Multi-Graph Learning

Jia Wu

Faculty of Engineering and Information Technology

University of Technology Sydney

A thesis submitted for the degree of

Doctor of Philosophy

02 December 2015

To dedicate this thesis to my loving parents:
Bo Wu, and *Yuanjiao Yang*.

CERTIFICATE OF ORIGINAL AUTHORSHIP

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Student: Jia Wu
Date: 02.12.2015

Acknowledgements

I benefited and learned a lot from my supervisors, my colleagues, and my friends during the PhD study in University of Technology Sydney, Australia. I wish to take this opportunity to thank all of them.

First of all, I would like to express my sincere appreciation and gratitude to Prof. Xingquan Zhu. For me, it is so lucky to have Prof. Zhu as my advisor. Without his support, I would never have been able to complete this thesis. I could remember the time we discussed some research details, mainly because his widely knowledge background often motivates me to target some interesting fields, which has always been an eye-opening experience. With his kind encouragement and pushing, I could able to turn the negative into the positive when facing the difficulties and challenges. I miss the days when we pursued the top conference deadline together (*e.g.,* ICDM, CIKM, SDM, AAAI, IJCAI, *etc.*), especially the continuous 36 hours in the lab for the ICDM2013 deadline. I also want to give thanks to him as a friend of mine, for giving me invaluable instructions and suggestions during my life. In a word, he is a lighthouse, guiding my research life, illuminating the road of the research headway. Without him, there is no research direction; without a direction, there is no research life.

I wish to express my appreciation to Prof. Chengqi Zhang for his guidance aiming at my further career. With his support, I have the opportunity to attend lots of excellent conferences, such as ICDM, IJCAI, etc. I also give thanks to Prof. Zhihua Cai for providing me the opportunity as a technical leader of his The National High Technology Research and Development Program of China (863 Program). I am also grateful to Prof. Geoff Webb, Prof. Phillip S. Yu, Prof. Xingdong

Wu, Prof. Dacheng Tao, Prof. Jian Pei, Prof. Jieping Ye, Prof. Jie Tang, Prof Xiong Hui, Prof. Zhihua Zhou, and Prof. Jeffrey Xu Yu for giving me useful suggestions and sharing their research experience on conference so that I could improve my research skills.

I would like to place on record my sincere thanks for the all the hard work and dedication put in by my best friends/researchers: Dr. Shirui Pan, Dr. Zhibin Hong, Dr. Junyu Xuan, Dr. Maoying Qiao, Dr. Ting Guo, Dr. Meng Fang, Dr. Lianhua Chi, Dr. Chunyang Liu, Dr. Mingsong Mao, Dr. Lianyang Ma, Dr. Peng Zhang, Dr. Guodong Long, Dr. Jing Jiang, Barbara Monday, Haishuai Wang, and Shaoli Huang. Especially, I am deeply indebted to the following three people: Dr. Shirui Pan, Dr. Zhibin Hong, and Dr. Junyu Xuan. Dr. Shirui Pan, as my close brother, often unselfishly shares me with his source code, and many other research resources, especially at the beginning of my PhD study. Without his patience and spending much time to teach me, I could not target the top conferences and journals. The other two guys often help to handle the experimental data sets (Dr. Zhibin Hong for Image, and Dr. Junyu Xuan for Text).

Finally, I would also like to express my gratitude to my family: my parents (Bo Wu and Yuanjiao Yang), my uncle (Hao Wu) and auntie (Qin Yang), my younger sister (Yalin Wu), and brothers (Bi Wu and Xinxin Wu) for the trust and support bestowed on me. Especially, I am incredibly grateful to my lovely girlfriend Shan Xue for her fully supporting all my final decisions in a loving way. For every gentle smile, that makes my everyday life and study worthwhile.

