Toward a computational theory of data acquisition and truthing

David G. Stork

Ricoh California Research Center
2882 Sand Hill Road Suite 115
Menlo Park, CA 94025-7022
stork@rii.ricoh.com

Abstract. *The creation of a pattern classifier requires choosing or creating a model, collecting training data and verifying or "truthing" this data, and then training and testing the classifier. In practice, individual steps in this sequence must be repeated a number of times before the classifier achieves acceptable performance. The majority of the research in computational learning theory addresses the issues associated with training the classifier (learnability, convergence times, generalization bounds, etc.). While there has been modest research effort on topics such as cost-based collection of data in the context of a particular classifier model, there remain numerous unsolved problems of practical importance associated with the collection and truthing of data. Many of these can be addressed with the formal methods of computational learning theory. A number of these issues, as well as new ones — such as the identification of "hostile" contributors and their data — are brought to light by the Open Mind Initiative, where data is openly contributed over the World Wide Web by non-experts of varying reliabilities. This paper states generalizations of formal results on the relative value of labeled and unlabeled data to the realistic case where a labeler is not a foolproof oracle but is instead somewhat unreliable and error-prone. It also summarizes formal results on strategies for presenting data to labelers of known reliability in order to obtain best estimates of model parameters. It concludes with a call for a rich, powerful and practical computational theory of data acquisition and truthing, built upon the concepts and techniques developed for studying general learning systems.*

Keywords: monitoring data quality, data truthing, open data collection, anomalous data detection, learning with queries, cost-based learning, Open Mind Initiative

1 Introduction

In broad outline, the creation of many practical systems to classify real-world patterns — such as acoustic speech, handwritten or omnifont optical characters, human faces, fingerprints, gestures, sonar images, and so on — involves the following steps:

select a model Select or design a computational model, specify its features, parameters and constraints or prior information about the unknown parameters

collect and verify training data Collect training data, verify or "truth" this data, and remove outliers and faulty data

train Train the model with this data, possibly employing regularization methods such as pruning, integrating multiple classifiers, or resampling methods such as boosting, and so on

test Test or estimate the performance of the classifier, either in the field, or more frequently in the lab using independent test data, to see if the classification performance is adequate for the application

These steps are not always followed in the sequence listed above (for instance, we may first collect our data before selecting a model), and in practice the steps are often repeated a number of times in an irregular order until the estimated performance of the classifier is acceptable.

The bulk of the research effort in computational learning theory, statistical learning theory and related discliplines has focused on model selection and training, and this has led to a wealth of powerful methods, including classifiers such as the nearest-neighbor method, neural nets, Support Vector Machines, and decision trees, regularization methods such as weight decay and pruning, general techniques such as multiclassifier integration and resampling, and theoretical results on learnability and convergence criteria, performance bounds, and much more [15].

But consider the databases of millions of labeled handwritten characters created by the National Institute of Standards and Technology (NIST), the immense volume of truthed postal data such as handwritten addresses and zip codes created by the Center for Excellence in Document Analysis and Recognition (CEDAR), or the transcriptions of tens of thousands of hours of speech created by the Linguistic Data Consortium (LDC), to mention but a few examples. These resources are invaluable to numerous groups developing classifiers and other intelligent software. The development of these databases requires a great deal of time, cost and effort, and relies on dozens of knowledge workers of varying expertise transcribing, checking, and cross-checking data in a model- and use-independent way.

Up to now, computational learning theory has contributed little to this vital process. In fact, most data acquisition teams rely on heuristics and trial and error, for instance in choosing the number of knowledge engineers that should truth a given dataset, how to monitor the reliability of individual engineers, and so on. Remarkably little of this information is published or otherwise shared. The goal of this paper is to begin to rectify this situation, by highlighting the need for large corpora of training data, describing some of the problems confronted in the creation of such datasets, suggesting results and techniques from computational learning theory that could be brought to bear, and providing some initial steps in the development of such a theory of data acquisition and truthing.

