Active Learning of Classification Models from Soft-Labeled Groups

Zhipeng Luo and Milos Hauskrecht
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
{zpluo, milos}@cs.pitt.edu

Abstract

Learning of classification models in practice often relies on nontrivial human annotation effort in which humans assign class labels to data instances. As this process can be very time consuming and costly, finding effective ways to reduce the annotation cost becomes critical for building such models. To solve this problem we develop a new approach that actively learns instance-based classification model from subpopulations (groups) of instances and their soft labels. A soft label represents a human estimate of the proportion of instances with one of the class labels in the subpopulation, or equivalently, the probability with which an instance with that class label is drawn from the subpopulation. To form the groups to be annotated, we develop a hierarchical active learning framework that divides the whole population into smaller subpopulations, which allows us to gradually learn a more refined model from the subpopulations and their soft labels. Our comprehensive experiments show that our method is competitive and outperforms existing approaches on reducing the human annotation cost.

1 Introduction

Learning of classification models from real-world data often requires heavy supervision on labeling data instances. The problem is that *instance-based* supervision is often time-consuming, and thus it may be unrealistic to assume that an arbitrarily large number of instances can be feasibly labeled. The key challenge then is to find ways to build accurate models from limited human supervision.

One popular approach to reduce the labeling effort is active learning which sequentially selects a subset examples to be labeled. However, instance-based active learning may still be insufficient, as the assumption that instances are easy for humans to label may not hold. For example when data instances are high-dimensional or very complex objects, e.g. Electronic Health Records, each instance labeling requires annotators with additional expertise and time.

To save annotation cost, another promising direction is to move from instance-based labeling and learning to *group-based* supervised learning. That is, instead of providing instance-based feedback a human gives labels on subpopulations or groups of instances. The essential advantage is that people can provide weak supervision on *multiple* instances at a time. In this work, we combine active learning and group learning to propose a new framework that seeks to build an instance-based binary classification model from human feedback on groups of instances. Our approach assumes the human feedback is provided in terms of a *soft range* label (e.g. *"70% to 90%" Positive*) that represents a weak estimate of the proportion of classes in the subpopulation (group), or equivalently, the probability with which examples that belong to one of the classes can be drawn from the subpopulation.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Our *Learning from Soft-Labeled Groups* framework, summarized in Algorithm 1, can actively learn a binary probabilistic model from soft-labeled groups. As initially there are no apparent groups in the data, our algorithm starts with a hierarchical clustering on all the unlabeled data (line 1), followed by a proper adjustment (line 2) to generate groups. Then we always maintain a fringe of groups which is a complete partition of the all the data (line 3), and perform standard active learning cycles (Line 4-9). We use maximum expected model change strategy to select the most promising group along the fringe to split, assess its child groups by human and replace the group with its children in the fringe.

Algorithm 1: The LSLG (Learning from Soft-Labeled Groups) Framework

Input: An unlabeled data pool $\mathcal{U}$; A labeling budget; Human annotator(s)
Output: An instance-based binary probabilistic model $P(y|\boldsymbol{x};\boldsymbol{\theta})$

1: $T \leftarrow$ Perform hierarchical clustering on $\mathcal{U}$
2: $T_G \leftarrow$ Adjust T to form a new tree of groups and learn description for each group (Section 2)
3: Iteration $t \leftarrow 0$; Current fringe $F^{(0)} \leftarrow \{(T_G)'s\ root\}$; $\boldsymbol{\theta}^{(0)} \leftarrow$ random initialization
4: **repeat**
5: Actively select a group G_* in $F^{(t)}$ based on $P(y|\boldsymbol{x};\boldsymbol{\theta}^{(t)})$ (Section 3)
6: Obtain the labels of G_*'s children from annotator(s)
7: $F^{(t+\Delta t)} \leftarrow \{F^{(t)} - G_*\} \cup \{G_*\text{'s children}\}$; $t \leftarrow t + \Delta t$; ($\Delta t =$ # of G_*'s children)
8: Retrain the model $P(y|\boldsymbol{x};\boldsymbol{\theta}^{(t)})$ based on current labeled groups $F^{(t)}$ (Section 4)
9: **until** the labeling budget runs out
10: **return** $P(y|\boldsymbol{x};\boldsymbol{\theta}^{(t)})$

