University of Alberta

Library Release Form

Name of Author: Chi-Hoon Lee

Title of Thesis: Modeling Spatial Correlations for Effective Discriminative Classifiers

Degree: Doctor of Philosophy

Year this Degree Granted: 2009

Chi-Hoon Lee
2250 Homestead Court, #108
Los Altos, CA
USA, 94024

Date: ____________________

University of Alberta

Modeling Spatial Correlations for Effective Discriminative Classifiers

by

Chi-Hoon Lee

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Department of Computing Science

Edmonton, Alberta
Spring 2009

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Modeling Spatial Correlations for Effective Discriminative Classifiers** submitted by Chi-Hoon Lee in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Prof. Russell Greiner

Prof. J. Ross Mitchell

Prof. Ivan Mizera

Prof. Osmar Zaïane

Prof. Dale Schuurmans

Date: ____________________

For my wife, Meejung Cheigh and my son, Samuel JungSoo Lee,
who offered me unconditional love and support throughout the course of this work.
You are my everything.

Abstract

Classification — i.e. categorizing data instances into pre-defined categories — is an interesting and challenging task. Many real world problems involve classification, in domains such as medical informatics, image analysis, and text tagging. We consider the challenge of *learning* a classifier from data. This is especially challenging when data instances are *correlated*.

Here, we focus on learning an image segmenter – e.g. a system that classifies each pixel of a magnetic resonance (MR) image of a brain as either tumor or non-tumor. Here the labels of neighboring pixels are correlated. By contrast, discriminative approaches that assume the data instances are independent and identically distributed (*i.i.d.*), such as Logistic Regression (LR) and Support Vector Machines (SVM), take a single pixel as an input to a fitted decision function and make a decision for that individual pixel that ignores the continuity of labels of neighboring pixels. To be effective here, it is important to also consider the *spatial correlations* of labels: that is, neighboring pixels tend to have same labels. This has led to the now-standard random field approach (eg, Conditional Random Fields, CRFs), which involves learning and using two potential functions: one for estimating relevant characteristics of the individual pixel, and the other that deals with interactions between adjacent pixels.

This dissertation presents extensions to CRFs to address the following three challenging issues: (1) Modeling spatial correlations more *effectively* by using a variant of support vector machines for the random field potential, leading to Support Vector Random Fields (SVRFs). (2) Using both *unlabelled* and *labelled* data in a *supervised* learning framework, leading to Semi-Supervised Discriminative Random Fields (SSDRFs) that produce more accurate model parameters. (3) Modeling spatial correlations more *efficiently*, leading to both Decoupled Conditional Random Fields (DCRFs) that decouple learning of the two potentials of a random field, and Pseudo Conditional Random Fields (PCRFs) that explicitly model spatial correlation only in *inference*.

Our empirical evaluations on complex tasks (such as segmenting brain tumors) show these systems perform statistically significantly better than existing methods and promise wide practical applications.

Acknowledgements

It is great honor and pleasure to get the chance to express my deep gratitude to many people who have influenced my thesis work. First of all, I am very thankful that I met Prof. Russ Greiner as my advisor, who has provided me with the opportunity to work in machine learning. His expertise in machine learning has guided me to face challenging problems and solve many hurdles that seemed like solid walls. I especially thank his infinite patience and commitment, devoting much time to reading my work over and over again, answering my non-sense questions. These all invaluable lessons have greatly motivated my thesis work as well as my attitude towards life, work, and people. Prof. Greiner meant more than an advisor to me.

I sincerely thank Prof. Dale Schuurmans not only for serving on committee, but for his insightful comments over my work. I met him as a student in his class, and his amazingly amusing class gave significant impacts on my knowledge and the way understanding problems. I also truly thank Prof. Osmar Zaïane who was my Master advisor, serving on doctoral committee. Since I met him, he has always shown his strong support on me including his encouragement and introduction to the research worlds. He has been always a big brother to me. Special thanks also go to Prof. Ivan Mizera and Prof. J. Ross Mitchell who served on my dissertation committee. Their constructive feedback and comments have been significantly useful in improving the quality of this dissertation. My thanks are also due to Prof. Ryan Hayward who introduced the University of Alberta to me as a graduate school. He has been devoted to helping students reach academic and professional success. He has been more than a professor to me!

I am also happy to mention the Brain Tumor Analysis Project group and its associated people. Without their help and support, I would have never come up with the most of the original contributions in my thesis. There are number of friends who have enriched my UofA life in various ways. I would like to thank Prof. Shoajun Wang and Dr. Feng Jiao who work on many interesting project together.

It would be an unbounded list to mention all other friends I am indebted to.

I also thank my parents and my sister who have been always there for me supporting my decisions. My parents-in-law deserve warm and special very special thanks for their endless love and care. Especially, I deeply express great gratitude to my parents-in-law who have shown their invaluable encouragement and support. Also, my thanks go to my brother-in-law, Dr. Chan-Ik Cheigh who has paid close attentions to me and my family.

Finally, but most importantly, my extraordinary thanks go to my lovely wife, Meejung Cheigh and my son, Samuel JungSoo Lee. They have sacrificed their precious time to make this happen. Even though our situations were in hard times, she has shown the true affection and dedication she always had towards me and JungSoo. She has been always with me as wife, friend, companion, and even consultant for statistical analysis tasks. Big thanks go to JungSoo who has brought us tons of happiness.

Lastly, I would like to take this great moment to foremost thank God for being my strength and guide for my life. May God bless us all.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

As our society evolves into the information age, we are all being overwhelmed by a tremendous amount of data. Many of our daily activities are recorded as data in computers, which represent experiences of different resources. This motivates a large range of systems designed to deal with such data, including *Machine Learning* algorithms that allow computers to automatically learn from experiences, to improve their performance of some specified tasks. Machine learning algorithms apply many principles to deal with a wide range of applications. One of the primary tasks using machine learning principles is building "classifiers," which categorize novel data instances into pre-defined categories.

Many real world problems involve classification tasks, in domains such as medical informatics (e.g. to diagnose if a patient has a particular form of cancer [45]), image analysis (e.g. to classify if a pixel from an magnetic resonance [MR] image is a tumor [54]), and text tagging (e.g. to find a particular gene name within a given sentence [26]). As many standard classification methods [23, 62] assume *independent* and *identically distributed* (iid) data, they therefore fail to produce high quality classification when the data instances to be categorized are *correlated*– e.g. if dealing with pixels of an image, neighboring pixels are likely to have the same class label. This dissertation focuses on this type of correlation, which we denote as *spatial correlations* [35, 36, 39, 40, 68].

1.1 Motivation

We focus on the image segmentation task. Discriminative approaches that deal with iid data instances, such as Logistic Regression [23, 49] and Support Vector Machines [9, 62], take features of a single pixel as an input into a fitted decision function, whose decision is based only on these single-pixel properties. This is suboptimal for this image segmentation task as it does not incorporate the fact that neighboring pixels tend to have the same class labels.

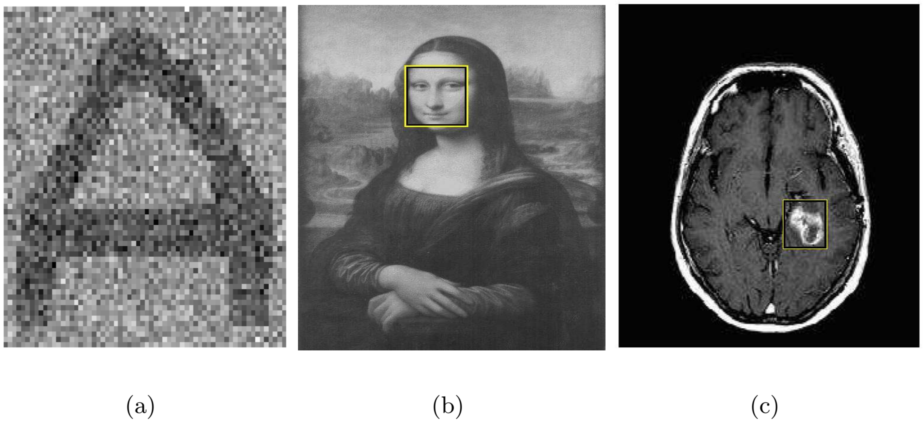
Figure 1.1 illustrates some tasks explicitly requiring spatial correlations. The task here is to classify pixels into pre-defined categories. Figure 1.1(a) makes it clear that simply classifying a pixel one by one, based only on its gray-value intensity, will not produce a high quality classification result since there are some pixels within the 'A' shape whose gray-level intensity is the same as the intensity of pixels in the background. However, by considering the fact that adjacent pixels tend to have the same class labels, a model can encode the spatial correlations of labels, improving the classification accuracy.

Another example is the face detection task: classifying each pixel in an image into either face or non-face (see Figure 1.1(b).) If pixel p is classified as a face (resp. non-face), then pixels around p have a high likelihood of being labelled a face (resp. non-face). If we can effectively model such correlations, we can then improve the quality of the face detection system.

Much of this dissertation is motivated by the third task: segmenting brain tumors. Each pixel in a magnetic resonance (MR) image (Figure 1.1(c)) is examined to determine which is in a tumor. The boxed white blob in the figure locates the only tumor area. As with previous examples, adjacent pixels are highly likely to have the same class labels. All three cases illustrate that modelling spatial dependencies of labels helps produce accurate classification results.

There are many challenges associated with incorporating spatial dependencies. First, it is important to model spatial correlations *accurately*. In the past, many researchers have used Markov Random Fields (MRFs) to model spatial correlations [4, 42]. Although they model spatial correlations of labels, MRFs are *generative* models that attempt to compute the joint probability model of the observations and their associated class labels, which incorporate a prior over class labels. We, however, have a *discriminative* task: to produce a conditional probability model of the class labels *given* the observations. This has motivated researchers to extend generative MRFs to discriminative Conditional Random Fields (CRFs) [35, 37].

Both of MRFs and CRFs have shown robust performance for classification tasks in a 2-D lattice structure, especially compared to iid classifiers. However, the limitations of their models can prevent achieving high accuracy. Most part of this research



(a) (b) (c)

Figure 1.1: An illustration of spatial dependencies among pixels: (a) Pixel Classification task in the presence of noise (b) Face detection task (c) Tumor Segmentation task

provides models that incorporate spatial dependency *effectively*.

Second, supervised learning frameworks typically require $(\mathbf{X}, \mathbf{Y})$ pairs of data instances to train the decision function $f : \mathbf{X} \rightarrow \mathbf{Y}$, where $\mathbf{X}$ denotes a data instance (e.g. an image) and $\mathbf{Y}$ the corresponding a set of labels of pixels in $\mathbf{X}$; we will later use the learned decision function f to classify a new testing instance (e.g. an image that is not observed when fitting f). We typically have lots of $\mathbf{X}$s but relatively few of the associated "ground truths" $\mathbf{Y}$s. The challenge here is to acquire enough true $y_x \in \mathbf{Y}$, where y_x is a ground truth for pixel $x \in \mathbf{X}$. Producing these labels usually involves human experts' judgements – e.g. medical doctors can manually produce such the ground truths for every MR image for the brain tumor segmentation tasks. As this can be very expensive, we often have a great number of MR images whose ground truths are not available. This leads us to explore semi–supervised learning: that is, using both *unlabelled* and *labelled* data when learning a classification model [11, 26]. As another motivation for semi–supervised learning approach, since even thousands of $\mathbf{X}$ examples can only sparsely cover the parameter space, using unlabelled examples may overcome the issue of sparseness.

Third, learning a model that incorporates spatial correlations of labels among adjacent data instances structured in a 2-D lattice increases the computational complexity: typically, CRF-based variants require fitting two sets of parameters, for the two potentials of a random field. Unfortunately, the algorithms for learning these 2-D CRF-variants involve intractable computations. In this thesis, we present two simple but effective frameworks that incorporate spatial correlations, but are relatively *efficient*: Decoupled Conditional Random Fields (DCRFs) and Pseudo Conditional Random Fields(PCRFs).

Here, we present several models to address the challenges of building the classifiers that incorporate spatial correlations. Our empirical experiments , on both synthetic and real data sets, demonstrate that our models are effective and efficient.

1.2 Thesis Outline

Chapter 2 reviews general classification models covering both iid approaches and beyond. This will highlight the general problems of learning iid classification models, and then their extensions to deal with 1-D and 2-D structured classification tasks including various random fields: generative Markov Random Fields and discriminative Conditional Random Fields (for 1-D) and Discriminative Random Fields (DRFs; for 2-D). Note that each of Chapter 4, 5, 6, and 7 also summarizes other related works relevant to that chapter.

Chapter 3 outlines data sets and an accuracy measurement to evaluate models' performance on classification tasks – denoising and brain tumor segmentation. Our motivation to use Jaccard score as a performance measure is highlighted.

Chapter 4 defines an important variant of conditional random fields, Support Vector Random Fields (SVRFs). SVRFs extend CRFs by incorporating Support Vector Machines, which improves their effectiveness. SVRFs address the first challenge previously discussed: how to incorporate spatial dependencies *effectively.* The empirical experiments on pixel classification problems over both synthetic and real data sets demonstrate the effectiveness of the model.

Chapter 5 discusses the challenge of learning a model using both *unlabelled* and *labelled* data, leading to the learning framework of Semi Supervised Discriminative Random Fields (SSDRFs). Our experimental results demonstrate that our SSDRF produces more effective classification results than Discriminative Random Fields (DRFs) based on a supervised learning framework.

For the next two chapters, each provides frameworks that address a computational challenge required when learning typical CRF-based models. Chapter 6 proposes an approximation to model spatial compatibility by decoupling the two potentials of random fields, leading to the De-coupled Conditional Random Fields (DCRFs) model. Experiments on synthetic and real data sets show that DCRFs can be learned efficiently and achieve the accuracy of standard CRF variant model, SVRFs.

Chapter 7 presents another framework that can be efficiently learned, which compactly defines a simple template to encode interactions between neighboring data instances as an alternative to DCRFs. While CRF-based models, including DCRFs, require learning parameter sets for each of *two* potentials, Pseudo Conditional Random Fields (PCRFs) only learn *one* parameter set, for the local conditional probability, but not the one used for modelling spatial correlations. This significantly simplifies the learning procedure. Spatial correlations, however, are considered in inference steps. We present empirical evidences of this model's robust performance over several baseline models: efficient learning producing an accurate classifier.

In Chapter 8, we summarize challenges in modelling spatial compatibility and review the contributions made in this thesis. We also discuss the future extensions and research directions not addressed in the thesis.

1.3 Related Publications

This dissertation extends the following publications:

- Support Vector Random Fields (Chapter 4)

  Chi-Hoon Lee, Russell Greiner, and Mark Schmidt. Support Vector Random Fields for Spatial Classification, *In Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD) (Joint with ECML)*, pp. 121-132, Oct, 2005.

- Semi-Supervised Random Fields (Chapter 5)

  Chi-Hoon Lee, Shaojun Wang, Feng Jiao, Dale Schuurmans, and Russell Greiner. Learning to Model Spatial Dependency: Semi-Supervised Discriminative Random Fields. *In Advances in Neural Information Processing Systems 19 (NIPS)*, pp. 793-800, Cambridge, MA 2007.

- Decoupled Conditional Random Fields (Chapter 6)

  Chi-Hoon Lee, Russell Greiner, and Osmar Zaïane. Efficient Spatial Classification using Decoupled Conditional Random Fields. *In Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD) (Joint with ECML)*, pp. 272-283, Germany Sep. 2006.

- Pseudo Conditional Random Fields (Chapter 7)

  Chi-Hoon Lee, Mattew Brown, Shaojun Wang, Albert Murtha, Russell Greiner. Constrained Classification on Structured Data. *In Proceedings of National Conference on Artificial Intelligence (AAAI)*, pp. 1812-1813, July 2008.

  Chi-Hoon Lee, M. Brown, R. Greiner, S. Wang, A. Murtha. Segmenting Brain Tumors using Pseudo-Conditional Random Fields, *In Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 359-366, Sep. 2008.

1.4 Thesis Statements

This thesis research proposes classification models that encode spatial correlations to support the following claims:

1. It is possible to model spatial correlations of labels *effectively*.
2. It is possible to effectively incorporate *unlabelled* data, as well as *labelled* data, to learn a model that can use spatial correlations.
3. It is possible to learn models that are computationally efficient when we consider spatial correlations.

