Int. J. Advance Soft Compu. Appl, Vol. 9, No. 1, March 2017
ISSN 2074-8523

The Un-normalized Graph p-Laplacian based Semi-supervised Learning Method and Speech Recognition Problem

Loc Tran[1] and Linh Tran[1]

[1]University of New Brunswick
Ho Chi Minh City University of Technology-VNU HCM
tran0398@umn.edu,linht@pdx.edu

Abstract

Speech recognition is the classical problem in pattern recognition research field. However, just a few graph based machine learning methods have been applied to this classical problem. In this paper, we propose the un-normalized graph p-Laplacian semi-supervised learning methods and these methods will be applied to the speech network constructed from the MFCC speech dataset to predict the labels of all speech samples in the speech network. These methods are based on the assumption that the labels of two adjacent speech samples in the network are likely to be the same. The experiments show that that the un-normalized graph p-Laplacian semi-supervised learning methods are at least as good as the current state of the art method (the un-normalized graph Laplacian based semi-supervised learning method) but often lead to better classification sensitivity performance measures.

Keywords: *graph p-Laplacian, MFCC, semi-supervised learning, speech recognition*

1 Introduction

Researchers have worked in automatic speech recognition for almost six decades. The earliest attempts were made in the 1950's. In the 1980's, speech recognition research was characterized by a shift in technology from template-based approaches to statistical modeling methods, especially Hidden Markov Models (HMM). Hidden Markov Models (HMM) have been the core of most speech recognition systems for over a decade and is considered the current state of the art method for automatic speech recognition system [1]. Second, to classify the speech samples, a graph (i.e. kernel) which is the natural model of relationship between speech samples can also be employed. In this model, the nodes represent speech samples. The edges represent for the possible interactions between nodes. Then, machine learning methods such as Support Vector Machine [2], Artificial Neural Networks [3], nearest-neighbor classifiers [4], or graph based semi-supervised learning methods [5] can be applied to this graph to classify the speech samples. The nearest-neighbor classifiers method labels the speech sample with the label that occurs frequently in the speech sample's adjacent nodes in the network. Hence neighbor counting method does not utilize the full topology of the network. However, the Artificial Neural Networks, Support Vector Machine, and graph based semi-supervised learning methods utilize the full topology of the network.

While nearest-neighbor classifiers method, the Artificial Neural Networks, and the graph based semi-supervised learning methods are all based on the assumption that the labels of two adjacent speech samples in graph are likely to be the same, SVM does not rely on this assumption. Graphs used in nearest-neighbor classifiers method, Artificial Neural Networks, and the graph based semi-supervised learning method are very sparse. However, the graph (i.e. kernel) used in SVM is fully-connected.

The un-normalized, symmetric normalized, and random walk graph Laplacian based semi-supervised learning methods are developed based on the assumption that the labels of two adjacent speech samples in the network are likely to be the same [5,6,7,16]. However, assuming the pairwise relationship between speech samples is not complete, the information a group of speech samples that show very similar patterns and tend to have similar labels is missed [8,9,10]. The natural way overcoming the information loss of the above assumption is to represent the data representing the set of speech samples as the hypergraph [8,9,10]. A hypergraph is a graph in which an edge (i.e. a hyper-edge) can connect more than two vertices. In [8,9,10], the symmetric normalized hypergraph Laplacian based semi-supervised learning method have been developed and successfully applied to text categorization, letter recognition, and protein function prediction applications. To the best of our knowledge, in [11], the hypergraph Laplacian based semi-supervised learning methods have been successfully applied to speech recognition problem.

In [12,13], the symmetric normalized graph p-Laplacian based semi-supervised learning method has been developed but has not been applied to any practical applications. To the best of my knowledge, the un-normalized graph p-Laplacian based semi-supervised learning method has been successfully developed and applied to bio-informatics problems such as protein function prediction problem and cancer classification problem [14,15]. Specifically, in this paper, we will try to apply the un-normalized graph p-Laplacian based semi-supervised learning method to speech recognition problem (i.e. the classification phase). This application is worth investigated because of two main reasons. First, the un-normalized graph p-Laplacian based semi-supervised learning method is the generalization of the un-normalized graph Laplacian based semi-supervised learning method (i.e. the current state of the art network based method solving classification problems in bio-informatics research area for *p=2*). Second, the un-normalized graph p-Laplacian based semi-supervised learning method outperforms the current state of the art network based semi-supervised learning method in terms of classification accuracy performance measures [14,15]. The un-normalized graph p-Laplacian based semi-supervised learning method is developed based on the un-normalized graph p-Laplacian operator definition such as the curvature operator of graph (i.e. the un-normalized graph 1-Laplacian operator). Please note that the un-normalized graph p-Laplacian based semi-supervised learning method is developed based on the assumption that the labels of two adjacent speech samples in the network are likely to be the same [5].