An illustration example Suppose we want to learn a binary classification model for predicting hospital admissions (the target label y) for all the patients encountered in the Emergency Room (ER) based on the a set of measurements such as heart rate, blood pressure and temperature (the predictors $\boldsymbol{x}$). Initially, an ER clinician may estimate the chance of admission to be 20% $\sim$ 30% for the entire ER population, but this assessment may go up significantly to say 60% $\sim$ 70% for the subpopulation with a high heart rate (120-140), and may further rise to 70% $\sim$ 90% for a subpopulation of patients with both a high heart rate (140-160) and a low blood pressure (Diastolic$<$60, Systolic$<$100), which are the signs of a significant blood loss. Our approach aims to take advantage of such subpopulations and their soft assessments to obtain an instance-based classifier for each patient.

2 The Concept of Groups

Given a pool of unlabeled data instances, represented as a real number matrix $\mathcal{U}_{n\times m}$ consisting of n m-dimensional instances, we first perform a standard hierarchical clustering (using ward linkage [13]) on $\mathcal{U}$ to output a tree T.

The Group Description The initial clusters (the potential groups) consist of sets of instances. To describe each group efficiently to humans, we use conjunctive patterns over the input space features, which matches precisely the hypercube definition of a typical decision tree algorithm. To convert the groups to hypercube representation we employ a C4.5 [8] classifier to automatically learn the group descriptions [9]. More specifically, if we want to learn the description of a group G_i indexed at i, we mark all instances in G_i as **1** and the rest of data ($\mathcal{U} - G_i$) as **0**. Then a C4.5 classifier will output a set of hypercubes $\mathcal{C}(G_i)$ that could potentially describe G_i. The fitness of each hypercube $c \in \mathcal{C}(G_i)$ can be assessed by $F1score = \frac{2\times precision\times recall}{precision+recall}$. That is, the description of G_i is a hypercube which is $\arg\max_{c\in\mathcal{C}(G_i)} F1score(c)$.

The Group Formation When C4.5 is directly performed on the clusters of the original hierarchical tree T, there may exist some clusters that are not hypercube-shaped. As a consequence, their best description hypercubes still have intolerably low $F1score$s. To mitigate this issue, we need to adjust the original tree structure to form a new tree T_G such that only hypercube-like clusters are conserved in T_G. Formally, we define that a cluster is *hypercube-like* if its description hypercube satisfies a minimum $precision(\geq 0.5)$ and $recall(\geq 0.5)$. Then we prune the original hierarchical tree T by preserving only hypercube-like clusters. As a result, the binary tree T may become a multi-nary tree T_G in which the nodes are all hypercube-like, and we will use T_G as a tree of unlabeled groups for succeeding learning process.

Point-Based Group Soft Label The human assessment of each group is made via a *soft* label, which is an estimate of the proportion of one of the classes in the group. For example, people could say that *90%* of instances in a certain group are *positive*. As the classifier model we want to build is instance-based, it is important to link the individual data instances to group labels. Suppose a group G_i consists of n_i instances $\{(\boldsymbol{x}_{ij}, y_{ij})\}_{j=1}^{n_i}$ and we can express under a probabilistic model that each class variable y_{ij} follows a Bernoulli distribution with parameter μ_{ij}. Then we can estimate the mean parameter of the group from instances observed in the group as $\tilde{\mu}_i = \frac{1}{n_i}\sum_j \mu_{ij}$. This averaged $\tilde{\mu}_i$ for a large sample from the group should converge to the true proportion of classes μ_i in the group. This true proportion is estimated by a human and it is the feedback that we seek to label the group.

