Semi-supervised Learning on Hypergraphs: Analyzing and Exploring

Project Plan

Chen Jiali 3035085695
Zhang Ying 3035084366

CONTENTS

Abstract……..……..…….3
1.Introduction……..……..3
· *Semi-supervised Learning*
· *Hypergraph v.s. Normal Graph*
· *Apply Hypergraph to Semi-supervised Learning*
2.Literature Review……..…….3
3.Motivation……..……4
4.Objectives……..…….5
5.Methodology……..…….5
· *First Stage*
· *Second Stage*
· *Third Stage*
6.Scope……..……6
7.Difficulties and Risks……..……...6
8.Schedule……..……...7
9.Conclusion…….……....7
References……..……...8

Abstract

Semi-supervised learning is a relatively new topic which has intimate connections between machine learning and pattern recognition. Meanwhile, graph based learning is one of the most significant fields in semi-supervised learning. The purpose of our project is to implement and compare two existing hypergraph based semi- supervised learning methods so as to derive a new method hopefully in both programming and math ways. In this project plan, some background knowledge, objectives, methodology, difficulties, project schedule and so on will be presented.

1.Introduction

Machine learning, theoretically speaking, is the process to give a computer the ability to imitate the learning behavior of human to acquire new knowledges or skills. Within recent decades, the topic of machine learning has attracted increasing attention. Because of the explosion of machines' computing ability, this technology experiences a rapid development. Examples of utilization can be found in various fields such as computer vision, search engine, computer security as well as daily issues like weather forecasting.

Semi-supervised learning is one type of machine learning skills with high learning accuracy and efficiency. Compared with supervised and unsupervised learning, this training process makes use of both labeled and unlabelled data and hence provides a better performance. Up to now, there are a number of semi-supervised learning methods proposed by scholars already, and among them, graph-based methods play a very significant role.

Since learning on normal graphs are relatively well-studied, we would like to mainly focus on hypergraphs, which is an expansion of normal graphs, so that more information can be included, connected and utilized. In general, we are going to analyze two existing hypergraph based semi-supervised learning methods by implementation and hopefully design our own method.

2.Literature Review

Semi-supervised Learning

Semi-supervised learning successfully finds a balance between supervised and unsupervised learning by combining both of them. Training with a large size of labeled data, supervised learning is accurate but time-consuming, expensive and annotation demanding. While unsupervised learning is just the opposite. By training using labeled and unlabeled data combinedly, semi-supervised learning gives consideration to both accuracy and efficiency[5].

Hypergraph v.s. Normal Graph

Traditionally, relationships among objects are assumed to be pairwise, thus normal graphs are used to model problems. However, in many real world problems, relations may be much more

complex so that modeling with normal graphs will degenerate them. Hence, hypergraphs are introduced for modeling. For instance, 3 features (could be if it is colorful, if it is smelly and if the shape is like an umbrella) are considered when classifying 7 types of mushrooms into edible kind or poisonous kind (Detailed comparison will be shown in the Figure1 below). Apparently, information is lost in normal graph since an edge between two vertices only implies that these two mushrooms share one same feature but not what exactly the feature is, while hypergraph can provide detailed illustration[2]. Thus learning on hypergraph could definitely provide solutions to the cases with much more complicated relations.

	e1	e2	e3
v1	1	0	0
v2	1	0	1
v3	0	0	1
v4	0	0	1
v5	0	1	0
v6	0	1	1

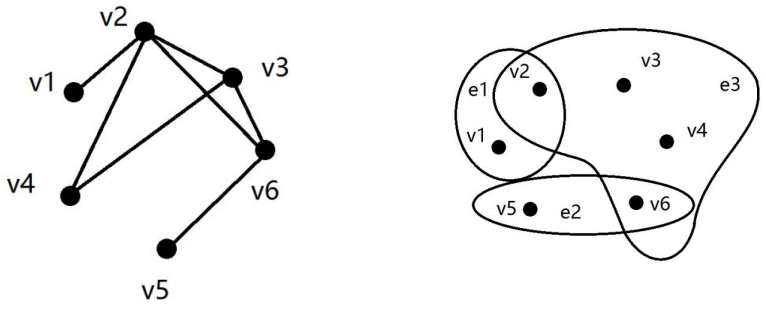

Figure 1: Simple graph(middle) v.s. hypergraph(right). Left: a matrix showing the the features of the mushrooms. Column set V = {v1, v2, v3, v4, v5, v6} represents different mushrooms and row set E = {e1, e2, e3} are the features. If mushroom i has feature j, we set entry (v_i, e_j) to be 1, otherwise 0. Middle: a normal graph where two vertices v_i, v_j are connected with an undirected edge if there are common features between mushroom i and j. Right: a hypergraph with which the complicated relationships are fully illustrated.

Apply Hypergraph to Semi-supervised Learning

As hypergraph is more powerful and deliver a more explicit modeling of cases, the combination of its theories and semi-supervised learning algorithms is able to broaden the application field of semi-supervised learning in real life. Hence, it is quite popular in recent years to research on different mathematical approaches to such combination and improve the efficiency of the implementation.

3.Motivation

Based on the current research performed by us, there already exist a number of methods on this topic delivered by some research groups. However, there are no research about analysing the performance of these methods with respect to both learning accuracy as well as time and space complexity. Thus our team would like to study on the performance of current methods and further deliver our own.

