

Text Categorization of Low Quality Images

David J. Ittner*, David D. Lewis[†], and David D. Ahn[‡]

AT&T Bell Laboratories; 600 Mountain Ave.; Murray Hill, NJ 07974

Appears (same pagination) in Fourth Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, NV, April 1995. ISRI; Univ. of Nevada, Las Vegas, pp. 301–315.

Abstract

Categorization of text images into content-oriented classes would be a useful capability in a variety of document handling systems. Many methods can be used to categorize texts once their words are known, but OCR can garble a large proportion of words, particularly when low quality images are used. Despite this, we show for one data set that fax quality images can be categorized with nearly the same accuracy as the original text. Further, the categorization system can be trained on noisy OCR output, without need for the true text of any image, or for editing of OCR output. The use of a vector space classifier and training method robust to large feature sets, combined with discarding of low frequency OCR output strings are the key to our approach.

1 Introduction

Text categorization is the automated assignment of texts to predefined categories. The classic application of text categorization is in assigning controlled vocabulary terms (e.g. Dewey Decimal numbers, *Com-*

puting Reviews categories) to documents to aid in their retrieval or routing [4, 14]. Text categorization can also aid other language processing tasks such as information extraction [21], word sense disambiguation [12], and even handwriting recognition [27].

This paper addresses the automated categorization of text *images*, based on the output of optical character recognition (OCR) applied to the image. Several recent studies have suggested that OCR output is quite adequate for the related task of text retrieval, at least when clean images are available [10, 32, 33].

Unfortunately, in practice not all images are clean. Even with high-resolution scanners, OCR systems are constantly faced with poor-quality images due to light originals, photocopies, poor contrast, etc. Facsimile machines are increasingly popular and produce images which are particularly noisy: the digitization resolution is coarse, skew may be non-linear, scanlines are dropped due to communication errors, and so on. State of the art OCR does not provide accurate transcription on images of such quality [24].

The errors present in OCR-produced text pose the same problems for text categorization that they do for text retrieval [33]. Crucial words may be garbled, noise strings may distort statistical weighting for-

**dji@research.att.com*

[†]To whom correspondence should be addressed.
lewis@research.att.com

[‡]Current address: Harvard University; Department of Computer Science; Cambridge, MA 02138.

mulas, and so on. In text retrieval, however, the user query provides at least a few strings that can be assumed to be good content indicators and likely to occur in relevant documents. The user query can compensate for many problems in a text representation [20]. Conversely, when analogous information is not available in a text categorization context, OCR errors become a particular problem.

On the other hand, research using artificial data has suggested that high error rates can be tolerable for text retrieval [31], and there has been some success at categorizing continuous speech data, where high word recognition error rates are currently inevitable [26]. In addition, Hoch [15] has presented an experiment in which images of 42 German business letters were categorized with 57% accuracy into one of six categories. While this is a promising initial result, a small data set and manual intervention at several points in the training process make the results difficult to interpret.

We show in this paper that at least for one data set low quality images can be categorized almost as accurately from OCR output as when the original text is available. Further, we show that the categorization system can, and indeed should, be trained without access to the original document text.

Section 2 describes the text categorization system and its training algorithm; section 3 outlines the OCR system used. Section 4 describes the experiment design, data set, and evaluation process. Section 5 presents results, and section 6 provides analysis. Finally, section 7 lists conclusions and areas for future work.

2 Our Text Classifiers and Training Algorithm

The implementer of a text categorization system is typically not its main user, and

may not be an expert in the categories to be assigned. Conversely, category experts may not know how to build a good categorizer. The use of machine learning to automatically produce categorizers from documents categorized by experts has therefore received much attention. Large quantities of categorized documents may be available from past manual categorization [4], or experts may be asked to categorize a small number of pivotal documents [19].

Our approach in this study was to train a single binary (yes/no) classifier for each category of interest. In this section, we describe how documents are represented in our categorization system and then describe the design and training of classifiers that can be applied to these document representations. The classifiers consist of two parts: a prototype or ideal document to which documents are compared, and a function which transforms a document's similarity to the prototype into an estimate of the probability the document belongs to the category of interest.

2.1 Document Representation

We represent both training and test documents as vectors of numeric weights, i.e.:

$$\langle w_{i1}, w_{i2}, \dots, w_{ik}, \dots, w_{it} \rangle$$

where w_{ik} is the weight given the k th indexing term in the i th document, and t is the number of indexing terms being used. An indexing term may be a word, phrase, character ngram, or other linguistic entity. In this paper we will be using single words as indexing terms, with the words being drawn either from the original text of a document or from OCR output. The weight that a term takes on for a particular document can be a function of the number of times the term occurs in that document, the number of documents the term occurs in, and other information.