We will organize the paper as follows: Section 2 will introduce the preliminary notations and definitions used in this paper. Section 3 will introduce the definition of the gradient and divergence operators of graphs. Section 4 will introduce the definition of Laplace operator of graphs and its properties. Section 5 will introduce the definition of the curvature operator of graphs and its properties. Section 6 will introduce the definition of the p-Laplace operator of graphs and its properties. Section 7 will show how to derive the algorithm of the un-normalized graph p-Laplacian based semi-supervised learning method from regularization framework. In section 8, we will compare the sensitivity performance measures of the un-normalized graph Laplacian based semi-supervised learning algorithm (i.e. the current state of art network-based method applied to speech recognition problem) and the un-normalized graph p-Laplacian based semi-supervised learning algorithms. Section 9 will conclude this paper and the future direction of researches of other practical applications in bioinformatics utilizing discrete operator of graph will be discussed.

2 Preliminary notations and definitions

Given a graph *G=(V,E,W)* where *V* is a set of vertices with $|V| = n$, $E \subseteq V * V$ is a set of edges and *W* is a $n * n$ similarity matrix with elements $w_{ij} > 0\ (1 \le i, j \le n)$.

Also, please note that $w_{ij} = w_{ji}$.

The degree function $d: V \to R^+$ is

$$d_i = \sum_{j \sim i} w_{ij}, \quad (1)$$

where $j \sim i$ is the set of vertices adjacent with *i*.

Define $D = diag(d_1, d_2, \ldots, d_n)$.

The inner product on the function space R^V is

$$< f, g >_V = \sum_{i \in V} f_i g_i \quad (2)$$

Also define an inner product on the space of functions R^E on the edges

$$< F, G >_E = \sum_{(i,j)\in E} F_{ij} G_{ij} \text{ (3)}$$

Here let $H(V) = (R^V, <.,.>_V)$ and $H(E) = (R^E, <.,.>_E)$ be the Hilbert space real-valued functions defined on the vertices of the graph *G* and the Hilbert space of real-valued functions defined in the edges of *G* respectively.

3 Gradient and Divergence Operators

We define the gradient operator $d: H(V) \to H(E)$ to be

$$(df)_{ij} = \sqrt{w_{ij}}(f_j - f_i), \text{ (4)}$$

where $f: V \to R$ be a function of $H(V)$.

We define the divergence operator $div: H(E) \to H(V)$ to be

$$< df, F >_{H(E)} = < f, -divF >_{H(V)}, \text{ (5)}$$

where $f \in H(V), F \in H(E)$

Next, we need to prove that

$$(divF)_j = \sum_{i \sim j} \sqrt{w_{ij}} (F_{ji} - F_{ij})$$

Proof:

$$< df, F > = \sum_{(i,j)\in E} df_{ij} F_{ij}$$
$$= \sum_{(i,j)\in E} \sqrt{w_{ij}} (f_j - f_i) F_{ij}$$
$$= \sum_{(i,j)\in E} \sqrt{w_{ij}} f_j F_{ij} - \sum_{(i,j)\in E} \sqrt{w_{ij}} f_i F_{ij}$$
$$= \sum_{k\in V} \sum_{i\sim k} \sqrt{w_{ik}} f_k F_{ik} - \sum_{k\in V} \sum_{j\sim k} \sqrt{w_{kj}} f_k F_{kj}$$
$$= \sum_{k\in V} f_k (\sum_{i\sim k} \sqrt{w_{ik}} F_{ik} - \sum_{i\sim k} \sqrt{w_{ki}} F_{ki})$$
$$= \sum_{k\in V} f_k \sum_{i\sim k} \sqrt{w_{ik}} (F_{ik} - F_{ki})$$

