Incorporating A Rich Linguistic Model into Whole-Book Recognition

Pingping Xiu & Henry S. Baird

Computer Science & Engineering Dept
Lehigh University
19 Memorial Drive West, Bethlehem, PA 18017 USA
E-mail: pix206@lehigh.edu, baird@cse.lehigh.edu
URL: www.cse.lehigh.edu/˜pix206, www.cse.lehigh.edu/˜baird

Abstract

Whole-book recognition, a technique that improves recognition of book-images using fully automatic mutual-entropy-based model adaptation, has achieved character error rate as low as 1.9% on 50 pages of real book images in our previous publications. However, the linguistic model for word recognition was simple, assuming a uniform distribution on the words in the dictionary, so that the algorithm is unaware of prior word-occurrence distribution. As a result, the statistics of the output transcript differs largely from that of a real distribution. In this paper, we propose a post-processing technique that improves the existing whole-book recognition results by applying the constraints of a rich linguistic model - a prior word-occurrence distribution. This technique further drives the character error rate down from 1.9% to 0.97%. We also show that the whole-book recognition algorithm combined with this post-processing technique shows faster improvements in which word error rates fall monotonically with passage length.

Keywords: *document image recognition, book recognition, isogeny, adaptive classification, anytime algorithms, model adaptation*

1. Introduction

We are investigating fully automatic methods for whole-book recognition. In [10] we introduced an information-theoretic framework for identifying significant disagreements between models—the *iconic* model and the *linguistic* model—and interpreting these as candidates for corrections of one or the other of the two models so that, when the updated models are reapplied to perform recognition, a lower error rate on the entire passage results.

Our research builds on over a decades' work showing that adaptive classifiers can improve accuracy without human intervention[6]. Tao Hong[3] showed that within a book, strong "visual" (image-based, iconic) consistency-constraints support automatic post-processing that reduces error. These successes appear, to us, to be due largely to *isogeny*— the tendency of particular documents to contain only a small subset of all the typefaces, languages, image qualities, and other variabilities that occur in large collections[8]. It is well known that if models of the typefaces, languages, etc were known, even if only approximately, optimizing recognition jointly across all the models improves the accuracy[1, 5, 7].

In a long, highly isogenous book, identical (or similar) character images will occur multiple times, and the same word will also occur multiple times, independently. If the models are inaccurate, the resulting errors cause repeated disagreements between the models, which can be measured at character, word, and passage scales. Correct model adaptation, which leads to a better accuracy, will presumably also lower passage-scale model disagreement. Therefore passage-scale mutual entropy can drive model correction and reduce error rates.

In [10], a small-scale experiment, on a single textline, using an adaptation algorithm we now call ME1.0, illustrated policies that allowed automatic corrections to be made to both models, and showed empirically that both character error-rates and word-error rates could fall as a result. In [9], using an improved algorithm (ME2.0) which copes with segmentation errors and runs faster, we experimented on passages up to ten pages in length, and observed that the word recognition rate for longer words increased significantly as passage length increased.

However, in previous papers [10], [9] and [11], the model adaptation that drives the disagreements down was not constrained to make word recognition results fit the prior word-occurrence distribution. As a result, the statistics of the recognition results differs largely from the real distribution, for which we believe that there is a large margin for enhancing the recognition results, as long as we transform them to a transcript that follows the constraint imposed by a rich linguistic model - the prior word-occurrence distribution.

In this paper, we design a post-process technique to enhance the word recognition result produced by the whole-book recognition. We formulate this as an optimization problem to minimize the difference between the (ideal) linguistic model and the OCR results' word-occurrence distribution. We show the effectiveness of this algorithm over a series of experiments from short to long passages up to 50 pages. We observed that the longer the input passage is, the better performance gain the algorithm achieves. We will show that the whole-book recognition combined with this post-processing technique can achieve a faster trend of improvement as the passage length goes up to 50 pages. Also we will show the algorithm reduces the character error rate from 1.9% to 0.97% on the 50-page experiment.

In Section 2, we introduce the mathematical framework of the whole-book recognition, placing emphasis on the post-processing step. In Section 3, we motivate the design of the present experiments and give details of the algorithm with post-processing. In Section 4, we present and analyze the results of the experiments. In Section 5, we discuss the results and draw conclusions.

