

Information Access in the Presence of OCR Errors

Kazem Taghva
Info. Sci. Research Institute
U. of Nevada, Las Vegas
Las Vegas, NV 89154-4021
taghva@isri.unlv.edu

Thomas Nartker
Info. Sci. Research Institute
U. of Nevada, Las Vegas
Las Vegas, NV 89154-4021
tom@isri.unlv.edu

Julie Borsack
Info. Sci. Research Institute
U. of Nevada, Las Vegas
Las Vegas, NV 89154-4021
jborsack@isri.unlv.edu

ABSTRACT

Over the last 15 years, the Information Science Research Institute (ISRI) at the University of Nevada, Las Vegas (UNLV) has conducted information access research in the presence of OCR errors. Our research has focused on issues associated with the construction of large document databases. In this paper, we will highlight our findings and detail our current activities.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*retrieval models*; I.7.5 [Document and Text Processing]: Document Capture—*optical character recognition (OCR)*

General Terms

Measurement, Performance, Experimentation

Keywords

information extraction, document conversion, markup, categorization

1. INTRODUCTION

The Information Science Research Institute (ISRI) has performed in-depth research with optical character recognition (OCR), information retrieval (IR), and related topics since 1989. Our work was initially focused on the effects of using optically recognized text in the IR environment. Since ISRI's work was the first comprehensive study in this area, there were many approaches we could have taken. We decided that understanding the effects of OCR with respect to particular IR models was a well-founded one. Section 2 gives an overview of this work and identifies published papers that describe this research in detail.

Our decision to test OCR with various models was constructive because it gave us insight into potential problems

that at first may not have been obvious. For example, normalization applied in vector-space systems caused irregular behavior in ranking. This was not a problem for the Boolean system we had previously studied. By analyzing the issues certain IR models had with OCR generated text, we were able to build post-processing systems that would improve OCR text prior to its use with these systems. In fact, certain issues identified by ISRI generated research activities for other groups as well[19]. Section 3 describes the most notable applications produced at ISRI from this work.

As other information access tasks became more mainstream, the same questions we asked about OCR and IR could also be asked of other text processing tasks like categorization and information extraction. In 2001, ISRI published several papers on the effects of OCR on text categorization. Section 4 gives an overview of our findings in these studies. Some of our more recent research has tried to evaluate the effects of OCR on information extraction. Although preliminary, it seems that OCR can cause unanticipated complications for this task as well. Section 5 discusses our initial research.

2. OCR AND INFORMATION RETRIEVAL

ISRI's initial research was ground-breaking in that no combined OCR/IR studies had previously been performed. Although the relationship seems obvious, the notion of applying IR directly to un-corrected OCR text was unprecedented. In 1989, ISRI was already performing accuracy tests of the top OCR devices to aid OCR vendors in their pursuit of improved recognition[14, 13, 9]. But the mission of our institute was to discover efficient ways for converting large paper document collections to electronic form for retrieval.

Our first objective was to evaluate retrieval effectiveness. As mentioned in the introduction, our approach examined specific IR models, but our most notable discovery is not model dependent: *average precision and recall is not negatively affected by OCR errors*. In fact, in all our experiments, recall for the OCR versions was actually higher.

The retrieval systems we used in our experiments were selected with their underlying models in mind. Among these are Boolean, probabilistic, and vector-space models. Most commercial products are based on techniques and principles derived from these models. We believe that characterizing the effects of OCR text on systems (either commercial or research prototype) implemented using these models will give insight into what one should expect when querying OCR text. In the following sections, we review our research and experimentation for these well-established models.

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

HDP'04, November 12, 2004, Washington, DC, USA.
Copyright 2004 ACM 1-58113-976-4/04/0011 ...\$5.00.

