Proceedings of the 4th IEEE International Conference on Data Mining,
(ICDM-2004), Brighton, UK, November, 2004

Active Feature-Value Acquisition for Classifier Induction

Prem Melville
Dept. of Computer Sciences
Univ. of Texas at Austin
melville@cs.utexas.edu

Maytal Saar-Tsechansky
Red McCombs School of Business
Univ. of Texas at Austin
maytal@mail.utexas.edu

Foster Provost
Stern School of Business
New York University
fprovost@stern.nyu.edu

Raymond Mooney
Dept. of Computer Sciences
Univ. of Texas at Austin
mooney@cs.utexas.edu

Abstract

Many induction problems include missing data that can be acquired at a cost. For building accurate predictive models, acquiring complete information for all instances is often expensive or unnecessary, while acquiring information for a random subset of instances may not be most effective. Active feature-value acquisition *tries to reduce the cost of achieving a desired model accuracy by identifying instances for which obtaining complete information is most informative. We present an approach in which instances are selected for acquisition based on the current model's accuracy and its confidence in the prediction. Experimental results demonstrate that our approach can induce accurate models using substantially fewer feature-value acquisitions as compared to alternative policies.*

1 Introduction

Many predictive modeling tasks include missing data that can be acquired at a cost, such as customers' buying preferences and lifestyle information that can be obtained through an intermediary. For building accurate models, ignoring instances with missing values leads to inferior model performance [7], while acquiring complete information for all instances often is prohibitively expensive or unnecessary. To reduce the cost of information acquisition, it is desirable to identify instances for which complete information is most informative to acquire.

In this paper we address this problem of *active feature-value acquisition* (AFA) for classifier induction: given a feature acquisition budget, identify the instances with missing values for which acquiring complete feature information will result in the most accurate model. Formally, assume m instances, each represented by n features $a_1, ..., a_n$. For all instances, the values of a subset of the features $a_1, \ldots, a_i$ are known, along with the class labels. The values of the remaining features $a_{i+1}, \ldots, a_n$ are unknown and can be acquired at a cost. The problem of feature-value acquisition is different from active learning [2] and optimum experimental design [3], where the class labels rather than feature-values are missing and costly to obtain.

The approach we present for active feature acquisition is based on the following three observations: **(1)** Most classification models provide estimates of the confidence of classification, such as estimated probabilities of class membership. Therefore principles underlying existing active-learning methods like uncertainty sampling [2] can be applied. **(2)** For the data items subject to active feature-value acquisition, the correct classifications are known during training. Therefore, unlike with traditional active learning, it is possible to employ direct measures of the current model's accuracy for estimating the value of potential acquisitions. **(3)** Class labels are available for all complete and incomplete instances. Therefore, we can exploit all instances (including incomplete instances) to induce models, and to guide feature acquisition.

The approach we propose is simple-to-implement, computationally efficient and results in significant improvements compared to random sampling and a computationally-intensive method proposed earlier for this problem [11].

2 Task Definition and Algorithm

Pool-based Active Feature Acquisition: Assume a classifier induction problem, where each instance is represented with n feature values and a class label. For the set of complete instances G of the training set T, the values of all n features are known. For all other instances in T, only the values of a subset of the features $a_1, \ldots, a_i$ are known. The values of the remaining features $a_{i+1}, \ldots, a_n$ are missing and the set can be acquired at a fixed cost. We refer to these instances as incomplete instances, and the set is denoted as I. The class labels of all instances in T are known.

Unlike prior work [11], we assume that models are induced from the entire training set (rather than just from G). This is because models induced from all available data have been shown to be superior to models induced when instances with missing values are ignored [7]. [1] Beyond im-

[1] It was also noted in [11] that such a setting may result in better models.

proved accuracy, the choice of model induction setting also bears important implications for the acquisition mechanism, because the estimation of an acquisition's marginal utility is derived with respect to the model. Note that induction algorithms either include an internal mechanism for incorporating instances with missing feature-values [7] or require that missing values be imputed first. Henceforth, we assume that the induction algorithm includes some treatment for instances with missing values.

