Multinomial Latent Logistic Regression

Zhe Xu

Faculty of Engineering and Information Technology

University of Technology, Sydney

A thesis submitted for the degree of

Doctor of Philosophy

November 2016

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: Zhe Xu
Date: 21/10/2016

I would like to dedicate this thesis to my loving parents
Nan Huang and *Yuan Xu*

Acknowledgements

I would like to take this good opportunity to appreciate my advisors, colleagues, friends and also my family for their significant help during my doctoral study in University of Technology, Sydney.

First of all, I would like to express my sincere appreciation and deep gratitude to my advisor supervisor **Prof. Dacheng Tao** for his unlimited patience, generous support, and supportive guidance. I'm particularly impressed by his incredible enthusiasm and high standard in academic research, which encourage me to breakthrough my own setting limits and submit papers to the leading journals or conferences in my research field.

I would also like to thank my advisor **Prof. Ya Zhang** from Shanghai Jiao Tong University. She always gives me plenty of freedom to explore and timely constructive suggestions to help me out of difficulties. Without the effect by her, Prof. Xiaokang Yang and Prof. Chengqi Zhang, I would not have the opportunity to study here in UTS as a dual-degree PhD student.

I have been fortunate to work in UTS and Centre for Quantum Computation and Intelligent Systems (QCIS) directed by Prof. Chengqi Zhang. QCIS provides full support for me to attend top conferences including ECCV, ICCV and KDD, where I got the opportunities to learn from many world-famous experts. It's really a pleasure for me to work in thus a great team and around so many brilliant minds in QCIS. Studying in QCIS and also UTS will be a fantastic memory that I will never forget.

I am also deeply indebted to Dr. Jun Zhu who led me into the field of computer vision. He shows me how interesting my research objective is, which encourages me constantly during the journey of exploration in the following years. Moreover, I also want to give special thanks to my excellent collaborators: Zhibin Hong and Shaoli Huang for their brilliant work and timely support, and also to my dear colleagues and friends I met in QCIS: Dr. Bozhong Liu, Chunyang Liu, Meng Fang, Tongliang Liu, Mingming Gong, Maoying Qiao, Qiang Li, Runxin Wang, Changxing Ding, Zhiguo Long, Caishi Fang, Wei Bian, Shirui Pan, Yong Luo, Xiao Liu, Dianshuang Wu, Weilong Hou, Chang Xu, Chen Gong, Sujuan Hou, Haishuang Wang, Jia Wu, Wei Yang, Qin Zhang, Yali Du, Hao Xiong, Jiankang Deng, Xiao Liu, Peng Hao, Liu Liu, Guodong Long, Jing Jiang, Peng Zhang, Barbara Munday, Prof. Bo Du, Prof. Xianhua Ben, Prof. Wankou Yang, Prof. Shigang Liu, Prof. Xianhua Zeng, for the inspiring discussions, kind support and companionship.

I am also grateful to all the other friends: Guanbo Huang, Wei Xu, Weiyuan Chen, Xiaohang Ren, Zhiyi Tan, Jialin Li, Qing Wang, Weiyuan Chen, for their support and company during both joyful and stressful times.

Finally, I would like to express my deeply felt gratitude to my parents, who never excoriate me when I make mistakes but show me how to do it the right way, who never asks for anything but give me everything they have, who gives me such a wonderful place to grow up. It's you who make me the man who I am now. Thank you, from the bottom of my heart.

Abstract

We are arriving at the era of big data. The booming of data gives birth to more complicated research objectives, for which it is important to utilize the superior discriminative power brought by explicitly designed feature representations. However, training models based on these features usually requires detailed human annotations, which is being intractable due to the exponential growth of data scale.

A possible solution for this problem is to employ a restricted form of training data, while regarding the others as latent variables and performing latent variable inference during the training process. This solution is termed *weakly supervised learning*, which usually relies on the development of latent variable models. In this dissertation, we propose a novel latent variable model - **multinomial latent logistic regression (MLLR)**, and present a set of applications on utilizing the proposed model on weakly supervised scenarios, which, at the same time, cover multiple practical issues in real-world applications.