Abstract

Multi-instance learning (MIL) is a special learning task where labels are only available for a bag of instances. Although MIL has been used for many applications, existing MIL algorithms cannot handle complex data objects, and all require that instances inside each bag are represented as feature vectors (*e.g.* being represented in an instance-feature format). In reality, many real-world objects are inherently complicated, and an object can be represented as multiple instances with dependency structures (*i.e.* graphs). Such dependency allows relationships between objects to play important roles, which, unfortunately, remain unaddressed in traditional instance-feature representations. Motivated by the challenges, this thesis formulates a new multi-graph learning paradigm for representing and classifying complicated objects. With the proposed multi-graph representation, the thesis systematically addresses several key learning tasks, including

Multi-Graph Learning: A graph bag contains one or multiple graphs, and each bag is labeled as either positive or negative. The aim of multi-graph learning is to build a learning model from a number of labeled training bags to predict previously unseen bags with maximum accuracy. To solve the problem, we propose two types of approaches: 1) Multi-Graph Feature based Learning (*gMGFL*) algorithm that explores and selects an optimal set of subgraphs as features to transfer each bag into a single instance for further learning; and 2) Boosting based Multi-Graph Classification framework (bMGC), which employs dynamic weight adjustment, at both graph- and bag-levels, to select one subgraph in each iteration to form a set of weak graph classifiers.

Multi-Instance Multi-Graph learning: A bag contains a number of instances and graphs in pairs, and the learning objective is

to derive classification models from labeled bags, containing both instances and graphs, to predict previously unseen bags with maximum accuracy. In the thesis, we propose a Dual Embedding Multi-Instance Multi-Graph Learning (*DE-MIMG*) algorithm, which employs a dual embedding learning approach to (1) embed instance distributions into the informative subgraphs discovery process, and (2) embed discovered subgraphs into the instance feature selection process.

Positive and Unlabeled Multi-Graph Learning: The training set only contains positive and unlabeled bags, where labels are only available for bags but not for individual graphs inside the bag. This problem setting raises significant challenges because bag-of-graph setting does not have features available to directly represent graph data, and no negative bags exits for deriving discriminative classification models. To solve the challenge, we propose a puMGL learning framework which relies on two iteratively combined processes: (1) deriving features to represent graphs for learning; and (2) deriving discriminative models with only positive and unlabeled graph bags.

Multi-Graph-View Learning: A multi-graph-view model utilizes graphs constructed from multiple graph-views to represent an object. In our research, we formulate a new multi-graph-view learning task for graph classification, where each object to be classified is represented graphs under multi-graph-view. To solve the problem, we propose a Cross Graph-View Subgraph Feature based Learning ($gCGVFL$) algorithm that explores an optimal set of subgraph features cross multiple graph-views. In addition, a bag based multi-graph model is further used to relax the labeling by only requiring one label for each graph bag, which corresponds to one object. For learning classification models, we propose a multi-graph-view bag learning algorithm (MGVBL), to explore subgraphs from multiple graph-views for learning.

Experiments on real-world data validate and demonstrate the performance of proposed methods for classifying complicated objects using multi-graph learning.

Contents

Contents ix

List of Figures xv

List of Tables xxv

Nomenclature xxvi

1 Introduction 1

1.1 Background 1

1.2 Motivation 2

1.2.1 Online Webpage Recommendation 2

1.2.2 Bio-pharmaceutical Activity Test 4

1.2.3 Online Product Recommendation based on Review 4

1.2.4 Scientific Publication Categorization 5

1.2.5 Advantages of Multi-Graph Representation 5

1.3 Multi-Graph Learning Tasks 6

1.3.1 Multi-Graph Learning 6

1.3.2 Multi-Instance Multi-Graph Learning 7

1.3.3 Positive and Unlabeled Multi-Graph Learning 7

1.3.4 Multi-Graph-View Learning 8

1.4 Research Challenges 8

1.5 Main Contributions and Road Map 9

1.6 Selected Publications Related to The Thesis 14

2 Literature Review **17**
2.1 Multi-instance learning 17
2.1.1 Single-Instance Learner Based MIL 17
2.1.2 Bag-Based MIL Algorithms 18
2.2 Graph Learning 19
2.2.1 Global Distance based Approaches 19
2.2.2 Local Subgraph Feature based Approaches 20
2.3 Positive and Unlabeled Learning 21
2.4 Multi-View Feature based Learning 22

