INFORMATION RETRIEVAL BASED ON OCR ERRORS IN SCANNED DOCUMENTS

Y. FATAICHA[1, 3], M. CHERIET[1, 2], J. Y. NIE[3], and C. Y. SUEN[2]

1. LIVIA Laboratory, École de Technologie Supérieure de Montréal, Québec, Canada
2. CENPARMI, Concordia University , Montréal, Québec , Canada
3. RALI laboratory, Université de Montréal, Québec, Canada

fyoussef@livia.etsmtl.ca; cheriet@gpa.etsmtl.ca; nie@iro.umontreal.ca; suen@cenparmi.concordia.ca

Abstract

An important proportion of documents are document images, i.e. scanned documents. For their retrieval, it is important to recognize their contents. Current technologies for optical character recognition (OCR) and document analysis do not handle such documents adequately because of the recognition errors. In this paper, we describe an approach that integrates the detection of errors in scanned texts without relying on a lexicon, and this detection is integrated in the research process. The proposed algorithm consists of two basic steps. In the first step, we apply editing operations on OCR words that generate a collection of error-grams and correction rules. The second step uses query terms, error-grams, and correction rules to create searchable keywords, identify appropriate matching terms, and determine the degree of relevance of retrieved document images. Algorithms has been tested on 979 document images provided by Media-team databases from Washington University, and the experimental results obtained show the effectiveness of our method and indicate improvement in comparison with the standard methods such as exact or partial matching, N-gram overlaps, and Q-gram distance.

Keywords: Image document, text processing, OCR, String Matching, N-gram statistics, confusion probability, query term expansion, information retrieval.

1. Introduction

In spite of all research done in processing document images, there are still some open problems in this field. First, due to the noise and the poor contrast in the images, many extraction features - intensity, texture, shape, entropy etc. - must be acquired to distinguish text from complex document image [1]. Second, it is difficult to recognize the text accurately. Word recognition is much more difficult because OCR errors may include edition operations such as characters substitution, deletion, and insertion [3, 4, 5, 6]. These problems are not trivial. It is difficult to arrive at an OCR result with high accuracy.

The goal of information retrieval (IR) is to search large textual databases and return the documents that the system considers relevant to the user's query [7]. Electronic documents produced by scanning and OCR software contain recognition errors, and the rate of errors increases significantly if the quality of document image degrades. Those particular documents may then become inaccessible using conventional retrieval on their OCR results. This fact significantly affects the retrieval results. The goal of this research is to design an IR method specifically for retrieving document images. In particular, we will take into account the possible recognition errors using the retrieval process.

Previous studies have tried to reduce recognition error with a correction step. Most approaches to the correction of scanning errors are based on lexicon. Errors are detected by searching the text for words that do not appear in the lexicon [3, 7]. This leads to many false alarms, since a lexicon is not able to cover everything. Many studies, see for examples [3, 4, 5, 6], show that three common mistakes – characters substitution, deletion and insertion - cause 80% to 90% of all typing errors. Taghfa and Stofski [3] describe approximate string matching using EMACS text parser to determine what we refer to as confusions. Ohta et al. [4] present probabilistic text retrieval methods to carry out a full-text search of English documents containing OCR errors. The validity of retrieved terms is determined based on error-occurrence and character-connection probabilities. All possible error information that was included in the confusion matrices significantly decreased the precision rate.

In an effort to reduce these losses, we have incorporated the use of edit-distance to locate OCR errors and to collect frequent error-grams and correction rules. The advantage here is that by focusing on a small set of common n-gram errors, more elaborate and reliable methods can be applied to enhance retrieval performance. N-grams statistics have been used since 1960s. Suen [2] tabulated the growth in the number of distinct n-grams as a function of vocabulary size, their word-positional dependence, and the influence of the selected corpus. Croft et al. [5] match extended query term by using Q-gram distance (number of n-grams contained in two words versus the number they share). This method needs better closeness measures to eliminate spurious terms in the expansion.