2.1 Effects on Implementation Components

There are three components of IR systems that impact or are impacted by the use of OCR text. The first of these, the inverted index, is the heart of the IR system. The words of the document, together with data about their distribution, are stored here. The inverted index is affected in several ways by the use of OCR text, but most notably, its size is increased. In our first experiment with 204 documents, the index of the OCR text was three times the size of the corrected text index. In our next two experiments, with 674 documents, the inverted index increased to 5 times its corrected version size. Some interesting statistics comparing OCR text with its corrected counterpart can be derived from an IR system's index. Table 1 reveals some of these variations. This table lists various statistics for the corrected collection and for three recognized versions of 674 documents. These statistics indicate that:

- An OCR collection requires more overhead (database size and terms occurring only once).
- There will be an increase in per document processing time (average number of terms per document).
- There is a considerable increase in the size of the index (number of unique indexed terms).
- There is an increase in useless terms (terms occurring only once).¹
- At least in our collection, the corrected text is missing some information that is picked up by the device (correctly spelled words).

Probably the most unanticipated set of figures is the *correctly spelled words*. There are more correctly spelled words in each of the three OCR collections than there are in the corrected collection. We attribute this anomaly to the removal of document text from the corrected collections[1].

Another source of problems not readily apparent are end of line hyphenations. Hyphenations are commonly used when documents are typeset for printing. With no special handling, the OCR outputs exactly what it "sees" and the IR system indexes what it "reads." The consequence: a percentage of words in the index are split consuming twice as much index space as they should. Further, the words are useless as they are and have a negative impact on retrieval since the *full* word is not available for querying.

Recall that some inconsequential words (stopwords) do not get indexed by an IR system. Sometimes, stopwords are misrecognized by the OCR device and in turn, are inadvertently indexed. In a boolean system, this recognition error probably will not make much difference.² But for systems that weight the importance of terms using term frequencies, this kind of transformation can disrupt term assignment and therefore document to query relevance in those documents affected. Also note that since stopwords are used frequently, these misrecognized terms can become the *most* frequent "terms" for these documents.

Stemming is another common practice employed by IR systems. In all our projects, we applied only S-removal

¹These are mostly misspellings and "graphic text" strings.

²Erroneous query results would occur only if a stopword had been translated to a correctly spelled query term (possible but not probable).

stemming to the OCR text. We disclosed no adverse effects from this application. Although it does not seem as if full stemming would produce very different results, actual experiments followed by a thorough analysis would be necessary to verify this claim.

2.2 Effects on IR Models

A boolean system relies on the presence of query terms within documents to determine relevance. We found in our first experiment using a boolean system that redundancy in the document text could compensate for most of the problems associated with OCR errors. Our results showed that most of the documents returned from querying the corrected collection (632) were also returned from querying the OCR collection (617). After applying our post-processing system[23], a precursor to the MANICURE system described in Section 3.2, seven more documents were retrieved so 98.7% of the corrected documents were returned when the OCR set was queried[27].

For systems that apply term weighting and document ranking, the translation of text the device produces causes a more significant impact. This consequence can be attributed directly to the mutations to the text that affect term frequencies, and hence cause skewed term weighting. An example of such an effect is the misrecognition of the stopword "the." In the probabilistic indexing system, the term with the maximum frequency is used to normalize term weights. When this value is uncharacteristically high, the weights of other document terms are underestimated. This effect eventually manifests itself to the user when the documents are ranked to a query.

The maximum term count is not the only frequency that may get altered. Other normalization values suffer as well. For example, we found in the vector space model that cosine normalization negatively affects document ranking. The SMART system[15], the vector space implementation used in our experiments, allows the application of several weighting techniques. One of these techniques applies vector length (i.e. square root of the sum of squares of the weights in the vector) to normalize the length of the documents. It functions as a way of equalizing the importance of terms in short documents even though they occur less frequently. This application works well for documents that are clean and manually typed. But for an OCR-generated collection where the vectors are artificially padded with misspellings and "graphic text" this normalization technique causes problems with term weighting, and in turn, document ranking. A complete analysis showed the variability was attributed directly to the cosine weighting component [22].

Pure document length (number of non-stopwords) is another factor that is sometimes used in term weighting calculations. But as we can see from Table 1 there is a marked increase in document length for the recognized text (note: *average number of terms per document*).