Of the variety of weighting methods possible, we used the Cornell “lrc” weighting commonly used with the vector space model of text retrieval [5, 6]:

$$w_{ik} = \frac{tf_{ik} \times \log(N_D/n_k)}{\sqrt{\sum_{j=1}^t (tf_{ij} \times \log(N_D/n_j))^2}}.$$

Here N_D is the number of documents in the training set, n_k is the number of documents in which term k appears, and tf_{ik} is:

$$tf_{ik} = \begin{cases} 0 & \text{if } f_{ik} = 0 \\ \log(f_{ik}) + 1 & \text{otherwise} \end{cases}$$

where f_{ik} is the number of occurrences of term k in document i .

2.2 A Prototype Classifier and Its Training

A wide variety of classifier types and training methods have been proposed for text categorization, but most have been tested on only one data set. We chose instead to adapt a classifier form and training method widely used in text retrieval. In vector space retrieval [28, pp. 313–319], a class is represented by a single idealized relevant document or *prototype* [11], [30, Ch. 5]. In other words, the classifier has the same form (a vector of term weights) that documents do, and classification is done by measuring the similarity of test documents to the classifier.

To produce the prototype for each class, we used Rocchio’s algorithm [13], [25], [28, pp. 319–321]. Rocchio’s algorithm specifies that the weight of term k in the prototype Q_c for class c should be

$$w_{ck} = \begin{cases} w'_{ck} & \text{if } w'_{ck} > 0 \\ 0 & \text{otherwise} \end{cases}$$

where

$$w'_{ck} = \beta \frac{1}{|R_c|} \sum_{i \in R_c} w_{ik} - \gamma \frac{1}{|\bar{R}_c|} \sum_{i \in \bar{R}_c} w_{ik}.$$

R_c is the set of training documents belonging to class c and \bar{R}_c are the documents not belonging to class c . The parameters β and γ control the relative impact of positive and negative examples on the classifier. We used the standard values $\beta = 16$ and $\gamma = 4$ [6]. The full Rocchio formula also takes into account terms suggested by a human user, but we omit this component.

Other approaches to constructing a prototype-based classifier for text categorization have been proposed [35, 38]. Our reason for choosing Rocchio’s algorithm was based on its ability to work well without sophisticated feature selection, in contrast to many learning algorithms used for relevance feedback [6]. This was an advantage in our experiments, since we were dealing with a number of large and poorly understood sets of indexing terms.

2.3 Converting Similarities to Probabilities

Classification in the vector space model is done by computing the similarity between the prototype Q_c and test document D_i using the dot product formula:

$$S(D_i, Q_c) = \sum_{k=1}^t (w_{ik} * w_{ck})$$

In using a vector space classifier for text retrieval, the similarity of each of a set of documents to the prototype is computed, the documents are ranked by similarity, and the most similar documents are displayed to the user of the text retrieval system.

In text categorization, however, documents cannot simply be ranked, with the user left to decide how far down the ranked list to go. Instead a strict decision must be made for each document as to whether or not it belongs to the category of interest. (This fact has unfortunately been ignored in many evaluations of text categorization systems.)

Making strict classification decisions is aided if the classifier produces not just an arbitrary similarity value but an actual estimate of $P(c|D_i)$, the probability that document D_i belongs to category c . While the vector space similarity is not a probability, we might expect it to be a good predictor of the probability of category membership.

Therefore our strategy was to first build a prototype classifier Q_c as described above. We then found the similarity of each training document to that prototype. This gave us pairs of the form

$$\langle S(D_i, Q_c), \text{class label} \rangle$$

where class label is 1 if the instance is a category member and 0 if not. We then applied logistic regression [1, 19] to these pairs. The output of the logistic regression are parameter values a_c and b_c such that:

$$\frac{e^{(a_c + b_c \times S(D_i, Q_c))}}{1 + e^{(a_c + b_c \times S(D_i, Q_c))}}$$

is a good approximation to $P(c|D_i) = P(c|S(D_i, Q_c))$, the conditional probability that a document belongs to class c , given that its similarity to the prototype for class c is $S(D_i, Q_c)$.

The hybrid vector-logistic classifier is used to categorize new documents as follows. Each document is converted to a vector of weights, w_{ik} , one for each indexing term being used. The weights incorporate the values of N_D/n_k from the full set of training documents. The similarity $S(D_i, Q_c)$ of the document vector to the category prototype is then computed, and then this similarity is transformed using the logistic function with parameters a_c and b_c into an estimate of $P(c|D_i)$. This estimate is then used to decide whether or not to assign the category to the document, as described in Section 4.4.

