Application of Active Learning to Astronomy Time Series

Wenshuai Ye

December 21, 2015

Active learning is a special case of semi-supervised machine learning in which a learning algorithm is able to interactively query the user to obtain the desired outputs at new data points. It is useful when we have a small portion of labeled data along with a larger portion of unlabeled data. In this report, we will discuss active learning from a Bayesian Modeling perspective. Section 1 discusses briefly the background and introduction of the problem we are faced, whereas section 2 introduces the model framework. Section 4 concludes the report.

1 Background and Introduction

Object labels in Astronomy proves to be relatively difficult to obtain, and one usually has to rely heavily on manual labeling by domain experts to improve the quality of the data. Through this process, a hierarchical model can be developed as we evaluate the existed models, objects, and domain experts (users).

Suppose we have an existed model M that attempts to predict the label of an object with a probability (confidence) corresponding to each possible class. To evaluate the prediction further, we develop an application that allows users to label the time series object by answering Yes / No to the random label generated. Of all the objects in the database, some of them have correct labels. The purpose of adding them lies in evaluating both the objects and the users. We can later incorporate this information and develop another hierarchy.

2 Bayesian Hierarchical Model

2.1 User Credibility

The credibility of a user depends on the accuracy of his / her answers when the application shows a time series object with the correct label. Below we model the credibility of a single user with the subscript j to denote the user.

Prior Distribution (a shared x for all classes)
We model the credibility of the user with a beta distribution. Namely,

$$f_j(x) \propto x^{a_j-1}(1-x)^{b_j-1} \tag{1}$$

where a_j and b_j are hyper-parameters to denote prior information on the user. We choose a uniform prior ($a_j = b_j = 1$) if we have no prior information. x is a variable that captures the credibility information of the user and falls in $[0, 1]$

Likelihood Function
The likelihood function can be modeled as a binomial distribution. When the application gives the correct label and the user identifies it correctly, we add a count to the credibility of the user. If the user fails to answer correctly by either giving a false positive or false negative response, we add a count to the flip side.

$$P(d) \propto \prod p_j^{\#I(answer==true)}(1-p_j)^{\#I(answer\neq true)} \tag{2}$$

$$\propto p_j^{\#I(answer==true)}(1-p_j)^{\#I(answer\neq true)} \tag{3}$$

Posterior Distribution
Using Bayes rules, we can derive the posterior distribution of x.

$$f_j(x|d) \propto x^{a_j+\#I(answer==true)-1}(1-x)^{b_j+\#I(answer\neq true)-1} \tag{4}$$

where $a\prime_j = a_j + \#I(answer == true)$ and $b\prime_j = b_j + \#I(answer \neq true)$. form the new shape.

Credibility for Each Class
It is easy to expand the model to customize p_j for each class if we assume each class is independent of each other and high credibility of a user on a particular class does not indicate the same credibility of a user on a different one. In this case, the posterior distribution can be written as followed.

$$f_j(\mathbf{x}|d) \propto \prod_{k=1}^{k=K} x_{kj}^{\alpha+\#I(answer==true)-1}(1-x_{kj})^{\beta+\#I(answer\neq true)-1} \tag{5}$$

where x_{kj} indicates the credibility of user j in class k.

Since users credibility will get updated, to introduce another level of uncertainty, we draw x_{kj} from the posterior distribution and use it for the object modeling part.

2.2 Time Series Object Modeling

We introduce a probabilistic model for the label of the time series object. Each object has the probability p_{ik} to fall into class k. Let $\mathbf{p}$ be the probability vector. The prior distribution in this case can be written as a dirichlet distribution.

$$f_i(\mathbf{p}) \propto \prod_{k=1}^{K} p_{ki}^{\alpha_k i-1} \tag{6}$$

$$= p_{1i}^{\alpha_1 i-1} p_{2i}^{\alpha_2 i-1} \cdots p_{Ki}^{\alpha_K i-1} \tag{7}$$

where $\alpha_\mathbf{i}$ a vector corresponding to the shape of the prior distribution of object i. If we have no prior information about the object, we set $\alpha_{1i} = \alpha_{2i} = \cdots =$

$\alpha_{Ki} = 1$ and use a uniform distribution to be our prior so the object has equal probability to fall into each class. If we have prior information, say, an existed model that predicts the probability, we can incorporate this information into the prior by specifying the expected value of the prior distribution equal to the probability from the existed model. Let p_{mki} be the probability from the existed model that object i falling into class k. We can solve the linear equation

$$E(p_{ki}) = \frac{\alpha_{ki}}{\sum_{c=1}^{K} \alpha_{ci}} = p_{mki} \tag{8}$$

for $\alpha_{\mathbf{ki}}$.

With the evaluation of the objects by users gathered from the application, we acquire more knowledge and hence can derive a likelihood function. We model the likelihood function as a multinomial-like distribution. However, instead of adding the full count, we add the credibility of the user as the partial count. The likelihood function can be roughly described as

$$P_i(D|\mathbf{p}) \propto \prod_{k=1}^{K} p_{ki}^{\sum x_{kj}} \tag{9}$$

for object i. In this case, if user j identifies object i to be class k, then x_{kj} will be added to the power of p_{ki} so that high credibility accounts for higher weight. Since multinomial and dirichlet form a conjugate pair, the posterior distribution is still a dirichlet distribution, which can be written as

$$f_i(\mathbf{p}|D) \propto \prod_{k=1}^{K} p_{ki}^{\alpha_{ki} + \sum x_{kj} - 1}. \tag{10}$$

This posterior distribution can be utilizes to determine new labels for the data points. As a result, we label object i to be class k only when we have sufficient confidence. To accomplish this, we can take advantage our labeled data and get the mode (or mean) of the posterior probabilities corresponding to the correct labels. These posterior probabilities can be together constructed an empirical distribution. If the mode (or expected value) of the data point we are making inference exceed a threshold of this empirical distribution, we flag this data point as a new labeled data point.

3 Conclusion

This model and the application discussed in the report simply describes a prototype and needs further testing. For example, if we are interested in adjusting the weight of the prior information from existed models and the likelihood function, we can introduce a weight parameter to tune this so that the posterior distribution of the object label becomes

$$f_i(\mathbf{p}|D) \propto \prod_{k=1}^{K} p_{ki}^{\omega\alpha_{ki} + (1-\omega)\sum x_{kj} - 1}. \tag{11}$$

where ω specifies how much weight we should put into the existed models. However, given the flexibility of the model, it is relatively easy to add another hierarchy if deemed nessary in the future.