

# User-Oriented Document Summarization through Vision-Based Eye-Tracking

Songhua Xu<sup>‡,§,†\*</sup>

<sup>‡</sup>: College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang, 310027, P.R. China

Hao Jiang<sup>‡</sup>

<sup>‡</sup>: Department of Computer Science, Yale University, New Haven, Connecticut, 06520-8285, USA

Francis C.M. Lau<sup>‡</sup>

<sup>‡</sup>: Department of Computer Science, The University of Hong Kong, Pokfulam Road, Hong Kong, P.R. China

## ABSTRACT

We propose a new document summarization algorithm which is personalized. The key idea is to rely on the attention (reading) time of individual users spent on single words in a document as the essential clue. The prediction of user attention over every word in a document is based on the user's attention during his previous reads, which is acquired via a vision-based commodity eye-tracking mechanism. Once the user's attentions over a small collection of words are known, our algorithm can predict the user's attention over every word in the document through word semantics analysis. Our algorithm then summarizes the document according to user attention on every individual word in the document. With our algorithm, we have developed a document summarization prototype system. Experiment results produced by our algorithm are compared with the ones manually summarized by users as well as by commercial summarization software, which clearly demonstrates the advantages of our new algorithm for user-oriented document summarization.

## General Terms

Algorithms, Design, Experimentation, Human Factors, Languages, Measurement, Performance

## Author Keywords

User-oriented document summarization, personalized discourse abstract, user attention, implicit user feedback, commodity eye-tracking

## ACM Classification Keywords

H.1.2 User/Machine Systems: human factors, human information processing; H.3.1 Content Analysis and Indexing:

\*Contact him at A DOT B AT C DOT com in which A = "songhua", B = "xu", and C = "gmail".

abstracting methods; H.3.3 Information Search and Retrieval: information filtering, relevance feedback; H.5.2 User Interfaces: input devices and strategies, interaction styles; I.2.7 Natural Language Processing: discourse, text analysis; I.7.5 Document Capture: document analysis

## INTRODUCTION

Document summarization has continuously been an active area of research in artificial intelligence, information retrieval and natural language processing for more than two decades [1]. Summarizations can help users to efficiently locate the desired information or quickly convey the key messages embedded in an article or a text corpus of any size. To the best of our knowledge, despite the importance of and the attention given to this area, studies on document summarization so far have rarely taken human factors into consideration. In this paper, we propose a *personalized* document summarization method which carefully considers the user's interests. Based on the method, we have implemented a personalized document summarization system. Our algorithm tracks a user's attention times over individual words using a vision-based commodity eye-tracking mechanism. Then user attention time over any arbitrary word is predicted by a data mining process according to pairwise word semantic similarity.

As an independent thread of research, eye-tracking has attracted researchers in the fields of human-computer interaction, user modeling, computer graphics and interactive techniques. The main advantage of eye-tracking is that it is unintrusive when acquiring user feedbacks. User feedbacks are needed for determining users' preferences in order to build adaptive systems. To date, however, very few research projects in such fields as information retrieval and natural language processing have taken advantage of modern eye-tracking technologies. In this paper, we present how we use commodity eye-tracking to develop a personalized document summarization system. With eye-tracking, we can acquire user attention data over individual words from online materials that the user has seen or read. These data can then be used to predict the user's attention on the contents of a document to be summarized. Based on this prediction at the sentence level, our algorithm produces a personalized sum-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IUI'09, February 8 - 11, 2009, Sanibel Island, Florida, USA.

Copyright 2009 ACM 978-1-60558-331-0/09/02...\$5.00.

marization of the document which is optimized with respect to the user’s reading interests.

### MAIN IDEA

For every sentence  $S_i$  in a document, we denote the user  $U_j$ ’s attention on it as  $AT(S_i, U_j)$ , which is equal to the time the user spends on reading the sentence. For two sentences  $S_{i_1}$  and  $S_{i_2}$  in the document, after they have both been read by the user, if we have  $AT(S_{i_1}, U_j) > AT(S_{i_2}, U_j)$ , then it is reasonable to infer that the user is more interested in  $S_{i_1}$  than  $S_{i_2}$ .

Based on the above, to produce an optimal summarization for a document for a particular user, our algorithm would predict the attention of the user on individual sentences in the document. Given the prediction, we can then return an ordered list of sentences in descending order in terms of the predicted user attention. With a user tunable cutoff threshold, we can then obtain a personalized document summarization result. Overall, this is very similar to the rating problem in e-commerce—i.e., given a user’s ratings on a number of items (sentences), how to predict his ratings on new items (sentences) which he has not rated.

Although similar to the classical rating scenario, our method is more comprehensive: traditional rating of an item is “atomic”, meaning that the rating is applied to the item as a whole, but not its subcomponents or its different features separately; in our scenario, by contrast, user attention is compositive—e.g., for a sentence, the user’s attention is the accumulated attention of reading the words within the sentence, and for a paragraph, its attention is the sum of the attention on its every sentence. In fact, in reality, when human beings form their preferences, it is often a decision based on smaller decisions. For a typical rating system under today’s e-commerce settings, it is uncommon to ask users to specify a rating over many components or facets of a product. The users would find it to be too bothersome. In fact, getting users to feedback on an entire document is already difficult; getting them to feedback on subcomponents might simply be impossible. Given the unintrusive nature of eye-tracking, we can overcome this barrier and obtain user evaluation on the subcomponents of the target document. In our algorithm design, we carefully make use of the compositive structure of user attention for inferring user preference. The convenience of vision-based commodity eye-tracking makes our approach easy-to-adopt on any scale.

The remainder of the paper is organized as follows. We first survey the most related work. We then explain how to acquire user attention time on a document. After that, we discuss how to infer user attention via content-based data mining. Given the user attention estimation, we then introduce our algorithm for personalized document summarization based on the predicted user attention. We present experiment results to demonstrate the effectiveness of our method. Finally we conclude the paper and point out some future work directions.