3 The OCR System

For our testing we used the experimental page reader described in [2]. The relevant stages of the page reader are:

1. *geometric layout analysis*: skew-correction, segmentation into text blocks, line finding within blocks, and character finding within lines. The underlying algorithms are described in [16].
2. *symbol recognition*: classification of symbols by shape, inference of text size and baseline, segmentation of lines into words by spacing, and shape-directed resegmentation to handle touching and broken characters. This produces, for each space-delimited set of characters, potentially multiple segmentations into symbols, with each symbol labeled with a list of interpretations and confidence value.
3. *contextual analysis*: application of language models to resolve ambiguity implied by alternative segmentations and interpretations. The methods are all data-directed; they merely select among alternatives generated by shape recognition.

Since the only text/non-text filter used during layout analysis is based on the physical size of a connected component, artwork which is actually noise or graphics may be treated as text. We took no special precautions for the poor quality images of interest here.

The methods used to construct the symbol classifier are described in [3]. The image training data, entirely synthetic at a resolution of 200x200dpi, included 25 commonly occurring font styles of the printable ASCII character set.

Unless otherwise stated, the contextual analysis stage consisted primarily of

simple typographic morphology and spell-checks. In particular, if a word interpretation spelled, it was promoted to top choice. If no alternatives spelled, then the top-choice remains, based only on shape recognition confidence. The programmable interface of the system made it particularly easy to vary the analysis and get at the internal data structures in a convenient way [17].

4 Experiments

Our main hypothesis was simply that fax quality images could in fact be categorized almost as accurately as the original text. This belief was motivated by the relatively good results of recent studies of text retrieval on OCR text. While text retrieval has the advantage of the user query to define some trusted features to look for, categorization has the advantage that, when errors are made systematically, these error-prone features can still be good predictors.

A second hypothesis was that a fax categorization system could be trained on OCR output, without access to any original document text. We were motivated here by a number of results in pattern recognition and machine learning (e.g. [23]) that suggest that when feature noise is present in test data, learning is actually more effective when the noise is present in training data as well. Training on OCR output would be a considerable advantage in setting up a fax categorization system, since the original text may not be available and post-editing is expensive.

Our final hypothesis was that deleting strings that appeared in the OCR output for only a few documents would improve categorization accuracy. Such strings are likely to be OCR errors and their presence can drown out the effect of better predictor words. While some low frequency strings that are legitimate words will be removed

as well, the fact that they are low frequency means their removal will have little impact on effectiveness. Note that removing low frequency strings produces text which is unreadable by people, so our hypothesis here is that the best representation for humans is not the best for a text categorization system.

The rest of this section describes how we tested these hypotheses, including the overall design of the experiments, the data set used, and our evaluation measures.

4.1 Experiment Design

The basic design of the experiments was as follows. A set of fax quality page images was separated into a training set and test set. We ran our page reader on all the images, producing its usual output text. We then modified the output text for both training and test pages by first removing stopwords, and then automatically removing strings which occurred in fewer than k training pages. (Avoiding training on the test data is critical to getting an accurate estimate of classification effectiveness, so we were careful to avoid referring to the test data even in determining the high frequency words.) A text classifier for each category was then trained on the training pages, and then the set of classifiers was tested on the test pages.

4.2 Data Set

Our data set was taken from the database of pages from technical journals and reports distributed on CD-ROM by the University of Washington [22]. The CD contains images and corresponding text for 1000 pages of English-language reports from a wide range of disciplines.

Categories were automatically assigned to each page based solely on the journal from which it was taken. The assignment of journals into categories was

taken from *Ulrich's International Periodicals Directory*[36], which provides a classification of newspapers, reports, and periodicals by subject. This was done to provide an objective, replicable set of categories. In an operational system, an arbitrary set of categories could be defined and a trainer could use an interface to assign images to categories. Note that in our experiments the image zones containing the name of the journal were not made available to the text categorization system.

After removing images of duplicate pages from the initial set of 1000 (e.g. the database contains an original and a photocopy), and removing those few journals not listed in the periodicals directory, we were left with 614 pages assigned to 73 overlapping categories. On average, each page was assigned to 1.6 categories with a maximum of 4. The largest category, *Computers*, contained 76 pages; six categories were assigned to only 2 pages (e.g. *Law* and *Paints and Protective Coatings*). Pages within a category were then split at random into a training set and a test set at a ratio of 4-to-1, while making certain that at least one page from each category was represented in both the training and test sets. This produced a training and test set of 480 and 134 pages, respectively.¹

The mean number of non-stopwords in the text of training set documents was 224 with a standard deviation of 123. The mean number of non-stopwords in the text of test set documents was 232 with a standard deviation of 125; the minimum number was 10, the maximum 534.

