【主論文の要旨】

学位報告４

報告番号	※甲　　　第　　　号

主　論　文　の　要　旨

論文題目

Study of Prediction on Manifolds with Almost No or No Labeled Data
(ラベル情報が極端に少ない、もしくは無い状況下における多様体を利用したラベル予測に関する研究)

氏　　名　　　和田　裕一郎

論 文 内 容 の 要 旨

<Problem Settings>

In this thesis, we investigate the following two fields: ①*Online Graph-Based Semi-Supervised Learning* (SSL) and ②*Deep Clustering*. In the former field, given a small size labeled dataset, the learning algorithm handles a continuous streamed unlabeled data points. The challenge is to predict the label of newly arrival data point quickly and precisely under sever memory constraints. Note that the distribution of streamed data may change as time goes. In the later field, given a large size unlabeled dataset and the number of clusters, the clustering algorithm estimates the cluster labels. A deep neural network is used to define the statistical model. The unlabeled data points are supposed to be generated from the same distribution independently. The challenge of this clustering is to estimate the cluster labels of given unlabeled dataset precisely as possible as we can.

<Previous Methods>

① online Graph-Based SSL

Online graph-based SSL is a relatively new filed of SSL studies. Although many online SSL algorithms have been proposed recently, most of them do not consider the processing time and severe memory constraints. Consequently, runtime and memory demands are increasing functions $\Omega(T)$ of streaming size T. On the other hand, an example of a few studies that account for processing time and memory constraints is online Quantized Label Propagation (QLP). This method is at each time, firstly to recompress the data adjacency graph by incorporating a newly arrived data point, then secondly to predict the label of new data point on the graph. The Doubling Algorithm (DA) and Label Propagation (LP) are employed as the graph compressing and label predicting methods, respectively. The other previous methods also take the same strategy, and both of them employ LP as the label predicting method. However, LP is known to not be robust against outliers. The reason is that LP predicts the labels by using all given data points, which often include outliers. In addition, LP is known to not be efficient computationally. As a possible alternate of LP, we can list Geodesic k-Nearest Neighbor (GkNN) algorithm. This method is not only computationally more efficient but also more robust against outliers than LP. The drawback of GkNN is that it does not perform well when the size of given labeled data is small.

