

A Comprehensive Study and Analysis of Semi Supervised Learning Techniques

Kiran Bhowmick
D.J. Sanghvi College of Engineering,
Computer Engineering Department, Mumbai- 400056
India

Meera Narvekar
D.J. Sanghvi College of Engineering,
Computer Engineering Department, Mumbai- 400056
India

Mohammed Aqid Khatkhatay
D.J. Sanghvi College of Engineering,
Computer Engineering Department, Mumbai- 400056
India

Abstract— Semi supervised learning is a technique that tries to draw inferences from partially labeled data. A large amount of data exists due to the technological advances in data generation which includes IoT, Big data, AI etc. But a large part of this data is unlabeled. To exploit the potential of this unlabeled data, semi supervised techniques have proven to be very useful. The research on the semi supervised learning is still in a nascent stage. There exists a large section of semi supervised learning to be explored. This paper introduces to the various techniques of semi supervised learning and provides an extensive analysis on the advantages, disadvantages and applications of these techniques.

Keywords—*Semi-supervised Learning, Self-Training, Co-training, graph based, cluster and label, S3SVM.*

I. INTRODUCTION

TRADITIONALLY there are two fundamentally different types of learning techniques viz., the unsupervised and the supervised learning. In unsupervised learning, the task of any learning model is to primarily find interesting structures or patterns on a given dataset $X = \{x_1, x_2, \dots, x_n\}$. The model's accuracy depends upon the fact that how well these patterns with similar data instances formed or not. So, it is important to check if similar items are grouped closer in the structure or not. The technique essentially has no information of the expected groups for each data item x_i and is based on the current structure. Supervised learning on the other hand has a prior knowledge of what the group or rather group label should be which is called as class label. Essentially these techniques are provided with a dataset. $X = \{x_1, x_2, \dots, x_n\}$ along with their class labels $Y = \{y_1, y_2, \dots, y_n\}$ and their task is to find a mapping between $X \rightarrow Y$. The accuracy of the model depends on how correctly is it able to map and $x_u \rightarrow y_u$.

Semi Supervised learning is halfway between unsupervised and supervised techniques. In addition to a large amount of unlabeled data, they are also given some amount of labeled data. i.e. $X_l = \{x_1, x_2, \dots, x_l\}$ $Y_l = \{y_1, y_2, \dots, y_l\}$ and $X' = \{x_{l+1}, \dots, x_{l+n}\}$. Acquiring labels to this unlabeled data requires expertise and time and is

expensive. Learning using this partially labeled and unlabeled data makes the semi supervised learning more suitable for real time problems. Basically, semi supervised learning essentially means devising a way to utilize labeled and unlabeled data to create better models [1]. The organization of this paper is as follows: Section 2 describes the general background of the Semi supervised learning; section 3 describes the various semi supervised learning techniques in detail; section 4 describes a detail analysis of these techniques and lastly conclusion and future work in section 5

II. GENERAL BACKGROUND OF SEMI SUPERVISED LEARNING

Machine learning basically consists of two types of learning viz., transductive learning and inductive learning.

A. Transductive Vs Inductive learning

Given labeled and unlabeled data, learning a function f to predict unseen (test) data is Inductive learning [2]. While given the same labeled and unlabeled data the only task is to predict the labels for unlabeled data is Transductive learning. For inductive learning, the learning model is not aware of the test data whereas for transductive learning the training and testing data both are known to the learning model. In the process of inductive learning the first step is the dissemination of labels into labeled set and unlabeled set either randomly or by some algorithmic means. The labeled set is then used for training purpose while the unlabeled set is used for the testing purpose by means of a hypothesis. Predictions are then obtained and crosschecked or validated depending on the method being used. In case of Transductive learning there is no hypothesis involved. The data is randomly split into the training and testing sets. These are then used to obtain predictions based on some evaluation metrics or threshold and a stopping goal test is also involved to avoid over fitting of data instances. Semi supervised learning can be both transductive as well as inductive [3].

