Information Retrieval, 3, 189–216, 2000

Information Retrieval can Cope with Many Errors

ELKE MITTENDORF elke.mittendorf@systor.com
Systor AG, CH-8048 Zürich, Switzerland

PETER SCHÄUBLE schauble@eurospider.ch
Eurospider Information Technology AG, CH-8006 Zürich, Switzerland

Received August 30, 1999; Revised January 7, 2000; Accepted January 19, 2000

Abstract. The retrieval of documents that originate from digitized and OCR-converted paper documents is an important task for modern retrieval systems. The problems that OCR errors cause for the retrieval process have been subject to research for several years now. We approach the problem from a theoretical point of view and model OCR conversion as a random experiment. Our theoretical results, which are supported by experiments, show clearly that information retrieval can cope even with many errors. It is, however, important that the documents are not too short and that recognition errors are distributed appropriately among words and documents. These results disclose that an expensive manual or automatic post-processing of OCR-converted documents usually does not make sense, but that scanning and OCR must be performed in an appropriate way and with care.

Keywords: probabilistic modelling, retrieval effectiveness, optical character recognition, data corruption

1. Introduction

There has been in recent years a growing need for the conversion of large paper and audio archives into electronic form so that these archives may be made accessible through electronic retrieval systems. Current generation optical character recognition and automatic speech recognition systems, the core technologies of this conversion process, are still prone to many recognition errors however. It is therefore important to establish the extent to which errors (*data corruption*) introduced in the conversion of paper and audio archives affect the performance of electronic retrieval systems used in accessing these archives.

The well-known redundancy in texts and the ability of modern information-retrieval (IR) systems to cope with uncertainty (e.g., Fuhr 1992) let us hope for a certain robustness of retrieval systems against errors. This robustness is indeed one of the early results of experiments on corrupted data in information retrieval (Smith and Stanfill 1988, Glavitsch et al. 1994, Schäuble and Glavitsch 1994, Croft et al. 1993). However, there are a lot of questions left open by experiments performed on corrupted data, e.g.: Why is information retrieval, to a surprisingly high degree, robust against data corruption? Under what circumstances is retrieval effectiveness decreased? What steps can be taken to improve retrieval on corrupted data? In this paper, we shall concentrate on answering these questions for the retrieval on data that contains errors because it was produced by optical character recognition (OCR).

There have been several attempts to assess the influence of data corruption on retrieval by a process of experimentation. These experiments have been expensive to develop and

have often produced non-intuitive results; results which have not always been successfully interpreted by the researchers involved. Information retrieval experiments without data corruption are difficult to interpret on their own; combines with data-corruption effects it is even more difficult to assess whether an observed pattern is incidental or represents a general pattern. In contrast to these expensive experiments, difficult to understand, we have chosen a new approach—a theoretical description of data corruption as a random experiment.

Though this work is based on a very theoretical analysis presented in Mittendorf and Schäuble (1996) and Mittendorf (1998), we focus here on the practical implications of this theory on projects that scan and convert document collections by OCR to make them searchable. For this reason we refrain from proving theorems, rather make the theorems plausible and refer to our earlier work. Practical implications are e.g.:

- The scanning of documents still demands a considerable amount of manual work and, thus, has a great potential for saving money at the cost of scanning quality. It is, however, the wrong step for saving money. The result of our work is that it is extremely important that all documents are digitised with considerable care and good quality. No OCR system and no intelligent retrieval system can compensate for what may be lost here.
- Fortunately, there is a potential to save money in the conversion process from pictures to text. It is a waste of money to aim at an error-less automatic recognition or even to manually type the documents. Good retrieval systems (based on feature-frequency weighting, inverse document-frequency weighting, and document-length normalisation other than cosine normalisation) are extremely robust against recognition errors.
- Optimisation criteria for OCR systems such as the minimal number of character errors or word errors per page do not optimise retrieval effectiveness on OCR-converted documents. The smallest degradation of retrieval effectiveness can be achieved if the distribution of errors among different words, different fonts and paper qualities and thus among different documents is as close to an equal distribution as possible.

This paper is structured as follows. In Section 2 we sketch the random model and the main result of the theoretical analysis, i.e., the *main theorem on robustness of retrieval ranking*. Section 4 analyses the model, derives statements about the behavior of collection statistics, such as feature frequencies, and compares the statements with test collections. Section 5, finally, derives answers to the questions concerning the influence of data corruption on information retrieval from the theory. Consequences of this theoretical analysis for practical digitisation projects are concluded in Section 6.

2. A probabilistic model for data corruption

Typically, relevance ranking in information retrieval consists of an *indexing* step and a *retrieval function*. The indexing step identifies the (*indexing*) *features* φ_i—e.g., Porter-reduced non-stopwords—within a document or a query. The set of all features is denoted by Φ.

For describing data corruption by a probability model we have to deal with sets of probability spaces on three levels of complexity: the feature level, the document level, and

the document collection level. The probability distributions on feature level that are of interest are the distributions for given token y_k being recognized as a certain feature φ_i, $P_{y_k}(\varphi_i) = P(x(y_k) = \varphi_i)$. Such a distribution depends on the feature φ_i that the token represents and the document in which the token occurs (e.g., on the page quality, old and yellowed paper or new and white paper), and may be it depends on the position of the token within a document (e.g., if the scanner has the habit of distorting in particular the bottom paragraph of a page).

Indexing represents a documents as a sequence of features. Thus, for our model it makes sense to describe the probability space on document level Ω_{d_j} as the product space of the corruption of its features. We assume that the features that represent d_j and $X(d_j)$. If for the document $d_j = \langle y_0, \ldots, yl(d_j) - 1\rangle$, $y_i \in \Phi$, the experiment $X(d_j)$ can be described as consisting of the *independent* experiments on feature level $X(d_j) = \langle x(y_0), \ldots, x(yl(d_j) - 1)\rangle$ then P_{d_j} is the product distribution of the distributions of features that occur in d_j, i.e.,

$$P_{d_j} = P_{y_0} \cdots P_{yl(d_j)-1}.$$

This description implicitly assumes that features do not disappear, are not merged or split, and do not emerge from, e.g., a stopword. We discuss some violations of this assumption below.

To understand the influence of data corruption on document ranking the probability space must describe all possible corruptions of a complete document collection D, that is e.g., all possible results of digitization and OCR conversion of a paper archive with a given scanner and an appropriate OCR device. The random process of taking one possible corruption of D is denoted by $X(D)$. Similarly to the product space for the document corruption, we define the probability space on D, Ω_D as the product space of the corruption of its documents $D := \{d_0, \ldots, d_{n-1}\}$, i.e., $\Omega_D = \Omega_{d_0} \times \cdots \times \Omega_{d_{n-1}}$. A random experiment of taking one possible corruption of D is denoted by $X(D)$, the random experiment of taking one possible corruption of a document $d_j \in D$ is denoted by $X(d_j)$. This probability model assumes that documents do not vanish, are not split, and do not magically appear from nowhere. These assumptions are realistic if the digitization is performed with appropriate care. That this care is essential for a digitization project seems to be obvious and we shall emphasise this later on.

In contrast to the corrupted documents and collections there are the *perfect documents* and *perfect collections*—which are the result of a process that converts the images or recordings perfectly, e.g., the manual typing of the documents under the assumption that the typing is performed without errors. We need perfect objects mainly as abstract concepts with which we compare the corrupted objects.

An elementary probability of the probability distribution on feature level is the *recognition probability* of a certain feature φ_i within a certain document d_j, i.e., $p_r(\varphi_i, d_j)$. If a token y_k within the document is an instance on the feature φ_i, then $p_r(\varphi_i, d_j)$ is the probability that an occurrence of φ_i in the document d_j is recognized as an instance of φ_i. Note that we assume dependence of the feature itself and of the document in which it occurs, but independence of the particular position of the token. This granularity is sufficient since we shall investigate the ranking of documents, it may not be sufficient if we try to understand, e.g., passage retrieval.

Important random functions on the document level for the description of documents are the following

– The *noisy feature frequencies*

$$\begin{aligned} &\mathrm{nff}(\varphi_i, d_j) : \Phi^* \to \mathbb{N}, \\ &\mathrm{nff}(\varphi_i, d_j) := \mathrm{ff}(\varphi_i, X(d_j)), \end{aligned} \tag{1}$$

where Φ^* denotes the set of all possible strings over Φ. Note that the notation may be misleading. For a given feature φ_i and for a given document d_j the noisy feature frequency is a random function on the probability space $\Omega = \Phi^*$ and not a function on the set of features and the set of documents.

– The *noisy document length*,

$$\begin{aligned} &\mathrm{nl}(d_j) : \Phi^* \to \mathbb{N}, \\ &\mathrm{nl}(d_j) := \mathrm{l}(X(d_j)), \end{aligned} \tag{2}$$

measured e.g., in the number of tokens or the number of different features within a document. Note that the well-known cosine length (Salton 1994) is not a random variable on document level, if the feature weighting is based on idf(φ) weighting.

Important random functions on the documents collection level are the following functions:

– The *noisy document frequency*,

$$\begin{aligned} &\mathrm{ndf}(\varphi_i) : \Omega_D \to \mathbb{N}, \\ &\mathrm{ndf}(\varphi_i) := |\{d_j \in D \mid \mathrm{ff}(\varphi_i, X(d_j)) > 0\}|, \end{aligned} \tag{3}$$

and the *noisy inverse document frequency*,

$$\begin{aligned} &\mathrm{nidf}(\varphi_i) : \Omega_D \to \mathbb{R}, \\ &\mathrm{nidf}(\varphi_i) := 1 - \frac{\log(1 + \mathrm{ndf}(\varphi_i))}{\log(1 + n)}, \end{aligned} \tag{4}$$

where $n := |D|$ is the number of documents in the collection D. (Note that we assume $|D| = |X(D)|$.)

