To Appear in *Proceedings of the 5th International Conference on Data Mining (ICDM-2005)*

An Expected Utility Approach to Active Feature-value Acquisition

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

In many classification tasks training data have missing feature values that can be acquired at a cost. For building accurate predictive models, acquiring all missing values is often prohibitively expensive or unnecessary, while acquiring a random subset of feature values may not be most effective. The goal of active feature-value acquisition *is to incrementally select feature values that are most cost-effective for improving the model's accuracy. We present an approach that acquires feature values for inducing a classification model based on an estimation of the expected improvement in model accuracy per unit cost. Experimental results demonstrate that our approach consistently reduces the cost of producing a model of a desired accuracy compared to random feature acquisitions.*

1 Introduction

In many modeling problems, feature values for training data are missing, but can be acquired at a cost. In general, the cost of acquiring the missing information may vary according to the nature of the information or of the instance for which information is missing. Consider, for example, patient data used to induce a model to predict treatment effectiveness for a given patient. While missing demographic information can be obtained at a low cost, missing test results can be significantly more expensive to obtain. Various solutions are available for learning models from incomplete data, such as imputation methods [4]. However, these solutions almost always undermine model performance as compared to that of a model induced from complete information. Since obtaining all missing values may be prohibitively expensive, it is desirable to identify the information that would be most cost-effective to acquire. In this paper we address this generalized version of the *active feature-value acquisition* (AFA) task for classifier induction [6]: given a model built on incomplete training data, select feature values that would be most cost-effective to acquire for improving the model's accuracy.

Unlike prior work [5], we study AFA in a setting where the total cost to be spent on acquisitions is not determined *a priori*, but rather can be determined on-line based on the model's performance as learning progresses. We propose a general setting for AFA that specifies an incremental acquisition schedule. The solution we propose ranks alternative feature-value acquisitions based on an estimation of the expected improvement in model performance per unit cost. Our approach is general, i.e., it can be applied to select acquisitions for any learner, and to attempt to improve any performance metric. Experimental results on decision tree induction demonstrate that our method does consistently result in significantly improved model accuracy per unit cost compared to random feature-value acquisition.

2 Task Definition and Algorithm

Active Feature Acquisition: Assume a classifier induction problem where each instance is represented with n feature values and a class label. A training set of m instances can be represented by the matrix F, where $F_{i,j}$ corresponds to the value of the j-th feature of the i-th instance. Initially, the class label, y_i, of each instance is known, and the matrix F is incomplete, i.e., it contains missing values. The learner may acquire the value of $F_{i,j}$ at the cost $C_{i,j}$. We use $q_{i,j}$ to refer to the query for the value of $F_{i,j}$. The general task of active feature-value acquisition is the selection of these instance-feature queries that will result in building the most accurate model (classifier) at the lowest cost. The framework for the generalized AFA task is presented in Algorithm 1. We view AFA as an iterative task, where at each step the learner builds a classifier trained on the current data, and scores the available queries based on this classifier. The query with the highest score is selected and the feature value corresponding to this query is acquired. The training data is appropriately updated and this process is repeated until some stopping criterion is met, e.g. a desirable model accuracy has been obtained. To reduce computation costs in our experiments, we acquire queries in fixed-size batches at each iteration.

Algorithm 1 General Active Feature-value Acquisition Framework

```
Given:
$F$ – initial (incomplete) instance-feature matrix
$Y = \{y_i : i = 1, ..., m\}$ – class labels for all instances
$T$ – training set = $< F, Y >$
$\mathcal{L}$ – base learning algorithm
$b$ – size of query batch
$C$ – cost matrix for all instance-feature pairs

Initialize set of possible queries $Q$ to $\{q_{i,j} : i = 1, ..., m; j = 1, ..., n$; such that $F_{i,j}$ is missing$\}$
Repeat until stopping criterion is met
    Generate a classifier, $M = \mathcal{L}(T)$
    $\forall q_{i,j} \in Q$ compute $score(M, q_{i,j}, C_{i,j}, \mathcal{L}, T)$
    Select a subset $S$ of $b$ queries with the highest $score$
    $\forall q_{i,j} \in S$,
        Acquire values for $F_{i,j}$
    Remove $S$ from $Q$
Return $M = \mathcal{L}(T)$
```