## RELATED WORK

### Document Content Summarization

Much research has been dedicated to automatic summarization for both generic and domain-specific documents. The online article [1] at <http://www.summarization.com/> gives a comprehensive list of research papers published on document summarization studies till October 2008, containing 758 papers.

Jones [23] suggests that a document summarization process usually consists of two steps: (1) source representation which is built from the source text, and (2) summary generation, which synthesizes a summarization based on the source representation. Alternatively, summary can be also obtained through analyzing the semantics of the source text. Gong and Liu [14] presented a generic document summarization method that employs latent semantic analysis to identify semantically important sentences to create document summaries. Others have applied machine learning algorithms. Lin [28] used a selection function for key information extraction from documents, and a machine-learning process to automatically learn an optimized function which combines several heuristics for document summarization. Yeh et al. [46] proposed a trainable document summarizer which takes into account such features as text position, positive keywords, negative keywords, centrality, and resemblance of text to paper title. The weights to attach to these different features are obtained from a scoring function trained by genetic algorithms.

There also exist many public domain document summarization tools. Two of the most widely used ones are the MEAD system [35] and the LexRank system [10]. The MEAD system is an open source multilingual summarization package which has been successfully used in a variety of summarization applications, including summarization on mobile devices and web page summarization for use in search engines. The LexRank system is a stochastic graph-based document summarization system which computes relative importance of textual units via natural language processing. For ordinary users using the Microsoft Windows operation system, the “AutoSummarize” functionality provided in MS Word [33] can usually provide a satisfying solution.

### Implicit User Feedback

#### Query history

Query history probably is the most widely used implicit user feedback at present. Google’s personalized search service (<http://www.google.com/psearch>) allows users to store their search history in their Google account which will be analyzed for personalizing their future search. In general, there exist two classes of methods for providing personalized search based on query history: those based on the whole query history of a user and those based on query history in a particular search session. For the methods that use the whole query history, usually a user profile is generated to describe his search preference. For example, Liu et al. [30] constructed user profiles using the whole search history through an adaptive Rocchio algorithm [20]. Speretta and Gauch [40] demonstrated that using user profiles can significantly improve search engine performance. The query history in

a query session is also often called *query chain* [36]. Query chain is used to automatically suggest or to complete a query question for a particular user based on the query history in the same search session [19]. It is also used to expand the current query based on the query chain history.

#### *Click data*

Click data is another type of implicit user feedback, which has been intensively utilized, e.g., [9, 21]. The basic idea is that when a user clicks on a document, the document is considered to be of more interest to the user than other unclicked ones. There are many ways to infer user preference from click behaviors. For example, a simple approach would be when a user clicks on the  $i$ -th link in a ranked list of webpages before clicking on any of the first  $i - 1$  links, we can safely infer that the first  $i - 1$  documents are no more important than the  $i$ -th document. Among the sophisticated approaches, people have applied ranking SVM algorithm to find the best webpage rank according to a user click dataset [22].

#### *Attention time*

Attention time, also often referred to as display time or reading time, is a newly recognized type of implicit user feedbacks that is receiving increasing popularity even though the reliability of which for predicting user interest has yet to be confirmed. One side of the opinion is represented by arguments made by Kelly and Belkin [25, 24], claiming that there is no reliable relationship between the interestingness of a document and its display time; in their study the display time is measured as the average reading time by a group of users on articles of different topics in the Web. The other side of the opinion is like what is pointed out by Halabi et al. [18], which is that for a fixed user in a certain query session, attention time gives a strong indication of the user interest—the more time a user spends on reading a document, the more important the document is to him. We think these different conclusions are not contradicting as display time is calculated differently by the two groups.

In our prior work, we proposed using attention time as an effective clue for producing personalized webpage ranking [45] as well as personalized recommendations for documents, images and videos [44]. We obtained positive results in both studies. In this paper, we introduce a new user-oriented document summarization algorithm based on user attention time.

#### *Other types of implicit user feedbacks*

Other types of implicit user feedbacks include display time, scrolling, annotation, bookmarking and printing behaviors. Some researchers have started recently to combine multiple types of implicit feedbacks in order to obtain better inference of user interest [31]. Fox et al. [12] have made a comprehensive study and proposed a decision tree based method augmented by Bayesian modeling to infer user preference from a set of mixed types of implicit user feedbacks.

## **Eye-Tracking Strategy**

An eye tracker accumulates a series of points at which the eyes are looking. These points then define the part of contents which the user is reading. In [37], the authors presented a software system that automates eye tracking data analysis for Web usability studies. Similar to this work is a recent eye-tracking study for highlighted text by Chi et al. [5]. They have shown evidences derived from the eye-tracking data that user foraging behaviors are influenced by different highlighting strategies, including no highlights for any parts of a document, only highlighting keywords, and other more sophisticated key sentence highlighting strategies. Their report suggests a compelling need for new interface design such that people can easily digest information in a document collection, especially those on the Web. Bulling et al. [3] analyzed the eye movements of people in transit in an everyday environment using a wearable electro-oculographic (EOG) system; they achieved a decent rate for recognizing reading activity in daily-life scenarios through analyzing the captured eye-tracking data. The purpose of their research is to study whether different reading behaviors and levels of user attention on written texts can be automatically detected with the aid of eye-tracking in an unobtrusive manner. In another paper, Bulling et al. [4] described the potential of using wearable EOG goggles for context-awareness and mobile HCI applications. Despite these fruitful research on eye-tracking, however, as of today, eye-tracking techniques have been rarely employed as a user feedback approach for intelligently tuning adaptive systems in a user friendly way. For online reading and Web browsing applications, such an approach is especially desired but has been largely overlooked by knowledge management and information retrieval researchers so far. The work introduced in this paper presents our preliminary attempt along this desired direction.

## **User-Oriented Search Engines**

User-oriented search engines—one of the most popular types of personalized online services—as a relatively new track of research is drawing more and more attention these days, e.g., [34, 8]. All the user-oriented search engines so far rely on user feedbacks of various kinds, which can be broadly classified into two categories—explicit and implicit; both of them aim at inferring user intentions or preferences for customizing some search engine [39, 42, 43]. Because users generally would be least interested in providing explicit feedbacks, the recent trend is to derive search preferences from implicit feedbacks [16, 17, 13]. The most popular implicit user feedbacks currently utilized in commercial search systems are query history and click data which we have already discussed.