B. Semi supervised classification Vs Semi supervised clustering

Semi supervised classification is problem of classifying using labeled and unlabeled data both. Here a classifier is built on the labeled data, predict for the unlabeled and then use this prediction further to train the classifier again [2]. Semi supervised clustering called as constraint clustering is a problem of creating clusters using labeled and unlabeled data both [3]. The semi supervised clustering follows classical assumptions of clustering which says:

Cluster assumption: If points are in the same cluster, they are likely to be of the same class [1]. This puts the constraints for clustering deciding if two points must be in the same cluster (must-link constraint) or two points must not be in the same cluster (cannot-link constraint) and the constraint for dimensionality reduction deciding if two points must be close after the projection.

III. SEMI SUPERVISED LEARNING TECHNIQUES

Some of the known semi supervised learning techniques are self-training, co-training, graph-based methods, multi view learning and mixture models which includes cluster and label models

A. Self -training

Self-training technique makes use of a small amount of initial labeled data to train the model and predict the labels of a small subset of unlabeled data, retrain and re-predict till the entire data set is labeled. The basic algorithm for self-training can be given as follows:

Algorithm 1: Self-Training

X both labeled and unlabeled dataset

X_i where $i=1$ to n ;

X_j where $j=1$ to m ;

n - Number of labeled instances;

m - Number of unlabeled instances;

$m >> n$

While some of X are not labeled do

Train(X_i)

labels \leftarrow Predict(X_j)

$X_i \leftarrow labels$

end

According to Vincent Ng and Claire Cardie [4] self-training is a single- view weakly supervised algorithm. According to C. Rosenberg, M. Hebert, and H. Schneiderman [5] self- training is based on 4 stages. The first stage is to choose a classifier and a small subset labeled data randomly. In the second stage a small subset of unlabeled data is classified and labeled. In the third stage the new labels are assessed and given a probability value as per predefined metrics. In the fourth and final

stage the instances with probability values beyond a certain predefined threshold are added to the training set and the entire process is repeated till the data set is completely labeled. Retraining can also stop after a certain condition is fulfilled.

Since the above stated generic approach relies on a small initial training data set misclassification error is not avoidable in many cases. Additionally, certain assumptions in the predefined metrics or probability threshold may add to misclassification error. According to Goldberg et. al., [6] self-training in cases like the k-nearest neighbors is outlier sensitive that is if there are a large number of outliers the errors in initial prediction may get reinforced to further sets. The presence of noise in the labeled data-set cannot be detected and is carried to further stages and keeps affecting the learner. The decision or predefined metrics such as confidence and threshold are difficult to select. According to Sadarangani and Jivani [7] Self training does not give much information relating to its convergence.

To minimize such misclassifications due to noise or outliers Li and Zhou [8] proposed a modified self-training model known as SETRED: Self- Training with Editing. It is basically a data filtering technique to remove noisy instances from the labeled data to avoid the drawbacks present in the generic model. SETRED makes use of a neighborhood graph in a p-dimensional feature space to actively detect miss classified labels with the help of some local information in the neighborhood graph. Thus, it uses active learning to improve generalization of the training hypothesis. The only drawback with it is that it is sensitive to imbalance data.

In order to address difficulties in selecting the confidence and probability thresholds Livieris et. al., in [9] proposed a new type of self-training SSL known as AAST: Auto-Adjustable Self-Training which uses a number of independent base learners instead of one. Each learner is selected dynamically based on a number of parameters. The algorithm works in two steps. The first step involving selecting the best classifier based on some confidence obtained from unlabeled data instance classification greater than some specified threshold. The second step involves iterative method of training the classifier until a terminating condition is reached. This method is highly successful in avoiding errors due to noise but suffers in terms of computation time for a very large number of independent base learners.

B. Co-Training

Co Training is a semi supervised training method where the data-set is split into two conditionally separate and independent views. The classifier trains on each view separately. After training a probability confidence value is checked against a predefined threshold for both the views. Values having a high confidence in view 2 are appended to the training set of view 1 and vice versa thereby each classifier trains and teaches the other.

