ACTIVE LEARNING OF HYPERSPECTRAL DATA WITH SPATIALLY DEPENDENT LABEL ACQUISITION COSTS

Alexander Liu   *Goo Jun*   *Joydeep Ghosh*

Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin TX 78712, USA
{ayliu, gjun}@mail.utexas.edu, {ghosh}@ece.utexas.edu

ABSTRACT

Supervised learners can be used to automatically classify many types of spatially distributed data. For example, land cover classification by hyperspectral image data analysis is an important remote sensing task where a supervised learner is trained on a large set of labeled data. However, while gathering unlabeled samples may be relatively easy, labeling large amounts of data can be very costly. Acting learning is one approach to reduce the amount of labeled data required to build a supervised learner that performs well. However, most active learning approaches assume that the cost of acquiring labels for all points is uniform. For spatially distributed data that requires physical access to spatial locations in order to assign labels, label acquisition costs become proportional to distance traveled in order to label a point. In this paper, we present results for applying a novel active learning method which takes variable label acquisition costs into account on two hyperspectral datasets.

Index Terms— hyperspectral data, remote sensing, classification, active learning, spatial information

1. INTRODUCTION

Supervised machine learning techniques can be used to classify various forms of spatially distributed data. However, supervised techniques rely on large amounts of labeled data in order to build accurate models. While the data itself may be comparatively easy to gather, labeling this data is often difficult and costly. For example, in [1], the authors state that labeling a single training sample is approximately 500 U.S. dollars for forestry applications. One machine learning technique to reduce the amount of labeled data required to build a supervised model is active learning. In active learning, one attempts to reduce the number of labeled training points required for a certain level of classifier performance by allowing the active learning algorithm itself to choose which points should be labeled.

Most active learners assume that: 1) the cost of acquiring the label for a particular point is independent of the costs for all other points and that 2) label acquisition costs are equal. When labeling spatially distributed data, both of these assumptions may be false. For example, for classification of land-cover using hyperspectral data, acquiring labels may involve traveling to a particular location and performing some sort of test such as determining land type or collecting various samples (e.g., soil, water, foliage) that requires physical access. Traveling to this point incurs some type of cost (e.g., gas or time) proportional to distance traveled. The distance traveled also depends on the order in which one labels the points.

In this paper, we present a novel framework for performing active learning while taking into account spatially sensitive labeling costs. In particular, we pose the problem as a traveling salesman problem with profits. We present example results using hyperspectral data, but the presented approach is also applicable to other spatially distributed data where supervised learning is used. The algorithms presented in this paper were previously introduced in [2] and [3]. In this paper, we focus mainly on experimental results.

Related works: Although many active learning strategies have been proposed during the last 15 years (see [4] for a recent survey), there exist few algorithms that consider spatial characteristics of unlabeled samples. In [5], the authors proposed an active learning algorithm for hyperspectral data that adapts a classifier for spatial variation of spectral signatures. However, it does not take into account any form of varying label acquisition costs based on spatial data. An active learning algorithm to efficiently model spatial phenomena with Gaussian processes has been proposed [6], but the algorithm is used to model spatially varying quantities and is not applicable to classification problems. Aside from our own work, we are unaware of any active learning studies which take spatially dependent label acquisition costs into account. However, both [7] and [8] present methods for incorporating non-spatial label acquisition costs with active learning for natural language processing tasks; one can show that the method we present below called “US/TSPP” is mathematically related.

This work was supported by NSF Grant IIS-0705815.

2. ACTIVE LEARNING WITH SPATIAL COSTS

Problem setting: In this paper, we adapt a pool-based active learning technique called uncertainty sampling [9] for handling spatially related label acquisition costs. As in "standard" active learning, active learning on spatial data occurs in an iterative fashion where, on each iteration i, points from some unlabeled set $\mathcal{U}$ are selected by the active learner based on some criteria, labeled by some oracle, and then placed in the labeled set $\mathcal{L}$. The labeler starts and ends each iteration at some "home location" [1]. On each iteration, the labeler labels points in $\mathcal{U}$ until some traveling budget is expended, where the traveling budget is the amount of time available for traveling and labeling per iteration.

