Using Element And Document Profile For Information Clustering

Jun Lai and Ben Soh
Department of Computer Science and Computer Engineering
La Trobe University Bundoora, VIC, Australia 3083
jun@cs.latrobe.edu.au *ben@cs.latrobe.edu.au*

Abstract

The tremendous growth in the amount of information available and the number of visitors to web sites in the recent years poses some key challenges for information filtering and retrieval. Web visitors not only expect high quality and relevant information, but also wish that the information be presented in an as efficient way as possible. The traditional filtering methods, however, only consider the relevant values of document. These conventional methods fail to consider the efficiency of documents retrieval. In this paper, we propose a new algorithm to calculate an index called document similarity score based on elements of the document. Using the index, document profile will be derived. Any documents with the similarity score above a given threshold will be clustered. Using these pre-clustered documents, information filtering and retrieval can be made more efficient.

Keywords: elements, clustering, information filtering, information retrieval, search engine, World Wide Web.

1. Introduction

The amount of information in the world is increasing far more quickly than our ability to process it. All of us have known the feeling of being overwhelmed by the number of new books, journal articles, and conference proceedings coming out each year. There is a need for new technologies that can help us sift through all the available information in a more efficient way.

Generally, there are three existing information filtering methods:

- Content-based filtering : Here, the system searches for items similar to those the user prefers based on a comparison of content using text-learning methods. Only the content and properties of a document contribute to the filtering, and each user operates independently. However, this approach has difficulty capturing different types of content and has problem of over-specialization.
- Collaborative: Documents are recommended for a user based on the likes of other users with similar tastes. User profiles are used to compare with each other. The major drawback of this method is that if a user whose taste is unusual would not get high quality recommendation.
- Rule-based filtering: It uses demographic or collected data of users to build user profiles and then define a set of rules to tailor the content delivery based on the facts specified in the user profiles. However, the creation and maintenance of rules are generally manual, as the system gets complicated, there will be difficulties managing it without conflict of logics.

All above conventional filtering systems consider

Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04)
0-7695-2073-1/04 $20.00 © 2004 **IEEE**

only the relevance and importance to the users in different ways without due consideration for efficiency. However, as the system gets complicated, the efficiency is also crucial. Surveys have shown that about 85% of Internet users using search engines and search service to find specific information are not satisfied with the performance of the current generation of search systems. The dissatisfaction arises from the slow retrieval speed, communication delays and poor quality of retrieved results [1].

In this paper, we propose a new efficient method called elements and document profile based information clustering which can cluster more than two documents based on elements of the documents. This method applies an algorithm to compute a correlation score of documents. The documents with the similarity score above a given threshold will be clustered together. A definition of document profile is derived. Then, all the documents are clustered based on the document profile. Our proposed algorithm computes the scores independently of the number of documents.

2. Restructuring Operation

Currently, there are number of metadata standards proposed for web pages. Among them are two well-publicized, solid efforts: the Dublin Core Metadata standard and the Warwick frame-word. The Dublin Core is a 15-Element Metadata Element Set proposed to facilitate fast and accurate information retrieval on the Internet [3]. In the paper, we propose a restructuring operation by using key words appearing in the elements of the documents. Those keywords will be matching the search query when web visitors search on the Internet. Keywords have different weight based on in which element this keyword is, ranging from 0 to 1. The weight is calibrated by the system designer based on the importance and relevance of the keyword. For example, the keyword appearing in the title should have a higher weight than the one appearing in the content.

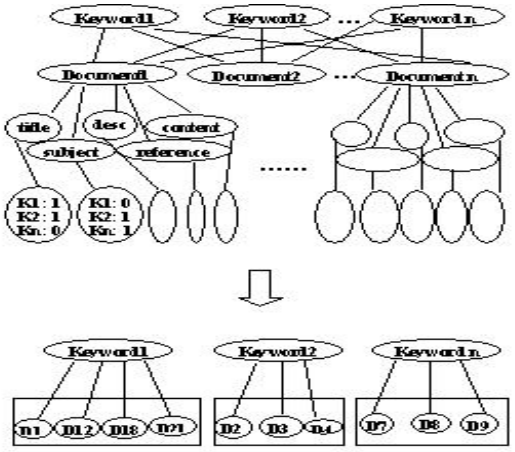

Figure 1. Restructuring operation of documents

Figure 1 shows the idea of restructuring operation of documents. On the top level of this tree structure, there are keywords in some documents, which are on the second level. The third level contains elements of the document. The numbers on the bottom level represent the number of times that keyword appears in that element. The documents are clustered based on the word in the elements of the document after the restructuring operation (The proposed new algorithm will be further discussed in the next section). In this way, when web visitors enter a keyword to search information, the pre-clustered documents will be presented in an efficient way.

3. Document Clustering

We have proposed a restructuring operation to cluster documents in section 2. In this section, we will discuss the algorithm and technique of documents clustering. The subsections are organized as follows: section 3.1 presents the approach of calculation of similarity score. Then the document profiles will be derived in section 3.2. Finally, the k-time clustering algorithm will be applied.

3.1 Computation of similarity score

To compute the similarity score of documents, we initially select some keywords appearing in those documents. Each word is assigned a weight, ranging form 0 to 1. Different word has different weight based on in which element that keyword appears. The value of weight is designed by the system administrator. For example, the value of weight in the title should be higher than in the description. The number of times that word appearing in the document also affects the value of relevance among documents.

Table 1 shows the number of times a keyword appears in the elements of a particular document which is indicated by document ID (DID).