Chapter 2

Background: iid and non-iid Classifiers

This thesis explores the challenges of learning a classifier to deal with labels that are spatially correlated in a 2-D lattice structure. In this chapter, we briefly review the general classification problem that predicts a class variable $y \in \mathbf{Y}$ given a vector of features $\mathbf{x} = (x_1, ..., x_d) \in \mathbf{X}$, where we focus on $\mathbf{X} = \Re^d$ and $\mathbf{Y}$ is a finite set.

For now, we will view feature vectors as descriptions of observations for iid data instances (later we remove this assumption). In order to perform a classification task for an input $\mathbf{x}$, a classifier, possibly represented as a probability model $p(y|\mathbf{x})$, is learned from a training data set, which is typically in form of a set of n pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$.

There are two approaches for modelling a classifier – *generative* versus *discriminative* [28]. We can use graphical models to illustrate this difference. In general, a graphical model has a set of nodes, each representing a variable, connected with arcs that encode dependencies. N.b. the absence of an arc is used to encode the claim that there are no direct dependencies between a pair of variables. In Figure 2.1, the probabilistic relationships between two nodes are represented with directed edges. Here, the directed arrow between nodes indicates direct conditional dependency. For instance, the edge from node y to node $\mathbf{x}$ in Figure 2.1(a) implies that $\mathbf{x}$ is conditional dependent on y. Therefore, the graphical model that represents relationships (eg, *conditional dependency*) among nodes has a major influence on how a probability model is formulated; the details are discussed in [28].

Section 2.1 and 2.2 introduce several approaches that deal with *independent* and *identically distributed* (iid) data instances. The iid assumption is then extended to 1-D and 2-D structures and related work is discussed in Section 2.3 and 2.4.

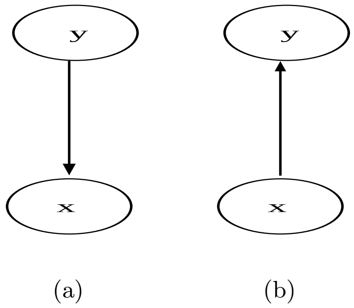


Figure 2.1: (a) Generative approach represented as a graphical model (b) Discriminative approach as a graphical model

2.1 iid Generative Models

Generative approaches, illustrated in Figure 2.1(a), view the probability distribution $p(y|\mathbf{x})$ for classification tasks as estimating a joint probability distribution $p(\mathbf{x}, y)$ [28, 48]. Given training data sets, we estimate two probability distributions: the class conditional probability density (a.k.a. likelihood) $p(\mathbf{x}|y)$ and the prior $p(y)$. These two probability distributions are used to solve the classification task as

$$p(y|\mathbf{x}) \quad = \quad \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})} \quad \propto \quad p(\mathbf{x}|y)p(y)$$

One well-known class of classifiers, based on the generative approach, are the Bayes classifiers [6, 21, 38]. A Bayes classifier is learned from training examples by estimating $p(\mathbf{x}|y)$ and $p(y)$. However, it is not trivial to accurately estimate the likelihood $p(\mathbf{x}|y)$. To see the difficulty in estimating parameters

$$\theta_{ij} \equiv p(\mathbf{x} = \mathbf{x}_i | y = y_j)$$

for the likelihood $p(\mathbf{x}|y)$, suppose $y \in \{+, -\}$ and $\mathbf{x}$ is a feature vector of d binary components. Since $\mathbf{x}_i$ takes on 2^d possible values and y_j takes one of two possible values, in general we may need to estimate 2^d different independent parameters. This requires the learner to observe unrealistically many training examples.

We can drastically reduce the number of parameters, and hence the required training size, if the features are independent. Naïve Bayes dramatically reduce the complexity by making a *conditional independence* assumption when modelling $p(\mathbf{x}|y)$ – specifically that the feature vector components are conditionally independent given a class label y. Therefore, given d components for $\mathbf{x}$, the likelihood is represented

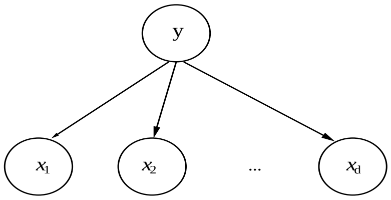


Figure 2.2: Graphical representation of Naïve Bayes

as

$$p(\mathbf{x}|y) \quad = \quad p(x_1, ..., x_d|y) \quad = \quad \prod_{i=1}^{d} p(x_i|y)$$

For example, if $\mathbf{x} = (x_1, x_2)$, then $p(x_1, x_2|y) = p(x_1|y)p(x_2|y)$. Figure 2.2 shows the Naïve Bayes graphical model, which embodies the claim that expresses the nodes – x_1, ..., x_d – are independent given y. That is, there is no edge among the feature components. This conditional independence dramatically reduces the number of independent parameters for $p(\mathbf{x}|y)$ to just $2d$.

Note that even with its unrealistic assumption, Naïve Bayes perform well on many challenging applications including text classification [16] and medical diagnosis [48].

2.2 iid Discriminative Models

As discussed in the previous section, *generative* approaches solve a classification problem by modelling a joint probability distribution over observations and class labels. By contrast, discriminative approaches directly model a conditional probability distribution $p(y|\mathbf{x})$; see Figure 2.1(b). One apparent reason for using discriminative approaches rather than generative is "One should solve the [classification] problem directly and never solve a more general problem as an intermediate step [such as modelling $p(\mathbf{x}|y)$]" [73]. Here, we start discussion with two most popular discriminative techniques – Logistic Regression and Support Vector Machine.

Logistic Regression

Logistic regression is one of the most popular discriminative approaches. The optimal decisions for a class label y for input $\mathbf{x}$ are based on the conditional probability $p(y|\mathbf{x})$. For binary classification, the model has the form

$$\log \frac{P(y = 1 \mid \mathbf{x})}{P(y = 0 \mid \mathbf{x})} \quad = \quad \mathbf{x}^T w \ , \tag{2.1}$$

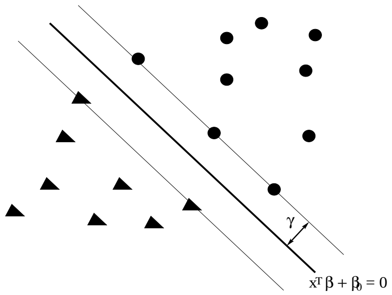


Figure 2.3: Support Vector Machine

parameterized by $w \in \Re^d$. We can re-write Equation (2.1) as a specific form for the conditional probability over labels:

$$P_w(y = 1 \mid \mathbf{x}) \quad = \quad \sigma(\mathbf{x}^T w), \tag{2.2}$$

where the logistic function $\sigma(a) = \frac{1}{1+\exp(-a)}$ turns the linear expression of Equation (2.1) into probabilities in $[0, 1]$. The model parameter w is learned from n training data instances using the maximum conditional log-likelihood criterion:

$$l(w) = \sum_{i=1}^{n} \log P_w(y_i|\mathbf{x}_i) = \sum_{i=1}^{n} \log \sigma(\mathbf{x}_i^T w) \tag{2.3}$$

A learning procedure is formulated as an optimization problem

$$\arg\max_{w} \sum_{i=1}^{n} \log \sigma(\mathbf{x}_i^T w) \tag{2.4}$$

However, as Equation (2.4) often leads to overfitting, many implementations add a regularization term

$$w^* = \arg\max_{w} \left\{ \sum_{i=1}^{n} \log \sigma(\mathbf{x}_i^T w) - \frac{\lambda \|w\|^2}{2} \right\} \tag{2.5}$$

Support Vector Machine

As another discriminative approach, Support Vector Machine (SVM) has been extensively explored for many interesting classification tasks [9, 23, 62]. One of the key concepts in SVM is seeking to find a margin maximizing hyperplane between the classes as a decision function. It in turn produces a signed distance between data instances and the hyperplane (see Figure 2.3).

Using n pairs of training data, $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, the hyperplane is constructed by solving the following optimization problem

$$\begin{aligned} &\max_{\beta,\beta_0,\|\beta\|=1} \gamma \\ &\text{subject to } \; y_i(\mathbf{x}_i^T\beta + \beta_0) \geq \gamma, \qquad i = 1, \ldots, n \end{aligned} \tag{2.6}$$

Since the decision function $f(\mathbf{x})$ is represented with parameters β and β_0 – i.e. $f(\mathbf{x}) = \mathbf{x}^T\beta + \beta_0$, Equation (2.6) can be reformulated as

$$\begin{aligned} &\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 \\ &\text{subject to } \; y_i(\mathbf{x}_i^T\beta + \beta_0) \geq 1, \qquad i = 1, \ldots, n \end{aligned} \tag{2.7}$$

Note that Equation (2.7) is a convex optimization problem, which can be solved by introducing Lagrange multipliers. Using the Lagrange (primal) function, we can obtain the dual optimization problem as

$$\begin{aligned} &\max_\alpha \textstyle\sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_i^n \sum_j^n \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j \\ &\text{subject to } 0 \leq \alpha_i, \quad i = 1, \ldots, n \quad \textstyle\sum_{i=1}^n \alpha_i y_i = 0 \end{aligned} \tag{2.8}$$

From Equation (2.7) and (2.8), we can reconstruct a solution vector $\hat{\beta}$ as a weighted combination of the training examples:

$$\hat{\beta} = \sum_i^n \hat{\alpha}_i y_i \mathbf{x}_i$$

where $\hat{\alpha_i}$ is the solution of Equation (2.8). This shows that the solution vector $\hat{\beta}$ is defined in terms of a linear combination of support vectors $\mathbf{x}_i$ – data instances whose corresponding $\alpha_i > 0$.

One main characteristic in Support Vector Machine is that we need to specify only the inner product (or different kernel) between the data instances – i.e. $\mathbf{x}_i^T\mathbf{x}$. We can use this observations to transform data instances into a higher dimensional space (i.e. feature space) where data instances that are not separable in the input space might be linearly separable. The further details, including ways to deal with class overlap in feature space, can be found in [9, 23, 62]

2.3 Non-iid Generative Models

Most Bayes classifiers, including Naïve Bayes, assume data instances are iid. But consider a POS tagging task, which is the process of assigning a part-of-speech tag

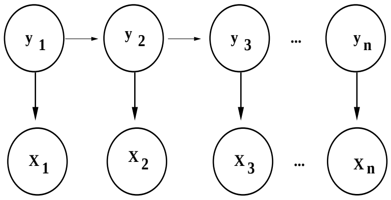


Figure 2.4: Graphical representation of Hidden Markov Model

such as noun, verb, pronoun, preposition, adverb, adjective or other tag to each word in a sentence. Simple Bayes classifiers ignore dependencies of labels among words (ie, pos tags of words in a sentence). For example, in the sentence "I called a travel agent to *book* hotels today.", there are nine words to be tagged (classified). Any iid classifier including Naïve Bayes would probably classify the word "book" as a noun (as that is the most likely interpretation, given only the word), but if we can use the context, then "book" can be classified as a verb by considering the correlations between "to" and "book", and between "book" and "hotels".

A Hidden Markov Model (HMM) – Figure 2.4 – relaxes the independence assumption, and allow correlations between the labels of words in a 1-D chain structure [57]. An HMM models the joint distribution $p(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ ($\mathbf{x}_i$ corresponds to an observation) and $\mathbf{Y} = \{y_i\}_{i=1}^{n}$ (y_i corresponds to a label for observation $\mathbf{x}_i$). An HMM assumes (1) that each class label y_i depends only on the label of its immediate predecessor y_{i-1} (this is the Markovian assumption), and (2) that feature observation $\mathbf{x}_i$ is conditionally independent of everything else, given only its class label y_i. These two assumptions provide an HMM with tractable computations to learn the model as well as to perform classification for non-iid data in 1-D structure. For example, an HMM would view each word (within a sentence) is an instance to be classified. Here, the links from y_{i-1} to y_i allow dependencies between the labels, which are therefore not independent. The joint probability of class labels $\mathbf{Y}$ and observations $\mathbf{X}$ is factorized as

$$p(\mathbf{X}, \mathbf{Y}) = \prod_{i=1}^{n} p(y_i|y_{i-1}) p(\mathbf{x}_i|y_i) \tag{2.9}$$

Named entity recognition, speech recognition, and gene/motif finding tasks are popular examples of HMM applications that require modelling correlations of adjacent labels for sequentially structured data [24, 25, 56, 57].

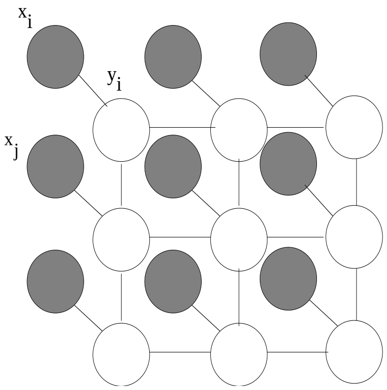


Figure 2.5: Graphical representation of Markov Random Fields. $\mathbf{x}_i$ denotes an observation at pixel i and y_i its class label.

2.3.1 Non-iid for 2-D structures

There has been much related work on using a random field theory to model class dependencies in 2-D structures and more recently discriminative contexts [42, 36]. Here, we will review *Markov Random Fields*, which is formulated as a generative classifier in 2-D structures.

Problem Formulation

Here, we will focus on the task of classifying elements (pixels or regions) of a two-dimensional image, although the discussed methods can be applied to higher-dimensional data. An image is represented with a set S of n pixels. For an instance $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, we seek to infer the most likely joint class labels:

$$\mathbf{Y}^* = (y_1^*, y_2^*, \ldots, y_n^*),$$

where $\mathbf{x} \in \Re^d$, and y_i^* is in a finite set. If we assume that the labels assigned to elements are independent, the following joint probability can be formulated: $P(\mathbf{Y}|\mathbf{X}) = \prod_{i \in S} P(y_i|\mathbf{X})$. However, conditional independency does not hold for 2-D like image data, since spatially adjacent elements are likely to receive the same labels. We therefore need to explicitly consider this local dependency.

Markov Random Fields (MRFs)

Markov Random Fields (MRFs) provide a mathematical formulation for modelling local dependencies, and are defined as follows [42]:

***Definition* 2.3-1.** A set of random variables $\mathbf{Y}$ is called a Markov Random Field on S with respect to a neighborhood N, if and only if the following two conditions are satisfied, where $S - \{i\}$ denotes the set difference, $y_{S-\{i\}}$ denotes random variables in $S-\{i\}$, N_i denotes the neighboring random variables of random variable i, and Ω is the space of all possible joint labellings:

1. $P(\mathbf{Y}) > 0, \forall \mathbf{Y} \in \Omega$

2. $P(y_i | y_{S-\{i\}}) = P(y_i | y_{N_i})$

Condition 2 (Markovianity) states that the conditional distribution of an element y_i is dependent only on its neighbors. Markov Random Fields have traditionally sought to maximize the joint probability $P(\mathbf{X}, \mathbf{Y})$ (a generative approach). In this formulation, the posterior over the labels given the observations is formulated using Bayes' rule as:

$$P(\mathbf{Y}|\mathbf{X}) \propto P(\mathbf{Y})P(\mathbf{X}|\mathbf{Y}) = P(\mathbf{Y}) \prod_{i \in S} P(\mathbf{x}_i | y_i) \tag{2.10}$$

In Equation (2.10), the equivalence between MRFs and Gibbs Distributions [5, 42] provides an efficient way to factor the prior $P(\mathbf{Y})$ over cliques defined in the neighborhood graph G (see Figure 2.5(a).) The prior $P(\mathbf{Y})$ is written as

$$P(\mathbf{Y}) = \frac{\exp(\sum_{c \in C} V_c(\mathbf{Y}))}{\sum_{\mathbf{Y}' \in \Omega} \exp(\sum_{c \in C} V_c(\mathbf{Y}'))} \tag{2.11}$$

where C is a set of cliques in G and $V_c(\mathbf{Y})$ is a clique potential function of labels for clique $c \in C$. From Equation (2.10) and (2.11), the target configuration $\mathbf{Y}^*$ is a realization of a locally dependent Markov Random Field with a specified prior distribution. Based on Equation (2.10) and (2.11) and using Z to denote the (normalizing) "partition function", then the distribution can be factored as:

$$P(\mathbf{Y}|\mathbf{X}) = \frac{1}{Z} \exp \Big[\sum_{i \in S} \log(P(\mathbf{x}_i | y_i)) + \sum_{c \in C} V_c(Y_c) \Big] \tag{2.12}$$

An MRF assumes the factorized likelihood to be Gaussian distributions [77]. The factorized data likelihood for $P(\mathbf{X}|\mathbf{Y})$ in Equation (2.10) allows straightforward Maximum Likelihood parameter estimation. Although there have been many approximation algorithms designed to find the optimal $\mathbf{Y}^*$, we focus on a local method called *Iterated Conditional Modes* (ICM) as it has proven to work effectively [5, 35], written as:

$$y_i^* = \arg\max_{y_i \in L} P(y_i|y_{N_i}, \mathbf{x}_i) \tag{2.13}$$

Assuming observations to be conditionally independent given class labels and a pairwise neighborhood system for the prior over labels

$$P(y_i|y_{N_i}, \mathbf{x}_i) = \frac{1}{Z_i} \exp\Big[\log(P(\mathbf{x}_i|y_i)) + \beta \sum_{j \in N_i} y_i y_j\Big],$$

ICM is formulated as:

$$y_i^* = \arg\max_{y_i \in L} \frac{1}{Z_i} \exp\Big[\log(P(\mathbf{x}_i|y_i)) + \beta \sum_{j \in N_i} y_i y_j\Big] \tag{2.14}$$

where β is a constant and L is a set of class labels.