According Goldberg et. al. [6], Co-Training makes two major assumptions. The first assumption is that the views are conditionally independent with regard to a class label. If this

is not the case artificial views are created. This assumption was called the independence assumption. The second assumption is that given sufficient data each view is independent enough to label properly. This assumption was called the sufficiency assumption. This approach however takes time. Co-training will work only if the two assumptions are satisfied. It may however happen that in many real-world scenarios these conditions are partially met or are not met at all. We therefore need to carefully decide the views or have required some method to satisfy the assumptions.

The co-training generalized algorithm as follows:

Algorithm 2: Co-Training

L- Data-set labeled;
U- Data-set Unlabeled;
n- Number of labeled instances;
m- Number of unlabeled instances;

X_i where $i=1$ to n

X_j where $j=1$ to m

$m \gg n$

for $i \in U$ **do**

$model_1 = train(L^1)$

$model_2 = train(L^2)$

If $p_{max}m_1(x) > Threshold$ and $p_{max}m_2(x) < Threshold$

then $L^2 \cup P_1(x)$

If $p_{max}m_2(x) > Threshold$ and $p_{max}m_1(x) < Threshold$

then $L^1 \cup P_2(x)$

end

To curtail drawback of the generic co training model, a PAC -style (Probably Approximately Correct) analysis framework was suggested by authors in [10]. A weighted bipartite graph was used to facilitate this assuming full compatibility. It was observed that any two examples that were part of the same component had the same label. The model was experimented on a collection of web pages collected from various colleges that were labeled by hand. Naive Bayes was used in the first step to train the two different classifiers. The values were then assigned to the unlabeled data. The models were to select p-positive and n-negative labels from examples from the set. Another $2p+2n$ examples were randomly chosen and replenished. This was looped for a fixed number of iterations. The results were preliminary in nature and showed that this method had potential benefits but required further study.

A further improvisation for improving the quality and reliability of communication between views is done by presenting a model named as COTRADE, i.e., Confident cOTRAining with Data Editing [11]. This approach had two major stages after the initial learning for a particular number of labeling rounds. In the first stage labeling confidence of each classifier is calculated by applying some data editing techniques. Then in the second stage the most appropriate set of labels is taken from each view to enhance the training set of the other view. It was concluded that COTRADE had the best training methods among the other co-training methods.

C. Graph based Semi Supervised Learning

Representing data in a graphical manner is one of the most easy and efficient methods. A graph can be used to represent data where nodes carry data instances with the corresponding edges representing their relations. Graph based Semi Supervised Learning methods can be scaled easily and efficiently to real world large data sets. The main advantage of using graph based semi supervised learning is that these techniques guarantee convergence for convex objective. According to [12] graph-based Semi Supervised Learning can elucidate to a very small amount of data that is available.

The other works in the world of Graph based semi supervised learning [12] [13] [14] [15] assume that the entire data set (labeled and unlabeled) lies in a low dimensional manifold that can be approximated using a graphical techniques where nodes are data instances and weighted edges represent their relations. It was also shown in [12] how SSL can take place on both shared and distributed memory and how reordering can make scaling graph-based SSL possible with an efficiency as high as 85 %.

There were two assumptions made by [12]. The first assumption was the manifold assumption which stated that irrespective of the data class being labeled or unlabeled, all data items were said to lie on a global low dimensional manifold within a high dimensional space. Secondly there might be a local manifold for each class which can be separated using a decision boundary thereby giving the label. The second assumption was the smoothness assumption which stated that if two points a and b are close on the graph then their corresponding labels are also closer. Closeness here implies some distance metric in terms of the manifold. A high density set on the manifold will have a corresponding high label probability.