– Based on a given formula for determining a retrieval status value (RSV), there is the *noisy retrieval status value*,

$$\begin{aligned} &\mathrm{nRSV}(q, d_j) : \Omega_D \to \mathbb{R}, \\ &\mathrm{nRSV}(q, d_j) := \mathrm{RSV}(q, X(d_j)). \end{aligned} \tag{5}$$

Other interesting random functions on collection level are e.g., the noisy list, which describes behavior of the ranking of documents after corruption, or the noisy average precision, which describes the behavior of the performance measure average precision after corruption.

Careful digitization: That documents do not disappear during scanning and OCR conversion requires a careful digitization process. OCR errors *corrupt* the information, which is not desirable but information retrieval can cope with it. Careless scanning is, however responsible for the *loss* of information. Since scanning is a part of the digitization that incurs high cost, projects are inclined to save money with this step.

The loss of documents, however, incurs costs as well because the information need of users cannot then be satisfied completely. Performing the capturing of documents with appropriate care, such that the loss of documents is minimized, also incurs costs. The expenses caused by an unsatisfied information need is recurring however, whereas the expense of capturing is a one-time cost.

We illustrate the importance of careful digitization: Let x be the probability that an arbitrary document d_i is lost. Further, let k be the number of documents that are relevant to a given query. Without loss of generality assume that $d_0, \ldots, d_{k-1}$ are relevant. If the event of losing a given document and the event of being relevant are independent then the probability that at least one relevant document is lost is

$$\begin{aligned} &P(\text{at least one relevant document is lost}) \\ &\quad = P(d_0 \text{ is lost} \vee d_1 \text{ is lost} \vee \ldots \vee d_{k-1} \text{ is lost}) \\ &\quad = 1 - P(d_0 \text{ is not lost} \wedge \ldots \wedge d_{k-1} \text{ is not lost}) \\ &\quad = 1 - (1-x)^k. \end{aligned} \tag{6}$$

If, for example, only 1 document out of 1000 documents is lost then $x = 0.001$ and if there are 100 relevant documents then

$$P(\text{at least one relevant document is lost}) = 0.095. \tag{7}$$

In a scenario of searches where users need "everything about a certain topic" (a typical example are searches for patents) this equation means that there is almost a 10% chance that a user's information need cannot be satisfied fully. This might be bearable if a user needs something about a topic, but not if he or she needs everything about a topic. Of course, in cases where it is obvious that users usually query for "something about a topic" the quality constraints for scanning may be relaxed.

Consider another scenario: Assume that a retrieval system has to serve about 100 *known-item searches* per day, i.e., the user knows that there is a document in the collection and he or she wants to find exactly this document. Assume that the 100 known-item searches ask for 100 different documents. Then the probability that at least one of the relevant documents (known items) is lost is, as in (7), 0.095, this number means that with almost a 10% chance the system is not able to answer all searches per day. For example, in a patent information system it can cause a lot of trouble if the patent that was searched for is present but the system cannot find it. In this case the loss of 0.1% of the documents is too expensive.

3. Overtaking probabilities and the main theorem on robustness of retrieval ranking

In information retrieval the recognition probabilities are only an intermediate piece of information on the way to knowing how seriously rankings are permutated by data corruption.

In particular for a highly ranked document we want to know how many lower ranked documents have *overtaken* them in the corrupted ranking. This point-of-view motivates the definition of *overtaking probabilities*.

Definition 1. The *overtaking probability* for the documents d_j and d_k with $\mathrm{RSV}(q, d_j) > \mathrm{RSV}(q, d_k)$ is the probability

$$P(\mathrm{nRSV}(q, d_k) > \mathrm{nRSV}(q, d_j)). \tag{8}$$

In this section we investigate the systematic effects of data corruption on the ranking by inspecting the expected value $\mathrm{E}(\mathrm{nRSV}(Q, d_j))$ and the variance $\mathrm{Var}(\mathrm{nRSV}(q, d_j))$.

An important concept in this context is, whether or not we can expect the documents to be ranked in the perfect order:

Definition 2. If for the pair (d_j, d_k) with $\mathrm{RSV}(q, d_j) > \mathrm{RSV}(q, d_k)$

$$\mathrm{E}(\mathrm{nRSV}(q, d_j)) > \mathrm{E}(\mathrm{nRSV}(q, d_k)) \tag{9}$$

holds we say the *quality condition is met,* if

$$\mathrm{E}(\mathrm{nRSV}(q, d_j)) < \mathrm{E}(\mathrm{nRSV}(q, d_k))$$

we say the *quality condition is violated.*

We denote the difference between a pair of retrieval status values by

$$\delta_{jk}(q) := \mathrm{RSV}(q, d_j) - \mathrm{RSV}(q, d_k). \tag{10}$$

and we denote the difference between a pair of expected noisy retrieval status values by

$$\Delta_{jk}(q) := \mathrm{E}(\mathrm{nRSV}(q, d_j)) - \mathrm{E}(\mathrm{nRSV}(q, d_k)). \tag{11}$$

We are now able to describe the behavior of the overtaking probabilities in more detail by bounding them. We report the bounds in the following theorem. The theorem is the basic theoretical result for our analysis of the influence of data corruption; so that we consider this theorem to be the *main theorem on robustness of retrieval ranking.*

Theorem 1. *Assume that* $\mathrm{nRSV}(q, d_j)$ *and* $\mathrm{nRSV}(q, d_k)$ *are stochastically independent. Let* $\mathrm{RSV}(q, d_j) > \mathrm{RSV}(q, d_k)$. *If the quality condition is met then*

$$P(\mathrm{nRSV}(q, d_k) > \mathrm{nRSV}(q, d_j)) \leq \frac{\mathrm{Var}(\mathrm{nRSV}(q, d_j)) + \mathrm{Var}(\mathrm{nRSV}(q, d_k))}{(\Delta_{jk}(q))^2}. \tag{12}$$

If the quality condition is violated then

$$P(\text{nRSV}(q, d_k) > \text{nRSV}(q, d_j)) \geq 1 - \frac{\text{Var}(\text{nRSV}(q, d_j)) + \text{Var}(\text{nRSV}(q, d_k))}{(\Delta_{jk}(q))^2}. \tag{13}$$

The proof, basically an application of Chebychev's inequality, is reported in the Appendix.

Interpretation of the main theorem: The statement of the main theorem is very abstract. We interpret the theorem to further illustrate the abstract statement of Theorem 1. Obviously, if the terms on the right side of Inequality (12) and (13) are greater than 1 or smaller than 0 then Theorem 1 is useless. For other cases it leads to some interesting preliminary conclusions about data corruption effects on information retrieval.

- Generally speaking, the smaller the variances of the noisy retrieval status values, the higher is the probability that the ranking of the documents will be in accordance with the expected noisy retrieval status values E(nRSV(q, d_j)).
- The two inequalities (12) and (13) indicate that, for a sufficiently small variance of noisy retrieval status values, the more pairs violate the quality condition, the more is the ranking corrupted.

In other words, the essence of Theorem 1 is as follows: The ranking is less corrupted if the quality condition is often met and the variance is small.

4. The behavior of collection statistics in theory and in reality

Modern retrieval functions are constructed from statistics that are computed on documents, such as feature frequencies and document length, and from statistics that are computed on collections, such as inverse document frequencies and average document length. In this section we investigate behavior and values of recognition probabilities on our test collection and also the behavior of the noisy feature frequencies. For the analysis, theoretically and empirically, of other statistics, such as the noisy logarithmic feature frequencies, noisy document lengths, noisy inverse document frequencies, etc. we refer to (Mittendorf 1998).

4.1. The test collection

***Documents*:** We used the collection from the TREC-5 confusion track (Voorhees and Kantor 1997), which is provided by the National Institute of Standards and Technologies (NIST) as a test collection. It consists of three parallel collections; one perfect collection $D = \text{FR94}$ and two different corrupted versions of FR94, $X(D) = \text{D5}$ and $X(D) = \text{D20}$. Note that in our model these two collections are samples generated by two different probability distributions. The collection FR94 consists of 250 Mbyte of data from the 1994 Federal Register, that are in total 55340 documents of varying length. All documents are written in American English.

It is described in Voorhees and Kantor (1997) how the test collection has been produced. The D5 collection is estimated to have a character recognition probability of 95% (a "degrade" of 5%, therefore the name D5). The D20 collection is estimated to have a character recognition probability of 80% (a "degrade"of 20%, D20). The document boundaries have been preserved. There were documents lost by the capturing process but we have excluded them from the test collection.

The production of the test collection at NIST compromised between cost and authenticity. There was no scanning involved, thus the test collection is realistic only to a certain extent. Typical errors due to scanning (crinkled or dirty paper, paper skew, or even the loss of documents) cannot be found in the corrupted collections. However, the errors in D5 and D20 are more realistic than the errors in collections with simulated data corruption; most importantly, the size of the test collection is more realistic than all parallel collections that have been produced by manual corrections in previous experiments, which have generally consisted of no more than a few thousand documents.

***Queries and relevance information*:** The NIST has provided 49 known-item searches on the FR94 collection, which are called CF1, ..., CF50 (CF29 has been removed from the query collection). For each query the set of relevant documents consists of exactly one document, $|D_{\text{rel}}(q)| = 1$. Known-item searches require a very specialized kind of queries, since they are highly precision-oriented searches. The queries in the test collection thus represent only a small subset of all possible queries.

***Indexing*:** We chose a standard indexing procedure for English texts for feature extraction (Ballerini et al. 1997). A feature class consists of all words of at least three consecutive alphanumeric characters, from which the first character must be a letter (a–z or A–Z), that are reduced to the same stem by Porter's algorithm (Porter 1980) and that are not an element of SMART's stoplist (Salton 1971). The feature extraction is case insensitive. We refer to these features as *Porter-reduced non-stopwords*.

The size of the queries, measured in the number of indexing features, is rather small. The smallest queries consist of only one feature; the longest query consists of 14 features. Altogether the 49 queries are made up of 250 features, which is an average of 5.1 features per query.