3 Multi-Graph Representation **25**
3.1 Single-Instance Representation 25
3.2 Single-Graph Representation 25
3.3 Multi-Instance Representation 27
3.4 Multi-Graph Representation 27

4 Multi-Graph Learning **29**
4.1 Introduction 29
4.2 Problem Statement 33
4.3 Feature based Multi-Graph Learning 36
4.3.1 Overall Framework of gMGFL 36
4.3.2 Multi-Graph Feature Learning Algorithm 36
4.3.2.1 Bag Constrained Subgraph Exploration 36
4.3.2.2 Bag Constrained Subgraph Search 40
4.3.2.3 gMGFL 44
4.3.3 Experiments 45
4.3.3.1 Data Sets 45
4.3.3.2 Baseline Methods 47
4.3.3.3 Experimental Settings 49
4.3.3.4 Experimental Results 50
4.4 Boosting for Multi-Graph Learning 62
4.4.1 Overall Framework of bMGC 62
4.4.2 Subgraph Exploration 64

4.4.2.1 Evaluation Criterion for Subgraphs 64
4.4.2.2 Upper Bound of bScore 68
4.4.2.3 Mining Bag Constrained Subgraph 68
4.4.3 bMGC 70
4.4.4 Experiments 73
4.4.4.1 DataSets 73
4.4.4.2 Baseline Methods 75
4.4.4.3 Experimental Settings 77
4.4.4.4 Experimental Results 78
4.4.5 Discussion 88
4.5 Conclusions 88

5 Multi-Instance Multi-Graph Learning 91
5.1 Introduction 91
5.2 Problem Statement 94
5.3 Overall Framework of MIMG Learning 95
5.4 Dual Embedding Feature Exploration 96
5.4.1 Embedding Feature Evaluation Criteria 98
5.4.1.1 FEC: Feature Evaluation Criteria 98
5.4.1.2 Dual Embedding Strategy 99
5.4.2 Embedding FEC for Subgraph Feature Exploration 99
5.4.3 Embedding FEC for Instance Feature Selection 104
5.4.4 Feature Concatenation 105
5.5 DE-MIMG Dual Embedding Algorithm 106
5.6 Experiments 107
5.6.1 DataSets 107
5.6.2 Baseline Methods 109
5.6.3 Experimental Setting 110
5.6.4 Experimental Results 111
5.6.4.1 Comparisons with MI Setting and MG Setting . . 112
5.6.4.2 Comparisons with Simple Graph and Instance Feature Concatenation 114
5.6.4.3 Efficiency of the Pruning Strategy 115