We first derive the proposed MLLR in Chapter 3, together with theoretical analysis including the concave and convex property, optimization methods, and the comparison with existing latent variable models on structured outputs. Our key discovery is that by performing "maximization" over latent variables and "averaging" over output labels, MLLR is particularly effective when the latent variables have a large set of possible values or no well-defined graphical structure is existed, and when probabilistic analysis is preferred on the output predictions. Based on it, the following three sections will discuss the application of MLLR in a variety of tasks on weakly supervised learning.

In Chapter 4, we study the application of MLLR on a novel task of architectural style classification. Due to a unique property of this

task that rich inter-class relationships between the recognizing classes make it difficult to describe a building using “hard” assignments of styles, MLLR is believed to be particularly effective due to its ability to produce probabilistic analysis on output predictions in weakly supervised scenarios. Experiments are conducted on a new self-collected dataset, where several interesting discoveries on architectural styles are presented together with the traditional classification task.

In Chapter 5, we study the application of MLLR on an extreme case of weakly supervised learning for fine-grained visual categorization. The core challenge here is that the inter-class variance between subordinate categories is very limited, sometimes even lower than the intra-class variance. On the other hand, due to the non-convex objective function, latent variable models including MLLR are usually very sensitive to the initialization. To conquer these problems, we propose a novel multi-task co-localization strategy to perform warm start for MLLR, which in turn takes advantage of the small inter-class variance between subordinate categories by regarding them as related tasks. Experimental results on several benchmarks demonstrate the effectiveness of the proposed method, achieving comparable results with latest methods with stronger supervision.

In Chapter 6, we aim to further facilitate and scale weakly supervised learning via a novel knowledge transferring strategy, which introduces detailed domain knowledge from sophisticated methods trained on strongly supervised datasets. The proposed strategy is proved to be applicable in a much larger web scale, especially accounting for the ability of performing noise removal with the help of the transferred domain knowledge. A generalized MLLR is proposed to solve this problem using a combination of strongly and weakly supervised training data.

Contents

Contents **vii**

List of Figures **xi**

List of Tables **xvii**

1 Introduction **1**

1.1 Background 1

1.2 Weakly Supervised Learning and Latent Variable Models 2

1.2.1 What is weakly supervised learning? 2

1.2.2 An intuitive example 4

1.2.3 Latent variable models with structured output or multi-class prediction 5

1.2.4 Motivation of the proposed latent variable paradigm 7

1.3 Significance and Organization 8

2 Related Work **11**

2.1 Latent Variable Models with Structured Outputs 11

2.1.1 Hidden conditional random field 12

2.1.2 Latent structural support vector machine 13

2.1.3 Marginal structured support vector machine 14

2.1.4 Latent support vector machine 15

2.1.5 Weak-label structural support vector machine 15

2.1.6 Epsilon-extension model 16

2.1.7 Three-dimensional uncertainty model 17

2.2 Optimization methods 18
2.2.1 Concave-convex procedure 19
2.2.2 Convex optimization solver 20
2.3 Weakly Supervised Learning 21
2.4 Webly Supervised Learning Approaches 23
2.5 Fine-Grained Visual Categorization 24
2.5.1 Feature representation. 25
2.5.2 Model design 25
2.5.3 Training supervision 26

3 Multinomial Latent Logistic Regression 28

3.1 Introduction 28
3.2 Multinomial Latent Logistic Regression 30
3.2.1 Multinomial logistic regression 30
3.2.2 Latent variables 31
3.2.3 Concave-convex procedure 33
3.2.4 Gradient descent 35
3.2.5 Coordinate descent using one-dimensional Newton directions with latent variables 36
3.2.6 Generalization to Structured Outputs 41
3.3 Connection and Difference between MLLR and Existing Methods 42
3.3.1 Maximization vs. marginalization over h 42
3.3.2 Max-margin vs. log-likelihood over y 43
3.3.3 Regularizer 44
3.4 Experiment 45
3.4.1 Handwritten digit recognition 45
3.4.2 PASCAL action classification 48
3.4.3 Sport action recognition 49
3.4.4 Animal classification 50
3.5 Summary 54