We study active feature-value acquisition policies within a generic iterative framework, shown in Algorithm 1. Each iteration estimates the utility of acquiring complete feature information for each incomplete example. The missing feature-values of a subset $S \in I$ of incomplete instances with the highest utility are acquired and added to T (these examples move from I to G). A new model is then induced from T, and the process is repeated. Different AFA policies correspond to different measures of utility. Our baseline policy, random sampling, selects acquisitions at random, which tends to select a representative set of examples [8].

Error Sampling: For a model trained on incomplete instances, acquiring missing feature-values is effective if it enables a learner to capture additional discriminative patterns that improve the model's prediction. Specifically, acquired feature-values are likely to have an impact on subsequent model induction when the acquired values pertain to a misclassified example and may embed predictive patterns that can be potentially captured by the model and improve the model. In contrast, acquiring feature-values of instances for which the current model already embeds correct discriminative patterns is not likely to impact model accuracy considerably. Motivated by this reasoning, our approach *Error Sampling* prefers to acquire feature-values for instances that the current model misclassifies. At each iteration, it randomly selects m incomplete instances that have been misclassified by the model. If there are fewer than m misclassified instances, then *Error Sampling* selects the remaining instances based on the *Uncertainty* score which we describe next. The uncertainty principle originated in work on optimum experimental design [3] and has been extensively applied in the active learning literature [2, 8]. The *Uncertainty* score captures the model's ability to distinguish between cases of different classes and prefers acquiring information regarding instances whose predictions are most uncertain. The acquisition of additional information for these cases is more likely to impact prediction, whereas information pertaining to strong discriminative patterns captured by the model is less likely to change the model. For a probabilistic model, the absence of discriminative patterns in the data results in the model assigning similar likelihoods for class membership of different classes. Hence, the *Uncertainty* score is calculated as the absolute difference between the estimated class probabilities of the two most likely classes. Formally, for an instance x, let $P_y(x)$ be the estimated probability that x belongs to class y as predicted by the model. Then the *Uncertainty* score is given by $P_{y_1}(x) - P_{y_2}(x)$, where $P_{y_1}(x)$ and $P_{y_2}(x)$ are the first-highest and second-highest predicted probability estimates respectively. Formally, the *Error Sampling* score for a potential acquisition is set to -1 for misclassified instances; and for correctly classified instances we employ the *Uncertainty* score. At each iteration of the AFA algorithm, complete feature information is acquired for the m incomplete instances with the lowest scores.

Algorithm 1 Active Feature-Value Acquisition Framework

Given:
G - set of complete instances
I - set of incomplete instances
T - set of training instances, $G \cup I$
$\mathcal{L}$ - learning algorithm
m - size of each sample

1. Repeat until stopping criterion is met
2. Generate a classifier, $C = \mathcal{L}(T)$
3. $\forall x_j \in I$, compute $Score(C, x_j)$ based on the current classifier
4. Select a subset S of m instances with the highest utility based on the score
5. Acquire values for missing features for each instance in S
6. Remove instances in S from I and add to G
7. Update training set, $T = G \cup I$
8. Return $\mathcal{L}(T)$

3 Experimental Evaluation

Methodology: We first compared *Error Sampling* to random feature acquisition. The performance of each system was averaged over 5 runs of 10-fold cross-validation. In each fold, the learner initially has access to all incomplete instances, and is given complete feature-values for a randomly selected subset of size m. For the active strategies, a sample of instances is then selected from the pool of incomplete instances based on the measure of utility. The missing values for these instances are acquired and the process is repeated until the pool of incomplete instances is exhausted. In the case of random sampling, the incomplete instances are selected uniformly at random. Each system is evaluated on the held-out test set after each iteration of feature acquisition. As in [11], the test data set contains only complete instances, since we want to estimate the true accuracy of the model given complete data. To maximize the gains of AFA, it is best to acquire features for a single instance in each iteration; however, to make our experiments computationally feasible, we selected instances in batches of 10 (i.e., sample

size m = 10).

To compare the performance of any two schemes, A and B we compute the percentage reduction in error of A over B for a given number of acquisitions and report the average over all points on the learning curve. The reduction in error is considered to be *significant* if the average errors across the points on the learning curve of A is lower than that of B according to a paired t-test ($p < 0.05$).