4.2. Recognition probabilities

In this section we estimate collection-wide recognition probabilities on the OCR test collection on a feature basis. We consider both the corruption of D5 and the corruption of D20 as random experiments. Note that from the model point of view in this experiments we have only one realization for each of the two random experiments. Note that we denote a realization in the same way as we denote the random variables themselves. We assume that the recognition probabilities are constant across documents, i.e., $p_r(\varphi_i, d_j) = p_r(\varphi_i), d_j \in D$, so as to have enough samples for a reliable estimation.

We count the collection frequency for each of the 250 features φ_i that occur in the query set of the test collection:

$$\mathrm{cf}(\varphi_i) := \sum_{d_j \in D} \mathrm{ff}(\varphi_i, d_j),$$

we similarly count the noisy collection frequency,

$$\mathrm{ncf}(\varphi_i) := \sum_{d_j \in D} \mathrm{nff}(\varphi_i, d_j) = \sum_{X(d_j) \in X(D)} \mathrm{ff}(\varphi_i, X(d_j)),$$

for $\chi(D) = \mathrm{D5}$ and for $\chi(D) = \mathrm{D20}$. The ratio of $\mathrm{cf}(\varphi_i)$ and $\mathrm{ncf}(\varphi_i)$ is unfortunately affected not only by the recognition probability but also by the false alarms. If we subtract the noisy feature frequency $\mathrm{nff}(\varphi_i, d_j)$ of those documents d_j where $\mathrm{ff}(\varphi_i, d_j) = 0$ from the noisy collection frequency, we eliminate a great portion of all false alarms. Thus, a rough estimate of the recognition probability is given by:

$$\hat{p}_r(\varphi_i) := \frac{\mathrm{ncf}(\varphi_i) - \sum_{d_j, \varphi_i \notin d_j} \mathrm{nff}(\varphi_i, d_j)}{\mathrm{cf}(\varphi_i)}. \tag{14}$$

Note that for features φ_i with a small document frequency the estimate in (14) is not a good estimate.

We eliminate features with zero collection frequency from the experiment and also ignore the feature "late", which happens to have extremely many false alarms.

Results and their interpretation: The values for the estimate $\hat{p}_r(\varphi_i)$ on D5 and on D20 are summarised in the histograms in figure 1. The histogram on the left and on the right show the estimates of recognition probabilities on D5 and on D20, respectively.

The collection D5 is supposed to have a recognition probability of characters of 95%. If the recognition errors are distributed equally across characters we expect word recognition probabilities of approximately $0.95^8 = 0.66$ (words with 8 characters) to $0.95^4 = 0.81$ (words with 4 characters). This is not represented in the the histogram for D5, which reflects a very skewed distribution. More than half of the query features have a recognition probability of 0.97 and higher, other features are never recognised.

The D20 collection contains many more features that are never or almost never recognized. Besides the unproportionally high number of features that are never recognized the collection D20 represents an equal error distribution of most of the features: D20 is supposed to have a character recognition probability of 80%. For words of 4 to 8 characters we can expect to have word recognition probabilities between $0.8^8 = 0.17$ and $0.8^4 = 0.41$. There are indeed many features with recognition probabilities between 0.2 and 0.4.

A closer look at the misrecognized features reveals the reason for the unequal distribution: *All 33* features out of the 250 features in the test set that have recognition probability $\hat{p}_r(\varphi_i) < 0.1$ on D5 contain the letter *j*, the letter *z*, the letter *k*, or they occur exclusively in capitalised form, such as the feature "indian". In this experiment the OCR device systematically fails for these three letters and capital characters. This behavior of an OCR device

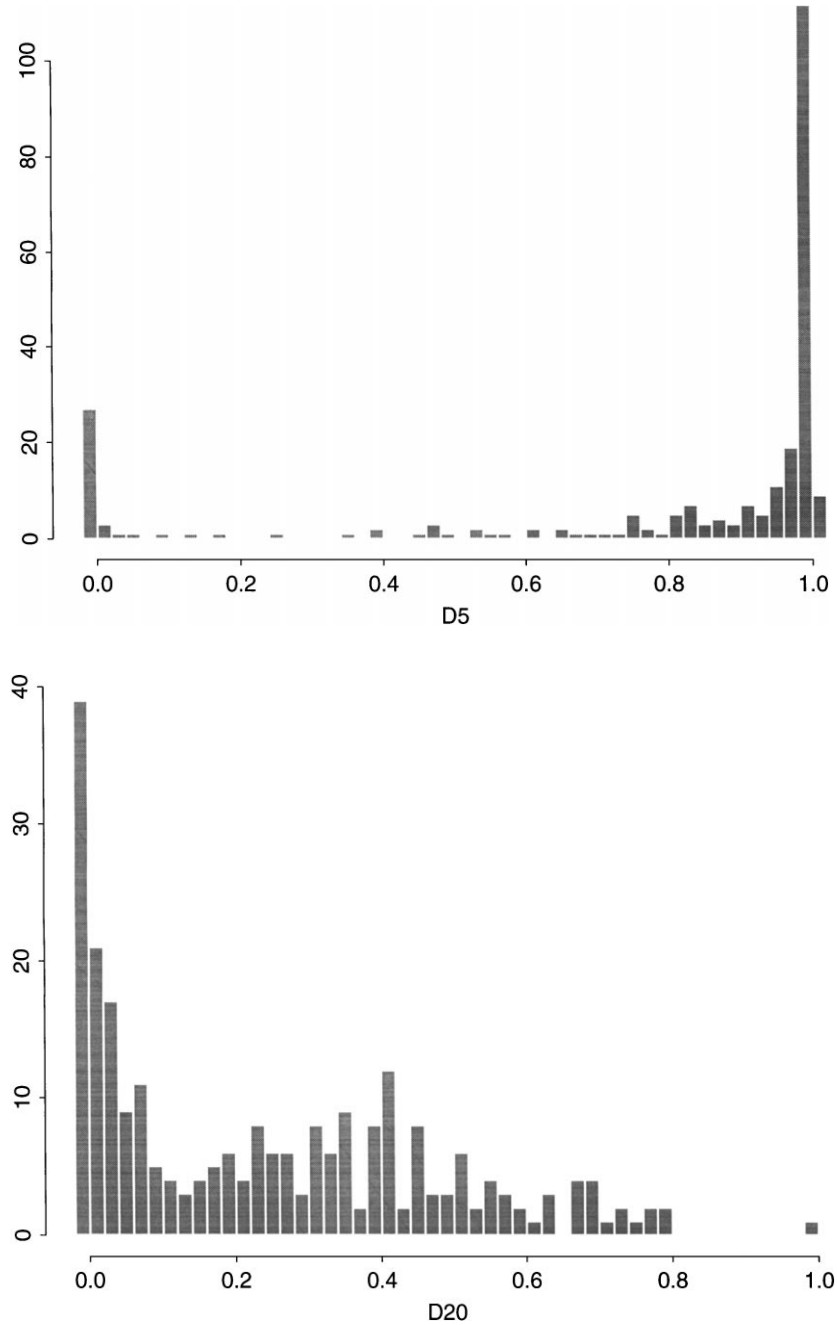


Figure 1. Histogram of recognition probabilities on D5 and on D20. The *y*-axis represents the feature count. Note that the two histograms are scaled differently.

is not uncommon and can be explained by the way these devices are trained. We shall see in Section 5.4 that the skewed distribution of recognition probabilities causes problems for retrieval and we briefly discuss the reason for the skew in Section 5.5.3.

4.3. Noisy feature frequencies

Feature frequencies are a very important component of effective retrieval functions. In general, the higher the feature frequencies of query features in a document, the higher is the probability that the document is relevant to the query (Robertson and Walker 1994). We show (theoretically and empirically) that high feature frequencies cause high noisy feature frequencies; they are indeed related proportionally with random deviations. Thus, we claim that noisy feature frequencies as well as feature frequencies can be used as reliable estimators for relevance probabilities.

We have provided a theoretical description of the random variables $\mathrm{nff}(\varphi_i, d_j), \varphi_i \in \Phi, d_j \in D$ in Mittendorf (1998) by rewriting noisy feature frequencies as a sum of Bernoulli variables of feature recognitions. We report here the result: If the false alarms are negligible then

$$E(\mathrm{nff}(\varphi_i, d_j)) = \mathrm{ff}(\varphi_i, d_j)\, p_r(\varphi_i, d_j). \tag{15}$$

and

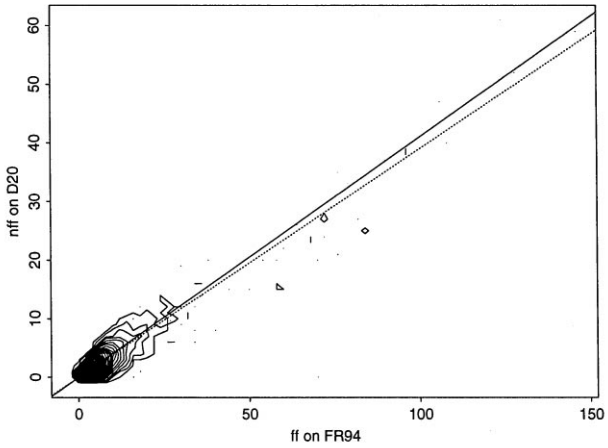
$$\begin{aligned} \mathrm{Var}(\mathrm{nff}(\varphi_i, d_j)) &= \sum_{y_k=\varphi_i} \mathrm{Var}(Y_k) + \sum_{y_k\neq\varphi_i} \mathrm{Var}(Y_k) \\ &= \mathrm{ff}(\varphi_i, d_j)\, p_r(\varphi_i, d_j)(1 - p_r(\varphi_i, d_j)) \end{aligned} \tag{16}$$

These formulas indicate that noisy feature frequencies increase proportionally with feature frequencies. The next section validates the formula on a collection-wide basis.