2. Mathematical Framework

2.1. Probabilistic Models

In our framework, two different kinds of models are required: an iconic model and a linguistic model. The iconic model, when applied to recognition, must allow the computation of *a posteriori* probabilities for all the character classes. (Of course, many such models are known [2]; we use Hamming-distance matching to multiple character image templates.) For a linguistic model, we expect to be given a lexicon (a dictionary containing valid words). The lexicon should cover most valid words, but may be incomplete; we also expect probabilities to be assigned to each word in the lexicon.

2.2. Independence Assumptions and Word Recognition

Now let X denote a sequence of T observations of character images (*i.e.* a word that is T characters long), and let S denote the true classes of these characters (in communication-theory terms, it is the inner state sequence that generates X):

$$X = (x_1, x_2, \cdots, x_T)\,, S = (s_1, s_2, \cdots, s_T) \tag{1}$$

where x_i are character images, and s_j are symbols of an alphabet. We adopt the following independence assumption, that each x_i is solely determined by its associated s_i:

$$P(x_i|s_i, \mathcal{F}) = P(x_i|s_i) \tag{2}$$

Where $\mathcal{F} = (Y, K)\,, Y \subseteq X - \{x_i\}$ and $K \subseteq S - \{s_i\}$. This assumption is similar to the one chosen by Kopec and Chou in their Document Image Decoding theory[4].

Our *linguistic model* is $P(S)$, the prior probability that word S is valid. Our independence assumption implies that

$$P(X|S) = \prod_{i=1}^{T} P(x_i|s_i) \tag{3}$$

And

$$P(x_1, x_2, \cdots, x_T) = \alpha \cdot \prod_{i=1}^{T} P(x_i) \tag{4}$$

where

$$\alpha = \sum_{(s_1 s_2 \cdots s_T)} \left[\prod_{i=1}^{T} P(s_i|x_i) \cdot \frac{P(s_1 s_2 \cdots s_T)}{\prod_{i=1}^{T} P(s_i)} \right] \tag{5}$$

Our *iconic model* is denoted by the function $P(s|x)$ for all symbols s and all character images x. So we can derive $P(S|X)$, the result of word recognition informed by both the iconic and linguistic models:

$$P(S|X) = \frac{1}{\alpha} \cdot \prod_{i=1}^{T} P(s_i|x_i) \cdot \frac{P(S)}{\prod_{i=1}^{T} P(s_i)} \tag{6}$$

The *mutual entropy* $M(P, P')$ between two distribution P and P' is defined as:

$$\mathcal{M}(P, P') = -\sum P \cdot \log P' \tag{7}$$

and we apply it to measure the difference or "disagreement" between the distributions $P(S|X)$ and $P'(S|X)$, where $P(S|X)$ is the *a posterior* probability distribution of the character string S given the image of the whole word X, and $P^{'}(S|X) = P(s_1|x_1) \cdot P(s_2|x_2) \cdot \cdots \cdot P(s_T|x_T)$ is the distribution of the character string assuming that there is no linguistic constraints or the distributions of individual characters are independent of one another. [10] and [9] utilizes this measurement to guide the model adaptation automatically to improve the models.

2.3. Incorporating Rich Linguistic Model For Post-Processing

After we get the word-level interpretation $P(S|X)$, we may post-process $P(S|X)$ to get the final interpretation of each word $P(S|X, \mathcal{X})$ using passage-level contexts, where $\mathcal{X}$ represents the whole passage's word images.

We denote the prior distribution of word occurrence in the passage as $P_f(S)$. For a given S_0, the value $P_f(S_0)$ indicates the frequency of occurrence of the entry S_0 in the corpus. $P_f(S)$ can be obtained beforehand from a large corpus, where the statistics of word occurrence is stable.

The word recognition result $P(S|X)$ obtained from Equation 6 is not good enough because the prior word occurrence distribution $P_f(S)$ - a passage-level information - is not incorporated. It is often the case that some unusual words frequently appear on the top choices of the candidate lists in the word recognition results. Therefore the average distribution of the word recognition results $\textbf{norm}\left(\sum_{X\in\mathcal{X}} P(S|X)\right)$ [1] differs largely from the prior word distribution $P_f(S|X)$, which tends to cause poor accuracy for $P(S|X)$.