Expected Utility Estimation: Specific solutions to the AFA problem differ based on the method used to score and rank queries. In our approach, we provide scores based on the *expected utility* of each query (defined below). For now we assume all features are nominal, i.e., they can take on values from a finite set of values. Assume feature j has K distinct values $V_1, ..., V_K$. The expected utility of the query $q_{i,j}$ can be computed as:

$$E(q_{i,j}) = \sum_{k=1}^{K} P(F_{i,j} = V_k)\mathcal{U}(F_{i,j} = V_k) \qquad (1)$$

where $P(F_{i,j} = V_k)$ is the probability that $F_{i,j}$ has the value V_k, and $\mathcal{U}(F_{i,j} = V_k)$ is the utility of knowing that the feature value $F_{i,j}$ is V_k, given by:

$$\mathcal{U}(F_{i,j} = V_k) = \frac{\mathcal{A}(F, F_{i,j} = V_k) - \mathcal{A}(F)}{C_{i,j}} \qquad (2)$$

where $\mathcal{A}(F)$ is the accuracy of the current classifier; $\mathcal{A}(F, F_{i,j} = V_k)$ is the accuracy of the classifier trained on F assuming $F_{i,j} = V_k$; and $C_{i,j}$ is the cost of acquiring $F_{i,j}$. For this paper, we define the utility of an acquisition in terms of improvement in model accuracy per unit cost. Depending on the objective of learning a classifier, alternate utility functions could be used. If we were to plot a graph of accuracy versus model cost after every iteration of AFA, our *Expected Utility* approach would correspond to selecting the query that is expected to result in the largest slope for the next iteration. If all feature costs are equal, this corresponds to selecting the query that would result in the classifier with the highest expected accuracy.

Since the true distribution of each missing feature value is unknown, we estimate $P(F_{i,j} = V_k)$ in Eq. 1 using a learner that produces class probability estimates. For each feature j, we train a classifier M_j, using this feature as the target variable and all other features along with the class as the predictors. When evaluating the query $q_{i,j}$, the classifier M_j is applied to instance i to produce the estimate $\hat{P}(F_{i,j} = V_k)$. In Eq. 2, the true values of $\mathcal{A}(.)$ are also unknown. However, since the class labels for the training data are available at selection time we can estimate $\mathcal{A}(F)$ and $\mathcal{A}(F, F_{i,j} = V_k)$ based on the training set accuracy. In our experiments, we used 0-1 loss to measure the accuracy of the classifiers. However, other measures such as class entropy or GINI index could also be used [5]. In our preliminary studies we did not observe a consistent advantage to using entropy.

When the *Expected Utility* method described here is applied to learn a Naive Bayes classifier and feature costs are assumed to be equal, it is similar to the *greedy loss reduction* approach presented in [5]. Similar approaches to expected utility estimation have also been used in the related task of traditional active learning [8].

Computing the estimated expectation $\hat{E}(.)$ for query $q_{i,j}$ requires training one classifier for each possible value of feature j. Selecting the best from *all* available queries would require exploring, in the worst case, mn queries. So exhaustively selecting a query that maximizes the expected utility is computationally very intensive and is infeasible for most interesting problems. We make this exploration tractable by reducing the search space to a random sub-sample of the available queries. We refer to this approach as *Sampled Expected Utility*. This method takes a parameter α ($1 \leq \alpha \leq \frac{mn}{b}$) which controls the complexity of the search. To select a batch of b queries, first a random sub-sample of αb queries is selected from the available pool, and then the expected utility of each query in this sub-sample is evaluated. The value of α can be set depending on the amount of time the user is willing to spend on this process. One can expect a tradeoff between the amount of time spent and the effectiveness of the selection scheme.

3 Experimental Evaluation

Methodology: We evaluated our proposed approach on four datasets from the UCI repository [1] – *car*, *audio*, *lymph*, and *vote*. For the sake of simplicity, we selected datasets that have only nominal features. None of the UCI datasets provide feature acquisition costs; so in our experiments we simply assume all costs are equal. For additional experiments with different cost structures, please refer to the extended version of this paper [7].

We compared our approach to *random feature acquisition*, which selects queries uniformly at random to provide a representative sample of missing values. For *Sampled Expected Utility* we set the exploration parameter α to 10.

