Query Expansion Using Gaze-Based Feedback on the Subdocument Level

Georg Buscher[1,2], Andreas Dengel[1,2], and Ludger van Elst[2]
[1]Dept. for Knowledge-Based Systems, University of Kaiserslautern
[2]Knowledge Management Dept., DFKI
Kaiserslautern, Germany
{georg.buscher, andreas.dengel, ludger.van_elst}@dfki.de

ABSTRACT

We examine the effect of incorporating gaze-based attention feedback from the user on personalizing the search process. Employing eye tracking data, we keep track of document parts the user read in some way. We use this information on the subdocument level as implicit feedback for query expansion and reranking.

We evaluated three different variants incorporating gaze data on the subdocument level and compared them against a baseline based on context on the document level. Our results show that considering reading behavior as feedback yields powerful improvements of the search result accuracy of ca. 32% in the general case. However, the extent of the improvements varies depending on the internal structure of the viewed documents and the type of the current information need.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance Feedback*

General Terms

Algorithms, Design, Experimentation, Measurement

Keywords

Personalization, implicit feedback, eye tracking, reading

1. INTRODUCTION

The importance of personalization in information retrieval (IR) increased over the last years because the incorporation of user preferences and user context can substantially enhance the quality of search results. On the one hand, methods for long-term modeling of the user's persistent interests have been developed. They range from analyzing the user's personal document collection [7, 9] to observing the user's actions like issued queries, visited web sites, etc. [26]. On the other hand, methods for short-term modeling of the user's immediate information needs have been investigated. They take the user's current actions into account like viewing, scrolling, and querying behavior (see Kelly and Teevan [16] for an overview). In our work we focus on short-term modeling of the user's interests and especially investigate the effect of incorporating eye tracking information.

Eye trackers rapidly evolved in recent years. Nowadays, they have reached such a state of development that they are unobtrusive and easy to use and that their accuracy is sufficient for many practical applications. Since their development is progressing further, there is a good chance that they will become more affordable and, as a consequence, become more widespread. By applying eye trackers as evidence source we are able to gather very detailed and precise data about how the user works with documents on the screen. Hence, it is worth studying how eye movement data can be used effectively in the process of information retrieval [4].

Eye trackers can be applied specifically to collect information about *which document parts* the user looked at and *how* he or she looked at them. In contrast to most other techniques that gather such implicit feedback on the *document level* (e.g., history-based techniques like documents opened [25], time spent on a document [15], etc.), eye tracking goes down to the *subdocument level*.

This is important, because documents often contain several topics that are somewhat related but clearly different when viewed from a more detailed perspective. Extreme examples are textbooks containing many chapters, e.g., a text book about medicine containing a chapter about simple aspects of physics important for understanding some processes in the human body. On the side of the search engine, such a heterogeneous topical structure of a document can be taken into account by applying algorithms for passage-based retrieval or by decomposing a document in topically different parts [28], for example. However, on the user's side, if a document is displayed it is difficult to determine which parts of the document actually matter to the user, i.e., the parts he or she actually works with. In this regard, an eye tracker provides very useful information about which parts of the document the user pays attention to. Thus, one can expect that eye trackers are very useful sources for more precise implicit feedback.

In this paper, we describe a user study to investigate whether and how much different variants considering eye movements to elicit the user's short-term context can enhance the quality of retrieval by query expansion and reranking. The general scenario is to record which document parts

SIGIR'08, July 20–24, 2008, Singapore.
Copyright 2008 ACM 978-1-60558-164-4/08/07 ...$5.00.

were paid attention to immediately before querying a search engine. Based on this information, we extract characteristic terms of the viewed document *parts* and use them for expanding the user's query. The expanded query is then applied for result reranking. We examine three eye tracking-based variants for query expansion and compare them against a baseline which operates on the document level, i.e., it extracts terms from *full-text* documents.

The paper is organized as follows: First, we provide background information on personalization methods and on the utilization of eye tracking in information retrieval. Then we describe our study design, first from the perspective of the participants and then from a technical perspective. The results of the experiment are viewed and analyzed from different angles and on different levels of detail. The paper concludes with a short discussion.

2. BACKGROUND AND RELATED WORK

In this paper, we combine research of two areas that were hardly related until now: implicit relevance feedback for personalized, improved result ranking, and applications of eye tracking in IR. In the following, a short overview of the relevant literature of both areas is given.

2.1 Implicit Feedback for Improved Ranking

A lot of research has been done in the area of generating implicit feedback for improved retrieval quality. One possibility is to subdivide the approaches into those interpreting concrete user actions and those analyzing the more or less static user environment.

Concerning user actions, there is a lot of work studying how implicit measures relate to explicit relevance feedback for documents from the user. Many papers primarily focus on the relation between them but do not actually apply the implicit measures for improved ranking (e.g., display time [15], click-through data [14], combinations of those, mouse movements, scrolling, etc. [2, 10]). It turned out that most of the implicit measures are indeed correlated with explicit relevance feedback in laboratory studies. In real world studies, however, some of them (especially display time [15]) are difficult to interpret.