Our proposed method takes advantage of the capacity of dynamic programming to generate error-grams derived from erroneous substrings, which are introduced in the retrieval process. The added words are gathered at level $_i$ (i is the number of corrections applied to word list). Erroneous substrings called Error-grams are weighted depending on their frequency. A commercial OCR has been applied to 979 images of Media-Team document database from Washington University. Error-grams and correction rules are then combined to extend query words, and the experiments show enhanced retrieval performance on OCR data.

In this paper, we describe our approach to obtain enhanced retrieval performance on OCR data. In Section 2, we categorize various OCR errors, then match algorithms to construct and validate error-grams and correction rules, and finally use all this information in the retrieval process. Section 3 presents experimental results comparing the most efficient algorithms presented. The conclusions, discussion of open questions and future work directions are presented in Section 4.

2. The proposed method

In order to measure the mapping between the input scanned image and its corresponding OCR output, we need a distance function to solve the problem of proximity matching. We take into account that the edit distance defines a metric space on the set of text substrings. To illustrate the power of edit-distance matching, we use a large database that contains original texts. There is a match routine that detects any common segment between the original word and each of the OCR words. The output of the match routine is a distance that means the transformations rendering the two words identical. In the first step, we apply editing operations on OCR words, generating a collection of error-grams and correction rules. The second step uses query terms, error-grams, and correction rules to create searchable keywords, identify appropriate matching terms, and determine the improvement of retrieval process on our collection of document images. Figure 1 shows the algorithm of the retrieval process. The details of the algorithm related to Figure 1 are presented in the next sections. The proposed method is described as follows:

Given a scanned image,

1. Locate and extract text objects in the image;
2. Compare OCR-recognised text with original text. Mistakes are described as a set of error-grams and correction rules;
3. Increase elements of retrieval systems such as document ranking, recall, and precision.

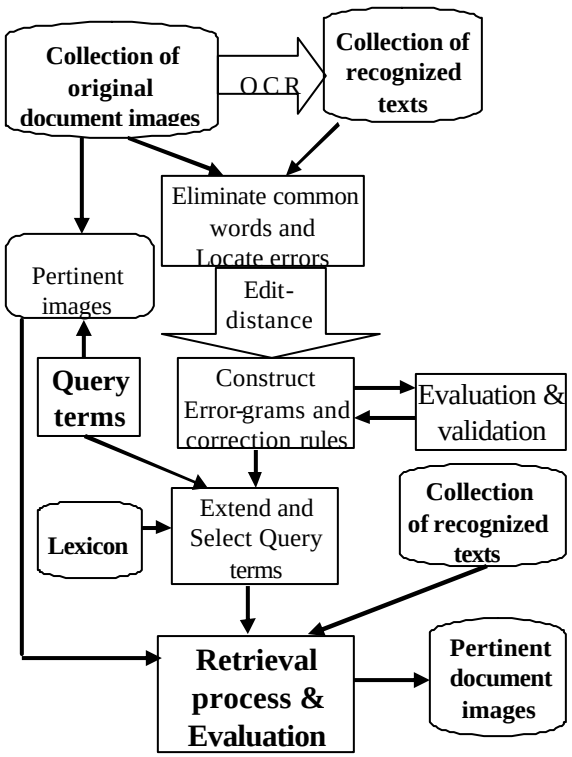

Figure1: Retrieval process based on OCR errors.

2.1 OCR errors

In this part of the experiment, a commercial OCR was used over the located text to perform character recognition. 528 315 words were constructed by mapping the extracted text back to the corresponding input images used in the previous experiment. The OCR engine makes some mistakes; see Table 1 for an overview of the error groups with examples.

Table 1: Error groups with examples

Error group	Correct word	Error example

Substitution	light	right
Deletion	Info	Nfo
Insertion	Kylie	Ikylie
Paste or Split	n-gram	n gram

2.2 Matching errors

Our hypothesis is that differences in the observed frequency between original input and recognized OCR output texts would indicate that the n-gram substring in question was incorrectly recognized. Since counting errors by hand is too costly in time, a simple error measure - Edit-distance - was adapted for the experiment. Edit-distance algorithm is based on dynamic programming and matches strings without lexicon or priori information. The distance between two words is the number of editing operations required to transform one of the words into the other.

