Prediction-time Active Feature-value Acquisition for Cost-effective Customer Targeting

Pallika Kanani
Department of Computer Science
University of Massachusetts, Amherst
Amherst, MA 01002
pallika@cs.umass.edu

Prem Melville
IBM T.J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
pmelvil@us.ibm.com

Abstract

In general, the prediction capability of classification models can be enhanced by acquiring additional relevant features for instances. However, in many cases, there is a significant cost associated with this additional information — driving the need for an intelligent acquisition strategy. Motivated by real-world customer targeting domains, we consider the setting where a fixed set of additional features can be acquired for a subset of the instances at test time. We study different acquisition strategies of selecting instances for which to acquire more information, so as to obtain the most improvement in prediction performance per unit cost. We apply our methods to various targeting datasets and show that we can achieve a better prediction performance by actively acquiring features for only a small subset of instances, compared to a random-sampling baseline.

1 Introduction

In many data mining domains, one is not presented with complete and definitive training data for modeling. Instead, several sources exists from which data can be acquired, usually incurring some acquisition or processing costs. The cost-effective acquisition of such selected data for modeling has been an emerging area of study which, in the most general case, is referred to as Active Information Acquisition [13]. Actively selecting feature values to acquire results in building effective models at a lower cost than randomly acquiring features. Fig. 1 demonstrates this on one such domain, where we see that the models built using Active Feature-value Acquisition (AFA) perform better compared to models build using the same amount of information acquired through uniform random sampling. In addition to reducing the cost of building models, actively selecting the most informative features to learn from can also avoid noise and lead to a better model than training on more data. This phenomenon can also be observed in Figure 1, where the model built using feature values for 100 selected training instances, performs better than using complete features information for 500 instances.

In previous work [10], we have studied the general task of AFA for cost-effective data acquisition for classifier induction. We have also examined the specific case of this problem, where a set of features maybe missing and all missing feature values can be acquired for a selected instance[9]. This Instance-completion setting allows for computationally cheap yet very effective heuristic approaches to feature-acquisition. In these previous studies, we have focused on the acquisition of features during model induction, i.e. at training time. In order to measure the generalization performance of these methods, we use complete test data to evaluate models built on actively acquired training data. However, it is realistic to assume that if features are missing and can be acquired at training time, they may also be acquired at a cost during model application. In this paper, we focus on this complementary problem of Active Feature-value Acquisition at prediction/test time.

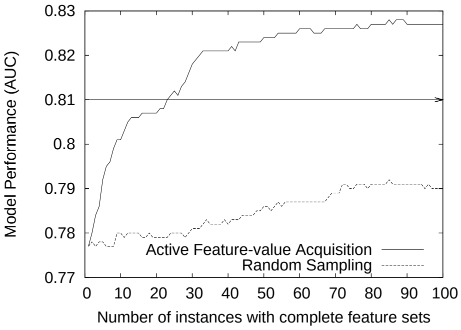

Figure 1: Comparing active acquisition of complete features for instances to random sampling, during induction. The arrow corresponds to the performance of a model using all features for all 500 instances.

We study prediction-time AFA in the context of different customer targeting domains. These domains exhibit a natural dichotomy of features, where one set of features is available for all instances, and the remaining features can be acquired, as a set, for selected instances. As such, these domains lend themselves to AFA in the Instance-completion setting; and in the past they have been used in studies of feature-acquisition during induction. At the time of induction, class labels are available for all instances — including the incomplete instances. This information can be used effectively to estimate the potential value of acquiring more information for the incomplete instances. However, this label information is obviously not present during prediction on test instances, and as such leads us to explore alternative acquisition strategies. In particular, we explore methods to estimate the expected benefit of acquiring additional features for an incomplete instance, versus making a prediction using only incomplete feature information. Extensive experimental results confirm that our approaches can effectively select instances for which it is beneficial to acquire more information to classify them better, as compared to acquiring additional information for the same number of randomly sampled instances.

