulated in Section 5.

Theorem 8.1. *Assume that the link structure for clusters, term content for clusters, and search query are generated as described in our model:* $\overline{W}$ *is an instantiation of* $\widetilde{W} = \widetilde{H}\widetilde{A}^T$, $\overline{S}$ *is an instantiation of* $\widetilde{S} = \widetilde{A}\widetilde{S}_A^T + \widetilde{H}\widetilde{S}_H^T$, $\overline{q}$ *is an instantiation of* $\tilde{q} = \overline{v}^T\widetilde{S}_H^T$, *the user's preference is provided by the vector* p, *and* $R(q)$ *is a vector whose entries correspond to the generic ranks of pages (i.e.,* $R(x, q)$ *corresponds to the generic rank of page* x *with respect to query* q*). Additionally, we have the following:*[9]

(1) $\overline{q}$ *has* $\omega(k \cdot r_k(\overline{W})^2 r_{2k}(\overline{M})^2 r_k(Z))$ *nonzero terms.*

(2) $\sigma_k(\overline{W}) \in \omega(r_{2k}(\overline{M})r_k(Z)\sqrt{m})$ *and* $\sigma_{2k}(\overline{M}) \in \omega(r_k(\overline{W})\ r_{2k}(\overline{M})r_k(Z)\sqrt{m})$,

(3) $\overline{W}$, $\overline{H}\overline{S}_A^T$ *and* $\overline{S}_H^T$ *are of rank* k, $\overline{M} = [\overline{W}^T|\overline{S}]$ *is rank* $2k$, $l = O(m)$, *and* $m = O(k)$.

Then the PSP algorithm computes a vector of personalized ranks that is very close to the correct ranking. More precisely, we have

$$\frac{\|V_{\text{PSP}} - V_{\text{expected}}\|_2}{\|V_{\text{expected}}\|_2} \in O(1),$$

[8] In [Fagin et al. 03], the authors note that this KTSim variant has a mild personalization factor for items not common to both orderings, whence the rather large values.

[9] The assumption $\ell = O(m)$ may seem somewhat artificial, since it may suggest that the number of terms is of order the number of clusters. We acknowledge that there will be many more terms than clusters, but we argue that the number of terms is much closer to the number of clusters than it is to the number of documents.

where $V_{\mathrm{PSP}} = \mathrm{diag}(R)Z\,\mathrm{diag}(P)\overline{q'}^{T}\overline{M}^{*-1}_{r}\overline{W}^{*}_{t}$ *and* $V_{\mathrm{expected}} = \mathrm{diag}(R)Z \cdot \mathrm{diag}(\tilde{P})\overline{v}^{T}\widetilde{A}^{T}$. *The proof of this theorem is given in the appendix (Section 11).*

9. Online Considerations

Given the dynamic nature of the Web, it is important to consider personalized search engines in an online scenario whereby pages are incrementally added, deleted, or updated. As a consequence, clusters are updated, and this may in turn result in a desired change of the clustering (i.e., whereby existing clusters are merged or split). Online considerations have not received much attention in this context.

The preprocessing phase of the PSP algorithm relies on the SVD computation of $\overline{M}$, representing the linkage and semantic relations between clusters. The online addition, deletion, or update of pages would then correspond to the addition, deletion, or update of fragments of rows and columns in $\overline{M}$ and the consequent online updating of the SVD. There is a rich literature concerning online SVD updating.

Recently, [Brand 03] proposed a family of sequential update rules for adding data to a "thin" SVD data model, revising or removing data already incorporated into the model, and adjusting the model when the data-generating process exhibits nonstationarity. Moreover, the author experimentally tested the practicability of the proposed approach in an interactive graphical movie recommender that predicts and displays ratings/rankings of thousands of movie titles in real time as a user adjusts ratings of a small arbitrary set of movies. By applying such methods, the relevant aspects of the preprocessing phase becomes an online computation.

If the existing clustering structure is modified by (say) merging the existing clusters (without affecting the existing pages), the online updating of our PSP algorithm can be efficiently implemented by merging ranks of affected clusters while renormalizing ranks of unaffected clusters, illustrated in the following: We first define what a merging of clusters means. We briefly consider how the online PSP handles merges (and analogously, splits). By a merging operation μ over a clustering $\mathcal{C}$, we refer to a transformation that modifies the given clustering $C = \{C_1, \ldots, C_m\}$ into a new clustering such that two clusters are merged. After clusters are renamed, the new clustering of G is $\tilde{C} = \{\tilde{C}_1, \ldots, \tilde{C}_{m-1}\}$ with $\tilde{C}_{m-1} = C_{m-1} \cup C_m$ and $\tilde{C}_s = C_s$ for every $s \leq m-2$. When the existing clustering structure is modified by such a merging, the online updating of the PSP cluster ranking can be efficiently implemented by merging ranks of

affected clusters while renormalizing ranks of unaffected clusters as justified by the following theorem.