Further studies aim at incorporating implicit feedback from user actions directly for an improved ranking function. Papers of Qiu and Cho [19] as well as Radlinski and Joachims [20] show that when considering click-through data the ranking of search results can significantly be enhanced. Agichtein et al. [1] go a step further and include a lot of additional user behavior measures. They prove in a large real-world study that core state-of-the-art ranking as well as reranking functions can be remarkably improved by learning how to weight the various implicit feedback measures (i.e., a gain of up to 24% in Mean Average Precision, and a gain of ca. 25% in Precision@5).

Work that is structurally similar to ours is that of Shen et al. [23] and Sugiyama et al. [25]. Both studies consider the browsing history of a user to capture the current query context. The former also considers click-through data and the query history. A context model is then used for query expansion (query reformulation, respectively) and for reranking of the results from a commercial web search engine. A case study in [23] shows a considerable improvement of precision after reranking the original results of the search engine (gain in Precision@5: 8%).

Instead of taking the immediate browsing history of the user as query context, Chirita et al. [7] consider the entire document collection stored on the user's computer as contextual user environment. They evaluate several methods for term extraction on the user's personal document collection and apply the extracted terms for query expansion. The best term extraction method overall yielded a Normalized Discounted Cumulative Gain (NDCG, [13]) of ca. 16% compared to no query expansion. Before, Teevan et al. [26] conducted a similar study, but modified the weighting schema of the core retrieval function for reranking instead of performing query expansion. Among other things, they found that richer representations of the user model yield better retrieval results.

All previously mentioned work gathered and included implicit feedback data only on the level of *entire, full-text* documents. We are only aware of very few IR personalization studies that try to go beyond the document level down to the subdocument level. Golovchinsky et al. [11] studied the effect of considering user-generated annotations like highlightings, underlinings, and circles as markers of interest. They used this kind of information on the word, sentence and paragraph level for the generation of personalized queries and compared it to a standard relevance feedback scenario on the document level. The annotation-based query generation technique performed significantly better. However, in real-world settings, manually created user annotations are rather rare. A second study by Yang et al. [27] examined the effect of using explicit relevance feedback on the passage level. However, they found that this produced too much additional cognitive load for the user.

By applying eye tracking, the problems of too much additional cognitive load and too less feedback data could be overcome.

2.2 Eye Tracking in IR

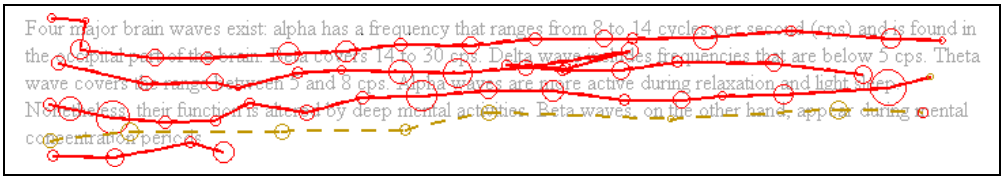
So far, eye trackers have not been applied very frequently to enhance information retrieval aspects. One of the most common areas of application are usability studies. For example, Granka et al. [12] used eye movement data to get a better understanding of how search result pages are used and how click-through data can be interpreted more accurately as implicit feedback. More recently, Cutrell and Guan [8] used gaze data to get insights about issues concerning result list presentation.

Besides, there are studies that used eye trackers actively as input devices. For example, when a user looked at a result entry for a while, Maglio et al. [17] interpreted that as user interest and opened the respective document automatically.

A relatively often approached field is to use eye movements as implicit feedback. Some of them study the correlations between eye movements and explicit relevance feedback, e.g., [3, 22]. They aim at automatically predicting relevance for a document in a classical relevance feedback scenario (compare Rocchio [21]). Other studies try to incorporate eye movement information directly in term ranking functions, e.g., [18], but this kind of research is at its very beginning.

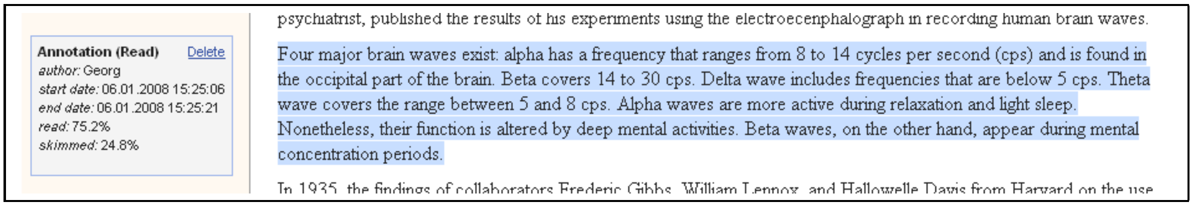
3. GAZE-BASED ANNOTATIONS AS PREREQUISITE FOR IMPLICIT FEEDBACK

Eye trackers generate lots of gaze data every second that has to be aggregated first to be of any use. For our work



Four major brain waves exist: alpha has a frequency that ranges from 8 to 14 cycles per second (cps) and is found in the occipital part of the brain. Beta covers 14 to 30 cps. Delta wave includes frequencies that are below 5 cps. Theta wave covers the range between 5 and 8 cps. Alpha waves are more active during relaxation and light sleep. Nonetheless, their function is altered by deep mental activities. Beta waves, on the other hand, appear during mental concentration periods.

