Toward Optimal Labeling Strategy under Multiple Unreliable Labelers

Chuck P. Lam
Lama Solutions LLC
496 West Charleston Road, Suite 302
Palo Alto, CA 94306
email: chuck.lam@lama-solutions.com

David G. Stork
Ricoh Innovations, Inc.
2882 Sand Hill Road, Suite 115
Menlo Park, CA, 94025-7022
email: stork@rii.ricoh.com

Abstract

One of the most resource intensive tasks in building a pattern recognition system is data collection, specifically the acquisition of sample labels from subject experts. The first part of this paper explores an EM algorithm to train classifiers using labelers of various reliability. Exploiting unreliable labelers opens up the possibility of assigning multiple labelers to judge the same sample. The second part of this paper examines an optimal strategy such that labelers are assigned to judge samples to maximize information given to the learning system. The optimal labeling strategy for the idealized case of two labelers with two samples is examined and illustrated.

Introduction

Building pattern recognition systems generally involves four steps: model selection, data collection, training, and testing. Model selection involves picking the right computational model for the application and the necessary parameters and features. Data collection gathers and insures the quality of training data. Training adjusts the model to best "fit" the training data. Testing measures the performance of the trained classifier, ensuring its adequacy for the application.

Model selection and training tend to be closely related and have received the bulk of research effort. Textbooks in the area (Duda, Hart, & Stork 2001; Hastie, Tibshirani, & Friedman 2001) generally devote many pages to various models (e.g., neural networks, decision trees, Bayes nets) and learning algorithms (e.g., maximum likelihood, gradient descent, Bayesian estimation) and the properties of those learning algorithms (e.g., rate of convergence, stability, computational complexity).

Unfortunately, little in the literature discusses problems related to collecting training data and insuring data quality. This is surprising considering the significant amount of time and effort spent on data collection. Many organizations, such as the Linguistic Data Consortium (LDC)[1], the Center for Excellence in Document Analysis and Recognition (CEDAR)[2], and the National Institute of Standards and Technology (NIST)[3], devote a considerable amount of resources to creating databases for pattern recognition researchers and developers. The majority of work in all these situations involves knowledge workers of varying expertise transcribing and checking data. The system must ensure a high quality dataset as the final output even though each individual knowledge contributor can be unreliable.

In particular, mechanisms should be set in place to verify, or "truth," the labels collected, since mislabelling is often a significant source of error. For example, consider a real-world task of labeling volcanos in radar images of Venusian surface from the Magellan spacecraft (Smyth *et al.* 1995). A labeling experiment involving several planetary geologists, who are experts in Venus volcanism, shows a level of disagreement among them such that *at least* one of them must have a mislabeling rate higher than 19% (Smyth 1996).

The process of collecting and truthing labels has traditionally been based on heuristics and trial-and-error. The goal of this paper is to introduce some formal computational methods and theories into data acquisition and truthing. We first explore redundant labeling as a mechanism to improve data quality, and a formal EM approach is introduced for learning from multiple labelers. Since each label collected costs time, money, and probably other resources, one would like to be economical in deploying labelers. We explore an information-theoretic approach that requests labels in an optimal way to maximize information gain. We conclude the paper with references to related work and discussion of areas for future exploration.

EM Learning from Multiple Unreliable Labelers

In many areas of science and decision making, one common technique to handle subjective and unreliable information is through redundancy. For example, peer review of scientific paper submissions ensures the general quality of the final publications. Similarly, in open source software development, peer review of code submissions ensures the reliability of the end system.