Note that even in the presence of these variations in document frequencies between the corrected text and its OCR generated counterpart, for these weighted systems, average precision was hardly affected. Our experiment with SMART gave us the opportunity to test OCR text within a vector space framework, and examine several weighting techniques, as discussed. These techniques were applied to our OCR collection and these results demonstrated that no significant difference is apparent when average precision is compared.

Table 1: Statistics for the corrected collection and three OCR collections

Statistic	Corrected	OCR 1	OCR 2	OCR 3
Database size (bytes)	15,686,772	37,080,489	40,918,148	42,247,537
Average number terms/document	6,114	9,321	8,583	7,925
Number of unique indexed terms	78,494	320,338	387,276	414,715
Terms occurring only once	36,742	223,058	278,907	296,572
Terms occurring more than once	41,752	97,280	108,369	118,143
Correctly spelled words	22,833	25,241	24,728	23,552

Relevance feedback is another well-known technique designed to use information from known relevant documents to improve query effectiveness[17]. In our experiment with SMART we were able to apply this technique to the OCR-generated text. We discovered an interesting phenomenon: as the queries were expanded with more feedback terms, the precision for the corrected collection improved while the precision for the OCR collection leveled off after the initial twenty word expansion.

After analysis of the expanded queries, we found that the difference between the results for these two collections was not due to query degradation but was attributable to a lack of improvement in rank for a few of the relevant documents in the OCR collection. The documents that did not improve in rank for the feedback runs were the same documents with a discrepancy in rank for the initial runs. In general, these documents represent poor candidates for recognition due to the poor quality of the original images or the large amounts of graphical material contained in the documents. In any case, these results show that feedback cannot overcome some of the shortcomings found in OCR generated collections[25].

3. OCR POST-PROCESSING

Following our initial studies on OCR and IR, the opportunity for improving raw OCR documents for subsequent information access seemed clear. ISRI's research became focused on developing automated systems to perform OCR post-processing.

3.1 OCRSpell

OCRSpell is a spelling correction system designed specifically for OCR generated text[33]. The system integrates many techniques for correcting errors induced by an OCR device. This system is fundamentally different from many of the common spelling correction applications. The system is based on static and dynamic device mappings, approximate string matching, and n-gram analysis. It is a statistically based, Bayesian system that incorporates a learning feature that collects confusion information at the collection and document levels. OCRSpell was designed to be as automatic as possible and to gain knowledge about the document set whenever user interaction becomes necessary.

Conceptually, the system is composed of five modules. The actual implementation of the system closely follows this model.

1. A parser designed specifically for OCR generated text

An effective scheme for parsing the text is essential to the success of the system. For OCRSpell, we chose to implement our parser in Emacs LISP[10] due to its robust set of high level functions for text search-

ing and manipulation. Rather than designing many parsing algorithms for different types of working text, we chose to make the parser as general as possible and provided the user with a robust set of filtering and handling functions. The system's treatment of word boundaries, word combining symbols, and other text characteristics is essential to the overall success of the system. The other components of the system rely heavily on the parser to make heuristically correct determinations concerning the nature of the current text being processed.

2. A virtual set of domain specific lexicons

Another important issue to address prior to the development of any candidate word selection method is the organization of the lexicon, or dictionary, to be used. Our system allows for the importation of Ispell[6] hashed dictionaries along with standard ASCII word lists. Since several domain specific lexicons of this nature exist, the user can prevent the system from generating erroneous words that are used primarily in specific or technically unrelated domains.

OCRSpell provides an infrastructure that is extremely conducive to lexicon management. Since the system allows for the importation of dictionaries, they can be kept separate. Optimally, each collection type (i.e. newspaper samples, sociology papers, etc.) would have its own distinct dictionary that would continue to grow and adapt to new terminology as the user interactively spell checks documents from that collection.

3. The candidate word generator with global/local training routines (confusion generators)

At the heart of the system is a statistically-based string matching algorithm that uses device mapping frequencies along with n-gram statistics pertaining to the current document set to establish a Bayesian ranking of the possibilities, or suggestions, for each misspelled word. This ensures that the statistically most probable suggestions will occur at the beginning of the choices list and allows the user to limit the number of suggestions without sacrificing the best word alternatives.