Figure 1: Sequences of fixations (circles) and saccades (lines) over text. Solid lines indicate reading behavior, dashed lines stand for skimming. Eye movements do not match text lines due to eye tracker inaccuracies.



Annotation (Read)
author: Georg
start date: 06.01.2008 15:25:06
end date: 06.01.2008 15:25:21
read: 75.2%
skimmed: 24.8%

psychiatrist, published the results of his experiments using the electroecenphalograph in recording human brain waves.

Four major brain waves exist: alpha has a frequency that ranges from 8 to 14 cycles per second (cps) and is found in the occipital part of the brain. Beta covers 14 to 30 cps. Delta wave includes frequencies that are below 5 cps. Theta wave covers the range between 5 and 8 cps. Alpha waves are more active during relaxation and light sleep. Nonetheless, their function is altered by deep mental activities. Beta waves, on the other hand, appear during mental concentration periods.

In 1935, the findings of collaborators Frederic Gibbs, William Lennox, and Hallowelle Davis from Harvard on the use

Figure 2: Gaze-annotated paragraph of a Wiki document.

we implemented and applied an algorithm for reading and skimming detection and stored this information as document meta-data. Due to space constraints, we just roughly sketch the process leading to gaze-annotated documents. Our technique is described in detail in [5, 6].

During reading, eye movements are composed of fixations and saccades. A fixation is a time interval of around 200ms where the eyes are focusing on one point. Saccades are jumps from one fixation to the next. During reading and skimming the eyes gaze over the text in a very characteristic manner (compare Figure 1). Our algorithm can detect such characteristic movements and additionally differentiates between reading and skimming behavior based on reading speed (i.e., saccade lengths).

Since the gaze data provided by an eye tracker is not always perfectly accurate, we apply techniques for optical character recognition (OCR)[1] to match the eyes' scan paths with the actual text rows. In that way, the read or skimmed text parts of a document can be determined precisely. We use the sematic Wiki *Kaukolu*[2] to store such meta-data for the documents. Figure 2 shows a screenshot of a gaze-annotated paragraph of a Wiki document.

4. EXTRACTING TERMS FOR QUERY EXPANSION

In our study, we capture the precise query context by analyzing at what document parts the user looked immediately before issuing the query. Therefore, we implemented three different methods for extracting query expansion terms based on gaze-based feedback, and one baseline extraction method not considering such feedback. They are all based on very simple term frequency and inverse document frequency (TF×IDF) scores of the document terms. The four variants described in the following are compared against each other in our evaluation.

- The *Baseline* method simply uses TF×IDF on the entire document and extracts the highest scoring terms.

- The *Gaze-Filter* method applies the score calculation of the baseline method (TF×IDF) only on gaze-annotated document parts. So, it just ignores all document parts without a gaze-annotation.

- The *Gaze-Length-Filter* method is an extension of the Gaze-Filter method. It ignores all not gaze-annotated document parts and calculates an *interest* score for every viewed term t as follows:

  $$interest(t) = \frac{LA(t)}{LA(t) + SA(t)} \qquad (1)$$

  $SA(t)$ is the number of gaze-annotations shorter than 230 characters the term t appears in. $LA(t)$ is the number of longer gaze-annotations containing t. The interest value for a term t is then multiplied by its TF×IDF value.

  This heuristic takes a length of 230 characters for the differentiation between long and short annotations since we think that shorter text parts rarely convey sophisticated ideas and concepts to the reader. The heuristic assumes that a person reading a part of a text shorter than 230 characters (i.e., it is a way of scanning) is not interested in the contents of this part. Therefore, it assumes that terms also contained in short viewed text parts do not characterize the current interest of the user very well and gives them a lower interest value.

- The *Reading-Speed* method extends the Gaze-Filter in a different way. As mentioned in the previous section, our reading detection algorithm differentiates between reading and skimming based on reading speed. For every gaze annotation a the percentage of read $r(a)$ and skimmed text $s(a)$ is stored (compare Figure 2 left). For every term t, a reading score $r(t)$ is calculated under consideration of all annotations A_t containing t:

  $$r(t) = \frac{1}{|A_t|} \sum_{a \in A_t} r(a) \qquad (2)$$

  The reading score for a term t is then multiplied by its TF×IDF value also used by the Gaze-Filter.

[1] We utilize the open-source tool OCRopus available at http://code.google.com/p/ocropus/.

