Deep Active Learning for Image Classification

Hiranmayi Ranganathan; Hemanth Venkateswara; Shayok Chakraborty; and Sethuraman Panchanathan

2017 IEEE International Conference on Image Processing (ICIP)

> **Computer Vision, Deep Learning, Deep Belief Networks, Active Learning, Entropy, Dataset, Labeling, Sampling**
> https://doi.org/10.1109/ICIP.2017.8297020

Abstract

In the recent years, deep learning algorithms have achieved state-of-the-art performance in a variety of computer vision applications. In this paper, we propose a novel active learning framework to select the most informative unlabeled samples to train a deep belief network model. We introduce a loss function specific to the active learning task and train the model to minimize the loss function. To the best of our knowledge, this is the first research effort to integrate an active learning based criterion in the loss function used to train a deep belief network. Our extensive empirical studies on a wide variety of uni-modal and multi-modal vision datasets corroborate the potential of the method for real-world image recognition applications.

Problem Statement and Research Objectives

- A fundamental challenge in training a deep neural network is **the requirement of large amounts of labeled training data**.
    - While gathering large quantities of unlabeled data is cheap and easy, **annotating the data (with class labels) is an expensive process** in terms of time, labor and human expertise.
    - Thus, developing algorithms to minimize human effort in training deep models is of paramount practical importance.
- **Active learning** algorithms automatically identify the salient and exemplar samples from large amounts of unlabeled data and reduce human annotation efforts in inducing a classification model.
    - Even though both **deep learning** and **active learning** have been extensively studied, **research on combining the two is still in a nascent stage**.
        - A deep model is first learned using a conventional loss function.
        - The active sampling condition is then defined based on the posterior probabilities obtained from the last layer or the distance of a sample from the decision boundary.
    - However, **the merit of a deep model** lies in its ability to learn a **discriminating set of features** for a given task.

Proposed Method

In standard active learning settings,

> 1. A classifier is first trained on the labeled data and used to obtain the predictions for the unlabeled data.
> 2. Entropy is then applied to obtain the uncertainty of such a classifier prediction.

In this two-step approach, the unlabeled data does not play a role in training the classifier.
➔ We propose an active learning model where we **combine the entropy measure along with the cross-entropy loss during training**.

1. Labeling Procedure

1. From the unlabeled set X_u, an oracle is given a batch of points B to be labeled.
2. After labeled, the batch B is then combined with the labeled set X_l along with the corresponding labels (added to Y_l) that are provided by the oracle.
3. These data points are removed from the unlabeled set X_u in order to ensure X_l and X_u are disjoint.
4. A new and improved classifier is estimated using the augmented labeled sets $\{X_l, Y_l\}$. This procedure is repeated until we run out of budget to get labeled data from the oracle.

Given the probability of label assignment for a data point, **entropy (from Information Theory)** can be used to obtain **a measure of uncertainty regarding its label assignment**. The set B can therefore be chosen by selecting the data points with the largest uncertainty.

2. Joint Loss for Active Learning

We combine the cross-entropy loss[1] and the entropy[2] to formulate a classifier with a joint loss that is given by,

$$\begin{aligned}\arg\min_{f\in\mathcal{F}}\ E(\mathcal{D};f) &= \frac{1}{n_l}\sum_{i=1}^{n_l} L(f(x_i),y_i)\\ &+\frac{\lambda}{n_u}\sum_{i=n_l+1}^{n} H(f(x_i))\end{aligned}$$

where λ controls the relative importance of the entropy loss.

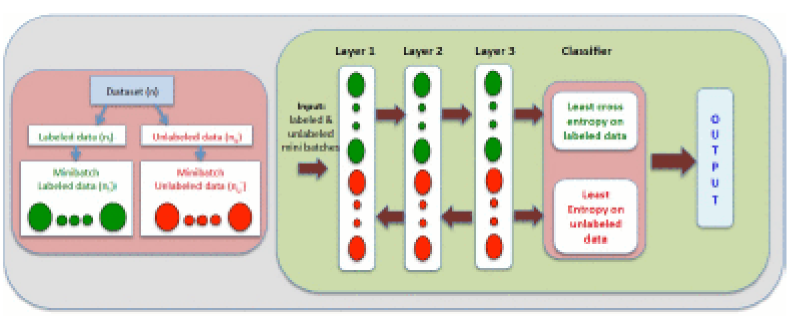
The output of the N-th layer of the network (before the loss) for a data point x_i, is given by the vector h_i^N. The loss in terms of probabilities is given by,

$$\begin{aligned}E(X_l,X_u,Y_l) = &-\frac{1}{n_l}\sum_{i=1}^{n_l}\sum_{j=1}^{C} 1\{y_i=j\}\log p_{ij}\\ &-\frac{\lambda}{n_u}\sum_{i=n_l+1}^{n}\sum_{j=1}^{C} p_{ij}\log p_{ij}\end{aligned}$$

During the training procedure, the derivative $\partial E/\partial h^N$ is back-propagated through the network in order to update the weights of the network.

$$\frac{\partial E}{\partial h_{pq}^N} = \begin{cases}\frac{1}{n_l}(p_{pq}-1\{y_p=q\})\,, & p\in[1,\dots,n_l]\\ \frac{\lambda}{n_u}p_{pq}\left(\sum_j^C p_{pj}h_{pj}^N - h_{pq}^N\right), & p\in[n_l+1,\dots,n].\end{cases}$$

3. Active Learning Network Architecture and Training


Dataset (n)
Labeled data (n_l)
Unlabeled data (n_u)
Minibatch Labeled data (n_l')
Minibatch Unlabeled data (n_u')
Input: labeled & unlabeled mini batches
Layer 1
Layer 2
Layer 3
Classifier
Least cross entropy on labeled data
Least Entropy on unlabeled data
OUTPUT

Our model is a three layer Deep Belief Networks (DBNs).

