L. Lipsky, D. Lopresti, G. Nagy, Optimal policy for labeling training samples, Proc. SPIE Document Recognition and Retrieval, DRR XX, SPIE VOl. 8658-09, San Francisco, February 2013

Optimal policy for labeling training samples

Lester Lipsky[a], Daniel Lopresti[b], George Nagy[c, *]

[a] Department of Computer Science, University of Connecticut, Storrs, CT 06269, USA
[b] Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015, USA
[c] Electrical, Computer, and Systems Engineering Department, RPI, Troy, NY 12180, USA

ABSTRACT

Confirming the labels of automatically classified patterns is generally faster than entering new labels or correcting incorrect labels. Most labels assigned by a classifier, even if trained only on relatively few pre-labeled patterns, are correct. Therefore the overall cost of human labeling can be decreased by interspersing labeling and classification. Given a parameterized model of the error rate as an inverse power law function of the size of the training set, the optimal splits can be computed rapidly. Projected savings in operator time are over 60% for a range of empirical error functions for hand-printed digit classification with ten different classifiers.

Keywords: Interactive classification, Classifier training, Learning curves, CAVIAR systems, Green interaction

1. INTRODUCTION

The goal of building a classifier is to achieve the highest possible accuracy for a specific task of interest at the lowest possible cost. This is often an unstated goal of published work. This goal can be met better by retraining the classifier again and again with larger and larger training sets than by "freezing" it. (This option was not available when OCR systems were implemented in hardware.) Furthermore, the required correction of residual errors is seldom exploited although the corrected output of the classifier makes an enlarged (and representative!) training set available gratuitously. Recycling human corrections is one aspect of what we call "green interaction". In error-sensitive applications, the additional effort of retraining a classifier to increase accuracy from 98% to 99% may be worthwhile when we consider both development and operational costs over the lifetime of the system. We formulate a model for a multi-stage training regimen and compute its expected benefits for hand-printed digit recognition.

OCR prices quoted on the web indicate that the cost of proofreading and correcting OCR errors is much higher than that of unverified OCR.[1] We assume that the cost of machine time is insignificant (or at least constant) and that the error rate of the classifier decreases as an inverse power of the size of the training set. We propose a method of minimizing the total cost of perfect labeling of training samples and proof-reading the output of the classifier.

We illustrate our approach on the task of error-free classification of 1000 patterns (say printed characters or handwritten words). The conventional approach is to manually label a random subset of the samples, say N_0 patterns, train the classifier on the labeled training set, classify the remaining $N_1 = 1000 - N_0$ samples, and then inspect the entire "test" set and correct any residual errors. Posit further the availability of an interactive labeling system where the cost (in human time) of entering a label or correcting an incorrect label is three times the cost of confirming a correct label. The error rate of the classifier under consideration is 15% when trained on 300 patterns, and 10% when trained on 600 patterns. These error rates are averages over randomly selected training and test sets.

The expected total cost is proportional to the cost of labeling the training samples, plus the cost of confirming correctly classified samples, plus the cost of correcting wrong labels.

For $N_0 = 300$, the total cost is:

$$\text{Cost}_{300} = 3\times300 + (700\times0.85) + 3\times(700\times0.15) = 1810.$$

If $N_0 = 600$, then the cost is

$$\text{Cost}_{600} = 3\times600 + (400\times0.90) + 3\times(400\times0.10) = 2280.$$

* Further author information: (Send correspondence to G.N.)
L.L. E-mail: lester@engr.uconn.edu
D.L E-mail: lopresti@cs.lehigh.edu
G.N. Email:: nagy@ecse.rpi.edu

Clearly choosing N_0=300 is more economical. Below we show how to determine the optimal partition, given the error function, i.e., the error rate on an (infinite) test set as a function of N_0 (the size of the training set). We also propose optimization for more than a single training set, where training and interactive verification (confirmation or correction) are interspersed over a set of $N_0, N_1, N_2, \ldots, N_n$ samples, with $N_0 + N_1 + N_2 + \ldots + N_n = N_{total}$.

The germ of this idea is described in a paper presented at ICPR 2012.[2] Here we use a more appropriate error model, and a fast numerical solution instead of exhaustive search for optimization that limited our results to small data sets, We compute the expected savings for ten different classifiers on the same hand-printed digits, which shows the sensitivity of the cost function to alternative parameterizations of error curves.