[2] Kaukolu, an open-source semantic Wiki available at http://kaukoluwiki.opendfki.de/

This heuristic assumes that more thoroughly read text parts (and therefore their terms) are more likely to be of interest to the user than cursorily viewed parts.

For all heuristics using gaze-based annotations the calculation of the term frequency function (TF) within TF×IDF is modified. It does not return the number of occurrences in the entire document but only in the gaze-annotated parts. The function IDF considers all Wikipedia.de articles as document corpus which we will further used in our experiment.

5. EXPERIMENTAL SETUP

The ultimate goal of considering implicit feedback for retrieval is to enhance the quality of the search results.

The aim of our study is to determine whether gaze-based feedback on its own is already sufficient for enhancing the quality of retrieval by query expansion and reranking. In addition, we explore which of the three methods exploiting gaze-based annotations works best. In the following, we describe our study first from the participants' and then from the system's perspective.

5.1 The Participants' Task

We designed an experiment where the participants had, first, to read text in front of a Tobii 1750 desk-mounted eye tracker, second, to formulate search queries, and third, to give explicit relevance ratings for the search results.

The participants were told to put themselves in the position of a journalist writing articles for a newspaper. The journalist normally gets emails from his or her manager stating about what topics he or she should write an article for the next issue. For our experiment, we created two emails, one about the topic "animal perception" and one about "political aspects of the Manhattan Project".

Besides a very short topic description (one sentence), an email contained several attached documents that were more or less related to the topic. The participants were told that the attached articles were quickly selected by the manager, but should only help them getting started in reading up on the topic. For the topic about perception, four German Wikipedia[3] articles about different species (i.e, cats, sharks, dogs, bats) were attached, each containing some paragraphs about perceptual organs of that species. The articles had a length of about 2000 to 7000 words. The email about the Manhattan Project had four attached biographies about researchers involved (i.e., Oppenheimer, Teller, Bethe, Fermi). The biographies had a length of about 1000 to 5000 words and contained some parts about how the researchers behaved during and after the project. However, the information was more widely distributed in the articles than for the perception topic, i.e., relevant information was not only in one specific part of each document.

The participants were told that they had around 10 to 15 minutes time to go through the four articles for one topic. Realistically, this did not allow for thorough reading of the complete texts. Before starting to read, the eye tracker was calibrated.

After reading the attachments concerning one topic the participants had to perform own searches to get further, more detailed information. Therefore, they had to pose 3 different queries for each topic to the Lucene[4]-based search engine DynaQ[5]. We used the German Wikipedia as document corpus for searching containing ca. 700 000 articles.

- The first query should be used to find more material on the proposed *main topic* of the article to write, i.e., material about perception or the Manhattan Project, respectively.
- The second query was about a *subtopic* mentioned in the attachments. For the topic perception, the participants should find more information about different kinds of eyes as organs for visual perception. For the topic Manhattan Project, they should find material about the Oppenheimer controversy.
- The third query should return material about a *related topic* not included in any of the attached articles. They should find material about perception organs for the earth's magnetic field or the effect of nuclear weapons in the cold war, respectively.

The topics for the queries were told orally to the participants. They were free in formulating the queries.

After issuing each query, the participants had to give explicit relevance feedback for the first 20 result entries. We applied a 6-point feedback scale ranging from "+++ perfect" to "– – – absolute nonsense". They were allowed to open the result document to take a quick look but were told that each rating should not take them more than approx. 15 to 20 seconds on average. However, this was no hard constraint. In fact, the participants sometimes took more than one minute for determining the relevance of a document. The relevance ratings were used to measure the quality of the four different query expansion variants (see next section).

After finishing the complete process for the first topic, the participants had to repeat it for the second topic. In general, we think that this experiment scenario without the explicit feedback in the end would be quite realistic and that it is structurally similar to a lot of search processes in reality.

5.2 Behind the Curtains

Since the aim of our study was to determine the effect of gaze-based feedback on the subdocument level, the eye tracking data was used to create gaze-based annotations of the read and skimmed parts of the attachments on-the-fly (compare Figure 2). All attachments were presented by the gaze-annotation-enabled Wiki system Kaukolu. The annotations were not visible to the participants.

After a participant looked through all attachments concerning one topic, the system automatically computed the query expansion terms for all four extraction variants. This resulted in one query for each of the four term extraction variants described in Section 4. The queries included the top 50 highest scoring terms. Since the Lucene framework allows for weighted query terms, they were weighted by their acquired scores. In that way, each query consisted of the original user-given terms and the 50 terms extracted by the appropriate variant. Weights have also been added to the user-given terms so that 60% of the query weight was put on the generated terms and 40% on the user-given ones.

Only the four attached articles for the currently regarded writing topic were considered as context for the computation of the expansion terms and their weights, i.e., the four

[3] http://de.wikipedia.org/

[4] http://lucene.apache.org/

[5] http://dynaq.opendfki.de/

biographies for queries concerning the Manhattan Project, and the four articles about the different species for queries concerning perception.