We will use the following notation. On the ith iteration, the algorithm selects n_i points for labeling where n_i depends on some traveling budget which we will denote as t_{max}. The actual cost of traveling and labeling points for the ith iteration will be denoted as t_i. t_i depends on the total distance d_i traveled on the ith iteration, the speed s of the labeler's vehicle, the cost of labeling a single point c_l, and the number of points labeled n_i. In particular, $t_i = (d_i/s) + (c_l * n_i)$ and the constraint is that $t_i < t_{max}$. We will measure t_i, t_{max}, and c_l in units of time, d_i in units of length, and s in units of length/time. Finally, we will denote the uncertainty score (as determined by uncertainty sampling) for the jth point in $\mathcal{U}$ as $u(j)$.

Solutions: A simple baseline is to start at home and continue labeling the next closest unlabeled point while $t_i < t_{max}$. We will call this baseline the "closest next" baseline. A second baseline is to pick points via the non-spatial, "traditional" machine learning methods of random sampling and uncertainty sampling. Then, using a solution to the traveling salesman problem, the shortest path through the chosen points is followed. We will refer to this baseline algorithm as "random/TSP" if random sampling is used or "US/TSP" if uncertainty sampling is used to select points.

However, the above techniques are somewhat naive, as they look at either only spatial locations or only the benefit to the classifier. A more sophisticated approach is to pose the problem as a traveling salesman problem with profits (TSPP) [10], allowing for both spatial information and benefit to the classifier to be examined simultaneously. Our first proposed method is to pose the problem as a TSPP problem where the profit for visiting the jth point is the uncertainty score $u(j)$ of that point, and the constraint is that the salesman can visit a variable number of cities per iteration as long as the total time required to travel along all cities and reach home is less than the traveling budget t_{max} for that iteration. We refer to this approach as "US/TSPP". Finally, we found a variant of US/TSPP to be empirically useful: instead of supplying all possible unlabeled points in $\mathcal{U}$ to the TSPP algorithm, only the top m points with the highest uncertainty scores (where $m \geq n_i$) are used. We refer to this approach as "US/TSPP (filtered)", and set $m = 100$ in experiments.

3. EXPERIMENTS

In this section, we present example results on the Kennedy Space Center (KSC) and Botswana hyperspectral datasets. The data is preprocessed using both the max-cut algorithm [11] and best-basis feature reduction [12], both of which are useful for classifying hyperspectral data. We run experiments using an LDA classifier and average results over five runs of ten-fold cross validation.

Experimentally, we tried a variety of values for s and c_l [2], but interestingly, specific values do not seem to affect general trends very much. Here, we present results where $s = 80$ kilometers per hour, $c_l = 10$ minutes, and $t_{max} = 8$ hours. Example results for these values are plotted in Figure 1. The results should be interpreted by looking at three aspects of each curve: the total amount of effort required to label all of U [3], how quickly the method reduces error rate, and the lowest error rate a method achieves. For both datasets, the initial reduction in error is similar for all but the random/TSP methods, but both US/TSPP and the "closest next" baseline tend to outperform the other techniques. Not surprisingly, the "closest next" baseline requires the least effort to label all points in $\mathcal{U}$, but US/TSPP is very competitive, and is followed by US/TSPP(filtered), US/TSP, and then random/TSP. However, in terms of the minimum error rate achieved, the "closest next" baseline does very poorly, and the US/TSPP method is preferable. In addition, US/TSPP(filtered) and US/TSP tend to achieve the lowest error rates. Thus, US/TSPP(filtered) appears to be the best tradeoff in terms of reducing error rate quickly and achieving a low error rate.

One interesting aspect of the results, particularly on the KSC dataset, is that the error rate can be much lower when only a subset of $\mathcal{U}$ has been labeled as opposed to when all of $\mathcal{U}$ has been labeled. This phenomenon has been observed in other works on active learning as well (e.g., [13]). Stopping the labeling process early can therefore be very useful in reducing overall classification error. One possible reason is that stopping the process early helps to avoid outliers in the training set. However, our current conjecture is that the phenomena in figure 1 is due to more than just the filtering of outliers, and we are currently investigating viable hypotheses explaining these results.

Finally, let us look at results on individual classes. To do this, we will use two common evaluation metrics, precision and recall. Let n^i_{tp} be the number of points in the test set from the ith class correctly identified as being from the ith class, let n^i_{fp} be the number of points in the test set incorrectly assigned

[1] This home location may correspond to where the labeler's vehicle is stored/refueled or the labeler's base of operations.