A two-level statistical device mapping word generator is used to generate possibilities for incorrect words. A simple level saturation technique is used to generate new words from static confusions. This technique relies heavily on a Bayesian ranking system that is applied to the subsequent candidate words.

The confusion generator determines the longest common subsequence and the subsequent confusions for words that have been manually replaced. It uses the

popular dynamic programming longest common subsequence algorithm. This algorithm was chosen so that heuristically optimal subsequences can be chosen. The method used here is from[3].

Dynamic device mappings are created and applied by the user interface in much the same way that static device mappings are applied in the level saturation generation process. A longest common subsequence process is invoked whenever the user manually inserts a replacement to a misspelling.

4. The graphical user interface

The user interface combines the word and confusion generator and adds many options and features to insure an easy to use, robust system. This interface was written in Emacs LISP. The interface can be controlled by a series of meta commands and special characters. Many of the commonly used interface options can be selected directly from the menu. The user can join the current word with the previous or next word, insert the highlighted word or character sequence into the lexicon, select a generated choice, or locally/globally replace the highlighted text by a specified string. If the user chooses to replace the text, the confusion generator is invoked and the subsequent confusions are added to the device mapping list. This means that any errors occurring later on in the document with the same confusions (e.g `rn` \rightarrow `m`) will have automatically generated choices in the interface's selection window. Of course, this means the effectiveness of OCRSpell improves as it gains more information about the nature of the errors in any particular document set. Table 2 contains a list of all of the interactive features of the system.

OCRSpell was designed to be a tool for preparing large sets of documents for either text retrieval or for presentation. It was also developed to be used in conjunction with MANICURE (see Section 3.2). OCRSpell is designed around common knowledge about typical OCR errors and dynamic knowledge which is gathered as the user interactively spell checks a document. For a complete description of this system, its design and evaluation, please see [33].

3.2 MANICURE

ISRI ran many experiments comparing OCR output to manually corrected versions of the same collections. From these analyses, we were able to characterize the problems that OCR text may cause when applied in retrieval. We found that by post-processing the *complete* document, mis-recognized words could be corrected and other post-cleanup could be performed with a very high level of accuracy[23, 21].

MANICURE (Markup ANd Image-based Correction Using Rapid Editing) is ISRI's software solution for performing automated OCR post-processing. The system is designed to take advantage of document characteristics such as word forms, geometric information about the objects on the page, and font and spacing between textual objects (if available) to mark the logical structure of a document. In addition, the system automatically detects and corrects OCR spelling errors by using dictionaries, approximation matching, the knowledge of typical OCR errors, and frequency and distribution of words and phrases in a document.

MANICURE is specifically designed to prepare text collections from printed materials for information retrieval applications. In this capacity, depending on the application, requirements on accuracy and text structure vary. In what follows, we will list important applications for which MANICURE is designed.

- MANICURE will try to distinguish between the actual content of the document and the superficial information which is part of its presentation. For example, the sequence of words in the printed document identifying the journal in which it appeared is presentational rather than being a part of a document's content. This extra text (and extra non-alphanumeric characters such as end of line hyphenation) can be both helpful and harmful. It is helpful in the sense that it identifies meta-knowledge about the document such as date of publication, but it could also be harmful if it is not properly removed or managed, such as with end of line hyphenation; this extra text will also clutter the retrieval system's index with useless information.
- ISRI studies on effects of OCR errors on retrieval[27, 24, 25] have pointed out that certain advanced functionalities of information retrieval systems, such as ranking, are not robust enough to overcome OCR errors. It is our view that the more advanced retrieval techniques and applications require a higher character accuracy rate and less graphic text. MANICURE can be used in both automatic and semi-automatic modes to produce text with higher levels of accuracy.
- Associated with each document is a list of structured data. This data generally contains information such as the author, the date of publication, document control number, title, and so on. Some of this information is already in the text of the document (such as the title) and some is added for the sake of record keeping (such as a document control number). By properly marking some of the structured data, MANICURE provides an environment in which this information can be extracted automatically, or in the case of structured based retrieval systems[11], can be manipulated by the retrieval engine itself.
- The logical structure of the text can be used in many retrieval applications. For example in [4, 7, 16, 2, 34] the individual sentences, paragraphs, sections, and section titles are analyzed as a part of the solution to these particular applications. MANICURE builds a logical representation for each document in which all these objects are marked.
- The text from printed materials can be produced in different formats for different uses. For example, the Hypertext Markup Language (HTML) (derived from SGML[5]) is a common format used for marking up documents for viewing by World-Wide Web. MANICURE, in addition to providing text in HTML format, also outputs documents in another SGML-based format with detailed information about logical structure, font, subscript, superscript, and geometry information. One use of this format is to enable retrieval systems to search on the text of a document and highlight hits on the original page images.