Having generated four queries based on the extraction variants, four separate query processes were started for each user query in parallel (compare Figure 3). First, the result set has been determined by the original, non-expanded user query. Documents being used as email attachments and therefore being read by the participants were excluded from the result set. Second, this result set has been re-ranked using the expanded queries. This procedure finally resulted in four separate variant-specific rankings of the search results for each user query. As core ranking function, BM25 ([24] equation 11) has been applied in all cases.

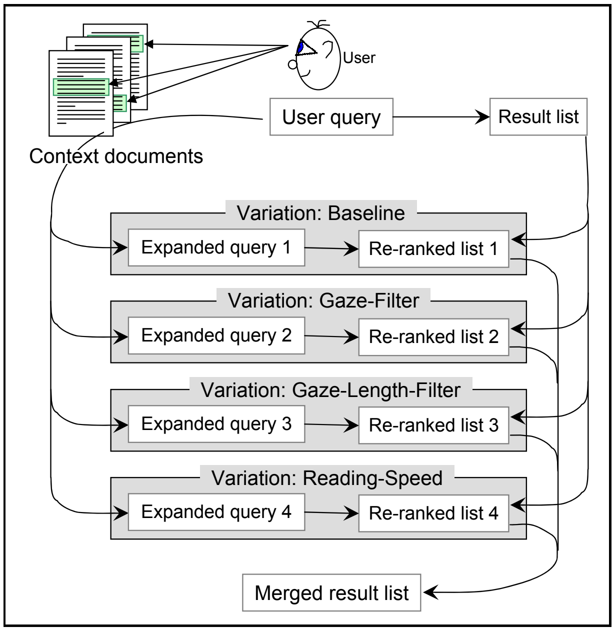


Figure 3: Experimental Setup: Expanding the user query in 4 variants and merging the results.

Because the participants should only see one result list, the four different rankings were merged into one. Merging was accomplished in a balanced way by always selecting that remaining result item from one of the four result lists that had the highest absolute ranking position. Figure 4 gives an example of how the result entries of the four result lists correspond to the entries of the merged list. In that way, if a participant provides explicit relevance ratings for the first 20 results of the merged list, it is guaranteed that there exist ratings for the first 5 consecutive result entries of each of the four lists (because $5 \cdot 4 = 20$ in case of no intersection). The explicit relevance ratings given by the users for the first 20 results of the merged list can then be ascribed to the entries of the four list variants.

5.3 Evaluation Measures

To compare the retrieval quality of the four variants with each other and to provide the possibility to compare the effect of our methods to others, we computed the following commonly used measures:

- **Precision at K**: The value of $Prec@K(q)$ is computed as the fraction of relevant documents within the top K results for a query q. Since this measure needs a binary classification in relevant and irrelevant documents, we interprete all positive labels of our rating scale (+++, ++, +) as relevant and the rest (– – –, – –, –) as irrelevant. The position of the relevant documents within the top K results is not considered by this measure.

merged list position	position in variant 1	position in variant 2	position in variant 3	position in variant 4
1	1	-	1	1
2	-	1	2	2
3	2	6	5	-
4	-	2	23	-
5	3	-	-	-
...				

Figure 4: Composition of a merged result list. E.g., the 2nd result of the merged list occurs in variant 2 on position 1 and in variants 3 and 4 on position 2.

- **MAP**: Mean Average Precision returns a single value for each variant and is computed as follows:

$$MAP(Q) = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{K} \sum_{k=1}^{K} Prec@k(q) \qquad (3)$$

  Q is a set of queries, K is typically chosen as $K = 10$ or $K = 5$. MAP takes the position of the results in the ranking into account but is based on binary relevance classification.

- **DCG at K**: The Discounted Cumulative Gain [13] explicitly takes the ranking of the first K results into account. It does not require binary relevance ratings and therefore rewards highly relevant results more than less relevant results. It is computed as:

$$DCG@K(q) = \sum_{j=1}^{K} (2^{r(j)} - 1)/log(1 + j) \qquad (4)$$

  Here, $r(j)$ returns an integer for the relevance rating given to the result at position j for the query q. For computing this measure, we aggregated all negative ratings because it is not important that a bad result is ranked before an even worse result. Therefore, the function r returned 0 for the ratings "– – –" or "– –" or "–", 1 for "+", 2 for "++", and 3 for "+++". The base of the logarithm was 2. We did not normalize this measure (i.e., we did not apply NDCG).[6]

6. RESULTS

The experiment has been conducted by 21 participants, all being university graduate or undergraduate students. Their attention has been drawn by notices on the university's bulletin boards and they were paid 10 Euros for about 60 to

[6] We only have ratings for the first 7 consecutive entries per result list. If we normalized the DCG measure so that the best possible ranking of the first 7 results would have a value of 1, then the measure would not account for the general quality of the ranking any more. It would merely consider the order of the entries. E.g., the ranking <1, 1, 0> would get a higher NDCG score than <3, 2, 3>, because the latter ranking is not in the best order possible (i.e., <3, 3, 2>). However, the overall quality of the latter ranking is far better. Non-normalized DCG does not show this behavior.