[2] experimental values for s ranged from 15 to 80 kilometers per hour, while values for c_l ranged from 10 to 50 minutes

[3] these points are plotted with an X in the graphs

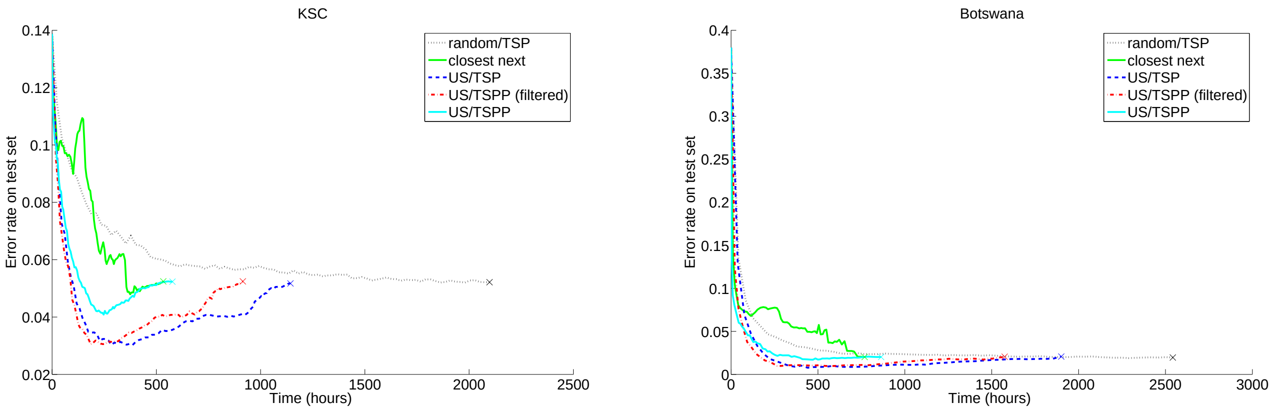

Fig. 1. Example results.

to class i, and let n^i_{fn} be the number of points in the test set from class i incorrectly classified as some class other than i. Then, for the ith class, precision $= n^i_{tp}/(n^i_{tp}+n^i_{fp})$ and recall $= n^i_{tp}/(n^i_{tp}+n^i_{fn})$. Note that both precision and recall range from 0 to 1, with 1 being the highest possible value for either metric.

In table 1, we look at the precision and recall of each class with respect to label acquisition costs for US/TSPP(filt). From the table, one can observe that the LDA classifier combined with active learning is capable of producing good precision and recall values fairly quickly (i.e., with not much label acqusition cost) on all classes.

4. REFERENCES

[1] R.R. Vatsavai, S. Shekhar, and B. Bhaduri, "A semi-supervised learning algorithm for recognizing subclasses," Dec. 2008, pp. 458–467.

[2] Alexander Liu, Goo Jun, and Joydeep Ghosh, "Active learning with spatially sensitive labeling costs," *NIPS Workshop on Cost-sensitive Learning*, 2008.

[3] Alexander Liu, Goo Jun, and Joydeep Ghosh, "Spatially cost-sensitive active learning," *SIAM International Conference on Data Mining*, 2008.

[4] Burr Settles, "Active learning literature survey," Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[5] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. on Geosci. and Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, 2008.

[6] Andreas Krause and Carlos Guestrin, "Nonmyopic active learning of gaussian processes: an exploration-exploitation approach," in *ICML '07: Proceedings of the 24th international conference on Machine learning*, New York, NY, USA, 2007, pp. 449–456, ACM.

[7] Robbie Haertel, Kevin D. Seppi, Eric K. Ringger, and James L. Carroll, "Return on investment for active learning," *NIPS Workshop on Cost-sensitive Learning*, 2008.

[8] Burr Settles, Mark Craven, and Lewis Friedland, "Active learning with real annotation costs," *NIPS Workshop on Cost-sensitive Learning*, 2008.

[9] David D. Lewis and Jason Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *In Proceedings of the Eleventh International Conference on Machine Learning*. 1994, pp. 148–156, Morgan Kaufmann.

[10] Dominique Feillet, Pierre Dejax, and Michel Gendreau, "Traveling salesman problems with profits.," *Transportation Science*, vol. 39, pp. 188–205, 2005.

[11] Yangchi Chen, M.M. Crawford, and J. Ghosh, "Knowledge based stacking of hyperspectral data for land cover classification," *Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on*, pp. 316–322, 2007.