All the experiments were run on 5 web-usage datasets (used in [6]) and 5 datasets from the UCI machine learning repository [1]. [2] The web-usage data contain information from popular on-line retailers about customer behavior and purchases. This data exhibit a natural dichotomy with a subset of features owned by a particular retailer and a set of features that the retailer may acquire at a cost. The learning task is to induce models to predict whether a customer will purchase an item during a visit to the store. Hence the pool of incomplete instances was initialized with the features privately owned by each retailer. For the UCI datasets, 30% of the features were randomly selected to be used in the incomplete instances. A different set of randomly selected features was used for each train-test split of the data.

The active framework we have proposed can be implemented using an arbitrary probabilistic classifier as a learner. For the results in this paper, we used J48, which is the Weka implementation of C4.5 decision-tree induction [10].

Results: The results comparing *Error Sampling* to random sampling are summarized in Table 1. All error reductions reported are statistically significant. As mentioned above, the main impact of AFA is lower on the learning curve. To capture this, we also report the percentage error reduction averaged over only the 20% of points on the learning curve where the largest improvements are produced. We refer to this as the *top-20% percentage error reduction*, which is similar to a measure reported in [8].

The results show that for all data sets using *Error Sampling* significantly improves on the model accuracy compared to random sampling. Figures 1 and 2 present learning curves that demonstrate the advantage of using an AFA scheme over random acquisition. Apart from average reduction in error, a good indicator of the effectiveness of an AFA scheme is the number of acquisitions required to obtain a desired accuracy. For example, on the *qvc* dataset once *Error Sampling* acquires approximately 400 complete instances, it induces a model with an accuracy of 87%; however, random sampling requires approximately 1200 complete instances to achieve the same accuracy. We also evaluated a policy that uses only the *Uncertainty* score for estimating the utility of potential acquisitions. This *Uncertainty Sampling* results in significantly better performance compared to random sampling, but is inferior to *Error Sampling*. Detailed results comparing alternative AFA policies can be found in the extended version of this paper [5].

[2] The details of the datasets used can be found in [5].

Table 1. Error reduction of *Error Sampling* with respect to random sampling.

Dataset	%Error Reduction	Top-20% %Err. Red.
bmg	10.67	17.77
etoys	10.34	23.88
expedia	19.83	29.12
priceline	24.45	34.49
qvc	15.44	24.75
anneal	22.65	49.27
soybean	8.03	14.79
autos	4.24	10.50
kr-vs-kr	36.82	53.23
hypo	16.79	40.48
Mean	16.93	29.83

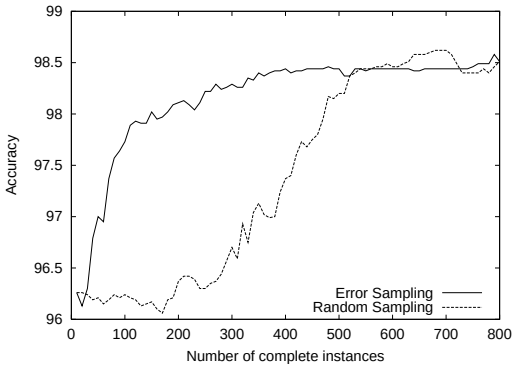

Figure 1. *Error Sampling* **vs.** *Random Sampling* **on** *anneal***.**

Comparison with GODA: The most closely related work to this paper is the study by Zheng and Padmanabhan [11] of the active feature-value acquisition scheme GODA. GODA measures the utility of acquiring feature-values for a particular incomplete instance in the following way. It adds the instance to the training set, imputing the values that are missing and then induces a new model. The instance that leads to the model with the best performance on the complete training data is selected for acquisition. GODA has two important differences from *Error Sampling*: it employs a different utility measure and it induces its models from only the complete instances. To compare to our approach, we implemented GODA as described in [11], using J48 tree induction as the learner and *multiple imputation* for missing value imputation. Experiments comparing *Error Sampling* to GODA were run as before; however, due to GODA's tremendous computational requirements, we only ran one run of 10-fold cross-validation on 5 of the datasets. Some