80 minutes of work. Overall, 111 user queries were issued and 2220 explicit relevance ratings for the merged result list have been provided. Five participants only completed one of the two task topics due to own time constraints. 15 participants started with the topic Manhattan Project, 6 with the topic perception. The general distribution of the explicit relevance ratings is given in Figure 5. The distribution is a bit skewed, because Wikipedia.de as document corpus does not contain enough (i.e., 20) relevant documents for every issued query.

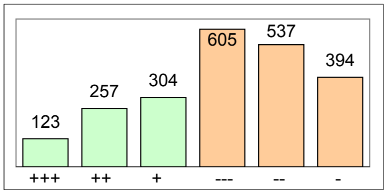


Figure 5: Distribution of the relevance ratings.

6.1 General View

We first determined the three measures described in Section 5.3 by computing their means *over all* 111 queries. For calculating the measures we only selected values for K up to $K = 7$. As stated previously, if a participant provides relevance ratings for the first 20 results of the merged result list, it is guaranteed that there exist ratings for the first 5 consecutive result entries of each of the four result list variants. Because the four result list variants were not completely disjoint, we found that we had consecutive ratings for the first 7 entries of each result list variant. However, for $K = 8$ there were a number of result list variants that did not get enough consecutive ratings.

Figures 6 and 7 depict Precision and DCG for the four variants. Table 1 shows the absolute MAP scores.

A matrix comparing all variants with each other is provided in Table 2. Here the meaning of the abbreviations for the term extraction variants is B = Baseline, GF = Gaze-Filter, GLF = Gaze-Length-Filter, and RS = Reading-Speed. One and two asterisks indicate significance ($p < 0.05$) and high significance ($p < 0.01$), respectively. An asterisk in brackets means minimal significance ($p < 0.1$). Significance was determined by applying the two-tailed, paired t-test.

All measures consistently show a powerful improvement of all gaze-based variants compared to the baseline. The Reading-Speed variant is a bit less effective than the Gaze-Filter and Gaze-Length-Filter variants. The Gaze-Length-Filter might be slightly better than the Gaze-Filter variant, but those results are not significant.

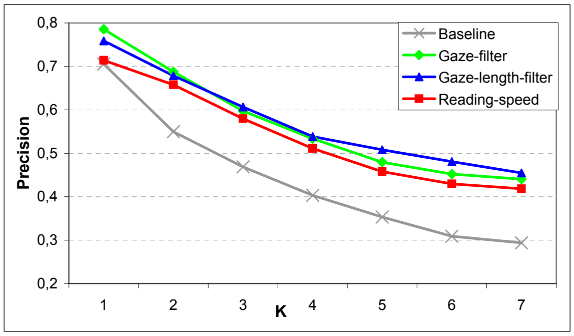


Figure 6: Precision at K for all variants.

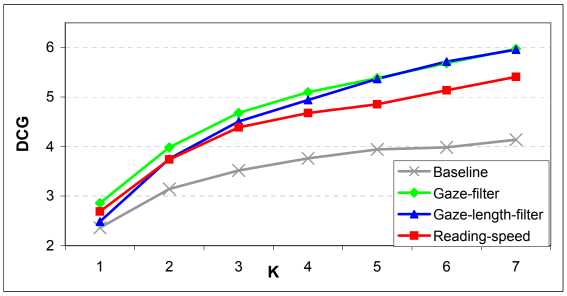


Figure 7: DCG at K for all variants.

Table 1: Mean Average Precision for all variants.

Variant	MAP
Baseline (B)	0.466
Gaze-Filter (GF)	0.557
Gaze-Length-Filter (GLF)	0.559
Reading-Speed (RS)	0.531

Table 2: Matrix for comparing the gains in MAP between all variants.

↓ better than →	B	GF	GLF	RS
GF	30.2% **	/	-	6.8% (*)
GLF	31.7% **	1.2%	/	8.1% *
RS	21.9% **	-	-	/

So, in general, incorporating gaze-based feedback on the subdocument level yields much better results than only incorporating feedback on the document level.

6.2 More Detailed Analysis

We also wanted to know how the effect of gaze-based feedback is influenced by query type and document structure. Therefore, we split up the queries in separate groups:

- To analyze the document structure as influential factor we split up the queries in those belonging to the topic “Manhattan Project” (→ label *weak structure*) and those belonging to “animal perception” (→ label *strong structure*). As mentioned previously, the documents read by the participants before issuing queries about the appropriate topic were structurally different. For the Manhattan Project topic the read documents were more weakly structured so that the relevant information was much more diffuse and widely distributed in the documents than for the perception topic. Therefore, the participants had to look at more different places in the documents and one can expect that gaze-based feedback is of less use here.

