Link-Based Clustering for Finding Subrelevant Web Pages

Tomonari Masada, Atsuhiro Takasu, Jun Adachi
The National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku Tokyo, Japan
{masada, takasu, adachi} @nii.ac.jp

Abstract

We propose a new Web page clustering. Typical search engines only provide relevant pages, i.e., the pages matching users' needs. However, we design our clustering method to provide non-relevant pages as search results when they refer to relevant pages and help users anticipate the contents of those relevant pages. We call such pages subrelevant. As it is difficult to improve Web search performance, we use subrelevancy to relax the criterion as to what kind of pages should appear in search results with the least drawback, i.e., one click away from a relevant page. Our clustering method is based on three concepts: THP, out-degree path length, and threshold parameter. We use clustering results to modify the feature vectors of Web pages. Hence, each clustering result induces a reranking of search results. We expect the reranking to raise the ranks of subrelevant pages. In the experiments with NTCIR-3 Web task test collection, our clustering largely improved the average precision by 13 percent in comparison with the baseline.

1. Introduction

This paper proposes a new Web page clustering method. Typical search engines provide relevant Web pages, i.e., the Web pages matching users' needs, as search results. However, it is difficult to substantially improve the Web search without harming scalability. As regards NTCIR Web retrieval task[4], far better search results are obtained with pseudo-relevance feedback and query expansion[11]. However, pseudo-relevance feedback makes search engines conduct the corpus twice for one query, and query expansion will add hundreds of terms to a original query including only a few terms. Therefore, we relax the criterion as to what kind of Web pages should appear in search results. We design our clustering method so that search results include non-relevant pages when the pages satisfy the following two conditions: (1) They refer to at least one relevant page, and (2) they allow users to make a good guess at what kind of pages they refer to. We call this kind of non-relevant pages subrelevant. The first condition says that some relevant pages may be reached only after clicking a hyperlink contained in one of the subrelevant pages in search results. As one click is the only price users have to pay, we do not excessively sacrifice the convenience. The second condition is required, because we should not call all Web pages referring to relevant pages subrelevant. For example, based on a Web page where a lot of spam links appear, we cannot make a good guess at what kind of Web pages are referred to. Therefore, such a spam-like page is not subrelevant even if it refers to relevant pages. By contrast, a table-of-contents like page is a promising example of subrelevant page.

2. Our clustering method

We propose a clustering method ranking subrelevant pages higher than spam-like ones. Our method places subrelevant pages in the clusters containing relevant pages, and keeps spam-like pages away from such clusters. Then we use clustering results to modify the feature vectors of Web pages so that the pages belonging to the same clusters come to have the vectors resembling each other. Therefore, only the feature vectors of subrelevant pages come to resemble those of relevant ones. As long as we use an adequate term weighting, e.g. TF-IDF[1], relevant pages are ranked high. Hence, we can raise the ranks of subrelevant pages. Our clustering is based on three concepts: THP (Two Hop return Probability), out-degree path length, and threshold parameter τ.

THP[8] is a real value between 0 and 1 assigned to every Web page. THP of a Web page v is defined by $\mathrm{THP}_v \equiv \sum_u$ is referred to by $_v\, 1/(d_v^+ \cdot d_u^+)$, where d_v^+ denotes the out-degree, i.e., the number of out-going

hyperlinks, of v. Intuitively speaking, a page of large THP occupies a central position among the Web pages forming a relatively independent group in the hyperlinked environment. Out-degree path length approximates the degree of difficulty in anticipating the contents of the Web pages we can reach by clicking hyperlinks. The out-degree path length is defined to be the out-degree sum of the pages forming the path except the terminal page. We assign this length only to the paths formed by the hyperlinks of the same direction. We reserve the term "path" only for a sequence of hyperlinks of the same direction. A cycle is a path whose initial and terminal pages are the same. Threshold parameter τ affects cluster granularity. As τ increases, the paths (or cycles) our algorithm enumerates get longer, and thus clusters get larger. Selecting too small τ results in too small clusters to include both subrelevant and relevant pages. Selecting too large τ results in too large clusters to keep non-subrelevant pages away from relevant ones. To see how these concepts work, we describe our clustering algorithm.

1. Select a seed page in decreasing order of THP from those still belonging to no clusters.
2. Among the shortest paths (or cycles) passing through the seed page, enumerate all paths (or cycles) of out-degree path length $\leq \tau$.
3. Make a cluster including all pages lying on any one of the enumerated paths (or cycles) in Step 2.
4. Repeat from Step 1 to 3 until every Web page belongs to at least one cluster.