Table 2: OCRSpell’s Interactive Features

Key	OCRSpell Feature
i	insert highlighted word into lexicon
r	replace word, find confusions
b	backward join (merge previous word)
j	forward join (merge next word)
g	global replacement
<space>	skip current word or highlighted region
<character>	replace highlighted word with generated selection
q	quit the OCRSpell session

Over the past several years, MANICURE’s most applied resource has been its ability to automatically correct misspellings in the OCR text. The number of corrections made seemed impressive but there was no measurable proof that MANICURE made a significant difference in error correction. In 2003, ISRI prepared an experiment that tested the corrections made by MANICURE. These tests showed that the word accuracy improvement of MANICURE output over raw OCR output ranged from 0.51% up to 2.55% (depending on the quality of the document) for all non-stopwords. This percentage of improvement is comparable to correcting as many as 30% of the misspellings. In addition, these results showed that the improvement provided by MANICURE increases as page quality decreases[12].

These tests prove that MANICURE improves the quality of OCR text for retrieval and has the potential to increase average precision[29]. For many conversion projects, a MANICURE’d collection may be all that is required. But MANICURE also provides a semi-automated correction interface that can bring a document to any required level of accuracy.

3.3 OCR QC

Having an automated way of measuring OCR quality is a critical part of any large document conversion project. Yet, the only measures available were the estimated character accuracy provided by the OCR or time-consuming ground-truthing and sampling. ISRI found a solution to measuring OCR quality by building on information we already had. Using word accuracy, actually *non-stopword* accuracy, instead of character accuracy was the first necessary modification to enhancing the quality control (QC) process.

To estimate non-stopword accuracy of recognized documents, statistics about non-stopwords must be collected. For example, the QC procedure must know the number of misspelled and correctly spelled words in a document to calculate its percentage of non-stopword correctness. Since MANICURE has access to this data as it processes a document, it made sense to include the QC procedure within MANICURE.

The QC process takes the number of correct words and misspellings on each page and for the entire document and computes a ratio. The document’s accuracy is then compared to a predetermined threshold. This process is completely automated. Documents that do not pass QC can be marked for subsequent review. For a complete description of the MANICURE QC, see [20].

4. CATEGORIZATION OF OCR

Our goal here was to formulate experiments that would

give us the most insight into what effect OCR errors may have on document categorization[31, 32]. Broadly, there are two ways in which errors can influence categorization. First, by introducing errors into the training set, and second, by reducing the ability of incoming documents to get categorized correctly. In [32] we report on four experiments that help explain both these possibilities.

Good Training/Bad Test Set (E1): In this experiment, the training set, although uncorrected OCR, was selected for its good quality. The test set was just the opposite; it was selected for its poor OCR quality.

Mixed Training/Mixed Test Set (E2): This test used the same set of documents as E1, but documents were selected randomly from the complete set for both training and testing.

Good Training/Auto-Corrected (E3): E3 is the same as E1, except that two difficult to categorize documents were initially run through MANICURE, a system we built to improve OCR documents prior to classification or retrieval[29].