D. Gaussian Mixture Models

A mixture model is a probabilistic model which accounts for the existence of a subset within the entire data set. In a given data set many values share some common characteristic; these can be clustered in order to use them for further analysis. In the world of learning there are two types of clustering techniques- Hard clustering and soft clustering. Hard clustering is where each point or data instance is either assigned a cluster or not assigned. No middle ground is held. Soft clustering is where each point

or data instance is assigned a probability or likelihood to be in the clusters present. When hard clustering methods such as k-means or agglomerative are used there is no uncertainty measure to associate a point to the cluster. In order to take care of this Gaussian Mixture Models or GMMs are used. A Gaussian Mixture is a mathematical function consisting of several normal distributions each with a mean μ covariance and mixing probability π . GMM due to its vast adaptiveness is superior to many traditional clustering algorithms. A simple version of GMM was used for MR (Magnetic Resonance) brain image segmentation in [15] by Portela et. al. The main aim of this paper was to improve the segmentation process and to achieve a fast convergence using no labeled set and minimum expert assistance. Initially 3D image slices were clustered and labeled manually by a human expert in to 3 clusters. Each cluster was either gray matter (GM), white matter (WM), or cerebrospinal fluid (CSF). Then GMM was applied on the remaining data. This gave it the advantage of not requiring a new classification step. It was also found that accuracy could improve if prior data is accurate because of GMMs initialization method sensitivity.

To improve the prior knowledge drawback another GMM based model was proposed for Channel-based Authentication Scheme by Gulati et. al., in [16]. The paper proposed the use of a number of GMMs and online parameter tuning. A vast number of wireless channels were used for feature selection. Data separation took place in a high dimensional space. Cluster labeling took place using the initial training messages. The paper concluded by showing that the proposed methods has a very low false alarm rates and miss detection rates.

GMMs can also be adjusted to function as Phonetic Classification [17]. Huang and Hasegawa applied GMM on a unified objective function and also compared a hybrid discriminative method and the generative method. The hybrid discriminative method involved an extra regularization term, likelihood of unlabeled data and a training criterion for labeled data. Gaussian mixture models of continuous spectral feature vectors were trained for phonetic classes and extended with HMM for transition probabilities. Auxiliary functions were used for objective maximization. It was found that more unlabeled data resulted in better results for the hybrid model. The hybrid objective function combined the discriminative labeled and the transductive unlabeled instances in a very good manner. The unified objective function was better than most self-training methods in terms of convergence.

E. Cluster and label Approach

Many machine learning algorithms perform differently when trained on different domains. This is due to many reasons such as different target distribution, data variables, etc. Many real-world applications may show performance lapses because of this. In order to avoid such shortcomings in conventional learning methods we use an approach known as cluster and label or cluster then label. This approach first finds clusters in a high-density space of the dataset which are then assigned labels. The learner

may then use a plane or line on the remaining low-density clusters for learning or separating purpose.

A cluster then label approach was proposed by [18] Peikari et. al. for pathology image classification. The data used in the paper consisted of both labeled and unlabeled points in a multidimensional feature space. The semi supervised learning method was supposed to identify high- and low-density cluster regions and use them for further boundary separation. The proposed model worked in a number of steps. The step was the identification of the high-density cluster regions. Then from the knowledge of these points and their structures, a supervised Support Vector Machine (SVM) was used to find the decision boundary. The SVM used Radial Basis Function kernel. Next the points were ordered in terms of their spatial distances and inclination of the unlabeled points from the labeled ones. The paper then applied the Semi-Supervised Seeded Density Based (S3DB) clustering approach in order to preserve the smoothness and cluster assumptions. 8-Fold Cross Validation technique was used for training. The method was found to be superior to traditional methods in terms of lower train time and accuracy.

Another improvement in cluster and label approach was suggested by Azab et. al., in [19] were particle swarm optimization (PSO) was used. Particle Swarm Optimization is an iterative computational optimization method which improves a candidate solution with regard to quality. The proposed model used a predefined number of clusters k where the neighborhood of each cluster attributed one of the clusters. Each neighborhood of the cluster was used to optimize its centroid. The clusters followed the information of the labeled data and a certain silhouette score

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (1)$$

where $b(i)$ is the average dissimilarity and $a(i)$ is the average similarity between objects. The results obtained in this method showed improved performance in the cluster and label approach but only for a limited size of data.