4.4. The analysis of the behavior of noisy feature frequencies on the test collection

To validate formula (15) we need many samples and thus must rely on features with recognition probabilities that are constant across documents, i.e., $p_r(\varphi_i, d_j) = p_r(\varphi_i)$ for all $d_j \in D$. We chose the feature $\varphi_i =$ "provid" on D20 for the following reasons: It occurs very often in the collection, 76546 times in 28946 documents, which promises reliable estimates. It has an overall recognition probability of $p_r(\varphi_i) = 0.41$. Recall that 0.41 is exactly the recognition probability of an eight-letter word (such as 'provides' or 'provider') if each character has probability 0.8. and thus we can expect that the errors are equally distributed among all occurrences of the feature in the collection. As most other features, the feature "provid" has a negligible false alarm rate. In Mittendorf (1998) two more feature examples have been analysed.

For the given feature $\varphi_i =$ "provid" we produce a set of tuples $(\mathrm{ff}(\varphi_i, d_j), \mathrm{nff}(\varphi_i, d_j))$, with $d_j \in \mathrm{FR94}$ and $X(d_j) \in D20$. A regression analysis ((Stahel 1995, p. 257) or (Venables and Ripley 1994)) is performed on the set of tuples; in particular, a "simple linear analysis



β	se_β	t-value	P-level
0.3918	0.0011	345.5401	0

min(res)	1st quartile	median	3rd quartile	max(res)
−27.43	−0.3918	−0.1345	0.6082	12.41

Number of Samples	σ	R-squared
28949	0.8562	0.8049

Figure 2. Contour plot and regression output for "provid" in D20, contour lines are spaced quadratically.

through the origin" for $\mathrm{E}(\mathrm{nff}(\varphi_i, d_j)) = \beta\ \mathrm{ff}(\varphi_i, d_j)$ is performed, which estimates β by a least square method.

Results: Figure 2 presents a contour plot of the histogram of the tuples $(\mathrm{ff}(\varphi_i, d_j), \mathrm{nff}(\varphi_i, d_j))$ for d_j with $\varphi_i \in d_j$ or $\varphi_i \in X(d_j)$. Note that (for efficiency reasons and reasons of proper presentation) the tuples (0, 0) are not represented in the diagram. The contour lines are spaced quadratically, i.e., (1, 4, 9, . . .).

The graph contains a solid line and a dotted line. The solid line represents the prediction of the theoretical model for the expected feature frequency according to (15). The dotted line is the prediction of a simple linear regression through the origin. The output of the regression is shown the tables in figure 2. In Stahel (1995, p. 261) it is explained in a detailed way how the output of a regression can be interpreted, e.g., the value *R*-squared can be interpreted as the percentage of samples that can be explained by the estimated model.

Interpretation of the results for "provid" on D20: We recognise that for the feature "provid" the model of linearly corrupted feature frequencies and the regression estimate are very close (figure 2). A slope of $\beta = 0.39$ in the regression fit deviates only a little bit from

the model parameter of 0.41. The high t-value and the small estimated standard error for β, se_β, show a high reliability of the estimate. A high percentage of about 80% of the 28949 samples fit the regression model (R-squared).

In summary, the model presented in Section 2 and the theoretically-derived formula (15) fit the OCR test data very well. We have not tried to show whether the formula for the variance (16) is valid or not. The tendency of increasing variance with increasing feature frequency, however, is traceable as shown in Mittendorf (1998).

5. The influence of data corruption on information retrieval effectiveness

5.1. Retrieval on corrupted data is feasible

It has been one of the first results in the field of analysing data-corruption effects on information retrieval that retrieval on corrupted data is feasible. Researchers have even been surprised at how well retrieval works on highly-corrupted data (Smith and Stanfill 1988, Schäuble and Glavitsch 1994, Croft et al. 1993, Glavitsch et al. 1994). There is however a certain difference in the results that emerge from experiments with simulated data corruption and from experiments on genuinely-corrupted data. Whereas in both sets of experiments the main result is that retrieval is in general robust against data corruption, the experiments on genuinely-corrupted data has shown that there are queries, for which the retrieval effectiveness has suffered to a very high degree. However, this section explains the high robustness; possible problems are discussed in the following sections.

Let us introduce a notation for a general retrieval function:

$$\mathrm{RSV}(q, d_j) := \frac{1}{\mathrm{norm}(d_j)} \sum_{\varphi_i \in q} a(\mathrm{ff}(\varphi_i, d_j)) b(\mathrm{ff}(\varphi_i, q)) w(\varphi_i), \tag{17}$$

where a, b, norm, and w are non-negative real functions, a and b must not depend on any collection-wide statistics such as document frequencies. Typically a and b stand for the identity function (linear feature frequency weighting) or for a logarithmic transformation such as $1 + \log(\mathrm{ff}(\varphi_i, d_j))$. The weight w collects all components of a weighting function that depend on collection-wide statistics for the feature φ_i, a typical function w is e.g., $w(\varphi_i) = \mathrm{idf}(\varphi_i)$. The function norm (d_j) stands for the factor that provides document length normalisation. We shall present here a rather informal explanation why data corruption is robust. For a formal proof on some retrieval functions refer to Mittendorf and Schäuble (1996) and Mittendorf (1998).

Let the following assumptions hold:

- The recognition probability is constant across features and documents $p_r(\varphi_i, d_j) = p_r, \varphi_i \in \Phi, d_j \in D$.
- We can approximate $\mathrm{E}(a(\mathrm{nff}(\varphi_i, d_j))) = Ca(\mathrm{ff}(\varphi_i, d_j)), C > 0$. For linear feature-frequency weighting this is a good model as shown in 4.3 and also for logarithmic feature-frequency weighting it is an appropriate model (Mittendorf 1998).
- The normalization is robust against data corruption norm $(X(d_j)) = \mathrm{norm}(d_j)$. We know e.g., that length normalization based on the number of tokens is extremely robust, but

cosine normalization is not, since the cosine length of a corrupted document is dominated by misrecognized low-frequency words and thus high-inverse-document-frequency words. This fact has been explained in Taghva et al. (1994).
- The $b(\mathrm{ff}(\varphi_i, q))$ is robust. In other words, the query is not corrupted.
- $w(\varphi_i)$ is robust. The inverse document frequency weight $w(\varphi_i) = \mathrm{idf}(\varphi_i)$ is not robust, but weights derived from a training collection are robust.

Then

$$\mathrm{E}(\mathrm{nRSV}(q, d_j)) = C\mathrm{RSV}(q, d_j),$$

and thus the quality condition (2) is always met, which means that overtaking probabilities are low (Theorem 1) and the ranking is likely to be robust.

This behavior reveals that it does not make sense to aim at an error-less recognition when digitizing for retrieval. It is more important to control the distribution of errors among documents and words as will be shown in the Section 5.4.

5.2. The influence of the document length

Previous experiments of retrieval on corrupted data, in particular experiments in Croft et al. (1993), have led to the *conjecture* that for long documents retrieval is more robust against data corruption than retrieval on short documents. Together with some intuitive considerations—i.e., long documents are more redundant and thus are less prone to the loss of feature occurrences—the conjecture became a conviction in the field of analyzing data corruption effects on information retrieval. We do not have to rely on intuitive considerations however, we have *proved* the conjecture formally and specified it in Mittendorf (1998) in the *Theorem on Robustness and Document Length*.

Theorem 2. *Assume that there is an infinite sequence of pairs*

$$\left(d_j^{(0)}, d_k^{(0)}\right), \left(d_j^{(1)}, d_k^{(1)}\right), \left(d_j^{(2)}, d_k^{(2)}\right), \ldots$$

which satisfy the following requirements.

1. *The corresponding lengths $l_j^{(h)}$ and $l_k^{(h)}$ and the normalization factors are unbounded, i.e.,*

$$\lim_{h\to\infty} \mathrm{norm}\left(d_j^{(h)}\right) = \infty \quad \textit{and} \quad \lim_{h\to\infty} \mathrm{norm}\left(d_k^{(h)}\right) = \infty. \tag{18}$$

2. *Assume that the quality condition is either met for all pairs in the sequence $(d_j^{(h)}, d_k^{(h)})$, $h \in \mathbb{N}$, or that it is violated for all pairs. Furthermore, assume that there exists a positive lower bound Δ such that for all $\Delta_{jk}^{(h)} := \mathrm{E}(\mathrm{nRSV}(q, d_j^{(h)})) - \mathrm{E}(\mathrm{nRSV}(q, d_k^{(h)}))$*

$$\left|\Delta_{jk}^{(h)}\right| > \Delta. \tag{19}$$

3. *Assume that there exists a* $K > 0$ *such that we have for all* $h \in \mathbb{N}$ *and for all* $\varphi_i \in q$

$$\begin{aligned} \operatorname{Var}\bigl(a\bigl(\operatorname{nff}\bigl(\varphi_i, d_j^{(h)}\bigr)\bigr)\bigr) &\leq K \operatorname{norm}\bigl(d_j^{(h)}\bigr), \\ \operatorname{Var}\bigl(a\bigl(\operatorname{nff}\bigl(\varphi_i, d_k^{(h)}\bigr)\bigr)\bigr) &\leq K \operatorname{norm}\bigl(d_j^{(h)}\bigr). \end{aligned} \tag{20}$$

Then we can conclude that, if for all $(d_j^{(h)}, d_k^{(h)})$, $h \in \mathbb{N}$, *the quality condition is met*,

$$P\bigl(\operatorname{nRSV}\bigl(q, d_k^{(h)}\bigr) > \operatorname{nRSV}\bigl(q, d_j^{(h)}\bigr)\bigr) \to 0 \tag{21}$$

or, if for all $(d_j^{(h)}, d_k^{(h)})$, $h \in \mathbb{N}$, *the quality condition is violated*,

$$P\bigl(\operatorname{nRSV}\bigl(q, d_k^{(h)}\bigr) > \operatorname{nRSV}\bigl(q, d_j^{(h)}\bigr)\bigr) \to 1, \tag{22}$$

as $h \to \infty$.