Interactive pattern recognition has been advocated for over 40 years.[3, 4, 5] CAVIAR systems[6, 7, 8, 9], active learning[10] and interactive polygons[11] are more recent manifestations. "Unsupervised" classification methods like co-training[12,13] also make use of classifier-assigned labels for retraining, but cannot guarantee the same accuracy as human correction. As suggested in earlier work, automated classification interspersed with interactive labeling can reduce costs in error-intolerant applications like financial and medical form entry.[14] Our contribution here is quantification of the economies in terms of the size of splits.

For the sake of completeness, in Section 2 we briefly review our model of the cost function, as we proposed in an earlier paper.[2] Section 3 presents the revised model of the error function, including work by others on the derivation of the inverse power law and on the effect of sample size on estimating its parameters. Section 4 describes the numerical optimization procedure. Section 5 reports the computed savings in cost or time for a range of sample sizes and parameters derived from a practical application of machine learning. Section 6 offers conclusions based on both theoretical considerations and computations.

2. MODEL OF THE COST FUNCTION

Our model of the cost function for multistage labeling just accumulates the three types of costs shown in the introductory example: (1) initial labeling, (2) confirmation of correctly classified samples, and (3) correction of errors. The cost (or time) of confirmation that a label is correct is unity, and the cost of label entry or correction (which are essentially the same) is $R > 1$. The cost of each stage of N_i samples depends on N_i, and on the fraction of the N_i samples that were misclassified. In the initial stage, all N_0 labels must be entered manually, therefore its cost is $R \times N_0$.

In subsequent stages i, the cost depends on N_i as well as on the number M_i of classifier errors made by the classifier when (re-)trained on all the previously verified patterns. M_i is given by the error function $f(N_{train}; \beta)$, where β is the parameter vector of the error function (which is often called a *learning curve*).

The ratio R of correction to confirmation depends on the application. It typically ranges from 3 to 10. It would be higher for word patterns than for character patterns, and higher for determination of table structure than for the content of individual table cells. It also depends on the device and modality used for corrections.[15] It can be determined readily from the logged time-stamps of the interaction in a short labeling session. We expect it to be roughly equal to the base-two logarithm of the number of classes because specifying one of C classes is equivalent to $\log_2 C$ dichotomies, as predicted by the Hick-Hyman Law of psychometrics[16,17,18].

The number of samples to be labeled is $N_{total} = \sum_{i=0}^{n} N_i$. The problem we solve is the optimal split of N_{total} into n sets of N_i samples. There are $n>0$ stages, including the first stage of N_0 initially unlabeled samples. All N_0 samples of the initial training set must be labeled, therefore $M_0 = N_0$, and the number M_i of errors to be corrected at each stage i is:

$$M_i = N_i f\left(\sum_{k=1}^{i} N_{k-1}; \beta\right)$$

The operator time T_i (the cost-of-confirmation +the cost-of correction at stage i) is:

$$T_i = (N_i - M_i) + RM_i = N_i + (R-1)M_i = N_i\left(1 + (R-1) f\left(\sum_{k=1}^{i} N_{k-1}; \beta\right)\right)$$

Therefore the total time T_{total} or cost is:

$$T_{total} = \sum_{h=0}^{n} T_h = RN_0 + \sum_{h=1}^{n} N_h \left(1 + (R-1) f\left(\sum_{k=1}^{h} N_{k-1}; \beta \right) \right)$$

The derivatives of T_{total} with respect to N_i are proportional to $(R\text{-}1)$. Therefore the optimum split is independent of R if $f(N_{train}, \beta)$. is a decreasing function, as assumed, of N_{train}. In other words the error rate drops—or at least does not increase—as the number training samples increases.

3. ERROR FUNCTIONS (LEARNING CURVES)

Since the error function is not available on patterns to be labeled, it must be extrapolated from a small training sample. The error function is the expected error rate E_N of a classifier as a function of N, number of training patterns. There is some agreement that for a wide range of applications the error function can be modeled as:

$$E_N = f(N, \beta) = aN^{-b} + c, \text{where } \beta = [a, b, c] \text{ with } a, b, c, > 0.$$

The constant a is called the *learning rate*, b is the *decay rate*, and c is the *Bayes error*. The learning rate sets the initial error rate. The decay rate governs the rate of decrease of the error rate with the number of training samples. The Bayes error is the asymptotic error rate with an infinite training sample.