Let M_{ori} be the set of words that contains the original document, and M_{ocr} is the set of words that contains the recognized document.

Let $s = e_1, e_2,\ldots,e_n$ be a sequence of edit operations for transforming a string x into another string y. The costs c(s) of this sequence are given by $c(s) = \sum_{i=1}^{n} c(e_i)$.

Given two strings x and y and given the costs of any edit operation which may be required for transforming x into y, we define the distance between x and y by

d(x,y) = min{c(s) : s is a sequence of edit operations which transforms x into y}.

In set notation, we have correctly recognized words $M_{rec} = \{words \hat{I} (M_{ori} \cap M_{ocr})\}$, and remaining words $M_{remo} = \{words \hat{I} (M_{ori} - M_{rec})\}$; $M_{remr} = \{words \hat{I} (M_{ocr} - M_{rec})\}$.

We show now how to adapt this measure to our algorithm to construct error-grams and correction rules in document images.

2.2.1 "edit-distance" algorithm [8]

The algorithm used to compute the edit-distance d() is based on dynamic programming. Its fills the matrix $D_{0..|x|,0..|y|}$, where $D_{i,j}$ represents the minimum number of operations to match strings $x_{1..i}$ to $y_{1..j}$, x is a string, |x| its length, and x_i is the i-th character of x. The costs relating to the editing operations are initialized to 1. We will present an iterative algorithm in section 2.2.3, modifying the costs gradually in order to improve the recognition.

$D_{i,0} = 0$; $D_{0,j} = 0$;

$D_{i,j}$ = if $(x_i = y_j)$ then $D_{i-1,j-1}$

else $1 + \min(D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1})$

Where at the end $D_{|x|,|y|} = d(x,y)$

2.2.2 Error-grams and correction-rules building

This algorithm treats the words which appear only in original documents M_{remo}. It uses edit-distance to find the nearest word in M_{remr}, and to locate the errors. Then, we verify the quality of pairing, extract the immediate predecessor and successor for each confused character, and classify the n-grams extracted by order depending on their occurrences. The algorithm consists of:

(i) For (each word ($x_i \hat{I} M_{remo}$)) {Scan words in M_{remr} and Select $x_j \hat{I} M_{remr}$ that $d(x_i,x_j)$ is the minimum obtained}.

(ii) Locate errors and verify the quality and the accuracy of matching.

(iii) For each recognition error, use characters below and above confused character to construct error-grams. Weights based on frequencies reflect the importance of those rules in the retrieval process.

The algorithm constructs 2822 error-grams. Table 3 shows the top 30 error-grams and correction-rules.

The correction rules contain the probabilities that any character A_i in document image can be regarded as B_j obtained by OCR, which is calculated using formula: $P(B_j/A_i) = (\#(A_iB_j) / \#A_i)$ where $\#(A_iB_j)$ is an occurrence of interchange, decomposition, or combination of original content A_i and recognized content B_j. We obtain results like *P(y/v); P(h/ln); P(d/cl); P(a l/al); and P(rul/mt)* for substitutions, insertions and deletions.

2.2.3 Edit-distance with automatic costs evaluation

In order to improve the recognition, we penalize at each loop, the cost of erroneous n-grams with high frequency. We use this iterative technique to generate the new matching; the goal is to increase the recognition of the original text. The costs are adjusted automatically until they became unchanged.

2.3. Retrieval process

IR is about finding the relevant information in a large text collection, and string matching is one of its basic tools. However, classical string matching is not enough for document images, because a word which is recognized incorrectly in the database cannot be retrieved anymore. For applications where it is desirable to find all occurrences of a particular term, there is the notion of exact string matching. When the data is noisy or corrupted, as the case with OCR text, exact string matching becomes inappropriate and another measure is needed to facilitate information retrieval on collections of OCR text. Conceptually, the retrieval system is composed of four modules:

1. Generate expanded query search terms.
2. Assign weights to the obtained list and select candidate terms.
3. Query collections of recognized document images.
4. Measure the performance of retrieval system and compare different methods.