Section 2 reviews the need for data, or more specifically, the proposition that progress in classifier design will rely increasingly on larger and larger datasets and less and less on minor alterations to existing powerful learning techniques. This, then, underscores the need for theoretical effort on making more efficient the collection of high-quality datasets. Section 3 outlines some practical background and trends relevant to data acquisition and truthing. It describes in some detail a novel method of open data collection over the World Wide Web employed by the Open Mind Initiative. Section 4 illustrates several data collection and truthing scenarios and attendant practical problems ammenable to analysis through the tools of statistical and computational learning theory.

Section 5 reports two theoretical results relevant to data acquisition. The first is a generalization of the measure of the value of labeled and unlabeled data to the more realistic case when the labeler, rather than being a perfect oracle, instead has a probability of making a random labeling mistake. The second is a strategy for requesting labels from imperfect labelers that, under a number of natural conditions, optimizes an information criterion related to the quality of the resulting dataset. Conclusions and future directions are presented in Sect. 6.

2 The need for large datasets

Nearly all software projects in pattern classification and artificial intelligence — such as search engines and computer vision systems — require large sets of training data. For instance, state-of-the-art speech recognition systems are trained with hundreds or thousands of hours of speech sounds transcribed or "labeled" by knowledge engineers; leading optical character recognition systems are trained with pixel images of several million characters along with their transcriptions; one commercial effort at building a knowledge base of common sense information has required 500 person-years of effort over 17 years so far, most of this in data entry [14].

There is theoretical and experimental evidence that given sufficiently large sets of training data a broad range of classifier methods yield similar high performance. From a probabilistic viewpoint, we know from Bayesian estimation theory that given a classifier model general enough to represent the true underlying class-conditional probability distributions, sufficiently large training sets can dominate or "swamp" poor prior information, thereby yielding accurate classifiers [1, 7]. Moreover, just as the limitations imposed by the bias-variance dilemma in regression can be overcome with larger and larger data sets, so too the only way to overcome the analogous limitation imposed by the (boundary) bias-variance dilemma in classification is to increase the amount of training data [10]. Under reasonable conditions, virtually all sufficiently power training methods give improved estimates and classifiers as the amount of high-quality training data is increased.

Experimental evidence of the value of large data sets comes from numerous classification competitions, where systems trained with the largest data sets generally excel, and from corporations, which often expend more effort and re-

lated handwritten characters. They found that all three classifiers attained very nearly the same high accuracy and that the trained classifiers exhibited nearly the same pattern of misclassification errors [11]. They concluded, in short, that the information in sufficiently large training sets swamped biases and priors in their classifier models, and the implication is that this is a general result which holds so long as the fundamental classifier model is sufficiently general (low bias).

The above discussion is, of course, not an argument against efforts to find good models when building a classifier. Instead, it is a suggestion that builders of classifiers and AI systems should turn their attention to algorithms and theory that support the collection of large sets of accurately labeled data [12]. While computational learning theory may tell us how many patterns are needed for a given expect generalization error for example, such theory has provided little guidance on how to efficiently *collect* such data in a classifier- or use-independent way.

2.1 An example

We now turn to an extreme illustration of poor generalization resulting from training parameterized model that is too impoverished to accurately approximate the true underlying distributions [7, pages 142–143]. While admittedly hardly a proof, this surprising example illustrates that even when we use a principled estimation method such as maximum-likelihood, we can get terrible results. Specifically, even though our model space contains a classifier that would yield near perfect results ($error = 0\%$), our estimation procedure produces a classifier with the worst possible generation ($error = 100\%$).

Consider a one-dimensional, two-category classification problem with equal priors $P(\omega_1) = P(\omega_2) = 0.5$, and the following class-conditional densities:

$$\begin{aligned} p(x|\omega_1) &= (1-k)\delta(x-1) + k\delta(x+X) \\ p(x|\omega_2) &= (1-k)\delta(x+1) + k\delta(x-X) \end{aligned} \tag{1}$$

where $\delta(\cdot)$ is the familiar Dirac delta function, which vanishes when its argument is non-zero and integrates to 1.0, as shown in Fig. 1. The scalar k (where $0 < k < 0.5$) is small, and will shrink toward zero in our construction; further, X is a distance from the origin, which will grow in our construction. Note that these two class-conditional densities are interchanged under the reflection symmetry operation $x \leftrightarrow -x$.