## **ACQUIRING USER ATTENTION VIA COMMODITY EYE-TRACKING**

### **Obtaining Gaze Samples through Vision-Based Commodity Eye Tracking**

Eye tracking is the technology to measure either the gaze, i.e., the spot a user is looking at, or the motion of the human eyes ([http://en.wikipedia.org/wiki/Eye\\_tracking](http://en.wikipedia.org/wiki/Eye_tracking)). In our work,

we use eye-tracking to measure the attention time of a user over a document appearing on the screen through identifying the part of the screen area the user is looking at and for how long. However, commercial eye-tracking devices are very expensive. Some researchers therefore turned to ordinary web cameras as eye-tracking devices [26, 7, 38, 29, 41, 15]. We followed suit and have assembled an eye-tracking setup using a simple web camera (Logitech Quickcam Notebook Pro) and an existent eye-tracking algorithm borrowed from the OpenGazer project [47]. Together with some vision techniques we created our custom eye-tracking component. This design of an eye tracking component is cost-effective and can be widely adopted on personal computers as many PCs these days are equipped with web cameras. The error of the detected gaze location on the screen by our commodity eye-tracking component is between 1–2 cm, depending on which area of the screen the user is looking at: the center of the screen has tracking error of around 1 cm whereas the boundary of the screen (a 19" screen monitor) has tracking error of around 2 cm.

### Anchoring Gaze Samples onto Individual Words

Through our commodity eye-tracking component, we obtain a number of fixation points on the screen, which indicate the detected gaze area of the user. For our summarization algorithm to work, we need to anchor these gaze samples onto individual words. Apparently, the more gaze samples a word receives, the more interesting the word is to the user. We now look at how to anchor gaze samples onto individual words.

We first introduce the term “snapshot of the document” to refer to the part of the document that is displayed on the screen at a certain moment. When the user changes the part of the document being displayed, we say a new snapshot is formed. This happens for example when the user resizes the displaying window or scrolls to a different part of the document. For each snapshot of the document, we assign the gaze samples onto the corresponding words in the document in a fractional manner. To carry this out, we introduce a Gaussian kernel in the assignment process. Assuming at a certain moment, the detected gaze central point is positioned at  $(x, y)$  on the screen space. For each word  $w_i$  that is displayed in the current document snapshot, we first compute the central displaying point of the word as the center of the bounding box of the word’s displaying region, which is denoted as  $(x_i, y_i)$ . Then the fraction of the gaze sample the word  $w_i$  receives is:

$$AT(w_i) = \exp\left(-\frac{(x_i - x)^2}{2\sigma_x^2} - \frac{(y_i - y)^2}{2\sigma_y^2}\right). \quad (1)$$

The free parameters  $\sigma_x$  and  $\sigma_y$  specify how “diffusively” a reader scans words when reading documents. In our current implementation, we initialize  $\sigma_x$  and  $\sigma_y$  to be the average width and height of a word’s displaying bounding box in the document. For each gaze detected by our eye-tracking module, we assign the gaze samples to the words in the document in this manner. The overall attention that a word in the document receives is the sum of all the fractional gaze samples it is assigned in the above process. Notice that when a

word occurs multiple times in the document, we accumulate all the gaze samples assigned to these occurrences. Finally, the overall attention of a user over a word is the sum of the word’s attention across all the documents the user has read previously. During processing, we remove the stop words ([http://en.wikipedia.org/wiki/Stop\\_words](http://en.wikipedia.org/wiki/Stop_words)) since they are not providing much meaning and thus should not really have attracted the user attention. Notice that for words in the documents that are not displayed, their attention is unspecified rather than being assigned zero.

## PREDICTION OF USER ATTENTION OVER A SENTENCE

### Predicting User Attention for Words

Our attention time prediction for a word is based on the semantic similarity of two words. We assume if the semantics of two words are sufficiently similar semantically, then a user shall have more or less the same amount of interest to read either of them. We use  $Sim(w_i, w_j)$  to denote the semantic similarity between word  $w_i$  and word  $w_j$ , where  $Sim(w_i, w_j) \in [0, 1]$ . Correct estimation on  $Sim(w_i, w_j)$  plays a critical role in our attention time prediction algorithm. Fortunately, there exists a large collection of algorithms on word semantic distance estimation. In our current system implementation, we calculate  $Sim(w_i, w_j)$  using the semantic similarity measuring algorithm proposed in [27].

We denote the user  $U_j$ ’s attention samples on the words  $w_1, \dots, w_n$  as  $AT(w_1, U_j), \dots, AT(w_n, U_j)$  respectively, which are acquired through our vision-based eye-tracker. For an arbitrary word  $w$  which is not among  $\{w_1, \dots, w_n\}$ , we calculate the similarity between  $w$  and every  $w_i (i = 1, \dots, n)$ . We then select  $k$  words which share the highest semantic similarity with  $w$ . In our current experiment,  $k$  is set as  $\min(10, n)$ , where  $n$  is the number of attention samples acquired via eye-tracking. Without loss of generality and for ease of notation, we assume they are the words  $w_i (i = 1, \dots, k)$ . Then we use the following equation to predict the attention time for  $w$ :

$$AT(w, U_j) = \frac{\sum_{i=1}^k (AT(w_i, U_j) Sim(w_i, w) \rho(w_i, w))}{\sum_{i=1}^k (Sim(w_i, w) \rho(w_i, w)) + \epsilon}, \quad (2)$$

where  $\epsilon$  is a small positive number to avoid the divide-by-zero error. The function of  $\rho(\cdot)$  filters out the effects of those documents whose similarity is below a certain threshold and is defined as:

$$\rho(w_i, w) = \begin{cases} 1 & \text{If } Sim(w_i, w) > 0.1; \\ 0 & \text{Otherwise.} \end{cases} \quad (3)$$