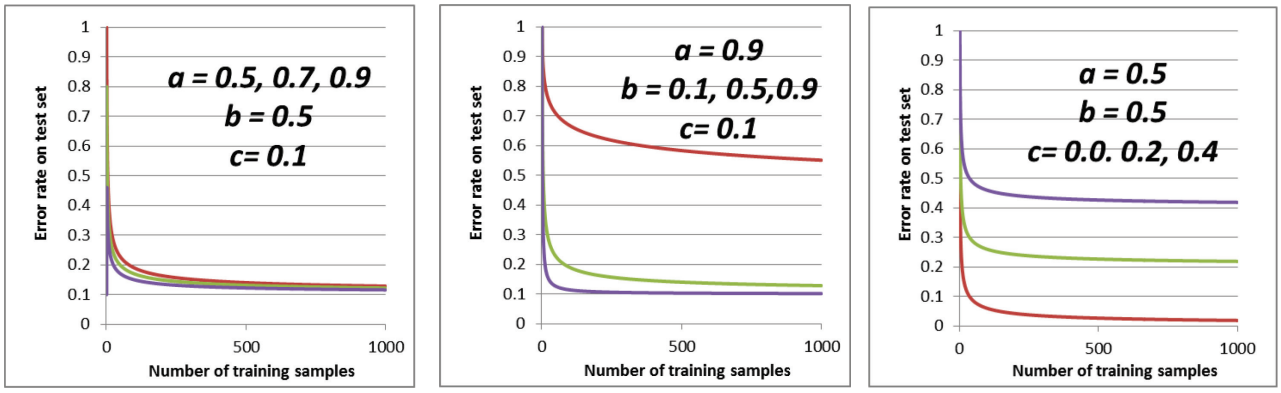

Figure 1. Effect of the three parameters on the error curve. a is the learning rate, b is the decay rate, and c is the Bayes error.

The effect of the learning rate, decay rate, and Bayes error parameters on the error function is illustrated in Fig.1, which shows plots of E_N vs. N for selected parameters according to the above error function equation. The actual values of the parameters depend on the distribution of the dataset (as represented in the chosen feature space) and on the classifier. The Bayes error is zero unless the same pattern is sometimes in one class and sometimes in another. In OCR applications it is near zero except in low-resolution, noisy documents. Ho and Baird reported an empirical estimate of the Bayes error on millions of patterns.[19] Although they present learning curves, we cannot use them here because they were derived on a dichotomy of two printed letters (**e** vs. **c**). In a two-class problem like this, both confirmation and correction require only a single bit per pattern, therefore the proposed method offers no cost saving whatsoever (R = 1). In multi-class problems, however, it can be expected that correction is more expensive than confirmation, and hence it is worthwhile to optimize the splits.

The inverse power law for regression or classification is justified from several different starting points (statistical mechanics, approximation theory) in Mukherjee et al.[20] In general, the error curve is not known for the data about to be labeled. It can, however, be estimated and extrapolated from one or more small random subsets of samples. The authors[20] derive the variability of the estimates of the parameters when the 25-percentile and 75-percentile error curves are estimated from subsamples. In addition to the uncertainty of the estimates due to the variability of the training set, estimating the error rate on a finite test set has also been intensively studied.[21,22,23]

Table 1 of the above article[20] reports mean values a ranging from 0.5 to 2.0, of b from 0.1 to 0.7, and of c from 0.00 to 0.01 on eight micro-array image datasets. However, the number of samples in these dichotomies (normal/abnormal), ranging from 58 to 280, is much smaller than in typical OCR datasets, and the feature space is also different.

We demonstrate the proposed labeling policy on a set of error curves reported by Liu, Sako and Fujisawa for hand-printed numerals classified by ten different classifiers.[24] Digit recognition is a good example for us, because in contrast to text OCR, language context cannot be used to correct errors.[25]

The primary objective of Liu, Sako and Fujisawa was to compare the performance of diverse classifiers under various conditions. Prof. C-L Liu kindly provided the numerical data underlying the published curves. The six progressively larger training sets for the learning curve experiments consisted of 2,064, 4,134, 8,273, 16,549, 33,103, and 66,214 samples. The test set contained 22,271 samples, roughly equally distributed between the ten classes of digits.