The CD holds 300 dpi images for each of these pages; the image quality varies as a result of poor originals, photocopying, low scanner contrast, etc. In order to simulate facsimile images, we further downsampled each of the page images to 200 dpi

¹Details of the training/test set split, category assignment, and other information are available from the authors.

in the horizontal direction and 100 dpi in the vertical direction (so-called “standard” G3 facsimile resolution). (In an informal attempt to validate the downsampled data set, we compared our OCR results to those obtained on a separate data set of 100 true facsimile images. The recognition rates and detailed confusion matrices were comparable between the data sets, providing evidence that the downsampled set is representative of facsimile images, at least for our purposes.)

In order to assess the impact of OCR error on categorization, we separately measured the per-character accuracy for each page of the test set, sorted them by accuracy, and split the list into 3 equal groups. Group A had the highest character accuracy, ranging from 87.0% to 96.1%. The OCR accuracy on group B ranged from 77.9% to 86.9%. Group C had the lowest character accuracy, ranging from 30.6% to 77.8%; the median accuracy of the group was 71.0%. Figure 1 gives a sample from the median image in each group, with the corresponding OCR output. (Note that these accuracy results were obtained on zoned images, so they don't include errors from incorrect zoning, introduction of spurious text due to line art, halftones, etc.)

4.3 Evaluation

The effectiveness measures we use are based on a contingency table (Figure 2) describing system behavior on a test set. Each entry in the table specifies the number of categorization decisions with the specified result. For instance, a is the number of times the system decided Yes (the category should be assigned), and Yes was in fact the correct answer.

The most commonly used effectiveness measures for text categorization are recall (here the proportion of category members assigned to the category) and precision (the proportion of documents assigned to the

The weighting scheme is based on conditions and activities. Since the results sensitivity of the results to the weights progress and further refinements on the element into the weighting scheme, are

The weighting scheme is based on experts' conditions and activities. Since the results sensitivity of the results to the weights progress and further refinements on the element into the weighting scheme, are

Three-dimensional scenes are an extension of two-dimensional ones. For the extension to three dimensions, stages 1 and 2 should be replaced by a method to find the

Three-dimensional scenes are an extension of two-dimensional ones. For the extension to three dimensions, stages 1 and 2 should be replaced by a method to find the

This is an inverted S-type function. I negative slope. We have selected these fuzzy membership functions. For each sometimes simply call them cases (a normal, p.w. linear, ridge-shaped and

This is an inverted S-type function. I negative slope. We have selected these fuzzy membership functions. For each sometimes simply call them cases (a normal, p.w. linear, ridge-shaped and

Figure 1: Top: Portion of a page image in Group A, those with highest OCR accuracy; directly below is OCR output. Middle: Portion of an image in Group B; directly below is OCR output (note the higher error rate and missed word break). Bottom: Portion of an image in Group C, those with worst OCR accuracy; directly below is OCR output (note low word accuracy).

	Yes is Correct	No is Correct	
Decides Yes	a	b	$a + b$
Decides No	c	d	$c + d$
	$a + c$	$b + d$	$a + b + c + d = n$

Figure 2: Contingency Table for a Set of Binary Decisions

category that did in fact belong to the category). In terms of the contingency table entries we have:

$$(1) \text{ recall} = a / (a + c)$$

$$(2) \text{ precision} = a / (a + b)$$

An ideal system would have both recall and precision of 1.0. Real systems typically fall below this level, but can be adjusted to show any of several tradeoffs between recall and precision.

A single measure of system effectiveness which takes into account the relative value of recall and precision to the system's users is the F-measure (1.0 minus Van Rijsbergen's E-measure [37, pp. 168–176]):

$$F_{\beta} = \frac{(\beta^2 + 1)a}{(\beta^2 + 1)a + b + \beta^2 c} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \quad (1)$$

where P is precision and R is recall. The parameter β ranges between 0 and infinity. A β of 1 corresponds to equal weighting of recall and precision, so F_1 is a single measure which gives equal weight to recall and precision. β 's less than 1 give more weight

to precision, and those greater than 1 give more weight to recall.

