

The Effects of Noisy Data on Text Retrieval

Kazem Taghva*, Julie Borsack, Allen Condit, and Srinivas Erva
Information Science Research Institute
University of Nevada, Las Vegas

May 1993

Abstract

We report on the results of our experiments on query evaluation in the presence of noisy data. In particular, an OCR generated database and its corresponding 99.8% correct version are used to process a set of queries to determine the effect the degraded version will have on retrieval. It is shown that with the set of scientific documents we use in our testing, the effect is insignificant. We further improve the result by applying an automatic post processing system designed to correct the kinds of errors generated by recognition devices.

*Email: taghva@cs.unlv.edu.

1 Introduction

As the accuracy of optical character recognition (OCR) devices improves, the practicality of using this generated text directly without human intervention becomes feasible. But even with a 99% character accuracy, which some devices claim, there could be up to 25 misspelled words per page.¹ As was shown in the Licensing Support System (LSS) Prototype, the cost associated with manually correcting these errors is high. From scanning the original document to loading its accurate ASCII text into a text retrieval system, the capture methodology was burdened with tedious, manual steps[11].

Since the objective of these efforts is the retrieval of pertinent information, the question becomes:

For a particular query, how much correction is necessary to retrieve precisely the same set of documents that would be retrieved from a corrected ASCII document set?

We try to answer this question by querying two sets of data: a 99.8% accurate set and a corresponding OCR set. No accuracy rates were calculated for the OCR document set but as stated in [2], a 99% character accuracy can only be attained by commercial OCR products if “a printed document is a fixed-pitch, typed original or a clean copy, in a simple paragraph format and in a common typing font.” For the type of input we used, a more realistic estimate would be an 80-90% level of character accuracy[3].

After making observations about the characteristics of OCR data, we designed a post-processing system that improves document recall on this kind of input. This system was applied to the OCR set. The results of these experiments are presented in this paper.

2 Preliminaries

Three technologies employed in our accuracy project are scanning, optical character recognition, and text retrieval. All three affect the results of our experimentation and therefore will be considered briefly here.

2.1 Scanning

A scanning device breaks down an input page into pixels and produces a matrix of 0's and 1's. This translation is called a bit-mapped image and is only a graphical representation of the page. This bit-mapped image produced by the scanner is the input to the OCR device. The device then recognizes and translates the image into a computer-coded format (in our case ASCII). This ASCII format is suitable for use by a text retrieval system.

The bit-mapped image produced by the scanning process is highly dependent on the condition of the page being scanned as well as the care taken by the operator scanning the page. Any flaws in the hard copy or skew introduced at scan time will greatly affect the ability of the OCR device to translate the image accurately. Both the threshold values and resolution settings of the scanner will also ultimately affect the character recognition of the device. For a more complete discussion of these scanner issues, see [5] and [12]. Examples of a flawed hard copy image and a poorly scanned image can be seen in Figures 1 and 2, respectively.

¹for a page with 2500-3000 characters

pertinent literature is given elsewhere [Neuman and Witherspoon, 1969a, 1969b].

Most of this work has focused attention on the effects within the aquifer being pumped. However, the difficulty with this approach is that observations within the pumped aquifer alone may not be adequate to characterize the hydrologic properties of the aquifer and its associated confining beds. Indeed as we attempt to demonstrate in another paper [Neuman and Witherspoon, 1969a], analyses based on current theories of leaky aquifers can sometimes lead to gross errors.