To illustrate the derivation of the parameters of the error function required to compute the optimal size of the training set, we select the widely known Nearest Neighbor classifier. The parameter values obtained by fitting the inverse power law model to the 1-NN learning curve with three points and with six points are:

$$a_3 = 2.13,\ b_3 = 0.579,\ c_3 = 0.0123 \quad \text{and} \quad a_6 = 0.962,\ b_6 = 0.452,\ c_6 = 0.0071.$$

Fig. 2 shows both the curves as reported in the Liu et al. IJDAR article, and as re-plotted in the format shown above from the numerical data. The curves with the above parameters provide an excellent fit to the experimental data, which confirms the accuracy of the model in this case. The learning rate a is higher than the values reported in by M.ukherjee et al.[20] Duda, Hart and Stork[26] suggest that all of the parameters must be less than unity, but that is not necessarily the case for published experimental data.

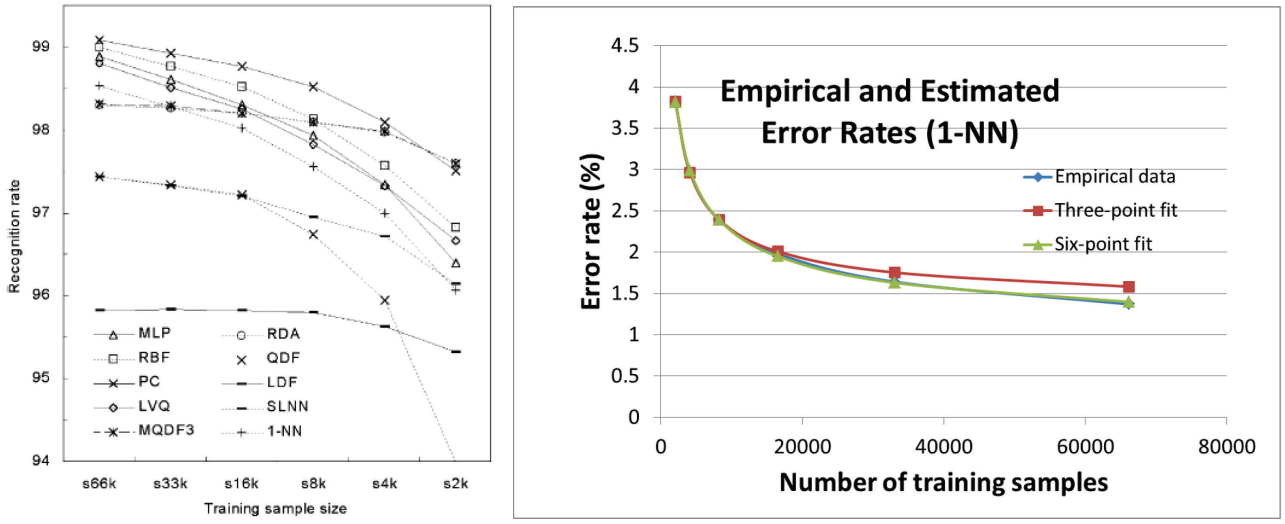

Figure 2. Learning curves for hand-printed digits. On the left is Fig. 5 reproduced from Liu et al.[24] On the right, the curve for the 1-NN (nearest-neighbor) classifier, re-plotted in the format of Fig. 1 above, and the superimposed three-point and six-point fitted inverse power-law curves. The difference between the empirical data and the curve fitted to the six points is barely visible. The three-point estimates of the error rate, extrapolated to large training sets, are higher.

4. NUMERICAL OPTIMIZATION

The model was fitted to experimental data by least-squares minimization subject to the constraint of keeping all the parameters positive. At least three observations are necessary to determine the three model parameters *a, b* and *c*. The effective loss due to using only the three observations on the first 8,273 training samples is indicated by comparing the savings obtained with these parameters to the savings using all six observations (up to 66,214 training samples). The parameters are fitted separately to each of the error functions from all ten classifiers.

The nonlinear least-squares optimization was performed by setting the derivative of the sum of the errors to zero. The nonlinear equation for b is solved by a linear scan in 0.1 step increments followed by application of the Secant Method to find the exact root of the equation. Then the values of a and c are obtained from their functional relation with b. The maximum fitting error on the last observation is only 15%. This indicates that fitting the parameters to only the first three observations is sufficient. The worst fit is the 1-NN classifier of Fig. 2. Values for four-point and five point fits are near those of the six-point fit. When all six points are fitted, the maximum relative error is 3%. Fig. 3 shows the re-plotted original data and the error curves for the ten classification experiments fitted with either three points or with all six.