Theorem 9.1. *Let*

$$\Lambda = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 1 \end{bmatrix},$$

and let $\overline{q'}^T = [0^m|\overline{q}^T]$ *and* $\overline{q''}^T = [0^{m-1}|\overline{q}^T]$. *Let* $\overline{M'} = [\overline{W'}^T|\overline{S'}]$ *and* $\overline{M''} = [\overline{W''}^T|\overline{S''}] = [\Lambda\overline{W'}^T\Lambda^T|\Lambda\overline{S'}]$, *where* $\overline{W'}^T, \overline{S'}$ *are the original link and term/document matrices and* $\overline{W''}^T, \overline{S''}$ *are the newly created link and term/document matrices (as produced by merging). Then*

$$\overline{q''}^T\overline{M''_r}^{-1}\overline{W''}_t = \overline{q'}^T\overline{M'_r}^{-1}\overline{W'}_t\Lambda^T. \tag{9.1}$$

Proof. One can first verify that $\overline{M''} = \Lambda\overline{M'}\Lambda^*$, where Λ^* is given in the form

$$\Lambda^* = \left[\begin{array}{c|c} \Lambda^T & 0_{l\times l} \\ \hline 0_{l\times m-1} & I_{l\times l} \end{array}\right],$$

where $I_{l\times l}$ denotes the identity matrix of size $l \times l$, and $0_{l\times m-1}$ denotes the $l \times (m-1)$ zero matrix. Let $\overline{M'} = U_1D_1V_1^T$ and $\overline{M''} = U_2D_2V_2^T$ be the SVD compositions for matrices $\overline{M'}$ and $\overline{M''}$. We have that the pseudoinverse $\overline{M''_r}^{-1}$ is given as

$$\begin{aligned} \overline{M''_r}^{-1} &= V_2D_2^{-1}U_2^T = V_2(U_2^{-1}\Lambda U_1D_1V_1^T\Lambda^*{V_2^T}^{-1})^{-1}U_2^T \\ &= V_2V_2^T(\Lambda^{*-1}V_1D_1^{-1}U_1^T\Lambda^{-1}U_2)U_2^T \\ &= \Lambda^{*-1}V_1D_1^{-1}U_1^T\Lambda^{-1} = \Lambda^{*-1}\overline{M'_r}^{-1}\Lambda^{-1}. \end{aligned}$$

Thus, one can verify that (9.1) holds, since

$$\overline{q''}^T\overline{M}_{r''}^{-1}\overline{W'}_t = \overline{q''}^T\Lambda^{*-1}\overline{M'_r}^{-1}\Lambda^{-1}\Lambda\overline{W}_t\Lambda^T = \overline{q'}^T\overline{M'_r}^{-1}\overline{W}_t\Lambda^T. \qquad \square$$

To complete the merging operation, we define a preference vector

$$\tilde{P}(\tilde{C}_{m-1}, q) = \frac{P(C_{m-1}, q) + P(C_m, q)}{2}.$$

and $\tilde{P}(\tilde{C}_s, q) = P(C_s, q)$ for $s \le m-2$. Finally, it is important to note that as new pages arrive and are placed into their relevant pages, the PSP ranking will change only gradually.

10. Conclusion

We have developed and implemented a computationally efficient "local-cluster" algorithm (PSP) for personalized search. Following [Achilioptas et al. 01], we can prove the correctness of the PSP algorithm relative to a probabilistic generative model. We propose some formal criteria for evaluating personalized ranking algorithms, and demonstrate both theoretically and experimentally that our algorithm is a good alternative to the Topic-Sensitive PageRank algorithm.

11. Appendix: Proof of Theorem 8.1

Using the Cauchy–Schwarz inequality yields

$$\begin{aligned}\|V_{\mathrm{PSP}} - V_{\mathrm{expected}}\| &= \|\mathrm{diag}(R) Z \,\mathrm{diag}(P) \overline{q'}^T \overline{M}^{*-1}_r \overline{W}^*_t - \mathrm{diag}(R) Z \,\mathrm{diag}(\tilde{P}) \overline{v}^T \widetilde{A}^T\|_2 \\ &\leq \|R\|_2 \|Z \,\mathrm{diag}(P) \overline{q'}^T \overline{M}^{*-1}_r \overline{W}^*_t - Z \,\mathrm{diag}(\tilde{P}) \overline{v}^T \widetilde{A}^T\|_2.\end{aligned}$$

Similarly,

$$\|V_{\mathrm{expected}}\| = \|\,\mathrm{diag}(R) Z \,\mathrm{diag}(P) \overline{v}^T \widetilde{A}^T\|_2 \geq \min_{i=1,\ldots,n} |R_i| \|Z \,\mathrm{diag}(P) \overline{v}^T \widetilde{A}^T\|_2.$$

Thus, it is sufficient to prove a lower bound on $\|Z \,\mathrm{diag}(P) \overline{v}^T \widetilde{A}^T\|_2$ and an upper bound on

$$\|Z \,\mathrm{diag}(\tilde{P}) \overline{q'}^T \overline{M}^{*-1}_r \overline{W}^*_t - Z \,\mathrm{diag}(\tilde{P}) \overline{v}^T \widetilde{A}^T\|_2,$$

where $\mathrm{diag}(P)$ denotes $\mathrm{diag}(p^T \overline{S}^{*T}_o)$, while $\mathrm{diag}(\tilde{P})$ denotes $\mathrm{diag}(p^T \widetilde{S}^T)$.

We first introduce some claims that will be used to prove lower and upper bounds. Let $\widetilde{M} = [\widetilde{W}^T | \widetilde{S}] \in \mathbb{R}^{m \times (m+l)}$, and let $\tilde{q}'^T = [0^m | \tilde{q}^T] \in \mathbb{R}^{m+l}$.