The *proof* is a direct consequence of Theorem 1 and is similar to the proof of the law of large numbers. It is worked out in detail in Mittendorf (1998).

This lengthy and somewhat formal theorem may be deterring, but the advantages of the formalism are that the theorem reveals interesting facts more than just the influence of the document length.

Interpretation:

- The main implication: The longer the documents in a collection the smaller is the probability that in the presence of errors the rankings are permutated (robustness).
- If documents become longer because they talk about many different topics (scope hypothesis (Robertson and Walker 1994)) the difference between the RSV of different documents vanishes, in particular if the RSV are normalised by the document length. Thus, Condition (19) is violated and we cannot imply the robustness. However, if documents become longer because the authors are very verbose (verbosity hypothesis (Robertson and Walker 1994)) upon one and the same topic the same features recur and with an appropriate document normalization Condition (19) holds.
- We have elaborated in Mittendorf (1998) that the conditions of the theorem hold for well-known retrieval functions such as proposed by Singhal et al. (1996) and Robertson and Walker (1994). It is important that the document length normalization is based on the number of features or the number of tokens in a document.
- The theorem does not state that any collection of long documents is superior to any collection of short documents in terms of robustness against data corruption. It only indicates a general tendency.

5.3. How can robustness be improved if your documents are short?

The Theorem on Robustness and Document Length suggests to digitize a complete document and not only its summary if both are available. On the one hand the theorem indicates

that wordy documents are easier to digitise than concise documents, unfortunately. On the other hand, the theorem also helps us to improve digitization of short and concise documents for retrieval purposes. We shall state some tricks to improve robustness. These tricks are in a sense ways of decreasing the variance of the noisy retrieval status values.

1. A common approach in retrieval of corrupted data is the use of n-gram features instead of feature classes containing words such as porter-reduced non-stopwords.

   A token of a word-based feature is only recognized correctly if all of its characters (except for some stemmer-dependent special cases) and thus all the constituting n-grams are recognized correctly. Thus features based on n-grams have usually higher recognition rates than word-based features. In addition to the positive effect of lower recognition probabilities there are more n-gram tokens in a document than tokens of word-class features. Thus, the documents are longer if length is measured in terms number of tokens and thus we can expect the variance $\mathrm{Var}(\mathrm{nRSV}(q, d_j))$ to be smaller.

   However, n-gram retrieval has a lower retrieval effectiveness than standard word-based retrieval on perfect data (Cavnar 1992, Teufel 1989, Wechsler and Schäuble 1995). Thus there is a trade-off which has to be considered. In practice n-gram retrieval often pays off for highly corrupted data.
2. A similar method is to use n-gram features *and* word-class features simultaneously, which is reported to be successful for experiments with highly-corrupted data in Harding et al. (1997). On perfect data however combining n-gram features with word-based features results in less effective retrieval than word-based features only.
3. One can improve the estimates of the feature frequencies by applying several independent recognition devices. The recognition quality can be improved by combining different recognition methods, e.g., (Garzotto 1994, Jäger 1996, Jones et al. 1996). However, as mentioned before, a slight improvement of recognition probabilities hardly decreases the ranking corruption.

   We rather plead for using the different devices to improve the estimate of the RSV. In terms of our probability model different devices yield different independent realizations of the random function $X(d_j)$. Note that in case of paper documents, for example once the documents are scanned, it is not too much of a problem to run a batch system that employs several OCR systems. This approach is, however, very memory intensive. Note that the OCR devices must be independent, that means, in particular, they must be different.
4. Probabilistic matching methods such as described in Mittendorf et al. (1995) have been developed to get better estimates of the perfect feature frequencies: Not only the exact occurrences of features (as in our probability model) are included in the computation of noisy feature frequencies but similar strings in the document are also taken into account in order to improve the estimate of the noisy feature frequency.

   Theorem 1 shows us that a small variance $\mathrm{Var}(\mathrm{nff}(\varphi_i, d_j))$ must be a design criterion for a good probabilistic-matching method that estimates feature frequencies, in particular if we have to deal with short documents and small recognition probabilities because both properties lead to high variance, Theorem 2 and Eq. (16). In these cases the additional effort of an improved estimate is worthwhile.

Unfortunately the effect of the document length cannot be validated with the test collection because ranking corruption is dominated by other factors (as we will show in Section 5.4). This effect has, however, been illustrated before by others (Croft 1993).

5.4. The influence of recognition probabilities that vary from document to document or from feature to feature

This section explains two factors that degrade information retrieval performance significantly. We first explain the reason for the degradation of retrieval performance and then show on our test collection that indeed these two factors are the major reasons for retrieval degradation under data corruption.

5.4.1. Variation from document to document. Let us first consider a case where the recognition probabilities of all features within one and the same document are constant, $p_r(\varphi_i, d_j) = p_r(d_j)$ for all $\varphi_i \in d_j$. Consider a general RSV-function (17). As in Section 5.1 assume uncorrupted normalization (such as pivoted normalization based on the number of tokens), uncorrupted feature weighting $w(\varphi_i)$ (such as inverse document frequency weighting based on document frequencies from an uncorrupted training collection), and a linear feature-frequency weighting. Then we have

$$\begin{aligned}
\mathrm{E}(\mathrm{nRSV}(q, d_j)) &= \frac{1}{\mathrm{norm}(d_j)} \sum_{\varphi_i \in q} \mathrm{E}(\mathrm{nff}(\varphi_i, d_j))\ \mathrm{ff}(\varphi_i, q) w(\varphi_i) \\
&= \frac{1}{\mathrm{norm}(d_j)} \sum_{\varphi_i \in q} p_r(d_j)\ \mathrm{ff}(\varphi_i, d_j))\ \mathrm{ff}(\varphi_i, q) w(\varphi_i) \\
&= p_r(d_j) \mathrm{RSV}(q, d_j). \qquad (23)
\end{aligned}$$

Note that a similar behavior of the corruption of $\mathrm{E}(\mathrm{nRSV}(q, d_j))$ can be observed for other weighting schemes, e.g., for logarithmic feature weighting (Mittendorf 1998).

Example. Figure 3 illustrates some overtaking scenarios. Two number axes are opposed to each other in both of the two pictures in the figure. The left axis in each of the pictures contains the documents ranked according to their $\mathrm{RSV}(q, d_j)$ and on the right axis in each of the pictures the documents are placed according to their $\mathrm{E}(\mathrm{nRSV}(q, d_j))$. The representatives of the same document on the left and right axes are connected with lines. The left picture (a) represents an example for which the recognition probabilities are almost constant across documents, let e.g., $0.7 \leq p_r(d_j) \leq 0.8$ for all documents. In this case the quality condition is always or almost always met; in the picture example the connecting lined do never cross, and thus overtakings are not very likely, besides the moderate recognition probabilities. The right picture (b) represents an example in which the probabilities vary heavily. Let e.g., $p_r(d_j) = 0.1$ for documents out of a certain subset (in the picture example for those document ranked 1st, 4th, and 5th by the perfect ranking) $p_r(d_j) = 0.9$ for all other documents (in the picture example the 2nd, 3rd, 5th, and 7th document on the perfect ranking). For such recognition probabilities the quality condition of a document pair (d_j, d_k)

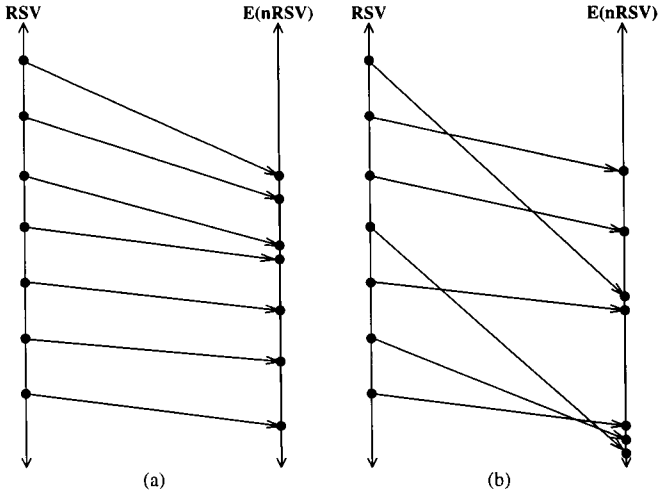


Figure 3. Documents have similar (a) or varying (b) recognition probabilities.

with RSV$(q, d_j) >$ RSV(q, d_k), $p_r(d_j) = 0.1$, and $p_r(d_k) = 0.9$ the quality condition is violated if RSV$(q, d_j) < 9$ RSV(q, d_k), which is usually the case for many such pairs and thus overtakings are likely. In the picture example (b) the quality condition is violated for six document pairs (six lines cross).

In summary, the more the recognition probability vary from document to document the likelier are overtakings.

5.4.2. Experiments on the variation from document to document. The test collection (Section 4.1) incidentally contains examples where the effect of recognition probabilities that vary from document to document can be illustrated. We can find several documents in which almost no features are recognized—neither in the version of the documents of D5 nor in the version of D20. We have collected a few of those documents are in the following set, $S :=$ {FR941230-0-00119, FR941230-0-00120, FR941230-0-00121, FR941202-2-00139, FR0527-2-00135}. All features in the documents in set S have a recognition probability $p_r(d_j)$ of almost zero in both versions, D5 and D20. (The reason is the capitalization of almost all words in those documents and the problems that the OCR device has with upper-case letters, Section 5.5.3.)

Incidentally, the particular documents of set S have been retrieved among the top-ranked documents on the perfect lists for several queries, may be because they contain unproportionally many low-frequency words.