The next step is finding the optical splits given the error functions. The number of variable values to be determined by minimizing T_{total} is equal to the number of classifier training stages n. The minimization of the total cost by choice of either N_0 or N_0 *and* N_1 is achieved by setting the derivatives (for $n=1$) or partial derivatives (for $n=2$) of the error function to zero. The resulting nonlinear algebraic equations are solved with a linear scan followed by the Newton-Raphson method. The convergence criterion of 10^{-10} was reached after four or five iterations. The constraint 0d c d 1 affects only the MLP and LVQ classifiers, where unconstrained extrapolation from three points would yield a slightly negative Bayes error. In these cases c was set to zero.

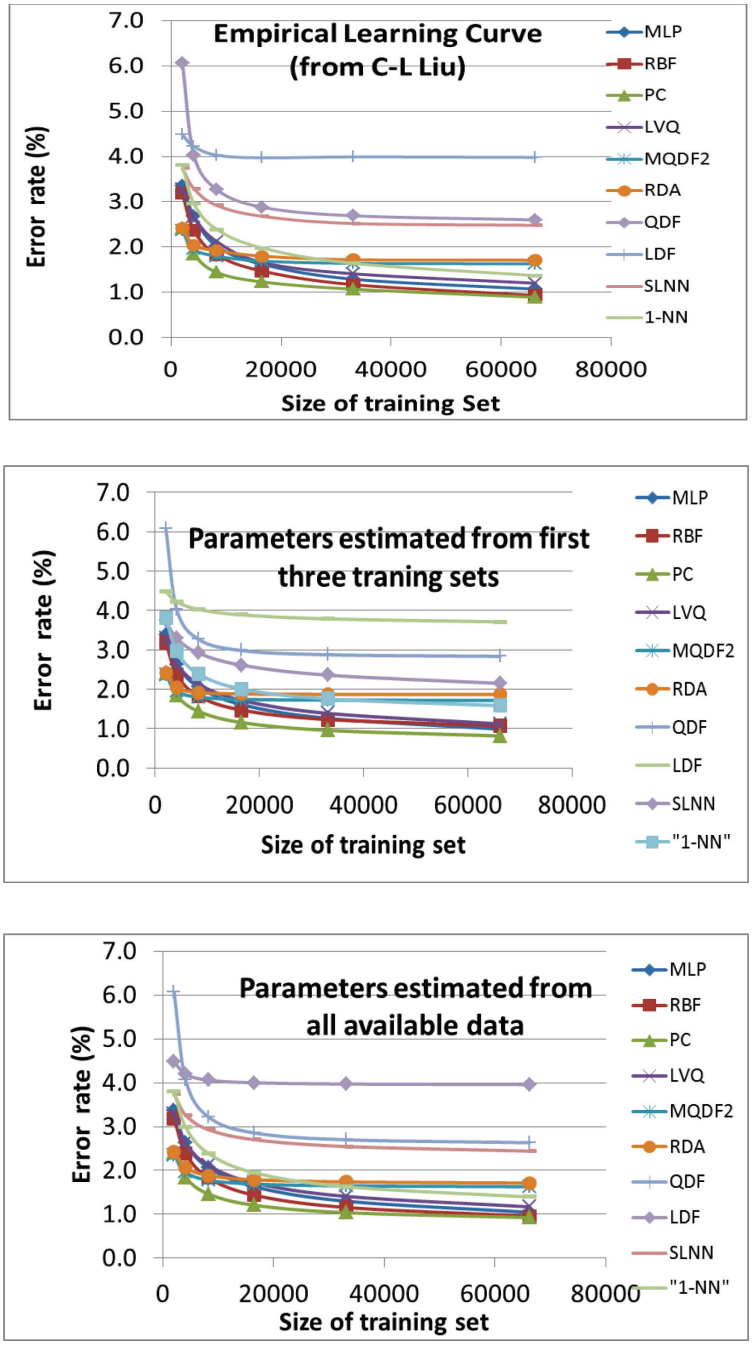

Figure 3. Learning curves for hand-printed digits estimated from 8,273 training samples and 66,214 training samples respectively. The number of training sets used for estimation makes little difference in the estimates of the error rates beyond a few hundred samples.