In regarding to this problem, our post-process technique takes the set $\{P(S|X), X \in \mathcal{X}\}$ as input, and produces the set $\{P(S|X, \mathcal{X})|X \in \mathcal{X}\}$ as output, which conforms to this constraint:

$$\lim_{N\to\infty} \mathbf{P}(S|\mathcal{X}) = P_f(S) \tag{8}$$

Where N denotes the total number of the words in the passage. and

$$\mathbf{P}(S|\mathcal{X}) = \textbf{norm} \sum_{X\in\mathcal{X}} P(S|X, \mathcal{X}) \tag{9}$$

$$= \frac{\sum_{X\in\mathcal{X}} P(S|X, \mathcal{X})}{N} \tag{10}$$

Generally, the step from $P(S|X)$ to a new distribution $P'(S|X)$ should follow this regularity: for any S_1,S_2 and X_1, X_2,

$$\frac{P'(S_1|X_1)/P'(S_2|X_1)}{P(S_1|X_1)/P(S_2|X_1)} = \frac{P'(S_1|X_2)/P'(S_2|X_2)}{P(S_1|X_2)/P(S_2|X_2)} \tag{11}$$

Which means during the transform from $P(S|X)$ to $P'(S|X)$, we process every word X in the passage consistently: the scaling changes on any two words' probability ratios are the same for all X from $P(S|X)$ to $P'(S|X)$. With this assumption, there exists one and only one vector J that satisfies:

$$\begin{cases} P'(S|X) = \textbf{norm}(P(S|X) \cdot \textbf{diag}J), \forall X \in \mathcal{X} \\ \qquad J = \textbf{norm}\, J \end{cases} \tag{12}$$

Where the **diag** operator means to transform a vector into a diagonal matrix whose diagonal elements equal the vector's. $P(S|X)$ is a horizontal vector, which is right multiplied by a diagonal matrix **diag**J, producing another horizontal vector.

In order to get the final word recognition result $P(S|X, \mathcal{X})$, we need to find some proper J_0 so that

$$P(S|X, \mathcal{X}) = \textbf{norm}(P(S|X) \cdot \textbf{diag}J_0) \tag{13}$$

provided that $P(S|X, \mathcal{X})$ satisfies the constraint of equation 8.

We can turn this problem into an optimization problem:

$$J^0 = \arg\min_J \max_S \frac{\mathbf{P}'(S|\mathcal{X})}{P_f(S)} \tag{14}$$

[1] The definition of the operator **norm** is as follows: $\textbf{norm}(V) = \dfrac{V}{V \cdot \mathbf{1}^T}$, where $\mathbf{1} = [1, 1, \cdots, 1]$, with the same dimension as V.

Where

$$\mathbf{P}'(S|\mathcal{X}) = \mathbf{norm} \sum_{X \in \mathcal{X}} \mathbf{norm}(P(S|X) \cdot \mathbf{diag} J) \tag{15}$$

$P_f(S)$ doesn't appear in an earlier stage as in formulae 5 and 6 because $P(S)$, the true-or-false information of a string S, is more important and effective on detecting the disagreements between the iconic model and the linguistic model on the level of a single word. The frequency information $P_f(S)$ is useful at a stage that the individual word recognition results are obtained.

3. Experimental Design

The principal goals of the work reported here is to test how much the post-process technique enhances the existing word recognition results produced by the whole-book recognition. As a result, we need to describe the whole-book recognition's workflow plus the design of the post-process algorithm.

In the experiment reported here (using ME2.0), model adaptation proceeds by a sequence of *epochs*. In one epoch, every word in the passage is examined: its top-choice word interpretation (resulting from the current models) assigns a character class label s_i to each character x_i in the word. Among these, the algorithm chooses the pair $(x*, s*)$ with the highest character-scale disagreement within the word, then attempts to adapt the iconic model for character class $s*$ by picking one of its templates at random and replacing it with $x*$. This attempted adaptation is evaluated, and may be accepted as a *correction*, or undone and discarded. Thus the total number of adaptations attempted in an epoch equals the number of words in the passage, and is in general larger than the number of corrections accepted. Evaluating an attempted adaptation is accomplished, within our theoretical framework, by recomputing the passage-scale mutual entropy due to the adaptation: if it decreases, the adaptation is accepted.