In our experiments, we ranked the documents according to the socalled Lnu.ltn weighting scheme (Singhal et al. 1996) (logarithmic feature frequency weighting in documents and queries, inverse document frequency weighting, pivoted document length normalization). The perfect list L has been computed on $D :=$ FR94 and the noisy list nL has been computed on $X(D) :=$ D5. Instead of noisy inverse document frequencies for the noisy list, the

perfect document frequencies have been used to eliminate the effects due to noisy document frequencies. For each of the 49 queries we then computed two measures of ranking corruption. The squared rank difference of the *top-ten* documents of the perfect list rc_{Sq10} and the overtaking cost rc_{CO}, both are formally defined in Mittendorf (1998). For the measure overtaking cost the overtakings are penalized with a linear function of the rank of the document in the perfect list.

We picture the results in the two graphs in figure 4. The queries have been ordered by their two different ranking-corruption values: by $\mathrm{rc}_{Sq}(q)$ in the left graph and by $\mathrm{rc}_{CO}(q)$ in the right graph. (Note that we had to scale the ranking-corruption values for numerical reasons and thus, the absolute values are not meaningful.)

The x-axis represents the rank with respect to the particular ranking-corruption value and the y-axis represents the the ranking-corruption value itself. Instead of dots, the scatter plot consists of numbers which represent the number of documents out of the set S that appear in the perfect list among the top ten. (By the way, none of the documents out of the set S appears among the top 500 in any of the corrupted lists and non of which is relevant to any query). The query identification (CF1–CF50) is given at the top of the figure.

Obviously the more the highly-corrupted documents from set S are ranked among the top-ten documents in a perfect ranking, the more the ranking itself is corrupted. This relationship can be observed in particular for the measure that is based only on the top-ten documents.

5.5. *Variation from feature to feature*

Now assume the dual scenario: The recognition probability of a given feature φ_i is constant from document to document but may vary from feature to feature. We abbreviate $p_r(\varphi_i) := p_r(\varphi_i, d_j)$.

Again the quality condition is often violated: We assume that the false alarm probabilities can be neglected and we introduce the notation

$$p_{\min}(q) := \min_{\varphi_i \in q} p_r(\varphi_i)$$

and

$$p_{\max}(q) := \max_{\varphi_i \in q} p_r(\varphi_i).$$

Again, we assume that the document independent feature weighting $w(\varphi_i)$ and the normalization factors norm (d_j) are not corrupted and that feature frequencies are weighted linearly: It is

$$\begin{aligned} \mathrm{E}(\mathrm{nRSV}(q, d_j)) &= \frac{1}{\mathrm{norm}(d_j)} \sum_{\varphi_i \in q} \mathrm{ff}(\varphi_i, q)\mathrm{E}(\mathrm{nff}(\varphi_i, d_j))w(\varphi_i) \\ &= \frac{1}{\mathrm{norm}(d_j)} \sum_{\varphi_i \in q} \mathrm{ff}(\varphi_i, q)\, p_r(\varphi_i)\mathrm{ff}(\varphi_i, d_j))w(\varphi_i). \end{aligned} \tag{24}$$

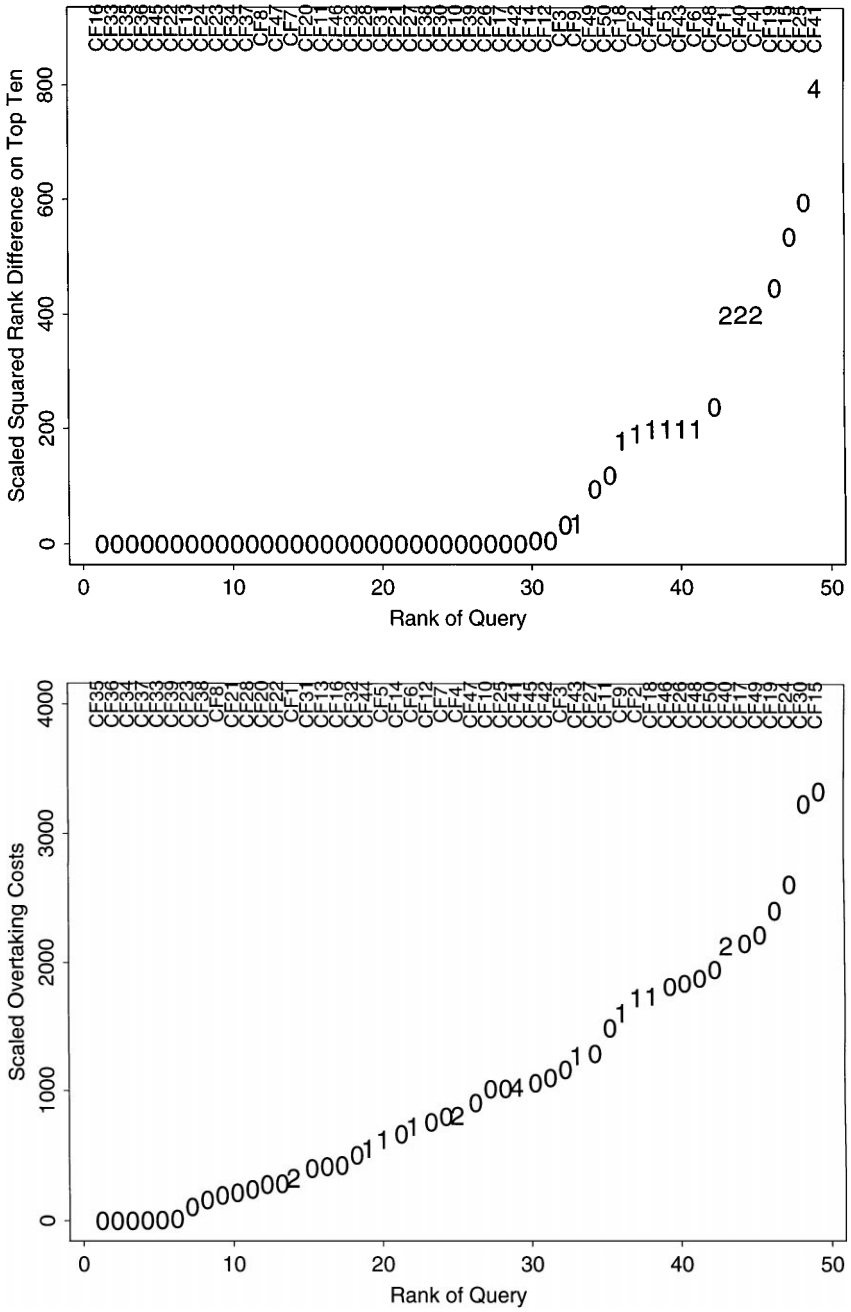


Figure 4. Documents with extremely low recognition probabilities and the influence on ranking corruption.

Then because

$$p_{\min}(q) \leq p_r(\varphi_i) \leq p_{\max}(q),$$

for all $\varphi_i \in q$, the expected noisy retrieval status values are bounded by

$$\begin{aligned} p_{\min}(q)\ \mathrm{RSV}(q, d_j) \leq \mathrm{E}(\mathrm{nRSV}(q, d_j)) \leq p_{\max}(q)\ \mathrm{RSV}(q, d_j), \\ p_{\min}(q) \leq a_j(q) \leq p_{\max}(q). \end{aligned} \tag{25}$$

We have to assume that the bounds $p_{\min}(q)$ and $p_{\max}(q)$ are inclusive because a document may contain only those query features with recognition probabilities $p_r(\varphi_i)$ that are equal to $p_{\min}$, another document may contain only those query features with $p_r(\varphi_i) = p_{\max}$. The behaviour of the expected overtakings resembles the one in figure 3. Thus, again, we have to expect a high ranking corruption. Note that in Mittendorf (1998) we have included other weighting schemes into this analysis.

5.5.1. Experiments on the variation from feature to feature. It is not possible to separate the different effects of document length, varying recognition probabilities across features, and varying probabilities across documents. Thus it is not easy to confirm the theoretical investigations by experiments. In a similar way to Section 5.4.1, we limit the experiments to extreme cases where we have features with recognition probabilities close to zero. We also limit our experiments to the D5 documents, because on D20 the degradation of query features varies heavily for all queries. In particular, for all queries there is at least one feature that has a recognition probability less than or equal to 0.15.

The following queries contain at least one feature with recognition probability on D5 less than or equal to 0.15. The respective features are specified:

CF2 “risk”	CF7 “mark”	CF10 “network”
CF15 “truck”	CF17 “truck”, “jackknife”	CF18 “alaska”
CF21 “amazon”	CF24 “size”	CF25 “project”
CF27 “mhz”	CF30 “alkali”, “character”, “milk”	CF31 “zoo”
CF32 “subject”	CF39 “jellyfish”	CF40 “wicker”
CF41 “jade”	CF42 “duck”	CF43 “jazz”
CF46 “rock”	CF48 “object”	CF49 “smoke”
CF50 “alaska”		

Figure 5 illustrates the influence of extremely varying recognition probabilities on ranking corruption. The experiment and the applied measures for figure 5 are exactly the same as for figure 4. The only difference between the scatter plots in the two figures is that the numbers that represent the positions in the scatter plots indicate the number of features in the query that belong to the set of features with $p_r(\varphi_i) \leq 0.15$.