5. COMPUTED RESULTS

The error curves for the 10 classifiers compared experimentally by Liu et al.[24] provide examples of the potential savings. We computed the error curve the parameters of and the optimal splits from training the classifier on the first 8273 training samples of each experiment, and also on all 66,214 training samples. The error rates were determined on a 22,271 sample test set without overlap with the training set. The cost ratio R was set to 3 H$\log_2$ 10. The cost savings S obtained by inspecting and correcting the classifier results instead of entering every label is computed as $S = (T_{max} - T_{total}) / T_{max} = 1 - T_{total} / R \times N_{total}$. Table 1 shows the computed values of the optimal splits (N_0 and N_1), the total labeling time or cost, and the resulting cost savings S for all 20 cases.

Table 1. Computed results from experimental runs. The first row for each classifier shows results for parameters estimated from 8,273 samples. The second row show the results for parameters fitted to the entire data set of 66,214 samples. The error rates are computed on 22,271 other samples. The ten classifiers tested are described in the article cited above[24].

	Parameters			n=1			n=2			
	a	*b*	*c*	N_0	T_{total}	*S (%)*	N_0	N_1	T_{total}	*S (%)*
MLP	0.595	0.385	0.002	1065	73922	62.8	224	7725	70301	64.6
	0.501	0.354	0.000	1035	73886	62.8	214	7944	70261	64.6
RBF	1.185	0.495	0.005	1191	73829	62.8	267	7136	70155	64.7
	2.153	0.588	0.008	1278	73929	62.8	301	6708	70260	64.6
PC	1.367	0.574	0.007	1003	72502	63.5	218	6029	69422	65.1
	0.858	0.497	0.005	953	72443	63.5	200	6431	69339	65.1
LVQ	0.478	0.365	0.003	966	73661	62.9	197	7624	70251	64.6
	0.349	0.310	0.000	897	73554	63.0	177	7983	70193	64.7
MDQF2	27.908	1.079	0.016	1081	72418	63.5	281	4021	69862	64.8
	355.874	1.431	0.017	126	72652	63.4	383	3662	70111	64.7
RDA	8.635	0.927	0.017	942	72264	63.6	223	4120	69830	64.8
	1527.526	1.642	0.019	1299	72764	63.4	425	3431	70322	64.6
QDF	381.997	1.220	0.026	2387	77986	60.7	844	6500	73624	62.9
	1467.822	1.405	0.028	2433	77942	60.8	911	6132	73788	62.9
LDF	28.268	1.124	0.039	966	74930	62.3	245	3605	72690	63.4
	0.279	0.438	0.035	538	74208	62.6	94	5420	72218	63.6
SLNN	0.969	0.543	0.022	898	74047	62.7	187	5915	71215	64.1
	0.204	0.269	0.011	659	73666	62.9	119	7630	70945	64.3
1-NN	0.962	0.452	0.008	1200	74664	62.4	267	7506	70890	64.3
	2.132	0.579	0.012	1316	74790	62.3	312	6869	71041	64.2

The savings vary relatively little between classifiers and between estimating the parameters on 8273 or 66,214 training samples. The large differences among the values of some of the fitted parameters are not reflected in large differences in T_{total} and S. The savings are all between 60% and 65%. The maximum possible value of the savings, with all samples confirmed and none corrected is $S_{max} = (R-1)/R$ = 66.7%. (It would, of course, be larger with $R>3$.) Therefore in practice the error function parameters could be estimated with any classifier on a small training sample (here about one eights of the total number of samples to be labeled). The resulting splitting policy and savings should apply to similar data and any other classifier.

For most classifiers N_0 is about 1000 samples with $n=1$. The error rates at N_0 range from 3.0% to 5.4%. Therefore fewer than 3000 samples need to be corrected on the remaining ~65,000 samples. This is, of course, the source of the savings.

We perform the numerical optimization only for $n=1$ and $n=2$ because even with large sample sizes the maximum savings is almost reached with $n=1$. The savings obtained with $n=2$ are, only 2% larger than those obtained with $n=1$. The total time for the two-stage process is 3%-5% less than for a single stage. This justifies using only one or two stages for retraining the classifier: further stages could decrease the total time and cost only slightly. In either case ($n=1$ or n=2), only a small fraction (N_0 / N_{total}) of the samples (less than 2%) needs to be initially manually labeled.

6. DISCUSSION

The results above show that on hand-printed digit classification, significant savings can be achieved by training the classifier on a small set of patterns and giving the human operator the option of quickly confirming correct labels. Our model assumes that the error function or learning curve is already known. When it is not known, our simulations suggest that the inverse power law learning curve can be accurately extrapolated from small training sets to large training sets. The savings obtained from an optimal split are nearly independent of what classifier is used. The size of the optimal training sample(s) increases slowly with the total number N_{total} of patterns to be recognized. Two-stage classification yields only a small advantage over the single optimal split.

