ML Tech-Report 1

Semi-supervised Learning

Het Dave - 17CH10018
Harsh Choudhary - 17CH30040
Naveen Gupta - 17CH30054
Chaitanya Rai - 17CH10058

Introduction

Nowadays massive raw data is generated everyday thanks to the development of data gathering and storage techniques. However manual labeling of the large dataset is very time and labor-consuming. In practice the number of unlabeled data is often far greater than that of labeled data. The cost associated with the labeling process thus may render large, fully labeled training sets infeasible, whereas acquisition of unlabeled data is relatively inexpensive. In such situations, semi-supervised learning can be of great practical value. Semi-supervised learning is also of theoretical interest in machine learning and as a model for human learning. It is an approach to machine learning that combines a small amount of labeled data with a large amount of unlabeled data during training. It falls between unsupervised learning and supervised learning. That's why, Semi supervised learning is of great interest in a wide variety of research areas, including natural language processing, speech synthesizing, image classification, genomics etc. Existing semi-supervised learning models can be categorized into three main categories: unsupervised feature learning approach, graph-based regularization approach and multi-manifold learning approach.

But In order to make any use of unlabeled data, some relationship to the underlying distribution of data must exist. Semi-supervised learning algorithms make use of at least one of the following assumptions:

Continuity assumption: If two samples in a high-density region are close, then their corresponding outputs are also close

Cluster assumption: If two samples are in the same cluster, they are likely to be the same class.

Manifold assumption: If two samples are close in high dimension, they are comparable in a low-dimension manifold. Manifold regularization is inductive and naturally handles unobserved instances.

Methods

Generative models

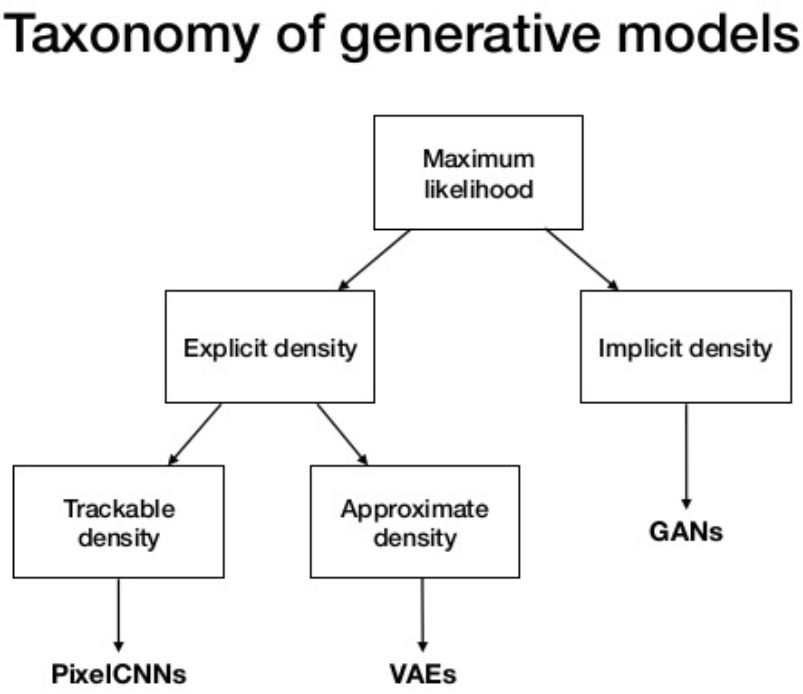
Generative models assume that the distributions take some particular form p (x | y , θ) parameterized by the vector θ. If these assumptions are incorrect, the unlabeled data may actually decrease the accuracy of the solution relative to what would have been obtained from labeled data alone. However, if the assumptions are correct, then the unlabeled data necessarily improves performance.

The unlabeled data are distributed according to a mixture of individual-class distributions. In order to learn the mixture distribution from the unlabeled data, it must be identifiable, that is, different parameters must yield different summed distributions. Gaussian mixture distributions are identifiable and commonly used for generative models.