Redundancy can also improve the overall reliability of datasets acquired from unreliable labelers. A straightforward approach, analogous to peer review, is to have multiple

[1] http://www.ldc.upenn.edu/

[2] http://www.cedar.buffalo.edu/

[3] http://www.nist.gov/

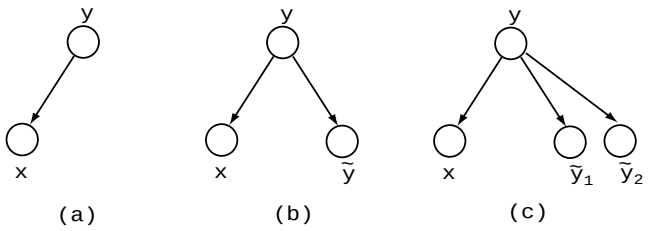

Figure 1: (a) A typical generative model of data. The probability $p(\mathbf{x}, y)$ is decomposed into $P(y)$ and $p(\mathbf{x}|y)$, which are then represented separately. (b) An extension of the model in which a contributor's judgement, $\tilde{y}$, is also a random value generated by the actual class value. An important assumption in this model is that a contributor's judgement is independent of the feature data given the actual label. (c) A further extension with multiple contributors. The contributors' judgements are also independent of each other given the actual class label.

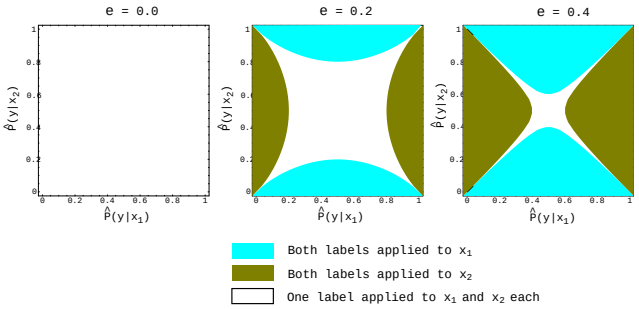

Figure 2: The optimal label acquisition strategy for two points $\mathbf{x}_1$ and $\mathbf{x}_2$ is shown for three levels of mislabeling rate ϵ, and $P(y = \omega_1|\mathbf{x}_1)$ as given by a classifier. In the limit of $\epsilon \to 0$, the strategy is to present one data point to contributor 1 and the other data point to contributor 2. This fits intuition since if the contributors are perfect, then it is only redundant to ask the two of them to judge the same point. In the large-ϵ case, the strategy is to present the most uncertain point (i.e., the one with $P(\omega_1|\mathbf{x})$ closest to 0.5) to both contributors.

labelers label the same sample, and the label used for learning is based on a majority vote. However, some information is lost in this approach, and it ignores the varying reliability of the different labelers. There is a more effective probabilistic approach to aggregate different labelers' judgement for learning, taking into account the labelers' reliability.

Many pattern classification approaches assume a generative model of data as graphically depicted in Figure 1(a). One supplies a set of feature data $\mathbf{x}$ and corresponding class labels y to train the system, and the trained classifier takes input $\mathbf{x}$ and outputs an estimate $\hat{P}(y|\mathbf{x})$, which is the classifier's belief of which class $\mathbf{x}$ belongs to. Figure 1(b) shows a generative model in which a contributor's (unreliable) judgement $\tilde{y}$ is generated only from y. That is, we assume the independence relationship $P(\tilde{y}|y, \mathbf{x}) = P(\tilde{y}|y)$. When given only $\mathbf{x}$ and $\tilde{y}$ but not y, the learning problem can be seen as one of learning with missing data, with the true label y being the missing data. A popular solution is the Expectation-Maximization (EM) learning algorithm (McLachlan & Krishnan 1996). The model in Figure 1(b) can be further extended to include multiple labelers, as shown in Fig. 1(c). One additional assumption is that the different contributors' judgements are also independent of each other given the actual class label. The EM algorithm can again be used with this extended model.