In practice, we envisage the following protocol:

1. Use a general-purpose classifier to assign labels to a given number, say 10,000, pattern samples.
   (If no general-purpose classifier is available, go to 2.)
2. Interactively verify the labels of these samples.
3. Split the data into a training set (say 6,000 samples) and a test set (4000 samples).
4. Train the target classifier on 2,000, 4,000, and 6,000 sample training sets.
5. Run each of the resulting classifiers on the test set.
5. Estimate the learning curve and the optimal splits from the three error counts on the 4000 sample test set.
6. If $N_0 < 6000$, use the already trained classifier to label the next N_1 samples before retraining it.
7. If $6000 < N_0 < 10000$, retrain the classifier on all 10,000 labeled patterns and classify the next N_1 samples.
8. If $10000 < N_0$, use the already trained classifier to assign labels to the rest of the N_0 patterns, verify them, and then proceed to the next N_1 samples.

In Table 1 we did not take into account the cost of labeling the 22,271 test samples because in practice the test set would require only interactive verification of a small number of samples classified by a general-purpose classifier. Labeling these samples should be considered part of the initial labeling cost although it could, of course, already take advantage of the labels assigned by the general purpose classifier. Large test sets and cross validation are desirable primarily in ascertaining the error rate of classifiers operating at low error rates, as in the *s66k* experiments of Liu et al.[24] Because the split is decided using only small training samples resulting in relatively high error rates, variability introduced by a small test set is not a major concern. We could not, however, follow this procedure because we did not have error counts on the remainder of the 66,214 training samples after the classifier was trained on a smaller subset. We therefore estimated the cost of verification from the error rates obtained on the 22,271 test samples.

In practice, the learning curves would usually be estimated from a sample smaller than N_0 and extrapolated to N_{total}. Our results suggest that a three-point estimate is adequate, but we did not have fine-enough-grain data to confirm that estimates based on training *and* test samples *smaller than* N_0 are acceptable.

We have not found any published results on the ratio R of label entry or correction to confirmation. We believe that our estimate of $R=3$ is conservative for digit classification. R will be higher for classification into more classes, like alphabetic characters or words. This will yield higher savings.

The actual value of R also is affected by the design of the interface and the skill and experience of the operators. Confirmation does not, of course, require an actual keystroke. It is sufficient to advance the display (or the operator's attention) to the next pattern to be verified. The optimal size of the context window, i.e., how many scanned patterns and

assigned labels are displayed at a time, is data dependent. Because humans are good at visual anomaly detection, in some applications it may be desirable to display the patterns grouped by assigned label rather than sequentially. The initial labeling stage can be speeded up by displaying cluster prototypes and labeling all the patterns in the cluster at once. Because these factors may all affect *R*, it is desirable to determine it from a log of time-stamped operator actions kept during the parameter estimation phase.

Although we did not consider the cost of computing time, even if the cost is negligible the elapsed time may not be. If computing time and operator scheduling really did not matter, the optimal policy would be retraining the classifier after each pattern is classified. In an operational environment, adequate provisions must be made for a smooth workflow and minimal operator down-time. Therefore a low value *n*, the number stages, is preferable. Future refinement of the model should take these factors into account.

ACKNOWLEDGMENTS

We are grateful to Professor C-L Liu, Deputy Director, National Laboratory of Pattern Recognition (NLPR), Institute of Automation of Chinese Academy of Sciences (CASIA), for sharing the numerical results of his experiments. Daniel Lopresti receives support from a DARPA/IPTO grant administered by Raytheon BBN Technologies.

REFERENCES

[1] Blostein, D. and Nagy, G. "Asymptotic cost in document conversion", *Proc. SPIE/DRR 2012*, San Francisco (2012).

[2] Lopresti, D. and Nagy, G., "Optimal partition for semi-automated labeling," *Proceedings of International Conference on Pattern Recognition* 2012, Tsukuba, Japan (2012.

[3] Kanal, L., "Interactive pattern analysis and classification systems: A survey and commentary," *Proceedings of the IEEE*, 60, 10, 1200-1215 (1972).

[4] Sammon, J.W. "Interactive pattern analysis and classification," *IEEE Trans. Computers* 19, 7, 594-616 (1970).