This concept has proven to be applicable in a wide variety of domains where there are correlations among neighboring instances. However, the generative nature of the model and the assumption that the observations are conditionally independent given class labels in a 2-D structure can be too restrictive to capture complex dependencies between neighboring elements or between observations and labels. In addition, the prior over labels is completely independent from the observations, thus the interactions between neighbors are not proportional to their similarity.

2.4 Non-iid Discriminative Models

The fundamental iid assumption in logistic regression and support vector machine needs to be relaxed to deal with correlations of labels in a 1-D sequence structure. There are two well known "discriminative" approaches – a Maximum-Entropy Markov Model (MEMM) and a Conditional Random Field (CRF) – to model correlations of labels in a 1-D structure [37, 46].

As an alternative to HMMs, an MEMM, shown in Figure 2.6, is able to handle the overlapping features and does not require enumeration of the space of all possible observations [46]. Given an observation $\mathbf{X}$ in 1-D sequence, the conditional probability in an MEMM over label sequence $\mathbf{Y}$ is formulated as

$$P(\mathbf{Y}|\mathbf{X}) \;=\; \prod_{i}^{n} P(y_i|y_{i-1}, \mathbf{x}_i), \tag{2.15}$$

where

$$P(y_i|y_{i-1}, \mathbf{x}_i) = \frac{1}{Z(y_{i-1}, \mathbf{x}_i)} \exp\Big(\sum_k \lambda_k f_k(y_i, \mathbf{x}_i)\Big), \tag{2.16}$$

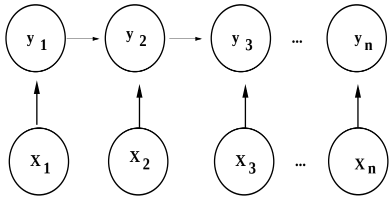


Figure 2.6: Graphical representation of an MEMM

where $Z(y_{i-1}, \mathbf{x}_i) = \sum_{y_i} P(y_i|y_{i-1}, \mathbf{x}_i)$ is the normalizing factor that makes the distribution sum to one across all y_i. Equation (2.16) is derived by the maximum entropy principles that state the best model for data is the one that maximizes the entropy given constraints [3, 46]. Here, the constraints applied are that the expected value $\tilde{E}(f_k)$ for k^{th} feature on the empirical distribution must be equal to its expected value $E(f_k)$ on the learned model distribution – ie. $\tilde{E}(f_k) = E(f_k)$.

Although MEMMs improves over HMMs by utilizing more descriptive feature representations, an MEMM suffers from a weakness called *label bias problem* – the probability transitions leaving any given state[1] must sum to one [37, 46]. This is clearly observed in Equation (2.16) – ie. the normalizing factor.

2.4.1 Non-iid for 2-D structures

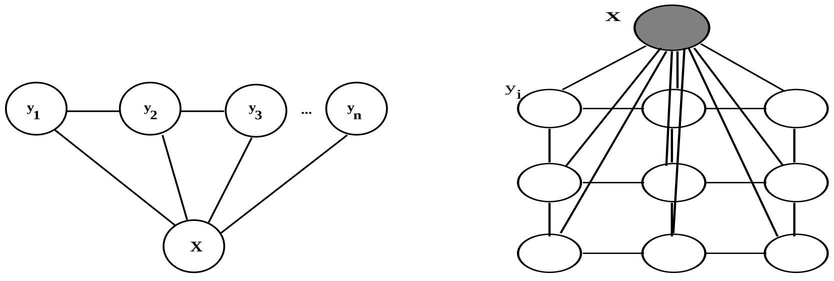
To overcome the disadvantages of HMMs and MEMMs, Lafferty *et al.* [37] proposed a Conditional Random Field (CRFs) as a single exponential model $P(\mathbf{Y}|\mathbf{X})$ of joint probability of entire state sequence $\mathbf{Y}$ given an observation $\mathbf{X}$.

CRFs seek to maximize the conditional probability of the labels given the observations $P(\mathbf{Y}^*|\mathbf{X})$ (a discriminative model), and is defined as follows [37]:

***Definition* 2.4-1.** Let $G = (S, E)$ be a graph such that $\mathbf{Y}$ is indexed by the vertices S of G. Then $(\mathbf{X}, \mathbf{Y})$ is said to be a Conditional Random Field if, when conditioned on $\mathbf{X}$, each random variable y_i obeys the Markov property with respect to the graph: $P(y_i|\mathbf{X}, y_{s\backslash i}) = P(y_i|\mathbf{X}, y_{N_i})$.

This model alleviates the need to model the observations $P(\mathbf{X})$, allowing the use of arbitrary attributes of the observations without explicitly modelling them. As illustrated in Figure 2.7(a), CRFs assume a 1-dimensional chain-structure where only immediate predecessor elements are neighbors. This allows the factorization of the joint probability over labels.

[1] Several states may correspond to a label. However, we assume each state has a single label.



(a)

(b)

Figure 2.7: Graphical representations of Conditional Random Fields in 1-D (a) and 2-D (b).

Discriminative Random Fields (DRFs), extending 1-dimensional CRFs to 2-dimensional structures [35], attempt to overcome the disadvantages of MRFs — notably its conditional independence assumption and the absence of observation in the second potential — by directly modelling the conditional probability distribution $P(\,\mathbf{Y}\,|\,\mathbf{X}\,)$. A CRF, defined as

$$P(\,\mathbf{Y}\,|\,\mathbf{X}\,) = \frac{1}{Z(\mathbf{X})} \exp\left(\sum_{i \in S} \Phi_{\mathbf{w}}(y_i, \mathbf{X}) \;+\; \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{x}) \right) \tag{2.17}$$

directly computes the probability distribution without modelling any prior; see Figure 2.7(b). The notation is essentially the same as in Equation (2.12): $Z(\mathbf{X})$ is the partition function, S is the set of instances, $\mathbf{X} = \{\mathbf{x}_i\}_{i \in S}$ is the set of descriptions of those pixels, and $\mathbf{Y} = \{y_i\}_{i \in S}$ is the set of labels. Here N_i is the set of neighbors of node $\mathbf{x}_i$ — in 2-D, the pixel at location (a, b) has 4 neighbors, at $(a-1, b)$, $(a+1, b)$, $(a, b-1)$ and $(a, b+1)$ [5, 31]; see Figure 2.7(b). In Equation (2.17), "$\Phi_{\mathbf{w}}(y_i, \mathbf{X})$" is called the "Association" potential, which deals with a single instance. While its value can depend on all of $\mathbf{X}$, it typically relies only on $\mathbf{x}_i$, quantifying the belief of $\mathbf{x}_i$ being class y_i. The "$\Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X})$" term is called the "Local-Consistency" (or "Interaction") potential in variants of CRFs; it is typically used to prefer labelling that assign the same class labels to neighboring pixels. (We can view $\Psi_{\boldsymbol{\nu}}(\cdot)$ as a data dependent smoothing function, which differs from MRFs, which instead use only a "data independent" term.) Here, $\mathbf{w}$ and $\boldsymbol{\nu}$ refer to the parameters associated with these potential functions.

Note that this is a much more powerful model than the Gaussian Association potential and the indicator function used as the Interaction potential (that does not consider the observations) in MRFs, avoiding the assumption associated with MRF's likelihood – the conditional independency assumption of observations given labels. (Refer to Equation (2.12).) However, the main drawback in a CRF framework is that it requires significant amount of training time. Sutton *et al.* [65] discusses the computation complexity challenge by proposing a "piecewise training" approach that approximates the computation of $Z(X)$ as an extension of MEMM.

In this chapter, we have reviewed general classification problems from iid to non–iid classification models, including in a 1-D chain structure and a 2-D lattice structure.

Chapter 3

Data Sets and Accuracy measure for Experiments

This chapter presents the data sets – both synthetic and real world – that we use to evaluate our various systems. It also motivates why we use the Jaccard score as the performance measure.

3.1 Synthetic Data sets

Our synthetic data sets are based on binary images (64 by 64 per image), which were corrupted by zero mean Gaussian noise with unit standard deviation. Each ground truth image, shown in the first row from Figure 3.1, contains pixel value 0 or 1 that indicates each pixel's class label – a background or a foreground. We have generated 150 images per each data set (different data sets have different shapes); 150 images are partitioned for training (100 images) and testing (50 images).

The motivations in using these synthetic data sets are (1) to see how *accurately* our models work with the binary image de-noising tasks, where the foreground pixels are corrupted by the synthetically generated noise, and (2) to compare our models with other related work [35, 36, 74] that reported experimental results on data sets generated by the same methods as we described here.

3.2 Real Data sets

We applied our models to the real-world problem of tumor segmentation in medical imaging. We focused on the task of brain tumor segmentation in MRI, an important task in surgical planning and radiation therapy, which is currently being laboriously done by human medical experts.

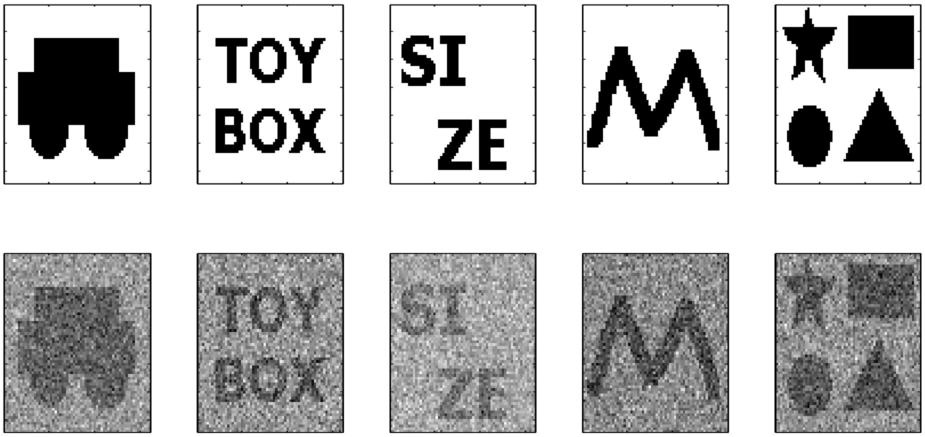
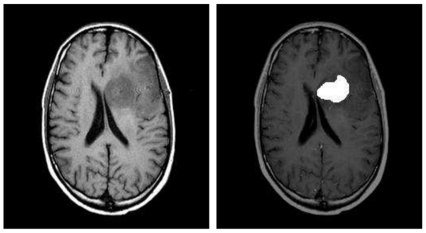


Figure 3.1: Examples on synthetic data sets. Ground truth images (each pixel has 1 or 0 value, indicating a foreground or a background class label) are shown in the first row and randomly corrupted images by $\mathcal{N}(0, 1)$ are displayed in the second row.



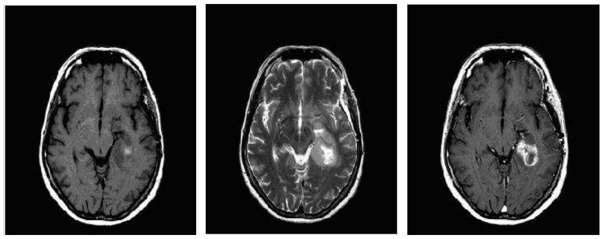
(a) (b)

Figure 3.2: (a) A slice of MR image (b) Its tumor areas (We have changed the brightness of non-tumor areas to highlight tumor areas.)

Here, our primary goal in using real world data is to quantify classification results from models that this dissertation has explored on tumor segmentation task. For instance, given a slice of image (Figure 3.2 (a)), we are interested in finding tumor areas (Figure 3.2 (b)) as effectively as possible.

Our experimental data set consisted of T1, T1c (T1 after injecting contrast agent), and T2 images (each 258 by 258 pixels; Figure 3.3) from patients, each having either a grade 2 astrocytoma, an anaplastic astrocytoma, or a glioblastoma multiforme.

The data were preprocessed with an extensive MR preprocessing pipeline (described in [60], and making use of [47, 63]) to reduce the effects of noise, inter-slice



(a) T1 (b) T2 (c) T1c

Figure 3.3: A multi-spectral MRI

intensity variations, and intensity inhomogeneity. In addition, this pipeline robustly aligns the different modalities with each other, and with a template image in a standard coordinate system (allowing the use of alignment-based features, mentioned below).

We used the most effective feature set identified in the comparative study in [60]. This multi-scale feature set contains traditional image-based features in addition to three types of 'alignment-based' features: spatial probabilities for the 3 normal tissue types (white matter, gray matter and cerebrospinal fluid), spatial expected intensity maps, and a characterization of left-to-right symmetry (all measured at multiple scales).

As with many of the related works[1] on brain tumor segmentation (such as [12, 17, 30, 76]), we employed a patient-specific training scenario, where training data for the classifier is obtained from the patient to be segmented: here we first train on subset P_a of studies for a patient and then test on subset P_b of the same studies for the patient. Note that P_a and P_b are disjoint.

3.3 Accuracy

To quantify the performance of each model, we use the Jaccard score

$$J = \frac{TP}{(TP + FP + FN)}, \tag{3.1}$$

where TP denotes true positives, FP false positives, and FN false negatives. We used this score to penalize the false negatives since many imaging tasks are very

[1] Here, our primary focus is to compare different classifiers' performance (e.g. accuracy and training time of proposed models). Therefore, issues related to MR images such as variations of noise, standardization of MR images, and uncertainty associated with ground truths are beyond of our discussion.

imbalanced: that is, only a small percentage of pixels are in the “positive” class. This allows fair evaluations when a classifier produces high volume of false negatives with very few of false positives. We carry out paired example t-tests to measure the statistical significance of comparisons of performance between algorithms, as widely used in literature [44, 61, 64, 71, 72].

Part I

Models – Effectiveness

Chapter 4

Support Vector Random Fields – SVRFs

4.1 Introduction

The task of classification has traditionally focused on data that is "independent and identically distributed" (iid), in particular assuming that the class labels for different data points are conditionally independent (ie. knowing that one patient has cancer does not mean another one will). However, real-world classification problems often deal with data points whose labels are correlated, which violates the iid assumption. There is extensive literature focusing on the 1-dimensional 'sequential' case (refer to [37]), where correlations in the labels of data points in a linear sequence exist, such as in strings, sequences, and language. This chapter focuses on the more general 'spatial' case, where these correlations exist in data with two-dimensional (or higher-dimensional) structure, such as in images, volumes, graphs, and videos.

Classifiers that make the iid assumption often produce undesirable results when applied to data whose labels are interdependent. For example, in the task of image labelling, such an iid-based classifier could classify a pixel as 'face', even if all adjacent pixels were classified as 'non-face'. As discussed in Chapter 2, this problem motivates the use of Markov Random Fields (MRFs) and more recently Conditional Random Fields (CRFs) for spatial data. These classification techniques augment the performance of an iid classification technique (often a Mixture Model for MRFs, and Logistic Regression for CRFs) by taking into account spatial class dependencies.

Support Vector Machines (SVMs) are classifiers that have appealing theoretical properties [62], and have shown impressive empirical results in a wide variety of tasks. However, this technique makes the critical iid assumption. This chapter propose an extension to CRFs that considers spatial correlations among data instances (as in Random Field models), while still taking advantage of the powerful discriminative properties of SVMs. We refer to this technique as Support Vector Random Fields (SVRFs)

Section 4.2 presents our Support Vector Random Field. Experimental results on synthetic and real data sets are given in Section 4.3, while a summary of our contribution is presented in Section 4.4.