Apart from valued data, the cluster and label approach also find its use in speech related applications as proposed by Albalate et. al., in [20]. It was used for utterance classification specifically for troubleshooting dialog systems. The proposed model for cluster and label involved two separate tasks. Firstly, the data set was clustered without any considering to any prior approach or whether labeled or unlabeled. Then a Support Vector Machine (SVM) is applied on the labeled data which is further enlarged using an optimal cluster labeling technique. Optimal Cluster labeling was achieved using Hungarian Algorithm. For this the authors used the concept of pattern silhouette. To avoid miss classification errors a further optimization was achieved through cluster pruning. The approach showed accuracy results to increase in proportion to the data size.

F. S3SVM

Support Vector Machines (SVMs) are supervised learning model that make use of an optimal hyper-plane

for classification of labeled data. We can modify supervised SVMs to Semi Supervised Support Vector Machines (SSSVM OR S3VM) by using labeled data in a Hilbert space by extending vector algebra tools and calculus from a two-dimensional Euclidean plane and three-dimensional space to spaces with a myriad number of dimensions. The separating decision hypothesis depends on the unlabeled data.

S3VM were first introduced by T. Joachims for transductive inference for text classification using SVMs in [21]. Before we get into S3VM we must be clear regarding transductive and Inductive learning. In inductive learning models produce labels for unlabeled data with the help of classifier whereas transductive learning models produce labels for unlabeled data without the help of a classifier. Transductive SVMs or S3SVMs were needed to improve the already high accuracy of SVMs for less labeled or training data. A new algorithm for training a very large dataset was also proposed. Transductive SVM was selected because of its high dimensional input space, sparse document vectors and aggressive feature selection. Precision Recall-Breakeven point method was used for performance measure. Results showed that Transductive SVM performed much better than SVM and Naive Bayes. There were however a few short comings of the Transductive SVM, the number of positive labels had to be specified which is difficult to estimate.

In order to take care of presetting the number of labels reasonably to avoid an unstable model, Xu Yu, Jing Yang and Jian-pei Zhang [22] proposed a spectral clustering approach. The approach was called TSVMSC (Transductive Support Vector Machine based on Spectral Clustering). It was based on finding the solution of the SVM optimization problem which is as follows:

There were four steps proposed. The first step involved specifying the parameters C , C^* and clustering number k generally between 3 and 7. The second step involved applying the spectral clustering algorithm. The third step involved marking all clusters and solving the optimization problem. The fourth and final step was outputting the labels of unlabeled samples.

$$\min_{\omega, b, \varepsilon_i, \varepsilon_j} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n \varepsilon_i + C^* \sum_{j=1}^k \varepsilon_j$$

such that $y_i(\omega(x_i) + b) + \varepsilon_i - 1 \geq 0, i = 1, 2, \dots, n$
 $\varepsilon_i \geq 0$

$$y_j(\omega(x_j) + b) + \varepsilon_j - 1 \geq 0, j = 1, 2, \dots, k \text{ & } \varepsilon_j \geq 0$$

In conclusion the results showed that spectral clustering in TSVM was a good approach to achieve stability.

Another method for improving the task of presetting the number of labels was proposed by Yu-Feng Li and Zhi-Hua Zhou in [23]. This paper suggested using hierarchical clustering for choosing the unlabeled data points in a greedy iterative manner. It was called S3VM- us. The algorithm involves four steps. The first step involves

performing hierarchical clustering. The second step involves calculating path lengths according to the linkage method or metric. The third step involved defining a set based on a threshold function. The fourth step involved choosing SVM or SVM-us. The fifth and final step involved labeling unlabeled instances. One advantage of this method is that hierarchical clustering does not suffer from the label initialization problem. Hold-out tests were used for evaluation. Mixed results were obtained when TSVM and S3VM-us were tested. When average accuracy was considered TSVM performed a little better than S3VM-us, but when performance degradation is considered S3VM is the best method to avoid degradation in terms of performance.