### Predicting User Attention for Sentences

We estimate the total attention of a certain user on a sentence as the sum of the user’s attention over all the words in the sentence. Mathematically, assuming a sentence  $s$  in a target document to be summarized consists of  $n$  distinct words  $w_1, \dots, w_n$ . We can then predict user  $U_j$ ’s attention over the sentence  $s$ , denoted as  $AT(s, U_j)$ , as follows:

$$AT(s, U_j) = \sum_{w_i \in s} AT(w_i, U_j) \delta(w_i, U_j). \quad (4)$$

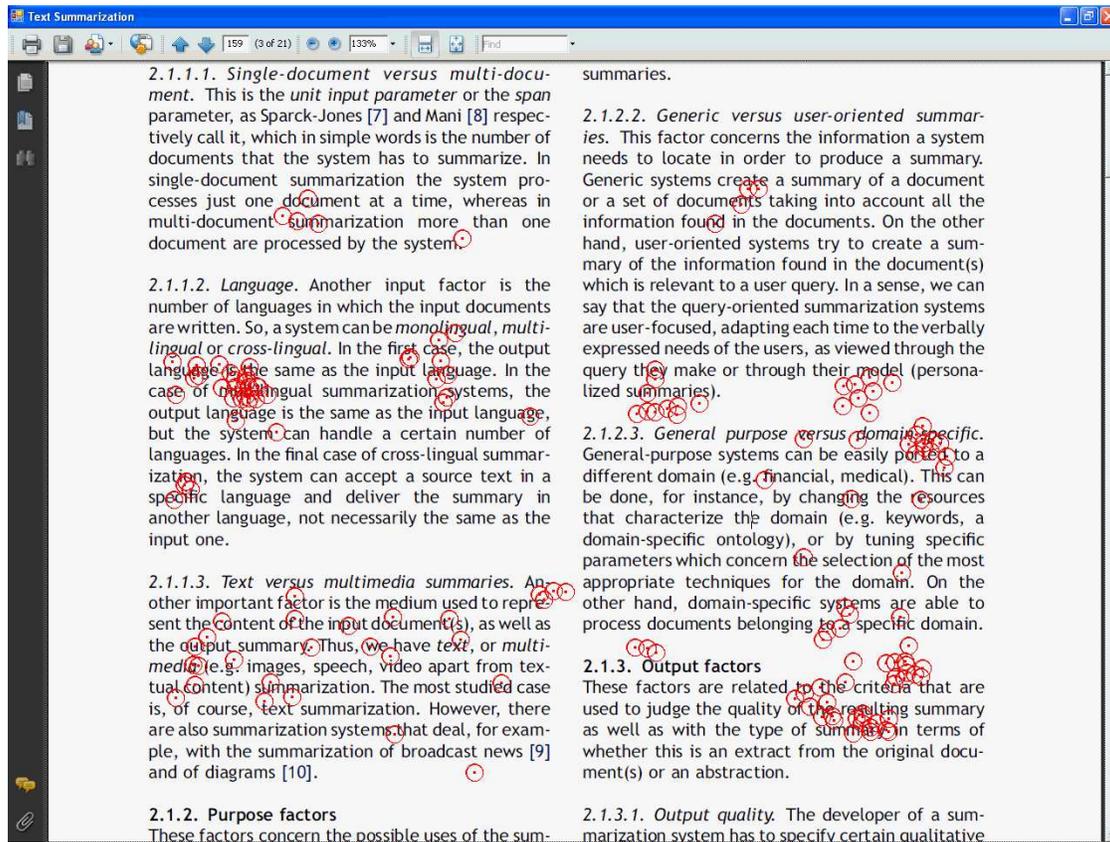


Figure 1. A snapshot of our custom document browser which works together with our vision-based eye tracking component to acquire user attention time samples. Each red circle represents a fixation point of the user. Only points inside the client area of the browser window are recorded. Except during debugging, these red circles will not be displayed. They are shown here for the purpose of illustration.

In the above, recall  $AT(w_i, U_j)$  is user  $U_j$ 's attention over the word  $w_i$ , which is either sampled from the user's previous reading activities via (1) or predicted via (2) if our algorithm has never sampled this word during this user's previous reading process. The term  $\delta(w_i, U_j)$  in the above equation gives different weights to the per word user attention term  $AT(w_i, U_j)$ . Currently, we configure the weighting function  $\delta(w_i, U_j)$  in the following way:  $\delta(w_i, U_j) = 0$  if the word  $w_i$  is a stop word;  $\delta(w_i, U_j) = 0.6$  if there is no attention sample for the user  $U_j$  over the word  $w_i$  known to our algorithm and thus has to be estimated via (2); otherwise,  $\delta(w_i, U_j) = 1$ , which means the user  $U_j$ 's attention over the word  $w_i$  is acquired from sampling over  $U_j$ 's previous reading activities via (1).

## PERSONALIZED DOCUMENT SUMMARIZATION

Now we can construct a personalized document summarization algorithm based on the acquired and predicted user attention data for individual users. To experiment with our algorithm, we developed a prototype program, which consists of a custom document browser interface for acquiring the gaze samples of individual users on individual words in a document, and a summarization module for producing a personalized document summarization based on the prediction of user attention on the target document to be automatically summarized.

## Custom Document Browser Interface

For the custom document browser interface, which looks very much like a document browser, the acquisition method introduced earlier is implemented. It operates with the vision-based eye-tracking device we assembled using a simple web camera (Logitech Quickcam Notebook Pro) and an existent eye-tracking algorithm coming from the Opengazer project [47]. The captured user gaze samples are provided to the summarization module. Any fixation points outside the browser window will be ignored during the user attention sampling process. A snapshot of our custom document browser interface is shown in Figure 1.