② Deep Clustering

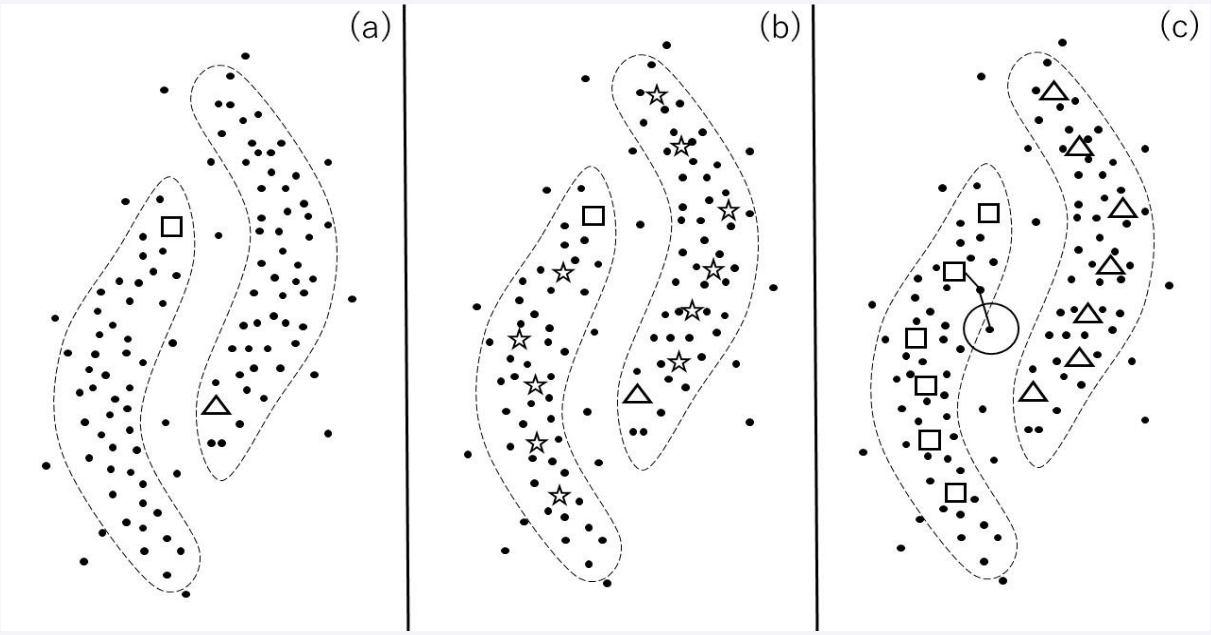
Thanks to the development of deep neural networks, we can now handle large datasets with complicated shapes. Consequently, the studies of clustering using deep neural networks has been proposed. One major direction in the studies is to combine deep AutoEncoders (AE) with classical clustering methods such as k-means. This AE is used to obtain a clustering friendly low dimensional representation. In another major direction, the methods directly group a given unlabeled dataset into the given number clusters in the original input space by employing a deep neural network to their statistical models. Though most of deep clustering methods are built on the following fundamental two assumptions: the smoothness and manifolds assumptions, these methods require additional key conditions where the methods perform well. For an example, CatGAN (Categorical Generative Adversarial Networks) learns discriminative neural network classifiers that maximize mutual information between the input data points and the cluster labels, while enforcing the robustness of the classifiers to data points produced by adversarial generative models. Since maximizing mutual information implicitly encourages the cluster-balance distribution of the model to be uniform, if the distribution of cluster-balance with the given unlabeled dataset is not uniform, then CatGAN will not perform well. To the best of our knowledge, like the example, most of their key conditions are not realistic. One of few examples is SpectralNet, whose clustering logic is based on the above two fundamental assumptions. As for the weakness of SpectralNet, the performance is not robust against the existence of outliers. In the learning process, it learns the pairwise similarities over all given data points. Therefore, the existence of outliers disturbs the method to learn the similarities precisely, and thus it returns inaccurate cluster labels.

<Our Research Ambitions, Proposed Methods and the Numerical Experiments>

① Online Graph-Based SSL

Our ambitions are, firstly, to invent a new label predicting offline SSL algorithm which can assist in creating more competitive online graph-based SSL algorithms. Our requirement toward the offline method is that the computational complexity should be as small as GkNN, and the predicting performance should be better than that of LP and GkNN. With the second ambition, after the invention of the offline SSL method, by combining the offline method and a conventional online clustering method, we then propose an online SSL method. As the result we could achieve the ambitions. Our proposed offline and online SSL methods are named *Robust Label Prediction* (RLP) and *online Quantized RLP* (online QRLP), respectively. The details of both methods are as follows.

[RLP]: RLP algorithm consists of three steps. On the basis of the neighbor graph, RLP first selects some unlabeled samples that represent the global structure of the data manifolds. The second step assigns labels to selected unlabeled samples by using LP. The third step predicts the labels on the remaining unlabeled samples by using GkNN. The unlabeled samples selected by the algorithm are collected into the *hub* dataset, which is denoted as H. The vertices selected for H are those with many neighbors on the data affinity graph G. In the following figure, we explain the detail of RLP.



This figure explains the working mechanism of the RLP algorithm.

(a): The labeled and unlabeled data (black square/triangle and black dots, respectively) are given in two dimensional space. The number of classes is two. The two banana shapes delineate the true data manifolds. By using both labeled and unlabeled data points, we construct data affinity graph G. The nodes correspond those data points. The set of edge is defined by k-Nearest Neighbor manner with Euclidean distance. On each edge, the similarity is defined by Gaussian kernel.