In these experiments, we use page images plus an imperfect OCR transcript for one of the books ("Popular Tales of the West Highlands") in the publicly released Google Book Search Dataset. In this book, each page contains roughly 350 words, and we use up to 50 pages on the experiments in this paper. We used this OCR transcript to perform word segmentation alignment, and we proofread the transcript and the alignment manually.

We initialized the iconic model from a short passage, yielding a low inital accuracy of sixty percent words correct and fifty-five percent characters correct. The linguistic model was initialized with the 4073 words occurring in 50 pages' groundtruth: thus it is a "perfect lexicon" for the 50 pages, and a superset for smaller passage lengths. (This contrasts with our previous papers, where the linguistic model was initialized from a public-domain dictionary containing 50,000 words which did not fully cover the test set.) The joint recognition results from these initialized models yielded an approximately 25% character error rate, and we runs the whole book recognition algorithm to get improved results for each experiment.

Our new post-process algorithm is as follows. The $P_f(S)$, the prior word occurrence distribution, is estimated using the 50 pages' groundtruth words. After the adaptation, we recompute the 50 pages' transcript, keeping the word choices with top five probability values for each word, and taking the rest as zeroes, which is a good approximation to $P(S|X)$. The J vector in formula 14 is initialized with a uniform distribution, and an initial average transformed distribution $\mathbf{P}'(S|\mathcal{X})$ is computed through formula 15. And our post-process stage consists of multiple iterations: in each iteration, it finds the S_0 with the maximal $\mathbf{P}'(S_0|\mathcal{X})/P_f(S_0)$, and adjust the value of $J(S_0)$ to make the $\mathbf{P}'(S_0|\mathcal{X})/P_f(S_0)$ approach 1.0. We use at least 100 iterations in total, and the longer the passage is, the more iterations we use. This is a greedy hill-climbing approach that is not guaranteed to achieve a global optimum, but it is effective in practice, as we show next.

4. Experimental Results

The principal experimental result is that word or character error rate falls as a function of the passage length of the transcripts operated upon by our post-processing algorithm. In Figure 1, the star (*) represents word error rates on the input of the algorithm, and the cross (+) represents word error rates on its output. The horizontal axis (passage length in pages) and the vertical axis (word error rate) are displayed in log scale. Likewise, in Figure 2, the data points are character error rates from the algorithm's input and output respectively. In experiments of Figure 1 and 2, the input of the algorithm is produced by the whole-book recognition algorithm operated on 50 pages. Passage lengths include 1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 17, 25 and 50 separately. The 50-page result was computed in a single experiment; the others, for $m < 50$ pages, were computed as averages over $\lfloor 50/m \rfloor$ experiments on nonoverlapping subsets of pages.

In Figure 1, the post-processing algorithm damages the results on the short passages, and improves the results on a majority long passages. The word error rate falls monotonically as the passage length goes up, and achieves the highest gain in the 50-page experiment. In Figure 2, there are similar phenomema, except the algorithm starts to improve instead of damage at a longer passage than the word case. The character error rate improves most in the 50-page experiment, from 1.9% to 0.97%. The monotonic trends for the margins of the improvements may be due to the fact that the longer the passage is, the more meaningful the passage-level statistics is, so that the constraints from the prior word-occurrence distribution is more effective on improving the results.

Figure 3 and 4 show the comparisons between the whole-book recognition algorithm and that combined with the post-processing. Similar to Figure 1 and 2, on these two Figures, the stars (*) are error rates for the output transcripts of the whole-book recognition algorithm,

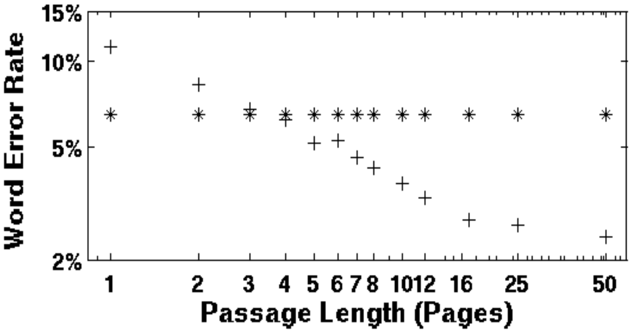

Figure 1. Word Error Rate as Function of the Passage Length for Post-Processing. The stars(*) represents data from the input of the post-procesing, or the output of the whole-book recognition; the crosses(+) represents data from the output of the post-processing.