The parameterized joint distribution can be written as p (x , y | θ) = p (y | θ) p (x | y , θ) by using the chain rule. Each parameter vector θ is associated with a decision function f θ (x) = argmax y p (y | x , θ). The parameter is then chosen based on fit to both the labeled and unlabeled data, weighted by λ:

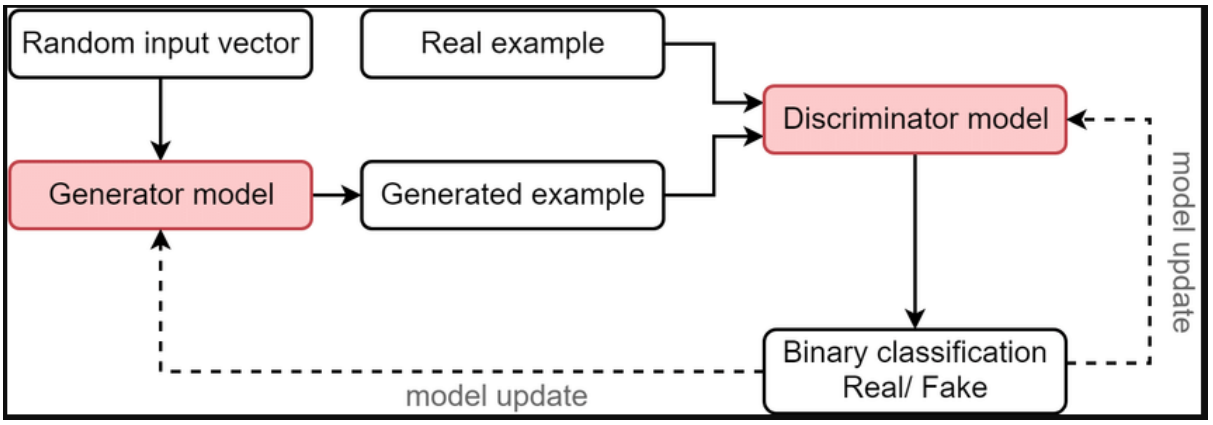
$$\underset{\Theta}{\operatorname{argmax}} \left(\log p(\{x_i, y_i\}_{i=1}^{l} | \theta) + \lambda \log p(\{x_i\}_{i=l+1}^{l+u} | \theta) \right)$$

Generative models require large amounts of data to generalize the results and learn patterns. There are several types of generative models which could be used for learning. The taxonomy for generative models is as follows:



And Now-a-days, out of all these different types of generative models, recently GANs have gained so much popularity due to their learning capability and achieving the human level intelligence or even surpassing them in several tasks. GANs are so powerful that they can even learn to synthesise speech correlated to intricate human expressions, corresponding head pose or eye gaze. This further can be applied to diagnose depression or other mental disorders in its infancy. The meteoric rise in the popularity of GANs has been accompanied by its equally increasing capability to reach domains which have been immune to AI.

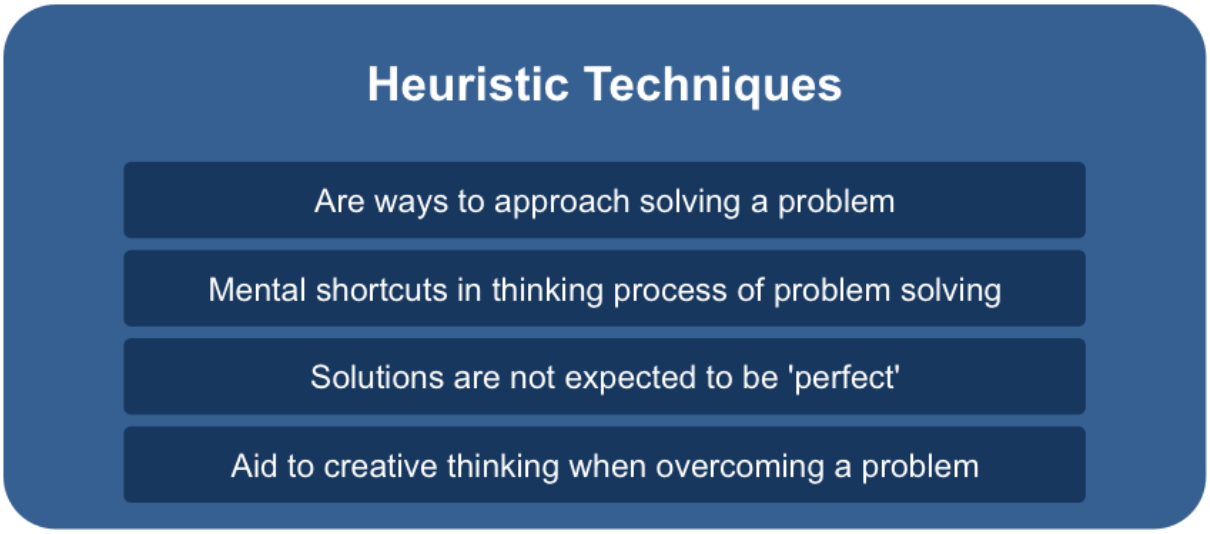
Here is a workflow describing the training of GAN model:



Heuristic approaches

Heuristic Methods are applied in Machine Learning when there are no concrete solutions available, or there are no accurate solutions to be implemented. It compensates one of the parameters from the main facets of the judging parameters for a model's speed.

It ranks and branches the results after each iteration according to the scores and then chooses the best out of them as it proceeds. There are multiple ways to understand the implementation of the approach, but it uses hypothesis to convert unlabelled data to labeled data.



Heuristic Approaches is making use of unlabeled training examples within a supervised learning framework but also use some methods for semi-supervised learning, which are not primarily constructed to be learning from both unlabeled and labeled data.

For example,

- In the first step of unsupervised learning, we have to decide on the representation: Distance Matrix or Kernel; for both the labeled and unlabelled elements.

- For the second step, supervised learning is applied to the labeled elements.

- In the third step, applying concise representation by reducing the dimension by using the previous step. Also, finally applying the steps learned previously, like low-density or graph-based methods for semi-supervised learning.

- Finally, you iteratively perform improved representations and then implement the previous step on drawing to enhance performance.

Self-training is a wrapper method for semi-supervised learning. It is integral for semi-supervised learning to have the initial step as training on labeled data. Using this, the classifier is constructed and applied on unlabelled data to convert them into labeled data to implement supervised learning algorithms. Only the labels with the best scores are added in the improvement step.
The advantages of heuristic approaches are:

- Heuristics can lead to poor decision-making based on a limited data set, but the speed of decisions can sometimes make up for the disadvantages.

- It can help in semi-supervised learning as the classifier can be based on the labeled part of the training data, using which the unlabelled data can be trained. This is advantageous in times of constraint.

- Heuristic approaches also combine the use of different semi-supervised learning and track the best structure for the classifier at each step. This makes things easier to implement.

The conclusion is, the accuracy, prediction, and classification of the model will suffer because of the compromise due to speed. This is allowed as the model compensated itself because it could not have been implemented otherwise easily based on the limited training set which was labeled.

Graph-based methods

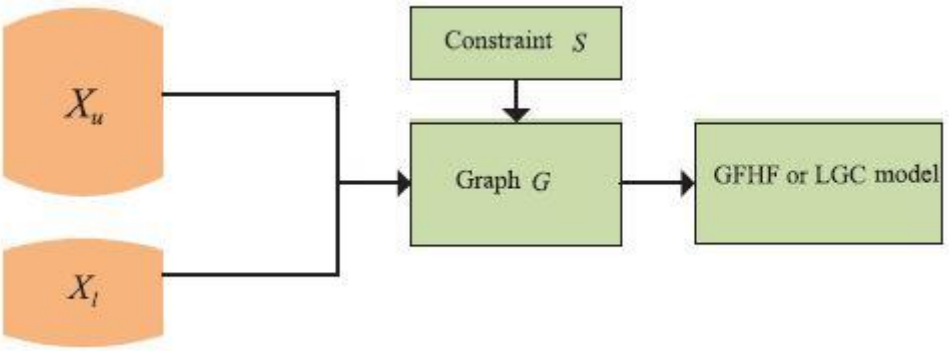
The main purpose of Graph-based methods is to construct better graphs to represent the given data structure. It is done in two steps:

1. Construct a graph from all the labeled and unlabeled samples.
2. The labeled information can be propagated to the unlabeled samples through the graph.

Note: Dataset X consists of small amounts of labeled samples and large amounts of unlabeled samples, i.e., X = { X l , X u } where X l = { (x i , y i) } l i =1 represents the labeled portion and X u = { x i } u i =1 represents the unlabeled samples.

