Personal Information Organization using Web Annotations

Laurent Denoue
FXPAL
3400 Hillview Ave, Building 4
Palo Alto, CA 94304, USA
denoue@pal.xerox.com

Laurence Vignollet
Laboratoire Syscom
Université de Savoie
73370 Le Bourget du Lac, France
Laurence.Vignollet@univ-savoie.fr

Abstract: As more information is made available online, users collect information in personal information spaces like bookmarks and emails. While most users feel that organizing these collections is crucial to improve access, studies have shown that this activity is time consuming and highly cognitive. Automatic classification has been used but by relying on the full text of the documents, they do not generate personalized classifications. Our approach is to give users the ability to annotate their documents as they first access them. This annotation tool is unobtrusive and welcome by most users who generally miss this facility when dealing with digital documents. Our experiments show that these annotations can be used to generate personalized classifications of annotated Web pages.

Introduction

Organizing Web pages collected in our bookmarks is a long and highly cognitive task. Most users would be happy to delegate this activity to an automatic classifier (Abrams 1998). User studies also show that the number of bookmarks grow linearly with time and that most users do not organize their bookmarks at creation time. Because they do not classify at creation time, they have to organize a large number of bookmarks and typically do not remember what was interesting in a particular page. Remembering what was interesting can also be impossible when the Web page has been modified or is not longer available, which is typical on the Web. One solution used in Vistabar is to index every Web page loaded in the browser (Marais & Bharat 1997, Li 1998). The other approach is to store a signature of each page which can help to retrieve the page from the Web when its URL has changed (Phelps & Wilensky 2000). But indexing or being able to retrieve the same page on a different location does not necessarily help the user to remember why this page has been saved as a bookmark. We believe that being able to annotate the document would certainly help users remember their activities around this document and thus help them classify it.

Even if marks can help users remember what was important in a document, there is still a need to help them automatically classify these documents. Automatic classification has been applied to cluster bookmarks (Maarek & Shaul 1996). The results do not show if these automatic classifications fit user expectations. We argue that since a Web page can contain several topics like home pages, listings and online news papers, using all the words to classify this Web page will probably lead to bad classifications because only a subset of this page might be of interest to the user.

In this paper, we use a Web annotation system to let users highlight and possibly annotate the Web pages they access. Annotating on paper is ubiquitous and some annotation systems like ThirdVoice, iMarkup and eQuill are now available to mark Web pages. In our experiences, we have used Yawas, a Web annotation system prototype that supports the creation of annotations in a fast and effective way using Dynamic HTML and the Document Object Model (Denoue & Vignollet 2000).

By annotating Web pages, users select words which can then be used to represent each document. This paper is organized as follows. In the first part, we present a pilot study which tries to understand the utility of annotations to help someone understand the topic of a document. In the second part, we present the results of applying an automatic classification algorithm to classify the documents, first using the traditional approach which indexes the full text of the documents, and secondly using only the annotated passages in each document to represent the documents. We show that representing the documents by their annotations provides more personalized classifications than when the documents are indexed with their full text.

Pilot Study: Manual Classification of Web Pages

Experimental Method

To gain an insight into the usefulness of annotations to classify Web pages, we asked two subjects to manually classify 333 annotated documents. Subjects were also asked to name the groups of documents. No time limit was imposed. The first subject is the author of the annotations, while the other subject was not familiar with the documents. Each subject had to complete two classifications. In the first setting, the original document was downloaded into a Web browser so the subject could see the complete page and manually classify it. In the second experiment, each document was represented by the list of annotated passages in this Web page.

Results

When accessing the original documents, the subjects took approximately 3 hours to classify the documents, and 2 hours and 30 minutes to classify them when using the annotations. This difference is primarily due to the time required to retrieve the documents from the Web and browse them. Subjects could not classify 10% of the documents because their URLs were broken. On the other hand, all documents have been classified when using the annotations.

For a given subject, the number of classes is not significantly different. However, the author of the annotations created 60 classes, while the other subject created 40 classes. This is not surprising since the author of the annotations is more apt at differentiating topic amongst the documents. For both subjects, the names created when considering the highlighted texts were more precise. For example, “conference” was used instead of the more appropriate “classification”, and “home page” was used instead of “annotation”. Using the annotations, subjects have then been able to generate more precise group names.

To test how annotations can help an automatic classifier to build a classification that better fits a particular user, we used the classification provided by the author of the annotations as a reference and compared it to the classifications produced by the second subject. The second subject misclassified 35 documents when using the full text of the documents, but only 5 when using the annotations to represent each document. This result suggests that an automatic classifier could also take advantage of the annotations to generate a personalized classification.

Moreover, we observed that most documents contained about 10 annotated words (after the indexing step). This further suggests that the annotations could be used to generate concise representations for documents, similar to the representations used by current search engines on the Web.

Second Study: Automatic Classification

Since the manual classification of the documents showed encouraging results, we conducted a second experiment using an automatic classification algorithm.