## Summarization Module

After the user attentions on all the individual words in a target document are known, which are either sampled from or predicted based on the user's previous reading processes, the whole document can be represented as a collection of sentences or key phrases, each of which has a known attention value. The functionality of our summarization module is to select several key sentences or phrases from the whole document which can best satisfy the user's reading interests. Our summarization algorithm can work at either the sentence level or the key word or phrases level, and in principle can be extended to sentence groups or paragraphs without having to change the algorithm in any way. The selected top sentences

or keywords that the user is most likely interested in form the personalized summarization of the document. Similarly, if we select the few top words from the whole document as the output, we have an algorithm for personalized keyword extraction.

In a typical summarization software package, end users can tune a free parameter called “compression rate” to specify the ratio between the length of the summarized texts and the length of the whole document. In some advanced summarization packages, more control parameters are allowed to specify the expectation over the summarization algorithm’s behavior. For simplicity, we are only concerned with the compression rate. More concretely, inspired by the summarization software provided in Microsoft Office Professional Edition 2003, called “Microsoft Word AutoSummarize”, we introduce a percentage parameter  $c$  into our algorithm, which restricts our algorithm to only output the top  $c\%$  sentences (with highest user attention times) as the summarization result for the target document.

### A Hybrid Summarization Approach

In our early experiments, we noticed that the performance of our user-oriented document summarization algorithm heavily depends on the amount of available user attention time samples. As with many machine learning based algorithms, our algorithm also has the cold start problem, i.e., if there are not enough samples to learn from, the machine summarized result appears to be inferior. To address the issue, we integrate our method with a conventional automatic document summarization algorithm, resulting in a hybrid approach for document summarization. The MEAD summarizer (or simply MEAD) [35], which is a portable multi-document summarization system, is adopted to perform the conventional summarization function in our hybrid document summarization approach. The reason for choosing this system is that it is in the public domain and yet its performance is comparable to other state-of-the-art systems. When the summarization module receives a summarization request submitted by a certain user, the application will first forward the request to MEAD to obtain the document’s summarized text. Then our user attention time prediction module will estimate the attention time of the user over each sentence in the document. After that, we use the following equation to compute an attention time offset for the user, denoted as  $U_j$ , on an arbitrary document sentence  $s_i$ :

$$AT_{offset}(s_i, U_j) \triangleq (1 - \kappa) \max_{k=1}^n \{AT(s_k, U_j)\} \widehat{\delta}(s_i, U_j), \quad (5)$$

where  $\widehat{\delta}(s_i, U_j) = 1$  if sentence  $s_i$  is selected by MEAD in its document summarization result and  $\widehat{\delta}(s_i, U_j) = 0$  otherwise. With such a value assignment for the function  $\widehat{\delta}(\cdot)$ , document sentences appearing in the summarization result produced by MEAD receive higher attention time offset values than those not included in the summary generated by MEAD. The free parameter  $\kappa$  used in (5) is user tunable, which balances the document summarization result produced by MEAD and the result generated by our user-oriented summarization method. When  $\kappa = 1$ , our hybrid

**Table 1. Statistics of the two sets of articles used in our experiment.** “Set I” corresponds to the set of articles on science topics and “Set II” is the set of articles on topics about entertainment and leisure. “Manual compression rate” is the average ratio between the word length of the user manual summary and the word length of the corresponding original article.

Article set	I	II	I + II
Articles in the set	60	60	120
Words per article	979.0	942.3	960.7
Sentences per article	37.6	53.2	45.4
Paragraphs per article	9.1	11.3	10.2
Sentences per manual summary	12.4	14.7	13.6
Manual compression rate	33.0%	27.6%	29.8%

document summarization approach operates without employing the MEAD procedure; when  $\kappa = 0$ , our hybrid approach degenerates into a summarization procedure performed solely by MEAD. In our experiment,  $\kappa$  is set as 0.5 by default.

Once the user  $U_j$ ’s attention time  $AT(s_i, U_j)$  and attention offset time  $AT_{offset}(s_i, U_j)$  are both known for a sentence  $s_i$ , we can derive a calibrated attention time for the sentence as:

$$AT_{cal}(s_i, U_j) \triangleq AT(s_i, U_j) + AT_{offset}(s_i, U_j). \quad (6)$$

After we derive the calibrated user attention times for all the sentences in the target document, our targeted document summarization can be produced by selecting the top  $c\%$  sentences in the document that achieve the largest overall user attention, i.e., the sum of these sentences’ calibrated attention times will be maximized.

### EXPERIMENT RESULTS

Two general classes of methods have been proposed for evaluating the performance of text summarization: extrinsic evaluations and intrinsic evaluations [32]. Extrinsic evaluations verify the quality of summarization texts according to the number of citations on the summarized texts by third parties; intrinsic evaluations determine the quality of summarization texts based on the coverage comparison with a ground truth summarization result. In this paper, we evaluate the quality of our document summarization result using the intrinsic evaluation approach because of its ease of execution. More concretely, we evaluate the performance of our algorithm in terms of summarization quality by comparing the document summarization results produced by our algorithm with those generated by two popular text summarization algorithms. To evaluate the summarization quality of any of the three algorithmic approaches, we compare the machine generated summary with the corresponding human summary result using the intrinsic evaluation method.

In our experiments, we use two sets of articles. Articles in the first set are all about science and articles in the second set are all about entertainment and leisure. We choose these two sets of articles because they cover two distinctive groups of readers and hence two types of reading behaviors—one tends to be more serious than the other one. Sixty scientific articles are randomly selected from the website of the

**Table 2. Statistics for comparing the performance of three automatic document summarization algorithms: the “Microsoft Word AutoSummarize” toolkit, the MEAD summarizer system and our algorithm. We provide performance evaluation statistics for each algorithm for three typical compression rates, 10%, 20%, 30%. In all these experiments, our algorithm consistently outperforms the other two summarization algorithms.**

Summarization Algorithm	Compression Rate								
	10%			20%			30%		
	Recall	Precision	F-rate	Recall	Precision	F-rate	Recall	Precision	F-rate
MS Word AutoSummarize	0.13	0.25	0.17	0.20	0.27	0.23	0.23	0.27	0.25
MEAD	0.18	0.47	0.26	0.25	0.44	0.32	0.30	0.42	0.35
Our Algorithm	0.28	0.64	0.39	0.42	0.60	0.49	0.53	0.58	0.55