Suppose we model these distributions by Gaussians parameterized by a mean and variance, that is, $p(x|\omega_i) \sim N(\mu_i, \sigma_i^2)$. This is admittedly a poor model in this case, nevertheless such a model is often used when there is little or no information about the underlying distributions. The maximum-likelihood estimate of the mean μ_1 is merely the mean of the data in ω_1 [7], that is, $\hat{\mu}_1 = (k+1) - kX$, and

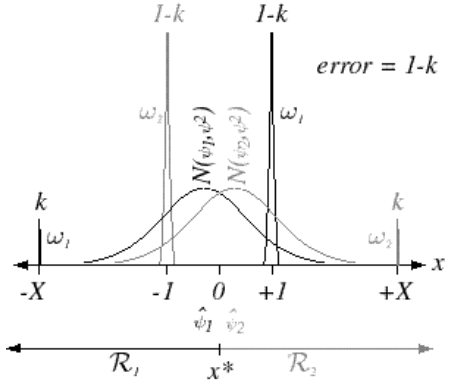

Fig. 1. A simple one-dimensional two-category classification problem in which a model's parameters are trained by maximum-likelihood methods on an infinitely large training set yields the worst possible classification ($error = 100\%$), even though the model space contains the best possible classifier ($error = 0\%$). The true or target (normalized) distribution for category ω_1 consists of a Dirac delta function at $x = +1$ of height $1 - k$, and a delta function at $x = -X$ of height k. The true distribution for category ω_2 is spatially symmetric to that of category ω_1, i.e., the one obtained under the interchange $x \leftrightarrow -x$, as given in Eq. 1 and shown in gray. The (poor) model for each distribution is a Gaussian, whose mean is estimated using an infinite amount of data sampled from $p(x|\omega_i)$ for $i = 1, 2$. For sufficiently large X, the estimated means obey $\hat{\mu}_1 < \hat{\mu}_2$, leading to an error of $1 - k$. If k is reduced and X increased accordingly, the training and generalization errors can be arbitrarily close to 100%.

analogously for $\hat{\mu}_2$. By the symmetry of the problem and estimation procedure, the (single) decision boundary will always be at $x^* = 0$.

For an arbitrary positive k, the estimated mean $\hat{\mu}_1$ is less than zero if $X > (k-1)/k$. Under equivalent conditions, the mean $\hat{\mu}_2$ is greater than zero. Informally speaking, in such a case the means have "switched positions" that is, $\hat{\mu}_2 > \hat{\mu}_1$. Thus the decision boundary is at $x^* = 0$ but the decision region for ω_1, i.e., $\mathcal{R}_1$, corresponds to all negative values of x, and $\mathcal{R}_2$ to all positive values of x. The error under this classification rule is clearly $1 - k$, which can be made arbitrarily close to 100% by letting $k \to 0$ and $X \to (k-1)/k + \epsilon$ where ϵ is an arbitrarily small positive number. Note that an infinite continuum of values of the parameters will yield a classifier with $error = 0\%$, specifically any for which $\hat{\mu}_2 > \hat{\mu}_1$ and $|\hat{\mu}_2| = |\hat{\mu}_1|$. (In fact, there are yet other values of the means that yield classifiers with $error = 0\%$, such as any that have equal variances, $\sigma_1^2 = \sigma_2^2$, and $\hat{\mu}_1 > \hat{\mu}_2$ with the intersection of the Gaussian densities lying between $x = -1$ and $x = +1$.)

This surprisingly poor classification performance is not an artifact of using limited training data or training to a poor local minimum in the likelihood function — in fact, neither of these are the case. Note too that even if the variances were parameters estimated from the data, because of the symmetry of the problem the decision boundary would remain at $x^* = 0$ and the *error* would be 100%. The informal lesson here is that even a well-founded estimation

method such as maximum-likelihood can give poor classifiers if our model space is poorly matched to the problem (high bias). In such cases we should expand the expressiveness of the models; this generally requires that we train using larger data sets.