2.3.1 Query expansion and selection

N-gram is an N-character slice of a character string. By treating a word not as a single unit but as a set of overlapping N-grams, this approach can partially overcome the problems mentioned above. A set *X* of expanded search term is generated by substituting all error-grams contained in the term by their corresponding correction. For example, if the word "light" is a term query, it is statistically uncertain because OCR confuses "i" with "l" and "c" with "e" etc. Thus, we generate 32 words. We need only the threshold value of the probability to decide whether the term should be included in the expansion or eliminated. For example, the percent occurrence greater than 0.01 generates the words below:

X=<light; llght; lieht; lighl; ligit; ligbt; lighd; iight; right >
% = <100; 0.22;0.09; 0.07; 0.05; 0.04; 0.02; 0.02; 0.01 >

A search refers here to a full-text search in which text is stored as a set of words. Prior to each search to retrieve an input query term, the n-grams and correction-rules generate multiple search terms as described in the example.

Finally, if the generated word exists in both the lexicon and the expanded term list, it will be deleted from the expanded query terms. Indeed, any word contained in the lexicon causes noise effects and confusion in the answers. For example, if the word "light" is a query term, "right" can be used as extended term and its uses harm within the meaning of the user request. The selection of the words to be used in the retrieval process depends on the word existence in the lexicon or not, which determines whether or not an extended term is judged to satisfy the input query.

2.3.2 Query collection and matching

For document representation, the most popular is vector-based model where each document is represented by a vector with each dimension being the existence of an indexed term. Various weighting scheme could then be adopted to approximate the 'importance' of a particular term. Given a large collection of documents, one is always confronted with the problem of locating the desired information. The task of text retrieval thus can be loosely described as effectively finding the documents which contain the information meet user's needs. This usually involves converting all documents and user's information need (query) into some internal representation ('indexing' documents and queries) and then matching the documents and the queries over the representations.

The exact matching consists of, given two strings P (the pattern) and T (the text) over a common alphabet, finding all of the occurrences of P in T. Some search engines will match on partial words that are found within a larger word. This is often referred as "word stemming". For example, with partial matching turned on, the word "program" would find a match within "programmer". However, words like "*companies*" will not always yield a match on *"company" since* "company" is not an exact "substring" of "companies".

2.3.3 Retrieval performances

Performance was determined based on the retrieval of 50 randomly selected words. All of our experiments are based on 979 images, which contain 499123 words (3 Mo of characters). We first extract the pertinent images of each query using original texts. Then we examine the effect of the expanded query list and retrieval condition. We compare expanded lists composed of 3-grams with lists produced by error-grams and correction rules. Finally we evaluate the performance of these methods based on retrieval effectiveness using average values and standard deviation of the rated recall and precision, which are calculated by using the following equations:

(i) $STDEV(s) = \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n(n-1)}}$

where n is the number of requests used and *x* is the rappel or precision obtained. The standard deviation is a measure of how widely values are dispersed from the average value (the mean).

(ii) RECALL is a measure of the ability of the system to present all relevant images. It's calculated by formula:

$$\frac{Total\ \ relevant\ \ images\ \ retrieved}{Total\ \ number\ \ of\ relevant\ \ images}$$

(iii) PRECISION is a measure of the ability of the system to present only relevant images. It's calculated by formula:

$$\frac{Total\ \ relevant\ \ images\ \ retrieved}{Total\ \ number\ \ of\ \ images\ \ retrieved}$$

(iv) We will also use the F-MEASURE [9] combines recall and precision in a single efficiency measure (it is the harmonic mean of precision and recall):

*F = 2 * (recall * precision) / (recall + precision)*

(v) Quality-Distance (QD) is used to measure the performance of the approach 3-gram overlaps: *QD(x,y) = G(x) + G(y) -4 -2*(G(x) + G(y))* where x, y are string to be matched. *G(x)* represents 3-grams overlaps x. Q-gram Distance (*QD*) is the number of n-grams contained in two words versus the number they share.