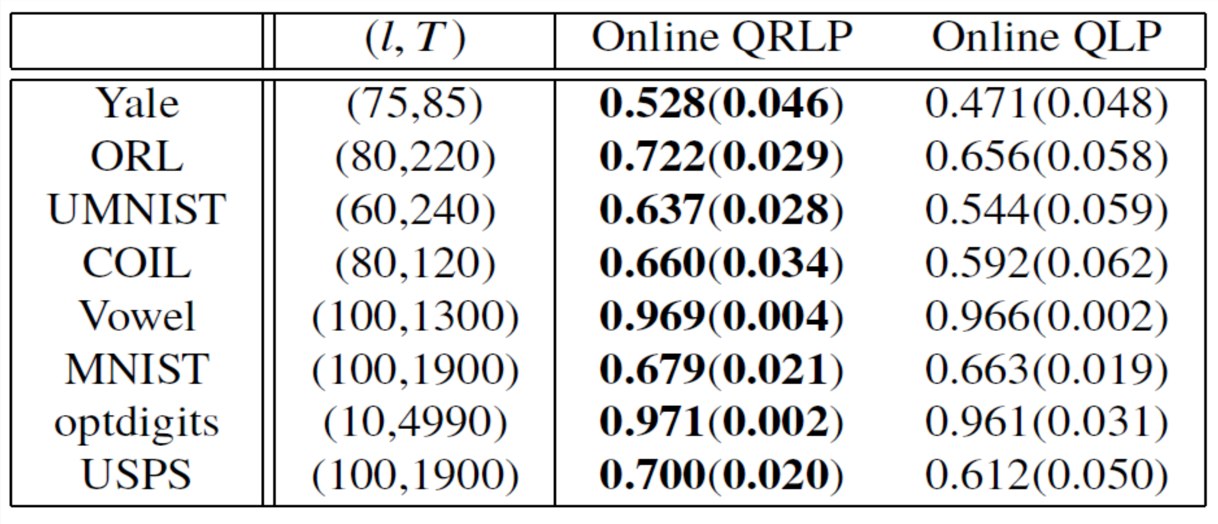
(b): After the graph construction, we define the hub set H. The elements are selected based on the number of degree. Top h highest degree nodes can belong the set. We consider h as the hyperparameter. In this case, the star symbols and the black dots denote the hub data and non-hub data points, respectively, where $h = 11$.

(c): The hub data points are assigned labels by modified LP and are considered new labeled data points. The labels of remaining unlabeled data points are predicted by GkNN (k=1). Note that the geodesic distance is approximated by the graph shortest path on G. As an example of label prediction by GkNN, the circled black dot is labeled with the square symbol indicated in the left banana shape.

[Online QRLP]: This method is obtained by combining DA and Quantized RLP, in which we conduct RLP on the compressed data affinity graph. DA is commonly used online clustering algorithm. In DA, given number of centroids, at each time, the set of centroids is updated.

The set can be seen the set of important data points in the original data affinity graph defined up until the time. With regarding the hyperparameter tuning of online QRLP, we can tune them if we could have small size labeled and unlabeled datasets before the stream. The computational cost of this method can be controlled by upper-bounding the number of hub data points to match that of GkNN.

[Numerical Experiments with online QRLP]



	(l, T)	Online QRLP	Online QLP
Yale	(75,85)	**0.528(0.046)**	0.471(0.048)
ORL	(80,220)	**0.722(0.029)**	0.656(0.058)
UMNIST	(60,240)	**0.637(0.028)**	0.544(0.059)
COIL	(80,120)	**0.660(0.034)**	0.592(0.062)
Vowel	(100,1300)	**0.969(0.004)**	0.966(0.002)
MNIST	(100,1900)	**0.679(0.021)**	0.663(0.019)
optdigits	(10,4990)	**0.971(0.002)**	0.961(0.031)
USPS	(100,1900)	**0.700(0.020)**	0.612(0.050)