Claim 11.1. *We claim that $\tilde{q}'^T$ is in the row space of $\widetilde{M}$.*

Proof. We can rewrite $\widetilde{M}$ as follows:

$$\widetilde{M} = [\widetilde{W}^T | \widetilde{S}] = [\widetilde{A}\widetilde{H}^T | \widetilde{A}\widetilde{S}^T_A + \widetilde{H}\widetilde{S}^T_H] = [\widetilde{A} | \widetilde{H}] \begin{bmatrix} \widetilde{H}^T & \widetilde{S}^T_A \\ 0^{k \times m} & \widetilde{S}^T_H \end{bmatrix}. \tag{11.1}$$

If $\mathrm{rank}(\widetilde{M}) = 2k$, then the row space of $\widetilde{M}$ is equal to the row space of the right-hand matrix in (11.1). We have

$$\tilde{q}'^T = [0^m | \tilde{q}^T] = [0^m | \overline{v}^T \widetilde{S}^T_H] = \overline{v}^T [0^{k \times m} | \widetilde{S}^T_H],$$

which implies that $\tilde{q'}^T$ is in the row space of $\widetilde{M}$. Therefore, there exists some $u \in \mathbb{R}^m$ in the column space of $\widetilde{M}$ such that $u^T\widetilde{M} = \tilde{q}'^T$. □

Claim 11.2. *Let $u \in \mathbb{R}^m$ be such that $u^T\widetilde{M} = \tilde{q}'^T$. Then $u^T\widetilde{W} = \overline{v}^t\widetilde{A}^t$.*

Proof. We can write

$$\widetilde{M} = [\widetilde{A}\widetilde{H}^T|\widetilde{H}\widetilde{S}_H^T + \widetilde{A}\widetilde{S}_A^T].$$

We know that $u^T\widetilde{M} = \tilde{q}'^T = [0^m|\tilde{q}^T]$. From this we learn that $u^T\widetilde{A}\widetilde{H}^T = 0^m$. since $\widetilde{H}^T$ is of rank k, it follows that $u^T\widetilde{A} = [0^k]$. Thus, $u^T\widetilde{M} = [0^m|u^T\widetilde{H}\widetilde{S}_H^T] = \tilde{q}'^T$, which implies $u^T\widetilde{H}\widetilde{S}_H^T = \overline{v}^T\widetilde{S}_H^T$. Since $\text{rank}(\widetilde{S}_H^T) = k$, this implies that $u^t\widetilde{H} = \overline{v}^T$. Multiplying by $\widetilde{A}$ gives us the required result $u^t\widetilde{W} = u^T\widetilde{H}\widetilde{A}^T = \overline{v}^T\widetilde{A}^T$. □

Claim 11.3. *Let $M^+ = [0^{k\times m}|\widetilde{H}\widetilde{S}_H^T]$. Then $u^T\widetilde{M} = \tilde{q}'^T$ iff $u^TM^+ = q^T$. It is important to note that M^+ is spanned by the columns of $\widetilde{H}$, which is of rank k, whereas our assumption on $\widetilde{M}$ is that it is of rank $2k$.*

Proof. In the course of arguing Claim 11.2, we showed that

$$u^T\widetilde{M} = \tilde{q}'^T = [0^m|\tilde{q}^T] = [0^m \mid u^T\widetilde{H}\widetilde{S}_H^T],$$

but it is also true that $u^TM^+ = [0^m \mid u^T\widetilde{H}\widetilde{S}_H^T]$. □

Claim 11.4. *Let $u^T\widetilde{M} = \tilde{q}'^T$, let $(M^+)^{-1}$ be the pseudoinverse of M^+, and let $\widetilde{M}^{-1}$ be the pseudoinverse of $\widetilde{M}$. Then $u^T = \tilde{q}'^T(M^+)^{-1} = \tilde{q}'^T\widetilde{M}^{-1}$.*

Proof. It is easy to see that $\tilde{q}'^T$ is in the row space M^+. From Claims 11.3 and 11.1, we have

$$u^TM^+ = \tilde{q}'^T = u^T\widetilde{M}.$$

Therefore,

$$u^T = \tilde{q}'^TM^{+^{-1}} = \tilde{q}'^T\widetilde{M}^{-1}.$$ □

Claim 11.5. *Let $\tilde{q}^T = \overline{v}^T\widetilde{S}_H^T$, $\tilde{q}'^T = [0^m \mid \tilde{q}^T]$, $M^+ = [0^{k\times m} \mid \widetilde{H}\widetilde{S}_H^T]$. Then we have*

$$\begin{aligned}\overline{v}^T\widetilde{A}^T &= u^T\widetilde{W} && \textit{(from Claim 11.2)}\\ &= \tilde{q}'^T(M^+)^{-1}\widetilde{W} = \tilde{q}'^T(\widetilde{M})^{-1}\widetilde{W} && \textit{(from Claim 11.4)}\end{aligned}$$

Claim 11.6. *Let $\tilde{q}^T = \overline{v}^T\widetilde{S}_H^T$, $\tilde{q}'^T = [0^m|\tilde{q}^T]$. Then $(\tilde{q}'^T\widetilde{M}^{-1})^T$ is in the column space of $\widetilde{W}$.*

Proof. Recall that $M^+ = [0^{k\times m}|\widetilde{H}\widetilde{S}_H^T]$ and that the SVD composition of M^+ is given by $U_{M^+}\Sigma_{M^+}V_{M^+}^T$. Given that M^+ is of rank k, the columns of U_{M^+} span the same space as the columns of $\widetilde{H}$. Since $\widetilde{W} = \widetilde{H}\widetilde{A}^T$ and $\widetilde{W}$ is of rank k, the columns of $\widetilde{W}$ and the columns of $\widetilde{H}$ span the same space: the column space of U_{M^+} is the same as the column space of $\widetilde{W}$. From Claim 11.4, we know that $\tilde{q}'^T\widetilde{M}^{-1} = \tilde{q}'^T M^{+-1} = (\tilde{q}'^T V_{M^+}\Sigma_{M^+}^{-1})$, from which it follows that $\tilde{q}'^T\widetilde{M}^{-1}$ is a linear combination of the rows of $U_{M^+}^T$, or alternatively that $(\tilde{q}'^T\widetilde{M}^{-1})$ is in the column space of U_{M^+}, which is the same space as the column space of $\widetilde{W}$. □