In addition to evaluating the effectiveness of a system in assigning a particular category, we would like measures of average effectiveness in assigning a set of categories. Two main methods, microaveraging and macroaveraging, have been used for this purpose in text retrieval [34]. In *microaveraging* the contingency tables for the individual categories are added cellwise, and then the effectiveness measures such as recall, F_{β} , and so on are computed. In *macroaveraging* effectiveness measures are computed separately for each category and averaged over the categories. Microaveraged scores treat each document equally and are heavily affected by the performance of the system on frequently present categories. Macroaveraged scores treat each category equally, though frequent categories still have somewhat more impact on effectiveness, through their typically higher effectiveness values. We present both microaveraged and macroaveraged values.

4.4 Optimization

Once an effectiveness measure is chosen, we want to set up our classifiers to assign documents in such a fashion that their expected effectiveness is the best possible. Of the six effectiveness measures we present (micro- and macroaveraged recall, precision, and F_1) we chose to optimize macroaveraged F_1 . This can be done by optimizing F_1 individually for each category. Optimization of F_1 for a category was done by first sorting the test documents by our classifier's estimate of $P(c|D_i)$. We then computed an approximate expected value of F_1 for putting no documents in the category, putting the top document in the category, putting the top two documents in the category, and so on, until the maximum expected F_1 was found. Details on estimating, approximating, and optimizing effectiveness measures for binary

classification are presented elsewhere [18].

5 Results

Our two main hypotheses were that effective categorization of low quality images was possible, and that training of categorizers could be done even if no source of ground truth for the text was available. Table 1 presents the relevant data, showing the average categorization effectiveness when the training of and/or the use of categorizers is on the raw OCR output, in contrast to the actual document text.

Since OCR accuracy varies dramatically with image quality, in Table 2 we break out the data of Table 1 by our three image quality groups. Unsurprisingly, categorization effectiveness declines with decreasing OCR accuracy.

We also hypothesized that, for the purposes of text representation, the quality of OCR output would be improved by discarding word types that occurred in few training instances. Table 3 contrasts the data from Table 1 with corresponding results when types that occurred in only one training text, or in only one or two training texts, were discarded from both training and test texts.

6 Analysis

To begin with, the categorization effectiveness of our hybrid vector-logistic classifier appears quite good given the small training set size, though of course only comparison with other algorithms will show how it stacks up in the range of text categorization possibilities.

Our first hypothesis, that content-based categorization of fax quality images is possible, was strongly supported. As shown in Table 1, macroaveraged $F_{\beta=1}$ declined only 6% from the text/text condition (training and testing on text) to the OCR/OCR con-

dition (training and testing on OCR). Not only is training without access to any original text possible, it is in fact desirable, since the decline in effectiveness was three times greater (17%) when training was on text but testing was on OCR.

Unsurprisingly, the difference between the text/text and OCR/OCR conditions was greatest (18%) for the subset of images with the highest rate of OCR error, as shown in Table 2. Still, effectiveness is surprisingly good given the extremely poor quality of some input.

The results in Tables 1 and 2 are based on discarding from the training and test documents those words that appear in the representations of fewer than 3 training documents. Table 3 shows, as predicted, that discarding words which occur in only one or even two training documents improves effectiveness. In comparison to the effectiveness of the best text/text condition, the penalty incurred for the OCR/OCR condition is reduced from 18% to 13% under microaveraged evaluation, and from 10% to 6% for the macroaveraged evaluation.

The impact of the frequency restriction can also be seen by noting that there are 50,710 distinct words in the training set produced from OCR, but only 11,437 that appear in two or more documents, and only 6,655 that appear in three or more documents. This is not too far from the 4,939 words which occur in three or more training documents when the actual text is used. The OCR errors that occur in three or more training documents tend to be small distortions of high frequency words, e.g. *computer*, *cornputer*, *computer*, *cumputer*. Those that occur in only one document are less well behaved, e.g. *acrnagnuifng*, *aaeaaooooeae*, *zvigalil*.

train (480 pages)	test (134 pages)	microaveraging			macroaveraging		
		recall	precision	$F_{\beta=1}$	recall	precision	$F_{\beta=1}$
text	text	.51	.63	.57	.53	.73	.58
text	ocr	.37	.63	.47 (-18%)	.42	.68	.48 (-17%)
ocr	ocr	.43	.66	.53 (-7%)	.48	.71	.54 (-6%)

Table 1: Average effectiveness of categorization when OCR vs. actual document text is used. In each case, the feature set is all words that that occurred in the representations of 3 or more training documents. Figures for both microaveraging and macroaveraging over 73 categories are shown.