3. Experimental results and discussions

The document collection used in our experiments consists of research papers from Media Team document database (Washington University) with 979 scanned images. We present the evaluation of our results based on the original texts provided and the OCR texts produced.

The results obtained by using edit-distance are presented in table 2. In the original images, we have 614 non-text fields eliminated from the original texts, which explains the higher number of words present in the OCR texts. Note that we can improve recognition by reducing noise and using features acquired to distinguish text that is considered as noise in OCR words.

We obtained 6933 substitutions, 2216 deletions, and 2319 insertions. The output of the edit-distance algorithm will serve as input for the error rules building algorithm to construct the error-grams and the correction rules. The algorithm constructs 2822 error-grams and correction-rules. Table 3 shows the top 30 error-grams.

Table2: Texts recognition using Edit distance. 979 scanned images recognized by commercial OCR.

	Total words	**Total characters**	**% reco--gnized words**
Original image	499 123	2.9 Mo	
OCR extracts	528 315	3 Mo	
Common words	468 619	2.74 Mo	93.8 %
Distance < 3	5 185	30 591	1.03 %
Total Recognition	473 804	2.78 Mo	94.83%

It might be interesting to note that error statistics are strongly influenced by the number of occurrences. As the frequency increases, 3-grams are rarely used. This effect is illustrated in Figure 2, which plots the n-grams occurrence versus the set of n-grams ranked on the percent occurrence and grouped by 500. For n=3, 3-grams decrease more rapidly with the percent occurrence. For n=2, the number of different 2-grams increases from 165, for a top 500 (first block), to 300 for n-grams ranked between 1501 and 2000 (4-th block). For n=1, 1-gram errors are 5 on the top 500 (block 1) and 120 on the block 6. This is due to the frequencies of the 1-grams in our corpus.

Table3: Percent occurrence of the top 30 error-grams.
A_i: n-gram in the original word.
B_j: n-gram in the recognize word regarded as A_i
f: Percent occurrence that A_i can be regarded as B_j

A_i	B_j	*f*		A_i	B_j	*f*
plc	pic	100		ctl	cd	3.49
pct	pet	100		kG	cG	3
11b	1ib	100		1]	l]	2.92
1b	ib	42.85		ize	ise	2.83
AHD	AMD	25		dle	die	2.76
pc	pe	23.07		tz	lz	2.67
HD	MD	11.53		efin	enn	2.20
lo	lo	10		iz	is	2.16
acie	ade	8.69		ze	se	2.10
Sie	Sle	8.10		dl	di	2.03
ll	ll	7.5		efi	en	1.83
z	1	7.42		iza	isa	1.66
itz	ilz	5.76		Si	Sl	1.35
oic	olc	3.72		za	sa	1.34
ctly	cdy	3.52		zi	si	1.30

*f = P(Bj/Ai) * 100 = (#(AiBj) / #Ai) * 100*
#X represents the number of events of X.

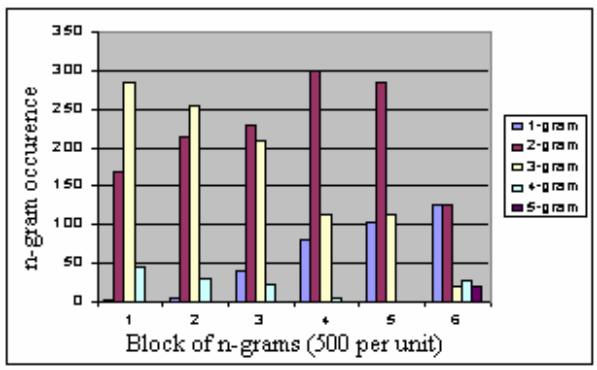

Figure 2: Distribution of n-grams errors.

In the retrieval process, performance was determined based on the retrieval of 50 randomly selected words. The experiments are based on the 979 corpus images. Our method is compared with exact and partial matching, as well as with Ukkonen's [5] Q-gram distance

(QD). The results obtained in table 4 show that our approach achieves an improvement in terms of recall and precision.