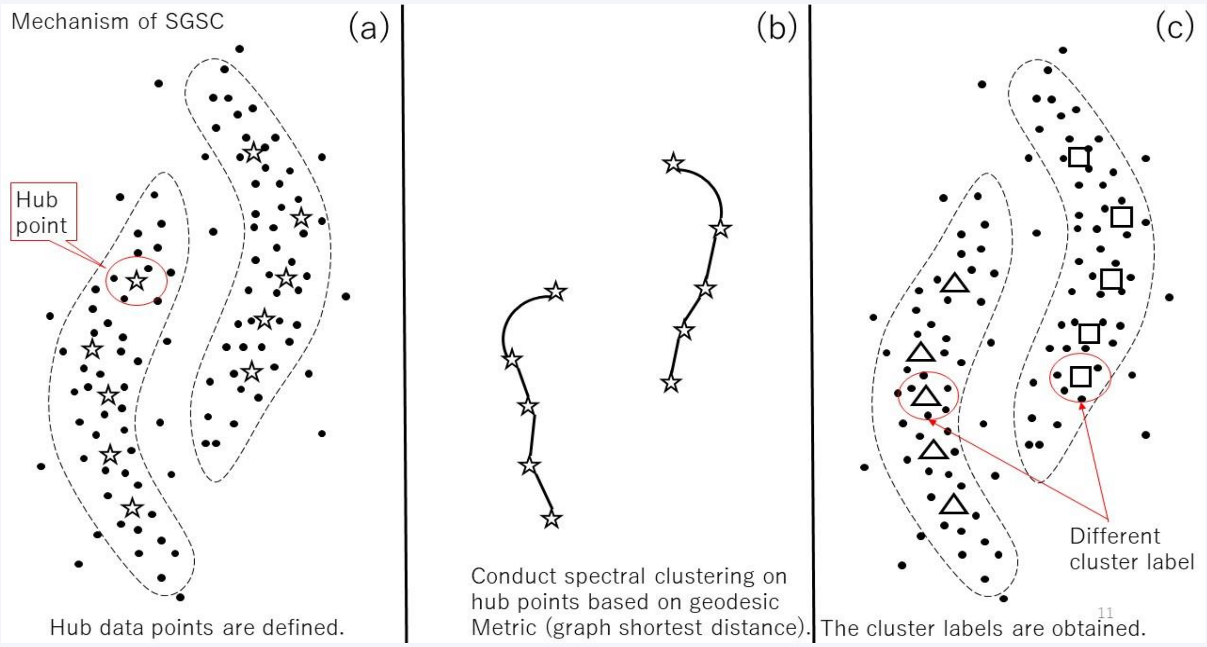
This table shows the averaged prediction accuracy with the standard deviation on unlabeled data in eight real-world data streams. In each dataset, *l* denotes the number of labeled data obtained before the arrival of each stream of size *T*. As you can see, our method outperforms the previous popular online SSL method based on LP.

② Deep Clustering

Our ambition is to invent outlier-robust deep clustering which is built only on the two fundamental assumptions. As mentioned before, the previous methods require the additional key conditions apart from the two assumptions. Therefore, for the unknown unlabeled dataset, their performances are not promised. If we can invent such the method, it can be a good candidate in practical situation. As the result we could achieve the ambitions. The proposed method is named *Spectral Embedded Deep Clustering* (SEDC). Given an unlabeled dataset and the number of clusters, SEDC directly groups the dataset into the given number clusters in the input space. Our statistical model is the conditional discrete probability distribution, which is defined by a fully connected deep neural network. SEDC does not require key condition except the smoothness and manifold assumptions, and it can be applied to various data domains. Moreover,
throughout our numerical experiments, we observed that our method was more robust against outliers than SpectralNet. The procedure of SEDC is composed of two stages. In the first stage, we conduct Spectral Clustering (SC) only on the unlabeled data points selected from high density region by using the geodesic metric to estimate the cluster labels. This selected data points exactly equal to hub data points, which was defined in online SSL study. This special type of SC is named as *Selective Geodesic Spectral*

Clustering (SGSC), which we propose for assisting SEDC as well. Thereafter, we conduct SSL to train the model by using the estimated cluster labels and the remaining unlabeled data points. Note that, in this SSL, we treat the estimated cluster labels of the selected unlabeled data points as the given true cluster labels. At last, by using the trained model, we obtain the estimated cluster labels of all given unlabeled data points. In the following, firstly, let us explain the detail of SGSC, then SEDC.

