http://www.zju.edu.cn/jzus; E-mail: jzus@zju.edu.cn
ISSN 1009-3095 *Journal of Zhejiang University SCIENCE* V.4, No.4, P.415-420, July-Aug., 2003

Web multimedia information retrieval using improved Bayesian algorithm*

YU Yi-jun(余轶军)†, CHEN Chun(陈纯), YU Yi-min(余轶民), Lin Huai-zhong(林怀忠)
(*Department of Computer Science & Engineering, Zhejiang University, Hangzhou 310027, China*)
†E-mail: yijunyu@mail.hz.zj.cn

Received Oct.21, 2002; revision accepted Jan.9, 2003

Abstract: The main thrust of this paper is application of a novel data mining approach on the log of user's feedback to improve web multimedia information retrieval performance. A user space model was constructed based on data mining, and then integrated into the original information space model to improve the accuracy of the new information space model. It can remove clutter and irrelevant text information and help to eliminate mismatch between the page author's expression and the user's understanding and expectation. User space model was also utilized to discover the relationship between high-level and low-level features for assigning weight. The authors proposed improved Bayesian algorithm for data mining. Experiment proved that the authors' proposed algorithm was efficient.

Key words: Relevant feedback, Web log mining, Improved Bayesian algorithm, User space model
Document code: A **CLC Number:** TP393

INTRODUCTION

People are becoming more interested not only in text information but also in multimedia information such as image, audio and video. Now more and more attention is being paid to content-based retrieval systems for web use because they play a key role in utilizing information available on the Internet. Many content-based retrieval systems had been developed (Flickner *et al.*, 1995; Gudivada *et al.*, 1995). Some current research efforts (Lu *et al.*, 2000) focused on how to combine low-level visual features and high-level semantic features together to retrieve multimedia information. However, how to obtain the high-level semantic features is a key issue. If multimedia information annotation is required, it encounters the same problem of annotation acquisition in traditional text-based multimedia information retrieval systems (Wang *et al.*, 2000).

In this paper, we combine low-level features and high-level features (whenever they are available and suitable) for web multimedia information search, in which the text content (e.g., image URLs and filenames, page titles, and surrounding text, etc.) on the web pages can be used as potential high-level semantic features to represent the multimedia information on the same pages. We therefore built the information space model, which is a representation of multimedia information using a set of (both visual and semantic) feature vectors, from the multimedia information and the text content of the web pages.

We constructed the user space model of the keyword vectors used by users to represent multimedia information in the database, from the user's log data of relevant feedback. The user space model was then combined with the information space model to eliminate mismatch between the page author's expression and the user's understanding and expectation. The relationship between the low-level features and the high-level features can also be discovered from the user log by user space model.

ARCHITECTURE OF THE WEB RETRIEVAL SYSTEM

There are three main components in the architecture of our web retrieval system (Han *et*

* Project (No. 20020335020) supported by Research Fund for Doctoral Program, Ministry of Education of China

al., 2001): user interface, information space model, and user space model. User interface is an information browser, which also provides a query interface and a user feedback interface (as in Fig.1). Information space model includes crawler, feature extractor, information indexer, matcher, query updater, and database, etc. Information space model architecture and user space model architecture are shown in Fig.2.

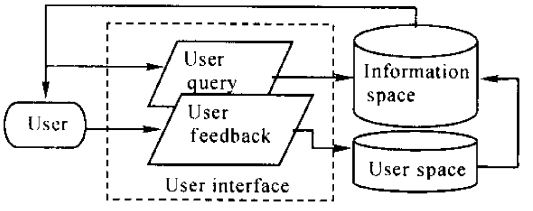

Fig.1 Architecture of web retrieval system

When web retrieval system begins to work, off-line crawler runs at regular intervals to collect potential web pages containing text, images, videos, etc and store them. Feature extractor extracts both low-level visual features and high-level semantic features from these pages and stores them in the database separately. Crawler and feature extractor work simultaneously. Then, indexer is applied. After user submits a query, matcher will yield retrieval results and users will specify returned result whether it is relevant or irrelevant to the user's intents by user feedback interface. Log miner builds user space model from user feedback log database. The user space model is then combined with information space model to update information space model.