Thus, we have

$$(divF)_j = \sum_{i\sim j} \sqrt{w_{ij}} (F_{ji} - F_{ij}) \text{ (6)}$$

4 Laplace operator

We define the Laplace operator $\Delta: H(V) \to H(V)$ to be

$$\Delta f = -\frac{1}{2} div(df) \quad (7)$$

Next, we compute

$$(\Delta f)_j = \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}\,((df)_{ij} - (df)_{ji})$$

$$= \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}\,(\sqrt{w_{ij}}(f_j - f_i) - \sqrt{w_{ij}}(f_i - f_j))$$

$$= \sum_{i\sim j} w_{ij}\,(f_j - f_i)$$

$$= \sum_{i\sim j} w_{ij} f_j - \sum_{i\sim j} w_{ij}\, f_i$$

$$= d_j f_j - \sum_{i\sim j} w_{ij}\, f_i$$

Thus, we have

$$(\Delta f)_j = d_j f_j - \sum_{i\sim j} w_{ij}\, f_i \quad (8)$$

The graph Laplacian is a linear operator. Furthermore, the graph Laplacian is self-adjoint and positive semi-definite.

Let $S_2(f) = <\Delta f, f>$, we have the following **theorem 1**

$$D_f S_2 = 2\Delta f \quad (9)$$

The proof of the above theorem can be found from [12,13].

5 Curvature operator

We define the curvature operator $\kappa: H(V) \to H(V)$ to be

$$\kappa f = -\frac{1}{2} div\left(\frac{df}{||df||}\right) \quad (10)$$

Next, we compute

$$(\kappa f)_j = \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}\,\left(\left(\frac{df}{||df||}\right)_{ij} - \left(\frac{df}{||df||}\right)_{ji}\right)$$

$$= \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}\,\left(\frac{1}{\|d_i f\|}\sqrt{w_{ij}}(f_j - f_i) - \frac{1}{\|d_j f\|}\sqrt{w_{ij}}(f_i - f_j)\right)$$

$$= \frac{1}{2}\sum_{i\sim j} w_{ij}\left(\frac{1}{\|d_i f\|} + \frac{1}{\|d_j f\|}\right)(f_j - f_i)$$

Thus, we have

$$(\kappa f)_j = \frac{1}{2}\sum_{i\sim j} w_{ij}\left(\frac{1}{\|d_i f\|} + \frac{1}{\|d_j f\|}\right)(f_j - f_i) \quad (11)$$

From the above formula, we have

$$d_i f = ((df)_{ij} : j \sim i)^T \text{ (12)}$$

The local variation of *f* at *i* is defined to be

$$\|d_i f\| = \sqrt{\sum_{j\sim i} (df)_{ij}^2} = \sqrt{\sum_{j\sim i} w_{ij}(f_j - f_i)^2} \text{ (13)}$$

To avoid the zero denominators in (11), the local variation of *f* at *i* is defined to be

$$\|d_i f\| = \sqrt{\sum_{j\sim i} (df)_{ij}^2 + \epsilon}, \text{ (14)}$$

where $\epsilon = 10^{-10}$.

The graph curvature is a non-linear operator.

Let $S_1(f) = \sum_i \|d_i f\|$, we have the following **theorem 2**

$$D_f S_1 = \kappa f \text{ (15)}$$

The proof of the above theorem can be found from [12,13].

6 p-Laplace operator

We define the p-Laplace operator $\Delta_p : H(V) \to H(V)$ to be

$$\Delta_p f = -\frac{1}{2} div(\|df\|^{p-2} df) \text{ (16)}$$

Clearly, $\Delta_1 = \kappa$ and $\Delta_2 = \Delta$. Next, we compute

$$(\Delta_p f)_j = \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}(\|df\|^{p-2} df_{ij} - \|df\|^{p-2} df_{ji})$$

$$= \frac{1}{2}\sum_{i\sim j} \sqrt{w_{ij}}(\|d_i f\|^{p-2}\sqrt{w_{ij}}(f_j - f_i) - \|d_j f\|^{p-2}\sqrt{w_{ij}}(f_i - f_j))$$

$$= \frac{1}{2}\sum_{i\sim j} w_{ij}(\|d_i f\|^{p-2} + \|d_j f\|^{p-2})(f_j - f_i)$$

Thus, we have

$$(\Delta_p f)_j = \frac{1}{2}\sum_{i\sim j} w_{ij}(\|d_i f\|^{p-2} + \|d_j f\|^{p-2})(f_j - f_i) \text{ (17)}$$

Let $S_p(f) = \frac{1}{p}\sum_i \|d_i f\|^p$, we have the following **theorem 3**

$$D_f S_p = p\Delta_p f \text{ (18)}$$

The proof of the above theorem can be found from [12,13].