4.2 Support Vector Random Fields (SVRFs)

This section presents Support Vector Random Fields (SVRFs), our extension of a CRF that allows the modelling of non-trivial 2-D (or higher) spatial dependencies using SVMs. As with all random fields, this model has two major components: The *observation-matching* potential function and the *local-consistency* potential function. The *observation-matching* function captures relationships between the observations and the class labels, while the *local-consistency* function models relationships between the labels of neighboring data points and the observations at data points. Since the selection of the observation-matching potential is critical to the performance of the model, the Support Vector Random Field model employs SVMs for this potential, providing a theoretical and empirical advantage over the logistic model used in DRFs and the Gaussian model used in MRFs, which produce unsatisfactory results for many tasks. We formulate the SVRF model as

$$P(\mathbf{Y}|\mathbf{X}) = \frac{1}{Z(\mathbf{X})} \exp\Big\{\sum_{i\in S} \log(O(y_i, \Upsilon_i(\mathbf{X}))) + \sum_{j\in N_i} V(y_i, y_j, \mathbf{X})\Big\}, \qquad (4.1)$$

where $\Upsilon_i(\mathbf{X})$ computes features from the observations $\mathbf{X}$ for location i, $O(y_i, \Upsilon_i(\mathbf{X}))$ is the observation-potential, and $V(y_i, y_j, \mathbf{X})$ is the local-consistency potential. The pair-wise neighborhood system is defined as a local dependency structure. We will now examine these potentials in more detail.

4.2.1 Observation-Matching

The observation-matching potential seeks to find a probability distribution that maps from the observations to corresponding class labels. Note that our observation-matching potential corresponds to $\Phi(.)$ in Equation (2.17). Kumar *et al.* [35] employs a Generalized Linear Models (GLM) for this potential. However, the estimation process in GLMs may not find “satisfactory” parameters that would give accurate results in data whose feature sets may have a high number of dimensions and/or several features have a high degree of correlation (refer to Section 4.3) [58].

Fortunately, the CRF framework allows a flexible choice of the observation-matching potential function. We overcome the disadvantages of the GLM by employing a Support Vector Machine classifier, seeking to find the margin maximizing hyperplane between the classes. This classifier has appealing properties in high-dimensional spaces and is less sensitive to class imbalance [1].

Parameter estimation for SVMs involves optimizing the following Quadratic Programming problem for training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ (where C is a constant that quantifies the misclassification error):

$$\max_\alpha \textstyle\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_i^n \sum_j^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\text{subject to } 0 \leq \alpha_i \leq C \quad \text{and} \quad \textstyle\sum_{i=1}^n \alpha_i y_i = 0 \tag{4.2}$$

Consequently, the decision function of SVMs, given the parameters α_i for the n training instances and bias term b, is $f(\mathbf{x}) = \sum_{i=1}^n (\alpha_i y_i \mathbf{x} \cdot \mathbf{x}_i) + b$. (for a more discussion of SVMs, we refer to Chapter 2.)

Unfortunately, the decision function $f(\mathbf{x})$ produced by SVMs measures distances to the decision boundary, which can be an arbitrary real number. We adopt the approach of [53] to convert the decision function to a probability function scaling values in [0,1]. This is done by using the sigmoid function:

$$O(y_i = 1, \Upsilon_i(\mathbf{X})) = \frac{1}{1 + \exp(B_1 f(\Upsilon_i(\mathbf{X})) + B_0)} \tag{4.3}$$

The parameters B_1 and B_0 are estimated from training data that are represented as pairs $(f(\Upsilon_i(\mathbf{X})), t_i)$, where $f(\cdot)$ is the Support Vector Machine decision function, and t_i denotes a relaxed probability that $y_i = 1$ as in Equation (4.3). We could set $t_i = 1$, if the class label at i is 1 (i.e. $y_i = 1$). However, in order to incorporate the possibility that $\Upsilon_i(\mathbf{X})$ has the opposite class label (ie. -1), we simply define: $t_i = \frac{N_+ + 1}{N_+ + 2}$, if $y_i = 1$, and $t_i = \frac{1}{N_- + 2}$, if $y_i = -1$, where N_+ and N_- are the number of positive and negative class instances. This acts as "regularization" that is applied to data samples, as opposed to parameter regularization, leading to accurate classification results [43, 53].

By producing the new forms of training instances, we can solve the following optimization problem to estimate parameters, substituting $O(y_i = 1, \Upsilon_i(\mathbf{X}))$ with $p(\Upsilon_i(\mathbf{X}))$:

$$\arg\max_{B_0,B_1}\sum_{i=1}^{n}\Big[t_i\log p(\Upsilon_i(\mathbf{X}))+(1-t_i)\log(1-p(\Upsilon_i(\mathbf{X})))\Big] \tag{4.4}$$

4.2.2 Local-Consistency

In MRFs, local-consistency considers correlations between neighboring data points, and is considered to be observation independent: that is, the observation similarity is not incorporated – $\beta y_i y_j$. CRFs provide more powerful modelling of local-consistency by removing the assumption of observation independence. In order to define a local-consistency that corresponds to $\Psi(\cdot)$ in a CRF (refer to Equation (2.17)), we need an approach to express "continuity" of labels between pairwise sites, including "similarity" between observations. For this, we use a linear function of pairwise continuity:

$$V(y_i, y_j, \mathbf{X}) = y_i y_j \nu^T \psi_{ij}(\mathbf{X}), \tag{4.5}$$

$\psi_{ij}(X)$ is a function that computes features for sites i and j based on observations X. While DRFs model the local-consistency by considering the absolute difference between pairwise observations, we propose a new mapping function $\psi(\cdot)$ and let the learning process learn parameter ν that helps to encourage continuity in addition to compensating for errors associated with Observation-matching potential (using $\max(\Upsilon(\mathbf{X}))$ to denote the vector of maximum values for each feature):

$$\psi_{ij}(\mathbf{X}) = \Big(\max(\Upsilon(X)) - \mid \Upsilon_i(\mathbf{X}) - \Upsilon_j(\mathbf{X}) \mid \Big) \cdot\!\diagup \max(\Upsilon(\mathbf{X})) \ , \tag{4.6}$$

where $\cdot\!\diagup$ denotes components wise division.

4.2.3 Learning and Inference

Our proposed model needs to estimate the parameters of the observation-matching function and the local-consistency function. We estimate these parameters sequentially (first parameters of the observation-matching, and then parameters of local-consistency), which has empirically proven to be more effective than the simultaneous learning approach of DRFs.

The parameters of the Support Vector Machine decision function $f(\cdot)$ are first estimated by solving the Quadratic Programming problem in Equation (4.2) (using SVMlight [27]). We then convert the decision function to a probability function using Equation (4.4) and the new training instances – pairs of $(f(\Upsilon_i(\mathbf{X})), t_i)$. Finally, we adopted pseudo-likelihood [35, 42] to estimate the local consistency parameters ν, due to its simplicity and fast computation. For training on n pixels from K images, pseudo-likelihood is formulated as:

$$\widehat{\nu} = \arg\max_{\nu} \prod_{k=1}^{K} \prod_{i=1}^{n} P(y_i^k | y_{N_i}^k, \mathbf{X}^k, \nu) \tag{4.7}$$

As in [35], to ensure that the log-likelihood is convex, ν is assumed as $\mathcal{N}(\nu; \mathbf{0}, \tau^2 \mathbf{I})$, where I is the identity matrix.

We compute the local-consistency parameters using its pseudo-likelihood in log space, $l(\widehat{\nu})$:

$$l(\widehat{\nu}) = \arg\max_{\nu} \sum_{k=1}^{K} \sum_{i=1}^{n} \Big\{ O_i^k + \sum_{j \in N_i} V(y_i^k, y_j^k, \mathbf{X}^k) - \log(z_i^k) \Big\} - \frac{1}{2\tau^2} \nu^T \nu \tag{4.8}$$

Note that we simplified the notation of $O(y_i, \Upsilon_i(\mathbf{X}^k))$ by O_i^k.

In this model, z_i^k is a partition function for each site i in image k, and τ^2 is a regularizing constant. Equation (4.8) is solved by gradient descent – computing its first derivatives, and assuming the observation matching function is a constant during this process.

As this uses the SVM learning procedure, the time complexity of learning for an image with n pixels is $O(n^2)$, although in practice it is much faster.

The inference problem is to infer an optimal labelling $\mathbf{Y}^*$ given a new instance $\mathbf{X}$ and the estimated model parameters. We herein adopted the Iterated Conditional Modes (ICM) approach described in Equation (2.13), which maximizes the local conditional probability iteratively. Although ICM is iterative, it often converges quickly to a high quality configuration, and each iteration has time complexity $O(n)$.

4.3 Experiments

We have evaluated our proposed model on synthetic and real-world binary image labelling tasks (refer to Chapter 3), comparing our approach to Logistic Regression (LR), SVMs, and DRFs for these problems. The accuracy is measured by the Jaccard score introduced in Chapter 3.

4.3.1 Synthetic image sets

As shown in Figure 4.2, two of five data sets contained balanced class labels (*Car* and *Objects*), while the other three contained imbalanced classes. For instance, a

Size image has 826 foreground and 3270 background pixels.

Example results and aggregated scores are shown in Figure 4.2. The last 4 columns from Figure 4.1 illustrate the outcomes from each technique– SVMs, LR, SVRFs, and DRFs.

Logistic Regression and subsequently DRFs performed poorly in all three imbalanced data sets (*Toybox*, *Size*, and *M* shown in Figure 4.1). In these cases, SVMs outperformed these methods and moreover our proposed SVRFs outperformed SVMs. In the first balanced data set (*Car*), DRFs and SVRFs both outperformed SVMs and Logistic Regression at the $p < 0.001$ level on a paired example t-test. However, DRFs performed poorly on the second balanced data set (*Objects*). This is due to DRFs simultaneous parameter learning, which tends to overweight the local-consistency potential. Since the observation-matching is underweighted, edges become degraded during inference. Terminating inference before convergence could reduce this, but this is not desirable for automatic classification. Overall, our Support Vector Random Field model demonstrated the best performance on all data sets, in particular those with imbalanced data and a greater proportion of edge areas.

4.3.2 Brain Tumor Segmentation

There has been significant research focusing on automating challenging task – brain tumor segmentation (see [18]). Markov Random Fields have been explored previously for this task [18], but recently SVMs have shown impressive performance [17, 76]. This represents a scenario where our proposed Support Vector Random Field model could have a major impact. We evaluated the four classifiers from the previous section over seven brain tumor patients. Results for two of the patients are shown in Figure 4.3, while average scores over the seven patients are shown in Figure 4.4. Note that 'SVM+prob' in Figure 4.3 denotes the classification results from the Support Vector Machine probability estimate computed by Equation (4.3). The Logistic Regression model performs poorly at this task, but DRFs perform significantly better. As with the synthetic data in cases of class imbalance, SVMs outperform both Logistic Regression and the DRFs. Finally, SVRFs improve the scores obtained by the SVMs by almost 5% (statistically significant at $p < 0.002$ on a paired example t-test.)

We compared convergence times (inference) of the DRFs and SVRFs by mea-

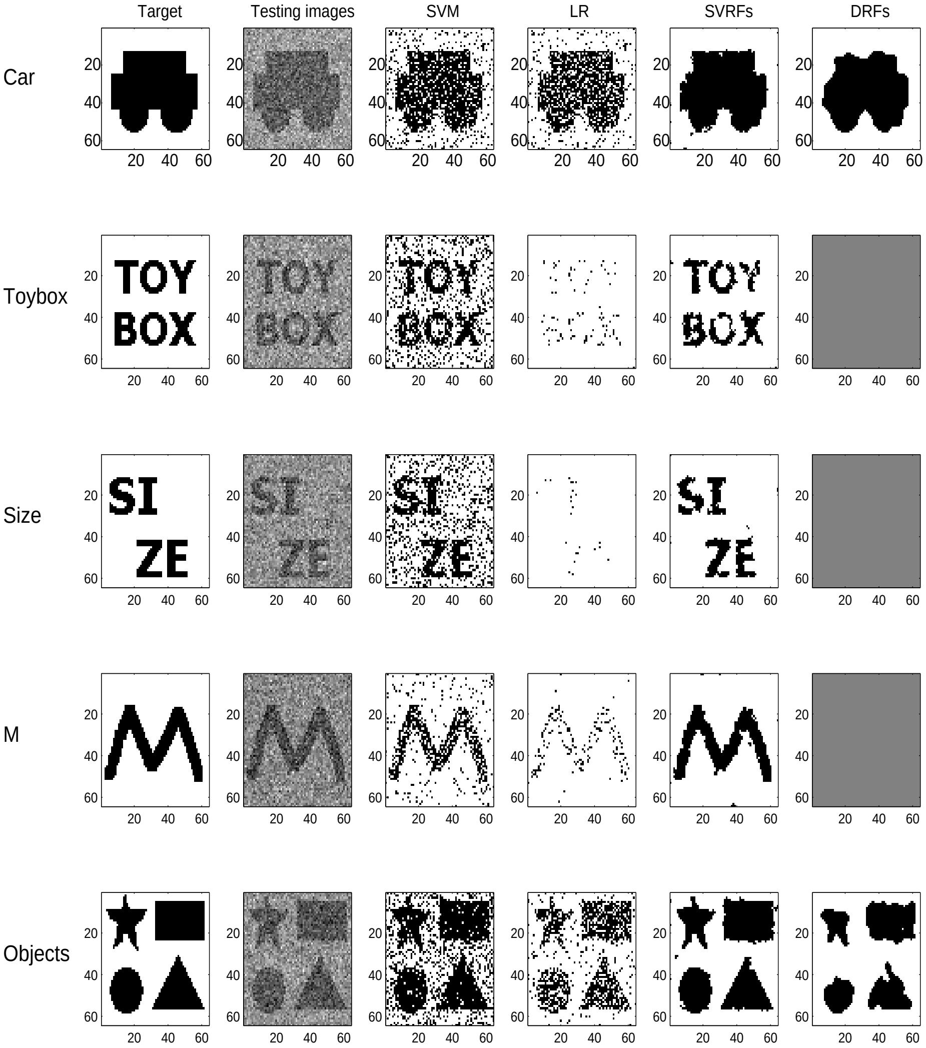


Figure 4.1: Examples on synthetic data sets

suring how many label changes occurred between inference iterations averaged over 21 trials (see Figure 4.5). These results show that DRFs on average require almost 3 times as many iterations to converge, due to the overestimation of the local-consistency potential.