3 The practice of collecting and truthing data

Given the manifest need for large data sets, we naturally ask: What are some of the sources of such vital data? Optical character recognition companies employ knowledge engineers whose sole task is to optically scan printed pages and then transcribe and truth the identities of words or characters [3]. Likewise, the Linguistic Data Consortium has dozens of knowledge engineers who transcribe recorded speech in a variety of languages on a wide variety of topics. Entering data by hand this way is often expensive and slow, however. An alternative approach, traditional data mining [8], is inadequate for many problem domains in part because data mining provides *unlabeled* data or because the data is simply not in an appropriate form. For instance the web lacks pixel images of handwritten characters and explicit common sense data and thus such information cannot be extracted by data mining. Moreover, accurately *labeled* data can be used in powerful *supervised learning* algorithms, while if the data is unlabeled only less-powerful *unsupervised learning* algorithms can be used. For this reason, we naturally seek inexpensive methods for collecting labeled data. Such a goal is virtually identical to that for transcribing audiotapes and videotapes.

As we shall see below, the internet can be used in a new way to gather needed labeled data: facilitating the collection of information contributed by humans.

3.1 Trends in open software and collaboration

Before we consider new methods for collecting and truthing data, we shall review some important trends. There are several compelling lessons from collaborative software projects that have major implications for systems supporting the collection of data. Consider the open source software movement, in which many programmers contribute software that is peer-reviewed and incorporated into large programs, such as the *Linux* operating system. Two specific trends must be noted. The first is that the average number of collaborators per project has increased over the past quarter century. For instance, in the late 1970s, most open collaborative software projects such as *emacs* involved several hundred programmers at most, while by the 1990s projects such as *Linux* involve over 100,000 software engineers. The second trend is that the average technical skill demanded of contributors has decreased over that same period. The programmers who contributed to *gcc* in the 1980s were experts in machine-level programming; the contributors to *Linux* know about file formats and device drivers; the contributors to the *Newhoo* collaborative open web directory need little if any technical background beyond an acquaintance with *HTML*.

3.2 The Open Mind Initiative

Let us review the following general facts and trends:

- pattern classifiers and intelligent software are improved with large sets of high-quality data
- open source software development techniques are applied increasingly to lower skilled collaborators
- open source development, and general collaborative projects, are expanding to larger groups as a result of the World Wide Web
- the internet can be used as an infrastructure for collecting data

These trends, and particularly the emergence of the World Wide Web, suggest that collaborative efforts can be extended to an extremely *large* pool of contributors (potentially anyone on the web), whose technical expertise can be *low* (merely the ability to point and click). These ideas were the inspiration underlying the creation of the Open Mind Initiative, the approach we now explore.

The central goal of the Open Mind Initiative (`www.OpenMind.org`) is to support non-expert web users contributing "informal" data needed for artificial intelligence and pattern recognition projects, as well as closely related tasks such as transcribing audio or video data. The Initiative thus extends the trends in open source software development to larger and larger groups of collaborators, allowing lower and lower levels of technical expertise. Moreover, the Initiative broadens the output of collaborative projects: while traditional open-source projects release software, the Initiative releases both software and data [17, 18].

A prototypical open data collection project in the Initiative is illustrated in skeleton form in Fig. 2. The project site contains a large database of isolated handwritten characters, scanned from documents, but whose character identities are not known. Individual segmented characters from this database are presented on standard web browsers of contributors who then identify or "label" the pattern by clicking buttons on a simple interface. These labelings are automatically sent to the project site, where they are collected and used to train software that classifies handwritten digits.

Some data acquisition projects in the Initiative could employ novel human-machine interfaces based on games. For instance, imagine an Open Mind Initiative chatbot project in which data is collected while contributors play a modified version of *Dungeons and Dragons*. In this new game, players read short texts — which discuss potions to drink, swords to brandish, rooms to enter, tasks to accomplish — generated by automated text generation programs. As part of the game, players must indicate how "natural" these texts are. This valuable feedback, collected at the project site, provides information for adjusting the parameters in the text generation programs, thereby yielding more natural generated text. In such game-based projects, contributors download the game software (presumably written in *Java*) from the project site. The data captured on the contributor's machine is stored locally and sent to the project site at the end of a game session.