G. Generative methods

Sometimes data instances occur in pairs of two parameters with their corresponding labels. In order to perform classification of such generative types of data we use specialized methods known as generative methods. It is a kind of soft clustering technique. They generally apply the Expectation Maximization algorithm. This method however suffers in real world data sets where labeled data is noisy. Another drawback is the presence of some inherent Bias.

In order to take of the noisy data Langevin et. al., in [24] proposed a model called mislabeled VAE or M-VAE (Variational Auto Encode) where additional low dimensional latent variables were used in the generative process. These were further used to approximate posterior distribution which reduced noisy instances. It was concluded that it outperformed standard generative methods and also did not suffer from imbalance.

The generic generative method suffers from bias which was improved by Fujino et. al., in [25]. The author proposed a hybrid discriminative approach with bias correction methods. The generative model made use of EM along with some additional classes. The bias correction model made use of MAP (Maximum A Posteriori) estimation on the training samples. The proposed learning model used Naive Bayes algorithm. Experiments were carried out to estimate accuracy on various datasets of varying sizes. It was observed that the Naive Bayes based model did not do well where size of dataset was small. The method had the potential to perform better than standard generative models.

IV. ANALYSIS OF SEMI SUPERVISED LEARNING TECHNIQUES

The following table provides a brief comparison of the various semi supervised learning techniques w.r.t to their advantages, disadvantages and applications.