4.4 Conclusion

We have proposed a novel model for classification of data with spatial dependencies. The Support Vector Random Field combines ideas from SVMs and CRFs, and outperforms SVMs and DRFs on both synthetic data sets and an important real-

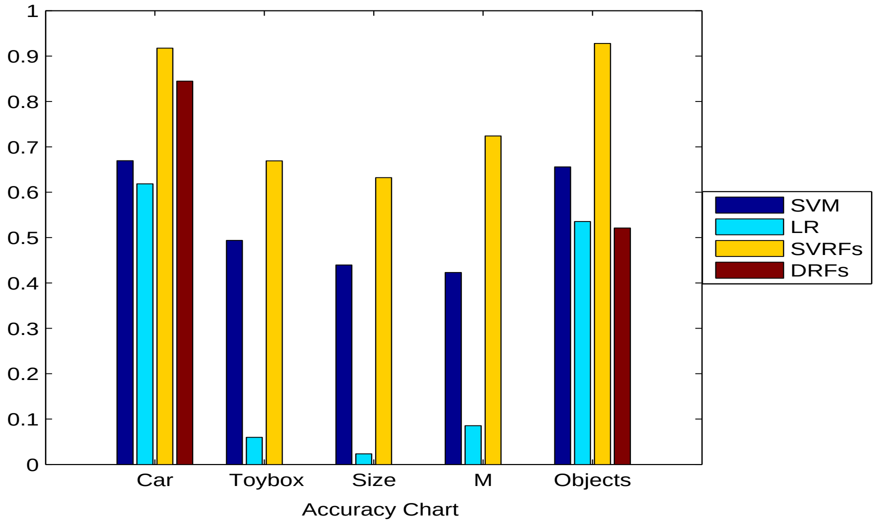
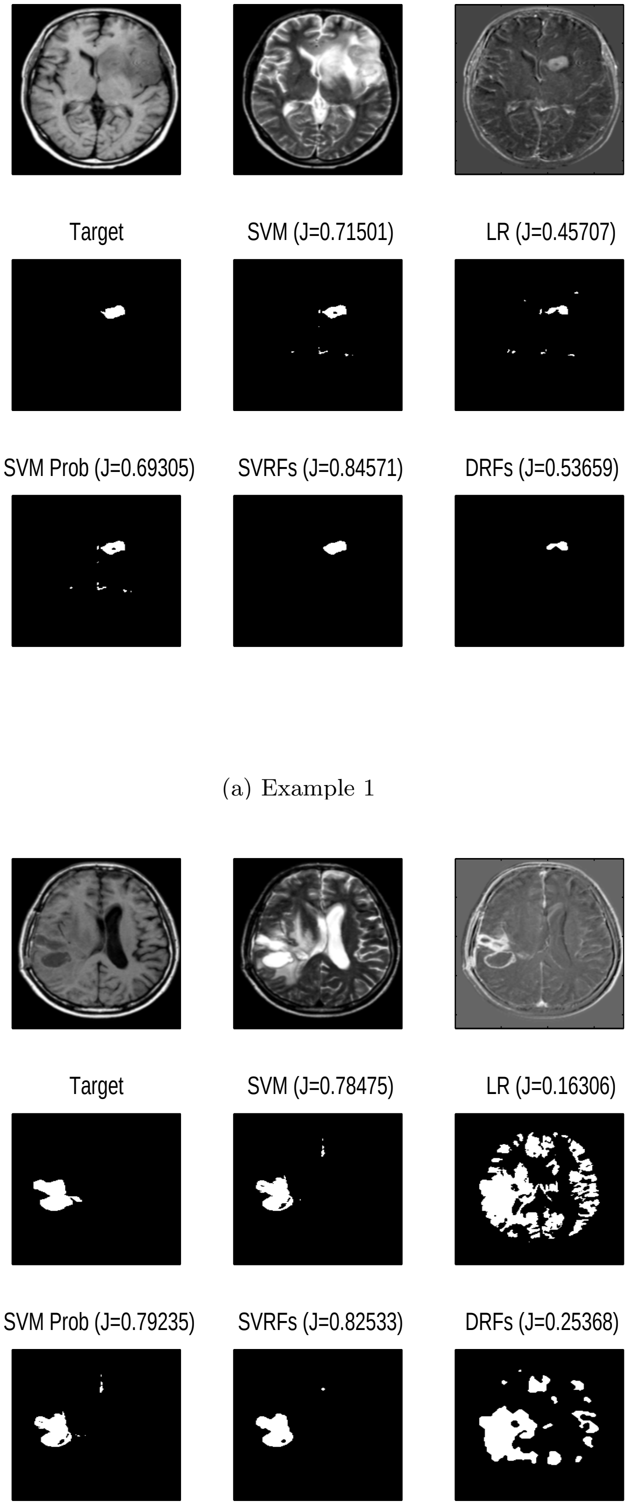


Figure 4.2: Averaged Jaccard scores on synthetic data sets

world application. Our Support Vector Random Field model appears robust to class imbalance, can be efficiently trained, converges quickly during inference, and can trivially be augmented with kernel functions to further improve accuracy.



(a) Example 1

(b) Example 2

Figure 4.3: Examples of the classification result

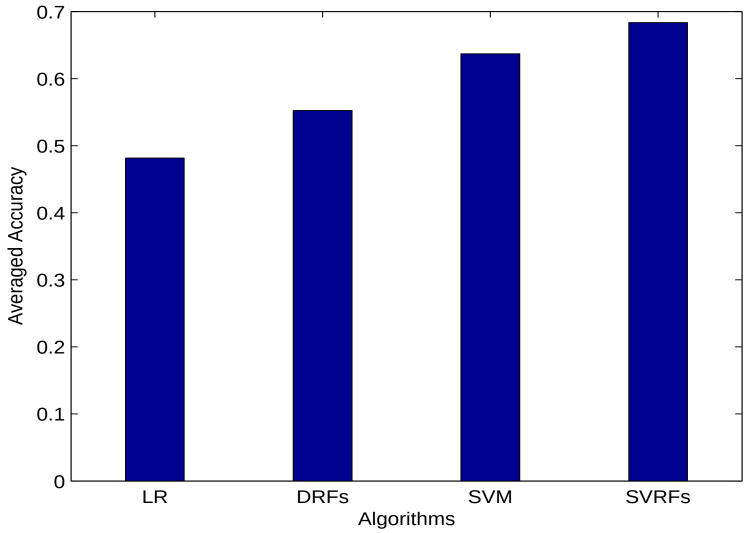


Figure 4.4: Averaged accuracy for MR image analysis. DRFs outperform LR significantly at $p < 0.005$ and SVRFs significantly improve the scores over SVMs at $p < 0.002$.

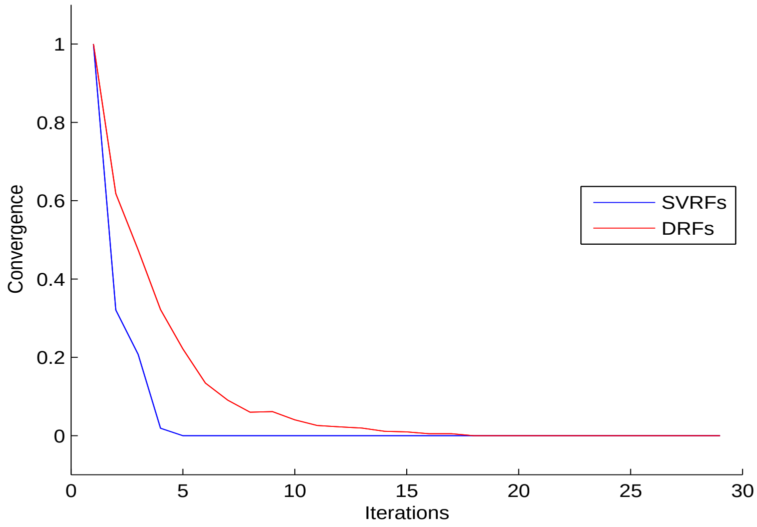


Figure 4.5: Convergence in inference

Chapter 5

Semi Supervised Discriminative Random Fields – SSDRF

5.1 Introduction

As discussed in Chapter 2, random field models are a popular probabilistic framework for representing complex dependencies in natural image data. Discriminative random fields (DRFs) [33, 36] directly model the *conditional* probability over the pixel label field given an observed image. Following the basic tenet of Vapnik [73], it is natural to anticipate that learning an accurate joint model should be more challenging than learning an accurate conditional model. Indeed, recent experimental evidences show that DRFs tend to produce more accurate image labelling models than MRFs do, in many applications like gesture recognition [55] and object detection [33, 36, 74, 69].

Although DRFs tend to produce superior pixel labellings to MRFs, partly by relaxing the assumption of conditional independence of observed images given the labels, the approach relies more heavily on supervised training. DRF training typically uses *labelled* image data where each pixel label has been assigned. However, it is considerably more difficult to obtain labelled data for image analysis than for other classification tasks, such as document classification, since hand-labelling the individual pixels of each image is much harder than assigning class labels to objects like text documents.

Recently, semi-supervised training has become important in many application areas due to the abundance of unlabelled data. Consequently, many researchers are now developing semi-supervised learning techniques for a variety of approaches, including generative models [51], self-learning [10], co-training [7], informationtheoretic regularization [13, 20], and graph-based transduction [78, 79, 80]. However, most of these techniques have been developed for univariate classification problems, or class label classification with a structured input [78, 79, 80]. Unfortunately, semi-supervised learning for structured classification problems, where the prediction variables are interdependent in complex ways, have not been as widely studied.

Current work on semi-supervised learning for structured predictors [2, 26] has focused primarily on simple sequence prediction tasks where learning and inference can be efficiently performed using standard dynamic programming. Unfortunately, the problem we address is more challenging, since the spatial correlations in a 2-D grid structure create numerous dependency cycles. That is, our graphical model structure prevents exact inference from being feasible. Kumar *et al.* [36] and Vishwanathan *et al.* [74] argue that learning a model in the context of approximate inference creates a greater risk of the over-fitting and over estimating.

In this chapter, we extend the work on semi-supervised learning for sequence predictors [2, 26], particularly the DRFs based approach [26], to semi-supervised learning of DRFs. There are several advantages of our approach to semi-supervised DRFs. (1) We inherit the standard advantage of discriminative conditional versus joint model training, while still being able to exploit unlabelled data. (2) The use of unlabelled data enhances our ability to avoid parameter over-fitting and over-estimation in grid based random fields even when using a learner that uses only approximate inference methods. (3) We are still able to model spatial correlations in a 2-D lattice, despite the fact that this introduces dependency cycles in the model. That is, our semi-supervised training procedure can be interpreted as a MAP estimator, where the parameter prior for the model on labelled data is governed by the conditional entropy of the model on unlabelled data. This allows us to learn local potentials that capture spatial correlations while often avoiding local over-estimation. We demonstrate the robustness of our model by applying it to a pixel denoising problem on synthetic images, and also to a challenging real world problem of segmenting tumor in magnetic resonance images. In each case, we have obtained significant improvements over current baselines based on standard DRF training.

5.2 Semi-Supervised DRFs (SSDRFs)

We formulate a new semi-supervised DRF training principle based on the standard supervised formulation of [33, 36]. Let **X** be an observed input image, represented by

$\mathbf{X} = \{\mathbf{x}_i\}_{i \in S}$, where S is a set of the observed image pixels (nodes). Let $\mathbf{Y} = \{y_i\}_{i \in S}$ be the joint set of labels over all pixels of an image. For simplicity we assume each component $y_i \in \mathbf{Y}$ ranges over binary classes $\mathcal{Y} = \{-1, 1\}$. For example, $\mathbf{X}$ might be a magnetic resonance image of a brain and $\mathbf{Y}$ is a realization of a joint labelling over all pixels that indicates whether each pixel is normal or a tumor. In this case, $\mathcal{Y}$ would be the set of pre-defined pixel categories (e.g. tumor versus non-tumor). A DRF is a conditional random field defined on the pixel labels, conditioned on the observation $\mathbf{X}$. More explicitly, the joint distribution over the labels $\mathbf{Y}$ given the observations $\mathbf{X}$ is written

$$p_\theta(\mathbf{Y}|\mathbf{X}) \quad = \quad \frac{1}{Z_\theta(\mathbf{X})} \exp\Big(\sum_{i \in S} \Phi_{\mathbf{w}}(y_i, \mathbf{X}) + \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X})\Big) \tag{5.1}$$

Here N_i denotes the neighboring pixels of i; $\Phi_{\mathbf{w}}(y_i, \mathbf{X}) = \log\Big(\sigma(y_i \mathbf{w}^T \mathbf{h}_i(\mathbf{X})\Big)$ denotes the node potential at pixel i, which quantifies the belief that the class label is y_i for the feature vector $\mathbf{h}_i(\mathbf{X})$, where $\sigma(t) = \frac{1}{1+e^{-t}}$; $\Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X}) = y_i y_j \boldsymbol{\nu}^T \mu_{ij}(\mathbf{X})$ is an edge potential that captures spatial correlations among neighboring pixels (here, the ones at positions i and j), such that $\mu_{ij}(\mathbf{X})$ is the pre-defined feature vector associated with positions i and j from $\mathbf{X}$. $Z_\theta(\mathbf{X})$ is the normalizing factor, also known as a (conditional) partition function, which is

$$Z_\theta(\mathbf{X}) = \sum_{\mathbf{Y}} \exp\Big(\sum_{i \in S} \Phi_{\mathbf{w}}(y_i, \mathbf{X}) + \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X})\Big) \tag{5.2}$$

Finally, $\theta = (\mathbf{w}, \boldsymbol{\nu})$ are the model parameters. When the edge potential $\Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X})$ is set to zero, a DRF yields a standard logistic regression classifier. The potentials in a DRF can use properties of the observed image, and thereby relax the conditional independence assumption of MRFs. Moreover, the edge potentials in a DRF can smooth discontinuities between heterogeneous class pixels, and also correct errors made by the node potentials.

Assume we have a set of independent labelled images $\mathbf{X}$ and their corresponding pixel labels $\mathbf{Y}$, $\mathcal{D}^l = \Big((\mathbf{X}^{(1)}, \mathbf{Y}^{(1)})), \cdots, (\mathbf{X}^{(M)}, \mathbf{Y}^{(M)})\Big)$, and a set of independent unlabelled images, $\mathcal{D}^u = \Big(\mathbf{X}^{(M+1)}, \cdots, \mathbf{X}^{(T)}\Big)$. Our goal is to build a DRF model from the combined set of labelled and unlabelled examples, $\mathcal{D}^l \cup \mathcal{D}^u$.

The standard supervised DRF training procedure is based on maximizing the log of the posterior probability of the labelled examples in $\mathcal{D}^l$

$$CL(\theta) \;=\; \sum_{k=1}^{M} \log P(\mathbf{Y}^{(k)}|\mathbf{X}^{(k)}) - \frac{\boldsymbol{\nu}^T\boldsymbol{\nu}}{2\tau^2} \tag{5.3}$$

We assume a Gaussian prior over the edge parameters $\boldsymbol{\nu}$ and a uniform prior over parameters $\mathbf{w}$. Here $p(\boldsymbol{\nu}) = \mathcal{N}(\boldsymbol{\nu}; \mathbf{0}, \tau^2\mathbf{I})$, where $\mathbf{I}$ is the identity matrix. The hyperparameter τ^2 adds a regularization term. In effect, the Gaussian prior introduces a form of regularization to limit over-fitting on rare features and avoid degeneracy in the case of correlated features.

There are a few issues regarding the supervised learning criteria (5.3). First, the value of τ^2 is critical to the final result, and unfortunately selecting the appropriate τ^2 is a non-trivial task, which in turn makes the learning procedures more challenging and costly [39]. Second, the Gaussian prior is data-independent, and is not associated with either the unlabelled or labelled observations a priori.

Inspired by the work in [20] and [26], we propose a semi-supervised learning algorithm for DRFs that makes full use of the available data by exploiting a form of *entropy regularization* as a prior over the parameters on D^u. Specifically, for a semi-supervised DRF, we attempt to find θ that maximizes the following objective function

$$\begin{aligned} RL(\theta) \;=\; & \sum_{m=1}^{M} \log P_\theta(\mathbf{Y}^{(m)}|\mathbf{X}^{(m)}) + \\ & \gamma \sum_{m=M+1}^{T} \sum_{\mathbf{Y}} P_\theta(\mathbf{Y}|\mathbf{X}^{(\mathbf{m})}) \log \mathbf{P}_\theta(\mathbf{Y}|\mathbf{X}^{(\mathbf{m})}) \end{aligned} \tag{5.4}$$

The first term of (5.4) is the conditional likelihood over the labelled data set $\mathcal{D}^l$, and the second term is a conditional entropy prior over the unlabelled data set $\mathcal{D}^u$, weighted by a tradeoff parameter γ. The resulting estimate is then formulated as a MAP estimate.

The goal of the objective (5.4) is to minimize the uncertainty on possible configurations over parameters. That is, minimizing the conditional entropy over unlabelled instances provides more confidence to the algorithm that the hypothetical labellings for the unlabelled data are consistent with the supervised labels, as greater certainty on the estimated labellings coincides with greater conditional likelihood on the supervised labels, and vice versa. This criterion has been shown to be effective for univariate classification [20], and chain structured CRFs [26]; here we apply it to the 2-D lattice case.

5.3 Parameter Estimation

Several factors constrain the form of training algorithm: Because of overhead and the risk of divergence, it was not practical to employ a Newton method. Although the criticism of the gradient descent's principle is well taken, it is the most practical approach we will adopt to optimize the semi-supervised MAP formulation (5.4) and allows us to improve on standard supervised DRF training.