5.7 Conclusions 116

6 Positive and Unlabeled Multi-Graph Learning 119

6.1 Introduction 119

6.2 Problem Statement 122

6.3 Overall Framework of puMGL Learning 124

6.4 Positive and Unlabeled MG Learning 124

6.4.1 Optimization Framework 125

6.4.2 Optimize Subgraph Features 126

6.4.2.1 Confidence Weight Embedding 126

6.4.2.2 Subgraph Evaluation Criteria 127

6.4.2.3 Subgraph Exploration 128

6.4.3 Confidence Weight Optimization 130

6.4.3.1 Margin Graph Pool (MGP) 130

6.4.3.2 Confidence Weight Updating 131

6.4.4 puMGL Algorithm 133

6.5 Experiments 135

6.5.1 Experimental Settings 135

6.5.2 Baseline Methods 135

6.5.2.1 Bag-level methods 136

6.5.2.2 Graph-level methods 136

6.5.3 Online Product Recommendation Results 137

6.5.4 Content-based Image Retrieval Results 138

6.5.5 Bio-pharmaceutical Activity Test Results 140

6.5.6 Scientific Publication Categorization Results 142

6.5.7 Effectiveness of MGP Studies 143

6.5.8 Efficiency of the Pruning Strategy Studies 145

6.6 Conclusions 146

7 Multi-Graph-View Learning 147

7.1 Introduction 147

7.2 Multi-Graph-View Graph Learning 148

7.2.1 Problem Statement 151

7.2.2 Overall Framework of gCGVFL 151
7.2.3 gCGVFL 153
7.2.3.1 Cross Graph-View Subgraph Exploration 153
7.2.3.2 Maximum Discrimination and Minimum Redundancy 157
7.2.4 Experiments 164
7.2.4.1 Experimental Settings 164
7.2.4.2 Baseline Methods 164
7.2.4.3 Scientific Publication Categorization Results . . . 166
7.2.4.4 Content-based Image Retrieval Results 169
7.2.4.5 Detailed Algorithm Performance Studies 171
7.3 Multi-Graph-View Bag Learning 175
7.3.1 Problem Statement 176
7.3.2 Overall Framework of MGVBL 177
7.3.3 Multi-Graph-View Bag Learning Algorithm 177
7.3.3.1 Maximum Bag Margin Formulation 177
7.3.3.2 MGVBL 184
7.3.4 Experiments 185
7.3.4.1 Experimental Settings 185
7.3.4.2 Baseline Methods 185
7.3.4.3 Scientific Publication Text Categorization Results 186
7.3.4.4 Content-based Image Retrieval Results 188
7.3.4.5 Efficiency of the Pruning Strategy Studies 189
7.4 Conclusions 190

8 Conclusions 191

List of Figures

1.1 Traditional supervised learning where the object for classification is an individual instance. 3

1.2 Traditional multi-instance learning where each bag contains a number of instances, and the object for classification is a bag containing a number of instances. 3

1.3 Traditional graph learning where the object for classification is an individual graph. 3

1.4 An example of multi-graph representation for a webpage in Flickr online photo sharing system (http://www.flickr.com/). Each rectangular box represents one type of information source in the current page. Although the current page is marked as "Tsunami watchers", Image 1 and Text 3 are not relevant to the Tsunami, but Text 1, 2 and Image 2 are relevant to it. By converting content in each box into a graph, the whole webpage can be represented as a bag of graphs (*i.e.* a multi-graph representation). 4

1.5 An example of multi-graph representation for online product review. Each product receives a number of customer reviews, where each review contains text descriptions of the product with respect to a number of key features such as "taste", "appearance", and "palate". The main body of each review can form a graph representation by using key-words as nodes and key-word correlations as edges. The multiple reviews therefore form a bag of graphs for each product. 5

1.6 An example of multi-graph representation for a scientific publication. Each paper is represented as a multi-graph bag, where each graph inside the bag corresponds to the abstract of the paper or the abstract of the reference cited in the paper (a graph is formed by using keywords of the abstract as nodes and their correlations as edges). 6

1.7 Relationships among the proposed learning frameworks in this thesis. 12

3.1 Single-instance for image learning where an image is represented as an instance with each detector point represented as a feature in the vector space. 26

3.2 Graph for image learning in which an image is represented as a graph with each node corresponding to a small region of the image and adjacency between regions as edges. 26

3.3 Multi-instance learning for image annotation/classification in which an image is represented as a bag of instances in the vector space. . 27

3.4 Multi-graph learning for content-based image annotation in which each image (*i.e.,* bag) consists of a number of regions (*i.e.,* graphs considering local structure), where the compartments in each region are superpixels [1]. It is worth noting that after we use a graph structure to represent each region for the above two images, we find an interesting observation that some regions (*e.g.,* regions #2 in image 1 of Figure 3.3 and #5 in image 2) share common patterns, which correspond to certain part of the object (*e.g.,* the body of "leopard"). Accordingly, a graph representation is more accurate in denoting and preserving the local structure of the images than simply treating the whole region as an instance (*i.e.,* by using statistical visual feature vector). 28

4.1 Multi-graph learning where each bag contains a number of graphs, and the object for classification is a bag containing a number of graphs. 30

4.2 An example of subgraph feature representation for bags. B_1^+ and B_2^- are positive and negative bags, respectively. G_1^+ is a positive graph and G_2^-, G_3^-, and G_4^- are labeled negative. The feature value of a bag corresponding to each subgraph g_1 or g_2 is set to 1, iff there is a graph in the bag contains the subgraph, and 0 otherwise. . . . 34