Lemma 11.7. [Achilioptas et al. 01] *For any $B \in \mathbb{R}^{i\times j}$, $i \geq j$, whose columns have 2-norm 1 and are mutually orthogonal, and any $z \in \mathbb{R}^i$, a vector in the column space of B, we have $\|z^T B\|_2 = \|z^T\|_2$.*

Lemma 11.8. *For any matrix A, $q \neq 0$, we have*

$$\|q^T A^T\|_2 \geq \|q\|_2 \cdot \sigma_k(A^T).$$

Proof. We prove that

$$\frac{\|q^T A^T\|_2}{\|q\|_2} \geq \min \frac{\|q^T A\|_2}{\|q\|_2} \geq \sigma_k(A),$$

from which follows the claim.

We have

$$\min_{q\neq 0} \frac{\|q^T A\|_2}{\|q\|_2} = \min_{q^T V^T \neq 0} \frac{\|q^T V\Sigma U^T\|_2}{\|q^T V\|_2} = \min_{y^T\neq 0} \frac{\|q^T V\Sigma\|_2}{\|q^T V\|_2} = \min_{y^T\neq 0} \frac{\|y^T\Sigma\|_2}{\|y^T\|_2}$$
$$= \min_{y\neq 0} \sqrt{\frac{\sum_{i=1}^n y_i^2\sigma_i^2}{\sum_{i=1}^n y_i^2}} \geq \sigma_k(A).$$

□

Lower Bound. We now prove the lower bound on $\|Z\,\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2$. From Lemma 11.8, we have

$$\|Z\,\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2 \geq \sigma(Z)\cdot\|\,\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2.$$

Since $\mathrm{diag}(p^T\widetilde{S}^T)$ is a diagonal matrix, we have

$$\sigma(Z)\cdot\|\,\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2 \geq \sigma(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\overline{v}^T\widetilde{A}^T\|_2.$$

From Claim 11.5, we know that

$$\overline{v}^T\widetilde{A}^T = \tilde{q}'^T\widetilde{M}^{-1}\widetilde{W} = \tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}U_{\widetilde{M}}^T U_{\widetilde{W}}\Sigma_{\widetilde{W}}V_{\widetilde{W}}^T.$$

Therefore, we have

$$\begin{aligned}\|Z\,\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2 &\geq \sigma(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\overline{v}^T\widetilde{A}^T\|_2\\ &\geq \sigma(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}U_{\widetilde{M}}^T U_{\widetilde{W}}\Sigma_{\widetilde{W}}V_{\widetilde{W}}^T\|_2\\ &\geq \sigma_k(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}U_{\widetilde{M}}^T U_{\widetilde{W}}\Sigma_{\widetilde{W}}\|_2\\ &\geq \sigma_k(Z)\cdot\sigma_k(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}U_{\widetilde{M}}^T U_{\widetilde{W}}\|_2.\end{aligned}$$

But since $(\tilde{q}'^T\widetilde{M}^{-1})^T$ is in the column space of $\widetilde{W}$ (consequently, $U_{\widetilde{W}}$) from Claim 11.6, and using Lemma 11.8, we may continue the chain of inequalities

$$\geq \sigma_k(Z)\cdot\sigma_k(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}U_{\widetilde{M}}^T\|_2.$$

Furthermore, we may continue with

$$\begin{aligned}&\geq\sigma_k(Z)\cdot\sigma_k(Z)\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\Sigma_{\widetilde{M}}^{-1}\|_2 \text{ (Since } V_{\widetilde{W}}^T \text{ is orthogonal)}\\ &\geq \sigma_k(Z)\cdot\sigma_k(\widetilde{W})/\sigma_1(\widetilde{M})\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T V_{\widetilde{M}}\|_2.\end{aligned}$$

Finally, since $\tilde{q}'^T$ is in the row space of $\widetilde{M}$ (consequently in the column space of $V_{\widetilde{M}}$) from Claim 11.1, we have

$$\geq \sigma_k(Z)\cdot\sigma_k(\widetilde{W})/\sigma_1(\widetilde{M})\cdot\min(p^T\widetilde{S}^T)\cdot\|\tilde{q}'^T\|_2.$$

Upper Bound. We now prove the upper bound. Using the equality of Claim 11.5, we have $\overline{v}^T\widetilde{A}^T = \tilde{q}'^T\widetilde{M}^{-1}\widetilde{W}$. If we let $e\in R^{m+l}$ be such that $\overline{q'}^T = \widetilde{q'}^T + e$, let $E_{\widetilde{M}^{-1}}\in R^{(m\times l)\times m}$ be such that $(\overline{M}_r^*)^{-1} = \widetilde{M}^{-1} + E_{\widetilde{M}^{-1}}$, let $E_{\widetilde{W}}\in R^{m\times m}$, $\overline{W}_r^* = \widetilde{W} + E_{\widetilde{W}}$, and let $E_{\widetilde{S}^T}\in R^{m\times l}$ be such that $\overline{S}_o^{*T} = \widetilde{S}^T + E_{\widetilde{S}^T}$, then we can write