Range-Based Group Soft Label One caveat of using the exact point estimate of the class proportions μ_i in the group G_i is that the estimate is often hard for a human to make and it can be subject to various bias and noise. In our work, we relax this requirement by considering a *range-based* soft-label feedback that allows weaker supervision on groups, such as *60% to 80% positive*. Suppose the group G_i is given a range label as $[lb_i, ub_i]$ ($0 \leq lb_i \leq ub_i \leq 1$), how can we interpret it and build it into our learning process? As humans often tend to give a symmetric interval estimation and place most of his/her confidence in the interval, our solution is to assume that the range label defines a symmetric interval and that the span of this interval reflects uncertainty of the annotator in the proportion estimate. To incorporate this idea into the model, we rely on a hierarchical Bayesian approach and assume μ_i is a random variable that follows Beta distribution $Beta(a_i, b_i)$, where a_i, b_i are the two shape parameters. We treat $[lb_i, ub_i]$ as a confidence interval to determine the two hyper-parameters as follows. First, the midpoint of the upper and lower bound of the interval is approximately equal to the mean of the Beta distribution. Then it leads to: $\mathbb{E}(\mu_i) = a_i/(a_i + b_i) = (lb_i + ub_i)/2$. Second, we assume that $1 - \alpha$ of the density of μ_i should fall into $[lb_i, ub_i]$, that is, $I_{ub_i}(a_i, b_i) - I_{lb_i}(a_i, b_i) = 1 - \alpha$, where $I_x(a_i, b_i)$ is the cumulative distribution function (or the regularized incomplete beta function). In this way, the range label of each group G_i can be interpreted as a $Beta(a_i, b_i)$ of μ_i.

3 Active Refinement of Groups

Given the group hierarchy T_G and the initial fringe $F^{(0)} = \{(T_G)'s\ root\}$ at $t = 0$, in each active learning cycle, we adopt *maximum expected model change* criterion [12, 3] to select a group G_* in $F^{(t)}$ to split, query its children and then replace G_* with its children in the fringe, renewed as $F^{(t+\Delta t)}$. The key idea is to select the group G_* which would lead to the greatest change to current model if we *were* to split it. To achieve this goal, we need to: (1) estimate the label distribution of each group's children; and (2) calculate the expected model change properly.

For each group G_i in the current fringe $F^{(t)}$, we model each of its child's soft-label distribution again using a Beta distribution, aka a posterior Beta. Please recall the classic Binomial-Beta model for Bayesian estimation, that the empirical counts from a Binomial likelihood plus a Beta prior yields a posterior Beta. This model precisely applies to our situation where the empirical counts of positives/negatives in a child group can be predicted by current model $\boldsymbol{\theta}^{(t)}$, and the prior Beta comes directly from its parent. One difference is we use *expected* rather than *hard* counts to preserve more precision. Formally, suppose $Beta(a_i, b_i)$ is G_i's label distribution (known), serving as a prior; for each child G_{ij}, the number of expected positive instances is $\hat{n}_{ij}^{+} = \sum_j \mathbb{E}(\mathbf{1}_{\hat{y}_{ij}=1}) = \sum_j \hat{\mu}_{ij}$, where $\hat{\mu}_{ij} = P(y_{ij}|\boldsymbol{x}_{ij}; \boldsymbol{\theta}^{(t)})$. $\hat{n}_{ij}^{-}$ can be calculated similarly. Based on the two statistics, the label distribution of G_{ij} follows a posterior $Beta(a_{ij}, b_{ij})$ where $a_{ij} = a_i + \hat{n}_{ij}^{+}$ and $b_{ij} = b_i + \hat{n}_{ij}^{-}$. After all child groups $\{G_{ij}\}$'s label distribution are estimated, we retrain the model (to be $\boldsymbol{\theta}_{[i]}^{(t)}$) based on $F_{[i]}^{(t)} = (F^{(t)} - G_i) \cup \{G_{ij}\}$, and compare it to $\boldsymbol{\theta}^{(t)}$. So $\boldsymbol{\theta}_{[i]}^{(t)}$ represents the new model parameter that we have *guessed* should be, had the group G_i be split.