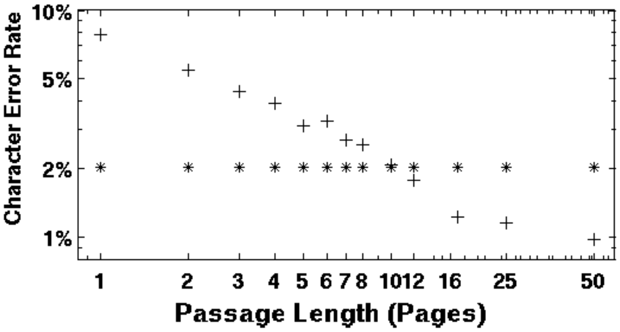

Figure 2. Character Error Rate as Function of the Passage Length for Post-Processing. The stars(*) represents data from the input of the post-procesing, or the output of the whole-book recognition; the crosses(+) represents data from the output of the post-processing.

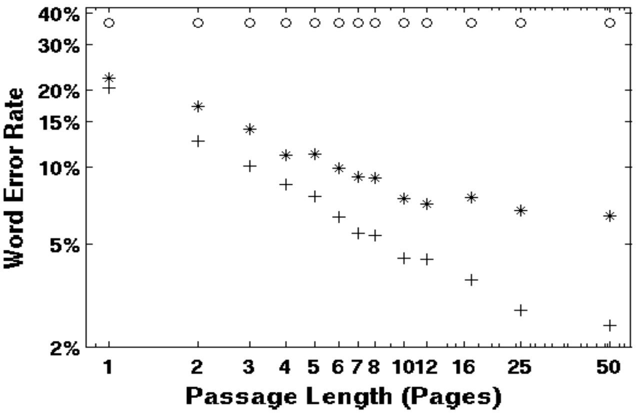

Figure 3. Word Error Rates as Functions of Passage Lengths. Circles (o) are the word error rates from the initial state. Stars(*) are the word error rates of the results of the whole-book recognition. Crosses(+) are the word error rates of the results of the post-processing, which is also the final results of the whole process.

Table 1. Wrong Recognition Results Corrected Through Post-Processing

GT	Before Post Processing		After Post Processing	
	1st cand.	2nd cand.	1st cand.	2nd cand.
and	nud:-32.0321	and:-32.6529	and:-32.6529	nud:-34.4321
	end:-25.3238	and:-25.6529	and:-25.6529	end:-30.1238
	nud:-34.9821	and:-35.9029	and:-35.9029	nud:-37.3821
	nud:-31.3821	and:-32.1029	and:-32.1029	nud:-33.7821
are	aro:-25.3098	are:-26.0248	are:-26.0248	aro:-28.1098
	aro:-22.5098	are:-22.9248	are:-22.9248	aro:-25.3098
	aro:-24.5598	are:-26.3748	are:-26.3748	aro:-27.3598
	aro:-22.7598	are:-24.6248	are:-24.6248	aro:-25.5598

and the crosses (+) are for the output transcripts of the post-processing algorithm. Experiments for Figure 3 and 4 are different from those for Figure 1 and 2: in Figure 3 and 4, the whole-book recognition's results on a certain page range are post-processed correspondingly, while in Figure 1 and 2, the same output from 50-page whole-book recognition are post-processed in parts with different passage lengths.

We observe strong correlations between the error rates and the passage lengths on all the data series. Especially, we see better linear relationships on the post-processed results. And for the data series of post-processing, the trends for reducing the error rates are faster than those without post-processing. In Figure 3 and 4, on short passages, the post-processing technique still get improvements, for the input transcripts' error rates are high so that they're easier for the post-processing technique to improve.