We show another option of our clustering algorithm. In Step 2, the shortest paths (or cycles) are enumerated. We propose three types of enumerations (Figure 1). (1) Fan-in enumeration: enumerate every shortest path, terminating at the seed page, of length $\leq \tau$. (2) Fan-out enumeration: enumerate every shortest path, originating from the seed page, of length $\leq \tau$. (3) Cyclic enumeration: enumerate every shortest cycle, passing through the seed page, of length $\leq \tau$. Three types of clusterings induced by these three types of enumerations are called fan-in clustering, fan-out clustering, and cyclic clustering, respectively.

We analyze the time complexity. In Step 2, Dijkstra's algorithm [3] is called. We used a trinomial heap[10] and reduced the time complexity of Dijkstra's algorithm to $O(m + n \log n)$, where n is the number of Web pages and m the number of hyperlinks. Our method enumerates only the shortest paths of limited length, i.e., $\leq \tau$. Hence, the actual time complexity will be of order far less than $O(m+n \log n)$. As the number of Web pages selected as seed pages is at most n, the total time complexity is bounded by $O(mn+n^2 \log n)$. We use no textual contents of Web pages and use only the hyperlink information in describing our clustering algorithm. This fact leads to computational efficiency in comparison with text-based clusterings.

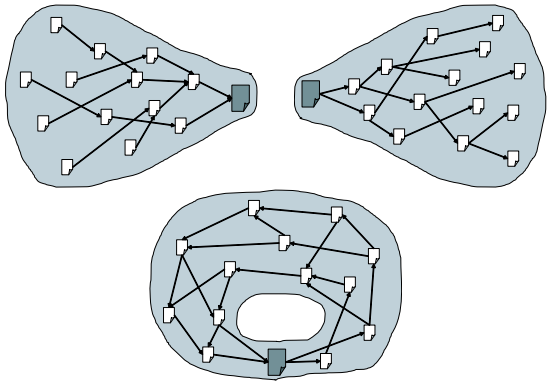

Figure 1. Three types of clusterings: fan-in (top left), fan-out (top right), and cyclic (bottom)

We use clustering results to make the feature vectors of the Web pages in the same clusters resemble each other. In this paper, we compute the feature vectors based on TF-IDF[1] and modify them by using a clustering result as follows. Let $\mathbf{x}_v$ be the feature vector of a page v. Suppose v belongs to a cluster C. Define the representative vector $\mathbf{y}_C$ of C to be $\mathbf{y}_C(i) \equiv \max_{u \in C} \mathbf{x}_u(i)$, where $\mathbf{y}_C(i)$ denotes the i-th entry of $\mathbf{y}_C$. Then we obtain the modified feature vector $\mathbf{x}'_v$ of v as $(1-\alpha) \cdot \mathbf{x}_v + \alpha \cdot \mathbf{y}_C$. The parameter $\alpha, 0 \leq \alpha \leq 1$ is used to control how strongly clustering information affects feature vector modification. This style of feature vector modification is proposed by [6]. We call α mixture ratio. When v belongs to more than one cluster $\mathcal{C} = \{C_1, \ldots, C_k\}$, we modify $\mathbf{x}_v$ by $\mathbf{x}'_v \equiv (1-\alpha) \cdot \mathbf{x}_v + \alpha \cdot \mathbf{y}_{\mathcal{C}}$, where i-th entry of $\mathbf{y}_{\mathcal{C}}$ is set to $\max_{C_j \in \mathcal{C}} \mathbf{y}_{C_j}(i)$.

3. Previous works

Sugiyama et al.[9] also use clustering results to modify the feature vectors. However, they use inter-document similarity obtained as feature vector similarity. Therefore, clustering cannot proceed in parallel with feature vector computation. By contrast, our clustering only uses hyperlink information. Kleinberg proposes a Web page clustering by applying his Hub/Authority analysis[7]. We can regard the World Wide Web as a gigantic digraph, where Web pages are vertices, and hyperlinks are directed edges. Let A be the adjacency matrix of this digraph. Kleinberg's method makes clusters by using a spectral decomposi-