Table 1. Number of times keywords appear in the elements of a particular documents

DID	21	45	567	789
Title	1	1	0	0
Subject	2	1	0	1
Description	2	7	3	7
Content	10	4	1	4
Reference	9	8	1	7

The similarity score is the sum of all product of keyword weight and the number of times that keyword appears in the document computed by the following formula:

$$SS_{(d,c)} = \sum_{j=1}^{n} (W_{Kj} * Count_{Kj}) \quad (1)$$

- SS is the similarity score of document based on keyword K.
- K j is the keyword in that document ($1 \leq j \leq n$).
- W_{Kj} is the pre-defined weight of the keyword K_j, determined by system admin.
- $Count_{Kj}$ is the number of times that keyword appearing in the document.

For instance, for the document 21, the word filtering appears in the title once, in the subject twice, in the description twice, in the content 10 times and in the reference 9 times:

$SS_{(21,\ filtering)}$ = (0.99*1+0.8*2+0.7*2+0.5*10+0.4*9) = 12.59

We can have table 2 based on formula (1).

Table 2. Similarity score of documents

DID	21	45	567	789
Title	1	1	0	0
Subject	2	1	0	1
Description	2	7	3	7
Content	10	4	1	4
Reference	9	8	1	7
SS	12.59	11.89	3	10.5

Table 3 shows the similarity score document 21 in term of keyword filtering, clustering, RDF and item respectively:

Table 3. Similarity score of document 21

DID	21	21	21	21
Keyword	filtering	clustering	rdf	item
Title	1	1	0	0
Subject	2	1	0	1
Description	2	7	1	6
Content	10	5	3	4
Reference	9	6	1	5
SS	12.59	11.59	2.6	9

3.2 Deriving document profile (DP)

From the calculation of similarity score, the document profile can be derived as follows:

$DP_{(d)} = \{ (k, SS_{(d,c)} \mid k \in C,\ 0 \leq SS_{(d,c)} \leq SS_{threshold} \}$ (2)

- k denotes a keyword.
- K is all keywords to which the document can be related.
- SS is the similarity score for document d.
- $SS_{threshold}$ is the minimum similarity score acceptable for a document to relate to that keyword.

From formula (2), each document can have a profile based on the keywords and similarity score calculated by formula (1). For example, if a given threshold is 9, the profile of document 21 is:

$DP_{(21)}$ ={(filtering,12.59),(clustering,11.59),(item, 9)}

3.3 Clustering Algorithm

Using the document profile, we can measure the correlation similarity score among documents. Table 4 shows the document profile.

Table 4: Document profile

DP	21	45	567	789
filtering	12.59	11.89	0	10.5
clustering	11.59	0	0	18.1
RDF	0	10.3	23	0
item	9	0	12.3	12.4

Table 4 shows the similarity score of each document in term of different keyword. There are various clustering algorithms available, we chose K-mean [4]. We have defined our input data set for a general clustering already. Hence, any algorithm can be applied. K-mean algorithm splits a set of objects into a selected number of groups. The basic idea of K-mean is to find a single partition of the data, which has K number of clusters such that objects within clusters are close to each other in some sense, and those in different clusters are distant. The object of clustering, in our case, is the document and the keyword appearing in the elements of the documents. Therefore, the documents in the same cluster will be considered as relevant to that keyword.

From the K-mean clustering, we will have K number of clusters. The documents belonging to the same cluster have the relevant information. For example, if the given threshold is 10, then the document 21 is not relevant to the keyword of RDF and item. The final pass of the algorithm produces the clustering of (21, 45) for keyword of filtering, (45, 567) for keyword of RDF, (567, 789) for keyword of item.

In this way, documents are pre-clustered in term of keywords. If a document in a cluster is relevant to the search, then the all other documents in the same cluster are relevant, which can make information retrieval more efficient. The efficiency of comparing other methods will be carried out in the future work.

4. Conclusions and Future Work

In this paper, we propose elements and document profile based information clustering method. This new method computes the correlation similarity score among documents. The document with similarity score above a given threshold will be clustered. Then we derive the document profile based on the similarity score. Therefore, the document will be clustered based on different keywords. Our approach computes similarity score and derives document profile offline. Ideally, the documents are pre-clustered in that it will improve the efficiency of information retrieval. For the future, we need to do the experiment in efficiency comparison and the optimization of this method.

5. References

1. Kobayashi, M and Takeda, K, "Information Retrieval on the Web", *ESSIR 2000, LNCS 1980*, Springer-Verlag 2000, pp. 242-285.
2. Meng, X and Chen, Z, "Personalized Web Search With Clusters", *International Conference on Internet Computing, IC'03*, 2003, pp. 46-52.
3. Dublin Core -- http://dublincore.org/documents/dces/
4. J, A Hartigan. "Clustering Algorithms. WILEY Publication, 1975.
5. S. Vrettos, A. Stafylopatis., "A Fuzzy Rule-Based Agent for Web Retrieval-Filtering", *Web Intelligence: Research and Development : First Asia-Pacific Conf.*, Springer-Verlag, October, 2001, pp. 448-453.
6. F. Yang, Y. Zhu, B. Shi, "A New Algorithm for Performing Ratings-Based Collaborative Filtering", *Web Technologies and Applications: 5th Asia-Pacific Web Conf.*, Springer-Verlag, 2003, pp. 239 – 250.
7. Robertson. S. and Walker. S., "Threshold setting in adaptive filtering", *Journal of Documentation,* 2000, 56, 3:312-331.