Results and interpretation: In general, the larger the ranking corruption in terms of the measure $\mathrm{rc}_{Sq10}(q)$ (scatter plot at the top of figure 5) as well as in terms of overtaking cost $\mathrm{rc}_{CO}(q)$ (scatter plot at the bottom) the higher is the number of highly-corrupted features in

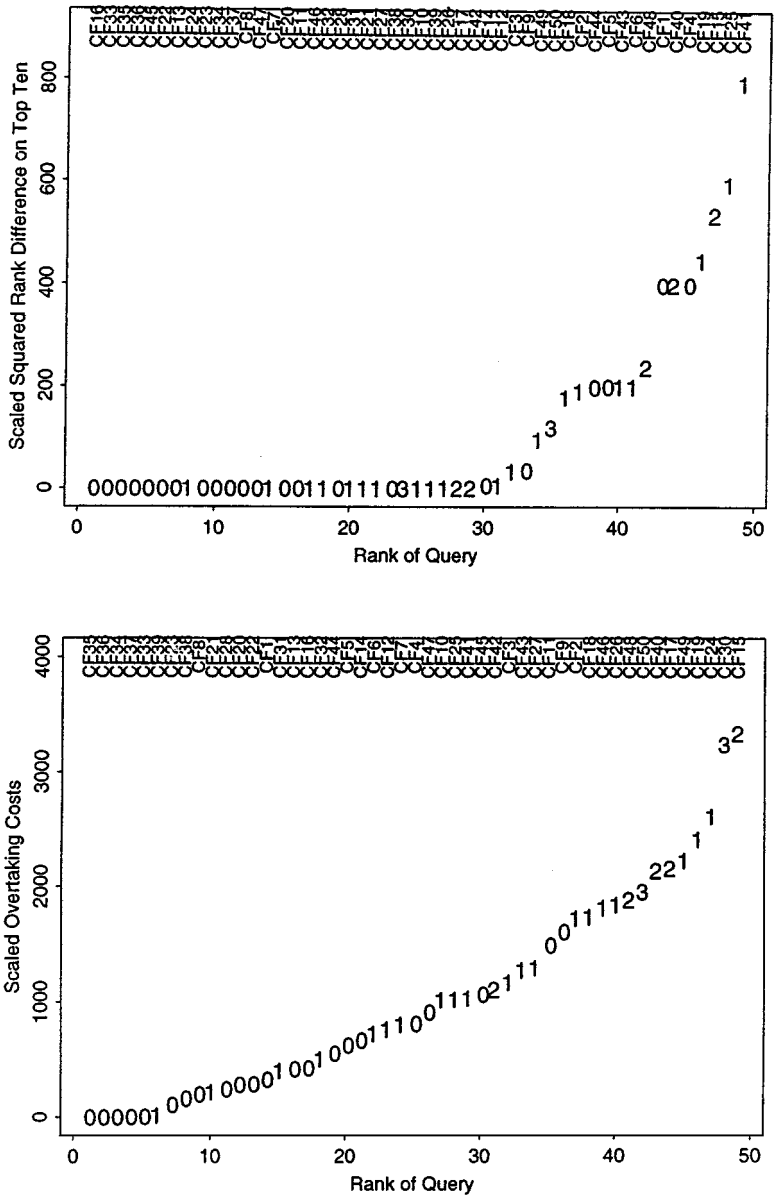


Figure 5. Ranking corruption and number of features with low recognition probabilities.

the scatter plot. Extreme variation of recognition probabilities from feature to feature harm retrieval performance.

A close look at the four scatter plots in figures 4 and 5 reveals that for the collection D5 high ranking corruption is *always due to a high variation in recognition probabilities*, either a variation from document to document or a variation from feature to feature.

5.5.2. Remarks on the use of post-processing systems. Post-processing systems are mainly dictionary based spelling correction systems that aim at improving the recognition probabilities of words by a dictionary look up of similar words. There is a strong belief in some research groups that the automatic post-processing of texts is beneficial in OCR-based text retrieval. This opinion is expressed e.g., in Taghva (1994), although in this research it could not be shown that retrieval is significantly better on post-processed data than on the originally corrupted data. The experimental results showed an increase in precision but a decrease in recall and it left the researchers wondering why the retrieval on the post-processed data is good for some queries but is heavily degraded for other queries. The investigations of Section 5.4 indeed explain the high query variation.

Wiedenhöfer et al. (1995) used an elaborate post-processing to improve the indexing process for scanned German business letters. They mention a higher efficiency of indexing and retrieval because of the smaller number of indexing features, but they mention also that the use of the post-processing system leads to effectiveness problems. They realized that the use of a dictionary boosts the recognition probability of some features and overrides the recognition probability of others. Their effectiveness problems due to post-processing systems coincide with our theoretical result.

A post-processing system might increase the overall recognition probability of characters and words. The biggest problem, however, is that a dictionary is always of finite size. After post-processing, features that are contained in the dictionary have high recognition probabilities and higher false alarm rates, whereas the recognition probabilities of features not contained in the dictionary remain low or even become zero if they have a high string similarity to one of the elements of the dictionary. The recognition probability varies highly from feature to feature and thus implies high ranking corruption. On the other hand, a post-processing system decreases ranking corruption if all query features are elements of the dictionary thereby giving higher recognition probabilities, (15) and (16). This explains the high query variation in retrieval effectiveness found in Taghva (1994).

A fact that intensifies the negative effect of the use of post-processing is that the features that are not contained in a dictionary usually are features with high inverse document frequencies and thus—whenever contained in a query—make important contributions to a good retrieval result. A post-processing system may be useful in a routing or filtering environment where the indexing vocabulary can be limited to the features that occur in the current query (profile) but is not useful for retrieval in general.

5.5.3. A word on the optimization of OCR systems. OCR systems are often trained by maximising the *overall recognition* probability or, equivalently, by minimizing the average number of errors per page.

Definition 3. Let A be the alphabet of all characters in a language $\mathcal{L}$ then the *overall character-recognition probability* is defined as

$$p_{\text{overall}} := \sum_{c \in A} P(c) P(x(c) = c), \tag{26}$$

where $P(c)$ is the probability that the character c occurs in texts that are written in language $\mathcal{L}$.

Looking at this formula, we can see that it is more important to be able to recognize frequent characters such as *e* (12.02%) or *t* (9.14%) than infrequent characters such as *z* (0.09%) or *j* (0.14%). (The numbers in brackets indicate the occurrence probabilities of the characters in the FR94 test collection in percent.)

Consider the following example: In the FR94 collection the occurrence probability of the character *i* is 0.079, whereas the occurrence probability of the character *j* is 0.0014. These two characters are easily confused by many OCR systems.

Assume that 5% of all characters *i* are recognized as *j*. Then p_{overall} is at least $0.079 \times 0.05 = 0.00395$ smaller than for a system that perfectly distinguishes between *i* and *j*. If you replace each *j* by an *i* without regarding how confident or uncertain the system is about its decision then p_{overall} decreases at most by 0.0014 compared with a system that perfectly distinguishes between the two characters. In cases of doubt one must decide against infrequent and in favour of frequent characters to gain an optimal overall recognition probability. In summary, optimisation of overall recognition probabilities is likely to cause an unequal distribution of errors among characters. Optimizing the average number of errors of, equivalently, the overall recognition probability is not optimal for retrieval purposes, where we need an equal distribution of errors among words and documents.

5.6. How can robustness be improved in the presence of varying recognition probabilities

The most important teaching of Section 5.4 is to aim at the avoidance of great variation of recognition probabilities. If possible the variation must be prevented at the time of the recognition process.

1. A first rule is, *do not post-process* the document collection with a dictionary. This rule has the positive side effect that an extensive and time-consuming step in the process of data conversion can be omitted and thus capturing remains cheaper. Pre-processing can be useful for search tasks where the queries are contained in a controlled vocabulary (dictionary), such as a filtering task.
2. *Watch the strategy for optimizing recognition probabilities*: We learned that an OCR system that works well for retrieval purposes is not necessarily optimized with measures that are standard in the evaluation of OCR systems. The same holds for speech recognition systems. However, most commercial OCR systems are trainable after purchase, speech recognition system have to be made application specific anyway. Thus there is a chance to outwit a wrong optimization strategy if the training strategy is revealed. For example, if the overall recognition probability is optimized, the recognition device must be trained on

material that consists of all types of recognizable units and all units should be distributed more or less equally on the material.

If the recognition process can not be influenced or if even after a better optimization there is a great feature variation, then we have to find ways and means to circumvent the problems that are due to varying recognition probabilities.

1. *Long queries or query expansion*: If several features are lost due to very low recognition probabilities then it may be possible to compensate for the loss of these feature by formulating long queries or by using an intelligent query-expansion algorithm. Although long queries or query expansion can compensate for the loss of a few features to a certain extent (Frei and Qui 1993, Xu and Croft 1996, Efthimiadis 1996) the queries must operate only with features that are well recognised by the system, and features good for query expansion are not necessarily recognized well. However, it is for users difficult to ask good, long queries and query expansion as well is neither an easy task nor a computationally cheap task and even after the high expenditure of query expansion the user has still to be satisfied with a suboptimal performance of the system.

   There is another reason why users should be encourage to use long queries: Long queries are beneficial in the presence of false alarms (Sanderson 1994). We have discussed this in detail in Mittendorf (1998).
2. *Probabilistic matching*: The algorithms that proceed from an exact feature matching to a weaker kind of feature matching (e.g., (Mittendorf et al. 1995, Myka and Güntzer 1995)) qualify themselves for the task of optimizing the retrieval results on corrupted data. The knowledge that we gained about the influence of recognition variation gives a clear idea how a good probabilistic matching algorithm must be designed: Equation 24 shows that the matching should be relaxed more for features with low recognition probabilities and still be tight for features with high recognition probabilities.
3. *The noisy inverse document frequency* increases in comparison to the perfect inverse document frequencies whenever a features is badly recognized. This behavior can, to a certain extent, compensate for the negative effects of varying feature-recognition probabilities. It is thus better to count the document frequency on the corrupted collection than on a perfect training collection. Besides, the corrupted document frequency count is beneficial in the presence of false alarms (Mittendorf 1998).
4. The simultaneous use of *several independent OCR devices* to estimate feature frequencies as already indicated to compensate for short documents Section 5.3 can also help to level the feature recognition probabilities. For this purpose the independent devices must be trained differently.

6. Summary

We applied probability theory to understand data corruption effects on information retrieval. The theory has been validated on a test collection. Our analysis yields results that do not only explain experimental results on information retrieval that could so far not be explained, it also implies measures for the proper realization of digitization projects for information retrieval purposes.

We briefly summarize the most important consequences for digitization projects.