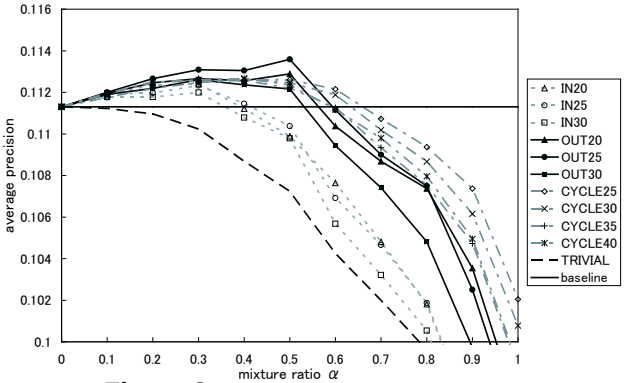

Figure 2. `qrel-fml.cont-all.lst`

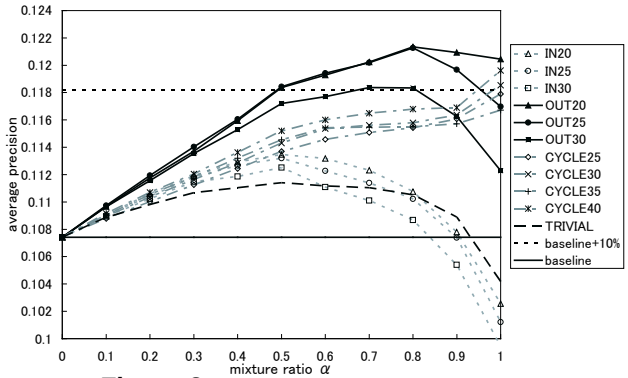

Figure 3. `qrel-fml.wlink-all.lst`

tion of AA^T or A^TA. As this method uses no textual contents of Web pages, clustering and feature vector computation can proceed in parallel. However, this method often gives a clustering result from which we cannot intuitively understand what the algorithm aims to achieve[2]. The clustering in [5] measures the proximity among Web pages with shortest path lengths. For each Web page, their method computes a vector with entries representing the shortest path lengths to all other pages. The inner products among these vectors measure proximity among Web pages. However, computing the shortest path length for every pair of Web pages is expensive. By contrast, our clustering enumerates only the shortest paths of length $\leq \tau$.

4. Evaluation experiments

We conducted evaluation experiments with NTCIR-3 Web task test collection[4]. We preprocessed the collection data as follows. First, we deleted the hyperlinks referring to the pages in different sites. In this paper, we mean by site a set of Web pages whose URLs are the same when compared by their prefixes which end at "`/`" appearing first after "`http://`". Preliminary experiments revealed that the hyperlinks to different sites provided almost no help. This deletion enhances parallelization of clustering. Next, as the collection data included the Web pages with no out-going links and those with no in-coming links, we added the following hyperlinks. For each page v of zero out-degree, we added a link from v to every page referring to v. For each page v of zero in-degree, we added a link referring to v from every page v refers. Consequently, we have 10,949,316 Web pages and 53,711,674 hyperlinks. For search result evaluation, we used two document sets: `qrel-fml.cont-all.lst` and `qrel-fml.wlink-all.lst`[4]. These are constructed by human assessors based on two different document models: page-unit document model and one-click-distance document model, respectively. When page-unit document model is adopted, assessors judge the relevance of a Web page only by conducting its textual contents. By contrast, when one-click-distance document model is adopted, assessors judge the relevance of a page not only by conducting its contents, but also by conducting the contents of the Web pages referred to by it. Of course, not all pages which refer to the pages relevant under page-unit document model receive a judgment of relevance under one-click-distance document model. Human assessors deliberately select only the pages that can help users when displayed in search results. Therefore, we can check if our method can raise the ranks of subrelevant pages with `qrel-fml.wlink-all.lst`.

We tested all three types of clusterings. As for fan-in and fan-out clustering, we set threshold parameter τ to 20, 25, and 30. As for cyclic clustering, we set τ to 25, 30, 35, and 40. We also tested a clustering, called trivial clustering, for comparison. Trivial clustering constructs for each Web page a cluster containing the page itself and all pages referred to by it. If assessors judge all Web pages referring to the pages relevant under page-unit document model as relevant also under one-click-distance document model, trivial clustering will greatly help in realizing effective Web search. After all, we obtained 11 clustering results in total. Our clustering algorithm ran in parallel on ten workstations with two Xeon 2.80 GHz processors and 6 GByte memory. We used each clustering result to modify the feature vectors of Web pages. After executing morphological analysis with MeCab[1], we computed a feature vector $\mathbf{x}_v$ of a Web page v as $\mathbf{x}_v(t) \equiv (1+\log(1+\log(\mathrm{TF}_v(t)+1))) \cdot (\frac{n}{\mathrm{DF}(t)})^{0.2}$, where $\mathrm{TF}_v(t)$ is the number of times the term t appears in v, $\mathrm{DF}(t)$ the number of Web pages where t appears, and

[1] `http://chasen.org/~taku/software/mecab/`

$\mathbf{x}_v(t)$ the entry value of $\mathbf{x}_v$ corresponding to t. We measured the relevance of a Web page v to a query with inner product. Different mixture ratio α induces different ranking. For each of 11 clustering results, we set α from 0.1 to 1.0 in steps of 0.1, and obtained 110 rankings. The top 1,000 documents from each ranking were evaluated with `qrel-fml.cont-all.lst` and `qrel-fml.wlink-all.lst`. When $\alpha = 0$, we obtain the baseline search result. We adopted average precision as evaluation measure. The definition of average precision appears in [4]. We took the average of the average precisions over all 47 queries prepared for NTCIR-3 Web survey task.