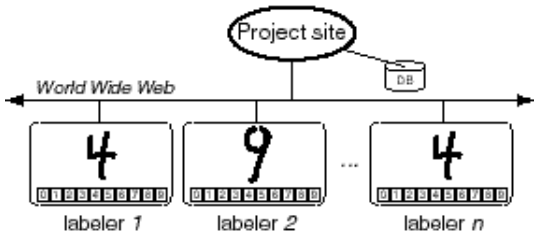

Fig. 2. This simplified, skeleton architecture shows the general approach in an open data collection project on isolated handwritten digit recognition. The unlabeled pixel images are presented on the browsers of non-expert web users, who indicate their judged category memberships by means of a button interface. Occasionally, the same pattern is presented to two or more independently selected contributors, to see if they agree; in this way, the reliability of contributors is monitored semi-automatically, and the quality of the data can be kept high.

While in most of the Initiative's projects contributors provide data through standard web browsers, in other projects contributors will require a more sophisticated human interface. For instance, in projects using a game interface, contributors will download the presentation and local cacheing software resident from the project site, and install it on their local machine. Data is collected while the contributor plays the game and is sent to the project home site at the end of a game session.

There are a number of incentives for people to contribute to Open Mind Initiative projects. Contributors seek benefit from the software (as in a text-to-speech generator); they enjoy game interfaces (as in online versions of *Dungeons and Dragons*); they seek public recognition for their contributions (as in *SETI@home*); they are interested in furthering the scientific goals of the project (as do amateur ornithologists through annual bird counts for the Audubon Society); they seek financial incentives such as lotteries, discounts, e-coupons or frequent-flier awards provided by third-party corporations [16].

The Open Mind Initiative differs from the Free Software Foundation and traditional open-source development in a number of ways. First, while open-source development relies on a hacker culture (e.g., roughly 10^5 programmers contributing to *Linux*), the Open Mind Initiative is instead based on a non-expert web user and business culture (e.g., 10^9 web users). While most of the work in open-source projects is directly on the final software to be released (e.g., source code), in the Initiative most of the effort is directed toward the tools, infrastructure and data gathering. Final decisions in open source are arbitrated by an expert or core group; in the Initiative contributed data is accepted or rejected automatically by software that is sensitive to anomalies or outliers. In some cases, data can be rejected semi-automatically, for instance by having data checked by two or more independently chosen contributors. Such "self-policing" not only helps to eliminate questionable or faulty *data*, it also helps to identify unreliable *contributors*, whose subsequent contributions can be monitored more closely or

blocked altogether, as we shall mention below. It must be emphasized that the Open Mind Initiative's approach also differs significantly from traditional data mining. In particular, in data mining a fixed amount of unlabeled information is extracted from an existing database (such as the web), whereas in the Initiative a possibly boundless amount of labeled data is *contributed*.

The Open Mind Initiative has four projects in progress: handwriting recognition, speech recognition and a small, demonstration AI project, *Animals*. These have been tested on intranets and are being debugged and load tested for full web deployment. The fourth project site, Open Mind common sense, is open and accepting contributed data over the web. As of May 2001, it has collected 400,000 common sense facts from 7000 separate contributors through a range of "activities," such as DESCRIBE A PICTURE and RELATE TWO WORDS. To date, data monitoring in this project has been semi-automatic whereby contributors "self-police" the contributions of each other.

4 Challenges and applications of a theory of data acquisition and truthing

Below are several scenarios and problems in data acquisition and truthing that are ammenable to computational theory, several are motivated by the challenges faced by the Open Mind Initiative. At base, many of these problems can be cast as learning the properties of the population of n labelers while simultaneously learning properties of the dataset.

- For open contributions of labels for handwritten characters, find the minimal conditions required to prove learnability of the character identities. This problem bears similarities to the approach of boosting, which will improve classification of weak learners [9]. Does the reliability of the contributors, weighted by the number of labels each provides, have to be greater than that of pure chance? Are there weaker conditions that nevertheless ensure learnability?
- A simple algorithm for improving the quality of contributed labels is "data voting," (or more generally "self-policing"), that is, presenting the same pattern to n_v labelers and accepting their majority vote. (This is related to the approach of collaborative filtering [4].) For a given total number of presentations of patterns to be labeled, if n_v is large, we collect a small amount of accurate data; conversely, if n_v is small, we get a large amount of less-accurate data. How do we set n_v to get a dataset that will lead to the most accurate classifiers? How does n_v change as the classifier is trained?
- How can we estimate the reliabilities of individual contributors while collecting data? How do we most efficiently identify "hostile" contributors, who seem to know the proper category identities, but deliberately submit false labels? (We assume that we can always associate a distinct identity with each contributor.)