4.3 The proposed multi-graph learning (gMGFL) framework: The objective is to find a set of discriminative subgraphs to convert each multi-graph bag into an instance in the new feature space for learning. It starts from subgraph candidate generation ①, and then uses the proposed bag constrained subgraph evaluation criteria to assign a score (gScore) for the discovered subgraph ②. After choosing the m subgraphs with the highest gScore in Definition 4.9, multi-graph bag representation ③ is used to help transfer each bag into a binary feature instance with its subgraph feature value set to 1 if the bag (*e.g.* B_1) contains the underling subgraph (*e.g.* g_1). At last, generic learning algorithms can be applied to train learning models for multi-graph learning ④. 35

4.4 A Search Space: DFS Code Tree with Branch-and-Bound Pruning. 41

4.5 An example of graph representation of a research paper entitled "Static analysis in datalog extensions" by using E-FCM [93]. The nodes are color coded with each color denoting one keyword. The edges represent correlations between keywords. For example, the correlation value between "describe" and "tree" is 0.007. In our experiments, all edges whose correlation values are less than threshold 0.005 are discarded. After that, we convert the weight graph in (A) into an unweighted graph showing in (B) for learning. . . . 46

4.6 Comparisons on DBLP data set by using different multi-graph learning algorithms: (A) KNN; (B) NB; (C) J48; and (D) SMO. . 49

4.7 Comparisons on NCI data set by using different multi-graph learning algorithms: (A) KNN; (B) NB; (C) J48; and (D) SMO. 50

4.8 Comparisons on Online Product Review data by using different multi-graph learning algorithms: (A) KNN; (B) NB; (C) J48; and (D) SMO. 51

4.9 Average classification accuracy (and standard deviation) comparisons between gMGFL and three multi-graph learning baseline models, gHSIC+MG, IG+MG and Topk+MG, on DBLP (A), NCI (B), and Online Product Review (C) data sets, with different number of subgraphs (varying from 10 to 200). 53
4.10 Comparisons on DBLP data set by using proposed gMGFL algorithm and generic multi-instance (MI) learning methods: (A) CitationKNN; (B) MISMO; (C) MIEMDD; and (D) MIOptimalBall. 54
4.11 Comparisons on NCI data set by using proposed gMGFL algorithm and generic multi-instance (MI) learning methods: (A) CitationKNN; (B) MISMO; (C) MIEMDD; and (D) MIOptimalBall. 55
4.12 Comparisons on Online Product Review data set by using proposed gMGFL algorithm and generic multi-instance (MI) learning methods: (A) CitationKNN; (B) MISMO; (C) MIEMDD; and (D) MIOptimalBall. 56
4.13 Average classification accuracy (and standard deviation) comparison between gMGFL and three multi-instance learning baseline models, gHSIC+MI, IG+MI and Topk+MI, over subgraphs varying from 10 to 200 on DBLP (A), NCI (B), and Online Product Review (C) data sets, respectively. 58
4.14 (a) Discriminative capability of selected subgraphs, and (b) bag coverage rate for the subgraph set on three real-world data sets, respectively. 59
4.15 Average CPU runtime and classification accuracy comparison between gMGFL *v.s.* unbounded gMGFL (UgMGFL) with respect to different *min_sup* and a fixed number of subgraphs m=100. . . 60
4.16 Average CPU runtime and accuracy comparison between gMGFL *v.s.* unbounded gMGFL (UgMGFL) with respect to different selected subgraph number (m values) and a fixed *min_sup* (4% for DBLP, 15% for NCI and 10% for Online Product Review data). . 61
4.17 An overview of the proposed bMGC framework. 63
4.18 Accuracy on DBLP(AI *vs.* CV) by using proposed bMGC and boosting MIL methods: (A) MIBoost and (B) MIOptimalBall. . . 77

4.19 Accuracy on DBLP(AI *vs.* DB) by using proposed bMGC and boosting MIL methods: (A) MIBoost and (B) MIOptimalBall. . . 77