- To analyze the query type as influential factor we divided the queries in those being *main topic*, *subtopic*, and *related topic* queries. (We got those query types due to the design of our study, see Section 5.1).

Due to space constraints, we only provide Mean Average Precision for the different query groups.

Document structure. The MAP scores and two comparison matrices for the query groups concerning document structure are reported in Tables 3, 4, and 5. The same abbreviations are used as for Table 2.

Table 3: MAP subdivided by document structure.

Variant	MAP (*weak structure*)	MAP (*strong structure*)
B	0.511	0.362
GF	0.505	0.637
GLF	0.539	0.616
RS	0.495	0.574

Table 4: Matrix for comparing the gains in MAP between all variants for the case *weak structure*.

↓ better than →	**B**	**GF**	**GLF**	**RS**
B	/	1.0%	-	3.2%
GF	-	/	-	2.1%
GLF	5.6% $^{(*)}$	6.7% *	/	8.9% $^{(*)}$

Table 5: Matrix for comparing the gains in MAP between all variants for the case *strong structure*.

↓ better than →	**B**	**GF**	**GLF**	**RS**
GF	75.8% **	/	3.5%	11.1% *
GLF	69.9% **	-	/	7.3%
RS	58.3% **	-	-	/

In the case that strongly structured documents have been read before submitting queries (i.e., topic animal perception), roughly the same trend as in the overall analysis can be recognized but with an effect that is multiple times stronger. However, the very high MAP gains have to be considered with caution since the absolute MAP score for the baseline is relatively low, i.e. 0.362.

For the weakly structured documents, the incorporation of gaze-based feedback does not seem to improve the ranking much. But what can be stated is that no gaze-based variant is significantly worse than the baseline.

Query type. MAP scores for the different query types are provided in Table 6. Due to space constraints we spare the comparison matrices for each query type and only indicate significance of the differences of the gaze-based variants compared to the appropriate baseline. Asterisks are used as above.

The MAP scores generally and consistently decrease from main topic over subtopic to related topic queries. One reason for this is that by our study design the information needs for the subtopic and especially for related topic queries were relatively narrow and the used document corpus Wikipedia.de did only contain a small number of relevant documents.

However the MAP score for the baseline decreases much more compared to the gradients of the Gaze-Filter and Gaze-Length-Filter variants (i.e., a difference of 0.25 in MAP from main to related topic compared to 0.17 and 0.18 for GF and GLF). A reason for this might be as follows: Considering context for query expansion and reranking imprisons the user in some way in this context. Metaphorically speaking, the more the user's information need moves away from the center of the context, the less relevant are the results. The baseline becomes much more worse, because it extracts a context whose center does not match the center of the user's current true concern. E.g., considering the topic "animal perception" in our scenario, the baseline's center of the extracted context would be represented by terms used to describe animal *species*. However, this is not the center of the user's current true concern, i.e., animal *perception*.

Table 6: MAP subdivided by query type.

Variant	MAP (*main topic*)	MAP (*subtopic*)	MAP (*related topic*)
B	0.543	0.460	0.297
GF	0.667 **	0.531 *	0.502 **
GLF	0.665 **	0.565 **	0.489 **
RS	0.631 *	0.569 **	0.393 *

Consequently, a related topic query should find documents containing parts about the topic currently *being of interest* to the user (i.e., the center of the user's true concern) and not necessarily containing parts that have not been of interest to the user before (as the baseline does it by also considering the not viewed parts of the documents). Additionally the returned documents for a related topic query should contain parts about the currently focused related topic expressed by user-given query terms that might not occur in any previously viewed document. In principal, the gaze-based variants work like that. This could be the reason why their MAP scores do not decrease with the same slope as the baseline's score (especially when regarding the variants GF and GLF).

7. DISCUSSION AND CONCLUSION

In our experiment, we applied eye tracking as new data source for attention-based feedback on the subdocument level and examined its effect on context-based query expansion and reranking. The results of our experiment show that reading behavior detection on its own provides useful information when applying it as implicit feedback source for query expansion and reranking. Moreover, considering additional information like reading speed and coherence does not seem to have high impact.

However, to prove the effect of gaze-based feedback we only used one task type and two different task topics in our study. This is a limitation dictated by the cost of the study, which already took more than 20 hours of eye tracking time. At this point, there is room for further studies to examine for what task types gaze-based feedback works best.

Compared to other methods for relevance feedback on the subdocument level (e.g., highlightings as implicit feedback [11], explicit passage-based feedback [27], maybe also display time [15]), gaze-based feedback seems to be sufficiently precise, is available in sufficient quantity, and does not induce any additional cognitive load on the user's side. Of course nowadays, eye trackers are very expensive. However, if their development proceeds further like in the last years, we expect the price to decrease substantially.