- Given an estimate of such reliabilities and other properties of all n contributors, and given a set of unlabeled data and a partially trained classifier, how do we choose the single point from the data and a particular candidate labeler such that the returned label is expected to improve the classifier the most? This problem is more subtle than traditional active learning, which typically presumes the labeler is an omniscient oracle [6, 19] (and see Sect. 5.2).
- How can we find the contributors who are "complementary," that is, where the weaknesses of one match the strengths of the other. For instance, in truthing handwritten OCR, one contributor might be very accurate on numerals, another on text letters. Clearly it would be most efficient to pair these contributors on a large text, than to use two who are both strong on numerals alone or on text alone.
- Optimal strategies for storing data and delaying decisions on whether to use it. A contributed point may seem like an outlier or hostile earlier in the data collection process, but no so, later in the context of more data.
- Consider the problem of transcribing a videotape by a number n of transcribers, each with a possibly different (estimated) accuracy and expertise. Suppose we have some measure of the $n \times (n-1)$ correlations between their labelings on representative texts. How do we find the smallest subset of labelers that will yield some criterion accuracy, say 99.5%?

At first consideration it appears that data collection such as in the Open Mind Initiative's handwriting project is an example of stochastic game theory. After all, we treat the contributors as random processes, with variable reliabilities, and the host seeks to minimize a cost. In fact, though, stochastic game theory addresses games in which opponents form strategies that must take into account random processes, such as the roll of dice in backgammon or sequence of cards dealt in poker [2]. There seems to be little or no work on computing optimal strategies in arrangments such as the Open Mind framework.

Collectively, questions of this sort are not properly data mining either, where there is a large fixed data set without human intervention. While closely related to cost-based training (where there is a cost for collecting data given a particular classifier or learning algorithm), in many cases we are building a dataset or transcribing a text and do not know which classification algorithm will later be applied.

5 Two results in the theory of labeling

We now summarize two results, derived and explored more fully elsewhere, in the theory of data labelling [13].

5.1 The Fisher information of samples labeled by an unreliable labeler

Recall first the statistical *score*, V, a random variable defined by

$$V = \frac{\partial}{\partial \theta} \ln p(\mathcal{D}; \theta), \tag{2}$$

where the data set $\mathcal{D}$ is sampled from the density $p(\mathbf{x};\theta)$ where θ is a scalar parameter. The *Fisher information* $J(\theta)$ is then the variance of the score, that is,

$$J(\theta) = \mathcal{E}_\theta \left[\frac{\partial}{\partial \theta} \ln p(\mathcal{D};\theta) \right]^2 . \tag{3}$$

The *Cramér-Rao* inequality states that the mean-squared error of an unbiased estimator $F(\mathcal{D})$ of the parameter θ is bounded from below by the reciprocal of the Fisher information, that is,

$$\mathrm{Var}[F] \geq \frac{1}{J(\theta)}. \tag{4}$$

Informally, we can view the Fisher information as the information about θ that is present in the sample $\mathcal{D}$. The Fisher information gives a lower bound on the error when we estimate θ from the data, though there is no guarantee that there must always exist an estimator that achieves this bound.