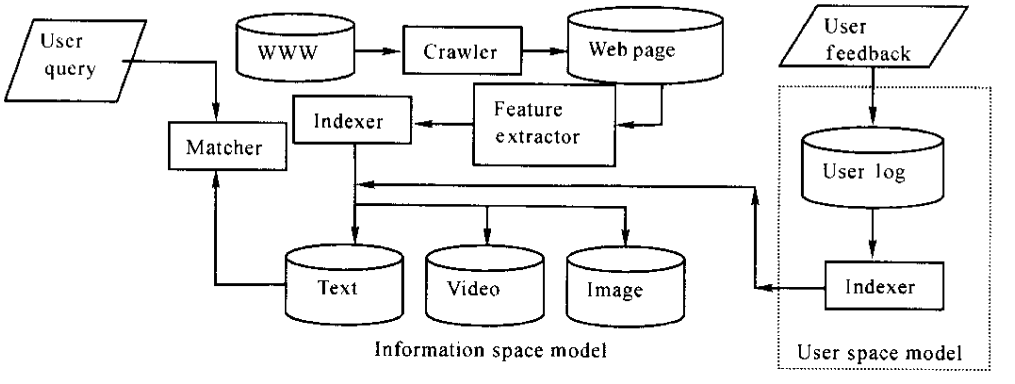

Fig.2 Information space model and user space model architecture

QUERY UPDATING USING RELEVANT FEEDBACK

Human interaction is used as a method to improve the retrieval performance. The user is asked to specify retrieval results as relevant or irrelevant. Relevance feedback inputs the user's judgements on previously retrieved information to construct a personalized query. So, the query is modified to guess the real intention of the user by relevance feedback (Wu *et al*., 2002). The theory of relevance feedback algorithms is well-developed for the traditional vector space model. These algorithms utilize the distribution of terms over relevant and irrelevant information to re-estimate the query term weights, resulting in an improved user query. We can use the traditional Rocchio's formula (1971) as follows.

$$q^{u} = q^{o} + \alpha \frac{\sum q^{+}}{n^{+}} - \beta \frac{\sum q^{-}}{n^{-}} \qquad (1)$$

Where, q^{o} is the original query, q^{+} is the set of positive (relevant) examples, n^{+} is the number of positive examples, q^{-} is the set of negative (irrelevant) examples, n^{-} is the number of negative examples, and q^{u} is the updated query.

WEB USER LOG MINING

Although we used relevance feedback to increase the retrieval accuracy, there are also

some problems. One is that text features extracted from the web page usually contain many irrelevant texts and some relevant texts. The other is short query problem caused by lazy users who like to submit short queries rather than full queries. It increases the confusion between the intent of the user and the page author's expression. Fortunately, there is the accumulation of all users' feedback information stored in the user log. To solve these problems, we built user space model from the user log to improve information space model.

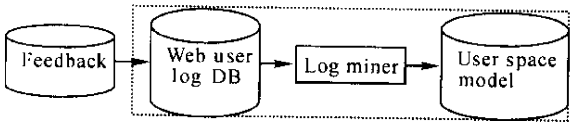

Fig.3 Web User Log Mining

1. Building user space model based on improved Bayesian algorithm

User space model is constructed from the data mined from user feedback log data. And information space model is constructed from original information extracted from the web pages. When a user submits a query, web retrieval system will return to the user some multimedia information found based on original information space model. Then the user can judge whether the return results are relevant or not by feedback user interface. Obviously, most users do not have the patience and time to mark all relevant and irrelevant results, although even a small set of feedback can provide very useful information.

Even though there are many data mining methods (Raghavan *et al.*, 2000) for building user space model, we selected improved Bayesian theory because it is simple to implement and is efficient. From web user log, we can calculate the probabilities listed below. We let Q be the set of total queries used until now and let $W_j(j=1..N)$ be the set of all individual words that appear in Q. For a query in Q, I_r is relevant information and I_i is irrelevant information specified by the user and stored in the user log.

$$P(I_r)=\frac{N_r}{N_Q} \tag{2}$$

$$P(I_i)=\frac{N_i}{N_Q} \tag{3}$$

where, N_r is the number of query times that Information I_r has been retrieved and marked as relevant. N_i is the number of query times that Information I_i has been retrieved and marked as irrelevant. N_Q is the total number of queries.

$$P(I_r \mid W_j)=\frac{N_r(W_j)}{N_Q(W_j)} \tag{4}$$

$$P(I_i \mid W_j)=\frac{N_i(W_j)}{N_Q(W_j)} \tag{5}$$

where, $N_r(W_j)$ is the number of query times that Information I_r has been retrieved and marked as relevant for those queries that contain word $W_j(j=1..N)$. $N_i(W_j)$ is the number of query times that information I_i has been retrieved and marked as relevant for those queries that contain word W_j, and $N_Q(W_j)$ is the number of queries that contain W_j. $P(I_r \mid W_j)$ And $P(I_i \mid W_j)$ represent prior probabilities.