To calculate the model change $MC(\boldsymbol{\theta}, \boldsymbol{\theta}')$, similar to [3, 10], we use all instances in $\mathcal{U}$ as a validation set, and calculate the soft label changes for all instances. Formally, $MC(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{\boldsymbol{x}\in\mathcal{U}} |\mu(\boldsymbol{x}; \boldsymbol{\theta}) - \mu(\boldsymbol{x}; \boldsymbol{\theta}')|$, where $\mu(\boldsymbol{x}; \boldsymbol{\theta}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta})$.

Finally, we select $G_* = \arg\max_{G_i \in F^{(t)}} MC(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_{[i]}^{(t)})$ to split. After that, G_*'s children $\{G_{*j}\}$ are sent for querying and a new fringe is updated as $F^{(t+\Delta t)} = (F^{(t)} - G_*) \cup \{G_{*j}\}$ for learning a new model. As each group label consumes one query, Δt is equal to the number of G_*'s children.

4 Learning a Model from Labeled Groups

The Loss Function Suppose at time t, there are N labeled groups $\{G_1, ..., G_N\}$ in fringe $F^{(t)}$ and let $\{\boldsymbol{x}_{i1}, ..., \boldsymbol{x}_{in_i}\}$ be a set of n_i instances covered by a group G_i. Our goal is to learn an instance probabilistic model $P(y|\boldsymbol{x}; \boldsymbol{\theta})$. Here $\boldsymbol{x} \in \mathbb{R}^m$ and $y \in \{0, 1\}$.

Let us first consider the case where each group G_i is given an *exact* label $\mu_i \in [0, 1]$ by an annotator. Inspired by previous work [7, 5, 6], this estimate μ_i is an approximation to the empirical average of all the instance μ_{ij}s in that group. Therefore, we formulate the learning problem as a least square regression that tries to minimize the squared error over each group label μ_i and the empirical average of all μ_{ij}s given by the model. That is, the error term for group G_i is $(\mu_i - \frac{1}{n_i}\sum_j \mu_{ij})^2$, where μ_i is given by the annotators and $\mu_{ij} = P(y_{ij} = 1|\boldsymbol{x}_{ij}; \boldsymbol{\theta})$ is given by the model. Further after weighting each term by group size n_i, the overall loss function is defined as:

$$L(\boldsymbol{\theta}) = \sum_i \{\frac{n_i}{n}(\frac{\sum_j \mu_{ij}}{n_i} - \mu_i)\}^2 + \lambda R(\boldsymbol{\theta})$$

Here $n = \sum_i n_i = |\mathcal{U}|$ and $R(\boldsymbol{\theta})$ is the regularization term weighted by a constant $\lambda > 0$.

However, providing an exact μ_i may be hard for a human. Instead we query a range label which has been interpreted as a Beta distribution $Beta(a_i, b_i)$ in Section (2). Then we take the expectation of u_i on the loss function to integrate out μ_i. Therefore the final loss function can be modified as:

$$\begin{aligned} L(\boldsymbol{\theta}) &= \sum_i \underline{\mathbb{E}_{\mu_i}}\{\frac{n_i}{n}(\frac{\sum_j \mu_{ij}}{n_i} - \mu_i)\}^2 + \lambda R(\boldsymbol{\theta}) \\ &= \frac{1}{n^2}\sum_i \{(\sum_j \mu_{ij})^2 - 2n_i \sum_j \mu_{ij}\underline{\mathbb{E}(\mu_i)} + n_i^2 \underline{\mathbb{E}(\mu_i^2)}\} + \lambda R(\boldsymbol{\theta}) \end{aligned}$$

As the above equation indicates, the main difference of using range labels is to use the first and second moments of μ_i to approximate the exact μ_i. $\mathbb{E}(\mu_i)$ preserves the major information of the point μ_i, while $\mathbb{E}(\mu_i^2)$ permits more uncertainty of μ_i.