To formulate a local optimization procedure, we need to compute the gradient of the objective (5.4) with respect to the parameters. Unfortunately, because of the nonlinear mapping function $\sigma(.)$, we are not able to represent the gradient of objective function as compactly as [26], which was able to express the gradient as a product of the covariance matrix of features and the parameter vector θ. Nevertheless, it is straightforward to show that the derivatives of objective function with respect to the node parameters $\mathbf{w}$ is given by [1]

$$
\begin{aligned}
&\frac{\partial}{\partial \mathbf{w}} RL(\theta) = \\
&\sum_{m=1}^{M} \sum_{i \in S^m} \left(y_i^{(m)} \Omega_{\mathbf{w}}(y_i^{(m)}, \mathbf{X}^{(m)}) - \sum_{\mathbf{Y}} p_\theta(\mathbf{Y}|\mathbf{X}^{(m)}) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \right) \mathbf{h}_i(\mathbf{X}^{(m)}) \quad (5.5) \\
&\qquad + \gamma \sum_{m=M+1}^{T} \sum_{i \in S^m} \left(\sum_{\mathbf{Y}} p_\theta(\mathbf{Y}|\mathbf{X}^{(m)}) \Lambda_{\mathbf{w},\boldsymbol{\nu}}(\mathbf{X}, y_i, y_j) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \right. \\
&\qquad\qquad - \Big[\sum_{\mathbf{Y}} p_\theta(\mathbf{Y}|\mathbf{X}^{(m)}) \Lambda_{\mathbf{w},\boldsymbol{\nu}}(\mathbf{X}, y_i, y_j) \Big] \\
&\qquad\qquad \left. \Big[\sum_{\mathbf{Y}} p_\theta(\mathbf{Y}|\mathbf{X}^{(m)}) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \Big] \right) \mathbf{h}_i(\mathbf{X}^{(m)}), \quad (5.6)
\end{aligned}
$$

where

$$
\begin{aligned}
\Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) &= 1 - \sigma(y_i \mathbf{w}^T \mathbf{h}_i(\mathbf{X}^{(m)})), \\
\Lambda_{\mathbf{w},\boldsymbol{\nu}}(\mathbf{X}, y_i, y_j) &= \Big(\Phi_{\mathbf{w}}(y_i, \mathbf{X}) + \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X}) \Big) ,
\end{aligned}
$$

and the terms in (5.5) are the gradient of the supervised component of the DRF over

[1] Note that the derivatives of objective function with respect to the edge parameters $\boldsymbol{\nu}$ are computed analogously.

labelled data, and the second terms are the gradient of conditional entropy prior of the DRF over unlabelled data.

It is intractable to compute the conditional partition function $Z_\theta(\mathbf{X})$. Therefore, as in standard supervised DRFs, we need to incorporate some form of approximation. Following [5, 33, 36], we incorporate the pseudo-likelihood approximation, which assumes that the joint conditional distribution can be approximated as a product of the local posterior probabilities given the neighboring nodes and the observation

$$\begin{aligned} p_\theta(\mathbf{Y}|\mathbf{X}) &\approx \prod_{i \in S} p_\theta(y_i|y_{N_i}, \mathbf{X}) && (5.7) \\ p_\theta(y_i|y_{N_i}, \mathbf{X}) &= \frac{1}{z_i(\mathbf{X})} \exp\Big(\Phi_{\mathbf{w}}(y_i, \mathbf{X}) + \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X})\Big) && (5.8) \end{aligned}$$

Using the factored approximation in (5.8), we can reformulate the training objective as

$$\begin{aligned} RL^{PL}(\theta) &= \sum_{m=1}^{M} \sum_{i \in S^m} \log p_\theta(\mathbf{Y}_i^{(m)}|\mathbf{Y}_{N_i}^{(m)}, \mathbf{X}^{(m)}) && (5.9) \\ &\quad + \gamma \sum_{m=M+1}^{T} \sum_{i \in S^m} \sum_{y_i} p_\theta(y_i|y_{N_i}, \mathbf{X}^{(m)}) \log p_\theta(y_i|y_{N_i}\mathbf{X}^{(m)}) \end{aligned}$$

Here, the derivative of the second term in (5.9), with respect to the potential parameters $\mathbf{w}$ and $\boldsymbol{\nu}$, can be reformulated as a factored conditional entropy, yielding

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{w}} RL^{PL}(\theta) = && (5.10) \\ &\sum_{m=1}^{M} \sum_{i \in S^m} \left(y_i^{(m)} \Omega_{\mathbf{w}}(y_i^{(m)}, \mathbf{X}^{(m)}) - \sum_{y_i} p_\theta(y_i|y_{N_i}, \mathbf{X}^{(m)}) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \right) \mathbf{h}_i(\mathbf{X}^{(m)}) \\ &+\gamma \sum_{m=M+1}^{T} \sum_{i \in S^m} \left(\sum_{y_i} p_\theta(y_i|y_{N_i}, \mathbf{X}^{(m)}) \Lambda_{\mathbf{w},\boldsymbol{\nu}}(\mathbf{X}, y_i, y_j) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \right. \\ &\qquad - \Big[\sum_{y_i} p_\theta(y_i|y_{N_i}\mathbf{X}^{(m)}) \Lambda_{\mathbf{w},\boldsymbol{\nu}}(\mathbf{X}, y_i, y_j) \Big] \\ &\qquad \left. \Big[\sum_{y_i} p_\theta(y_i|y_{N_i}, \mathbf{X}^{(m)}) y_i \Omega_{\mathbf{w}}(y_i, \mathbf{X}^{(m)}) \Big] \right) \mathbf{h}_i(\mathbf{X}^{(m)}), \end{aligned}$$

Assuming the factorization, the true conditional entropy and feature expectations can be computed in terms of local conditional distributions. This allows us efficiently to approximate the global conditional entropy over unlabelled data. Note that there may be an over-smoothing issue associated with the pseudo-likelihood approximation, as mentioned in [36, 74]. However, due to the fast and stable performance of this approximation in the supervised case [5, 36] we still employ it, but below show that the over-smoothing effect is mitigated by our data-dependent prior in the MAP objective (5.4).

5.4 Inference

As a result of our formulation, the learning method is tightly coupled with the inference steps. That is, for the unlabelled data, $\mathbf{X}_U$, each time we compute Equation (5.10), we perform inference steps for each node i and its neighboring nodes N_i. Our inference is based on iterative conditional modes (ICM), and is given by Equation (2.14).

We could alternatively compute the marginal conditional probability $P(y_i|\mathbf{X}) = \sum_{y_{S\setminus i}} P(y_i, y_{S\setminus i}|X)$ for each node using the sum-product algorithm (i.e. loopy belief propagation), which iteratively propagates the belief of each node to its neighbors. Clearly, there are a range of approximation methods available including Globerson *et al.* [19] that approximates computations of marginal conditional probability, each entailing different accuracy-complexity tradeoffs. However, we have found that ICM yields good performance at our tasks below, and is probably one of the simplest possible alternatives.

5.5 Experiments

In this section, we present a series of experiments on synthetic and real data sets using our novel semi-supervised DRFs (SSDRFs). In order to evaluate our model, we compare the results with those using maximum likelihood estimation (MLE) of supervised DRFs [33]. We consider the standard MLE DRF from [33], instead of the parameter regularized DRFs from [36], as we want to compare different performance of "learned parameters" from the MLE and MAP [36, 39]. That is, proper regularization helps find "good" parameters achieving accurate classification results.

To quantify the performance of each model, we used the Jaccard score as defined in Chapter 3.

The tradeoff parameter, γ, was hand-tuned and then held fixed at 0.2 for all the experiments.

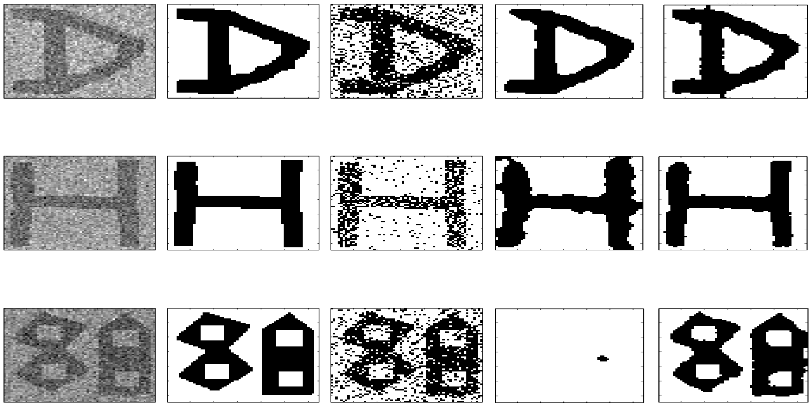
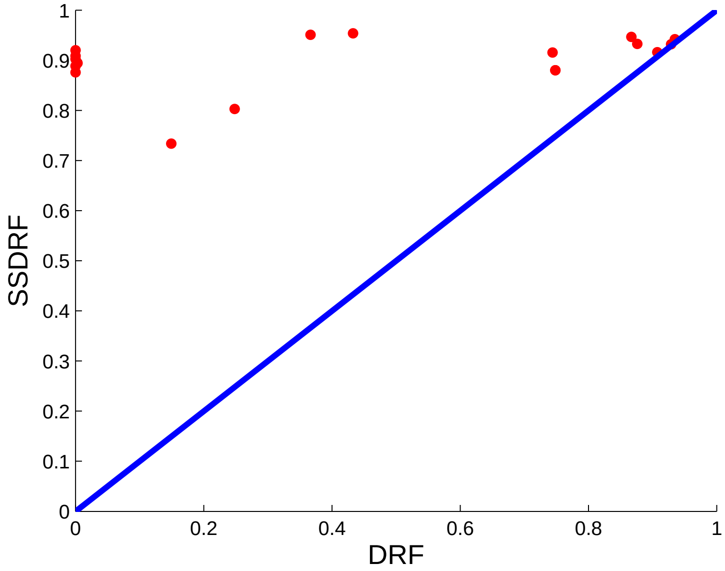


Figure 5.1: Sample outputs from synthetic data sets. From left to right: Testing instance, Ground Truth, Logistic Regression (LR), DRF, and SSDRF

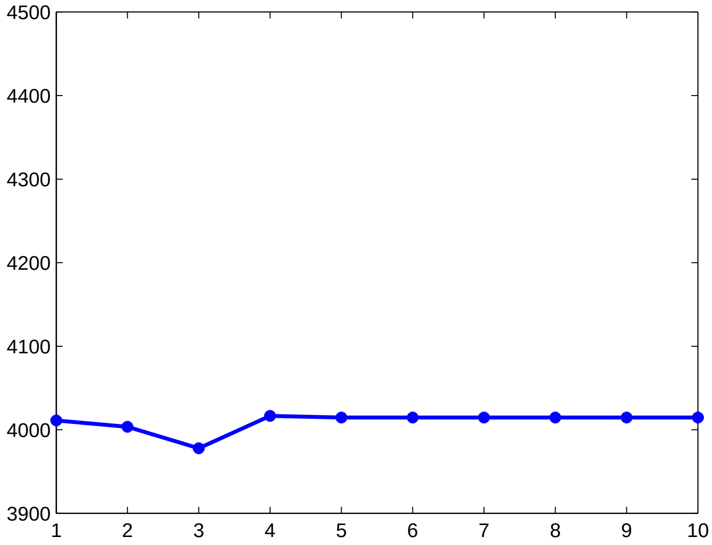
5.5.1 Synthetic image sets

To see if our semi-supervised learning approach learns model parameters that achieve good quality of classification results, we used 18 synthetic data sets, each with its own shape (refer to Chapter 3). Figure 5.1 shows the results of using supervised DRFs, as well as semi-supervised DRFs. Kumar *et al.* and Vishwananthan *et al.* [36, 74] reported over-smoothing effects from the local approximation approach of pseudo-likelihood (PL) while our experiments indicate that the over-smoothing is caused not only by PL approximation, but also by the sensitivity of the regularization to the parameters. However, using our semi-supervised DRF as a MAP formulation, we have dramatically improved the performance over standard supervised DRF.

Note that the first row in Figure 5.1 shows good results from the standard DRF, while the oversmoothed outputs are presented in the last row. Although the ML approach may learn proper parameters from some of data sets, unfortunately its performance has not been consistent since the standard DRF's learning of the edge potential tends to be overestimated. For instance, the last row shows that overestimated parameters of the DRF segment almost all pixels into a class due to the complicated edges and structures containing non-target area within the target area. Our semi-supervised DRF performance is, however, not degraded at all. Overall, by learning more statistics from unlabelled data, our model dominates the standard DRF in most cases. This is because our MAP formulation avoids the overestimate of potentials and uses the edge potential to correct the errors made by the node potential. Figure 5.2(a) shows the results over 18 synthetic data sets. Each point



(a) Jaccard scores from DRF and SSDRF for all 18 synthetic data sets



(b) Log likelihood values (Y axis) for a testing image by increasing ratio (X axis) of unlabelled instances for SSDRF

Figure 5.2: Accuracy and Convergency

Table 5.1: Jaccard scores for D^U

Studies	Testing from D^U		
	LR	DRF	SSDRF
p_1	53.84	59.81	59.81
p_2	83.24	83.65	84.67
p_3	30.72	30.17	75.76
p_4	72.04	76.16	79.02
p_5	73.26	73.59	75.25
p_6	88.39	89.61	87.01
p_7	69.33	69.91	75.60
p_8	58.49	58.89	73.03
p_9	60.85	56.49	83.91
Average	65.57	66.48	**77.12**

Table 5.2: Jaccard scores for D^S

Studies	Testing from D^S		
	LR	DRF	SSDRF
p_1	68.01	68.75	68.75
p_2	69.61	69.73	70.06
p_3	23.11	21.90	71.13
p_4	56.52	63.07	68.40
p_5	51.38	52.36	51.29
p_6	85.65	86.35	85.43
p_7	66.71	68.68	70.27
p_8	44.92	45.36	73.09
p_9	21.11	20.16	38.06
Average	54.11	55.15	**66.27**

above the diagonal line in Figure 5.2(a) indicates SSDRF producing significantly higher Jaccard scores for a data set at $p < 0.001$ level on a paired example t-test.

Note that our learning approach shows stable convergence as we increased the ratio (nU/nL) of unlabelled data sets in our learning, as in Figure 5.2(b), where nU denotes the number of unlabelled images and nL the number of labelled images. This implies that model parameters tend to be insensitive to the increment of the number of unlabelled data while fixing nL. Similar results have also been reported in simple single variable classification task [20].

5.5.2 Brain Tumor Segmentation

We applied three models to the classification of nine studies from brain tumor MR images. For each study, i, we partitioned the MR images into three disjoint sets: D_i^L, D_i^U, and D_i^S, where D_i^L denotes labelled, D_i^U unlabelled, and D_i^S testing data sets.

Per study i, LR and DRF take D_i^L as the training set, and test on D_i^U and D_i^S. Our SSDRFs is trained with labelled and unlabelled data: that is, D_i^L and D_i^U. The tests are performed on D_i^U and D_i^S. Note that even though the ground truths of D_i^U are available, they are not considered during training steps.

We segmented the "enhancing" tumor area, the region that appears hyper-intense after injecting the contrast agent (we also included non-enhancing areas contained within the enhancing contour). Table 5.1 and 5.2 present Jaccard scores of testing D_i^U and D_i^S for each study, p_i, respectively. While the standard supervised DRF improves over its degenerate model LR by 1%, semi-supervised DRF signifi-

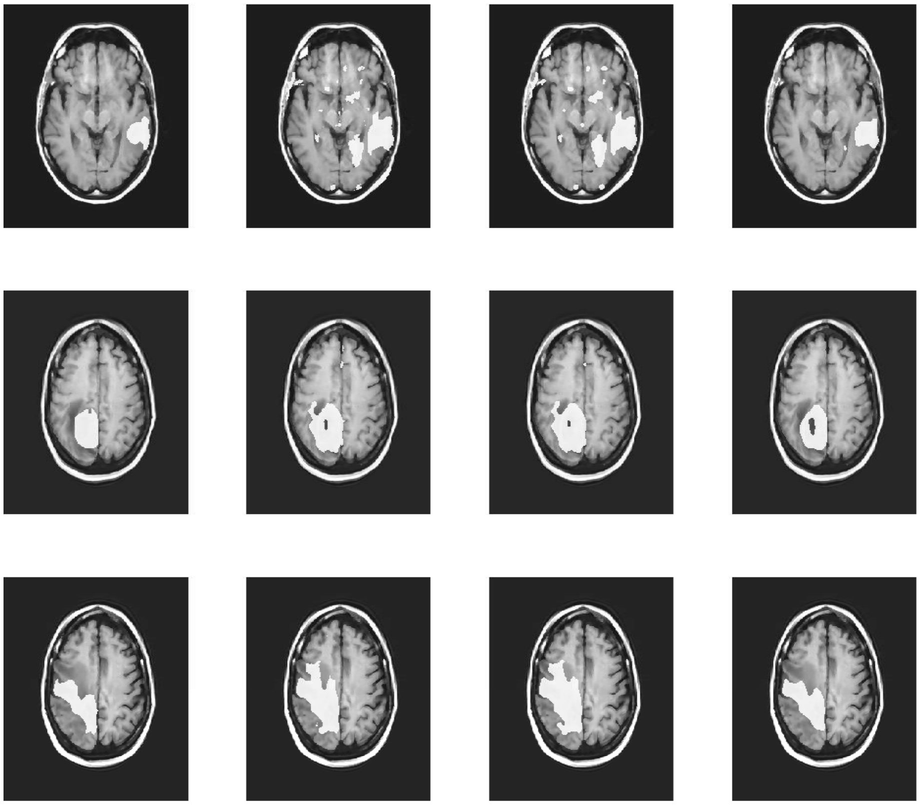


Figure 5.3: From Left to Right: Human Expert, LR, DRF, and SSDRF

cantly improves over the supervised DRF by 11%, which is significant at p <0.006 using a paired example *t test*. Considering the fact that MR images contain much noise and the three modalities are not consistent among slices of the same patient, our improvement is considerable. Figure 5.3 shows the segmentation results by overlaying the testing slices with segmented outputs from the three models. Each row demonstrates the segmentation for a slice, where the white blob areas for the slice correspond to the enhancing tumor area.