Good Training/Manually-Corrected (E4): This set of runs, labeled E4, is the same test as E3 except that the two documents in E3 are *manually corrected*.

In each experiment described above, we performed several runs based on both the Bernoulli and multinomial probability models. In addition, each experiment included a limited vocabulary run that applies the multinomial probability technique. The limited vocabulary “list” consists of several merged dictionaries that include domain specific terms that a general dictionary may have missed in the indexing process.

As with other classification experiments[18], our results showed that dimensionality reduction improved categorization. Dimensionality reduction eliminates terms that contribute the least amount of information for the categories. With respect to OCR text, this includes terms that are mis-recognized by the device and contribute no value to the category. Removal of OCR errors through dimensionality reduction clearly improves the accuracy of categorization.

Another observation we made with respect to categorization of OCR documents is that OCR-generated text may have little or no effect in general when incoming documents are being classified but the selection of good quality OCR documents for training is essential.

Finally, we have seen examples of degraded documents that were incorrectly categorized while the error-free version

of the same document is categorized correctly. We discovered in several of our experiments with information retrieval and OCR that some poorly recognized documents were unretrievable without some corrective intervention[27, 26]. This dilemma is paralleled in categorization.

In nearly all our experimental runs, there were two poorly recognized documents that just couldn't seem to get categorized properly. In experiment E3 listed above, we applied MANICURE to see if automated OCR cleanup and error correction could improve classification. In fact, one of the two documents did get categorized correctly after running the documents through MANICURE. The fact that automatic correction helped classify this document correctly is just part of the story. We also reported on the improvement to the category itself. Both of these poorly recognized documents belonged to a single category. After MANICURE and retraining, the percentage of correctly spelled category terms also improved. Full manual correction of these documents (experiment E4) offered no additional categorization improvement over the automatically MANICURE'd runs.

5. HMM EXTRACTION IN OCR

Like other information access tasks, extraction from OCR is complicated by erroneous recognition. This includes both character and zoning errors. Our personal address extraction task is a case in point. We apply the Hidden Markov Model (HMM) for discovering personal addresses in free text and found that several difficulties caused by OCR affected HMM training. For example, the order of the address components in the OCR output did not always correspond to the correct order of the address on the image. This affected the tagging applied by the HMM because the order of the tags did not fit the topology of the HMM. In some documents, personal addresses on a page image were either partially or completely lost in the OCR process. Also, contextual information useful for identifying an address would occasionally be lost as well.

These OCR issues raised the question of whether tagged training data should be "cleaned up" to reflect our preconceived ideas about how the HMM should be structured, or whether it should be left "as is" to reflect the nature of the data. In our testing, we decided to compare the performance of an HMM trained on tagged data which included "incorrect" personal addresses (*default*) with an HMM trained only on correct personal addresses (*correct*) where the training addresses fit our assumed topology[30].

Human experts identified 251 documents which contained personal addresses. Addresses in 187 of these documents were tagged with the METAmarker tool[28]. Eighty-nine documents were randomly selected for the training set and were designated as the *default* HMM. Some of these 89 documents contain "incorrect" tagging due to the OCR misrecognition. A second training set was derived from the default HMM by removing all examples which contained "incorrect" tagging. The HMM based on this training set was called the *clean* HMM. Three experiments were performed. In the first, the default HMM was run on the test set. In the second, the clean HMM was run against the same data. Finally, the emission table for the clean HMM was optimized using shrinkage.

The standard performance measures of precision, recall, and F1 were calculated for each experiment. Let TP be the number of true positives, that is, the number of documents

in which both experts and the HMM agreed contained personal addresses. Let FN be the number of false negatives, i.e., the number of documents which experts said contain personal addresses, but the HMM marked as not having personal addresses. We then define *recall* as

$$recall = \frac{TP}{TP + FN} \quad (1)$$

Letting FP signify the number of false positives, i.e., those documents which the HMM marked as containing personal addresses but which experts decided does not, *precision* is defined as

$$precision = \frac{TP}{TP + FP} \quad (2)$$

The harmonic mean of precision and recall is called the $F1$ measure, defined as:

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (3)$$

The results of our experiments are presented in Table 3. The value TN represents the number of true negatives, i.e., documents where both the experts and the HMM agreed no personal addresses were discovered.