4 MLLR for Architectural Style Classification 55

4.1 Introduction 55

4.2 Architectural Style Dataset 58
4.3 Model Description 60
4.3.1 Deformable part-based model 60
4.3.2 Latent SVM 64
4.3.3 DPM-MLLR framework 65
4.4 Experiment 67
4.4.1 Classification task 71
4.4.2 Inter-class relationships between styles 71
4.4.3 Individual building analysis 72
4.5 Summary 73

5 MLLR for Fine-grained Categorization 76
5.1 Introduction 76
5.2 General Initialization Strategies for MLLR 80
5.3 Initialization via Multi-task Co-localization 81
5.3.1 Preliminary 83
5.3.2 Co-localization by discriminative clustering 84
5.3.3 Multi-task discriminative clustering 85
5.3.4 Optimization 86
5.3.5 Fine-grained classifiers 87
5.4 Experiment 88
5.4.1 Dataset and implementation details 88
5.4.2 Localization results 89
5.4.3 Classification results 92
5.5 Summary 98

6 MLLR for Webly Supervised Learning 99
6.1 Introduction 99
6.2 Webly Supervised Learning via Deep Domain Adaptation 102
6.2.1 Preliminary 102
6.2.2 Objective function via generalized MLLR 103
6.2.3 Knowledge extraction on the strongly supervised dataset . 104
6.2.4 Knowledge transfer to the weakly supervised dataset . . . 107

6.3 Experiments . . . 112
6.3.1 Dataset and implementation details . . . 112
6.3.2 Detection results and analysis of discovered part patches . 113
6.3.3 Classification results . . . 115
6.3.4 Visualization . . . 117
6.4 Summary . . . 118

7 Conclusions 121
7.1 Thesis Summarization . . . 121
7.2 Future Work . . . 123

References 125

Publication 146

List of Figures

1.1 Illustration of unsupervised learning, supervised learning, and weakly supervised learning. A variable within a circle indicates given information, while others indicate hidden variables. 3

1.2 Demonstration of the requirement of object-level bounding-box annotations. The models in the right shows template-HOG results learned by DPM [47] using object-level supervision. 4

1.3 The structure of the thesis. 9

2.1 Illustration of latent variable models with structured prediction. The three dimensions are: h standing for latent variables, s standing for strong predictions and y standing for weak labels. The axes represent degree of uncertainty over the three dimensions. 18

3.1 Illustration for the importance of updating latent assignments when performing line search in CDN. Figure (a)(b) shows the situation where no updating process is performed, while (c)(d) shows the situation after updating latent assignments. 39

3.2 L1-norm of the model parameter vectors for different angles learned by MLLR. x-axis stands for angles and y-axis reveals model responses. Although the L1-regularizer is not specified to produce group sparse models, the resulting model parameters follow a similar pattern. 47

3.3 Visualization of the result of MLLR for 3 human actions (*cricket-defensive battling*, *tennis-forehand* and *croquet*). Detected root filters are displayed in red, and part filters are shown in yellow. Note that for the images of the class *croquet*, people usually have strong interactions with the background, which degrades the performance. 49

3.4 Confusion matrices of MLLR in the task of Mammal dataset (a) and Sports dataset (b). 51

3.5 Visualization of the result of distributed MLLR on the mammal dataset. The first column contains the HOG models trained by non-latent linear SVM. Given the non-latent linear SVM model as the initialization status, MLLR models remove some of the noise data in the models, as shown in the second column. The last five columns visualize typical results of the latent position found by MLLR. The rows show three of the object categories, which are *bison*, *elephant* and *giraffe* respectively. 52

3.6 Visualization of how the latent variable (object location) changes during learning. Starting from the full bounding boxes, the algorithm iteratively finds the highest scored location of the object. The numbers underneath indicate the output probabilities of MLLR at various stages. 53