Figure 1: Flawed hard copy image

front, 1980]. ... front and Stumm, 1976; Westall and

As chemical equilibrium problems are normally posed, the equilibrium concentrations of the species are to be found, given the total (analytical) concentrations of all components and the stoichiometry and stability constants of the species. A computer code, MICROQL, was developed to solve such a chemical equilibrium problem [Westall, 1979]. MICROQL is a scaled-down version of the comprehensive chemical equilibrium computer code, MINEQL [Westall et al., 1976], and

TABLE 2. Component Material Balance Equations

Component	Material Balance Equation
$T_1 = Cl_T$	$= [Cl^-] + [CdCl^+] + 2[CdCl_2]$
$T_2 = Br_T$	$= [Br^-] + [CdBr^+] + 2[CdBr_2]$
$T_3 = Cd_T$	$= [Cd^{2+}] + [CdCl^+] + [CdCl_2] + [CdBr^+] + [CdBr_2] + [CdOH^+] + [SOCd^+]$
$T_4 = SOH_T$	$= [SOH] + [SOH_2^+] + [SO^-] + [SOCd^+]$
$T_5 = T_{sT}$	$= [SOH_2^+] - [SO^-] + [SOCd^+]$
$T_6 = H_T$	$= [H^+] + 2[SOH_2^+] - [CdOH^+] - [OH^-] - [SO^-] - [SOCd^+]$

T_s represents the charge on the surface, defined as the excess of positive groups over negative groups.

Figure 2: Poorly scanned image

2.2 OCR

The OCR process can be broken down into two distinct functions: *block segmentation* followed by *text/graphics recognition*. We do not consider or evaluate graphics recognition in our experimentation; therefore, these methods will not be discussed. For our purposes, OCR can be defined as the classification and translation of textual segments of bit-mapped images into their ASCII representations.

The block segmentation function, also called *zoning*, segments the page into its logical parts. For example, if the input page is formatted with two columns of text, the reading order should be preserved—the OCR device should not concatenate the lines from left to right. Without the ability to properly zone the page, the characters and words may be correct but the text is randomly ordered. For OCR devices that do not support automatic zoning, manual zoning must be done prior to recognizing this kind of input page.

The next component is text recognition. There are three general methods currently being used by commercially available OCR devices: font matching, pattern recognition, and omnifont feature extraction[3]. These methods are not mutually exclusive and other features may be added to an OCR device to enhance the accuracy of these methods.

Font (template) matching was one of the first methods used by OCR devices. This method uses a predefined set of character templates to match input characters. It is very restrictive—especially if only a single font is defined for the device. Font matching is heavily dependent on the condition of the scanned page[3].

Pattern recognition is a second method used by OCR devices. This method applies a set of rules to the construction of characters. The shape of the input character is compared with the set of rules and the ASCII character is selected based on its closeness to a rule. Pattern recognition reduces the OCR device’s dependency on input page condition but some text features, such as broken or overlapping characters, still adversely affect its ability to determine the correct ASCII character[3].

Omnifont employs a hierarchical level of algorithms called “experts” to determine ASCII characters. Each expert level determines added information about the input. These kinds of systems combine feature recognition with heuristic processing and lexical analysis to recognize the text input[3].

Again, the OCR process is highly dependent on the quality of the image. If the image is poor, it is guaranteed that the generated text will have numerous errors. Also, the technology used by a particular OCR device dictates the kind of input that should be given to that device.

2.3 Text Retrieval

Text retrieval systems are designed to provide tools to store, manipulate, and retrieve textual information. Unlike conventional database systems, text retrieval systems can not use an exact record match retrieval method to extract information. These systems must instead calculate similarity between a user’s request and the data stored. The *hits* (documents retrieved based on calculated similarity) returned for a query may not be what the user expects or needs. The ability to evaluate the true effectiveness of a text retrieval system is reliant on user satisfaction—a highly subjective measure. *Recall*, the ratio of the number of relevant documents returned to the total number of relevant documents in the database, and *precision*, the ratio of the number of relevant documents returned to the total number of documents returned from the query[10], are currently the standard measures used.

Among well known models of information retrieval, the inverted file, vector space, and the probabilistic model are the most popular[9]. Other more heuristic text retrieval models, such as pattern recognition and concept-based clustering, are also commercially available. Describing each of these models here would be beyond the scope of this paper. We use a boolean logic inverted file based system in our experimentation and therefore describe this model in some detail below. For a more complete description of the inverted file and other models, see [10].