2 Related work

The problem setting as well as some of the approaches proposed in this paper are influenced by previous work [9, 8, 11], which explores the problem of learning models from incomplete instances by acquiring additional features at induction time. Other interesting induction time settings are budgeted learning and active learning. Under the budgeted learning scenario [7], 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. Traditional *active learning* [2] assumes access to unlabeled instances with complete feature values and attempts to select the most useful instances for which to acquire class labels while training. In contrast to these works, we focus here on Active Feature-value Acquisition at the time of model application or prediction. There has also been some work on prediction-time AFA, but the focus has been on selecting a subset of features to acquire, rather than selecting a subset of instances for which to acquire the features. For example, Bilgic et al. [1] exploit the conditional independence between features in a Bayesian network for selecting a subset of features. Similarly, Sheng et al. [14] aim to reduce acquisition cost and misclassification under different settings, but their approach also focuses on selecting a subset of features. Krause et al. [5] apply the theory of value of information, but their method is mostly restricted to chain graphical models. The general case of acquiring randomly-missing values in the instance-feature matrix is the most interesting, and in previous work [10], we have addressed this problem for induction time. This general setting needs to be explored further for prediction time. The only other example of instance-selection for prediction-time AFA is [3]; but in their case, the test instances are not independent of each other, and the impact of acquisition is studied in the context of graph partitioning. Finally, Kapoor et al. [4] provide a theoretical analysis of budgeted learning when the learner is aware of cost constraints at prediction-time. We differ from this approach, because we focus on the Instance-completion setting, which leads to alternative and computationally efficient solution.

3 Prediction-time active feature-value acquisition for instance-completion

Assume that we are given a classifier induced from a training set consisting of n features and the class labels. We are also given a test set of m instances, where each instance is represented with n feature values. This test set can be represented by the matrix F, where $F_{i,j}$ corresponds to the value of the i^{th} feature of the j^{th} instance. The matrix F may initially be incomplete, i.e., it contains missing values. At prediction time, we 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 prediction-time AFA is the selection of these instance-feature queries that will result in the most accurate prediction over the entire test set at the lowest cost.

As noted earlier, the generalized AFA setting has been studied previously for induction-time. Under the induction-time AFA setting, the training instances have missing features values, which can be acquired at a cost and the goal is to learn the most accurate model with the lowest cost. This model is usually tested on a test-set of complete instances. Here, we are interested in the complementary task of Active Feature-value Acquisition at the time of prediction. The fundamental difference between these two settings is that for induction-time AFA, our goal is to learn a model that would make most accurate predictions on a test set with complete instances, whereas, for prediction-time AFA, the model is trained from a set of complete instances, and the goal is to select queries that will lead to most accurate prediction on incomplete test instances. A third scenario is when the feature values are missing at both induction and prediction time, and the learner is aware of the cost constraints at prediction-time. Hence, the goal of the learner is to learn the most accurate model that optimizes cost at both train and test time. In future, we would like to explore this third scenario.

Here, we consider a special case of the prediction-time AFA problem mentioned above; where feature values for an instance may naturally be available in two sets — one set of features is given for all instances, and the second set can be acquired from an external source at a cost. The task is to select a subset of instances for which the additional features should be acquired to achieve the best cost-benefit ratio.

The two sets of features can be combined in several ways to build a model (or make a prediction at test time). The features from the two sets can be merged before building a model, which is referred to as *early fusion*. Alternatively, two separate models are built using the two sets of features and their outputs are combined in some way to make the final prediction — known as *late fusion*. The alternative strategy we employ in our work is called *Nesting* [8] — in which we incorporate the output of a model using the second set of *additional* features (inner model) as an input to the model using the first set of *given* features (outer model). Specifically, we add another feature in the outer model, corresponding to the predicted probability score for the target variable, as given by the inner model.

The general framework for performing prediction-time AFA for instance-completion setting is described in Algorithm 1. We assume that we are given two models, one induced only from the given features and another one induced from both given and additional features. At prediction time, we are given a set of incomplete instances. We compute a score for each of the incomplete instances based on some acquisition strategy. We sort all instances based on this score and acquire additional features in the sorted order until some stopping criterion is met. The final prediction is made using the appropriate model on the entire set of instances. Note that induction-time AFA has a similar framework, but the main difference is that at induction-time, after each batch of feature acquisition, we need to relearn the model, and hence, recompute the score. On the other hand, at prediction-time, acquiring additional features for one instance has no effect on the prediction of another instance, and as such we can generate the score on the entire set once before starting the acquisition process. This makes large scale, prediction-time AFA feasible on a variety of domains.

4 Acquisition strategies

In this section we describe alternative approaches to selecting instances for which to acquire additional feature values.