7 Discrete regularization on graphs and speech recognition problem

Given a speech network *G=(V,E)*. *V* is the set of all speech samples in the network and *E* is the set of all possible interactions between these speech samples. Let *y* denote the initial function in *H(V)*. y_i can be defined as follows

$$y_i = \begin{cases} 1 \; if \; speech \; sample \; i \; belongs \; to \; the \; class \\ -1 \; if \; speech \; sample \; i \; does \; not \; belong \; to \; the \; class \\ 0 \; otherwise \end{cases}$$

Our goal is to look for an estimated function *f* in *H(V)* such that *f* is not only smooth on *G* but also close enough to an initial function *y*. Then each speech sample *i* is classified as $sign(f_i)$. This concept can be formulated as the following optimization problem

$$argmin_{f \in H(V)} \{ S_p(f) + \frac{\mu}{2} \|f - y\|^2 \} \quad (19)$$

The first term in (19) is the smoothness term. The second term is the fitting term. A positive parameter μ captures the trade-off between these two competing terms.

7.1 2-smoothness

When *p=2*, the optimization problem (19) is

$$argmin_{f \in H(V)} \{ \frac{1}{2} \sum_i \|d_i f\|^2 + \frac{\mu}{2} \|f - y\|^2 \} \quad (20)$$

By theorem 1, we have

Theorem 4: The solution of (20) satisfies

$$\Delta f + \mu(f - y) = 0 \quad (21)$$

Since Δ is a linear operator, the closed form solution of (21) is

$$f = \mu(\Delta + \mu I)^{-1} y, \quad (22)$$

Where *I* is the identity operator and $\Delta = D - W$. (22) is the algorithm proposed by [5].

7.2 1-smoothness

When *p=1*, the optimization problem (19) is

$$argmin_{f \in H(V)} \{ \sum_i \|d_i f\| + \frac{\mu}{2} \|f - y\|^2 \}, \quad (23)$$

By theorem 2, we have

Theorem 5: The solution of (23) satisfies

$$\kappa f + \mu(f - y) = 0, \quad (24)$$

The curvature κ is a non-linear operator; hence we do not have the closed form solution of equation (24). Thus, we have to construct iterative algorithm to obtain the solution. From (24), we have

$$\frac{1}{2}\sum_{i\sim j} w_{ij}\left(\frac{1}{\|d_i f\|}+\frac{1}{\|d_j f\|}\right)(f_j - f_i) + \mu(f_j - y_j) = 0 \quad (25)$$

Define the function $m: E \rightarrow R$ by

$$m_{ij} = \frac{1}{2} w_{ij}\left(\frac{1}{\|d_i f\|}+\frac{1}{\|d_j f\|}\right) \quad (26)$$

Then (25)

$$\sum_{i\sim j} m_{ij}(f_j - f_i) + \mu(f_j - y_j) = 0$$

can be transformed into

$$\left(\sum_{i\sim j} m_{ij} + \mu\right) f_j = \sum_{i\sim j} m_{ij} f_i + \mu y_j \quad (27)$$

Define the function $p: E \rightarrow R$ by

$$p_{ij} = \begin{cases} \frac{m_{ij}}{\sum_{i\sim j} m_{ij} + \mu} & if\ i \neq j \\ \frac{\mu}{\sum_{i\sim j} m_{ij} + \mu} & if\ i = j \end{cases} \quad (28)$$

Then

$$f_j = \sum_{i\sim j} p_{ij} f_i + p_{jj} y_j \quad (29)$$

Thus we can consider the iteration

$$f_j^{(t+1)} = \sum_{i\sim j} p_{ij}^{(t)} f_i^{(t)} + p_{jj}^{(t)} y_j \text{ for all } j \in V$$

to obtain the solution of (23).

