Semi-supervised learning algorithm

A semi-supervised learning algorithm combines elements of both supervised and unsupervised learning, utilizing both labeled and unlabeled data to train a model. This approach allows for the efficient utilization of limited labeled data, thereby enhancing the model's performance and generalization, making it particularly suitable for situations where data is scarce or labeling is costly.

1，Label Propagation

Label Propagation involves propagating labels from known samples to unknown samples. Label Propagation is a type of semi-supervised learning method, typically used for node classification problems in graph or network data. The basic principle involves using data nodes with known labels (seed nodes) to propagate labels and classify unlabeled nodes. The Label Propagation algorithm attempts to make the labels of neighboring nodes similar, thereby achieving label propagation across the entire graph data.

Basic Principles:

1. Seed Nodes: The Label Propagation algorithm requires some data nodes with known labels, often referred to as seed nodes or initial label nodes. Typically, these seed nodes contain some true labels of the data.
2. Label Propagation: At the start of the algorithm, the labels of the seed nodes are used as initial labels. Then, by iteratively updating the labels of the unlabeled nodes to become similar to those of their neighboring nodes. This similarity is often based on relationships or similarity measures between the data nodes.
3. Label Update Rules: Label Propagation algorithms typically employ the following rules to update node labels:

- For each unlabeled node, calculate the average of its neighboring nodes' labels (or weighted average labels) as the new label.
- Repeat the above step until the labels no longer significantly change or until a predefined number of iterations is reached.

The effectiveness of the Label Propagation algorithm is influenced by the selection of seed nodes, the method used for calculating weights, and the convergence criteria. The algorithm typically handles label propagation problems in graph data well, but in some cases, issues like label oscillation or convergence to an unstable state might arise. Therefore, it's crucial to carefully adjust parameters and monitor the algorithm's convergence to address these potential problems.

2，Self-training

Self-training is a semi-supervised learning method based on the fundamental principle of training a model and making label predictions using both labeled and partially unlabeled data.

Basic Principles:

1. Labeled Data: Initially, a small set of labeled data is used for the initial model training. This step represents the traditional supervised learning phase, used to establish the initial model.
2. Label Propagation: Subsequently, the trained model is used to predict labels for the unlabeled data. These predicted labels are added to the unlabeled data, thereby expanding the labeled dataset.

3. Retraining: The expanded dataset, comprising both the initially labeled data and the unlabeled data with estimated labels, is used to retrain the model. This new model often better captures the data distribution and features since it incorporates more data.
4. Iteration: The process iterates by performing label propagation and retraining steps until a stopping criterion is met (such as reaching a maximum number of iterations or when label predictions no longer significantly change).

The core of self-training involves the processes of label prediction and retraining.

Label Prediction: Using a trained model to predict labels for unlabeled data. Typically, this involves computing scores for each category (e.g., using the softmax function) and selecting the category with the highest score as the predicted label.

Retraining: Retraining the model using the augmented dataset. This usually involves using labeled data and unlabeled data with estimated labels to compute the loss function and updating model parameters through backpropagation. Retraining can utilize traditional supervised learning algorithms.

Self-training is a useful method for enhancing model performance using unlabeled data, especially in cases where data is scarce. However, self-training also poses challenges, such as the potential accumulation of errors in label propagation, leading to decreased model performance. Therefore, careful consideration of data quality and iteration strategies is essential when applying self-training to achieve optimal performance.