Algorithm 1 Prediction-time AFA for Instance-completion using Nesting

```
Given:
$I$ - Set of incomplete instances, which contain only given features
$C$ - Set of complete instances, which contain both given and additional features
$T$ - Set of instances for prediction, $I \cup C$
$M_g$ - Model induced from only given features
$M_c$ - Model induced from both given and additional features
$\forall x_j \in I$, compute the score $S = Score(M_g, x_j)$, based on the AFA strategy
Sort instances in $I$ by score, S.
Repeat until stopping criterion is met
    Let $x_j$ be the instance in $I$ with the next highest score
    Model $M = M_g$ if $x_j \in I$ and $M = M_c$ if $x_j \in C$
return Predictions on $T$ using the appropriate model M
```

4.1 Uncertainty Sampling

The first AFA policy we explore is based on the uncertainty principle that has been extensively applied in the traditional active learning literature [6], as well as previous work on AFA [9]. In Uncertainty Sampling we acquire more information for a test instance if the current model cannot make a confident prediction of its class membership. There are different ways in which one could measure uncertainty. In our study, we use unlabeled margins [9] as our measure; which gives us the same ranking of instances as entropy, in the case of binary classification. The unlabeled margin captures the model's ability to distinguish between instances of different classes. 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 $P_{y1}(x) - P_{y2}(x)$, where $P_{y1}(x)$ and $P_{y2}(x)$ are the first-highest and second-highest predicted probability estimates respectively. Here, a lower score for an instance corresponds to a higher expected benefit of acquiring additional features.

4.2 Expected Utility

Uncertainty Sampling, as described above, is a heuristic approach that prefers acquiring additional information for instances that are currently not possible to classify with certainty. However, it is possible that additional information may still not reduce the uncertainty of the selected instance. The decision theoretic alternative is to measure the expected reduction in uncertainty for all possible outcomes of a potential acquisition. According to an optimal strategy, the next best instance, for which we should acquire features is the one that will result in the greatest reduction in uncertainty per unit cost, in expectation. Since true values of missing features are unknown prior to acquisition, it is necessary to estimate the potential impact of every acquisition for all possible outcomes. Ideally, this requires exhaustively evaluating all possible combinations of values that the additional (missing) features can take for each instance. However, in our Nesting approach to combining feature sets, we reduce the additional features into a single score, which is used as a feature along with the other given features. This allows us to dramatically simplify the complexity of this approach, by only treating this score as a single missing feature, and estimating the utility of possible values it can take. Of course, calculating expectation over this single score does not give us the true utility of the additional features, but it makes the utility computation feasible, especially when we have a very large number of additional features. As such, the expected utility can be computed as:

$$EU(q_j) = \int_x U(S_j = x, C_j) P(S_j = x) \tag{1}$$

Where, $P(S_j = x)$ is the probability that S_j has the value x and $U(S_j = x, C_j)$ is the utility of knowing that S_j has value x. In other words, it is the benefit arising from obtaining a specific value x for score S_j, at cost C_j. In practice, in order to compute the expected utility, we discretize the values of S and replace the integration in Eq. 1 with piece-wise summation. The two terms, U and P in Eq. 1 must be estimated only from available data. We discuss how we empirically estimate these quantities below.

Estimating utility: The utility measure, U, can be defined in one of several different ways. In the absence of class labels, we resort to using measures of uncertainty of the model prediction as a proxy for prediction accuracy. One obvious choice here is to measure the reduction in entropy of the classifier after obtaining value x — similar to what is done in traditional active learning [12], i.e.,

$$U(S_j = x, C_j) = -\frac{H(X \cap S_j = x) - H(X)}{C_j} \tag{2}$$

Where, $H(X \cap S_j = x)$ is the entropy of the classifier on the instance with features X, augmented with $S_j = x$, $H(X)$ is the entropy of the classifier on the instance with features X and C_j is the cost of feature score S_j.

However, using reduction in entropy may not be ideal. We illustrate this through Fig. 2, which compares entropy and unlabeled margins as a function of the predicted class membership probability, $\hat{p}(y|x)$. Note that it does not matter which class y we choose here. We see from the figure, that for the same Δx difference in class membership probability, the corresponding reductions in entropy are different. In particular, the further we are from the decision boundary the higher the change in entropy, i.e. $\Delta y_2 > \Delta y_1$. All else being equal, this measure would prefer acquisitions that would reduce entropy further from the classification boundary; which is less likely to affect the resulting classification. Alternatively, one could use unlabeled margins, which is a linear function of the probability estimate on either side of the decision boundary. This gives the following *expected unlabeled margin* utility measure:

$$U(S_j = x, C_j) = \frac{UM(X \cap S_j = x) - UM(X)}{C_j} \tag{3}$$

Where, $UM(X)$ is the unlabeled margin as described in Sec. 4.1.

