CS 165: Foundations of Machine Learning **Winter 2019**

Lecture 14: 02/27/2019

Lecturer: Anima Anandkumar *Scribes: Victor Dorobantu, Andrew Taylor*

A fundamental component of active learning is characterizing uncertainty to better educate the choice of new data. We discuss some methods of approximation, and comment on additional problems with active learning.

14.1 Obtaining Bayesian Uncertainty

Better yet, we are interested in obtaining Bayesian uncertainty cheaply. Instead of modeling the exact posterior, which can be expensive, we formulate the following problem: find another distribution, $q(w|\theta)$, that approximates the data distribution, $P(w|D)$.

The closeness of this approximation can be measured in a number of ways, including the Wasserstein metric or the commonly used Kullback Leibler (KL) divergence. In the latter case, we formulate the optimization problem

$$\min_{\theta} KL(q(w|\theta)||P(w|D)). \tag{14.1}$$

Recall that KL divergence is defined as

$$KL(p||q) = \mathbb{E}[\log(p/q)], \tag{14.2}$$

for probability densities p and q. Intuitively, this stands in as a "ratio of densities."

It is worth noting that optimizing over KL divergence is not appropriate for some settings, such as robust estimation. However, with appropriate choice of parameter class for θ, the cost in the optimization problem is differentiable and the solution can be approximated with local search.

How should we pick such a class of parameters? A good first choice is to pick Gaussian distributions parameterized by mean and covariance, initializing the problem from standard normal. This may be intractable for models with a large number of weights, so this approximation may be applied to only the last layer of a network.

14.2 Active Learning Heuristics

For applications in sequence tagging, we need better ways to characterize uncertainty rather than increasing uncertainty with the length of a sequence. In other words, we need a heuristic for normalizing data. A simple choice is to normalize by the length of a sequence. Interestingly, deep learning models seem to perform well even with a few examples if they are chosen actively with such heuristics.

14.3 Active Learning with Partial Feedback

Since data annotation can be expensive in many contexts, it may be desirable to collect data at several resolutions of annotator feedback. In particular, annotators can be asked coarser (is there a dog in this

picture) or finer (is there a dachshund under the chair in the upper-right corner of this picture) resolution questions, and a good active learning framework should make use of both answers. One way to guide such a data collection process is to ask questions that reduce (in expectation) entropy (high information gain) or the number of potential classes left in a heirarchy (low coarseness). A related method is to reduce the number of remaining classes in a heirarchy, which performed best in empirical studies.

14.4 Problems with Active Learning

Many problems with active learning stem from the inability to roll back a decision to collect data. In particular, this makes retrospective analysis of models difficult and complicates hyperparameter tuning. Bayesian models seem to have had success, which may be attributed to their ability to overcome epistemic uncertainty (as opposed to using softmax).

Additionally, transferring over tasks often leads to overfitting. Unertainty based sampling is heavily biased towards exploitation in the exploration-exploitation tradeoff. For this reason, it may be useful to incorporate bandit approaches into active learning settings. Reinforcement learning (RL) may also be used to learn aquisition functions (hinting at a flavor of meta-learning), but the data needs of RL may simply defeat the purpose of active learning.

14.5 Data Annotation Issues

When data annotation is noisy, multiple annotators have been used to establish majority votes. Of particular importance to active learning is the ability to learn from noisy singly-labeled data. That is, can we learn something meaningful from potentially corrupted data even when only one annotator labels a datum?

In order to address this question, it is necessary to incorporate a model of annotator noise. One such model is to randomly change labels. This may be an appropriate model for annotators who repeatedly press random buttons rather than label data accurately. A resulting confusion matrix can be approximated from the output of a classifier (this is needed as a proxy since the ground truth is unavailable), and the annotators can be reweighted accordingly. This may be appropriate for active learning settings, since empirical studies have suggested that learning systems may in fact be less sensitive to label noise than they are to small datasets.