Transductive learning

The aim of the transductive algorithm is to learn the predicted function from the labeled samples and unlabeled samples to estimate the labels for only the unlabeled samples.



A. GFHF and LGC models

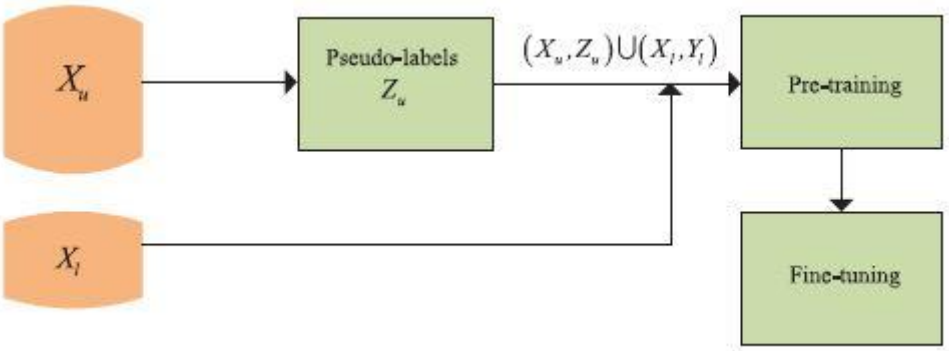
The goal of the GFHF algorithm is to obtain the smooth label assignment over the graph, which can initially prop- agate the label information of labeled samples to the unlabeled samples via label propagation. Also, the harmonic property means that the value of an unlabeled sample is the weighted average of labeled neighbor samples.

Compared to the GFHF model, LGC has two major differences
(1) the LGC model aims to spread every sample's label information to its neighbors until achieving a global convergence

2) The label for each sample is penalized by the constraint, which ensures the regular graph in the influence of high degree samples.

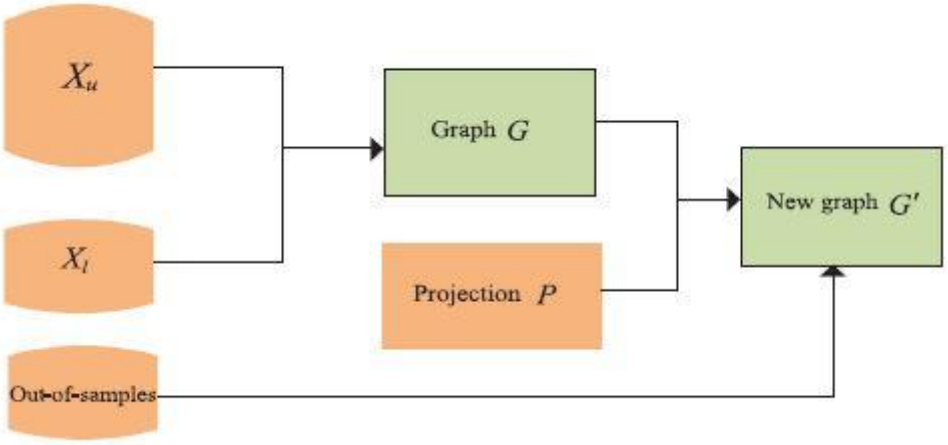
B. Sparse models
Sparse representation graphs also called l1 graphs have been proved to be critical for the high-performance classification of high-dimensional data, which can automatically seek the sparsest representation between the various classes. These algorithms can be efficiently computed by convex optimization. In addition, low-rank representation has been proposed to perform robust recovery of subspace structures.

Neural Networks



Neural network-based methods combine unsupervised and supervised learning to perform semi-supervised learning, which effectively generalizes from small labeled samples to large unlabeled samples.
Most of the semi-supervised neural networks can be divided into two stages. To begin with, we can obtain the pseudo-labels of the abundant unlabeled samples by cluster method, which combines the original labeled samples together to pre-train a neural network. Then using the available labeled samples to fine-tune the neural network.

Inductive learning



The problem to be solved in the inductive learning field is the typical out-of-sample. The goal of inductive learning is that it can predict the labels of the new samples via graph structure when the new (unseen) samples come into the graph structure between the original labeled and unlabeled samples. Firstly, the graph can be constructed by labeled and unlabeled samples. Then, imposing the constraint such as projection classifier and the other constraint exploited features with class memberships on the graph to obtain the new graph for inductive learning. Finally, the obtained new graph for inductive learning is able to handle the out-of-samples problem.