To simplify our exposition, we will assume the availability of just two labelers, $\tilde{y}_1$ and $\tilde{y}_2$, although in principle more labelers can be handled in the same way. The joint distribution of all the variables can be decomposed into

$$p(\mathbf{x}, y, \tilde{y}_1, \tilde{y}_2) = P(y)p(\mathbf{x}|y)P(\tilde{y}_1|y)P(\tilde{y}_2|y).$$

Each of the factors $P(y)$, $P(\tilde{y}_1|y)$, $P(\tilde{y}_2|y)$, and $p(\mathbf{x}|y)$ are modeled separately and trained with the EM algorithm. Note that not all factors are necessarily used for all data points in EM learning; if there is no label $\tilde{y}_2$ for a data point $\mathbf{x}$, then there is no need for the factor $P(\tilde{y}_2|y)$. Furthermore, if there is no label at all for a sample (i.e., both $\tilde{y}_1$ and $\tilde{y}_2$ are missing), then the EM algorithm is only applied to learn $P(y)$ and $p(\mathbf{x}|y)$ and it reduces to EM learning with unlabeled data (Nigam *et al.* 2000).

While all four factors are modeled and learned, only $P(y)$ and $p(\mathbf{x}|y)$ are needed for final deployment. That is, for classification one is calculating $P(y|\mathbf{x})$, which by Bayes' rule is equal to

$$P(y|\mathbf{x}) = \frac{P(y)p(\mathbf{x}|y)}{\sum_y P(y)p(\mathbf{x}|y)}.$$

Optimal Labeling Strategy

In the presence of multiple labelers, it seems uneconomical to have all labelers label all samples. One insight from active learning (Cohn, Ghahramani, & Jordan 1996) is that careful choice of samples for labeling can reduce the labeling effort without sacrificing classifier performance. In the context of multiple unreliable labelers, careful labeling strategy should also reduce the labeling effort while retaining the same accuracy.

If multiple imperfect labelers are available, and they all work for free, then it is obvious that each sample should be labeled by all the labelers. Unfortunately, there is no free lunch in life. Even where labelers are volunteers, they will only provide so many labels before being exhausted. Therefore the labelers are still a limited resource to be employed carefully. An interesting problem is determining the optimal strategy for employing these labelers.

One approach is to maximize learning by asking labelers to provide the most informative labels. A principled implementation of which is to minimize uncertainty of the unknown true labels y. A natural measure of uncertainty is the information entropy (Cover & Thomas 1991), defined as $H(X) = \sum -p(X) \log_2 p(X)$.

To help us gain insights, we study a simple example. Assume there are two data points, $\mathbf{x}_1$ and $\mathbf{x}_2$, and two labelers, $\tilde{y}_1$ and $\tilde{y}_2$. Each labeler is willing to judge only one data

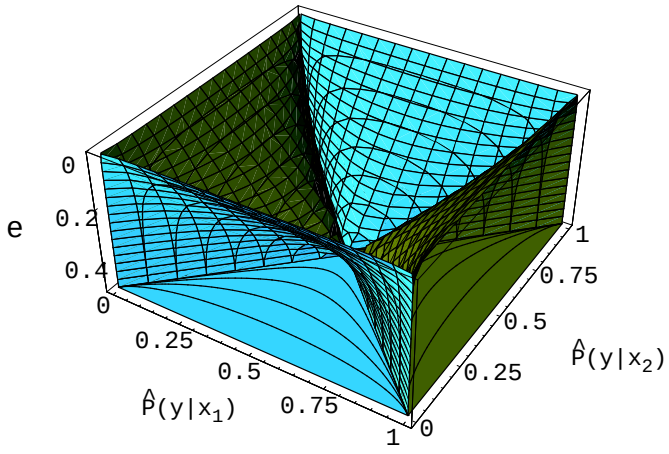

Figure 3: The optimal label acquisition strategy for two points $\mathbf{x}_1$ and $\mathbf{x}_2$ with mislabeling rate ϵ between 0 (ceiling) and 0.5 (floor). Fig. 2 shows essentially just three horizontal cross sections of this 3-D graph, although the solid cyan and green sections have been carved out to show the contour projections on the floor of the 3-D box. The color legend is the same as in Fig. 2.