Given the computational complexity of *Expected Utility* it is not feasible to run the exhaustive *Expected Utility* approach on all datasets. However, we did run *Expected Utility* on the *vote* dataset. For all methods, as a base learner we used J48 decision-tree induction, which is the Weka implementation of C4.5 [10]. Laplace smoothing was used with J48 to improve class probability estimates.

The performance of each acquisition scheme was averaged over 10 runs of 10-fold cross-validation. In each fold of cross-validation, we generated learning curves in the following fashion. Initially, the learner is given a random sample of feature values, i.e. the instance-feature matrix is partially filled. The remaining instance-feature pairs are used to initialize the pool of available queries. At each iteration, the system selects a batch of queries, and the values for these features are acquired. This process is repeated until a desired number of feature values is acquired. Classification accuracy is measured after each batch acquisition in order to generate a learning curve. One system (A) is considered to be *significantly* better than another system (B) if the average accuracy across the points on the learning curve of A is higher than that of B according to a paired t-test ($p < 0.05$). As in [6], the test data contains only complete instances, since we want to approximate the true generalization accuracy of the constructed model given complete data for a test instance. For each dataset, we selected the initial random sample size to be such that the induced model performed at least better than majority class prediction. The batch size for the queries was selected based on the difficulty of the dataset. For problems that were harder to learn, we acquired a larger number of feature-values and consequently used larger batch sizes.

Results: Our results are presented in Figure 1. For all datasets, *Sampled Expected Utility* builds more accurate models than random sampling for any given number of feature acquisitions. These results demonstrate that the estimation of the expected improvement in the current model's accuracy enables effective ranking of potential queries. Consequently, *Sampled Expected Utility* selects queries that on average are more informative for the learner than an average query selected at random. The differences in performance between these two systems on all datasets is significant, as defined above. Since *Sampled Expected Utility* was proposed in order to reduce the computational costs of our original *Expected Utility* approach, we also examined the performance and computational time of the exhaustive *Expected Utility* algorithm for *vote*. We computed the average time it took to select queries in each iteration for each of the methods. We observed that the average selection time for *Expected Utility*, *Sampled Expected Utility* and random sampling was 3.77×10^5, 6.64×10^3, and 3.8 milliseconds respectively. These results show that constraining the search in *Expected Utility* by random sampling can significantly reduce the selection time (by two orders of magnitude in this case) without a significant loss in accuracy.

Additional experiments (not presented here) with different cost structures demonstrate that for the same cost, *Sampled Expected Utility* builds more accurate classifiers than the cost-agnostic random feature acquisition approach. Its performance is also more consistent than that of a simple cost-sensitive method which acquires feature values in order of increasing cost. Details of these results may be found in [7].