$$\begin{aligned}&\|Z\cdot\mathrm{diag}(p^T\cdot\overline{S}_o^{*T})\overline{q'}^T\overline{M}_r^{*-1}\overline{W}_t^* - Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2\\ &\quad=\|Z\cdot\mathrm{diag}(p^T\cdot(\widetilde{S}^T+E_{\widetilde{S}^T}))\cdot((\tilde{q}'^T+e^T)(\widetilde{M}^{-1}+E_{\widetilde{M}^{-1}})(\widetilde{W}+E_{\widetilde{W}})\\ &\qquad - Z\,\mathrm{diag}(p^T\widetilde{S}^T)\tilde{q}'^T\widetilde{M}^{-1}\widetilde{W}\|_2\\ &\quad\leq\|Z\cdot\mathrm{diag}(p^T(\widetilde{S}^T)+E_{\widetilde{S}^T})(e^T\widetilde{M}^{-1}\widetilde{W})\|_2 &(11.2)\\ &\qquad+\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))\cdot(\tilde{q}'^T\widetilde{M}^{-1}E_{\widetilde{W}}+e^T\widetilde{M}^{-1}E_{\widetilde{W}})\|_2 &(11.3)\\ &\qquad+\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}}\widetilde{W}+e^T E_{\widetilde{M}^{-1}}\widetilde{W})\|_2 &(11.4)\\ &\qquad+\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}}+e^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}})\|_2. &(11.5)\end{aligned}$$

To prove upper bounds for (11.2), (11.3), (11.4), and (11.5), we need some additional claims.

Claim 11.9. *For any fixed matrix $B \in \mathbb{R}^{(m+l)\times j}$ with constant rank i, $\|e^T B\|_2 \leq O(1)\sqrt{i} \cdot \sigma_1(B)$ with high probability.*

Claim 11.10. *If $\sigma_i(\widetilde{W}) \in \omega(\sqrt{m})$ for $1 \leq i \leq k$, then PSP chooses $r = k$ and $E_{\widetilde{W}} \in O(\sqrt{m})$ with high probability. Similarly, if $\sigma_i(\widetilde{M}) \in \omega(\sqrt{m+l})$ for $1 \geq i \geq 2k$, then PSP chooses $m = 2k$ and $\|E_{\widetilde{M}}\|_2 \in O(\sqrt{m+l})$ with high probability, where $E_{\widetilde{M}}$ is the matrix such that $\overline{M}_r = \widetilde{M} + E_{\widetilde{M}}$. Finally, if $\sigma_i(\widetilde{S}^T) \in \omega(\sqrt{m})$ for $1 \geq i \geq k$, then PSP chooses $r = k$ and $\|E_{\widetilde{S}^T}\|_2 \in O(\sqrt{m})$ with high probability.*

Claim 11.11. *If $\sigma_i(M) \in \omega(\sqrt{m+l})$ for $1 \leq i \leq 2k$, then with high probability*

$$\|E_{\widetilde{M}^{-1}}\|_2 \leq \frac{O(\sqrt{m+l})}{(\sigma_{2k}(\widetilde{M}))^2}.$$

The proofs for the above claims are very similar to the those of similar results in [Achilioptas et al. 01].

Claim 11.12. *We have*

$$\frac{\|Z \cdot \operatorname{diag}(p^T(\widetilde{S}^T + E_{\widetilde{S}^T}))(e^T\widetilde{M}^{-1}\widetilde{W})\|_2}{\|Z \cdot \operatorname{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2} \leq O(1) \cdot \frac{(\max(p^T\widetilde{S}^T) + \max(p^T)}{\min(p^T\widetilde{S}^T)}.$$

Proof. We have

$$\begin{aligned}
\frac{\|Z \cdot \operatorname{diag}(p^T(\widetilde{S}^T + E_{\widetilde{S}^T}))(e^T\widetilde{M}^{-1}\widetilde{W})\|_2}{\|Z \cdot \operatorname{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2} &\leq \frac{\|Z\|_2\|p^T\|_2\|(\widetilde{S}^T + E_{\widetilde{S}^T}\|_2\|e^T\widetilde{M}^{-1}\widetilde{W}\|_2}{\|Z\operatorname{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2} \\
&\leq \frac{((\sigma_1(Z)(\max(p^T\widetilde{S}^T) + \max(p^T)O(\sqrt{m}))O(1)\sqrt{\operatorname{rank}(\widetilde{M^{-1}})}\sigma_1(\widetilde{M}^{-1})\sigma_1(\widetilde{W})}{\|Z\operatorname{diag}(p^T\widetilde{S}^T)\overline{v}^T\widetilde{A}^T\|_2} \\
&\leq \frac{(\sigma_1(Z)(\max(p^T\widetilde{S}^T) + \max(p^T)O(\sqrt{m}))O(1)\sqrt{\operatorname{rank}(\widetilde{M^{-1}})}\sigma_1(\widetilde{M}^{-1})\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})}{\sigma_k(Z)\min(p^T\widetilde{S}^T)\|\tilde{q}'^T\|_2\sigma_k(\widetilde{W})} \\
&\leq \frac{r_k(Z)(\max(p^T\widetilde{S}^T) + \max(p^T)O(\sqrt{m}))O(1)\sqrt{\operatorname{rank}(\widetilde{M^{-1}})}\sigma_1(\widetilde{M}^{-1})\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})}{r_k(Z)\min(p^T\widetilde{S}^T)\sqrt{K}r_k(\widetilde{W})r_{2k}(\widetilde{M})\sigma_k(\widetilde{W})} \\
&\leq \frac{O(1)\sqrt{\operatorname{rank}(\widetilde{M^{-1}})}\sigma_1(\widetilde{M}^{-1})\sigma_1(\widetilde{M})((\max(p^T\widetilde{S}^T) + \max(p^T)O(\sqrt{m})))}{\sqrt{K}r_{2k}(\widetilde{M})\sigma_k(\widetilde{W})\min(p^T\widetilde{S}^T)} \\
&\leq O(1) \cdot \frac{(\max(p^T\widetilde{S}^T) + \max(p^T)}{\min(p^T\widetilde{S}^T)}.
\end{aligned}$$