5.6 Conclusion

We have proposed a new semi-supervised learning algorithm for DRFs, which was formulated as *MAP* estimation with conditional entropy over unlabelled data as a data-dependent prior regularization. We introduced a simple approximation approach for this new learning procedure that exploits the local conditional probability to efficiently compute the derivative of the objective function.

We have applied this new approach to the problem of image pixel classification tasks. By exploiting the auxiliary unlabelled data, we are able to improve the performance of the state of the art supervised DRF approach. Our semi-supervised DRF approach shares all of the benefits of the standard DRF training, including the ability to exploit arbitrary potentials in the presence of dependency cycles, while improving accuracy through the use of the unlabelled data.

The main drawbacks of our SSDRF (in comparison with DRFs) are (1) the increased training time involved in computing the derivative of the conditional entropy over unlabelled data and (2) the challenge in selecting an appropriate γ. Nevertheless, the algorithm is efficiently trained on unlabelled data sets, and obtains a significant improvement in classification accuracy over standard supervised training of DRFs as well as the iid logistic regression classifier. To further accelerate the performance with respect to accuracy, we may apply loopy belief propagation [75] or graph-cuts [8] as an inference tool. Since our model is tightly coupled with inference steps during the learning, the proper choice of an inference algorithm will most likely improve segmentation tasks.

Part II

Models – Efficiency

Chapter 6

De-coupled Conditional Random Fields – DCRFs

6.1 Introduction

There are a number of random field approaches for classification tasks of spatially correlated data instances, including generative models like Markov Random Field (MRF) [31, 42], as well as discriminative models, including Conditional Random Field (CRF) [37] and its variants – Discriminative Random Field (DRF) [35], Associative Markov Nets (AMN) [68] , and our recent Support Vector Random Field (SVRF) [40]. As MRFs assume conditional independence among observations given class labels, their learning procedures tend to be faster than the discriminative models (variants of CRFs); however, this assumption means they are not as accurate. The more accurate models, unfortunately, can be prohibitively slow, which may not be tolerable to classification tasks such as image segmentations.

In this chapter, we propose a novel approach to our discriminative random fields model to make it more efficient. We develop a "decoupled" learner, DCRF to avoid the expense of learning parameters in the framework of random fields. We found that, as expected, the resulting DCRF is much faster to train than the corresponding (non-decoupled) SVRFs. Moreover, we were pleasantly surprised to find that this improvement in speed did not cost a degradation in accuracy: that is, our DCRF is essentially as accurate as SVRFs!

Section 6.2 presents a quick overview of related systems. It motivates our approach by noting that these systems – especially the ones that produce accurate labelling – can be very slow to train. Section 6.3 introduces our novel "Decoupled Conditional Random Field" (DCRF) approach, and provides details for both learning the parameters and for inference (i.e. classification — here segmentation). Section 6.4 demonstrates the accuracy and efficiency of our model by presenting experimental results over various domains, including the challenging real-world problem of segmenting brain tumor from MRI scans.

6.2 Related Work

In the MRF framework, the probability over the n joint labels $\mathbf{Y}$ given the observations $\mathbf{X}$ is written as

$$P(\,\mathbf{Y}\,|\,\mathbf{X}\,) \;\propto\; P(\,\mathbf{Y}\,)\,P(\,\mathbf{X}\,|\,\mathbf{Y}\,) \;=\; P(\,\mathbf{Y}\,)\prod_{i}^{n} P(\,\mathbf{x}_i\,|\,y_i\,)$$

As the factorization of the likelihood is only a crude approximation to reality, this approach will typically produce inferior labels. The prior $P(\,\mathbf{Y}\,)$ can explicitly incorporate dependencies among the labels. Due to the equivalence between MRFs and Gibbs Distributions [5], an MRF is formulated as

$$P(\,\mathbf{Y}\,|\,\mathbf{X}\,) \propto \frac{1}{Z(\mathbf{X})}\exp\left(\sum_{i\in S} D(\mathbf{x}_i, y_i) + \sum_{j\in N_i} V(y_i, y_j)\right), \tag{6.1}$$

where S is the set of nodes (i.e. pixels), $V(y_i, y_j)$ is a potential function of labels, y_i and y_j, N_i is a set of neighbors of node i, and the "partition function" $Z(\mathbf{X}) = \sum_{\mathbf{Y}} \exp\left[\sum_{i\in S} D(\mathbf{x}_i, y_i) + \sum_{j\in N_i} V(y_i, y_j)\right]$ is used to normalize the equation.

Notice $V(y_i, y_j)$ depends only on the labels y_i and y_j, but not on the information about the pixels $\{\mathbf{x}_i\}_{i\in S}$. Therefore, an MRF prefers a set of labels $\mathbf{Y}^*$ where neighbors have the same value. Also, as the partition function $Z(\mathbf{X})$ involves summing over all $|L|^{|S|}$ possible labellings (assuming there are $|L|$ labels for each pixel), it is very expensive to compute. However, an MRF assumes $D(\mathbf{x}_i, y_i)$ to be Gaussian distributions, and hence estimating maximum likelihood parameters is computationally efficient [4, 29, 35, 77].

CRFs have been extended to two well-defined models that differ by their choice of Association potentials: Discriminative Random Fields (DRFs) [35], which use Logistic Regression, and Support Vector Random Fields (SVRFs) [40], which use Support Vector Machines (SVM) [9]. Note that CRF variants produce better accuracy than their generative alternative, MRFs. However, their good performance compromises the efficiency in learning steps.

For example, the learning task in DRFs and SVRFs involves estimating the parameters $\mathbf{w}$ and $\boldsymbol{\nu}$ that maximize the log-likelihood of the given data sample. Both systems use a regularization term to avoid overfitting. The parameters are estimated by maximizing the log-likelihood for M images formulated as

$$\langle\, \hat{\mathbf{w}}, \hat{\boldsymbol{\nu}}\,\rangle = \operatorname*{argmax}_{\mathbf{w},\nu} \left(\sum_{k=1}^{M} \sum_{i \in S} \Phi_w(y_i^{(k)}, \mathbf{X}^{(k)}) + \sum_{j \in N_i} \Psi_{\boldsymbol{\nu}}(y_i^{(k)}, y_j^{(k)}, \mathbf{X}^{(k)}) - \log(Z^{(k)}(\mathbf{X})) \right) - \frac{\boldsymbol{\nu}^T \boldsymbol{\nu}}{2\tau^2} \tag{6.2}$$

Although SVRF significantly improves the accuracy of DRF even when features may be correlated, SVRF has shown that selecting the appropriate τ^2 in SVRF and DRF is a non-trivial task, which makes the overall learning procedures more challenging and costly. Coordination Classifiers [22], an ensemble classifier, expresses the spatial correlations by synthetically creating "neighborhoods" among iid data instances. Its performance depends on how the neighborhoods are determined. Associative Markov Nets (AMN) [68], a variant of Max-Margin Markov Nets [67], discriminatively train Markov nets. AMNs exploit the spatial correlations by adopting the maximum-margin principle of maximizing the margin between target labels and the best runner-up label assignments. Hence, this process employs the same ideas underlying SVM. SVRFs differ by actually performing the same basic computations that an SVM performs. Note that a Boosted Random Field (BRF) [70] combines a set of iid classifiers that correspond to Association potentials, where each potential is trained on a specific class to quantify the likelihood of a class on a pixel. Hence, a BRF does not explicitly consider the spatial correlation. We see there are problems in training each of the systems mentioned in this section: some are inaccurate (as they use inappropriate models), while others require significant amount of computation time, or user inputs.

6.3 The DCRF System

This section presents the foundations to formalize our Decoupled Conditional Random Field, DCRF. We first motivate our approach of decoupling the training of the two potentials, then discuss inference — i.e. how to use the resulting system to segment an image.

First, if we ignore the dependencies among the labels of the pixels (i.e. assume that they are independent and identically distributed), we would use only the "Association" potential, which attempts to maximize

$$P_A(\,\mathbf{Y}\,|\,\mathbf{X}\,) \quad \propto \quad \exp\left(\sum_{i\in S}\Phi(y_i,\mathbf{X})\right) \tag{6.3}$$

Many existing classifiers (e.g. Naïve Bayes, Logistic Regressions, SVM, etc.) are (perhaps implicitly) attempting to optimize Equation (6.3).

Alternatively, a discriminative model that only considers spatial coherence would attempt to optimize

$$P_{LC}(\,\mathbf{Y}\,|\,\mathbf{X}\,) \quad \propto \quad \exp\left(\sum_{i\in S}\Psi(y_i,y_{N_i},\mathbf{X})\right) \tag{6.4}$$

where y_{N_i} are the labels of i's neighbors.

Equation (6.3) and (6.4) provide different frameworks for approximating the probability distributions $P(\,\mathbf{Y}\,|\,\mathbf{X}\,)$. Each is only partial, in that the first (second) does not properly incorporate spatial coherence (resp., the local observations).

Notice typical CRF models involve the sum of these equations — written in log space as

$$\sum_{i\in S}\Phi(y_i,\mathbf{x}) \quad + \quad \sum_{i\in S}\Psi(y_i,y_{N_i},\mathbf{X}) \tag{6.5}$$

(Compare to Equation (2.17). Note that the neighborhood is considered in $\Psi(\cdot)$ explicitly.)

We now observe that the potentials forms in Equation (6.5) follows MAP formulations for the joint probability over labels: that is, we can approximate the global optimal joint class labels by maximizing the local probability distribution using the principles of pseudo-likelihood and Iterative Conditional Modes (ICM)[1] [9] — i.e. $P(\,\mathbf{Y}\,|\,\mathbf{X}\,) = \prod_{i\in S}P(\,y_i\,|\,y_{N_i},\mathbf{X}\,)$. Thus, for each pixel i, the formulation to model $P(\,y_i\,|\,y_{N_i},\mathbf{X}\,)$ given its neighbors y_{N_i} is:

$$\Phi_{\mathbf{w}}(y_i,\mathbf{x}) \;+\; \sum_{j\in N_i}\Psi_{\boldsymbol{\nu}}(y_i,y_j,\mathbf{X}) \tag{6.6}$$

N.b., as we will only be seeking the argmax, we can safely omit the normalizing "$-\log(z_i)$" term from Equation (6.2), as it will be constant here.

[1] Although pseudo-likelihood and ICM principles are only guaranteed to achieve local maxima, the discussion of the global optimality issues is beyond the scope of this chapter.

Equation (6.6) shows that we can approximate a CRF model using a decoupled system, corresponding to the simple sum of two different potentials. (This differs from standard ensemble methods [14], as we are directly combining *potentials* rather than classifiers.) We will see that, as expected, it is much faster to learn these individual summands *individually*, before combining them. Our empirical evidence shows that, surprisingly, the resulting DCRF system can be as accurate!

6.3.1 Association-only Potential

The association potential provides a local likelihood being class label y_i for feature characteristics $\mathbf{x}_i$ to describe pixel i: $P_A(\, y_i \,|\, \mathbf{x}_i \,)$. Our "decoupling" principle allows us to select a function that quantifies the conditional probability for a given observed instance. We incorporate a maximal margin approach where the two classes of pixels are classified based on a hyperplane that maximizing the distances between the two classes.

As suggested above, we consider a potential based on SVMs; note this method inherits the SVM's relative insensitivity to class imbalance, and their ability to typically outperform other discriminative classifiers such as GLMs, especially in cases where the classes overlap [62], which is common case in imaging applications.

We find a decision function $f(\mathbf{x})$ by solving the optimization problem as in Equation (2.8) over the α_is, and produce $f(\mathbf{x}) \;=\; \sum_{i=1}^{n} \alpha_i \, y_i \, \mathbf{x}_i^T \, \mathbf{x} \;+\; \beta_0$ then use the decision function $sign(\, f(\mathbf{x})\,)$ to classify a test instance $\mathbf{x}$. Our implementation actually uses Sequential Minimal Optimization (SMO), which is even more efficient than standard SVM implementations [52].

Notice that $f(\mathbf{x})$ computes the distance to the hyperplane from the instance $\mathbf{x}$. We can use this to compute a sigmoid function [53, 43]:[2]

$$\Phi_{\mathbf{w}}(y_i, \mathbf{X}) \;=\; \frac{1}{1 + \exp(A_A \times y_i(\mathbf{w}^T \mathbf{x}_i) + B_A)} \qquad (6.7)$$

using the parameters A_A and B_A.

As noted above, our approach (like SVRFs) differs from Max-Margin Markov Nets (M^{3}N) [67] and AMN [68] as those system explicitly maximize a margin between the target labels and most probable label assignments considering joint labels.

[2] We augment the instance $\mathbf{x}_i$ by including a constant 1, and hence the $\mathbf{w}$ include a "constant" term as well.

6.3.2 Local-Consistency-only Potential

We use our "local-consistency-only" potential to model the "neighborhood coherence" between pixels. Its goal is to encourage instances within the specified neighborhood system to have the same labels when their feature characteristics are similar, and therefore is mainly to smooth regions (and hence remove errors) produced by the Association-only potential.

For similar instances in a neighborhood to have similar (in our discrete case, "identical") class labels, we introduce a max-margin based potential, which tries to make the labels of a testing instance same as the labels of its neighbors. This potential learns a pairwise max-margin model that quantifies the likelihood that two pixels will have the same class labels, given their descriptions:

$$\Psi_{\boldsymbol{\nu}}(y_i, y_j, \mathbf{X}) \;=\; I(y_i, y_j) \times [\nu^T \langle\, \psi(\mathbf{x}_i, \mathbf{x}_j), 1\,\rangle] \tag{6.8}$$

where $I(y_i, y_j)$ returns $+1$ if $y_i = y_j$, and -1 otherwise. (We define $\psi(\mathbf{x}_i, \mathbf{x}_j)$ below.) Equation (6.8) reduces the pairwise discriminative learning problem to the binary class problem, over similar versus dissimilar classes. That is, we apply Quadratic Programming (QP) (refer to Equation (2.8)) to the training set

$$S_{new} = \quad \{\ (\psi(\mathbf{x}_r, \mathbf{x}_j),\ I(y_r, y_j)) \mid j \in N_r\ \}$$

over all instances r with neighbors $j \in N_r$, to find the optimal parameter $\boldsymbol{\nu}$.

Note that each pair of pixels is projected by $\psi(\cdot)$ onto a similarity feature space. In this chapter, we use $\psi(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$, which produces a scalar: the cosine measure of the similarity. Note this attains its largest value when the two vectors match one another. Due to "localized" neighborhood system for the Local-consistency potential, the increment to the training data size only grows linearly with the number of pixels. Notice that feature-wise space depends on $\psi(\cdot)$.

As we will need to combine this potential with the Association-only one, we need to produce values within a "comparable" range. We therefore convert Equation (6.8) to the probability scale, using the same transformation used to produce Equation (6.7).

$$\Psi(y_i, y_j, \mathbf{X}) = \frac{1}{1 + \exp\left(A_{LC} \times I(y_i, y_j)(\boldsymbol{\nu}^T \langle\, \psi(\mathbf{x}_i, \mathbf{x}_j), 1\,\rangle) + B_{LC}\right)} \tag{6.9}$$

where again A_{LC} and B_{LC} are set to optimize the fit to a sigmoid, which produces a probability distribution as in Association-only potential.