Classification Algorithm

Numerous algorithms are available to classify data and most of them can be applied to document classification. Each document is represented by a vector of features. Features are typically words, but other data could be used like the date of creation, the author, etc. Since most users tend to classify their documents by topics, we represented the documents by their words. Basically, two approaches can be used to classify a set of documents: supervised classification using machine learning techniques and unsupervised classification (also known as clustering).

Supervised classification requires a set of pre-classified documents where each document is associated to a predefined class. A machine learning algorithm like decision trees or Naïve Bayes is used to find a function which maps documents to their class (Mitchell 1997). This function is then used to predict the class of a new document. Classifying bookmarks with this approach would require users to pre-classify a subset of the bookmarks. Some authors have avoided this step by using the Yahoo! classification scheme where thousands of Web pages are already classified (Chen & Dumais 2000, Li 1998, Marais & Bharat 1997). In this experiment, we didn't use this approach since we wanted to discover a personalized classification. Yahoo! would have given us a more general classification which we believe is not necessarily adapted to every user.

On the other hand, unsupervised classification does not require a set of pre-classified documents. Documents are compared to each other and the algorithm tries to structure them. This structure depends on the nature of the algorithm. Some of them - like the single pass algorithm - produce a flat classification of the documents. Others like the hierarchical agglomerative clustering (HAC) induce a hierarchy of classes. Although flat classifications can further be split to create a hierarchy, we chose to use the HAC algorithm because it does not require a priori knowledge about the number of classes and does not require a threshold when comparing two documents (see (Rasmussen 1992) for details).

In the first step of the HAC algorithm, each document is put in one class. At each subsequent step, the two most similar classes are merged. The algorithm typically ends when there is just one class. There are 3 basic alternatives to compute the similarity between two classes, also known as "single link", "complete link" and "group average" described in (Rasmussen 1992). We implemented all of them in our experiment. Documents were indexed by removing common words from a stop list and by further applying the simple Porter stemming algorithm (Porter 1980).

Experimental Method

We used the 400 annotated documents used for the pilot study. We have implemented the 3 most common variants of the HAC algorithm "single link", "complete link" and "group average". We ran the classifier for each variant, first using the full text to represent the documents, and then using only the annotated words in each document. To compute the quality of the classifications, we compared each of them to the reference classification obtained from the author of the annotations using the quality measure used by Zamir and al. (Cutting et al. 1992):

$$Quality(C) = \sum_{g \in C} \sqrt{t(g)} - \sqrt{f(g)}$$

where t(g) is the number of pairs of documents in one cluster which have also been classified together in the reference classification, and f(g) is the number of pairs of documents in one cluster which have not been classified together in the reference classification.

The global quality sums the difference between t(g) and f(g) for every cluster in the current classification. In our experiment, each document has been indexed by removing common words from a stop list and by further applying a simple stemming algorithm (Porter 1980).

Results and Discussion

(Figure 1) shows that the quality is always better when using the annotations. The graphic show the quality of the classification at different merging steps as the HAC algorithm is running. At the first step, the quality is null since no documents are merged: the number of correctly classified documents is equal to the number of misclassified documents. Using the full text of the documents, the quality never goes far above zero, meaning that the classification is never useful for the user: the same number of documents is correctly of incorrectly classified.

On the other hand, the quality reaches a maximum above zero when the annotations are used, suggesting that at some point the classification becomes useful (more documents are correctly classified). Determining this point can be useful but this is not crucial since the HAC algorithm never modifies its previous merges as it runs, thus preserving the quality of the classification obtained at the previous steps. However, the classes become populated with noisy documents and it might be desirable to stop the HAC algorithm before. On our sample of 450 annotated documents, we found that this maximum was reached when there was only one word left to represent a class. In our implementation, each class is represented by the intersection of the words included in the documents of this class.

One typical problem with HAC algorithms is that they tend to produce a deep hierarchy, which is not desirable if users need to browse it. A maximum of 5 levels is usually recommended (Larson & Czerwinski 1998). One solution is to arbitrarily slice the hierarchy into 5 or more levels (Maarek & Shaul 1996), but it is not clear how this threshold fits all tasks and how valuable distinctions among documents is lost with this simplification. In our sample of annotated documents, the deepest level is about 4 in the classification when the curve reaches a maximum.