Low - density separation

Algorithm for classification

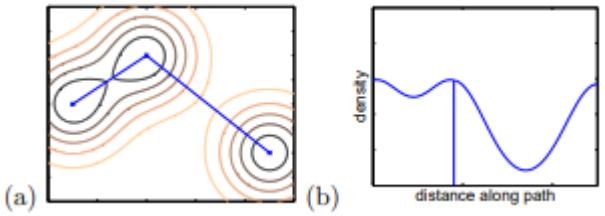
Given a small set of labeled data and a large set of unlabeled data, semi-supervised learning (SSL) attempts to leverage the location of the unlabeled data points in order to create a better classifier than could be obtained from supervised methods applied to the labeled training set alone. Effective SSL imposes structural assumptions on the data, e.g. that neighbors are more likely to share a classification or that the decision boundary lies in an area of low density.

We propose three semi-supervised algorithms: 1. deriving graph-based distances that emphasize low-density regions between clusters, followed by training a standard SVM; 2. optimizing the Transductive SVM objective function, which places the decision boundary in low-density regions, by gradient descent; 3. combining the first two to make maximum use of the cluster assumption.

According to the cluster assumption, the decision boundary should preferably not cut clusters. A way to enforce this for similarity-based classifiers is to assign low similarities to pairs of points that lie in different clusters. To do so, we construct a Parzen window density estimate with a Gaussian kernel of width $1/\sqrt{2}\,\sigma$.

$$\hat{p}(\mathbf{x}') = \frac{1}{\sqrt{\pi}\sigma} \sum_{i=1}^{n+m} \exp\left(-\frac{||\mathbf{x}' - \mathbf{x}_i||^2}{\sigma^2}\right).$$

If two points are in the same cluster, it means that there exists a continuous connecting curve that only goes through regions of high density; if two points are in different clusters, every such curve has to traverse a density valley. We can thus define the similarity of two points by maximizing over all continuous connecting curves the minimum density along the connection, but this is hard to compute.



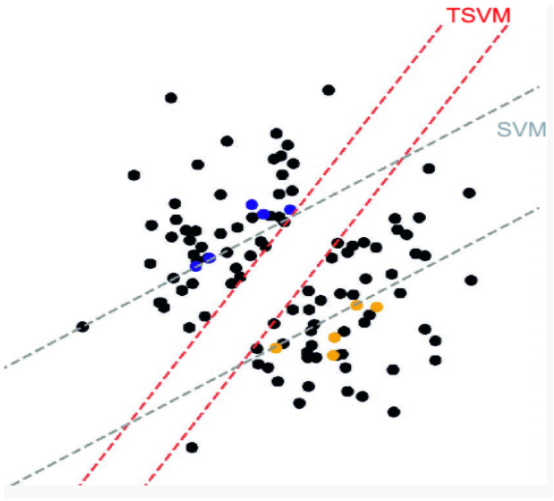
Optimal connecting curves are well approximated by paths of short distance edges on a graph.

Mathematical working:

Low-density separation methods attempt to find a decision boundary that best separates one class of labeled data from the other. The quintessential example is the transductive support vector machine (tsvm: [1,6,8,15,21,30]), a semisupervised maximum-margin classifier of which there have been numerous variations. As compared to the standard svm (cf., e.g., [2,32]), the tsvm additionally penalizes unlabeled points that lie close to the decision boundary. In particular, for a binary classification problem with labels yi ∈ {−1, 1}, it seeks parameters w, b that minimize the non-convex objective function where fw,b : Rd → R is the linear decision function fw,b(x) = w · x + b, and H(x) = max(0, 1 − x) is the hinge loss function

$$J(w,b) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l} H(y_i \cdot f_{w,b}(x_i)) + C^* \sum_{i=l+1}^{u} H(|f_{w,b}(x_i)|),$$

The hyperparameters C and C∗ control the relative influence of the labeled and unlabeled data, respectively. Note that the third term, corresponding to a loss function for the unlabeled data, is non-convex, providing a challenge to optimization.



A schematic for TSVM(Transductive SVM) segmentation. The grey lines correspond to maximum margin separation for labeled data using a standard svm; the red lines correspond to additionally penalizing unlabeled points that lie in the margin.