3.7 Visualization of the comparison between MLLR (left) and LSVM (right). The text on the bounding box indicates the prediction label and the number shows its probability. We use the sigmoid function to turn the decision values of LSVM into probabilities. Although both algorithms find the same bounding boxes for the classes “elephant” and “rhino”, MLLR correctly classifies the object due to a better calibration process. 53

4.1 Schematic illustration of architectural style classification using Multinomial Latent Logistic Regression (MLLR). Given a new large-scale architectural style dataset, we model the façade of buildings using deformable part-based models. The resulting classifiers can provide probabilistic analysis along with the standard classification results. 57

4.2 Illustration of the architectural style dataset. Each of the 25 styles is represented by a circle with the respective number in the middle, where different colors indicate broad concepts, such as modern architecture and medieval architecture. The styles are arranged according to time order, where newer ones are placed in the right of ancient ones. Various inter-class relationships exist between the styles, *e.g.*, lines between circles stand for following relationships; smaller circles around large ones indicate sub-categories. Typical images of the styles are shown in the background. Better viewed in color. 59

4.3 Illustration of the feature pyramid in a DPM. Part filters are placed at twice the spatial resolution than the root filter. Original figure can be found in [47]. 61

4.4 Visualization of the use of DPM in architectural style classification. (a)(c)(d) show detection results for different testing images. The trained model for *Gothic* architectural style is shown in (b). . . . 63

4.5 MLLR maps the classifier results of multiple classes to a unified score function. The resultant scores are directly comparable. . . . 68

4.6 Testing results for the ten architectural styles. The first two columns visualize the result root and part filters for each model. From top to bottom: *Baroque, Chicago school, Gothic, Greek Revival, Queen Anne, Romanesque* and *Russian Revival* architecture. Detected root filters are displayed in red, and part filters are shown in yellow. Better viewed in color. 69

4.7 Confusion matrix for MLLR on the two experimental settings. . . 70

4.8 An architectural style relationship map generated by the proposed algorithm. The confusion probability between style A and B is obtained by summing the probabilities with regard to B for all images labeled by A. Only links whose weight exceeds a given threshold are shown in the figure. Modern styles, such as *Postmodern* and *International* style, are connected, while the links between modern and medieval styles are weak. The figure is drawn using NetDraw [16]. 73

4.9 MLLR detects the optimized latent position for each class and outputs a global list of probabilities for each class. (a) Parts shared by different styles. (b) A building that combines several styles. (c)-(f) Typical detection results for the four styles appearing in (a) and (b), i.e., from left to right, *Baroque, Russian Revival, Queen Anne* and *Greek Revival*. 74

5.1 Illustration of the effect of inter-class relationships in fine-grained categorization under weakly supervised settings. In the proposed algorithm, fine-grained categories first act as “friends” in the localization phase against varied backgrounds, then turn back to “foes” in the following classification phase. 79

5.2 Illustration of the difference of discriminative regions for detecting objects from the background and classifying fine-grained categories. For an image of *Red winged Blackbird* in (a), object parts such as forehead, eyes, back and tail make it possible to detect the object, as shown in (c),(d). On the contrary, the subcategory different from other ones mainly from its red wing, resulting in a much smaller region of interest, as shown in (f),(g). 81

5.3 Illustration of the proposed method for weakly supervised fine-grained recognition. The first co-localization phase aims to detect foreground regions from the background. We propose a multi-task algorithm to perform co-localization on multiple subcategories simultaneously. The localization results are then employed to initialize a multi-instance learning process to learn the final object classifiers. 82

5.4 The impact of model parameters μ and λ. In the first figure, λ was fixed as 1, and the second figure set $\mu = 0.1$. 91
5.5 Localization results in different stages. Column 1 to 4: best-scored MCG candidate; results after performing co-localization; results after performing multi-instance classification; best scored bounding box according to SVMs trained using full images. 92
5.6 Example localization results for testing images after performing multi-instance learning. The rightmost column shows cases in which the proposed method failed to classify correctly due to multiple objects, uncommon object pose, and background clutter. . . 94