Learning We want to find the best parameters $\boldsymbol{\theta}^*$ that minimize $L(\boldsymbol{\theta})$. This loss function can generalize to any classifier that outputs instance-level probabilities used with differentiable objective functions. If the second order derivative is readily available, one can apply Newton's optimization to minimize the loss. For more complicated models, one can learn parameters using a gradient-based optimization method, such as BFGS [4].

5 Experiments

We conduct an empirical study to evaluate our proposed approach on 3 real binary classification data sets collected from UCI machine learning repository [1]. Please see Table 1. *Wine* has been used widely in previous work [9, 14]; *Music* is high-dimensional and thus instance-based labeling can be time-consuming; *Seismic* has an unbalanced class distribution, and instance-based labeling may not cover the minor class information properly.

Methods Tested We compare our approach (LSLG) to four related methods: (1) Density-Weighted Uncertainty Sampling (DWUS) [11] which combines uncertainty sampling and data density on deciding which instance to be queried next; (2) RIQY, is a state-of-the-art group-based active learning approach that also queries groups. (But they did not deal with group learning problem, as they directly propagate group labels to instances.) (3) Hierarchical Sampling (HS) [2] which uses a hierarchical tree to guide instance sampling; (4) Multi-Instance Active Learning (MIAL), which queries instances in positive bags in Multiple-Instance Learning framework.

Soft Group Label Simulation For group queries required by our approach, we simulate the human feedback as RIQY did, by counting the class proportion based on instance labels within each group's description region represented by conjunctive pattern. To simulate a variety of range labels in our framework, we experiment with four separate levels of precisions: 5%, 10%, 20% and 30%, where each one is a fixed width for all the range labels in that experiment. For example, "LSLG(10%)" means that we run our method with all the group range labels simulated at a fixed width=10% (e.g.

Table 1: 3 UCI data sets

Name	Description	# Data	# Features	Positive Class	Feature Type*
Seismic	Seismic bumps	2584	18	7%	Num-Ord-Cat
Music	Music origin	1059	68	53%	Num
Wine	Wine quality	4898	11	67%	Num

*'Num', 'Ord' and 'Cat' stand for Numerical, Ordinal and Categorical respectively.

60%∼70% positive). To work out the exact range label provided for each group query, we perform the following steps: (1) find all instances in $\mathcal{U}$ that fall into the group's description (i.e. a hypercube); (2) count and calculate the positive portion among the included instances, marked as p; (3) Add a uniform noise to p: $p_\epsilon = p + \epsilon$ where ϵ follows $Uniform(-w/2, w/2)$, and w is the fixed range label width; (4) A candidate range label appears as $[p_\epsilon - w/2, p_\epsilon + w/2]$; (5) Fix the candidate label to be within [0,1] by truncating the out portion.

Model Setting We employ Logistic Regression as the base model for all methods. Also there are two hyper-parameters in our learning phase. The first is α that is used to interpret a range label into a Beta distribution; and the other is λ that serves as the penalty in regularization. We have experimented many times by cross-validation and find out our model is not sensitive to their choices. α can be picked $\in [0.01, 0.1]$ and $\lambda \in [10^{-5}, 10^{-2}]$. By default, we set $\alpha = 0.05$ and $\lambda = 0.001$.

Evaluation Metrics We adopt Area Under the Receiver Operating Characteristic curve (AUC) to evaluate the generalized classification quality of Logistic Regression on the test data. Our graphs will plot the AUC scores iteratively after each $k \leq 200$ queries are posed.