[12] S. Kumar, J. Ghosh, and M. M. Crawford, "Best-bases feature extraction algorithms for classification of hyperspectral data," *IEEE Trans. on Geosci. and Remote Sens.*, vol. 39, no. 7, pp. 1368–1379, 2001.

[13] Greg Schohn and David Cohn, "Less is more: Active learning with support vector machines," in *Proc. 17th International Conf. on Machine Learning*. 2000, pp. 839–846, Morgan Kaufmann, San Francisco, CA.

Table 1. Averaged values for precision/recall on each class for US/TSPP(filt)

dataset	class	precision(25 hrs)	precision(50 hrs)	precision(100 hrs)	max precision
ksc	scrub	0.952	0.964	0.968	0.993
ksc	willow swamp	0.922	0.951	0.965	0.996
ksc	cabbage palm hammock	0.769	0.857	0.911	0.950
ksc	cabbage palm hammock/oak hammock	0.712	0.806	0.887	0.957
ksc	slash pine	0.834	0.861	0.885	0.985
ksc	oak/broadleaf hammock	0.757	0.804	0.870	0.952
ksc	hardwood swamp	0.764	0.813	0.865	0.942
ksc	graminoid marsh	0.900	0.930	0.951	0.979
ksc	spartina marsh	0.938	0.946	0.957	0.992
ksc	cattail marsh	1.000	1.000	1.000	1.000
ksc	salt marsh	1.000	1.000	1.000	1.000
ksc	mud flats	0.934	0.937	0.951	1.000
ksc	water	1.000	1.000	1.000	1.000
bots	water	0.995	0.997	1.000	1.000
bots	hippo grass	0.976	0.990	0.994	0.995
bots	floodplain grasses 1	0.940	0.977	0.996	1.000
bots	floodplain grasses 2	0.928	0.954	0.970	0.999
bots	reeds1	0.887	0.921	0.950	0.996
bots	riparian	0.732	0.831	0.928	0.988
bots	firescar2	0.994	0.997	1.000	1.000
bots	island interior	0.955	0.976	0.989	1.000
bots	acacia woodlands	0.806	0.888	0.946	1.000
bots	acacia shrublands	0.787	0.855	0.916	0.996
bots	acacia grasslands	0.916	0.951	0.975	0.998
bots	short mopane	0.874	0.932	0.975	1.000
bots	mixed mopane	0.814	0.877	0.913	1.000
bots	exposed soils	0.998	1.000	1.000	1.000
dataset	class	recall(25 hrs)	recall(50 hrs)	recall(100 hrs)	max recall
ksc	scrub	0.928	0.946	0.967	0.987
ksc	willow swamp	0.879	0.898	0.928	0.972
ksc	cabbage palm hammock	0.832	0.883	0.921	0.968
ksc	cabbage palm/oak hammock	0.689	0.781	0.840	0.911
ksc	slash pine	0.736	0.821	0.876	0.937
ksc	oak/broadleaf hammock	0.812	0.874	0.915	0.975
ksc	hardwood swamp	0.823	0.916	0.920	0.989
ksc	graminoid marsh	0.933	0.937	0.945	0.987
ksc	spartina marsh	0.938	0.962	0.978	0.995
ksc	cattail marsh	0.982	0.981	0.983	0.999
ksc	salt marsh	0.925	0.931	0.948	1.000
ksc	mud flats	0.958	0.962	0.984	1.000
ksc	water	1.000	1.000	1.000	1.000
bots	water	0.998	0.999	1.000	1.000
bots	hippo grass	0.978	0.989	0.995	1.000
bots	floodplain grasses 1	0.949	0.969	0.987	1.000
bots	floodplain grasses 2	0.969	0.990	0.995	1.000
bots	reeds1	0.840	0.900	0.939	0.991
bots	riparian	0.729	0.819	0.900	0.994
bots	firescar2	0.985	0.994	0.997	1.000
bots	island interior	0.941	0.973	0.998	1.000
bots	acacia woodlands	0.829	0.908	0.965	0.995
bots	acacia shrublands	0.879	0.947	0.977	0.998
bots	acacia grasslands	0.855	0.896	0.939	0.997
bots	short mopane	0.783	0.823	0.865	1.000
bots	mixed mopane	0.818	0.900	0.965	1.000
bots	exposed soils	0.884	0.902	0.930	1.000