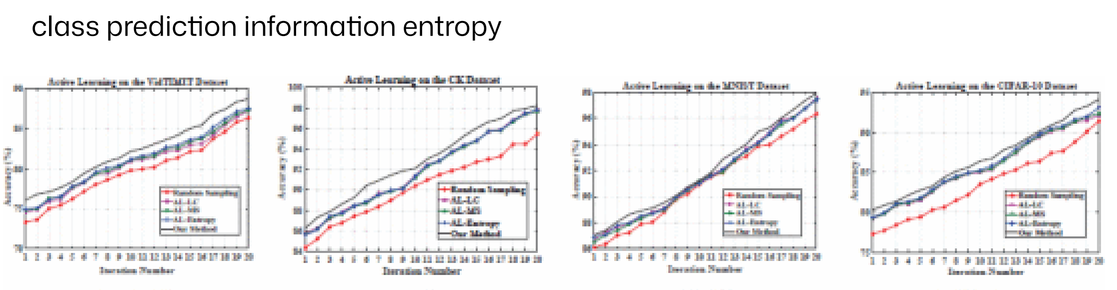
- **One epoch**: When the network has seen all the data points in the training set (labeled and unlabeled).
- **One training iteration t of the active learning algorithm**: Repeating the training procedure over multiple epochs until convergence.
    - At the end of every iteration t, we sample the most informative batch of unlabeled data points (using Equation[2]) to form B.
    - We iterate until we run out of unlabeled data to be labeled or we run out of budget to get them labeled.

Evaluation and Results

➔ The goal was to study the improvement in performance on the test set with increasing sizes of the training set.

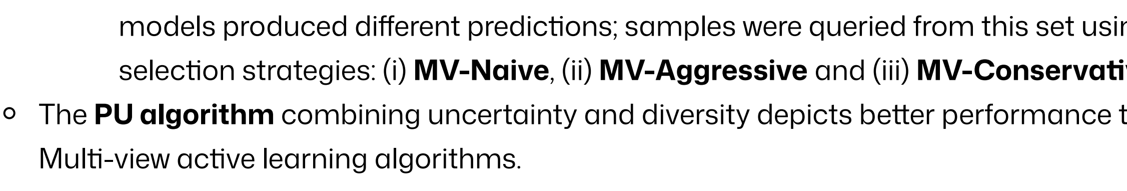
> - For a given batch size k, each algorithm selected k instances from the unlabeled pool to be labeled in each iteration.
>   - After each iteration, the selected points were removed from the unlabeled set, appended to the training set and the performance was evaluated on the test set.
>   - The experiments were run for 20 iterations.

- **Uni-modal datasets**
    1. **Random Sampling**: selects a batch of unlabeled samples at random from the unlabeled pool
    2. **Active Labeling with Least Confidence (AL-LC)**: selects the samples with the smallest of the maximum activations
    3. **Active Labeling with Margin Sampling (AL-MS)**: selects the samples with the smallest separation between the top two class predictions
    4. **Active Labeling with Entropy (AL-Entropy)**: selects the unlabeled samples with the largest class prediction information entropy



(a) VidTIMIT (b) CK (c) MNIST (d) CIFAR-10

- **Multi-modal datasets**
    - Muslea et al.[3] proposed the **Multi-view (MV)** algorithm in which a separate classification model was trained for each modality (view).
        - A set of *Contention Points* was identified from the unlabeled set, where at least two models produced different predictions; samples were queried from this set using three selection strategies: (i) **MV-Naive**, (ii) **MV-Aggressive** and (iii) **MV-Conservative**.
    - The **PU algorithm** combining uncertainty and diversity depicts better performance than the Multi-view active learning algorithms.



(a) EmoFBVP (b) Mind Reading (c) EmoFBVP: Face and Voice (d) EmoFBVP: Face, Body and Voice

Notes

- Cross-Entropy Loss: $L(f(x_i),y_i) = -\sum_{j=1}^{C} 1y_i = j\log f_j(x_i), \forall i\in[1,\dots,n_l]$ ↵
- Entropy: $H(f(x_i)) = -\sum_{j=1}^{C} f_j(x_i)\log f_j(x_i), \forall i\in[n_l+1,\dots,n]$ ↵
    - $f_j(x_i)$ is the probability of assigning x_i to category j.
    - $p_i := [f_1(x_i), f_2(x_i), \dots, f_C(x_i)]^\intercal$ is the probability vector for x_i
    - $p_{ij} := f_j(x_i) = e^{h_{ij}^N} / \sum_{j'} e^{h_{ij'}^N}$ is the probability that data point x_i belongs to class j.

1. [16] I. Muslea, S. Minton, and C. Knoblock. Active learning with multiple views. In Journal of Artificial Intelligence Research, 2006. ↵