7.3 p-smoothness

For any number *p*, the optimization problem (19) is

$$argmin_{f\in H(V)}\{\frac{1}{p}\sum_i \|d_i f\|^p + \frac{\mu}{2}\|f - y\|^2\}, \quad (30)$$

By theorem 3, we have

Theorem 6: The solution of (30) satisfies

$$\Delta_p f + \mu(f - y) = 0, \quad (31)$$

The *p-Laplace* operator is a non-linear operator; hence we do not have the closed form solution of equation (31). Thus, we have to construct iterative algorithm to obtain the solution. From (31), we have

$$\frac{1}{2}\sum_{i\sim j} w_{ij}\left(\|d_i f\|^{p-2} + \|d_j f\|^{p-2}\right)(f_j - f_i) + \mu(f_j - y_j) = 0 \quad (32)$$

Define the function $m: E \rightarrow R$ by

$$m_{ij} = \frac{1}{2} w_{ij}(\|d_i f\|^{p-2} + \|d_j f\|^{p-2}) \quad (33)$$

Then equation (32) which is

$$\sum_{i\sim j} m_{ij}(f_j - f_i) + \mu(f_j - y_j) = 0$$

can be transformed into

$$\left(\sum_{i\sim j} m_{ij} + \mu\right) f_j = \sum_{i\sim j} m_{ij} f_i + \mu y_j \quad (34)$$

Define the function $p: E \rightarrow R$ by

$$p_{ij} = \begin{cases} \frac{m_{ij}}{\sum_{i\sim j} m_{ij} + \mu} & if\ i \neq j \\ \frac{\mu}{\sum_{i\sim j} m_{ij} + \mu} & if\ i = j \end{cases} \quad (35)$$

Then

$$f_j = \sum_{i\sim j} p_{ij} f_i + p_{jj} y_j \quad (36)$$

Thus we can consider the iteration

$$f_j^{(t+1)} = \sum_{i\sim j} p_{ij}^{(t)} f_i^{(t)} + p_{jj}^{(t)} y_j \text{ for all } j \in V$$

to obtain the solution of (30).

8 Experiments and Results

In this paper, the set of 1000 speech samples recorded of 20 different words (50 speech samples per word) are used for training. Then another set of 1000 speech samples of these words are used for testing the sensitivity measure. This dataset is available from the IC Design lab at Faculty of Electricals-Electronics Engineering, University of Technology, Ho Chi Minh City. After being extracted from the conventional MFCC feature extraction method, the column sum of the MFCC feature matrix of the speech sample will be computed. The result of the column sum which is the R^{26*1} column vector will be used as the feature vector of the un-normalized graph p-Laplacian based semi-supervised learning algorithms.

There are three ways to construct the similarity graph from these feature vectors:

a. The ε-neighborhood graph: Connect all speech samples whose pairwise distances are smaller than ε.

b. k-nearest neighbor graph: Speech sample *i* is connected with speech sample *j* if speech sample *i* is among the k-nearest neighbor of speech sample j or speech sample j is among the k-nearest neighbor of speech sample i.

c. The fully connected graph: All speech samples are connected.

In this paper, the similarity function is the Gaussian similarity function

$$s\left(f(:,i),f(:,j)\right) = \exp\left(-\frac{d\left(f(:,i),f(:,j)\right)}{t}\right),$$

where $f(:,i)$ is the feature vector of speech sample i.

In this paper, t is set to 10^6 and the 3-nearest neighbor graph is used to construct the similarity graph from this dataset.

In this section, we experiment with the above proposed un-normalized graph p-Laplacian methods with *p=1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9*, the current state of the art network based semi-supervised learning method (i.e. the un-normalized graph Laplacian based semi-supervised learning method *p=2*) in terms of the sensitivity performance measure. All experiments were implemented in Matlab 6.5 on virtual machine. The sensitivity measure Q is given as follows

$$Q = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are defined in the following table 1:

Table 1: Definitions of TP, TN, FP, and FN

		Predicted Label	
		Positive	*Negative*
Known Label	*Positive*	*True Positive (TP)*	*False Negative (FN)*
	Negative	*False Positive (FP)*	*True Negative (TN)*

In these experiments, the parameter μ is set to 1. For this dataset, the sensitivity performance measures of the above proposed methods and the current state of the art methods (applied to the speech network of 20 words) are given in the following table 2:

Table 2: The comparison of the sensitivities of proposed methods with different p-values

Sensitivity performance measures (%)	p=1	99.4
	p=1.1	99.4
	p=1.2	99.4
	p=1.3	99.4
	p=1.4	99.4
	p=1.5	99.4
	p=1.6	99.4
	p=1.7	99.4
	p=1.8	99.4
	p=1.9	99.4
	p=2 (the current state of the art network based semi-supervised learning method)	99.3

The results from the above table shows that the un-normalized graph p-Laplacian semi-supervised learning methods are at least as good as the current state of the art method (*p=2*) in speech recognition problem but often lead to better sensitivity performance measures.