train	test	microaveraging			macroaveraging		
		recall	precision	$F_{\beta=1}$	recall	precision	$F_{\beta=1}$
full (480 pages)	Group A (35 pages)						
text	text	.59	.41	.48	.33	.38	.33
text	ocr	.48	.34	.40 (-17%)	.26	.34	.27 (-19%)
ocr	ocr	.54	.35	.42 (-12%)	.30	.38	.32 (-5%)
full (480 pages)	Group B (35 pages)						
text	text	.64	.42	.51	.41	.45	.42
text	ocr	.54	.38	.44 (-13%)	.37	.46	.39 (-7%)
ocr	ocr	.59	.44	.50 (-1%)	.38	.46	.40 (-3%)
full (480 pages)	Group C (34 pages)						
text	text	.66	.41	.50	.40	.45	.41
text	ocr	.37	.24	.29 (-42%)	.26	.36	.28 (-31%)
ocr	ocr	.48	.30	.37 (-26%)	.34	.36	.33 (-18%)

Table 2: Average effectiveness of categorization when test set (but not training set) is split into classes based on OCR quality. In each case, the feature set is all words that that occurred in the representations of 3 or more training documents. Figures for both microaveraging and macroaveraging over 73 categories are shown.

train (480 pages)	test (134 pages)	microaveraging			macroaveraging		
		recall	precision	$F_{\beta=1}$	recall	precision	$F_{\beta=1}$
text-w3+	text-w3+	.51	.63	.57 (-7%)	.53	.73	.58*
text-w3+	ocr-w3+	.37	.63	.47 (-23%)	.42	.68	.48 (-17%)
ocr-w3+	ocr-w3+	.43	.66	.53 (-13%)	.48	.71	.54 (-6%)
text-w2+	text-w2+	.55	.56	.55 (-10%)	.54	.73	.58 (-0%)
text-w2+	ocr-w2+	.38	.59	.46 (-25%)	.44	.68	.49 (-16%)
ocr-w2+	ocr-w2+	.46	.63	.53 (-13%)	.48	.72	.54 (-7%)
text-w1+	text-w1+	.54	.72	.61*	.52	.76	.57 (-2%)
text-w1+	ocr-w1+	.34	.74	.46 (-25%)	.40	.73	.48 (-17%)
ocr-w1+	ocr-w1+	.36	.81	.50 (-18%)	.43	.79	.52 (-10%)

Table 3: Average effectiveness of categorization when words with few occurrences in training documents are discarded. Results for a minimum of one (w1+), two (w2+), and three (w3+) training documents with the word are presented. Percentage reductions in effectiveness are in comparison with the best text/text condition (*'ed) for each of the two methods of averaging over categories. Figures for both microaveraging and macroaveraging over 73 categories are shown.

7 Discussion

Our results are based on a single small dataset, so replication on a different and larger data set is a priority for us. It is worth pointing out that page images drawn from scientific journals have many more content words than, for instance, the average letter sent by fax. Whether effective categorization can be done of images that are less dense in words is an open question.

On the other hand, much can be done to improve on the methods presented here. Throwing out words is only the crudest way to produce a better text representation for categorization. In particular, we treated our page reader as a black box producing a single interpretation for each section of the input image. While this may be appropriate for producing readable text it is not necessary, and probably not optimal, for producing a representation to be categorized. We could instead output at each point multiple word hypothesis, say those whose confidence is over a threshold. The

confidence associated with each hypothesis could be accounted for in a probabilistic indexing model [9]. Nor is it necessary to output words. Other possibilities includes character n-grams [7] and word shape tokens [29].

On the text categorization side, we have tried only one, albeit plausible, approach to categorizing low quality images. Other methods should be tried, particularly since OCR output is poorly understood representation from a text classification standpoint.

8 Summary

We have shown that subject categorization of fax quality images can be done with little degradation in accuracy from categorization of the original text. Key to this accuracy is the realization that the text representation used for categorization need not be one that would be best or even appropriate for human reading. Users can interact with the document images, both when training categorizers and when using au-

tomatically categorized documents. The OCR output need never be seen; it can be merely a data structure aiding an image-centered approach to document processing [8]. The ability to categorize text images should be a useful component in a variety of document processing systems.

Acknowledgement

We thank Henry Baird and Tin Ho for helpful comments on this paper.

