Active Learning with Crowdsourced Labels

Matthew Alger
The Australian National University

April 16, 2016

Crowdsourcing provides an active learning domain where many standard active learning assumptions are broken: There is no longer just one labeller, labellers may be non-expert, labellers may not be independent, labellers' accuracy may differ depending on the examples presented, and different labellers may have different accuracies.

Yan et al.[1] introduce a probabilistic model of the crowdsourced active learning problem. Denote examples as $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ with $\boldsymbol{x}_i \in \mathbb{R}^D$, true (unknown) labels as $z_1, \ldots, z_N$, and labels given by the labeller t as $y_1^t, \ldots, y_N^t$. Not all y_i^t are observed and generally no z_i are observed. Denote the $N \times D$ matrix of all examples as X, the $N \times 1$ matrix of all true labels as Z, and the $N \times T$ matrix of all labeller-generated labels as Y (where T is the number of labellers). Then

$$p(Y, Z \mid X) = \prod_i p(z_i \mid \boldsymbol{x}_i) \prod_{t=1}^{T} p(y_i^t \mid \boldsymbol{x}_i, z_i).$$

This model makes the label y_i^t dependent on not only the true label z_i but also the specific example $\boldsymbol{x}_i$. As such, it addresses the problem of labellers' accuracy differing depending on the examples presented as well as differing from each other in general. $p(z_i \mid \boldsymbol{x}_i)$ models the likelihood; Yan et al. use logistic regression:

$$p(z_i \mid \boldsymbol{x}_i) = (1 + \exp(-\boldsymbol{a} \cdot \boldsymbol{x}_i - \beta))^{-1}$$

$p(y_i^t \mid \boldsymbol{x}_i, z_i)$ models the labeller; for binary classification, Yan et al. use a Bernoulli model with

$$p(y_i^t \mid \boldsymbol{x}_i, z_i) = (1 - \eta_t(\boldsymbol{x}_i))^{|y_i^t - z_i|} \eta_t(\boldsymbol{x}_i)^{1 - |y_i^t - z_i|}$$

where η_t is a logistic function with parameters $\boldsymbol{w}$ and γ. This model can be trained with expectation maximisation.

Yan et al.[2] use this model to select an unlabelled example, and then to select a labeller to show the example to. First, they select an unlabelled example using uncertainty sampling[3]. This amounts to finding $\tilde{\boldsymbol{x}}$ such that

$$\tilde{\boldsymbol{x}} = \min_{\boldsymbol{x}_i} \left(\frac{1}{2} - p(z_i \mid \boldsymbol{x}_i) \right)^2$$

Under logistic regression, this defines a hyperplane of $\boldsymbol{x}$ that we may select to label:

$$\boldsymbol{\alpha} \cdot \boldsymbol{x} + \beta = 0$$

We then want to choose a point on this hyperplane and a labeller such that the labeller has minimum error — i.e., we want to find $\tilde{\boldsymbol{x}}$ and $\tilde{t}$ such that

$$\tilde{\boldsymbol{x}}, \tilde{t} = \min_{\tilde{\boldsymbol{x}}, \tilde{t}} \eta_t(\tilde{\boldsymbol{x}})$$

Choosing both examples and labellers in this way results in improved performance over just choosing the examples (and dealing with label noise by majority vote) and just choosing the labeller (and randomly sampling examples).

Mozafari et al.[4] make use of two nonparametric bootstrap methods to decide which examples to present to the crowd in a binary classification task. For a classifier θ trained on a set of training data L, and a data point $u \in L$, we want to find the uncertainty in $\theta_L(u)$. If we had many different L drawn from the same distribution, then by evaluating $\theta_L(u)$ for all of these different L, we could measure properties of the distribution of $\theta(u)$, such as the variance. However, we generally can't draw more L

from the original distribution. This is where a nonparametric bootstrap method comes in: Consider L as a proxy for the original distribution L was drawn from, and draw sets of independent and identically distributed samples from L with replacement. These sets are called "bootstrap replicates". Bootstrap replicates should have the same cardinality as L. Training a classifier on each of the bootstrap replicates then allows the distribution of $\theta(u)$ to be approximated. For bootstrapping to work, it is sufficient that θ is smooth. By using bootstrapping, the two methods proposed by Mozafari et al. work on most classifiers (most importantly, non-probabilistic classifiers) and treat these classifiers as "black boxes" (i.e., the internal state of the classifier is not used). According to the paper, bootstrapping gives less biased estimates of uncertainty than entropy- or margin-based approaches.

The two methods are the Uncertainty method and the MinExpError method. Both methods generate scores for each data point u. Uncertainty is a modification of uncertainty sampling[5], and is given as

$$\text{Uncertainty}(u) = \text{var}(\theta_L(u))$$

where the variance is found by bootstrapping. MinExpError scores data points higher the more that their labelling affects the model. Let l be the label of u found by θ_L. Train k classifiers on k bootstrap replicates and use these to generate k labels $l_1, \ldots, l_k$ for u. Then, use these to estimate the probability l is incorrect as

$$\hat{p}(u) = \frac{\sum_{i=1}^{k} 1(l_i = l)}{k}$$

where $1(c)$ is 1 when c is true and is 0 otherwise. The MinExpError score is then given by

$$\text{MinExpError}(u) = \hat{p}(u)\hat{e}_{\text{right}} + (1 - \hat{p}(u))\hat{e}_{\text{wrong}}$$

where e_{right} is the error of the classifier trained on L and (u, l) and e_{wrong} is the error of the classifier trained on L and $(u, 1 - l)$.

Mozafari et al. also propose a method of dealing with crowd label noise, by dynamically finding the level of redundancy needed for different subsets of unlabelled examples.

References

[1] Yan Yan, Rómer Rosales, Glenn Fung, Mark W Schmidt, Gerardo H Valadez, Luca Bogoni, Linda Moy, and Jennifer G Dy. Modeling annotator expertise: Learning when everybody knows a bit of something. In *International conference on artificial intelligence and statistics*, pages 932–939, 2010.

[2] Yan Yan, Rómer Rosales, Glenn Fung, and Jennifer G. Dy. Active learning from crowds. *Proceedings of the 28th International Conference on Machine Learning*, pages 1161–1168, 2011.

[3] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers, 1994.

[4] Barzan Mozafari, Purnamrita Sarkar, Michael J. Franklin, Michael I. Jordan, and Samuel Madden. Active learning for crowd-sourced databases. *CoRR*, abs/1209.3686, 2012.

[5] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2):201–221, 1994.