9 Conclusion

We have developed the detailed regularization frameworks for the un-normalized graph p-Laplacian semi-supervised learning methods applying to speech recognition problem. Experiments show that the un-normalized graph p-Laplacian semi-supervised learning methods are at least as good as the current state of the art method (i.e. *p=2*).

Moreover, these un-normalized graph p-Laplacian semi-supervised learning methods can not only be used in classification problem but also in ranking problem. In specific, given a set of genes (i.e. the queries) involved in a specific disease (for e.g. leukemia), these methods can also be used to find more genes involved in the same disease by ranking genes in gene co-expression network (derived from gene expression data) or the protein-protein interaction network or the integrated network of them. The genes with the highest rank then will be selected and then checked by biologist experts to see if the extended genes in fact are involved in the same disease. This problem is biomarker discovery in cancer classification.

Finally, the un-normalized graph p-Laplacian semi-supervised learning methods are only applied to small network of 20 words due to the high time complexity of these proposed methods. In the future, we will try to apply these methods to larger networks of 30, 40, and 50 words to check their scalability. Moreover, in order to reduce the time complexity of these generalized methods, we can also implement the parallel version of these methods using C and MPI.

ACKNOWLEDGEMENTS
This work is funded by the Ministry of Science and Technology, State-level key program, Research for application and development of information technology and communications, code KC.01.23/11-15

References

[1] Rabiner, Lawrence, and Biing-Hwang Juang. "Fundamentals of speech recognition." (1993)
[2] Ganapathiraju, Aravind. *Support vector machines for speech recognition*. Diss. Mississippi State University, 2002.
[3] Marshall, Austin. "Artificial neural network for speech recognition." *2nd Annual Student Research Showcase* (2005).
[4] Labiak, John, and Karen Livescu. "Nearest Neighbors with Learned Distances for Phonetic Frame Classification." *INTERSPEECH*. 2011.
[5] Trang, Hoang, and Loc Hoang Tran. "Graph Based Semi-supervised Learning Methods Applied to Speech Recognition Problem." *Nature of Computation and Communication*. Springer International Publishing, 2014. 264-273.
[6] Zhu, Xiaojin, and Zoubin Ghahramani. *Learning from labeled and unlabeled data with label propagation*. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.
[7]Zhou, Dengyong, et al. "Learning with local and global consistency." *Advances in neural information processing systems* 16.16 (2004): 321-328.
[8] Zhou, Dengyong, Jiayuan Huang, and Bernhard Scholkopf. "Beyond pairwise classification and clustering using hypergraphs." *Proceedings of the Neural Information Processing Systems*. 2005.
[9] Zhou, Dengyong, Jiayuan Huang, and Bernhard Schölkopf. "Learning with hypergraphs: Clustering, classification, and embedding." *Advances in neural information processing systems*. 2006
[10]Tran, Loc. "Hypergraph and protein function prediction with gene expression data." *arXiv preprint arXiv:1212.0388* (2012).
[11]Hoang Trang, Tran Hoang Loc, Huynh Bui Hoang Nam "Hypergraph based semi-supervised learning algorithms applied to speech recognition problem: a novel approach" Journal of Science and Technology, VietNam Academy of Science and Technology, ISSN: 0866 708X
[12]Zhou, Dengyong, and Bernhard Schölkopf. "Regularization on discrete spaces."*Pattern Recognition*. Springer Berlin Heidelberg, 2005. 361-368.

[13]Zhou D. and Schölkopf B. “Discrete Regularization” Book chapter, Semi-Supervised Learning, 221-232. (Eds.) O. Chapelle, B. Schölkopf and A. Zien, MIT Press, Cambridge, MA, 2006

[14]Tran, Loc. "The Un-normalized graph p-Laplacian based semi-supervised learning method and protein function prediction problem." *Knowledge and Systems Engineering*. Springer International Publishing, 2014. 23-35.

[15]Loc Hoang Tran and Linh Hoang Tran “Un-normalized graph p-Laplacian semi-supervised learning method applied to cancer classification problem” Journal of Automation and Control Engineering, Vol. 3, No. 1, pp. 23-28, February 2015

[16]Liu, Yuzong, and Katrin Kirchhoff. "Graph-based semi-supervised acoustic modeling in DNN-based speech recognition." *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014