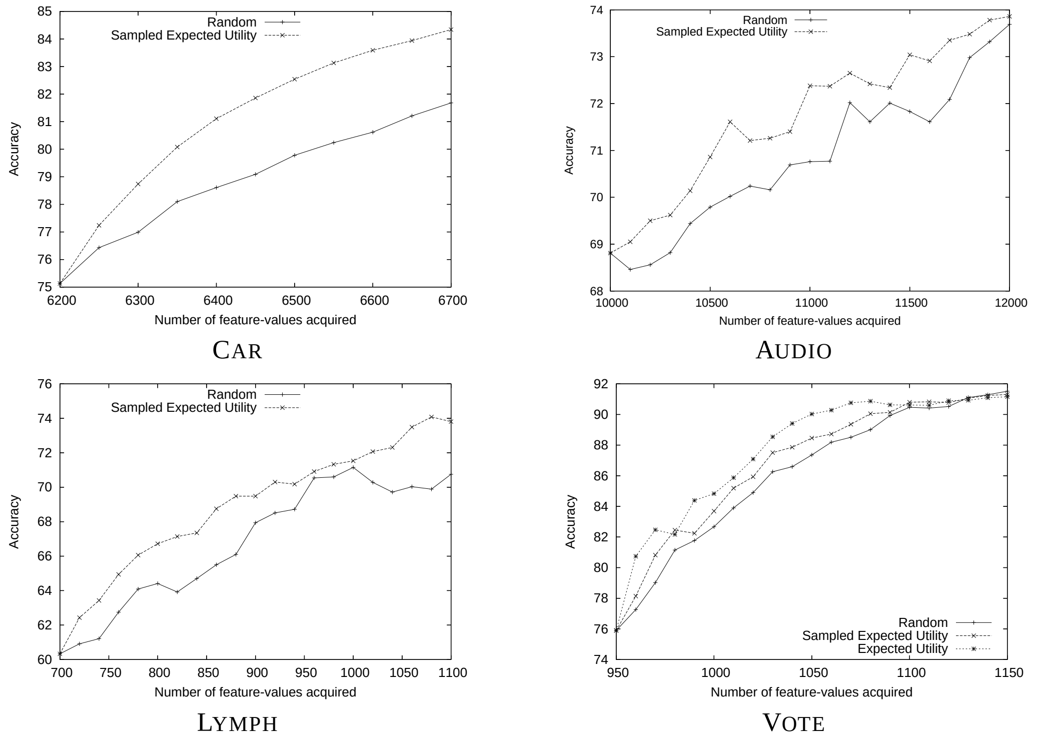
4 Related Work

Lizotte et al. [5] study AFA in the *budgeted learning* scenario, in which the total cost to be spent towards acquisitions is determined *a priori* and the task is to identify the best set of acquisitions for this cost. In contrast, our setting aims to enable the user to stop the acquisition process at any time, and as such the *order* in which acquisitions are made is important. Given this criterion, we attempt to select the next acquisition that will result in the most accurate model per unit cost. Other work on AFA has focused on the *instance-completion* setting, in which all missing feature values are acquired for a selected training instance [11, 6]. The instance-completion methods estimate the utility of having complete feature-value information for a particular instance, and unlike our approach, do not evaluate acquisitions of individual feature values. However, our method can also be used in the instance-completion setting, e.g., by selecting the instance with the highest sum of utilities of individual feature-value acquisitions.

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 data and attempts to select the most useful examples for which to acquire class labels. Active feature-value acquisition is a complementary problem that assumes labeled data with incomplete feature data and attempts to select the most useful additional feature values to acquire.

5 Future Work and Conclusions

In *Sampled Expected Utility* we use a random sample of the pool of available queries to make the *Expected Utility* estimation feasible. However, it may be possible to improve performance by applying *Expected Utility* estimation to a sample of queries that is better than a random sample. One approach could be to first identify potentially informative instances, and then select candidate queries only from these



CAR

AUDIO

LYMPH

VOTE

Figure 1. Comparing alternative active feature-value acquisition approaches.

instances. Such instances can be identified using methods proposed for the instance-completion setting for AFA, such as *Error Sampling* [6]. Preliminary results in this direction can be found in [7]. An alternative approach could be to restrict the set of candidate queries to only the most informative features. A subset of such features could be picked using a *feature selection* technique that can capture the interactions among feature values, such as the wrapper approach of John et al. [3].

In this paper, we propose an expected utility approach to active feature-value acquisition, that obtains feature values based on the estimated expected improvement in model accuracy per unit cost. We demonstrate how this computationally intensive method can be made significantly faster, without much loss in performance, by constraining the search to a sub-sample of potential feature-value acquisitions. Experiments with uniform feature costs show that this *Sampled Expected Utility* approach consistently builds more accurate models than random sampling for the same number of feature-value acquisitions.

Acknowledgments

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. 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] G. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In *Proc. of 11th Intl. Conf. on Machine Learning*, pages 121–129, 1994.

[4] R. Little and D. Rubin. *Statistical Analysis with Missing Data*. John Wiley and Sons, 1987.

[5] D. Lizotte, O. Madani, and R. Greiner. Budgeted learning of naive-Bayes classifiers. In *Proc. of 19th Conf. on Uncertainty in Artificial Intelligence*, 2003.

[6] Prem Melville, Maytal Saar-Tsechansky, Foster Provost, and Raymond Mooney. Active feature-value acquisition for classifier induction. In *Proc. of 4th IEEE Intl. Conf. on Data Mining (ICDM-04)*, 2004.

[7] Prem Melville, Maytal Saar-Tsechansky, Foster Provost, and Raymond Mooney. Economical active feature-value acquisition through expected utility estimation. In *Proc. of the KDD-05 Workshop on Utility-Based Data Mining*, 2005.

[8] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proc. of ICML-2001*, pages 441–448, 2001.

[9] P. D. Turney. Types of cost in inductive concept learning. In *Proc. of ICML Workshop on Cost-Sensitive Learning*, 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 Intl. Conf. on Data Mining*, 2002.