Based on the Bayesian theory, we have

$$P(W_j \mid I_r)=\frac{P(I_r \mid W_j)P(W_j)}{P(I_r)} \tag{6}$$

$$\frac{P(W \mid I_r)}{P(W \mid I_i)}=\prod_{j=1}^{N_T}\frac{P(I_r \mid W_j)P(W_i)}{P(I_i \mid W_j)P(I_r)} \tag{7}$$

We use logarithmic and exponential operation for Eq. (7). Then we can have

$$P(W \mid I_r)=\frac{\exp\left(\log P(I_i)/P(I_r)+\sum_{j=1}^{N_T}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)}{1+\exp\left(\log P(I_i)/P(I_r)+\sum_{j=1}^{N_T}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)} \tag{8}$$

As relevant and irrelevant information is mutually exclusive, we have

$$P(I_r)=1-P(I_i) \tag{9}$$

We put Eq.(9) into Eq.(8) to yield

$$P(W \mid I_r)=\frac{\exp\left(\log P(I_i)/(1-P(I_r))+\sum_{j=1}^{N_T}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)}{1+\exp\left(\log P(I_i)/(1-P(I_i))+\sum_{j=1}^{N_T}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)} \tag{10}$$

Although we can have good results by Eq. (10), there are also some problems. $\exp(\log P(I_i)/(1-P(I_r)))$ lead to zero when $P(I_i)$ is very large, even $\sum_{j=1}^{N_r}\log P(I_r \mid W_j)/P(I_i \mid W_j)$ is equal to zero when we have no relevant information that $P(W \mid I_r)$ is a very big remnant value and distributes in small space. If we calculate $P(W \mid I_r)$ using Eq. (10), it will cause warp. To solve these problems, we use logarithm regression method to improve Eq. (10). So we can have improved Bayesian algorithm.

$$P(W \mid I_r) = \frac{\exp\left(a + b\sum_{j=1}^{N_r}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)}{1+\exp\left(a + b\sum_{j=1}^{N_r}\log P(I_r \mid W_j)/P(I_i \mid W_j)\right)} \quad (11)$$

The regressive parameters a and b are determined by least square method. It is obvious that the above problems can be solved well by using Eq. (11), but it is not improved evidently when the word's number N is very large.

For given information I, $P(W \mid I)$ calculated using Eq. (11) forms a vector for I. We call this vector the user space model of information I; of course the information space model of Information I is built from the related features extracted from the web pages.

2. Updating information space model using user space model

User space model is good supplement for the original information space model, but we cannot completely replace information space model user space model because few users like to tag all relevant and irrelevant information in retrieval result. So we integrate the user space model into the original information space model to improve the accuracy of new information space model.

For each information I, vector $\boldsymbol{M}_I$ is the high-level feature in the information space model and vector $\boldsymbol{U}$ is the high-level feature in the user space model. We simply use the linear combination method to integrate these two vectors.

$$\boldsymbol{M}_I' = c\boldsymbol{M}_U + (1-c)\boldsymbol{M}_I \quad (12)$$

where, vector M_I' is denotes updated information space model. c is the confidence of $\boldsymbol{M}_U$ vector in user space model to adjust the weight between the user space model and the information space model. The value of c is related to the accuracy and comprehensiveness of vector $\boldsymbol{M}_U$. The times that information is marked in feedback by user can be used to determine the value of c.

3. Weight adjustment using user space model

The weight used to balance the importance of low-level features and high-level features is almost impossible to assign in advance accurate weight values to different sources. So we initially assign the same set of weight values. It is very efficient for us to build up the baseline system. The weight can be automatically adjusted to a suitable value by the system through the user's feedback as to the relevancy of certain returned information. Moreover, after we collect enough user log information of user feedback, data mining technology can be applied to find out the importance of the low-level feature and high-level feature for different concepts. For example, if the semantic similarity of relevant feedback results is higher than the visual similarity, we can say that the semantic features are more important than the visual features.