point. The two labelers are assumed to provide labels that are random flippings of the true labels; labeler $\tilde{y}_1$ flips with probability ϵ_1 and labeler $\tilde{y}_2$ flips with probability ϵ_2. There are four labeling strategies we can choose from:

1. Both labelers judge $\mathbf{x}_1$
2. Both labelers judge $\mathbf{x}_2$
3. Labeler $\tilde{y}_1$ judges $\mathbf{x}_1$ and labeler $\tilde{y}_2$ judges $\mathbf{x}_2$
4. Labeler $\tilde{y}_1$ judges $\mathbf{x}_2$ and labeler $\tilde{y}_2$ judges $\mathbf{x}_1$

The uncertainty resulting from each of the four data collection strategies are respectively

1. $E[H(y)] = \mathcal{E}_{\tilde{y}_1,\tilde{y}_2}[H(y|\mathbf{x}_1, \tilde{y}_1(\mathbf{x}_1), \tilde{y}_2(\mathbf{x}_1))] + H(y|\mathbf{x}_2)$
2. $E[H(y)] = H(y|\mathbf{x}_1) + \mathcal{E}_{\tilde{y}_1,\tilde{y}_2}[H(y|\mathbf{x}_2, \tilde{y}_1(\mathbf{x}_2), \tilde{y}_2(\mathbf{x}_2))]$
3. $E[H(y)] = \mathcal{E}_{\tilde{y}_1}[H(y|\mathbf{x}_1, \tilde{y}_1(\mathbf{x}_1))] + \mathcal{E}_{\tilde{y}_2}[H(y|\mathbf{x}_2, \tilde{y}_2(\mathbf{x}_2))]$
4. $E[H(y)] = \mathcal{E}_{\tilde{y}_1}[H(y|\mathbf{x}_2, \tilde{y}_1(\mathbf{x}_2))] + \mathcal{E}_{\tilde{y}_2}[H(y|\mathbf{x}_1, \tilde{y}_2(\mathbf{x}_1))]$

where $\tilde{y}_1(\mathbf{x})$ and $\tilde{y}_2(\mathbf{x})$ are the judgements of the first and second labeler on data point $\mathbf{x}$, respectively.

Among the four choices, the optimal strategy is to choose one that minimizes $E[H(y)]$. Using the conditional independence of our generative model, $P(y|\mathbf{x}, \tilde{y}_1(\mathbf{x}), \tilde{y}_2(\mathbf{x}))$, $P(y|\mathbf{x}, \tilde{y}_1(\mathbf{x}))$, $P(y|\mathbf{x}, \tilde{y}_2(\mathbf{x}))$, and $P(y|\mathbf{x})$ can all be computed using just $P(y|\mathbf{x})$, $P(\tilde{y}_1(\mathbf{x})|y)$, and $P(\tilde{y}_2(\mathbf{x})|y)$. In turn, $P(\tilde{y}_1(\mathbf{x})|y)$ and $P(\tilde{y}_2(\mathbf{x})|y)$ are simply functions of ϵ. So if one lets $P(y|\mathbf{x}_1)$ and $P(y|\mathbf{x}_2)$ be approximated by the classifier's current probabilistic beliefs $\hat{P}(y|\mathbf{x}_1)$ and $\hat{P}(y|\mathbf{x}_2)$, respectively, then one can compute $E[H(y)]$ under each labeling strategy and choose the optimal one. Figure 4, which we will refer to as the Labeling-Strategy Graph, shows the resulting decision regions for different ϵ's.

To understand the implications of the Labeling-Strategy Graph, we first examine the lower-left-to-upper-right diagonal, where the reliability of the two labelers are equal. There are really only three strategies to choose from, as Strategy 3 and 4 are equivalent. The white spaces denote the strategy where one labeler labels $\mathbf{x}_1$ and the other labels $\mathbf{x}_2$.