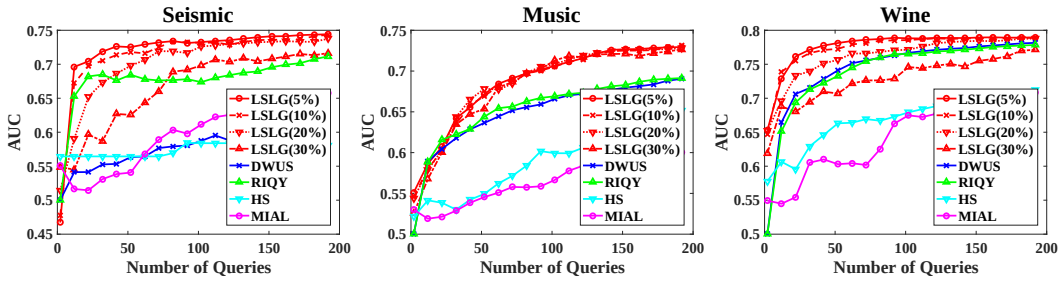

Figure 1: Performances of different methods on 3 UCI data sets.

Experiment Results The results are shown in Figure 1. Overall, when range labels are fairly precise (i.e. range width $\leq$20%), our LSLG (in red lines) outperforms other methods. There are two primary strengths: firstly, our group queries are more informative than the same number of instances queries and our group learning algorithm benefits from the richer class information provided by groups. Secondly, our active refinement of groups can select the group that can potentially lead to maximum model change and thus accelerates the whole learning process.

Effects of Range Label Precision We have experimented our framework with four levels of precisions. According to the results (shown in red lines with different styles) , we observe that for all datasets, the performance of our method gradually drops as the range label uncertainty increases from 5% to 30%, as expected. We recommend a robust range width as 20%, based on our results.

6 Summary

We propose a group-based active learning framework that can efficiently learn instance-based classifiers from soft-labeled groups. We have dealt with the problems of group generation, representation and label collection when labeled groups are not readily available. Therefore, our framework is able to generalize to conventional binary classification tasks and is best suited when providing group labels is more feasible and less costly than instance labeling. In future work, real user studies are necessary to further evaluate the feasibility and efficiency of our proposed approach.

Acknowledgement

The work presented in this paper was supported by NIH grants R01GM088224 and R01LM010019. The content of the paper is solely the responsibility of the authors and does not necessarily represent the official views of NIH.

References

[1] Arthur Asuncion and David Newman. Uci machine learning repository, 2007.

[2] Sanjoy Dasgupta and Daniel Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215. ACM, 2008.

[3] Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In *European Conference on Computer Vision*, pages 562–577. Springer, 2014.

[4] Geof H Givens and Jennifer A Hoeting. *Computational statistics*, volume 710. John Wiley & Sons, 2012.

[5] Dimitrios Kotzias, Misha Denil, Nando De Freitas, and Padhraic Smyth. From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 597–606. ACM, 2015.

[6] Hendrik Kück and Nando de Freitas. Learning about individuals from group statistics. *CoRR*, abs/1207.1393, 2012.

[7] Novi Quadrianto, Alex J Smola, Tiberio S Caetano, and Quoc V Le. Estimating labels from label proportions. *Journal of Machine Learning Research*, 10(Oct):2349–2374, 2009.

[8] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[9] Parisa Rashidi and Diane J Cook. Ask me better questions: active learning queries based on rule induction. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 904–912. ACM, 2011.

[10] Nicholas Roy and Andrew McCallum. Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown*, pages 441–448, 2001.

[11] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

[12] Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. In *Advances in neural information processing systems*, pages 1289–1296, 2008.

[13] Joe H Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244, 1963.

[14] Yanbing Xue and Milos Hauskrecht. Active learning of classification models with likert-scale feedback. In *SIAM Data Mining Conference, 2017*. SIAM, 2017.