References

- [1] A. Agresti. "Categorical Data Analysis", John Wiley, New York, 1989.
- [2] H. Baird. "Anatomy of a Versatile Page Reader", *Proceedings of the IEEE*, 80(7):1059–1065, July 1992.
- [3] H. Baird and R. Fosse. "A 100-Font Classifier", *1st Int'l Conference on Document Analysis and Recognition*, pp. 332-340, St. Malo, France, 1991.
- [4] P. Biebricher, N. Fuhr, G. Lustig, M. Schwantner, and G. Knorz. "The Automatic Indexing System AIR/PHYS—From Research to Application", *11th Int'l ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 333–342, 1988.
- [5] C. Buckley, G. Salton, and J. Allan. "Automatic Retrieval with Locality Information Using SMART". In D. K. Harman, ed., *The First Text REtrieval Conference (TREC-1)*, pp. 59–72. NIST Special Publication 500-207, March 1993.
- [6] C. Buckley, G. Salton, and J. Allan. "The Effect of Adding Relevance Information in a Relevance Feedback Environment", *17th Int'l ACM/SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, June, 1994.
- [7] W. Cavnar and J. Trenkle. "N-Gram-Based Text Categorization", *Symposium on Document Analysis and Information Retrieval*, pp. 171–179, Las Vegas, Nevada, April, 1994.
- [8] K. Church, W. Gale, J. Helfman, and D. Lewis. "Fax: An Alternative to SGML", *16th International Conference on Computational Linguistics*, pp. 525–529, Kyoto, Japan, August, 1994.
- [9] W. Croft. "Experiments with Representation in a Document Retrieval System", *Information Technology: Research and Development*, 2:1–21, 1983.
- [10] W. Croft, S. Harding, K. Taghva, and J. Borsack. "An Evaluation of Information Retrieval Accuracy with Simulated OCR Output", *3rd Symposium on Document Analysis and Information Retrieval*, pp. 115–126, Las Vegas, Nevada, April, 1994.
- [11] M. de la Maza. "A Prototype Based Symbolic Concept Learning System", *Machine Learning: Proceedings of the Eighth International Workshop on Machine Learning*, pp. 41–45, 1991.
- [12] W. Gale, K. Church, and D. Yarowsky. "A Method for Disambiguating Word Senses in a Large Corpus", *Computers and the Humanities*, 26:415–439, 1993.
- [13] D. Harman. "Relevance Feedback and Other Query Modification Techniques" In *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall, Englewood Cliffs, NJ, 1992, pp. 241–263.
- [14] P. Hayes and S. Weinstein. "CONSTRUE/TIS: A System for Content-Based Indexing of a Database of News

- Stories”, *2nd Conference on Innovative Applications of Artificial Intelligence*, 1990.
- [15] R. Hoch. “Using IR Techniques for Text Classification in Document Analysis” *17th Int’l ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 31–40, Dublin, Ireland, June, 1994.
- [16] D. Ittner and H. Baird. “Language-Free Layout Analysis”, *2nd Int’l Conference on Document Analysis and Recognition*, pp. 336-340, Tsukuba Science City, Japan, October, 1993.
- [17] D. Ittner and H. Baird. “Programmable Contextual Analysis”, *IAPR Workshop on Document Analysis Systems*, pp. 77-92, Kaiserslautern, Germany, 1994.
- [18] D. Lewis. “Evaluating and Optimizing Autonomous Text Classification Systems”. Submitted to Eighteenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR-95).
- [19] D. Lewis and W. Gale. “A Sequential Algorithm for Training Text Classifiers”, *17th Int’l ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 3–12, Dublin, Ireland, June, 1994.
- [20] D. Lewis and K. Sparck Jones. “Natural Language Processing for Information Retrieval”, *Communications of the ACM*, to appear.
- [21] D. Lewis and R. Tong. “Text Filtering in MUC-3 and MUC-4”, *4th Message Understanding Conference (MUC-4)*, Los Altos, CA., pp. 51–66, June, 1992.
- [22] I. Phillips, S. Chen, and R. Haralick. “CD-ROM Document Database Standard”, *2nd Int’l Conference on Document Analysis and Recognition*, pp. 478-483, Tsukuba Science City, Japan, October, 1993.
- [23] J. Quinlan. “The Effect of Noise on Concept Learning”, In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, eds., “Machine Learning. An Artificial Intelligence Approach. Volume II”, pp. 149–166, Morgan Kaufmann, Los Altos, CA, 1986.
- [24] S. Rice, J. Kanai, and T. Nartker. “The Third Annual Text of OCR Accuracy”, *UNLV Information Science Research Institute Annual Report*, 1994.
- [25] J. Rocchio, Jr. “Relevance Feedback in Information Retrieval”, In *The SMART Retrieval System: Experiments in Automatic Document Processing*, Prentice-Hall, Inc, 1971, pp. 68–73.
- [26] R. Rose. “Techniques for Information Retrieval from Speech Messages”, *The Lincoln Laboratory Journal*, 4(1):45–60, 1991.
- [27] T. Rose and L. Evett. “Text Recognition using Collocations and Domain Codes”, Workshop on Very Large Corpora, pp. 65–73, Columbus, OH, June 1993. Association for Computational Linguistics.
- [28] G. Salton. “Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer”, Addison-Wesley, Reading, MA, 1989.
- [29] P. Sibun and A. Spitz. “Language Determination: Natural Language Processing from Scanned Document Images”, *Fourth Conference on Applied Natural Language Processing*, pp. 15–21, Stuttgart, Germany, 1994.