4.20 Accuracy on NCI(1) by using proposed bMGC and boosting MIL methods: (A) MIBoost and (B) MIOptimalBall. 78

4.21 Accuracy on NCI(109) by using proposed bMGC and boosting MIL methods methods: (A) MIBoost and (B) MIOptimalBall. 78

4.22 Accuracy on DBLP(AI *vs.* CV) by using bMGC and generic MIL methods: (A) CitationKNN; (B) MIRI; (C) MIEMDD, and (D) MISMO. 80

4.23 Accuracy on DBLP(AI *vs.* DB) by using bMGC and generic MIL methods: (A) CitationKNN; (B) MIRI; (C) MIEMDD, and (D) MISMO. 81

4.24 Accuracy on NCI(1) by using bMGC and generic MIL methods: (A) CitationKNN; (B) MIRI; (C) MIEMDD, and (D) MISMO. . . 82

4.25 Accuracy on NCI(109) by using bMGC and generic MIL methods: (A) CitationKNN; (B) MIRI; (C) MIEMDD, and (D) MISMO. . . 83

4.26 Accuracy comparisons by using bMGC and bMGC-G on DBLP and NCI datasets, respectively. 85

4.27 The error rate curves on DBLP (AI *vs.* CV, AI *vs.* DB) and NCI(1 and 109) multi-graph datasets in terms of the number of iterations. 86

4.28 CPU runtime for bMGC *vs* ubMGC with different min_sup under subgraph number $m = 100$ on DBLP and NCI datasets, respectively. 87

5.1 Multi-instance multi-graph learning where each bag contains a number of instances and graphs in pairs. 91

5.2 A conceptual view of DE-MIMG: The overall framework is to find optimal representation to convert a bag ① into an instance in the new feature space ③ by using dual embedding strategy ②. The dual embedding consists of two parts: 1) finding subgraph features by embedding instance distributions, and 2) selecting instance features by embedding graph distribution information. The former first builds an instance embedded matrix L (a), which embeds instance distributions to help discover informative subgraphs (b), which are further used to re-represent graphs, as shown in (c). Alternately, the latter builds a graph embedded matrix L' (d) to find an optimal set of instance features (e) to re-represent instances in refined instance space (f). The optimal subgraph features and instance features are concatenated to represent each bag in concatenation space for MIMG learning. 97

5.3 The instance embedding is to help mine optimal subgraphs ② to represent graphs ③ under the FEC rule (*i.e.* objects with the same label are close each other, and are separated from objects with different labels.) and meanwhile keep the distribution information in instance domain ①. The process starts from (a) using instance distributions to discover optimal subgraphs (b), and further represent graphs in graph space (d). 98

5.4 Accuracy comparisons on Online Product Review dataset by using proposed DE-MIMG and IG based baselines on MIL methods: (A) CitationKNN; (B) MIEMDD; (C) MIOptimallBall; and (D) MISVM.110

5.5 Comparisons on Online Product Review dataset by using proposed dual embedding algorithm DE-MIMG and gSSC based baselines on MIL methods: (A) CitationKNN; (B) MIEMDD; (C) MIOptimallBall; and (D) MISVM. 111

5.6 Accuracy comparisons on DBLP dataset by using proposed dual embedding algorithm DE-MIMG and IG based baselines on MIL methods: (A) CitationKNN; (B) MIEMDD; (C) MIOptimallBall; and (D) MISVM. 112

5.7 Accuracy comparisons on DBLP dataset by using proposed DE-MIMG and gSSC based baselines on MIL methods: (A) CitationKNN; (B) MIEMDD; (C) MIOptimallBall; and (D) MISVM. . 113

5.8 Average CPU runtime for DE-MIMG *v.s.* unpruned UDE-MIMG with different *min_sup* under a fixed number of subgraphs m=200 on Online Product Review and DBLP datasets, respectively. . . . 116

6.1 Multi-graph learning with only positive and unlabeled bags in which the object for classification is a bag of graphs. 120