□

Claim 11.13. *We have*

$$\frac{\|Z \cdot \mathrm{diag}((p^T)(\widetilde{S}^T + E_{\widetilde{S}^T}))(\tilde{q}'^T \widetilde{M}^{-1} E_{\widetilde{W}} + e^T \widetilde{M}^{-1} E_{\widetilde{W}})\|_2}{\|Z \cdot \mathrm{diag}(p^T \widetilde{S}^T)\overline{v}^T \widetilde{A}^T\|_2} \leq O(1) \cdot \frac{(\max(p^T \widetilde{S}^T) + \max(p^T)}{\min(p^T \widetilde{S}^T))}.$$

Proof.

$$\begin{aligned}
&\frac{\|Z \cdot \mathrm{diag}((p^T)(\widetilde{S}^T + E_{\widetilde{S}^T}))(\tilde{q}'^T \widetilde{M}^{-1} E_{\widetilde{W}} + e^T \widetilde{M}^{-1} E_{\widetilde{W}})\|_2}{\|Z \cdot \mathrm{diag}(p^T \widetilde{S}^T) \cdot \overline{v}^T \widetilde{A}^T\|_2} \\
&\leq \frac{\|Z \cdot \mathrm{diag}(p^T(\widetilde{S}^T + E_{\widetilde{S}^T}))\|_2 \cdot (\|(\tilde{q}'^T \widetilde{M}^{-1} E_{\widetilde{W}})\|_2 + \|e^T \widetilde{M}^{-1} E_{\widetilde{W}}\|_2)}{\|Z \cdot \mathrm{diag}(p^T \widetilde{S}^T) \cdot \overline{v}^T \widetilde{A}^T\|_2} \\
&\leq \frac{\sigma_1(Z)(\max(p^T \widetilde{S}^T) + \max(p^T) O(\sqrt{m}))}{\|Z \cdot \mathrm{diag}(p^T \widetilde{S}^T) \cdot \overline{v}^T \widetilde{A}^T\|_2} \\
&\quad \times \left(\frac{\|\tilde{q}'^T\|_2 O(\sqrt{m})}{\sigma_{2k}(M)} + O(1)\sqrt{\mathrm{rank}(\widetilde{M}^{-1})} O(\sqrt{M}) \sigma_1(\widetilde{M}^{-1}) \right) \\
&\leq \frac{(\max(p^T \widetilde{S}^T) + \max(p^T))}{\min(p^T \widetilde{S}^T)} \frac{\sigma_1(Z)}{\sigma_k(Z)} \\
&\quad \times \left(\frac{O(\sqrt{m})\sigma_1(\widetilde{M})}{\sigma_{2k}(\widetilde{M})\sigma_k(\widetilde{W})} + \frac{O(1)\sqrt{\mathrm{rank}(\widetilde{M}^{-1}}\sigma_1(\widetilde{M}^{-1} O(\sqrt{m})\sigma_1(\widetilde{M})\sigma_1(Z)}{\sigma_k(\widetilde{W})\|\tilde{q}'^T\|_2 \sigma_k(Z)} \right) \\
&\leq \frac{(\max(p^T \widetilde{S}^T) + \max(p^T))}{\min(p^T \widetilde{S}^T)} \\
&\quad \times \left(\frac{O(\sqrt{m}) r_{2k}(\widetilde{M}) r_k(Z)}{\sigma_k(\widetilde{W})} + \frac{O(1)\sqrt{\mathrm{rank}(\widetilde{M}^{-1}}\sigma_1(\widetilde{M}^{-1} O(\sqrt{m})\sigma_1(\widetilde{M}) r_k(Z)}{\sigma_k(\widetilde{W})\sqrt{K} r_k(\widetilde{W}) r_{2k}(\widetilde{M} r_k(Z)} \right) \\
&\leq \frac{(\max(p^T \widetilde{S}^T) + \max(p^T))}{\min(p^T \widetilde{S}^T)} \left(O(1) + O(1) \frac{\sqrt{\mathrm{rank}(\widetilde{M}^{-1}} O(\sqrt{(m)})}{r_{2k}(\widetilde{M})\sqrt{m}\sqrt{k} r_k(\widetilde{W}) r_{2k}(\widetilde{M})} \right) \\
&\leq \frac{(\max(p^T \widetilde{S}^T) + \max(p^T))}{\min(p^T \widetilde{S}^T)} O(1).
\end{aligned}$$

□

Claim 11.14. *We have*

$$\frac{\|Z \cdot \mathrm{diag}((p^T)(\widetilde{S}^T + E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}} \widetilde{W} + e^T E_{\widetilde{M}^{-1}} \widetilde{W})\|_2}{\|Z \cdot \mathrm{diag}(p^T \widetilde{S}^T) \cdot \overline{v}^T \widetilde{A}^T\|_2} \leq O(1) \frac{(\max(p^T \widetilde{S}^T) + \max(p^T))}{\min(p^T \widetilde{S}^T)}.$$

Proof.