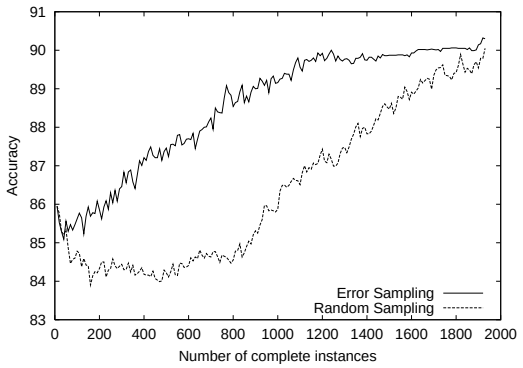

Figure 2. *Error Sampling* **vs.** *Random Sampling* **on** *qvc***.**

datasets were also reduced in size. A summary of the results, along with the reduced dataset sizes, is presented in Table 2. The results show that in spite of the high computational complexity of GODA, it results in inferior performance compared to *Error Sampling* for all 5 domains. All improvements obtained by *Error Sampling* with respect to GODA are statistically significant. These results suggest that the ability of *Error Sampling* to capitalize on information from incomplete instances, and to utilize this knowledge in feature acquisition, allows it to capture better predictive patterns compared to those captured by GODA.

Table 2. Error reduction of *Error Sampling* **with respect to** GODA**.**

Dataset	Size	% Error Reduction
etoys	270	37.58
priceline	447	14.19
bmg	200	19.48
expedia	200	22.96
qvc	100	20.03

4 Related Work and Conclusions

Recent work on *budgeted learning* [4] also addresses the issue of active feature-value acquisition. However, the policies developed in [4] assume feature-values are discrete, and consider the acquisition of individual feature-values for randomly selected instances of a given class, rather than for specific incomplete instances. Some work on *cost sensitive* learning [9] has addressed the issue of inducing economical classifiers but it assumes that the *training* data are complete and focuses on learning classifiers that minimize the cost of classifying incomplete *test* instances. Traditional *active learning* [2] assumes access to unlabeled instances with complete feature-values and attempts to select the most useful examples for which to acquire class labels. Active feature acquisition is a complementary problem that assumes labeled data and attempts to acquire the most useful feature-values.

We have presented a general framework for active feature acquisition that can be applied to different learners and can use alternate measures of utility for ranking acquisitions. Within this framework, we propose an effective and simple-to-implement policy that results in superior accuracy and is also significantly more efficient computationally compared to an existing approach.

Acknowledgments

We would like to thank Balaji Padmanabhan and Zhiqiang Zheng for providing us with the web usage datasets. Prem Melville and Raymond Mooney were supported by DARPA grant HR0011-04-1-007.

References

[1] C. L. Blake and C. J. Merz. UCI repository of machine learning databases. http://www.ics.uci.edu/˜mlearn/MLRepository.html, 1998.

[2] D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.

[3] V. Federov. *Theory of optimal experiments*. Academic Press, 1972.

[4] D. Lizotte, O. Madani, and R. Greiner. Budgeted learning of naive-bayes classifiers. In *Proc. of 19th Conf. on Uncertainty in Artificial Intelligence (UAI-03)*, Acapulco, Mexico, 2003.

[5] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. Active feature acquisition for classifier induction. Technical Report UT-AI-TR-04-311, University of Texas at Austin, 2004.

[6] B. Padmanabhan, Z. Zheng, and S. O. Kimbrough. Personalization from incomplete data: what you don't know can hurt. In *Proc. of 7th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD-2001)*, pages 154–163, 2001.

[7] J. R. Quinlan. Unknown attribute values in induction. In *Proc. of 6th Intl. Workshop on Machine Learning*, pages 164–168, Ithaca, NY, June 1989.

[8] M. Saar-Tsechansky and F. Provost. Active sampling for class probability estimation and ranking. *Machine Learning*, 54:153–178, 2004.

[9] P. D. Turney. Types of cost in inductive concept learning. In *Proc. of the Workshop on Cost-Sensitive Learning at the 17th Intl. Conf. on Machine Learning*, Palo Alto, CA, 2000.

[10] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, San Francisco, 1999.

[11] Z. Zheng and B. Padmanabhan. On active learning for data acquisition. In *Proc. of IEEE Intl. Conf. on Data Mining*, 2002.