Furthermore, one might choose to prefer a difference in $\hat{p}$ closer to the decision boundary; since this is more likely to result in an alternative classification for an instance. We can capture this relationship, by using the log of the unlabeled margins, which gives us the following *expected log margin* measure of utility:

$$U(S_j = x, C_j) = \frac{\ln(UM(X \cap S_j = x)) - \ln(UM(X))}{C_j} \tag{4}$$

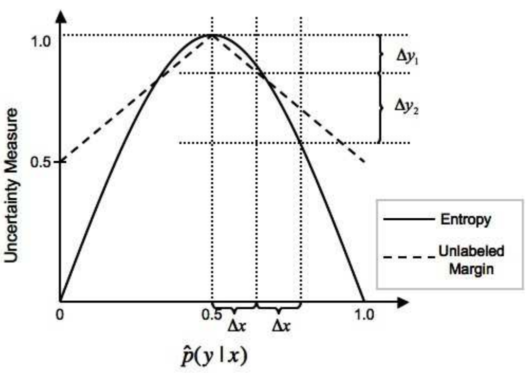

Figure 2: Comparison of unlabeled margin and entropy as measures of uncertainty.

Estimating feature-value distributions: Since the true distribution of the score S_j is unknown, we estimate $P(S_j = x)$ in Eq.1 using a probabilistic learner. We use a model trained only on the additional features and the class to predict the value of S_j and discretize it. We then use S_j as the target variable and all given features as the predictors to learn a classifier M. When evaluating the query q_j, the classifier M is applied to incomplete instance X_j to produce the estimate $\hat{P}(S_j = x)$.

5 Empirical evaluation

We tested our proposed feature-acquisition approaches on four datasets from customer targeting applications, as described below.

5.1 Data description

Our first dataset, *Rational*, comes from a system developed at IBM to help identify potential customers and business partners. The system formerly used only structured firmographic data to predict the propensity of a company to buy a product. Recently, it has been shown that incorporating information from company websites can significantly improve these targeting models. However, in practice, processing websites for millions of companies is not desirable due to the processing costs and noisy web data. Hence we would like to select only a subset of companies for which to acquire web-content, to add to the firmographic data, to aid in prediction. This is a case of the instance-completion setting, where firmographic features are available for all instances, and the web features are missing and can be acquired at a cost. Instance-completion heuristics have been applied to this data during induction [8]; and, here, we study the complementary task of prediction-time AFA. The remaining three web-usage datasets, *qvc*, *bmg* and *expedia* come from a study by Zheng and Padmanabhan [15]. These datasets contain information about web users and their visits to retail web sites. The given features describe a visitor's surfing behaviors at a particular site, and the additional features, which can be purchased at a cost from an external vendor, provides aggregated information about the same visitor's surfing behavior on other e-commerce sites. The target variable indicates whether or not the user made a purchase during a given session. This setting also fits naturally in the Instance-completion setting of AFA.

5.2 Comparison of acquisition strategies

For all datasets, we use Nesting to combining the two separate feature sets. We experimented with different combinations of base classifiers in Nesting, and found that using decision trees for the additional features and logistic regression for the composite model is most effective for the web-usage datasets. For *Rational*, we use multinomial naive Bayes for the web features, and logistic regression for the composite model. Since there is a small proportion of instances from the target class in *Rational*, and it is a ranking problem, we use AUC instead of accuracy as a performance metric (as done in [8]). For all other datasets, we use accuracy as done in their previous usage [15].

We ran experiments to compare Random Sampling and the AFA strategies described in Sec. 4. The performance of each method was averaged over 10 runs of 10-fold cross-validation. In each fold, we generated acquisition curves as follows. After acquiring additional features for each actively-selected test instance, we measure accuracy (or AUC, in case of *Rational*) on the entire test set using the appropriate model (see Algorithm 1). In the case of Random Sampling, instances are selected uniformly at random from the pool of incomplete instances. For, the expected utility approaches described in Sec. 4.2, we used 10 equal-width bins for the discretization of the score S_j in Eq. 1.