An inverted file system is simply a list of the documents' *non stop words* with pointers to the documents in which they are found. Most of these systems do not index insignificant words (e.g. the, and, this), called *stop words*, since they add no meaning to the document's content. Besides the document identification of an indexed word, an inverted file system may keep track of other information such as word frequency and the exact position of the word within the document. This kind of information is necessary when certain kinds of user requests and ranking of the hits are expected.

The model employed for data storage influences the method of retrieval. The most commonly used method for extracting information from an inverted file text retrieval system is a boolean logic query language[10]. These languages can be used to locate document information using combinations of boolean expressions or *queries*. The answers to the queries are computed by manipulating the stored information for the terms² that appear in each query. Additional query language features include wildcarding, stemming and canonical forms, proximity searching, and a thesaurus. These features are used to try to improve precision and recall.

For an OCR device, the input page can be used as a measure of the effectiveness of the method used to produce the output; however, this same kind of evaluation does not transfer to text retrieval. Although there are standard measures to evaluate text systems, these measures are not as sensitive to the input data as character error analysis. We assume for our evaluation that the correct database has some percentage of relevant documents retrieved for the queries. We use these retrieved sets to determine how much effect the noisy data will have on the text retrieval system.

3 Experimental environment

Our experimental environment is unique in the sense that we were given a set of documents by the Department of Energy (DOE) that had been manually corrected, together with their corresponding images. These documents were part of the LSS prototype system. The purpose of the LSS prototype was to simulate on a small scale the capture and tracking of documents pertaining to the site licensing proceedings of the Nuclear Regulatory Commission. We use both the corrected ASCII and the images generated by the LSS in our testing environment. Although we do not use the complete LSS prototype database, our document set was selected without bias. The set consists of 204 documents, for which we have images, corrected³ ASCII text on line, and hard copy.

Our collection is heterogeneous. There are numerous fonts, differing qualities of hard copy, and there is a diversity of content. The documents are scientific in nature. They contain formulas, graphs, photos, and maps. All sixteen subject areas (concepts) contained in the complete LSS are covered by our 204 documents. We use the full document text, with

²We use *term* and *word* interchangeably.

³to a level of 99.8% character accuracy[6]

documents ranging from a single page to 679 pages and an average length of thirty-eight pages. For a more complete description of these documents, consult [6].

3.1 Scan and OCR environment

The scanning of the images was not controlled in this experiment. The images produced by the contractors of the LSS are the same images we use for our testing. The use of these gives more credibility to our experimentation in the sense that they can be considered real world samples. According to our records, the images were produced with either a Ricoh or Fujitsu scanner at 300 dpi[6]. We have no information on the threshold values.

The scanned images were converted into a format usable by ISRI's vendor-independent interface[6] prior to the OCR process. Each image was then recognized using ExperVision RTK (beta version 1), a software-based OCR system for PC-DOS. For a complete accuracy assessment of this device and other OCR devices please see [7]. Eighty-one of these page images could not be recognized using this beta version, so we completed the collection using the Calera RS 9000.

We use automatic zoning for two reasons:

1. Manual zoning of 9,300 pages would have been labor intensive and time consuming.
2. The correct text had been manually zoned by the DOE contractors using a complex set of rules. There was no guarantee that the zones we selected would have matched their set exactly.

The lack of manual zoning may have had some adverse effect on the accuracy of the corresponding output. Sciences Applications International Corporation (SAIC)⁴ claims “[manual] zoning. . . results in higher output accuracy which, in turn, reduces required OCR editing” [11]. Also as stated in [2], the presence of non-text data and noise increases the difficulty of character classification and recognition.

Another side effect of automatic zoning is the generation of *graphic text*. Since graphics are not always recognized by OCR devices, non-text data, such as maps, photos, and graphs are translated to ASCII. This erroneous translation produces lines of unreadable ASCII characters.