Obviously, when both labelers are perfect (that is, $\epsilon_1 = \epsilon_2 = 0$), they should each label a different data sample. This situation is represented by the completely white square at the lower-left corner of the Labeling-Strategy Graph. For imperfect labelers, we see that sometimes it is better for them to "collaborate" and label the same sample, in effect "truthing" each other's contribution. Specifically, when the classifier is fairly certain about the class of a data sample (i.e., $\hat{P}(y = \omega_1|\mathbf{x})$ is close to 0 or 1), there is little information to be gained from labeling it, and it is more productive for both labelers to examine the less certain sample. On the Labeling-Strategy Graph, we see that Strategy 1, in which both labelers judge $\mathbf{x}_1$, is optimal in the upper and lower regions where the classifier is confident in its classification of $\mathbf{x}_2$. Conversely, Strategy 2, in which both labelers judge $\mathbf{x}_2$, is optimal in the left and right regions where the classifier is confident in its classification of $\mathbf{x}_1$.

As the mislabeling rate of the labelers increases, one moves toward the upper right corner of the Labeling-Strategy Graph. The white region correspondingly shrinks. In other words, it becomes more productive for the labelers to collaborate and focus on getting the label for just one sample right. Except when the classifier has roughly equal uncertainty about the two samples, it tends to learn more by directing the labelers to the less certain sample.

Besides the diagonal, other interesting areas on the Labeling-Strategy Graph are the boundary conditions where either $\epsilon_1 = 0$ or $\epsilon_2 = 0$. Again the two labelers should examine different samples, as there is no point in having the imperfect labeler double-check the work of the perfect labeler. The choice is between Strategy 3 and 4. Looking at the Labeling-Strategy Graph, we see that the perfect labeler should always label the less certain sample, while the imperfect labeler should label the more certain one.

Even in cases where neither ϵ_1 or ϵ_2 are 0, the better (more reliable) labeler should always judge the less certain sample. This formalizes our intuitive tendency to assign more difficult tasks to experts while leverage non-experts to work on easier tasks. However, when the easier task becomes *too* easy (i.e., the classifier is very confident about the class of the sample), the less reliable labeler becomes more productive by joining the better labeler in tackling the harder task. That is, one enters the Strategy 1 or 2 areas that was discussed previously.

Related Work

The issues in using unreliable labelers to build pattern recognition systems are best highlighted by the Open Mind Initiative (Stork 2000). Under the Open Mind Initiative, the labelers are most likely unreliable because they are just un-