Fig. 3 shows the effectiveness of each strategy in ordering the instances so as to get the most benefit with the least cost of data acquisition. We assume, for these experiments, that there is a unit cost of acquiring additional features for each instance. In all cases, active acquisition clearly out-performs Random Sampling, resulting in improved prediction performance for the same amount of feature information acquired for the test instances. Also, a large amount of improvement in accuracy is achieved by acquiring complete feature sets for only a small fraction of instances. Which suggests that it is not critical to have complete feature information for all instances to correctly classify them.

Even with the gross approximations and estimations done in Sec. 4.2, the *Expected Utility* approach still manages to perform quite well compared to random sampling. Furthermore, using reduction in log margins tends to slightly outperform the alternative utility measures, for the reasons discussed in Sec. 4.2. However, in general, the *Expected Utility* methods still do not exceed the performance of *Uncertainty Sampling*, as one would expect. It is possible that the estimations done in the computation of *Expected Utility* are too crude and need to be improved. One source of improvement could be through better estimation of the probability distribution of missing feature values. Currently this is being reduced to estimating the probability of a single discretized score, representing the output of a model built using the additional features. In order to evaluate the room for improvement in

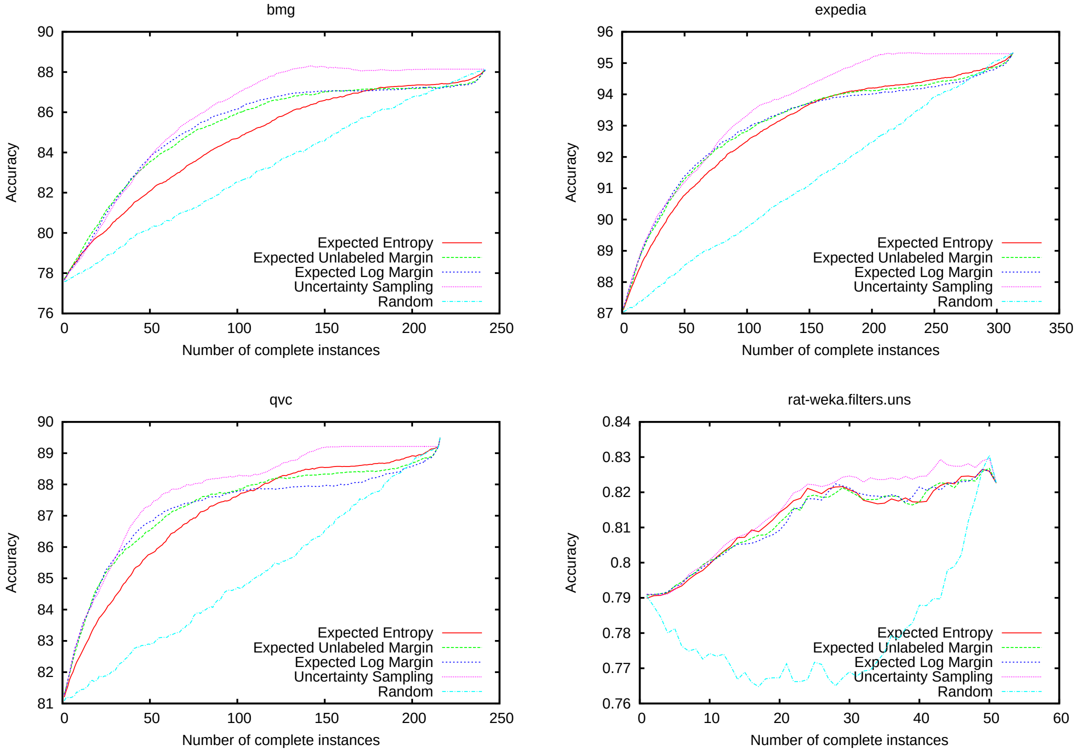

Figure 3: Comparison of acquisition strategies

this estimation, we use the true value of the discretized score while calculating the expectation in Eq. 1. This *Expected Log Margins* with Oracle approach is shown in Fig. 4, in comparison to the estimated *Expected Log Margins* approach. We see that, indeed, if we had the true probability estimate $P(S_j = x)$, we can perform much better than using the estimation approach described in Sec. 4.2. However, this by itself is still insufficient to outperform *Uncertainty Sampling*. We may be losing too much information by compressing the additional feature set into a single score. Using alternative feature-reduction techniques may lead to a more meaningful estimation of the missing value distribution, without too much increase in computational complexity brought about by having to estimate the joint distribution of features. Perhaps a better estimate of utility U is also required to make the *Expected Utility* approach more effective. We are exploring these avenues of improvement in future work.