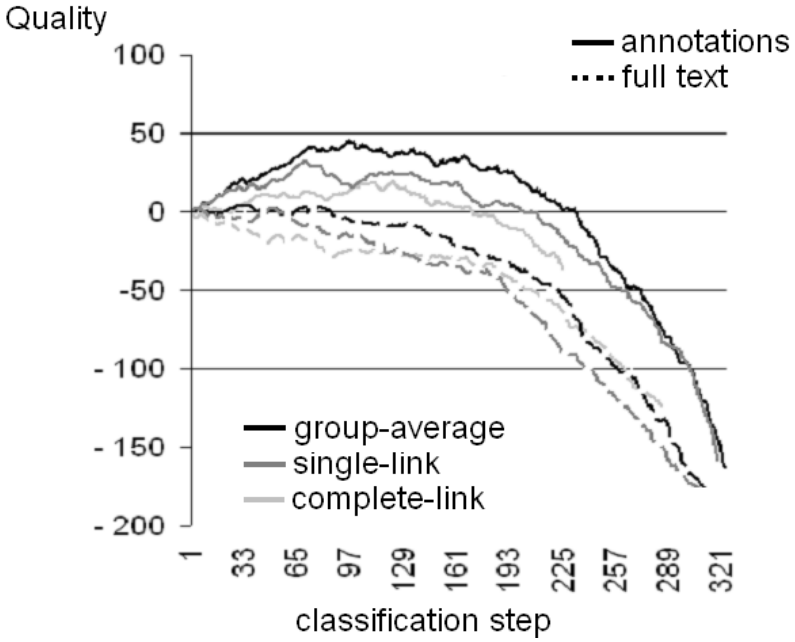

Figure 1: Quality of the classification at different steps of the HAC algorithm; Dashed lines represent results when the full text is used and solid lines represent results when the annotations are used; 3 variants of the HAC algorithm are shown in different gray scale for the full text and for the annotation.

Using the intersection of words contained in one class to name this class, we found that annotation-based classification generate better and more concise names, typically varying from 10 to only 1 word. This is consistent with our preliminary findings and in sharp contrast with the names obtained when the full text is used. In this later case, class names are useless and further filtering based on TFIDF would be required. To browse the hierarchy, users also need to see document representations. Since our pilot study suggested that annotation-based document representations were useful to understand the meaning of a document, we chose to use this representation in the classification as well.

Concluding Remarks

We believe that being able to annotate digital documents like Web pages can dramatically improve the user experience of accessing information online. Annotating is very natural on paper and few tools are currently supporting this activity for digital documents (see Xlibris in Schilit et al. 1999). But digital annotations have the potential to push the limits of their paper-based counterpart. Our experiments suggest that they can

support users in other activities like the classification of personal information. Using the same approach, annotations can also be used to improve information retrieval (Golovshinsky et al. 1999). Annotation tools will play an increasing role in personal information management systems by helping users to filter and organize vast amounts of information they collect in their personal spaces. Email and Web pages are current examples. More importantly, by annotating their documents, users build personal profiles which could be used to retrieve documents out of the personal space.

References

Abrams, D. (1998). Information Archiving with Bookmarks: Personal Web Space Construction and Organization. *CHI98*, *Conference on Human Factors in Computing Systems (CHI 98), 1998*, Los Angeles, CA.

Chen, H., & Dumais, S.T. (2000). Bringing order to the web: Automatically categorizing search results. *Conference on Human Factors in Computing Systems (CHI 2000), 2000,* La Hague, The Netherlands 145-152.

Cutting, D.R., Karger, D.R., Pederson, J.O., & Tukey, J.W. (1992). Scatter/Gather : A cluster-based approach to browsing large document collections. *Research and Development in Information Retrieva (SIGIR92), 1992*, Copenhague, Danemark 318-329.

Denoue, L., & Vignollet, L. (2000). An Annotation tool for Web browsers and its applications to information retrieval. Recherche d'Information Assistée par Ordinateur (*RIAO2000), 2000,* Paris, France.

Golovchinsky, G., Price, M., & Schilit, B. (1999). From Reading to retrieval: Freeform Ink Annotations as Queries. *ACM SIGIR99, 1999*, Berlekey, CA. 19-25.

Li, W.S. (1999). PowerBookmarks: A System for Personalizable Web Information Organization, Sharing, and Management. *ACM Sigmod 1999, 1999,* Philadephia, USA 565-567.

Larson, K. & Czerwinski, M. (1998). Web Page Design: Implications of Memory, Structure and Scent for Information Retrieval. *Conference on Human Factors in Computing Systems (CHI 1998), 1998,* Los Angeles, CA USA 25-32.

Maarek, Y.S., & Shaul, I.Z.B. (1996). Automatically Organizing Bookmarks per Contents. 5*th International World Wide Web Conference (WWW5), 1996*, Paris, France.

Marais, H., & Bharat, K. (1997). Supporting cooperative and personal surfing with a desktop assistant *ACM Symposium on User Interface Software and technology (UIST 97), 1997*, Alberta, Canada.

Mitchell, T.M. (1997). *Machine Learning*. McGraw-Hill International Editions, Computer science Series, ISBN 0-07-042807-7.

Phelps, T., & Wilensky, R. (2000). Robust Hyperlinks: Cheap, Everywhere, Now. *Digital Documents and Electronic Publishing (DDEP00), 2000*, Munich, Germany.

Porter, M.F. (1980). An Algorithm For Suffix Stripping. *Program*, 14(3), 130-137.

Rasmussen, E. (1992). *Clustering Algorithms*. Information Retrieval: Data Structures and Algorithms: William Frakes & Ricardo Baeza-Yates, Prentice-Hall, ISBN-0-13-463837, 419-442.

Schilit, B., Golovchinsky, G., Takana, K., Marshall, C.C. (1999). As We May Read: The Reading Appliance Revolution. *Computer,* 32(1), 65-73.

Acknowledgments

This work has been conducted at the Syscom laboratory, University of Savoie. We wish to thank François Rechenmann for his critical remarks during our experiments. We also thank our subjects for the time they kindly devoted to our experiences.