(a) Algorithm performance statistics for article set I

Summarization Algorithm	Compression Rate								
	10%			20%			30%		
	Recall	Precision	F-rate	Recall	Precision	F-rate	Recall	Precision	F-rate
MS Word AutoSummarize	0.16	0.23	0.19	0.21	0.28	0.24	0.23	0.30	0.26
MEAD	0.18	0.36	0.24	0.26	0.45	0.33	0.29	0.60	0.39
Our Algorithm	0.25	0.70	0.37	0.44	0.64	0.52	0.56	0.61	0.58

(b) Algorithm performance statistics for article set II

Summarization Algorithm	Compression Rate								
	10%			20%			30%		
	Recall	Precision	F-rate	Recall	Precision	F-rate	Recall	Precision	F-rate
MS Word AutoSummarize	0.15	0.24	0.18	0.20	0.27	0.23	0.23	0.29	0.26
MEAD	0.18	0.41	0.25	0.25	0.44	0.32	0.30	0.51	0.37
Our Algorithm	0.27	0.67	0.38	0.43	0.62	0.50	0.54	0.59	0.57

(c) Algorithm performance statistics for both article sets

“Science” magazine [11] to form the first article set, and another sixty articles are randomly selected from the travel and sports section on “New York Times” [6] to form the second article set. We also invited twelve people with different knowledge backgrounds to read some selected articles from the two article sets using our custom document browser equipped with our eye-tracking device. After reading each article, they are asked to provide a summary for the article they just read. In this way, we develop a manual summary set for the two sets of articles. Key statistics on the two sets of articles are reported in Table 1.

Inspired by the typical way to measure the performance of an information retrieval system [2], three measurements—Recall ( $R$ ), Precision ( $P$ ) and F-rate ( $F$ )—are introduced to evaluate the machine summarization quality against the human summary result. Let  $SU_e$  be the human summary result for an article, and  $SU$  be the summary automatically generated by an algorithm. We then calculate the above three measurements as follows:

$$P \triangleq \frac{\text{Number of common sentences in } SU_e \text{ and } SU}{\text{Number of sentences in } SU}, \quad (7)$$

$$R \triangleq \frac{\text{Number of common sentences in } SU_e \text{ and } SU}{\text{Number of sentences in } SU_e}, \quad (8)$$

$$F \triangleq \frac{2PR}{P + R}. \quad (9)$$

As mentioned earlier, we asked the twelve people to help us develop human summaries for our article sets. In our experiment, for each participant, the person was asked to read some selected articles from the two article sets using our cus-

tom document browser with our eye-tracking device turned on. After reading an article, the person was further asked to provide a summary for the article he just read. And then our algorithm used the person’s user attention time samples over the article to generate a machine summary of the article for the user. Given the machine generated summary and the user provided manual summary, we can measure the quality of the article summarization produced by our algorithm using (7)–(9). Finally, the overall performance of our algorithm is measured as the average performance of our algorithm for all the articles read and summarized by these twelve participants.

We compare the performance of our algorithm with two popular document summarization software packages—“Microsoft Word AutoSummarize” as provided in Microsoft Office Professional Edition 2003 and the MEAD summarizer system. Concrete statistics for comparing the performance of the three algorithms in summarizing documents are provided in Table 2. As revealed by these experiment results, our algorithm consistently outperforms the other two algorithms with significant margins.

As discussed earlier, our hybrid document summarization approach relies on a modulating parameter  $\kappa$  to balance its behavior between the traditional discourse analysis algorithm and our personalized document summarization algorithm. We also conduct an experiment to evaluate the performance of our hybrid approach under different settings for the parameter  $\kappa$ . Results of the experiment are shown in Table 3. In this experiment, the document compression rate is always set as 30%. From these experiment results, we can see the hybrid approach can effectively help us to overcome the cold

**Table 3. Performance measurement statistics of our summarization algorithm under different settings for the free parameter  $\kappa$  which balances our hybrid document summarization approach between the pure discourse analysis based approach and the machine learning based approach. (a)–(c) report performance measurement statistics for article sets I and II, and for both sets. The experiment repeats with different values for  $\kappa$  (0.00, 0.25, 0.50, 0.75, 1.00). In all these experiments, the compression rate is always set to 30%.**

Measurement	$\kappa$				
	0.00	0.25	0.50	0.75	1.00
Recall	0.33	0.41	0.53	0.58	0.61
Precision	0.35	0.44	0.58	0.65	0.70
F-rate	0.34	0.42	0.55	0.61	0.63

(a) Performance measurement statistics for article set I

Measurement	$\kappa$				
	0.00	0.25	0.50	0.75	1.00
Recall	0.39	0.44	0.56	0.56	0.46
Precision	0.42	0.48	0.61	0.60	0.50
F-rate	0.40	0.45	0.58	0.58	0.47

(b) Performance measurement statistics for article set II

Measurement	$\kappa$				
	0.00	0.25	0.50	0.75	1.00
Recall	0.36	0.42	0.54	0.57	0.54
Precision	0.38	0.46	0.59	0.62	0.60
F-rate	0.37	0.44	0.57	0.60	0.55

(c) Performance measurement statistics for both article sets

start problem when there are not enough learning samples for our user-oriented document summarization algorithm to perform. Also, from this experiment we can see for summarizing articles on entertainment and leisure topics where personal reading interests vary significantly, using our hybrid document summarization approach is especially advantageous.

In conclusion, by the results of the above experiments, we confirm that our user-oriented document summarization algorithm can indeed produce personalized document summaries that are more reflective of a user’s reading interest and summary preference than other approaches.

## CONCLUSION AND FUTURE WORK

In this paper, we propose a new user-oriented document summarization algorithm based on individual users’ attention time on document words and sentences. In our experiments, we compare the performance of our algorithm with both the human summarization result and two popular existing text summarization algorithms implemented in the Microsoft Word AutoSummarize toolkit and the MEAD summarizer system respectively. The experiment results clearly show that our novel algorithm can satisfactorily produce user-oriented document summarization in better agreement with the user’s expectation and preference.