[5] Ascher, R.N., Koppelman, G., Miller, M.J. and Nagy, G. "An Interactive System for Reading Unformatted Printed Text," *IEEE Transactions on Computers*, 20, 12, 1527-1543 (1971).

[6] Zou, J. and Nagy, G., "Evaluation of model-based interactive pattern recognition," *Proceedings of International Conference on Pattern Recognition XVII*, vol.II, 311-314 (2004).

[7] Evans, A., Sikorski, J., Thomas, P., Cha, S-H. Tappert, C., Zou, J., Gattani, A., and Nagy, G., "Computer Assisted Visual Interactive Recognition (CAVIAR) Technology," *IEEE International Conference on Electro-Information Technology*, Lincoln, NE (2005).

[8] Nagy, G. and Zhang, X., CalliGUI: "Interactive Labeling of Calligraphic Character Images," *Proc. ICDAR* 11, Beijing (2011).

[9] Nagy, G., and Tamhankar, M., "VeriClick, an efficient tool for table format verification," *Proc. SPIE/EIT/DRR*, San Francisco (2012).

[10] Cohn, A., .Ghahramani, Z, Jordan, M.I., "Active Learning with Statistical Models", *J. Artificial Intelligence Research*, Vol. 4, 129-145 (1996).

[11] Zhu, Y., Shen, T., Lopresti, D., "Huang, X., "Interactive Polygons in Region-Based Deformable Contours for Medical Images," Int. Conf. Biomedical Imaging,(ISBI_ Boston, 37-40 (2009).

[12] Blum, A. and Mitchell, T. "Combining Labeled and Unlabeled Data with Co-Training,", *Proc. 11th Conf. on Computational Learning Theory* (COLT), 92-100 (2008).

[13] Frinken, V., Fischer, A., Bunke, H., and Fornes, A., "Co-Training for Handwritten Word Recognition," *Proc. ICDAR 2011*, 314-318 (2011).

[14] Klein, B., and Dengel, A., "Problem-adaptable document analysis and understanding for high-volume applications," *IJDAR 6*, 3, 167-180 (2003).

[15] Arif, A.S., and Stuerzlingerm, W., "Predicting the Cost of Error Correction in Character-Based Text Entry Technologies," *CHI 2010*, Atlanta, GA (2010).

[16] Hick,W. E. "On the rate of gain of information," *Quarterly Journal of Experimental Psychology*, 4, 11–26 (1952).

[17] Hyman, R., Stimulus information as a determinant of reaction time, *Journal of Experimental Psychology,* 45, 188–196 (1953).
[18] Seow, S. C. "Information theoretic models of HCI: a comparison of the Hick-Hyman law and Fitts' law," *Human-Computer Interaction* 20, 3 (2005).
[19] Ho, T.K. and Baird, H.S., "Estimating the intrinsic difficulty of a recognition problem," *Proc. ICPR12*, 178-193 (1994).
[20] Mukherjee, S., Tamayo, P, Rogers, P, Rifkin, R., Engle, A., Campbel, C., Golub, T.R. and Meriov, J.P., "Estimating dataset size requirements for classifying DNA Microarray Data," *Journal of Computational Biology* 10, 2, 119-142 (2003).
[21] Raudys, S, and Jain, A.K., "Small Sample Size Effects in Statistical Pattern Recognition: Recommendations for Practitioners." *IEEE Trans. Pattern Anal. Mach. Intell*. (PAMI) 13(3):252-264 (1991).
[22] Devoye, L., Gyorfi,. I. and Lugosi, G.A., *[A Probabilistic Theory of Pattern Recognition],* Springer-Verlag, New York (1997).
[23] Guyon, I., Makhoul, J., Schwartz, R., and Vapnik, V. "What size test set gives good error estimates?" *IEEE IEEE Trans. Pattern Anal. Mach. Intell*. (PAMI) 20, 52-64 (1998).
[24] Liu, C.-L., Sako, H,. and Fujisawa,:H., "Performance evaluation of pattern classifiers for handwritten character recognition," *International J. of Document Analysis and Recognition* 4, 191-204 (2002).
[25] Taghva, K. and Stofsky, E., "OCRSpell: an interactive spelling correction system for OCR errors in text," *International J. of Document Analysis and Recognition* 3,3, 125-`137 (2001).
[26] Duda, R., Hart, P., and Stork, D., [Pattern Classification], Wiley, Second Edition, 9.6.7, p. 492 (2001).