6.2 A conceptual view for positive and unlabeled multi-graph learning framework (puMGL): Once the training multi-graph set is given, each unlabeled bag is initially assigned with a confidence value (*i.e.,* weight) in order to establish the Margin Graph Pool (MGP) for further learning. And then some informative subgraphs will be explored to represent those graphs in MGP and the training set (b) by utilizing the proposed confidence weight embedding approach (a). After that, a classifier can be built based on the MGP (in vector feature space) to update the weight value of the unlabeled bag. Accordingly, some "reliable negative bags" could be identified from unlabeled set (d) to help update the graphs in MGP (e), which consists of the "least negative graph" from the unlabeled set and those "most positive graph" from the positive set. The quality of the "reliable negative bags" and the corresponding graphs in MGP will continue to be improved through the iterative process until we get the optimal classifier. 123

6.3 F-score comparisons on *Online Product Review* with respect to different r values (A) and different number of subgraphs m (B), respectively. 138

6.4 Example images used in the experiment from the COREL image database. The first three rows show images in category "Cats", including "Lion" (The first row), "Tiger" (The second row) and "Leopard" (The third row), and the last three rows represent the unlabeled image examples. 139

6.5 F-score comparisons on *Corel Region based Image* with respect to different r values (A) and different number of subgraphs m (B), respectively. 140

6.6 F-score comparisons on *NCI Bio-pharmaceutical* with respect to different r values (A) and different number of subgraphs m (B), respectively. 141

6.7 F-score comparisons on *DBLP Scientific Publication* with respect to different r values (A) and different number of subgraphs m (B), respectively. 142

6.8 Some "most positive patterns" examples explored by using the proposed MGP framework. Each item in MGP corresponds to one region (*e.g.,* #5 in the image on the left corner), which is selected under the criterion proposed in Section 6.4.3. The objective of building MGP is to mine subgraph feature (*e.g.,* g as shown in first row) with high puScore for further learning. Example of less "positive" examples are also reported in the second row. 144

6.9 Average CPU runtime for puMGL *v.s.* unpruned UpuMGL with different *min_sup* under a fixed number of subgraphs m=60 on *Online Product Review*, *Corel Region based Image*, *Bio-pharmaceutical Activity*, and *DBLP Scientific Publication* data set, respectively. . 145

7.1 An example of multi-graph-view representation for images where each image is represented as graphs with multiple graph-views (*e.g.,* colour view #1 and texture view #2). For the same object, patterns shared by the objects may not exist in all feature views. For example, the two images have no common pattern in the colour space (*i.e.,* the graph G^1 and G'^1 in colour view), due to different lighting conditions, but they share same patterns in the texture space (*i.e.,* the graph G^2 and G'^2 in texture view). Therefore, multi-graph-view representation is more powerful in representing the object content than single-graph-view representation (the graph composition is detailed in Section 7.2.4.4). 149

7.2 Multi-Graph-View Learning where graphs are represented from different graph-views. 150

7.3 A conceptual view of Cross Graph-View Subgraph Feature based Leaning (gCGVFL) for graph classification: After representing the original dataset (*e.g.,* images) into multi-graph-view graphs ①, gCGVFL intends to find optimal subgraph features to convert a multi-graph-view graph ② as an instance in the new feature space ④ by using proposed cross graph-view subgraph feature selection ③. More specifically, given a number of labeled multi-graph-view graphs with v views, gCGVFL starts from assigning equal weight values to each view. For each single graph-view, gCGVFL first builds a label distribution information embedded matrix L, *e.g.,* L^1 or L^v (a), to help discover informative subgraphs. By combining the view evaluation and redundancy checking, gCGVFL can obtain irredundant cross graph-view subgraphs (b). Because the initial view weights are imprecise to capture the importance of each view, an iterative weight updating process (detailed in Section 7.2.3.2) is proposed to update view weight value and refine the quality of selected cross graph-view subgraphs (c). 152

7.4 Comparisons on *DBLP dataset* on each single view: (A) Reference Relationship (B) Abstract. 167

7.5 Comparisons on *DBLP dataset* from multiple views via different view combination methods. 168

7.6 Comparisons on *Image dataset* on each single view: (A) HSV (Colour) (B) LBP (Texture). 169

7.7 Comparisons on *Image dataset* from multiple views via different view combination methods. 170

7.8 (a) Discriminative capability of selected subgraphs and the corresponding redundancy, and (b) Discriminative capability for the subgraph updating candidate, respectively. 173