6.1 Illustration of the proposed semi-supervised method via web data. A strongly supervised dataset is introduced to "teach" web images how to learn properly. 101
6.2 Flowchart of the proposed algorithm. Green lines show modules of strongly supervised method adopted in our framework, while red lines are additional operations of semi-supervised learning. 103
6.3 Detection results on weakly supervised images. Green frames indicate the detected bounding box for part "body". Image labels in the top two rows are correctly classified; the bottom two rows show cases in which classification has failed. Beyond the classification results, part patches in rows 1 and 3 are associated with high detection scores, while rows 2 and 4 have low detection scores. . . 110
6.4 Examples of detected part patches from web images selected as valid training patches. From top to bottom: whole object, head, body. The leftmost five columns show top-scoring detections, while the right two columns show patches with the lowest detection scores.114

6.5 Visualization of the classification process using the proposed method with a root and two parts: head and body. (a) Test image with a ground-truth label of 80. (b) Activation map for the three detectors. (c) Located part bounding boxes. The top 9 nearest neighbours for the detected parts from the training images are shown in (d)-(f). The original strongly supervised method using training data only misclassified the test image into class 81, as shown in (d). Green boxes demonstrate the image patches of label 80, and red boxes for label 81. After re-fine-tuning part-CNNs with the augmented training set, the new feature representations guaranteed that the test image was correctly classified. (e) Nearest neighbours from the strongly supervised training set only using the new feature representations. (f) Results after putting weakly supervised images into the training set either. Yellow boxes indicate images in the weakly supervised dataset with label 80. (g) and (h) show typical training images from class 80 (*Green_Kingfisher*) and 81 (*Pied_Kingfisher*) respectively. 119

List of Tables

2.1 Relationship between latent variable models in the view of unified extension model. We use the same representation form as [107]. . 17

3.1 Prediction accuracies for four digit pairs. N stands for SVM or LR methods without using latent variable models. GD and CDN stand for the two optimization methods in Section 3.2. MLLR-CDN algorithm consistently outperforms two LSSVM-based algorithms. 46

3.2 Average Precision on the PASCAL VOC 2011 Action Classification task. 48

3.3 Classification results for the mammal dataset. Linear SVM is trained without latent variables. All algorithms use the same feature extraction method. We show the mean/std of classification accuracies over 10 rounds of experiments. 52

4.1 Results on the architectural style classification dataset. MLLR consistently outperforms LSVM. Multiple features (*e.g.,* MLLR+SP) are combined by adopting a late fusion method using the softmax function on classifier outputs. 69

5.1 *CorLoc* results for different co-localization strategies on the CUB-14 dataset. MCG denotes the baseline where boxes with the top objectness scores obtained by MCG were adopted without performing co-localization methods. "@n" denoted the best result among top-n candidates. 90

5.2 Localization results for CUB-200-2010, CUB-200-2011 and Stanford Dogs. 93
5.3 A detailed comparison with baselines of different localization strategies and classification methods on the CUB-200-2010 dataset. Row 1-3 show results by training classifiers solely on the detected foreground regions. Row 4-6 show results by performing a multi-instance learning (MIL) approach initialized by the respective localization results. The final row presents an upper bound of our algorithm by using ground-truth bounding box supervision. 95
5.4 Effect of fine-tuning CNNs. We achieved an accuracy of 77.37% on the CUB-200-2011 dataset under the weakly supervised scenario. 96
5.5 Performance comparison to the state-of-the-art results in the literature with or without the use of ground-truth bounding boxes at the training stage. 97

6.1 Part localization accuracy in terms of PCP on the CUB-200-2011 dataset. 114
6.2 CUB-200-2011 Ablation study of different choices of fine-tuning, classifier, detector, and denoising. 115
6.3 Accuracy comparison on the CUB-200-2011 dataset. To conduct fair comparison, we only list methods which use no annotation at testing time; for all the methods, we report their results using the same CNN architecture (AlexNet) if possible. 117
6.4 Accuracy comparison on the Oxford-IIIT Pet Dataset. 118