Our testing indicates that shrinkage does not significantly improve an HMM's performance for this task. In fact, there was some degradation although probably only an insignificant amount. Furthermore, the surprising success of the default HMM seems to indicate OCR noise is affecting the ability of the HMM to locate addresses.

Although it has been shown that OCR errors do not affect the average effectiveness of information retrieval[25, 26], some studies have indicated that noisy data can degrade information extraction tasks such as text summarization [8]. For this limited sample, it seems the default HMM worked best. This could indicate that an HMM designed for clean text would give degraded results with OCR generated text.

6. CONCLUSION

In the last fifteen years, ISRI has concentrated its efforts in a very focused area: the combined use of OCR with information access. But our research has truly had a significant impact on government, large corporations, and on entities performing large-scale document conversions. We believe our solutions to many of these problems can aid other related projects. ISRI is referenced as the research authority in this area and our technologies are currently being applied in several conversion projects.

7. REFERENCES

- [1] J. Borsack and B. Huey. Verification of GT1. Technical Report 94-05, Information Science Research Institute, University of Nevada, Las Vegas, July 1994.
- [2] J. P. Callan. Passage-level evidence in document retrieval. In *Proc. 17th Intl. ACM-SIGIR Conf. on Research and Development in Information Retrieval*, pages 302-310, Dublin, Ireland, July 1994.
- [3] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. The MIT Press, tenth edition, 1993.
- [4] M. Fuller, E. Mackie, R. Sacks-Davis, and R. Wilkinson. Structured answers for a large

Table 3: HMM experiment results

experiment	TP	FP	FN	TN	precision	recall	F1
default	100	6	15	493	0.869	0.943	0.905
clean	98	9	17	490	0.852	0.916	0.881
shrinkage	99	9	16	490	0.861	0.917	0.888