(a) Selected subgraphs and related redundancy 173

(b) Subgraph updating candidate 173

7.9 Average CPU runtime comparison between gCGVFL *v.s.* unbounded gCGVFL (UgCGVFL) with respect to different *min_sup* values on *DBLP* (A) and *Image* (B) dataset, respectively. 174

7.10 Multi-Graph-View Graph-Bag Learning where the object for classification is a bag containing lots of graphs constructed from different graph-views. 177

7.11 The proposed multi-graph-view learning for graph-bag classification (MGVBL). In each iteration, MGVBL selects an optimal subgraph g_* (step a). If the algorithm does not meet the stopping condition, g_* will be added to the subgraph set $\mathbf{g}$ or terminates otherwise (step c). During the loop, MGVBL solves a liner programming to update the weights for training graph-bags and graphs. The weights are continuously updated until obtaining the optimal classifier. 178

7.12 Experimental results for multi-graph-view graph-bag learning on (A) DBLP Text and (B) Corel Image data set. 187

7.13 Average CPU runtime comparison between MGVBL *v.s.* unbounded MGVBL (UMGVBL) with respect to different *min_sup* values. . . 188

List of Tables

4.1 Pairwise *t*-test results of gMGFL v.s. different multi-graph learning algorithms. A, B, C, and D denote proposed gMGFL, gHSIC+MG, IG+MG, and Topk+MG, respectively. 52

(a) *t*-test on DBLP data set 52

(b) *t*-test on NCI data set 52

(c) *t*-test on Online Product data set 52

4.2 Pairwise *t*-test result of gMGFL v.s. different multi-instance learning algorithms. A, B, C, and D denote proposed gMGFL, gHSIC+MI, IG+MI, and Topk+MI, respectively. 57

(a) *t*-test on DBLP data set 57

(b) *t*-test on NCI data set 57

(c) *t*-test on Online Product data set 57

4.3 DBLP dataset used in experiments 74

4.4 NCI Cancer Screen Datasets: NCI(1) and NCI(109). 75

4.5 Pairwise *t*-test result of bMGC *vs.* boosting based MI learning methods. A, B, and C denote bMGC, IG+MI, and Topk+MI, respectively. $\mathcal{H}_1$ and $\mathcal{H}_2$ denote MIBoost and MIOptimalBall, respectively. 79

(a) *t*-test on DBLP (AI vs CV) dataset 79

(b) *t*-test on DBLP (AI vs DB) dataset 79

(c) *t*-test on NCI(1) dataset 79

(d) *t*-test on NCI(109) dataset 79

4.6 Pairwise t-test result of bMGC *vs.* general MI learning methods. A, B, and C denote bMGC, IG+MI, and Topk+MI, respectively. $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ *and* $\mathcal{H}_4$ denote CitationKNN, MIRI, MIEMDD and MISMO, respectively. 84
(a) t-test on DBLP (AI vs CV) dataset 84
(b) t-test on DBLP (AI vs DB) dataset 84
(c) t-test on NCI(1) dataset 84
(d) t-test on NCI(109) dataset 84

5.1 Pairwise t-test result on Online Product Review dataset. A denotes DE-MIMG, with B, C and D denoting MI, MG and MIMG setting. $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ *and* $\mathcal{H}_4$ denote CitationKNN, MIEMDD, MIOptimalBall and MISVM, respectively. 114
(a) Comparison with IG based Approaches 114
(b) Comparison with gSSC based Approaches 114
5.2 Pairwise t-test result on DBLP dataset. A denotes DE-MIMG, with B, C and D denoting MI, MG and MIMG setting, respectively. $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ *and* $\mathcal{H}_4$ denote CitationKNN, MIEMDD, MIOptimalBall and MISVM, respectively. 115
(a) Comparison with IG based Approaches 115
(b) Comparison with gSSC based Approaches 115

7.1 Comparison of gCGVFL *w.r.t.* different r: %. 174