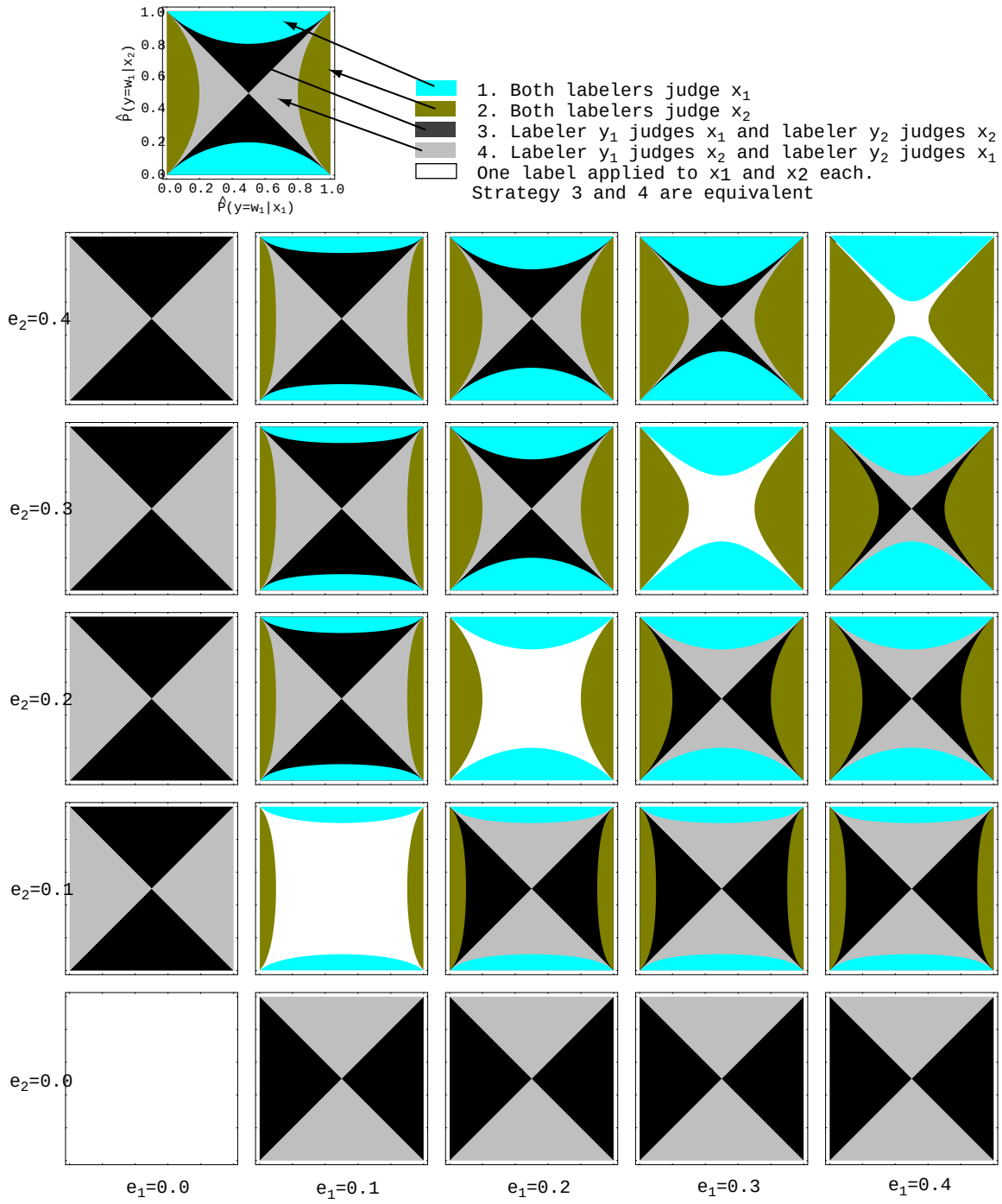

Figure 4: Labeling-Strategy Graph. The figure shows the optimal labeling strategy for two data points from two labelers who will each contribute one label. The contributors' mislabeling rates are ϵ_1 and ϵ_2. The classifier's current class estimates for the two data points are $\hat{P}(y = \omega_1|\mathbf{x}_1)$ and $\hat{P}(y = \omega_1|\mathbf{x}_2)$.

known volunteers gathered thru the Web. In a broader context, the general ability of unreliable information sources being aggregated to provide reliable, intelligent results has been demonstrated in practice by such projects as Wikipedia (http://www.wikipedia.org) and various decision markets (Pennock *et al.* 2001). Richardson and Domingos (2003) further explore the use of multiple weak experts to build the structure of Bayesian networks.

Learning from training data with unreliable labels has been investigated by many researchers (Angluin & Laird 1988; Katre & Krishnan 1989; Lugosi 1992) and the learning efficiency has been examined as well (Krishnan 1988; Lam & Stork 2003b). The main divergence of our approach is that we allow each data point to have *multiple* labels that can be gathered from experts of *different* reliability.

Active learning (Cohn, Ghahramani, & Jordan 1996) deals with selectively choosing data points for labeling, such that each labeling is most "educational" in terms of teaching a classifier. The labeling strategy examined in this paper shares the same goal. However, active learning has traditionally assumed a single perfect 'oracle' as labeler, while our approach deals with multiple unreliable labelers.