We use the simplest similarity models, the liner combination of semantic similarity and visual similarity, to calculate overall similarity in Eq. (13). The weight α determined by the user space model is the factor that balances the weight between high-level and low-level feature. There are many similarity methods such as cosine model, contrast model, roughness model, etc (Harman *et al*., 1992) for semantic similarity and visual similarity. In this paper, the semantic similarity between the query and multimedia information is calculated using the dot product of the query's text feature vector and the multimedia information's text feature vector. The visual similarity between the query and information is calculated using the Euclidean distance model.

$$\mathrm{Sim}_o(q, \boldsymbol{I}) = \alpha\mathrm{Sim}_h(\boldsymbol{q}_h, \boldsymbol{I}_h) + (1-\alpha)\mathrm{Sim}_l(\boldsymbol{q}_l, \boldsymbol{I}_l) \quad (13)$$

Where, Sim_o is overall similarity, Sim_h is semantic similarity, Sim_l is visual similarity, q is the query, $\boldsymbol{q}_l$ is the low-level feature vector of the query, $\boldsymbol{q}_h$ is the high-level feature vector of the query. $\boldsymbol{I}$ represents information, $\boldsymbol{I}_l$ is the

low-level feature vector of $\boldsymbol{I}$, and $\boldsymbol{I}_h$ is the high-level feature vector of $\boldsymbol{I}$.

EXPRIMENTAL DETAILS

In order to prove the advantage of web log mining, we built three systems: baseline system, relevance feedback system, web log mining system. We selected about 100 Chinese computer science department websites and used the web pages on them as candidates. For multimedia information, we used image as example. The crawler was used to collect images from these hyperlinks. In total, we collected more than 7 000 images from these websites. All related semantic features, including image filenames, ALT texts, surrounding texts, etc. as well the low-level visual features were also extracted using the feature extractor at the same time. The images were stored in the database and indexed with their high-level and low-level features. It was difficult for us to calculate the recall of the system because it was a tedious job to browse the entire image database and specify the ground truth manually (Wu *et al.*, 2002). Therefore, we only chose 10 queries to demonstrate the performance of the improved system. Furthermore, rough calculation of recall was done after scanning the top 80 images returned for each query.

Relevance feedback system improved the performance of baseline system. The retrieved images were re-ranked based on the positive and negative examples. But users were often unwilling to provide feedback. Therefore, in our experiment, we let the testers to mark only 2 positive/negative examples and then evaluated the precision-recall after relevance feedback.

For ranked sets of retrieved information, it is standard practice to average the performance measures like precision, recall and fallout over different cut-off levels. Recall, fallout and precision were related just as the corresponding probabilities were related to the probability of relevance. A simple standard (TREC conferences) approach for calculating the expected precision was the average precision measure (Hiemstra *et al.*, 2001):

$$\text{expected precision} = P(k) \cdot \frac{\text{expected } recall}{\text{expected } fallout} \tag{14}$$

$$\text{average precision} = \left(\sum_{k \in \text{ranks of relevant info}} \text{precision at } k\right)/R \tag{15}$$

This is the expected precision if we assume that the probability $P(k)$ of a user looking for k relevant information is uniformly distributed for $1 \leqslant k \leqslant R$. R is the total number of known relevant information.

After we implemented the baseline and feedback system, we only obtained a small sized user log containing about 2000 entries of feedback on about 100 queries. The preliminary performance result was just based on the analysis of this small user log.

The feedback from a single user was limited, so we used all the accumulated users' feedback information stored in the user log to find more accurate information about the web multimedia information. The user space model was constructed from the user log and used to improve the information space model and further improve the retrieval performance. The log mining could not only improve the precision when the recall was low, but could also improve the precision when the recall was high. In other words, the overall performance of the system was improved after log mining. It can be seen from Fig.4.

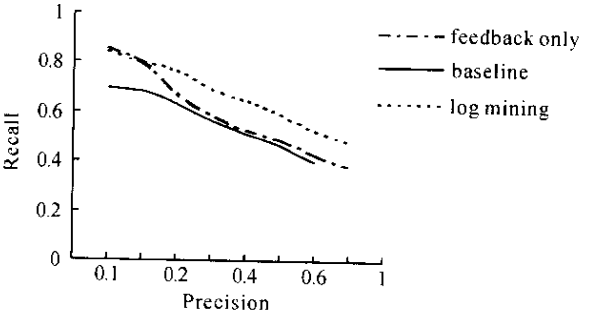

Fig.4 The average precision-recall curve of retrieval performance for all queries

There is another important thing. From Fig. 5, we can see that relevance feedback did better than log mining for some queries. This was because there was only one user log entry about the query in our user log. So the performance was barely improved after log mining. But in average, we can see from Fig.4 that the performance after log mining was better than that of relevant feedback. This also proved that a large user log collection is necessary to improve the overall

performance of the system.