- [30] E. Smith and D. Medin. “Categories and Concepts”, Harvard University Press, Cambridge, MA, 1981.
- [31] S. Smith and C. Stanfill. “An Analysis of the Effects of Data Corruption on Text Retrieval Performance”, Technical Report DR90-1, Thinking Machines Corporation, December, 1988.
- [32] K. Taghva, J. Borsack, and A. Condit. “Results of Applying Probabilistic IR to OCR Text”, *17th Int’l ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 202–211, Dublin, Ireland, June, 1994.
- [33] K. Taghva, J. Borsack, A. Condit, and S. Erva. “The Effects of Noisy Data on Text Retrieval”, *Journal of the American Society for Information Science*, 45:50–58, 1994.
- [34] J. Tague. “The pragmatics of information retrieval experimentation.” In *Information Retrieval Experiment*, Butterworths, London, 1981, pp. 59–102.
- [35] T. Tokunaga and M. Iwayama. “Text Categorization based on Weighted Inverse Document Frequency”. Technical Report 94-TR0001, Dept. of Computer Science, Tokyo Institute of Technology, March, 1994.
- [36] “Ulrich’s International Periodicals Directory”, published by R.R. Bowker, Reed Reference Publishing Company.
- [37] C. van Rijsbergen. “Information Retrieval”, 2nd edition, Butterworths, London, 1979.
- [38] Y. Yang. “Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval”, *17th Int’l ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 13–22, Dublin, Ireland, June, 1994.

Appendix

The stoplist used in these experiments combined two public stoplists, one from the FreeWAIS distribution, and one used in early Cornell University SMART project experiments with the Time collection, along with some additional words appropriate to a data set of journal pages (names of days and months, and a few others). The resulting stoplist is reproduced below:

a about above according across actually add
 added adj after afterwards again against ago al
 all almost alone along already also although al-
 ways am among amongst an and another any
 anyhow anyone anything anywhere apr april are
 around as asked at aug august

b back bad be became because become be-
 comes becoming been before beforehand began
 begin beginning behind being below beside be-
 sides best better between beyond big biggest bil-
 lion both brought but by

c call called came can cannot caption cent
 co come complete continued could

d day dec december decided declared de-
 spite did do does down during

e each early eg eight eighty either else else-
 where end ending enough entire ep et etc even
 ever every everyone everything everywhere ex-
 cept

f face faced fact failed far feb february fell
 few fifty fig finally find first five for former for-
 merly forty found four fri pfriday from further

g gave get give given go going good got

h had has have having he held hence her
 here hereafter hereby herein hereupon hers her-
 self him himself his hour hours how however
 hundred

i idea ie if in inc including indeed instead
 into is it its itself

j jan january journal journals jul july jun
 june

k keep know known knows

l lack last later latter latterly least led less
 let like likely little long longer look lot ltd

m made magazine magazines make makes
 making man many mar march matter may
 maybe me means meantime meanwhile men
 might miles million miss moment mon monday
 month months more moreover morning most
 mostly mr mrs much must my myself

n named namely near nearly necessary need
needed needs neither never nevertheless next
night nine ninety no nobody none nonetheless
noone nor not note nothing nov november now
nowhere

o oct october of off often on once one only
onto or other others otherwise our ours our-
selves out outside over overall own

p page part past per perhaps place point
proved put

q qm question

r rather really recent recently reported
round

s said same sat saturday say says sec sec-
ond section see seem seemed seeming seems
sense sep sept september set sets seven sev-
enty several she short should showed since sin-
gle six sixty small so some somehow some-
one something sometime sometimes somewhere
soon start started still stop such sunday

t take taken takes taking ten than that the
their them themselves then thence there there-
after thereby therefore therein thereupon these
they thing things third thirty this those though
thought thousand thousands three through
throughout thru thu thur thurs thursday thus
time tiny to today together told too took to-
ward towards trillion tue tues tuesday twenty
two

u under unless unlike unlikely until up upon
us use used using

v very via vol

w warning was way we wed wednesday
week weeks well went were what whatever when
whence whenever where whereafter whereas
whereby wherein whereupon wherever whether
which while whither who whoever whole whom
whomever whose why will with within without
word words would

x

y year years yes yet you your yours yourself
yourselves

z