TABLE I. SUMMARY OF SEMI SUPERVISED LEARNING TECHNIQUES

SSL	Advantages	Disadvantages	Applications
Self-Training	<ul style="list-style-type: none"> Implementation is simple and easy. Faster Computation. 	<ul style="list-style-type: none"> Errors or noise get reinforced Sensitive to imbalanced data Predefined terms difficult to estimate 	<ul style="list-style-type: none"> Chest X-Ray Image classification Text Classification
SETRED - SELf TRaining with Editing	<ul style="list-style-type: none"> Strong Noise resistance. Can identify miss classification errors. Better results than standard self-learning. 	<ul style="list-style-type: none"> Sensitive to imbalanced data Error convergence can occur 	<ul style="list-style-type: none"> Hepatitis data classification [8] Wine classifier [8]
AAST - Auto Adjustable Self Training	<ul style="list-style-type: none"> Best Self Training method. All major drawbacks in standard self-training methods are taken care. 	<ul style="list-style-type: none"> Computation cost proportion to k. 	<ul style="list-style-type: none"> SONAR [9] Mushroom Classification [9]
Co-training	<ul style="list-style-type: none"> Better confidence values. 	<ul style="list-style-type: none"> Both Assumptions need to be satisfied. Unsuitable for real world scenario. Views need to be carefully chosen. 	<ul style="list-style-type: none"> Image Segmentation.
Co-training with PAC Framework	<ul style="list-style-type: none"> High potential for improvement. 	<ul style="list-style-type: none"> Method preliminary in nature. Requires further study. Views need to be carefully chosen. 	<ul style="list-style-type: none"> Image Segmentation [10]
COTRADE-Confident Co-training with Editing	<ul style="list-style-type: none"> Reliable view communication. Best training procedure among other given methods. 	<ul style="list-style-type: none"> Performance lacks for some datasets. 	<ul style="list-style-type: none"> Advertisement Image Filtering [11] Web-page classification [11]
Graph Based SSL	<ul style="list-style-type: none"> Works very well for correct assumptions. 	<ul style="list-style-type: none"> A high number of parameters result in zero cross validation error. 	<ul style="list-style-type: none"> Person identification [12]
Cluster then label	<ul style="list-style-type: none"> Lower training time. High accuracy. 	<ul style="list-style-type: none"> A high number of parameters result in zero cross validation error. Miss classification errors exist. 	<ul style="list-style-type: none"> Pathology Image Classification [18]
Cluster then label PSO [19]	<ul style="list-style-type: none"> Uses best fitness function. 	<ul style="list-style-type: none"> Shows improvement but only for a limited size of data. 	<ul style="list-style-type: none"> Computer Aided Diagnosis (CAD) [19]
Cluster then label Hungarian [20]	<ul style="list-style-type: none"> Miss classification error avoided Accuracy proportional to size. 	<ul style="list-style-type: none"> Polynomial Time complexity. 	<ul style="list-style-type: none"> Utterance Classification [20]
GMM	<ul style="list-style-type: none"> Identifies hidden relationships. Better cluster shapes than k-means. 	<ul style="list-style-type: none"> Optimizing the loss function is difficult. Initialization method sensitive. 	<ul style="list-style-type: none"> Pathology Image Classification [18]
GMM-Online Learning	<ul style="list-style-type: none"> Robust learning technique. 	<ul style="list-style-type: none"> Shows improvement but only for a limited size of data. 	<ul style="list-style-type: none"> Channel Based authentication. [17]
GMM- Hybrid Discriminative Method [26]	<ul style="list-style-type: none"> Better objective function. 	<ul style="list-style-type: none"> Continued training hardly results in any further information exchange 	<ul style="list-style-type: none"> Phonetic Classification
S3VM	<ul style="list-style-type: none"> High Dimensional Input Space. Sparse Document vectors. Aggressive feature selection. 	<ul style="list-style-type: none"> Label initialization problem. Initialization method sensitive. 	<ul style="list-style-type: none"> Text classification [21]
TSVMSC [22]	<ul style="list-style-type: none"> Spectral Clustering improves stability. No label initialization problem. 	<ul style="list-style-type: none"> Lacks rational cluster calculation approach 	<ul style="list-style-type: none"> Text classification [22]
S3VM-us [23]	<ul style="list-style-type: none"> Does not suffer from label initialization problem. 	<ul style="list-style-type: none"> Continued training hardly results in any further information exchange 	<ul style="list-style-type: none"> Optical Recognition of Handwritten Digits. [23]
Generative Method- EM	<ul style="list-style-type: none"> Works for paired input data Works for large dataset 	<ul style="list-style-type: none"> Suffers from noisy data. Performance hampered due to bias 	<ul style="list-style-type: none"> Text classification
Generative Method- M-VAE [24]	<ul style="list-style-type: none"> Does not suffer from noise. Does not suffer from data imbalance. 	<ul style="list-style-type: none"> Classification penalty overhead 	<ul style="list-style-type: none"> RNA Classification [24]
Generative Method- Discriminative [25]	<ul style="list-style-type: none"> Bias Correction. Performs better for large data. 	<ul style="list-style-type: none"> Poor performance for small datasets Further study needed 	<ul style="list-style-type: none"> Text classification [25]

V. CONCLUSION

This paper provides a detailed description of the various available and known semi supervised learning techniques. The paper provides an analysis of the different generic semi supervised learning techniques, their underlying assumptions and parameters, requirements and the diverse improvements that have been suggested over the years. Each technique has its own set of advantages and disadvantages. The table provides a detailed analysis of this in terms of their performance and applications. This paper is a sincere effort to analyze the various semi-supervised learning techniques and the areas where these can be suitably applied.

REFERENCES

- [1] O. Chapelle, B. Scholkopf and Z. Alexander, *Semi-Supervised Learning*, Massachusetts London, England: The MIT Press Cambridge, 2006.
- [2] P. Rai, *Semi-supervised Learning*, 2011.
- [3] X. Zhu, "Semi-supervised learning literature survey," Madison, 2008.
- [4] C. C. Vincent Ng, *Weakly Supervised Natural Language Learning Without Redundant Views*, Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, 2003.
- [5] M. H. a. H. S. C. Rosenberg, *Semi-supervised self-training of object*

detection models, Proceedings of the 7th IEEE Workshop on Applications of Computer Vision (WACV'05), 2005.