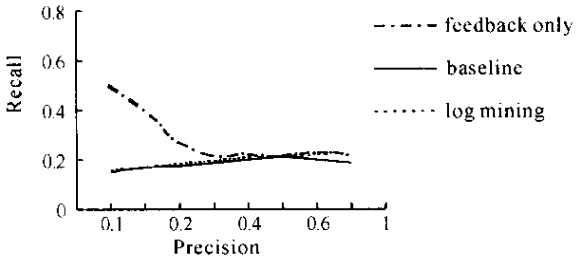

Fig.5 The precision-recall for "mechanical"

The average precision-recall curve of retrieval performance using Bayesian algorithm and improved Bayesian algorithm is compared in Fig.6. It is obvious in Fig.6 that the precision of the system using improved Bayesian algorithm is better than that using Bayesian algorithm.

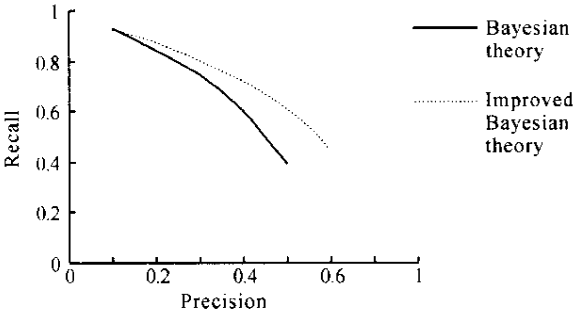

Fig.6 The average precision-recall: curve of retrieval performance with log mining using Improved Bayesian algorithm and using Bayesian algorithm

CONCLUSIONS

We focused on obtaining high-level semantic features in this work, but there were many difficulties such as irrelevant information, mismatch between the page author's expression and the user's understanding, and difficulty in finding out the relationship between low-level features and high-level features. To overcome these difficulties, we first constructed the information space model from the web page content. The text content on the web pages was used as the high-level semantic features for the web multimedia information, which was further combined with the low-level visual features in the retrieval process. Then we collected the web log data of user's feedback and applied log mining to build the user space model to improve the accuracy of the information space model. We used improved Bayesian algorithm used to build the user space model. Experiment proved that a large user log collection is necessary to improve the overall performance; and that our proposed algorithm was efficient. In the future we plan to try to study more efficient data mining methods that will lead to more precise results.

References

Flickner, M., Harpreet, S.S., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D. and Yanker, P., 1995. Query by image and video content. *IEEE Computer*, **28**(9): 23 – 32.

Gudivada, V.N. and Raghavan, J.V., 1995. Content-based image retrieval systems. *IEEE Computer Magazine*, **28**(9): 18 – 22.

Han, J.W., Meng, X.F., Wang J. and Li, S.E., 2001. Research on WEB mining. *Journal of Computer Research & Development*, **38**: 405 – 414.

Harman, D., Fox, E., Baeza-Yates, R.A., Lee, W.C., 1992. Inverted Files, Information Retrieval: Data Structures and Algorithms. Prentice-Hall Inc., New Jersey, p.28 – 43.

Hiemstra, D. and Stephen, E.R., 2001. Relevance feedback for best match term weighting algorithms in information retrieval. Proc of the Second DELOS Network of Excellence Workshop on Personalization and Recommender Systems in Digital Libraries. Dublin City University, Ireland, http: //www.ercim.org/publication/ws-proceedings/DelNoe02/hiemstra.pdf.

Lu, Y., Hu, C.H., Zhu, X.Q., Zhang, H.J. and Yang, Q., 2000. A unified framework for semantics and feature based relevance feedback in image retrieval systems. Proc. of the 8th ACM Multimedia Conference. Los Angeles, USA, 31 – 37.

Raghavan, V.V. and Aladdin, H., 2000. Dynamic data mining. Proc. of the 13th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, New Orleans, USA, p. 220 – 229.

Rocchio, J.J., 1971. Relevance feedback in information retrieval.: The SMART Retrieval System. Prentice Hall Inc., New Jersey, USA, p.313 – 323.

Wang, J.C., Pan, J.G. and Zhang, F.Y., 2000. Research on web text mining. *Journal of Computer Research& Development*, **37**(5): 513 – 520(in Chinese).

Wu, H., Li, M.J., Zhang, H.J. and Ma, W.Y., 2002. Improving image retrieval with semantic classification using relevance feedback. Proc. of IFIP TC2/WG2.6 6th Working Conference on Visual Database Systems. Brisbane, Australia, p. 327 – 339.