The Fisher information of the prior probability $P(\omega_1) \equiv P_1$ chosen from a density $p(\mathbf{x}|\omega_1)$ was shown by Castelli and Cover [5] in the labeled and unlabled cases to be

$$J(P_1) = \frac{1}{P_1(1-P_1)} \tag{5}$$

$$J(P_1) = \int \frac{(p_1(\mathbf{x}) - p_2(\mathbf{x}))^2}{P_1 p_1(\mathbf{x}) + (1-P_1)p_2(\mathbf{x})} d\mathbf{x} \tag{6}$$

respectively. Lam and Stork [13] have generalized these results to the more realistic case where labelers are unreliable. We model such unreliability as if the labeler had perfect information and employed Bayes decision rule but then, with probability α (where $0 \leq \alpha \leq 1$), reported a *different* label. Under these conditions, the Fisher information is:

$$J(P_1) = \int \left[\frac{(\alpha p_1(\mathbf{x}) - (1-\alpha)p_2(\mathbf{x}))^2}{\alpha P_1 p_1(\mathbf{x}) + (1-\alpha)(1-P_1)p_2(\mathbf{x})} + \frac{((1-\alpha)p_1(\mathbf{x}) - \alpha p_2(\mathbf{x}))^2}{(1-\alpha)P_1 p_1(\mathbf{x}) + \alpha(1-P_1)p_2(\mathbf{x})} \right] d\mathbf{x}. \tag{7}$$

The case $\alpha = 0$ is equivalent to the labeled case above. The case $\alpha = 1$ corresponds to a "hostile contributor" who always willfully provides the wrong label. In the two-category case, however, the hostile contributor is in fact very helpful. All we need is a single, reliable bit of information, provided by a trusted expert for instance, to identify the true labels from the hostile data.

5.2 An optimal strategy for requesting labels

A general labeling strategy is an algorithm for deciding which unlabeled data points are to be presented to which of a set of n labelers given some information

about the labelers and the data in order to optimize some criterion. Consider the following specific case. Suppose we have two independent labelers, each known or assumed to have the same unreliability α. Suppose too that we have a set of unlabeled data in which each point is to be assigned one of two categories, ω_1 or ω_2. We have two unlabeled points, $\mathbf{x}_1$ and $\mathbf{x}_2$. Suppose we can exploit just two (total) labeling decisions from labelers, and our goal is to learn "as much as possible" under these conditions. Which pattern should be assigned to which labeler?

In this case, the natural measure of information to be learned is

$$\begin{aligned} I &= I(\omega|\mathbf{x}_1) + I(\omega|\mathbf{x}_2) \\ &= -\sum_{j=1}^{2} P(\omega_j|\mathbf{x}_1)\log_2 P(\omega_j|\mathbf{x}_1) - \sum_{j=1}^{2} P(\omega_j|\mathbf{x}_2)\log_2 P(\omega_j|\mathbf{x}_2), \end{aligned} \tag{8}$$

where $I(\omega|\mathbf{x}_i)$ is the information about the categories given a label on pattern $\mathbf{x}_i$ and the $P(\omega_j|\mathbf{x}_i)$ are probability estimates given by the current state of the classifier. The optimal strategy depends upon α and these estimated category memberships. Figure 3 summarizes the optimal strategy.

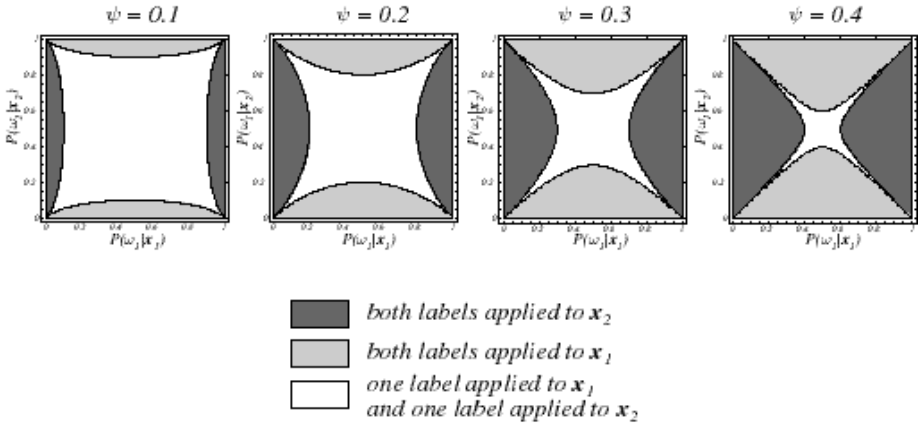

Fig. 3. The optimal data labeling strategy for two points $\mathbf{x}_1$ and $\mathbf{x}_2$ is illustrated for various levels of the contributor unreliability, described by α, and $P(\omega_1|\mathbf{x}_i)$ as given by a classifier. In the limit of small α, this strategy is to present one pattern to labeler 1 and one pattern to labeler 2. In the large-α case, the strategy is to present the most uncertain point (i.e., the one with $P(\omega_1|\mathbf{x}_i) \simeq 0.5$) to both labelers.