[SGSC]: The motivation behind SGSC is to assist the semi-supervised learning in SEDC. SGSC conducts SC only on hub (selected unlabeled) data points with the geodesic metric, then returns the estimated cluster labels of hub points. The hub points are defined as data points in high density region, and these data points are approximated by the highest degree nodes on the affinity data graph. The geodesic metric is approximated by the graph shortest path distances on the graph. Empirically speaking, the estimation accuracy of cluster labels with hub points tends to not only be robust against the existence of outliers but also be competitive. This tendency can help SEDC return competitive clustering result. The reason of robustness is that the selection of data points from high density region tends to not be affected by the existence of outliers. The reason to employ the geodesic metric is that the metric is known to be useful to capture the structure of the data manifolds especially when the number of given data points is large. The below figure is image with mechanism of SGSC.


Mechanism of SGSC
(a)
Hub point
Hub data points are defined.
(b)
Conduct spectral clustering on hub points based on geodesic Metric (graph shortest distance).
(c)
Different cluster label
The cluster labels are obtained.
11

(a) : Given unlabeled data points, SGSC computes the hub data points on data affinity graph, which is constructed by k-NN manner with Euclidean distance. The hub data points are expressed by star symbols, and the number of hub data points is ten in this case.

(b) : SGSC focuses only on the hub data points, then conducts SC with the geodesic metric on those hub points, where we set one to the number of neighbor.

(c) : As the results, we obtain the cluster labels of hub points. The triangle and square symbols mean different labels. Note that an actual output of SGSC is the estimated conditional discrete probability distributions with hub data points, but it is no problem for us to obtain the estimated cluster labels from the distributions.

[SEDC]: Given an unlabeled dataset X and the number of clusters, SEDC optimizes the following objective function to train the statistical model, which is the conditional discrete probability distribution parameterized by a fully connected deep neural network:

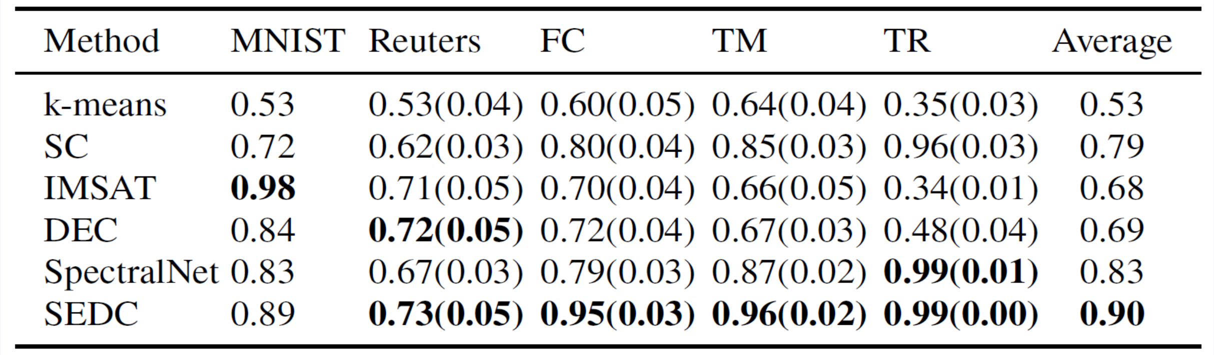
$$\min_{\theta}\left\{R_{VAT}(\theta) + CH\left(\left\{x_{(i)},\hat{p}_{(i)}\right\}_{i=1}^{h},\theta\right) + H(Y|X)\right\} ,$$

where the first, second and third loss are the Virtual Adversarial Training (VAT) loss, pseudo empirical loss and the conditional Shannon entropy loss, respectively. With the definitions of symbols in the objective, x, θ and h mean the element of unlabeled dataset, the parameters in the deep neural network and the number of hub data points. The

$\hat{p}_{(i)}$ means the estimated distribution of cluster labels with hub data point indexed by (i).