The process described above was performed on each of the 9,300 pages. The ASCII pages generated were concatenated into complete documents for loading into the text database.

3.2 Text Retrieval environment

BASISplus is the text retrieval system we use for our experimentation. This system is based on the traditional boolean logic positional inverted file methodology[3]. BASISplus incorporates a relational database for querying structured fields on top of its original full text retrieval system (BASIS). The inverted file model was chosen for our experimentation because it is the most widely used technology[9].

3.2.1 Document environment

The correct text and raw OCR document sets were loaded as continuous text structures using the default options such as stop word lists and break characters (e.g. blank . , :).

⁴SAIC was one of the LSS contractors.

Test Query INJD-T3-Q1

LSS Prototype Test Question: Your office is trying to trace the evolution of NRC's position on repository sealing concepts (e.g., shaft and borehole seals). You need to produce a listing of all documents (including meeting material) discussing seals.

Text only translation: *Find documents discussing repository sealing concepts (shaft and borehole seals).*

FQM translation: find document where text include phrase like 'repository' & 'seal' or text include phrase like 'shaft' & 'seal' or text include phrase like 'borehole' and 'seal' order by docid

Figure 3: Example test query translation

Since the OCR text was not formatted neatly like the correct document set, a number of load parameters had to be adjusted before BASISplus would accept this OCR text properly. In particular, the *index sort parameters* needed to be adjusted. The number of “terms” to be indexed was 150,000—three times the size of the corresponding correct set. Each time a character is incorrectly translated by the OCR device, a new word is formed and in turn, indexed by the text retrieval system.

3.2.2 Query environment

BASISplus provides a query language called *Fundamental Query and Manipulation* or FQM. FQM is a command language based on boolean logic that supports wildcarding and proximity searching. These features are used infrequently in our queries. One of our queries uses wildcarding and only phrase proximity searching is employed. Although thesaurus facilities are available with FQM, none were used.

The queries we use for our testing are a subset of the LSS prototype test questions. These queries were artificially constructed to evaluate how well users were able to retrieve needed information from the database—a very different intention than ours—and therefore should reflect no bias in our testing. Many of the queries were written to retrieve information from the structured fields of the records, not the actual text. Some of these structured fields are: author name, title, descriptor field, and document type. Because of this difference, some of the original queries were excluded from our test set; many others were reworded so as to reference only the text of the document. The translation of the original English queries to their FQM representation was done by a geologist, two computer scientists, and two research assistants to ensure correctness. The interpretation of the original queries was not lost and they represent an unbiased set of seventy-one queries. Figure 3 is an example of an original test query, its text-only interpretation, and its FQM translation.

There are 205 unique search terms for the seventy-one queries. The average number of terms for the queries is five. The queries were quite relevant to the subset of 204 documents used in our testing since there was an average of eight hits per query. The same set of queries was automatically run on each database—no interactive searching was done.

4 Evaluation, Results, and Conclusion

4.1 Evaluation

The purpose of our experimentation is to determine the effect of a single independent variable, the input data, on the performance of a boolean logic inverted file text retrieval system. The dependent variable under assessment is the retrieved documents from the queries. Keeping all other variables constant, we would like to measure differences using the number of hits returned in the correct database as a benchmark. As discussed in [14] we would like to ensure the validity, reliability, and efficiency of our experiment and its results.

First, we would like to point out that we are not trying to evaluate each individual technology separately—we are evaluating the results of their synthesis. This unification introduces a number of variables into the experiment: different scanners, different settings different OCR devices and different text retrieval systems will give different results. But relative to the environment we have used for our experiments, we believe our testing is valid. The independent variable, the OCR input data, is a good indicator of the concept under investigation[14].