4.Objectives

In this project, the objectives can be generally separated into two phases: comparing and exploring.

The first stage of this project aims to implement the two algorithms from several existing research studies and conduct testing experiments. Performance of the algorithms which includes accuracy as well as time and space complexity, will be compared and analyzed. The results of the analysis are expected to be demonstrated as charts.

After the previous implementation and comparison, a deep understanding should be built. Ultimately, the goal of our team is to provide an algorithm with higher efficiency and lower memory consumption, which will also be implemented. The final result will be derived by both mathematical computation and programming.

5.Methodology

First Stage

At the first stage of this project when accumulating background knowledge, the algorithms including many math equations should be fully understood. Thus we are required to follow the math contents of the papers line by line and perform the computation at the same time. During this process, both self works and discussions are essential in understanding the principles behind.

Second Stage

With the base of all the principles, next stage could be reached, in which when implementing the algorithms, Python will be chosen as the programming language. (Matlab would be another alternative that may be taken into consideration in the future.) Detailed steps of implementation are stated below:

1. Derive the methods of analyzing outcome errors as well as time and space complexity.
2. Import data sets normally provided by some research institution and could be acquired freely from the Internet. Before being imported, the size and the contents of a data set will be checked in case this data set is not suitable for our experiment.
3. Write programs to implement both algorithms in order to get the machine learning results. This procedure should be performed a great many times using various sets of test cases in terms of distinct sources and different sizes to ensure the reliability of the result. Besides, the same data set will be used for both algorithms in the same round for comparison. Afterwards, the average performance will also be compared as their general character. Finally, multiple plots should be generated to show the result.

Third Stage

The remaining work will be finding the deficiency or the limitations of the current methods so as to make improvements or even design our own method. Some potential modifications or equations are expected to be proposed at first. After that they will be implemented and tested by huge amount of data, which would be a similar process as the second stage, to see if there is any progress achieved. If possible, direct proof by means of math of our proposal will also be derived.

6.Scope

Our project will mainly concentrate on the chosen two hypergraph based semi-supervised learning algorithms. Other machine learning algorithms will not be included such as supervised learning and reinforcement learning. Also, learning algorithms based on normal graphs will not be discussed.

When conducting the implementation, ten proper data sets from the link below which is a database for machine learning will be selected and utilized in our experiment.
https://archive.ics.uci.edu/ml/machine-learning-databases

7.Difficulties and Risks

According to our estimation of this project, there might be two main problems: the limitation of the computation ability of our private computer; the uncertainty of the existence of other approaches.

Due to the limited computation power, size of the test cases will be restricted. Thus it will be a problem whether the limited size will have adversely affection on the final performance analysis. To relieve or even resolve the impact, we plan the following two solutions: First, despite the limited case size, the quantity of cases can be increased. With more rounds of tests, the results, as the average of a really large number of tests, will be more reasonable. Second, testing with the computers in CS lab which is much more powerful can handle computation of higher level .

For the second risk, there are quite a number of mathematical approaches toward this problem, thus, although we might not find a generally better solution, it is very likely to deliver our own method whose performance is at least not worse than the others. Relevantly, the advantages and disadvantages of this method, compared with others, should be considered. For the worst case, we can further analyze the previous approaches and determine their applications as our final result.

8.Schedule

Sept 30	*Deliverables of project website and detailed project plan.
Oct 20	*The subgradient approach (method 1) understood. *Implementation of method 1 *Analysis report of the performance (accuracy and complexity)
Nov 10	*The total variation approach (method 2) understood. *Implementation of method 2 *Analysis report of the performance (accuracy and complexity)
Nov 20	*Two approaches tested with different test data sets *Further and comprehensive comparison made
Dec 31	*Comparison result analyzed *Deliverable of Interim report
Jan 31	*Problem analysis and improvement direction determined
Feb 28	*Mathematically implementation of the improvement
Mar 20	*Implement our own approach to verify the improvement
Mar 31	*Analyze the performance and uniqueness of our approach
April 10	*Finalized conclusion *Final report
April 17	*Rehearse for the presentation.

9.Conclusion

So far we have described the background knowledge and detailed plan of our project. This project will deliver a performance analysis report by comparing two current semi-supervised learning methods, which are both hypergraph based. A new approach which might have better performance will also be proposed. Our team believe that this project will definitely benefit our group members a lot and hopefully can make a contribution to semi-supervised learning area.

References

[1] Zhu, Xiaojin, Zoubin Ghahramani, and John Lafferty. "Semi-supervised learning using gaussian fields and harmonic functions." *ICML*. Vol. 3. 2003.

[2] Zhou, Dengyong, Jiayuan Huang, and Bernhard Schölkopf. "Learning with hypergraphs: Clustering, classification, and embedding." *Advances in neural information processing systems*. 2006.

[3] Hein, Matthias, et al. "The total variation on hypergraphs-learning on hypergraphs revisited." *Advances in Neural Information Processing Systems*. 2013.

[4] Zhou, Dengyong, et al. "Learning with local and global consistency."*Advances in neural information processing systems* 16.16 (2004): 321-328.

[5] Blum, Avrim, and Shuchi Chawla. "Learning from labeled and unlabeled data using graph mincuts." (2001): 19.