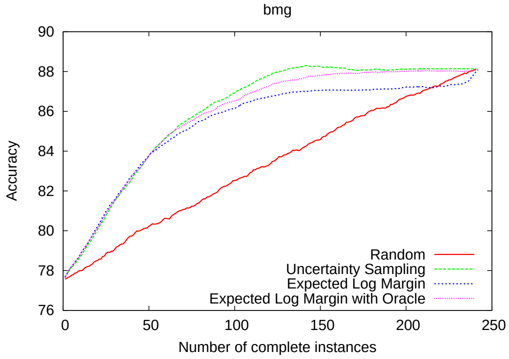

Figure 4: Comparison of acquisition strategies using an Oracle

6 Conclusion

In many domains we can acquire additional information for selected test instances that may help improve our classification performance on the test set. When this additional information comes at a cost, or is potentially noisy, it is best to actively select the instances that are most likely to benefit from acquiring additional information. We study this problem of prediction-time Active Feature-value Acquisition in the context of several customer targeting data sets that naturally lend themselves to this setting. We demonstrate that our approaches of measuring the uncertainty of predictions, and the expected reduction of uncertainty through additional feature-acquisition, are much more effective than the baseline approach of uniformly sampling instances for acquiring more information. Empirical results show that estimating the expected reduction in uncertainty of a prediction is an effective acquisition strategy. However, it is not as effective as just selecting instances based on the uncertainty of their prediction using incomplete information. We suggest methods for improving the proposed *Expected Utility* approach as directions for future work.

Acknowledgments

We would like to thank Richard Lawrence, Claudia Perlich, Maytal Saar-Tsechansky and Andrew McCallum for useful discussions on this study.

References

[1] M. Bilgic and L. Getoor. Voila: Efficient feature-value acquisition for classification. In *AAAI*, pages 1225–1230. AAAI Press, 2007.

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

[3] P. Kanani, A. McCallum, and C. Pal. Improving author coreference by resource-bounded information gathering from the web. In *Proceedings of IJCAI*, 2007.

[4] A. Kapoor and R. Greiner. Learning and classifying under hard budgets. In *ECML*, pages 170–181, 2005.

[5] A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Twenty-first Conference on Uncertainty in Artificial Intelligence (UAI*, page 05, 2005.

[6] D. D. Lewis and J. Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proceedings of the Eleventh International Conference on Machine Learning (ICML-94)*, pages 148–156, San Francisco, CA, July 1994. Morgan Kaufmann.

[7] D. Lizotte, O. Madani, and R. Greiner. Budgeted learning of naive-Bayes classifiers. In *Proceedings of 19th Conference on Uncertainty in Artificial Intelligence (UAI-2003)*, Acapulco, Mexico, 2003.

[8] P. Melville, S. Rosset, and R. D. Lawrence. Customer targeting models using actively-selected web content. In Y. Li, B. Liu, and S. Sarawagi, editors, *KDD*, pages 946–953. ACM, 2008.

[9] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. Active feature-value acquisition for classifier induction. In *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM-04)*, pages 483–486, 2004.

[10] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. An expected utility approach to active feature-value acquisition. In *Proceedings of the International Conference on Data Mining*, pages 745–748, Houston, TX, November 2005.

[11] B. Padmanabhan, Z. Zheng, and S. O. Kimbrough. Personalization from incomplete data: what you don't know can hurt. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, pages 154–163, 2001.

[12] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of 18th International Conference on Machine Learning (ICML-2001)*, pages 441–448. Morgan Kaufmann, San Francisco, CA, 2001.

[13] M. Saar-Tsechansky, P. Melville, and F. Provost. Active information acquisition for model induction. In *Management Science*, 2008.

[14] V. S. Sheng and C. X. Ling. Feature value acquisition in testing: a sequential batch test algorithm. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 809–816, New York, NY, USA, 2006. ACM.

[15] Z. Zheng and B. Padmanabhan. On active learning for data acquisition. In *Proceedings of IEEE International Conference on Data Mining*, 2002.