- Information retrieval on corrupted data is feasible even with only moderate recognition probabilities. Although it is obvious that the less errors the more robust is the retrieval, it is a waste of money to aim at an error-less recognition (Section 5.1). The scanning process must however be performed with appropriate care (Section 2). and a solid retrieval system must be applied that employs weighted retrieval (Section 5.1).
- One must be careful with the interpretation of experiments that are performed with simulated data corruption, if they do not contain variations of recognition probabilities. Variations are the cause of the most permutations in retrieval rankings (Section 5.4).
- If one has the choice of scanning full documents or only document extracts one should scan the full documents to achieve more robustness against data corruption (Section 5.2).
- One must be careful with the use of post-processing systems (Section 5.5.2) and the application of standard strategies for optimising the recognition because they favor the variation of recognition probabilities (Section 5.5.3).
- Cosine normalization is harmful (Taghva 1994).
- Noisy inverse document frequency is better than inverse document frequencies estimated on training collections, because noisy inverse document frequency compensates for several negative effects of recognition probabilities (Section 5.6).
- There are some promising approaches to improve retrieval on corrupted data that are advisable to employ, such as using several independent OCR devices, probabilistic feature matching, and n-gram based retrieval (Sections 5.3 and 5.6).

Appendix A. Proof of the main theorem (Theorem 1)

For all pairs of documents d_j and queries q with $\mathrm{RSV}(q, d_j) > 0$ there exists an *degradation value* $a_j(q)$

$$\mathrm{E}(\mathrm{nRSV}(q, d_j)) = a_j(q)\ \mathrm{RSV}(q, d_j), \tag{27}$$

with $a_j(q) > 0$. Then

$$\mathrm{nRSV}(q, d_j) = a_j(q)\ \mathrm{RSV}(q, d_j) + s(q, d_j), \tag{28}$$

where $s(q, d_j)$ describes the variance

$$\begin{aligned} \mathrm{E}(s(q, d_j)) &= 0, \\ \mathrm{Var}(s(q, d_j)) &= \mathrm{Var}(\mathrm{nRSV}(q, d_j)). \end{aligned}$$

Using the description of the relation between retrieval status values and noisy retrieval status values (28) and the abbreviations for the differences (10) and (11) we get:

$$\begin{aligned} &P(\mathrm{nRSV}(q, d_k) > \mathrm{nRSV}(q, d_j)) \\ &\quad = P(a_k(q)\ \mathrm{RSV}(q, d_k) + s(q, d_k) > a_j(q)\ \mathrm{RSV}(q, d_j) + s(q, d_j)) \end{aligned}$$

$$
\begin{aligned}
&= P(a_k(q)\ \mathrm{RSV}(q, d_k) + s(q, d_k) > a_j(q)\ \mathrm{RSV}(q, d_k) + a_j(q)\delta_{jk} + s(q, d_j)) \\
&= P(s(q, d_k) - s(q, d_j) > \Delta_{jk}(q)) \\
&\leq P(|s(q, d_k) - s(q, d_j)| > \Delta_{jk}(q)).
\end{aligned}
$$

Chebychev's inequality, the independence of nRSV(q, d_j) and nRSV(q, d_k), and the fact that for two independent variables X and Y, $E(XY) = 0$ (Pythagoras' theorem for stochastic variables) yield

$$
\begin{aligned}
P(\mathrm{nRSV}(q, d_k) > \mathrm{nRSV}(q, d_j)) &\leq \frac{\mathrm{Var}(s(q, d_j) - s(q, d_k))}{(\Delta_{jk}(q))^2} \\
&= \frac{\mathrm{Var}(\mathrm{nRSV}(q, d_j)) + \mathrm{Var}(\mathrm{nRSV}(q, d_k))}{(\Delta_{jk}(q))^2},
\end{aligned}
$$

which proves Inequality (12).

Similarly with Chebychev's inequality we show

$$
P(\mathrm{nRSV}(q, d_j) \geq \mathrm{nRSV}(q, d_k)) \leq \frac{\mathrm{Var}(\mathrm{nRSV}(q, d_j)) + \mathrm{Var}(\mathrm{nRSV}(q, d_k))}{(\Delta_{kj}(q))^2}.
$$

We have $\Delta_{jk}(q) = -\Delta_{kj}(q)$ and thus $(\Delta_{kj}(q))^2 = (\Delta_{jk}(q))^2$, then

$$
\begin{aligned}
P(\mathrm{nRSV}(q, d_k) &> \mathrm{nRSV}(q, d_j)) \\
&= 1 - P(\mathrm{nRSV}(q, d_j) \geq \mathrm{nRSV}(q, d_k)) \\
&\geq 1 - \frac{1}{(\Delta_{jk}(q))^2}(\mathrm{Var}(\mathrm{nRSV}(q, d_j)) + \mathrm{Var}(\mathrm{nRSV}(q, d_k))),
\end{aligned}
$$

which proves Inequality (13).

References

Ballerini J-P, Büchel M, Domenig R, Knaus D, Mateev B, Mittendorf E, Schäuble P, Sheridan P and Wechsler M (1997) SPIDER retrieval system at TREC-5. In: TREC-5 Proceedings.

Cavnar WB (1992) N-gram-based text filtering for TREC-2. In: TREC-2 Proceedings.

Croft WB, Harding S, Taghva K and Borsack J (1993) An evaluation of information retrieval accuracy with simulated OCR output. In: Symposium on Document Analysis and Information Retrieval, pp. 115–126.

Efthimiadis E (1996) Query expansion. Annual Review of Information Science and Technology, 31:121–187.

Frei HP and Qui Y (1993) Effectiveness of weighted retrieval in an operational IR environment. In: Information Retrieval '93. Universitätsverlag Konstanz, pp. 41–45.

Fuhr N (1992) Probabilistic models in information retrieval. The Computer Journal, 35(3):243–255.

Garzotto A (1994) Vollautomatische Erkennung von Schriftzeichen in gedrucktem Schrittgut. PhD Thesis, Universität Zürich.

Glavitsch U, Schäuble P and Wechsler M (1994) Metadata for integrating speech documents in a text retrieval system. SIGMOD RECORD, 23(4):57–63.

Harding SM, Croft WB and Wein C (1997) Probabilistic retrieval OCR degraded text using n-grams. In: Research and Advanced Technology for Digital Libraries, First European Conference, ECDL'97, pp. 345–359.

Jäger Th (1996) OCR and voting shell fulfilling specific text analysis requirements. In: Symposium on Document Analysis and Information Retrieval, pp. 287–302.

Jones GJF, Foote JT, Sparck Jones K and Young SJ (1996) Retrieving spoken documents by combining multiple index sources. In: ACM SIGIR Conference on R&D in Information Retrieval, Zurich, pp. 30–38.

Mittendorf E (1998) Data corruption and information retrieval. PhD Thesis, ETH Zurich, Institute of Computer Systems.

Mittendorf E and Schäuble P (1996) Measuring the effects of data corruption on information retrieval. In: Symposium on Document Analysis and Information Retrieval, pp. 179–189.

Mittendorf E, Schäuble P and Sheridan P (1995) Applying probabilistic term weighting to OCR text in the case of a large alphabetic library catalogue. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 328–335.

Myka A and Güntzer U (1995) Automatic hypertext conversion of paper document collections. In: Adam N, Bhargava B and Yesha Y, Eds., Advances in Digital Libraries—Current Issuses, Springer-Verlag, Berlin, pp. 65–90. Lecture Notes in Computer Science, Vol. 916.

Porter MF (1980) An algorithm for suffix stripping. Program, 14(3):130–137.

Robertson SE and Walker S (1994) Some simple effective approximations of the 2-Poisson model for probabilistic weighted retrieval. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 232–241.

Salton G (1971) The SMART Retrieval System-Experiments in Automatic Document Processing. Prentice Hall, Englewood, Cliffs, New Jersey.

Salton G (1990) Automatic Text Processing. Addison-Wesley, Reading, MA.

Sanderson M (1994) Word sense disambiguation and information retrieval. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 142–151.

Schäuble P and Glavitsch U (1994) Assessing the retrieval effectiveness of a speech retrieval system by simulating recognition errors. In: ARPA Workshop on Human Language Technology (HLT'94), pp. 370–372.

Singhal A, Buckley C and Mitra M (1996) Pivoted document length normalization. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 21–29.

Smith S and Stanfill C (1988) An analysis of the effects of data corruption on text retrieval performance. Thinking Machines Corporation, Cambridge, MA.

Stahel W (1995) Statistische Datenanalyse: Eine Einführung für naturwissenschaftler. Lehrbuch, Angewandte Mathematik. Vieweg, Wiesbaden.

Taghva K, Borsack J and Condit A (1994) Effects of OCR errors on ranking and feedback using the vector space model. Technical Report TR 94-06, University of Nevada, Las Vegas.

Taghva K, Borsack J and Condit A (1994) Results of applying probabilistic IR to OCR text. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 202–211.

Teufel B (1989) Informationsspuren zum numerischen und graphischen Vergleich von reduzierten natürlichsprachlichen Texten. PhD Thesis, Swiss Federal Institute of Technology, VdF-Verlag, Zürich.

Venables WN and Ripley BD (1994) Modern applied statistics with S-plus. Statistics and Computing. Springer-Verlag, New York.

Voorhees E and Kantor P (1997) TREC-5 confusion track. In: TREC-5 Proceedings.

Wechsler M and Schäuble P (1995) Speech retrieval based on automatic indexing. In: Ruthven Ian Ed., Proceedings of the Final Workshop on Multimedia Information Retrieval (MIRO'95), Electronic Workshops in Computing, Springer, Glasgow.

Wiedenhöfer L, Hein H-G and Dengel A (1995) Post-processing of OCR results for automatic indexing. In: Third International Conference on Document Analysis and Recognition, Montreal, August 1995. IEEE Computer Society Press, Silver Spring, MD, pp. 592–597.

Xu J and Croft WB (1996) Query expansion using local and global document analysis. In: ACM SIGIR Conference on R&D in Information Retrieval, pp. 4–11.