Having validated the domain specific prototype document summarization system we have developed here via experimentation, we hope to demonstrate more the potential of employing commodity eye-tracking techniques for acquiring pervasive and unintrusive user feedbacks in building various

types of user-oriented information retrieval and extraction systems in the future.

Achieving an optimal balance between document summarization following the traditional discourse analysis approach and our learning based approach is an open problem. In our current implementation, such a decision is left to the end users via a user tunable free parameter. In the future we plan to explore adaptive methods which can adjust this parameter following a reinforcement learning based method via utilizing some user feedbacks. In that way, we expect to develop an intelligent algorithm for assisting users to find the optimal balance between traditional discourse analysis based summarization and our machine learning based personalized summarization in order to generate better user-oriented document summarizations that match more closely with individual users’ preferences.

Right now, when producing user-oriented document summarization for a user, our algorithm relies on the use of the user’s attention time samples for the current document. This restricts our current algorithm to be applicable only in scenarios where a person has read an article and wants to make an automatic summary for the article for future reference by himself or for later readers. We plan to explore the possibility to remove this constraint so that even for an article which the user has not read, it is still possible to generate automatic user-oriented summary of the article for the user.

In our algorithm, estimating semantic similarity between two words or sentences is crucial for producing a quality user-oriented document summarization. Hence in the future, we intend to improve the text content similarity metrics used in our system by incorporating some human intelligence through learning the human feedbacks. This can be realized as an on-line learning algorithm. We also intend to strengthen the data mining capabilities of our algorithm, especially to optimize the performance of our algorithm in predicting user attention over words and sentences. Finally, setting up a robust user-oriented document summarization system based on our algorithm for large-scale industrial strength deployment would be very meaningful and commercially attractive.

## ACKNOWLEDGEMENTS

The first author would like to thank David Gelernter for many inspiring discussions on document summarization research, from which the idea of this paper stemmed. This work has a patent pending.

## REFERENCES

1. Bibliography summarization papers. <http://www.summarization.com/summ.pdf>, last updated on October 20, 2008. *last visited on December 11, 2008.*
2. R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.*
3. A. Bulling, J. A. Ward, H. Gellersen, and G. Tröster. *Robust recognition of reading activity in transit using*

- wearable electrooculography. In *Pervasive '08: Proceedings of the 6th International Conference on Pervasive Computing*, pages 19–37, 2008.
4. A. Bulling, D. Roggen, and G. Tröster. *It's in your eyes: towards context-awareness and mobile HCI using wearable EOG Goggles*. In *UbiComp '08: Proceedings of the 10th International Conference on Ubiquitous Computing*, pages 84–93, New York, NY, USA, 2008. ACM.
  5. E. H. Chi, M. Gumbrecht, and L. Hong. *Visual foraging of highlighted text: An eye-tracking study*. In *HCI '07: Proceedings of HCI International Conference*, pages 589–598, 2007.
  6. *The New York Times Company. The New York Times*, <http://www.nytimes.com/>, last visited on December 11, 2008.
  7. T. Darrell, N. Checka, A. Oh, and L. Morency. *Exploring vision-based interfaces: How to use your head in dual pointing tasks*. MIT AI Memo 2002-001, 2002.
  8. Z. Dou, R. Song, and J.-R. Wen. *A large-scale evaluation and analysis of personalized search strategies*. In *WWW '07: Proceedings of International Conference on World Wide Web*, pages 581–590, New York, NY, USA, 2007. ACM.
  9. G. Dupret, V. Murdock, and B. Piwowarski. *Web search engine evaluation using clickthrough data and a user model*. In *WWW '07: Proceedings of International Conference on World Wide Web*, Banff, Canada, 2007.
  10. G. Erkan and D. Radev. *Lexrank: Graph-based lexical centrality as salience in text summarization*. *Journal of Artificial Intelligence Research (JAIR)*, 22:457–479, 2004.
  11. *American Association for the Advancement of Science. Science magazine*, <http://www.sciencemag.com/>, last visited on December 11, 2008.
  12. S. Fox, K. Karnawat, M. Mydland, S. Dumais, and T. White. *Evaluating implicit measures to improve web search*. *ACM Transactions on Information Systems*, 23(2):147–168, 2005.
  13. X. Fu. *Evaluating sources of implicit feedback in web searches*. In *RecSys '07: Proceedings of the 1st ACM International Conference on Recommender Systems*, pages 191–194, New York, NY, USA, 2007. ACM.
  14. Y. Gong and X. Liu. *Generic text summarization using relevance measure and latent semantic analysis*. In *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 19–25, New York, NY, USA, 2001. ACM.
  15. D. Gorodnichy. *Perceptual cursor—a solution to the broken loop problem in vision-based hands-free computer control devices*. National Research Council Canada Publication, NRC-48472:1–23, 2006.
  16. L. A. Granka, T. Joachims, and G. Gay. *Eye-tracking analysis of user behavior in www search*. In *SIGIR '04: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 478–479, New York, NY, USA, 2004. ACM.
  17. Z. Guan and E. Cutrell. *An eye tracking study of the effect of target rank on web search*. In *CHI '07: Proceedings of SIGCHI Conference on Human Factors in Computing Systems*, pages 417–420, New York, NY, USA, 2007. ACM.
  18. W. S. A. Halabi, M. Kubat, and M. Tapia. *Time spent on a web page is sufficient to infer a user's interest*. In *IMSA '07: Proceedings of IASTED European Conference*, pages 41–46, Anaheim, CA, USA, 2007. ACTA Press.
  19. C.-K. Huang, Y.-J. Oyang, and L.-F. Chien. *A contextual term suggestion mechanism for interactive web search*. In *WI '01: Proceedings of Asia-Pacific Conference on Web Intelligence: Research and Development*, pages 272–281, London, UK, 2001. Springer-Verlag.
  20. T. Joachims. *A probabilistic analysis of the rocchio algorithm with tfidf for text categorization*. In *ICML '97: Proceedings of International Conference on Machine Learning*, pages 143–151, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.
  21. T. Joachims. *Optimizing search engines using clickthrough data*. In *KDD '02: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142, New York, NY, USA, 2002. ACM.
  22. T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. *Accurately interpreting clickthrough data as implicit feedback*. In *SIGIR '05: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 154–161, New York, NY, USA, 2005. ACM.
  23. K. S. Jones. *What might be in a summary*. In *Information Retrieval 93: Von der Modellierung zur Anwendung*, pages 9–26, 1993.
  24. D. Kelly and N. J. Belkin. *Reading time, scrolling and interaction: exploring implicit sources of user preferences for relevance feedback*. In *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 408–409, New York, NY, USA, 2001. ACM.
  25. D. Kelly and N. J. Belkin. *Display time as implicit feedback: understanding task effects*. In *SIGIR '04: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 377–384, New York, NY, USA, 2004. ACM.