Figure 2 and 3 present the results. The horizontal axis represents α, and the vertical axis average precision. In the legend, a set of average precisions, obtained by varying α with a fixed clustering result, is denoted by a string composed of uppercase alphabets and digits. The alphabets tell a clustering method, and the digits a value of threshold parameter. For example, "IN25" denotes the average precisions obtained when we used a clustering result of a fan-in clustering with $\tau = 25$ to modify the feature vectors with various settings of mixture ratio. "TRIVIAL" denotes the average precisions obtained when we modified the feature vectors by using a result of trivial clustering. When we used `qrel-fml.cont-all.lst` as the set of correct search results, we obtained average precisions depicted in Figure 2. The figure shows that average precisions get worse for all clustering results as α increases. Our clustering method failed to raise the ranks of pages relevant under page-unit document model. However, our aim is to raise the ranks of subrelevant Web pages. Therefore, we can use this result to differentiate the problem we are addressing. In fact, Figure 3 shows our success. The figure includes the results when we used `qrel-fml.wlink-all.lst`. As α increases, average precisions get better. OUT20 and OUT25 result in the best average precision 0.1213, about 13 percent raise of the baseline, when $\alpha = 0.8$. The execution times of this most successful fan-out clustering was about 20 hours, nearly one-fifth of the time required for morphological analysis. While cyclic clustering could not give the best search result, it showed a stable behavior for a wide variety of τ. Fan-in and trivial clustering could not remarkably improve the average precison.

5. Conclusion

This paper proposed a novel clustering algorithm, which aims to raise the search result ranks of subrelevant Web pages. We could also prove the efficiency of our algorithm with respect to NTCIR-3 Web task test collection under the one-click-distance document model. It is well-known that the retrieval performance can vary significantly depending on test collections. Therefore, the most important future work is to check if our method can improve the Web search performance with respect to other test collections. Note that we should use a test collection distributed with a set of correct search results human assessors construct by taking into account not only the relevancy of textual contents but also the usefulness of hyperlinks as in case of NTCIR-3 Web's `qrel-fml.wlink-all.lst`.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison Wesley Longman, 1999.

[2] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas. Finding authorities and hubs from link structures on the World Wide Web. In Proceedings of the 10th International World Wide Web Conference, pages 415–429, 2001.

[3] E. W. Dijkstra. A note on two problems in connexion with graphs. Numerische Mathematik, 1:269–271, 1959.

[4] K. Eguchi, K. Oyama, E. Ishida, N. Kando, and K. Kuriyama. Overview of the Web retrieval task at the third NTCIR Workshop. In Proceedings of the 3rd NTCIR Workshop, 2003.

[5] J. Hou and Y. Zhang. Utilizing hyperlink transitivity to improve Web page clustering. In Proceedings of the 14th Australasian Database Conference on Database Technologies, pages 49–57, 2003.

[6] T. Kanazawa, A. Aizawa, A. Takasu, and J. Adachi. The effects of the relevance-based superimposition model in cross-language information retrieval. In Proceedings of the 5th European Conference on Digital Libraries, pages 312–324, 2001.

[7] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM, 46(5):604–632, 1999.

[8] T. Ramaswamy, B. Gedik, and L. Liu. Connectivity based node clustering in decentralized peer-to-peer networks. In Proceedings of the 3rd International Conference on Peer-to-Peer Computing (P2P-2003), pages 66–73, 2003.

[9] K. Sugiyama, K. Hatano, M. Yoshikawa, and S. Uemura. Refinement of TF-IDF schemes for Web pages using their hyperlinked neighboring pages. In Proceedings of the 14th ACM Conference on Hypertext and Hypermedia, pages 198–207, 2003.

[10] T. Takaoka. Theory of trinomial heaps. In Proceedings of the 6th Annual International Computing and Combinatorics Conference (COCOON 2000), pages 362–372, 2000.

[11] M. Toyoda, M. Kitsuregawa, H. Mano, H. Itoh, and Y. Ogawa. University of Tokyo/RICOH at NTCIR-3 Web retrieval task. In Proceedings of the 3rd NTCIR Workshop, 2002.