To get a deeper understanding of why the post-processing technique works, see some examples in Table 1. (In this table, the numeric values are log probability values that are not normalized.) Because mutual-entropy-based model adaptation doesn't distinguish words with different frequencies, there are often words recognized incorrectly as an infrequent word in the corpus, which may be corrected with the prior word-occurrence distribution.

5. Discussion and Conclusions

Our post-processing technique, which attempts to transform the post-OCR word distributions to match a prior word-occurrence distribution, is shown to be effective for enhancing the whole-book recognition's results. We have shown that the longer the passage this post-processing technique operates on, the higher performance gain it achieves. We have also shown the near monotonic trends for improving performance of the enhanced system as the passage lengths increase.

Future works may include: (1) incorporating the post-processing into every iteration of the whole-book recognition algorithm so that

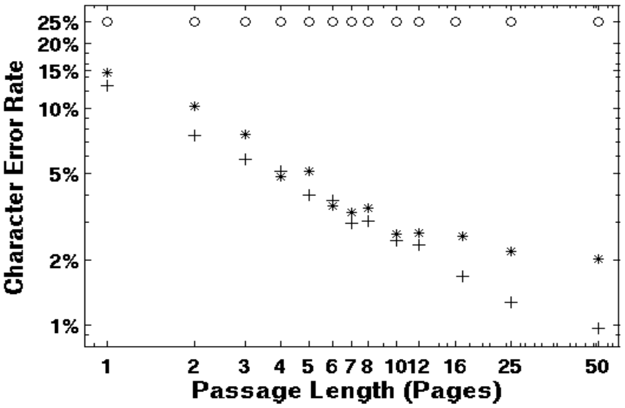

Figure 4. Character Error Rates as Functions of Passage Lengths. Circles (o) are the character error rates from the initial state. Stars(*) are the character error rates of the results of the whole-book recognition. Crosses(+) are the character error rates of the results of the post-processing, which is also the final results of the whole process.

the constraints from the prior word-occurrence distribution can directly affect the model adaptation process, which may yields higher performance; (2) analyzing how the optimization problem (equation 14) corresponds to the increasing word or character accuracy; (3) analyzing the relationship of the passage length the post-processing algorithm operates on and its effectiveness. (4) a stopping rule for the optimization.

References

[1] T. Breuel and K. Popat. Recent work in the document image decoding group at xerox PARC. In *Proc., DOD-sponsored Symposium on Document Image Understanding Technology (SDIUT 2001)*, Columbia, Maryland, April 2001.

[2] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification, 2nd Edition*. Wiley, New York, 2001.

[3] T. Hong. *Degraded Text Recognition Using Visual And Linguistic Context*. PhD thesis, 1995.

[4] G. Kopec and P. Chou. Document image decoding using Markov source models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI–16:602–617, June 1994.

[5] G. Kopec, M. Said, and K. Popat. N-gram language models for document image decoding. In *IS&T/SPIE Electronic Imaging 2002 Proc. of Document Recognition and Retrieval IV*, San Jose, California, January 2002.

[6] G. Nagy and H. S. Baird. A self-correcting 100-font classifier. In *Proc., IS&T/SPIE Symp. on Electronic Imaging: Science & Technology*, San Jose, CA, 1994.

[7] P. Sarkar, H. S. Baird, and X. Zhang. Training on severely degraded text–line images. [submitted to] IAPR Int'l Conf. on Document Analysis & Recognition, Edinburgh, August, 2003.

[8] P. Sarkar and G. Nagy. Style consistent classification of isogenous patterns. *IEEE Trans. on PAMI*, 27(1), January 2005.

[9] P. XIU and H. Baird. Towards whole-book recognition. In *Proceedings., 8th IAPR Document Analysis Workshop (DAS'08)*, Nara, Japan, September 2008.

[10] P. XIU and H. Baird. Whole book recognition using mutual-entropy-based model adaptation. In *Proc., IS&T/SPIE Document Recognition & Retrieval XII Conf.*, San Jose, CA, January 2008.

[11] P. XIU and H. Baird. Scaling-up whole book recognition (to appear). In *Proceedings, IAPR 10th Int'l Conf. on Document Analysis and Recognition (ICDAR'09)*, Barcelona, Spain, July 2009.