This estimated distribution is a part of outputs of SGSC. The minimization of VAT loss imposes the model to follow the smoothness assumption. The minimization of the third term imposes the model to have the large margin between the clusters. After this optimization, we obtain the estimated cluster labels of all given unlabeled data points, which are computed by the trained statistical model.

[Numerical Experiments with SEDC]



Method	MNIST	Reuters	FC	TM	TR	Average
k-means	0.53	0.53(0.04)	0.60(0.05)	0.64(0.04)	0.35(0.03)	0.53
SC	0.72	0.62(0.03)	0.80(0.04)	0.85(0.03)	0.96(0.03)	0.79
IMSAT	**0.98**	0.71(0.05)	0.70(0.04)	0.66(0.05)	0.34(0.01)	0.68
DEC	0.84	**0.72(0.05)**	0.72(0.04)	0.67(0.03)	0.48(0.04)	0.69
SpectralNet	0.83	0.67(0.03)	0.79(0.03)	0.87(0.02)	**0.99(0.01)**	0.83
SEDC	0.89	**0.73(0.05)**	**0.95(0.03)**	**0.96(0.02)**	**0.99(0.00)**	**0.90**

This table shows the averaged estimation accuracy of cluster labels with all given unlabeled data points. Inside of () shows the standard deviation. Seven times experiments are conducted for the average and the standard deviations. k-means to SEDC are the name of clustering methods. MNIST to TR are the name of datasets. Our proposed method is SEDC. As you can see, our method averagely outperforms the other clustering methods. As for the MNIST dataset, IMSAT performed pretty well since the dataset satisfy the key condition.

<Conclusions and Future Works>

① Online Graph-Based SSL

We proposed a generic graph-based SSL algorithm, called RLP.

We confirmed that RLP is robust against noisy data and provides more accurate predictions than LP and GkNN. The computational efficiency of RLP matches that of GkNN.

Furthermore, we confirmed the power of RLP as a core technique in the online SSL framework. In the online scenario, the proposed method has two tunable hyperparameters, namely then number of neighbor and hub data points. Future works should focus on the choice of number of hub data points. In this thesis, the upper bound of number of hub points was determined by considering the computational cost. The prediction accuracy when the number is based on other criteria should also be examined. Furthermore, an adaptive method that determines both hyperparameters would be useful for practical online learning.

②Deep Clustering

In this thesis, we propose a deep clustering method named SEDC. Given an unlabeled dataset and the number of clusters, the method groups the dataset into the given number clusters. Regarding its advantages, it does not require an additional key condition except two fundamental assumptions: smoothness and manifolds assumptions. In this point, only SpectralNet is comparable. In addition, SEDC also can be applied to various data domains since it does not have preferred data domains, as long as raw data is transformed to feature vectors. Furthermore, the performance of SEDC can be robust against existence of outliers unlike SpectralNet. According to these advantages, our proposed method can be expected to averagely perform better than previous deep clustering methods. As a result, this expectation is empirically confirmed by conducting numerical experiments on five commonly used datasets: see [Numerical Experiments with SEDC]. Therefore, we think our method can be a competitive candidate for users in some practical clustering scenarios where prior knowledge of the given unlabeled dataset is limited. Let us then discuss two limitations of SEDC. On the one hand, since the method needs hyperparameter tuning, if we do not have appropriate labeled source domains to learn them from and transfer, then it may fail. On the other hand, since the method requires the number of clusters, it does not work for datasets where nothing is known on the number of clusters such as genome datasets. Finally, we discuss about our two future works. The first one is to invent a more noise-robust semi-supervised learning framework and then apply it to SEDC instead of the above objective function. Since some of the estimated cluster labels by SGSC are not perfectly accurate, we need to invent such the framework to stabilize the performance of SEDC.
The second one is to modify our method for handling structured data, i.e., graph data or sequential data.