Dawid and Skene (1979) has used the same EM algorithm as ours to deal with observer errors. However, they do not use it to train a classifier. Their goal is to simply estimate the error rate of each observer (i.e., labeler), which our system can also do.

While this paper addresses some of the issues in *learning* from unreliable labelers, Lam and Stork (2003a) discusses methods for *evaluating* classifiers given unreliable labelers. The issue of handling noisy testing data in the specific case of text corpora is addressed in Blaheta (2002).

Future Work and Conclusion

This paper first presented a EM algorithm for training classifiers using multiple unreliable labelers. The learning algorithm allows each sample to have labels from zero, one, two, or more experts. As a side effect, the algorithm also gives a maximum-likelihood estimate of the error rate of each expert.

The second part of this paper considered the acquisition of each judgement from an expert to incur a fixed cost, and it proceeded to examine the optimal assignment of labelers to samples. Specifically, it illustrated the information-theoretic optimal strategy for the two-labeler, two-sample scenario. The optimal strategy is shown by the Labeling-Strategy Graph to match intuition: the more difficult sample should be examined by the better labeler, while the easier sample will be looked at by the less-skilled labeler, although it sometimes make sense to assign both labelers to judge the difficult sample if the other sample is relatively very much easier.

The theoretical exploration in this paper provided practitioners with guidelines for improving data acquisition in the context of building pattern recognition systems. Empirical examination remained for future work.

References

Angluin, D., and Laird, P. 1988. Learning from noisy examples. *Machine Learning* 2(4):343–370.

Blaheta, D. 2002. Handling noisy training and testing data. In *Proceedings of the 7th conference on Empirical Methods in Natural Language Processing (EMNLP)*, 111–116.

Cohn, D.; Ghahramani, Z.; and Jordan, M. 1996. Active learning with statistical models. *Journal of Artificial Intelligence Research* 4:129–145.

Cover, T. M., and Thomas, J. A. 1991. *Elements of Information Theory*. John Wiley & Sons.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics* 28(1):20–28.

Duda, R. O.; Hart, P. E.; and Stork, D. G. 2001. *Pattern Classification*. John Wiley & Sons, 2nd edition.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

Katre, U. A., and Krishnan, T. 1989. Pattern recognition with an imperfect supervisor. *Pattern Recognition* 22(4):423–431.

Krishnan, T. 1988. Efficiency of learning with imperfect supervision. *Pattern Recognition* 21(2):183–188.

Lam, C. P., and Stork, D. G. 2003a. Evaluating classifiers by means of test data with noisy labels. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, 513–518.

Lam, C. P., and Stork, D. G. 2003b. Upper bounds on learning rate with unreliable labelers. In *Proceedings of the Workshop on Distributed and Collaborative Knowledge Capture (DC-KCAP) at K-CAP*.

Lugosi, G. 1992. Learning with an unreliable teacher. *Pattern Recognition* 25(1):79–87.

McLachlan, G. J., and Krishnan, T. 1996. *The EM Algorithm and Extensions*. New York: Wiley-Interscience.

Nigam, K.; McCallum, A.; Thrun, S.; and Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning* 39(2):103–134.

Pennock, D. M.; Lawrence, S.; Nielsen, F. Å.; and Giles, C. L. 2001. Extracting collective probabilistic forecasts from web games. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, 174–183.

Richardson, M., and Domingos, P. 2003. Learning with knowledge from multiple experts. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, 624–631.

Smyth, P.; Fayyad, U. M.; Burl, M. C.; Perona, P.; and Baldi, P. 1995. Inferring ground truth from subjective labelling of Venus images. In Tesauro, G.; Touretzky, D.; and Leen, T., eds., *Advances in Neural Information Processing Systems*, volume 7, 1085–1092. Cambridge, MA: MIT Press.

Smyth, P. 1996. Bounds on the mean classification error rate of multiple experts. *Pattern Recognition Letters* 17(12):1253–1257.

Stork, D. G. 2000. Using open data collection for intelligent software. *Computer* 104–106.