Examine the $\alpha = 0.1$ case. In this low-noise case, the labelers are reliable, and for most values of $P(\omega|\mathbf{x}_i)$ the optimal strategy is to request a label for $\mathbf{x}_1$ and for $\mathbf{x}_2$. However, if $P(\omega_1|\mathbf{x}_2)$ is very small (e.g., 0.05), then providing a label for $\mathbf{x}_2$ will not provide much information or refine our estimate of $P(\omega_1|\mathbf{x}_2)$. As such, our strategy in that case is to request both labelers to label $\mathbf{x}_1$, as shown by the light gray region.

In the high noise case, $\alpha = 0.4$, the range of values of estimated probabilities where we request separate points to be labeled separately is small. This is because we can gain more information by having two labels of a single point. In an extreme case $\alpha = 0.499$, not shown, then the labels are essentially the result of a coin toss, and provide no information. It is best, then, to apply both to the same point.

6 Future work

There remains much work to be done on the computational theory of data acquisition and truthing. No doubt, there are formal similarities between subcases of the data acquisition and truthing problem and cases in more traditional computational learning. We should explore and exploit such formal similarities. Nevertheless, the manifest importance of collecting high-quality data sets in a number of application environments provides great opportunities for developing useful theory leading to improved real-world systems.

Acknowledgements

I thank Chuck Lam for his many useful discussions, analyses and comments on this paper.

References

1. José M. Bernardo and Adrian F. M. Smith. *Bayesian Theory.* John Wiley and Sons, New York, NY, 1994.
2. David Blackwell and M. A. Girshick. *Theory of Games and Statistical Decisions.* Dover Publications, New York, NY, 1979.
3. Mindy Bokser, 1999. Personal communication (Caere Corporation).
4. John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Gregory F. Cooper and Serafín Moral, editors, *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pages 43–52, San Francisco, 1998. Morgan Kaufmann.
5. Vittorio Castelli and Thomas M. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *Proc. IEEE Transactions on Information Theory*, IT-42(6):2102–2117, 1996.
6. David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.
7. Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification.* Wiley, New York, NY, second edition, 2001.
8. Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramaswamy Uthurusamy, editors. *Advances in knowledge discovery and data mining.* MIT Press, Cambridge, MA, 1996.
9. Yoav Freund. Boosting a weak learning algorithm by majority. In *Proceedings of the Third Workshop on Computational Learning Theory*, pages 202–216, San Mateo, CA, 1990. Morgan Kaufmann.

10. Jerome H. Friedman. On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77, 1997.
11. Tin Kam Ho and Henry S. Baird. Large-scale simulation studies in pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-19(10):1067–1079, 1997.
12. Chuck Lam. *Open Data Acquisition: Theory and Experiments*. PhD thesis, Stanford University, Stanford, CA, 2002. in preparation.
13. Chuck Lam and David G. Stork. Optimal strategies for collecting and truthing data from contributors of varying reliabilities, 2001. submitted for publication.
14. Doug B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.
15. Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.
16. David G. Stork. The Open Mind Initiative. *IEEE Intelligent Systems & their applications*, 14(3):19–20, 1999.
17. David G. Stork. Open data collection for training intelligent software in the Open Mind Initiative. In *Proceedings of the Engineering Intelligent Systems Conference (EIS 2000)*, Paisley, Scotland, 2000.
18. David G. Stork. An architecture supporting the collection and monitoring of data openly contributed over the World Wide Web. In *Proceedings of the Workshop on Enabling Technologies, Infrastructure for Collaborative Enterprises (WET ICE)*, Cambridge, MA, 2001.
19. Richard Valliant, Alan H. Dorfman, and Richard M. Royall. *Finite population sampling and inference: A prediction approach*. Wiley Series in Probability and Statistics. John Wiley and Sons, New York, NY, 2000.