Table 4: Retrieval effectiveness in searching

Retrieval condition	Recall (%) value	s	Precision(%) value	s
Exact Matching	93.05	22.97	89.36	23.5
Partial matching	94.81	18.16	77.12	28.23
3-gram Overlaps				
Qgram (QD < 2)	92.10	23.74	68.71	31.29
Qgram (QD < 3)	97.09	9.08	55.67	30.01
Our approach				
error-gram substitution	99.08	2.45	91.22	13.28

When we use exact matching on the recognized texts, the recall and precision rates were 93.05% and 89.36%, respectively. The recall increased to 97.09 for 3-gram overlaps with Q-gram distance (QD) less than 3, but the precision decreased to 55.67%.

Our approach presents the best recall and precision. Furthermore, the standard deviation (s) shows that most of the examples in a set of queries are closer to the average than other methods.

Table 5 shows the results which combine recall and precision, equally weighted, in a single efficiency measure, F-measure, as described above.

Table 5: Efficiency measure of retrieval combining precision and recall.

Retrieval condition	F-measure
Exact Matching	91.16
Partial matching	85.06
3-gram Overlaps	
Q-gram (QD < 2)	78.70
Q-gram (QD < 3)	70.76
Our approach	
n-gram substitution	94.98

We observe that our approach, in comparison with other methods, achieves better overall retrieval effectiveness. This is due to the statistic's characteristics to extract and classify expanded words based on their importance, relatively to the confusions, in the training test.

4. Conclusion

This work presents an approach to process and improve retrieval of textual blocks contained in the composite document images. String processing in textual corpus is a very fertile and useful research area. Current OCR does not work well for poor quality or scanning document images. The proposed method collects frequent error-grams and correction-rules that can be used to extend query terms and to improve retrieval performance.

979 scanned document images from Media Team document database (Washington University) have been tested. Experimental results indicate a noticeable improvement of the retrieval effectiveness in comparison with exact, partial, and n-gram overlaps matching. Further research is undertaken actually to outperform our newly introduced approach.

References

[1] Y. Fataicha, M. Cheriet, J.Y. Nie, and C.Y. Suen. **Content analysis in document images: A Scale Space Approach**. Proceedings 16th International Conference on Pattern Recognition, Volume: 3, Page(s): 335 -338, 2002.

[2] C.Y. Suen. **N-Gram Statistics for Natural Language Understanding and Text Processing**. IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. PAMI-1, N° 2,p 164-171, April 1979.

[3] K. Taghva, E Stofsky. **OCRSpell: an interactive spelling correction system for OCR errors in text**. IJDAR, Vol. 8, no. 1, p. 125-137, 2001.

[4] M. Ohta, A. Takasu, J. Adachi. **Probabilistic retrieval methods for text missrecognized OCR characters**. IEEE Trans. PAMI, Vol. 22, no. 11, p. 1224-1240, 1998.

[5] S.M. Harding, W.B. Croft, and C. Weir. **Probabilistic retrieval of OCR degraded text using n grams**. Research and Advanced Technology for Digital Libraries: First Eupropean Conference,ECDL, Pisa, Italy, Springer lecture notes in Computer Science. N° 1324, 1997.

[6] Gonzalo Navarro, Ricardo Baeza-Yates, Erkki Sutinen and Jorma Tarhio. **Indexing Methods for Approximate String Matching**. *IEEE Data Engineering Bulletin* 24(4):19-27, 2001, Special issue on "Managing Text Natively and in DBMSs". Invited paper.

[7] Ricardo Baeza-Yates, Berthier Ribeiro-Neto. **Modern Information Retrieval**. Addison-Wesley, Paperback, Published May 1999, 513 page.

[8] E. Ukkonen. **On approximate string matching**. Proc. Int. Conf. on Foundations of Comp. Theory, Springer-Verlag, LNCS **158** p487-495, 1983.

[9] C. J. Van Rijsbergen, editor. Information Retrieval. Butterworths, London, 1979.