$$\begin{aligned}
&\frac{\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}}\widetilde{W}+e^T E_{\widetilde{M}^{-1}}\widetilde{W})\|_2}{\|Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\cdot\overline{v}^T\widetilde{A}^T\|_2}\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\|Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\cdot\overline{v}^T\widetilde{A}^T\|_2}\\
&\qquad\times\left(\frac{\|\tilde{q}'^T\|_2 O(\sqrt{m+l})\sigma_1(\widetilde{W})\sigma_1(Z)}{\sigma_{2k}(\widetilde{M})^2}+O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\cdot\sigma_1(E_{\widetilde{M}^{-1}})\sigma_1(\widetilde{W})\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{(m+l)})\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})\sigma_1(Z)}{\sigma_{2k}(\widetilde{M})^2\sigma_k(\widetilde{W})\sigma_k(Z)}\right.\\
&\qquad\qquad\left.+\frac{O(1)\sqrt{\mathrm{rank}(E_{M^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})\sigma_1(Z)}{\sigma_k(\widetilde{W})\|\tilde{q}'^T\|_2\sigma_k(Z)}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{(m+l)})r_k(\widetilde{W})r_{2k}(\widetilde{M})r_k(Z)}{\sigma_{2k}(\widetilde{M})}+\frac{O(1)\sqrt{\mathrm{rank}(E_{M^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})}{\sigma_k(\widetilde{W})\sqrt{k}r_k(\widetilde{W})r_{2k}(\widetilde{M})}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{(m+l)})r_k(\widetilde{W})r_{2k}(\widetilde{M})}{O(\sqrt{m})r_k(\widetilde{W})r_{2k}(\widetilde{M})}+\frac{O(1)\sqrt{\mathrm{rank}(E_{M^{-1}})}\|E_{\widetilde{M}^{-1}}\|_2\sigma_1(\widetilde{W})\sigma_1(\widetilde{M})}{\sigma_k(\widetilde{W})\sqrt{k}r_k(\widetilde{W})r_{2k}(\widetilde{M})}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\times\left(\frac{O(\sqrt{(m+l)})}{O(\sqrt{m})}+\frac{O(1)\sqrt{\mathrm{rank}(E_{M^{-1}})}O(\sqrt{m+l})\sigma_1(\widetilde{M})}{\sqrt{k}r_k(\widetilde{M})\sigma_{2k}(\widetilde{M})^2}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\times\left(\frac{O(\sqrt{(m+l)})}{O(\sqrt{m})}+\frac{O(1)\sqrt{\mathrm{rank}(E_{M^{-1}})}O(\sqrt{m+l})}{\sqrt{k}\sigma_{2k}(\widetilde{M})}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\times\left(\frac{O(\sqrt{(m+l)})}{O(\sqrt{m})}+\frac{\sqrt{\mathrm{rank}(E_{M^{-1}})}O(\sqrt{m+l})}{\sqrt{k}r_k(\widetilde{W})r_{2k}(\widetilde{M})O(\sqrt{m}}\cdot)\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}O(1).
\end{aligned}$$

□

Claim 11.15. *We have*

$$\frac{\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}}+e^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}})\|_2}{\|Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\cdot\overline{v}^T\widetilde{A}^T\|_2}\le O(1)\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}.$$

Proof.

$$
\begin{aligned}
&\frac{\|Z\cdot\mathrm{diag}((p^T)(\widetilde{S}^T+E_{\widetilde{S}^T}))(\tilde{q}'^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}}+e^T E_{\widetilde{M}^{-1}}E_{\widetilde{W}})\|_2}{\|Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\cdot\overline{v}^T\widetilde{A}^T\|_2}\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\|Z\cdot\mathrm{diag}(p^T\widetilde{S}^T)\cdot\overline{v}^T\widetilde{A}^T\|_2}\\
&\qquad\times\left(\|\tilde{q}'^T\|_2\frac{O(\sqrt{m+l})O(\sqrt{m})}{\sigma^2_{2k}(\widetilde{M})}\sigma_1(Z)\ +O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})O(\sqrt{m})\sigma_1(Z)\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{m+l})O(\sqrt{m})\sigma_1(\widetilde{M})\sigma_1(Z)}{\sigma^2_{2k}(\widetilde{M})\sigma_k(\widetilde{W})\sigma_k(Z)}\ +\frac{O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})O(\sqrt{m})\sigma_1(\widetilde{M})\sigma_1(Z))}{\|\tilde{q}'^T\|_2\sigma_k(\widetilde{W})\sigma_k(Z)}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{m+l})O(\sqrt{m})}{\sqrt{m}\sigma_{2k}(\widetilde{M})}\ +\frac{O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})O(\sqrt{m})\sigma_1(\widetilde{M})r_k(Z))}{\|\tilde{q}'^T\|_2\sigma_k(\widetilde{W})}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{m+l})O(\sqrt{m})}{\sqrt{m}\sigma_{2k}(\widetilde{M})}+\frac{O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})O(\sqrt{m})\sigma_1(\widetilde{M}))}{\sqrt{m}\cdot\sqrt{k}\cdot r_{2k}(\widetilde{M})}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{m+l})O(\sqrt{m})}{m\cdot r_{2k}(\widetilde{M})r_k(\widetilde{W})r_k(Z)}+\frac{O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(E_{\widetilde{M}^{-1}})O(\sqrt{m})\sigma_1(\widetilde{M}))}{\sqrt{m}\cdot\sqrt{k}\cdot r^2_{2k}(\widetilde{M})r_k(\widetilde{W})r_k(Z)}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}\\
&\qquad\times\left(\frac{O(\sqrt{m+l})}{m\cdot r_{2k}(\widetilde{M})r_k(\widetilde{W})r_k(Z)}+\frac{O(1)\sqrt{\mathrm{rank}(E_{\widetilde{M}^{-1}})}\sigma_1(\widetilde{M})O(\sqrt{m+l})}{\sqrt{m}\cdot\sqrt{k}\cdot r^2_{2k}(\widetilde{M})r_k(\widetilde{W})r_k(Z)\sigma_{2k}(\widetilde{M})^2}\right)\\
&\quad\le\frac{(\max(p^T\widetilde{S}^T)+\max(p^T))}{\min(p^T\widetilde{S}^T)}O(1).
\end{aligned}
$$

□

Combining Claims 11.12–11.15, we obtain the desired proof of the theorem.