26. K.-N. Kim and R. Ramakrishna. *Vision-based eye-gaze tracking for human computer interface*. SMC '99: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, 2:324–329, 1999.
27. Y. Li, Z. A. Bandar, and D. Mclean. *An approach for measuring semantic similarity between words using multiple information sources*. IEEE Transactions on Knowledge and Data Engineering, 15(4):871–882, 2003.
28. C.-Y. Lin. *Training a selection function for extraction*. In CIKM '99: Proceedings of the 8th ACM International Conference on Information and Knowledge Management, pages 55–62, New York, NY, USA, 1999. ACM.
29. Y.-P. Lin, Y.-P. Chao, C.-C. Lin, and J.-H. Chen. *Webcam mouse using face and eye tracking in various illumination environments*. EMBS '05: Proceedings of 27th IEEE Annual International Conference of Engineering in Medicine and Biology Society, pages 3738–3741, 2005.
30. F. Liu, C. Yu, and W. Meng. *Personalized web search by mapping user queries to categories*. In CIKM '02: Proceedings of the 11th ACM International Conference on Information and Knowledge Management, pages 558–565, New York, NY, USA, 2002. ACM.
31. Y. Lv, L. Sun, J. Zhang, J.-Y. Nie, W. Chen, and W. Zhang. *An iterative implicit feedback approach to personalized search*. In ACL '06: Proceedings of International Conference on Computational Linguistics, pages 585–592, Morristown, NJ, USA, 2006. Association for Computational Linguistics.
32. I. Mani. *Advances in Automatic Text Summarization*. MIT Press, Cambridge, MA, USA, 1999.
33. Microsoft. *Word (software)*, <http://office.microsoft.com/word/>, Microsoft Corporation, last visited on December 11, 2008.
34. J. Pitkow, H. Schütze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. *Personalized search*. Communications of the ACM, 45(9):50–55, 2002.
35. D. Radev, T. Allison, S. Blair-Goldensohn, J. Blitzer, A. Çelebi, S. Dimitrov, E. Drabek, A. Hakim, W. Lam, D. Liu, J. Otterbacher, H. Qi, H. Saggion, S. Teufel, M. Topper, A. Winkel, and Z. Zhang. *MEAD—A platform for multidocument multilingual text summarization*. In LREC '04: The 2nd International Conference on Language Resources and Evaluation, Lisbon, Portugal, 2004.
36. F. Radlinski and T. Joachims. *Query chains: learning to rank from implicit feedback*. In KDD '05: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pages 239–248, New York, NY, USA, 2005. ACM.
37. R. W. Reeder, P. Pirolli, and S. K. Card. *Webeyemapper and weblogger: tools for analyzing eye tracking data collected in web-use studies*. In CHI '01: CHI '01 Extended Abstracts on Human Factors in Computing Systems, pages 19–20, New York, NY, USA, 2001. ACM.
38. R. Ruddaraju, A. Haro, K. Nagel, Q. T. Tran, I. A. Essa, G. Abowd, and E. D. Mynatt. *Perceptual user interfaces using vision-based eye tracking*. In ICMI '03: Proceedings of 5th International Conference on Multimodal Interfaces, pages 227–233, New York, NY, USA, 2003. ACM.
39. G. Salton and C. Buckley. *Improving retrieval performance by relevance feedback*. Journal of the American Society for Information Science, 41(4):288–297, 1990.
40. M. Speretta and S. Gauch. *Personalized search based on user search histories*. In WI '05: Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence, pages 622–628, Washington, DC, USA, 2005. IEEE Computer Society.
41. M.-C. Su, S.-Y. Su, and G.-D. Chen. *A low-cost vision-based human-computer interface for people with severe disabilities*. Biomedical Engineering Applications, Basis, and Communications, 17:284–292, 2005.
42. R. White, J. M. Jose, and I. Ruthven. *Comparing explicit and implicit feedback techniques for web retrieval: Trec-10 interactive track report*. In TREC, 2001.
43. R. White, I. Ruthven, and J. M. Jose. *The use of implicit evidence for relevance feedback in web retrieval*. In Proceedings of BCS-IRSG European Colloquium on IR Research, pages 93–109, London, UK, 2002. Springer-Verlag.
44. S. Xu, H. Jiang, and F. C. Lau. *Personalized online document, image and video recommendation via commodity eye-tracking*. In RecSys '08: Proceedings of the 2nd ACM International Conference on Recommender Systems, pages 83–90, New York, NY, USA, 2008. ACM.
45. S. Xu, Y. Zhu, H. Jiang, and F. C. M. Lau. *A user-oriented webpage ranking algorithm based on user attention time*. In AAI '08: Proceedings of the 23rd AAI Conference on Artificial Intelligence, pages 1255–1260, Chicago, Illinois, USA, 2008. AAI.
46. J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I.-H. Meng. *Text summarization using a trainable summarizer and latent semantic analysis*. Information Processing and Management, 41(1):75–95, 2004.
47. P. Zielinski. *Opengazer: open-source gaze tracker for ordinary webcams (software)*, Samsung and The Gatsby Charitable Foundation. <http://www.inference.phy.cam.ac.uk/opengazer/>, last visited on December 11 2008.