The reliability and efficiency of our testing stems from the diversity and size of the collection we use. Although the number of documents may seem small in comparison to other text retrieval experiments, the number of pages (9,300) and the number of index terms (150,000 in the OCR database), is quite sufficient for the kind of testing we do.

The technologies we use represent a reasonable sample of the those currently available. The OCR and TR systems, the input data, and the queries were not selected or designed with this kind of testing in mind. Their selections were not only independent of this experiment, they were independent of each other. Further, since no human influence is introduced in our retrieval testing, many of the considerations for evaluating experimental results[14] are eliminated.

The only factor that could possibly alter our results to some degree would be a modification in the definition of *correct text*. We state the correct text has a 99.8% character accuracy. We assume this measure to be correct; however, we only performed a cursory scan of the text. Further, a complex set of rules was used to determine the formatting, inclusion, and exclusion of text. If these were changed, it may affect the outcome.

Although precision and recall are the standards for evaluating performance, we do not use these criteria for our current measure of evaluation. Instead, we report on the comparison of the result sets for each query run on both collections. This evaluation method, although simplistic, will give us some indication of the answer we should expect to the question proposed in the introduction. Since it turns out that, in general, these result sets are identical, we do not expect a significantly different conclusion if precision and recall are used. We would eventually like to consider precision and recall, and also ranking[8][10] as a means of evaluation on a larger test set.

4.2 Results: experiment 1

Experiment 1 includes the loading, querying, and comparing of the correct document set with the raw OCR set. The results of the seventy-one queries that were run on the 204 documents appear in Table 1. Of the seventy-one queries that were run, sixty-three of the OCR database result sets were identical to the correct database result sets. For these 71 queries, there were a total of 632 documents returned in the correct database and 617 in the

Total number of documents retrieved for correct data	632
Total number of documents retrieved for OCR data	617
Percentage returned	97.6%
Number of queries for which result sets are identical	63
Number of queries for which result sets are different	8

Table 1: Experiment 1 query results

Poor original images or hard copy	5
Hyphenation errors	3
Original document misspellings	1
OCR character errors	6
Total	15

Table 2: Experiment 1 source of errors

OCR database. Fifteen documents were missing from the OCR result sets. The source of errors for these fifteen missing documents can be found in Table 2. Since the images were not generated by us, we do not correct errors caused by poor scanning or bad hard copy. But by massaging these OCR documents, removing end-of-line hyphenations, and making some spelling corrections, the other missing documents should be retrieved. This kind of automatic document processing is described in [13]. Cleaning up this OCR text leads us to the second version of our experiment: re-examining the query results after automatic correction of the OCR text with the post-processing system.

4.3 Results: experiment 2

Experiment 2 is experiment 1 with an additional processing step. Before loading the OCR documents into the text database, they are filtered through an end-of-line hyphenation remover and the post-processing system. No manual correction was made to these documents; only two automatic processes were applied. For this set, the break character list was adjusted to aid in the location of misspellings. For example, if the OCR device cannot make a decision on what a character should be, it puts a “~” in its place. Since the tilde is a default break character for BASISplus, this substitution caused incorrect word breaks and therefore, partial words were indexed. Although these adjustments helped the post-processing system locate errors, it may have had an adverse effect on other properly indexed terms. Evaluation of this effect was not considered.

The results of the query retrieval are documented in Table 3. Since no attempt was made to improve the images by rescanning, the errors due to poor images are not corrected. Of the remaining ten, the automatic post-processing corrected seven. Only three documents were not recalled. It is difficult to say whether two of the remaining three errors can actually be attributed to OCR error. The only necessary condition for both these documents to be

Total number of documents retrieved for correct data	632
Total number of documents retrieved for OCR data	624
Percentage returned	98.7%
Number of queries for which result sets are identical	65
Number of queries for which result sets are different	6

Table 3: Experiment 2 query results

Poor original images or hard copy	5
Missing SCP string	2
Incorrect OCR translation not corrected by the post-processing system	1
Total	8

Table 4: Experiment 2 source of errors

retrieved was the inclusion of the string SCP. Both documents in the correct database had had only a single occurrence of this string. After examining the hard copies, we found the string was not part of the original document text and therefore was not relevant to the query. In any case, these are counted as errors in Table 3.