Acknowledgments. We greatly appreciate the careful and very constructive comments of the referees.

References

[Achilioptas et al. 01] D. Achilioptas, A. Fiat, A. R. Karlin, and F. McSherry. "Web Search via Hub Synthesis." In *Proceedings of the 42nd IEEE Symposium on Founda-*

tions of Computer Science, pp. 500–509. Washignton, DC: IEEE Computer Society, 2001.

[Aktas et al. 04] M. Aktas, M. Nacar, and F. Menczer. "Personalizing PageRank Based on Domain Profiles." In *Proceedings of WebKDD 2004: KDD Workshop on Web Mining and Web Usage Analysis*, 2004. Available at http://maya.cs.depaul.edu/webkdd04/final/aktas.pdf.

[Borodin et al. 05] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. "Link Analysis Ranking: Algorithms, Theory, and Experiments." *ACM Trans. Internet Techn.* 5:1 (2005), 231–297.

[Brand 03] M. Brand. "Fast Online SVD Revisions for Lightweight Recommender Systems." In *Proceedings of the Third SIAM International Conference on Data Mining*, Vol. 3, pp. 37–46. Philadelphia: SIAM, 2003.

[Brin and Page 98] S. Brin and L. Page. "The Anatomy of a Large-Scale Hypertextual Web Search Engine." *Computer Networks* 30:1–7 (1998), 107–117.

[Chirita et al. 05] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschuetter. "Using ODP Metadata to Personalize Search." In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 178–185. New York: ACM, 2005.

[Deerwester et al. 90] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. "Indexing by Latent Semantic Analysis." *Journal of the Society for Information Science* 41:6 (1990), 391–407.

[Donato et al. 05] D. Donato, S. Leonardi, and P. Tsaparas. "Stability and Similarity of Link Analysis Ranking Algorithms." In *Automata, Languages and Programming: 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11–15, 2005, Proceedings*, Lecture Notes in Computer Science 3580, pp. 717–729. New York: Springer, 2005.

[Fagin et al. 03] R. Fagin, R. Kumar, and D. Sivakumar. "Comparing Top K Lists." In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 28–36. Philadelphia: SIAM, 2003.

[Ferragina and Gulli 05] P. Ferragina and A. Gulli. "A Personalized Search Engine Based on Web-Snippet Hierarchical Clustering." In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, pp. 801–810. New York: ACM, 2005.

[Haveliwala 03] T. H. Haveliwala. "Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search." *IEEE Transactions on Knowledge and Data Engineering* 15:4 (2003), 784–796.

[Jeh and Widom 03] G. Jeh and J. Widom. "Scaling Personalized Web Search." In *Proceedings of the 12th International Conference on World Wide Web*, pp. 271–279. New York: ACM, 2003.

[Kamvar et al. 03] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub. "Exploiting the Block Structure of the Web for Computing PageRank." Technical report, Stanford University, March 2003.

[Kleinberg 99] J. M. Kleinberg. "Authoritative Sources in a Hyperlinked Environment." *J. ACM* 46:5 (1999), 604–632.

[Lee and Borodin 03] H. C. Lee and A. Borodin. "Perturbation of the Hyper-linked Environment." In *Computing and Combinatorics: 9th Annual International Conference, COCOON 2003 Big Sky, MT, USA, July 25-28, 2003, Proceedings*, Lecture Notes in Computer Science 2697, pp. 272–283. Berlin: Springer, 2003.

[Liu et al. 02] F. Liu, C. T. Yu, and W. Meng. "Personalized Web Search by Mapping User Queries to Categories." In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, pp. 558–565. New York: ACM, 2002.

[Ng et al. 01] A. Y. Ng, A. X. Zheng, and M. I. Jordan. "Link Analysis, Eigenvectors and Stability." In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 903–910. San Francisco: Morgan Kaufmann, 2001.

[Qiu and Cho 06] F. Qiu and J. Cho. "Automatic Identification of User Interest for Personalized Search." In *Proceedings of the 15th International Conference on World Wide Web*, pp. 727–736. New York: ACM, 2006.

[Sun et al. 05] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. "CubeSVD: A Novel Approach to Personalized Web Search." In *Proceedings of the 14th International Conference on World Wide Web*, pp. 382–390. New York: ACM, 2005.

[Teevan et al. 05] J. Teevan, S. T. Dumais, and E. Horvitz. "Personalizing Search via Automated Analysis of Interests and Activities." In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 449–456. New York: ACM, 2005.

[Tsaparas 04] P. Tsaparas. "Using Non-linear Dynamical Systems for Web Searching and Ranking." In *Proceedings of the Twenty-Third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 59–70. New York: ACM, 2004.

Hyun Chul Lee, Thoora Inc., 350 Bloor St. East, Toronto, ON M4W 0A1, Canada (chul.lee@thoora.com)

Allan Borodin, Department of Computer Science, University of Toronto, 10 Kings College Road, Toronto, ON M5S 3G4, Canada (bor@cs.toronto.edu)

Received June 30, 2009; accepted June 3, 2010.