- structured document collection. In *Proc. 16th Intl. ACM/SIGIR Conf. on Research and Development in Information Retrieval*, pages 204–213, Pittsburgh, PA, June 1993. ACM Press.
- [5] C. F. Goldfarb. *The SGML Handbook*. Oxford University Press, 1990.
- [6] R. E. Gorin, P. Willisson, W. Buehring, G. Kuenning, et al. Ispell, a free software package for spell checking files. The UNIX community, 1971. version 2.0.02.
- [7] M. A. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *Proc. 16th Intl. ACM/SIGIR Conf. on Research and Development in Information Retrieval*, pages 59–68, Pittsburgh, PA, June 1993. ACM Press.
- [8] H. Jing, D. Lopresti, and C. Shih. Summarizing noisy documents. In *Proceedings of SDIUT'03*, pages 111–119, Greenbelt, MD, April 2003.
- [9] J. Kanai, S. V. Rice, T. A. Nartker, and G. Nagy. Automated evaluation of ocr zoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):86–90, January 1995.
- [10] B. Lewis, D. LaLiberte, and the GNU Manual Group. *The GNU Emacs Lisp Reference Manual*. Free Software Foundation, 1.05 edition, 1992.
- [11] I. A. Macleod. A query language for retrieving information from hierarchic text structures. *The Computer Journal*, 34(3):254–264, 1991.
- [12] T. Nartker, K. Taghva, R. Young, J. Borsack, and A. Condit. OCR correction based on document level knowledge. In *Proc. IS&T/SPIE 2003 Intl. Symp. on Electronic Imaging Science and Technology*, volume 5010, pages 103–110, Santa Clara, CA, January 2003.
- [13] T. A. Nartker and S. V. Rice. OCR accuracy: UNLV's third annual test. *INFORM*, 8(8):30–36, September 1994.
- [14] T. A. Nartker, S. V. Rice, and J. Kanai. OCR accuracy: UNLV's second annual test. *INFORM*, 8(1):40–45, January 1994.
- [15] G. Salton. *The SMART Retrieval System, Experiments in Automatic Document Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1971.
- [16] G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In *Proc. 16th Intl. ACM/SIGIR Conf. on Research and Development in Information Retrieval*, pages 49–58, Pittsburgh, PA, June 1993. ACM Press.
- [17] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.
- [18] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, March 2002.
- [19] A. Singhal, G. Salton, and C. Buckley. Length normalization in degraded text collections. In *Fifth Annual Symposium on Document Analysis and Information Retrieval*, pages 149–162,, University of Nevada, Las Vegas, 1996.
- [20] I. Staff. Measuring and delivering 95% non-stopword document accuracy. Technical Report 2003-04, Information Science Research Institute, University of Nevada, Las Vegas, September 2003.
- [21] K. Taghva, J. Borsack, B. Bullard, and A. Condit. Post-editing through approximation and global correction. *International Journal of Pattern Recognition and Artificial Intelligence*, 9(6):911–923, 1995.
- [22] K. Taghva, J. Borsack, and A. Condit. Effects of OCR errors on ranking and feedback using the vector space model. Technical Report 94-06, Information Science Research Institute, University of Nevada, Las Vegas, August 1994.
- [23] K. Taghva, J. Borsack, and A. Condit. An expert system for automatically correcting OCR output. In *Proc. IS&T/SPIE 1994 Intl. Symp. on Electronic Imaging Science and Technology*, pages 270–278, San Jose, CA, February 1994.
- [24] K. Taghva, J. Borsack, and A. Condit. Results of applying probabilistic IR to OCR text. In *Proc. 17th Intl. ACM/SIGIR Conf. on Research and Development in Information Retrieval*, pages 202–211, Dublin, Ireland, July 1994.
- [25] K. Taghva, J. Borsack, and A. Condit. Effects of OCR errors on ranking and feedback using the vector space model. *Inf. Proc. and Management*, 32(3):317–327, 1996.
- [26] K. Taghva, J. Borsack, and A. Condit. Evaluation of model-based retrieval effectiveness with OCR text. *ACM Transactions on Information Systems*, 14(1):64–93, January 1996.
- [27] K. Taghva, J. Borsack, A. Condit, and S. Erva. The effects of noisy data on text retrieval. *J. American Soc. for Inf. Sci.*, 45(1):50–58, January 1994.
- [28] K. Taghva, M. Cartright, and J. Borsack. An efficient tool for xml data preparation. Technical Report 2004-01, Information Science Research Institute, University of Nevada, Las Vegas, July 2004.
- [29] K. Taghva, A. Condit, J. Borsack, J. Kilburg, C. Wu, and J. Gilbreth. The MANICURE document processing system. In *Proc. IS&T/SPIE 1998 Intl. Symp. on Electronic Imaging Science and Technology*, San Jose, CA, January 1998.
- [30] K. Taghva, J. Coombs, and R. Pereda. Address extraction using hidden markov models. For submission to Proc. IS&T/SPIE 2004 Intl. Symp. on Electronic Imaging Science and Technology, July 2004.
- [31] K. Taghva, T. Nartker, J. Borsack, S. Lumos, A. Condit, and R. Young. Evaluating text

- categorization in the presence of OCR errors. In *Proc. IS&T/SPIE 2001 Intl. Symp. on Electronic Imaging Science and Technology*, pages 68–74, San Jose, CA, January 2001.
- [32] K. Taghva, T. A. Nartker, and J. Borsack. Recognize, categorize, and retrieve. In *Proc. of the Symposium on Document Image Understanding Technology*, pages 227–232, Columbia, MD, April 2001. Laboratory for Language and Media Processing, University of Maryland.
- [33] K. Taghva and E. Stofsky. Ocrspell: An interactive spelling correction system for OCR errors in text. *Intl. Journal on Document Analysis and Recognition*, 3(3):125–137, March 2001.
- [34] R. Wilkinson. Effective retrieval of structured documents. In *Proc. 17th Intl. ACM-SIGIR Conf. on Research and Development in Information Retrieval*, pages 311–317, Dublin, Ireland, July 1994.