In our study, we only used very basic methods for term extraction (basically TF×IDF) and (re-)ranking (basically BM25) and we had a relatively small document corpus to search on (i.e., Wikipedia.de with approx. 700 000 documents). Therefore, the absolute measured retrieval quality is generally not high. However, since we applied the same basic methods and used the same corpus for all examined variants, our results are not less expressive.

Generally, there is much room for more advanced research in this area, e.g., combining gaze-based feedback on the subdocument level with other methods, more accurately interpreting such gaze-based feedback, or using this kind of feedback for retrieval issues other than term extraction and query expansion.

8. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable comments. This work was supported by the German Federal Ministry of Education, Science, Research and Technology (bmb+f), (Grant 01 IW F01, Project Mymory: Situated Documents in Personal Information Spaces).

9. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *SIGIR '06*, pages 19–26, 2006. ACM.

[2] E. Agichtein, E. Brill, S. Dumais, and R. Ragno. Learning user interaction models for predicting web search result preferences. In *SIGIR '06*, pages 3–10, 2006. ACM.

[3] P. Brooks, K. Y. Phang, R. Bradley, D. Oard, R. White, and F. Guimbretière. Measuring the utility of gaze detection for task modeling: A preliminary study. In *International Conference on Intelligent User Interfaces (IUI'06), Workshop on Intelligent User Interfaces for Intelligence Analysis*, Sydney, 2006.

[4] G. Buscher. Attention-based information retrieval. In *SIGIR '07*, pages 918–918, 2007. ACM.

[5] G. Buscher, A. Dengel, and L. van Elst. Eye movements as implicit relevance feedback. In *CHI '08 extended abstracts on Human factors in computing systems*, pages 2991–2996, 2008. ACM.

[6] G. Buscher, A. Dengel, L. van Elst, and F. Mittag. Generating and using gaze-based document annotations. In *CHI '08 extended abstracts on Human factors in comp. syst.*, pages 3045–3050, 2008. ACM.

[7] P. A. Chirita, C. S. Firan, and W. Nejdl. Personalized query expansion for the web. In *SIGIR '07*, pages 7–14, 2007. ACM.

[8] E. Cutrell and Z. Guan. What are you looking for?: an eye-tracking study of information usage in web search. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 407–416, 2007. ACM.

[9] A. Dengel. Six thousand words about multi-perspective personal document management. In *Proc. IEEE-EDM: IEEE Internat. Workshop on the Electronic Document Management in an Enterprise Comp. Env., Key Note Paper*, pages 1–10, 2006.

[10] 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.

[11] G. Golovchinsky, M. N. Price, and B. N. Schilit. From reading to retrieval: freeform ink annotations as queries. In *SIGIR '99*, pages 19–25, 1999. ACM.

[12] L. A. Granka, T. Joachims, and G. Gay. Eye-tracking analysis of user behavior in www search. In *SIGIR '04*, pages 478–479, 2004. ACM.

[13] K. Järvelin and J. Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In *SIGIR '00*, pages 41–48, 2000. ACM.

[14] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR '05*, pages 154–161, 2005.

[15] D. Kelly and N. J. Belkin. Display time as implicit feedback: understanding task effects. In *SIGIR '04*, pages 377–384, 2004. ACM.

[16] D. Kelly and J. Teevan. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum*, 37(2):18–28, 2003.

[17] P. P. Maglio, R. Barrett, C. S. Campbell, and T. Selker. Suitor: An attentive information system. In *IUI-2000: International Conference on Intelligent User Interfaces*, pages 169–176, 2000.

[18] T. Miller and S. Agne. Attention-based information retrieval using eye tracker data. In *K-CAP '05: Proceedings of the 3rd international conference on Knowledge capture*, pages 209–210, 2005. ACM.

[19] F. Qiu and J. Cho. Automatic identification of user interest for personalized search. In *WWW '06*, pages 727–736, 2006. ACM.

[20] F. Radlinski and T. Joachims. Query chains: learning to rank from implicit feedback. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 239–248, 2005. ACM.

[21] J. Rocchio. *The SMART Retrieval System: Experiments in Automatic Document Processing*, chapter Relevance feedback in information retrieval, pages 313–323. Prentice Hall, 1971.

[22] J. Salojärvi, K. Puolamäki, and S. Kaski. Implicit relevance feedback from eye movements. In *ICANN'05*, pages 513–518, 2005.

[23] X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 824–831, 2005. ACM.

[24] K. Spärck-Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: development and status. Technical Report 446, University of Cambridge Computer Laboratory, 1998.

[25] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW '04*, pages 675–684, 2004. ACM.

[26] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *SIGIR '05*, pages 449–456, 2005. ACM.

[27] K. Yang, K. L. Maglaughlin, and G. B. Newby. Passage feedback with IRIS. In *Information Processing and Management*, 37, 3, pages 521–541, 2001.

[28] S. Yu, D. Cai, J.-R. Wen, and W.-Y. Ma. Improving pseudo-relevance feedback in web information retrieval using web page segmentation. In *WWW '03*, pages 11–18, 2003. ACM.