[6] X. Z. a. A.-d. B. Goldberg, Introduction to Semi-Supervised Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, 2009.

[7] A. S. a. D. A. Jivani, A SURVEY OF SEMI-SUPERVISED LEARNING, zenodo.159333, 2016.

[8] &. Z.-H. Z. Min-Ling Zhang, CoTrade: Confident Co-Training With Data Editing, EEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 41(6), 16121626.doi:10.1109/tsmcb.2011.2157998, 2011.

[9] A. K. ,. V. T. a. P. Ioannis E. Livieris, An Auto-Adjustable Semi-Supervised Self-Training Algorithm, 2018.

[10] A. &. M. T. Blum, Combining labeled and unlabeled data with co-training, Proceedings of the Eleventh Annual Conference on Computational Learning Theory - COLT 98, 1998.

[11] J. B. a. Subramanya, Parallel Graph-Based Semi-Supervised Learning, 2011.

[12] A. a. C. S. Blum, Learning from Labeled and Unlabeled Data Using Graph Mincuts, Pages 1926 of: Proc. 18th International Conf. on Machine Learning. Morgan Kaufmann, San Francisco, CA, 2001.

[13] X. G. Z. a. L. J. Zhu, Semi-supervised learning using Gaussian fields and harmonic function, Proc. of the International Conference on Machine Learning (ICML), 2003.

[14] M. a. J. T. Szummer, Partially labeled classification with Markov random walks, Advances in Neural Information Processing Systems, vol. 14, 2001.

[15] G. D. C. I. R. Nara M. Portela, Semi-supervised clustering for MR brain image segmentation, Expert Systems with Applications, 2014.

[16] R. G. K. R. a. J. M. W. Nikhil Gulati, GMM based Semi-Supervised Learning for Channel-based Authentication Scheme, Vehicular Technology Conference, 1988, IEEE 38th, 2013.

[17] J.-T. H. a. M. Hasegawa-Johnson, On Semi-Supervised Learning of Gaussian Mixture Models, Volume: Proceedings of the NAACL HLT 2009 Workshop on Semi-supervised Learning for Natural Language Processing, 2009.

[18] S. S. S. N.-M. & A. L. M. Mohammad Peikari, A Cluster-then-label Semi-supervised Learning Approach for Pathology Image Classification, 2018.

[19] M. F. A. H. H. A. H. Shahira Shaaban Azab, Semi-supervised Classification: Cluster and Label Approach using Particle Swarm Optimization, ArXiv 2017 DOI:10.5120/ijca2017913013 , 2017.

[20] A. S. D. S. a. W. M. Amparo Albalate, A semi-supervised cluster-and-label approach for utterance classification, Volume 8: Workshop Proceedings of the 6th International Conference on Intelligent Environments, 2010.

[21] T. Joachim, Transductive inference for text classification using support vector machines, ICML, pages 200–209, 1999.

[22] J. Y. a. J.-p. Z. Xu Yu, A Transductive Support Vector Machine Algorithm Based on Spectral Clustering, 2012 AASRI Conference on Computational Intelligence and Bioinformatics, 2012.

[23] Z.-H. Z. Yu-Feng Li, Improving Semi-Supervised Support Vector Machines Through Unlabeled Instances Selection, 2011.

[24] E. M. J. R. R. L. M. I. J. N. Y. Maxime Langevin, A Deep Generative Model for Semi-Supervised Classification with Noisy Labels, 2018.

[25] N. U. a. K. S. Akinori Fujino, A Hybrid Generative/Discriminative Approach to Semi-supervised Classifier Design, 2005.

[26] M. L. a. Z.-H. Zhou, SETRED: self-training with editing, Berlin, Germany: Advances in Knowledge Discovery and Data Mining: 9th Pacific-Asia Conference, PAKDD 2005, Hanoi, Vietnam, May 18-20, 2005. Proceedings, vol. 3518 of Lecture Notes in Computer Science, pp. 611-621, Springer, 2005.

[27] J.-T. Huang and M. Hasegawa-Johnson.