Since there is little room for improvement from experiment 1 to experiment 2, the impact of the post-processing system is not obvious. But 1100 misspellings of the 205 distinct query terms were actually corrected in the OCR'd text. We believe that the impact of this post-processing system will be more apparent when applied to a larger test collection and ranking is used as an evaluation tool.

4.4 Conclusion

The simulation efforts for the LSS prototype demonstrated how painstaking and error-prone the conversion of hard copy documents to ASCII text could be. Further, not only was it found that manual intervention was costly, it increased the possibility of human error[11]—error that is usually inconsistent and less obvious. What our experimentation suggests is that with fairly good hard copy, with care taken during scanning, with a reasonable OCR device, and with some automatic correction, most manual preprocessing of documents is unnecessary prior to its use with a text retrieval system.

As previously stated, we are in the process of expanding the size of the database. We are also planning to perform a much larger scale experiment using OCR devices with different levels of accuracy. For this more full-blown version, we will calculate precision and recall and incorporate ranking into the evaluation. For this purpose, we will be using the *INQUERY retrieval system*[1]. These standard measures should further validate our results. We hope to empirically identify a rate of character accuracy with which a text retrieval system can cope.

5 Acknowledgments

We thank Dr. Bruce Croft and Dr. George Nagy for many valuable comments and suggestions.

References

- [1] J. P. Callan, W. B. Croft, and S. M. Harding. The INQUERY retrieval system. In *Proc. 3rd Intl. Conf. on Database and Expert Systems Applications*, pages 78–83, 1992.
- [2] Richard G. Casey and Kwam Y. Wong. *Image Analysis Applications*, chapter 1, pages 1–36. Marcel Dekker, 1990.
- [3] Delphi Consulting Group, Inc. Text retrieval systems: A market and technology assessment, 1991.
- [4] D. Harman and G. Candela. Retrieving records from a gigabyte of text on minicomputer using statistical ranking. *J. American Soc. for Inf. Sci.*, 41(8):581–589, 1992.
- [5] G. Nagy. Optical scanning digitizers. *IEEE Computer*, pages 13–24, 1983.
- [6] T. A. Nartker, R. B. Bradford, and B. A. Cerny. A preliminary report on UNLV/GT1: A database for ground-truth testing in document analysis and character recognition. In *Proc. 1st Symp. on Document Analysis and Information Retrieval*, pages 300–315, Las Vegas, NV, March 1992.
- [7] Stephen V. Rice, Junichi Kanai, and Thomas A. Nartker. A report on the accuracy of OCR devices. Technical Report 92-02, Information Science Research Institute, University of Nevada, Las Vegas, March 1992.
- [8] S. E. Robertson. The probability ranking principle in information retrieval. *Journal of Documentation*, 33:294–304, 1977.
- [9] G. Salton. *Automatic Text Processing*. Addison-Wesley, Reading, MA, 1989.
- [10] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw Hill, New York, 1983.
- [11] Science Applications Intl. Corp. Capture station simulation: Lessons learned, Final Report, for the Licensing Support System, November 1990.
- [12] S. N. Srihari. Document image understanding. In *Proc. Fall Joint Conference*, Dallas, TX, 1986. ACM-IEEE Computer Society.
- [13] Kazem Taghva, Julie Borsack, Bryan Bullard, and Allen Condit. Post-editing through approximation and global correction. Technical Report 93-05, Information Science Research Institute, University of Nevada, Las Vegas, March 1993.
- [14] Jean Tague-Sutcliffe. The pragmatics of information retrieval experimentation, revisited. *Inf. Proc. and Management*, 28(4):467–490, 1992.