6.3.3 Inference

Our goal in producing this DCRF system is then to find relevant regions within images — e.g. tumor regions within MR images of a brain. This involves inferring a binary label (tumor versus non-tumor) for each individual pixel. As noted above, this corresponds to computing the most likely vector $\mathbf{Y}^* = \mathrm{argmax}_{\mathbf{y}}\, P(\,\mathbf{Y} \,|\, \mathbf{X}\,)$ given the evidence $\mathbf{X}$, based on the (possibly unnormalized) potential functions. In our case, we will use the potential function in Equation (6.6), which is the sum of the Association-only $P_A(\,\cdot\,)$ (Equation (6.3)) and Local-Consistency-Only $P_{LC}(\,\cdot\,)$ (Equation (6.4)) potentials. While the inference seeking $\mathbf{Y}^*$ can be expensive, there are several existing approximation algorithms for CRFs, including Iterative Conditional Modes (ICM) [9], Graph-Cuts (GC) [8], and Loopy Belief Propagation (LBP) [28].

DCRF uses ICM since it converges quickly and has been shown empirically to produce accurate results [40, 5].[3] ICM iteratively maximizes the local conditional probabilities, assuming the other labels are correct:

$$\begin{aligned} y_i^* &= \underset{y_i \in \{+1,\,-1\}}{\mathrm{argmax}}\; P(\,y_i \,|\, y_{N_i}, \mathbf{X}\,) \\ &= \underset{y_i \in \{+1,\;-1\}}{\mathrm{argmax}}\; \Phi(y_i, \mathbf{X}) \;+\; \Psi(y_i, y_{N_i}, \mathbf{X}) \end{aligned} \tag{6.10}$$

Of course, we could add the normalization factor z_i in Equation (6.10), which constrains outputs to follow probability axioms. However, the constant factor is irrelevant, since our inference approach seeks only the most likely value.

Our DCRF model uses QP within SMO. Assuming each image has n pixels, and each pixel has E neighbors then learning the Association-only potential requires $O(n^2)$ steps per image, and Local-Consistency-only potential requires $O(\,(n \times E)^2\,)$ per image. Here, we used E is 4. Inference (here, classifying the regions in a test image) requires $O(n)$ per iteration. Empirically, we found that ICM converged after 5 iterations, on average.

[3] While GC and LBP are considered be the best inference methods, even if the graph structure has loops, we used ICM for the reasons shown above. Note this issue is orthogonal to the goal of this chapter, which is to compare the training time and accuracy of our DCRF to other CRF-related models.

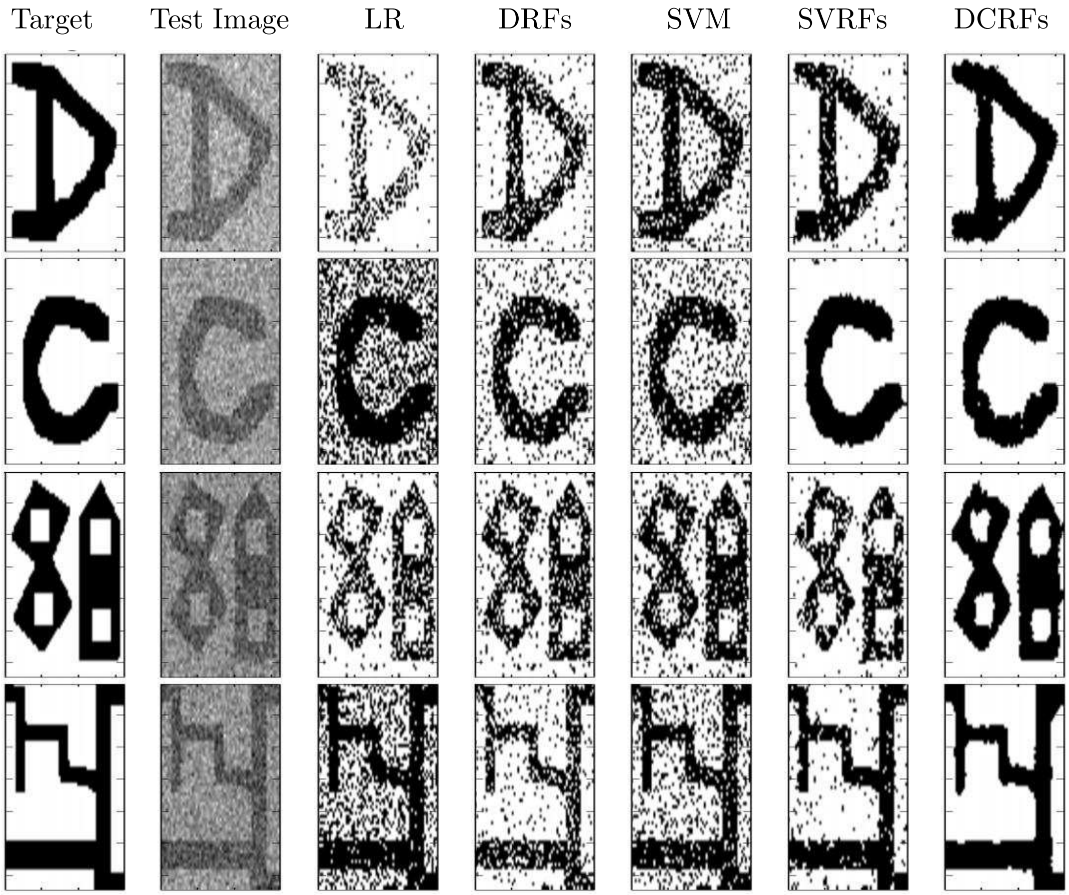


Figure 6.1: Results from synthetic image sets. Left to right: Target, Test Image, LR, DRFs, SVM, SVRFs, and DCRFs. Rows 1 to 5 from the top down correspond respectively to datasets 1, 3, 10, and 11 in Figure. 6.2

6.4 Experiments

We implemented the Decoupled CRFs described above, DCRFs, and compared it with other random field techniques on both synthetic and real-world tasks. As many imaging tasks are very imbalanced (in that the “positive” class includes only a small percentage of the pixels), the standard evaluation criteria of “accuracy” is problematic. We therefore use the Jaccard score. The details about data sets and Jaccard score are discussed in Chapter 3.

6.4.1 Synthetic image sets

We first apply our DCRFs to artificially generated images where foreground and background pixels are significantly corrupted by noises. This in turn provides us with an opportunity – how our approach relaxes classification results based on iid

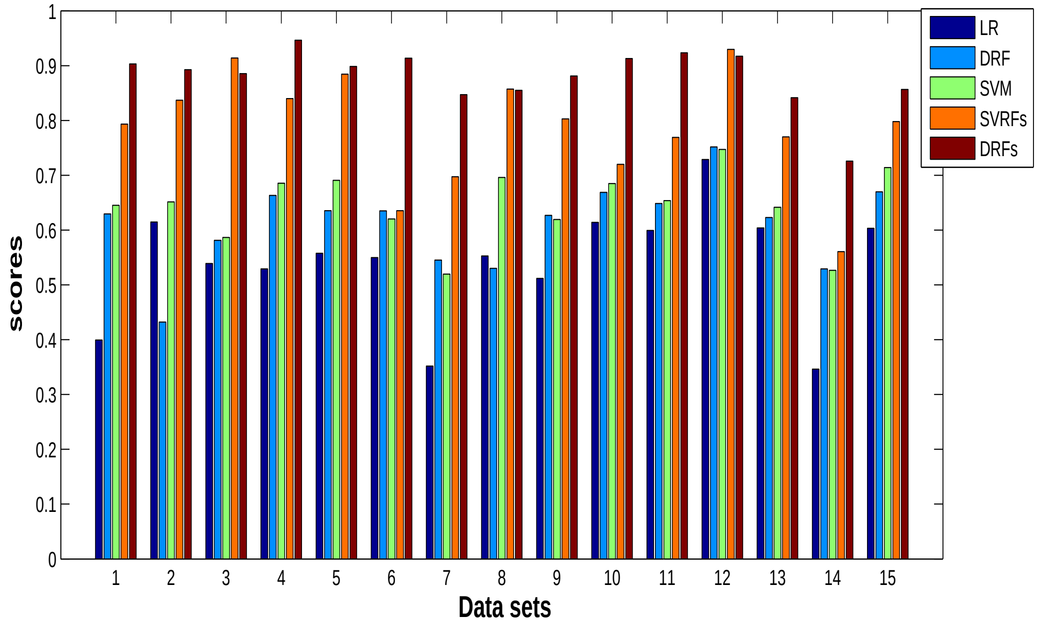


Figure 6.2: Averaged Jaccard scores on synthetic data sets

assumption by encoding spatial correlations.

Figure 6.1 shows some of the experiment results. Each row in Figure 6.1 presents one example, showing (from left to right), the true labels, the test images, and outputs from Logistic Regression (LR), DRFs, SVM, SVRFs, and DCRF. We see that, overall, SVRFs and DCRFs are most accurate. Especially when the test images are imbalanced, LR (third column) and DRFs (fourth column) produce degraded outputs caused by the poor parameter estimations from the imbalanced data.

As shown in Figure 6.1, SVRFs' results do not always produce highest Jaccard scores, which implies that regularization term τ^2 in the SVRF frameworks impacts on the accuracy. The "appropriate" value of this parameter for data samples helps find "optimal" model parameter $\boldsymbol{\nu}$ producing good segmentation results. In general, it is not trivial to find such "good" values. While we can use cross-validation method to estimate this parameter, others [35, 40] have argued that this does not guarantee effective performance.

Figure 6.2 shows that the DCRF and the SVRF are the two best performers overall, at this segmentation task, dealing with both the balanced and imbalanced data: each was significantly better than the others at the $p <\ 0.001$ level based on a paired example t-test; moreover, our DCRF performs better than the SVRF at the

Table 6.1: Average elapsed learning time (seconds)

	DRF	SVRF	DCRF
Synthetic	1581.3	714.5	21.2
Brain Tumor	1392.4	1209.4	82.3

$p < 0.004$ level. Note that the SVRF can sometimes produce better results than the DCRF— see data sets 3, 9, and 12 in Figure 6.2. Here, we assume that the SVRF found good estimates for τ^2. It is also shown in data sets 6, 7, 9, 12, and 14 that the good estimation of the regularization of DRFs help DRFs perform better than SVM.

The first row of Table 6.1 reports the average learning time for DRFs, SVRFs and DCRFs over these fifteen cases. Notice first that our DCRF requires *significantly* less time than the other two approaches — 30 times faster than SVRF and over 70 times faster than DRF. This is because there are fast ways to solve DCRF's underlying QPs. We found the SVRF was superior to DRF at the $p < 0.001$ level. We attribute this to the observation that the SVRF learner regards the Association potential as a constant while learning the Local consistency potential, but DRFs attempt to optimize both potentials simultaneously. Finally, recall that our DCRF does not compute the partition function during the training.

6.4.2 Brain Tumor Segmentation

We next apply our various models to the task of segmenting brain tumors from MR images. In our experiment, we evaluate the following seven classifiers on thirteen different time points from seven patients. Maximum Likelihood (ML $\equiv$ degenerate MRF), Logistic Regression (LR $\equiv$ degenerate DRF), SVM (degenerate SVRF), MRF, DRF, SVRF and DCRF. For each of the Random Field methods, we initialize inference with the corresponding degenerate classifier (i.e. Maximum Likelihood, Logistic Regression, or SVM). To provide a fair comparison between SVM-based models (SVRF and DCRF) and the other models, we only used the linear kernel. We consider the following 3 tasks, each using ground truth defined by an expert radiologist:

The first task is the relatively easy one of segmenting the "enhancing" tumor areas — the region that appears hyper-intense after injecting a contrast agent. (Note this includes non-enhancing areas contained within the enhancing contour – e.g. necrotic areas.) The second task is to segment the entire edema area associated

Table 6.2: Jaccard scores (percentage) for Enhancing tumor areas

Studies	Enhancing tumor Area						
	ML	MRF	LR	DRF	SVM	SVRF	DCRF
1-1	23.1	24.6	44.4	46.1	50.7	52.8	**53.2**
2-1	0.0	0.0	61.3	61.5	87.4	**87.7**	87.1
3-1	69.2	69.7	61.8	61.8	83.0	84.8	**86.8**
3-2	40.1	40.3	84.8	84.6	85.7	**85.8**	**85.8**
4-1	26.9	27.3	49.1	50.4	78.8	81.7	**82.6**
4-2	58.9	59.7	68.3	70.2	76.7	77.9	**79.2**
4-3	49.2	50.2	71.3	71.6	88.2	88.1	**88.8**
4-4	65.6	68.2	**87.5**	87.1	87.0	87.1	86.9
5-1	67.0	67.5	52.2	51.4	82.8	**84.3**	84.1
6-1	37.4	37.6	76.4	76.2	79.2	**80.4**	80.0
7-1	63.2	63.0	75.5	76.7	81.0	**81.4**	81.1
7-2	37.7	39.3	75.9	75.8	86.5	**87.3**	86.8
7-3	45.3	45.6	81.8	81.5	87.7	87.6	**87.8**
Average	44.9	45.6	63.6	68.8	81.1	82.1	**82.3**

with the tumor, which is significantly more challenging due to the high degree of similarity between the intensities of edema areas and normal cerebrospinal fluid in the various modalities. The final task is segmenting the gross tumor area as defined by the radiologist. This can be a subset of the edema but a superset of the enhancing area, and is inherently a very challenging task even for human experts, given the modalities examined.

Tables 6.2, 6.3 and 6.4 present the classification results for the three tasks. Over all three tasks, we see that the best results are typically obtained by either DCRFs and SVRFs, which are comparable to one another, and statistically better than the rest: The differences between SVRFs and the next best, SVM, across the three tasks is significant at the $p < 0.000002$ level based on a paired example t-test, but the same t-test between SVRFs and DCRFs across the tasks indicates no difference — i.e. here $p = 0.37$. However, Table 6.1 (second row) shows that our method requires significantly less training time — by a factor of 14! Although SVM performed very well visually on the three tasks(see Figure 6.3), just as we saw on the synthetic data results, this performance can not always be guaranteed.

In Table 6.2, the results from the second patient "2-1" produced an interesting observation; significant overlap between Gaussians in the high dimensional feature space leads ML and subsequently MRFs to misclassify all areas as non-tumors. This example shows that inappropriate modelling of $P(\,\mathbf{X}\,|\,\mathbf{Y}\,)$ can generate extremely

Table 6.3: Jaccard scores (percentage) for Edema tumor areas

Studies	Edema Area						
	ML	MRF	LR	DRF	SVM	SVRF	DCRF
1-1	21.9	21.6	35.7	36.7	58.0	**58.2**	58.0
2-1	33.3	34.2	59.2	61.4	**89.4**	89.2	89.3
3-1	34.4	34.4	75.5	77.2	81.7	**82.2**	81.9
3-2	47.6	48.1	73.6	74.1	80.3	**81.1**	80.5
4-1	28.3	29.1	38.6	41.2	54.0	**55.4**	54.6
4-2	43.2	46.8	45.3	46.7	54.7	**57.7**	54.9
4-3	35.4	35.4	69.9	**70.6**	69.2	69.1	69.1
4-4	44.1	43.7	78.6	79.0	77.7	77.3	**79.5**
5-1	47.8	48.6	63.6	65.7	74.8	**76.9**	74.6
6-1	40.3	40.1	79.3	79.7	82.2	**83.7**	82.9
7-1	74.9	77.7	91.2	92.4	94.8	**94.9**	**94.9**
7-2	39.2	40.4	80.9	82.7	**83.1**	82.8	**83.1**
7-3	54.1	53.9	79.3	80.7	84.6	84.5	**85.6**
Average	41.9	42.6	62.2	68.3	75.7	**76.4**	76.1

poor performance (see the first row of Figure 6.3). Although the segmentation tasks for edema and gross tumor areas are very hard, the best discriminative approaches (i.e. SVRF and DCRF) still produce segmentations that are typically very similar to the manual segmentations, for all three tasks.

6.5 Conclusions

As standard independent and identically distributed classification algorithms do not consider spatial correlations, they typically fail to correctly classify such correlated data instances. Such spatial correlations can, however, be effectively modelled by various Random Field frameworks. However, these systems (especially the ones that work effectively.) can require a significant amount of time to learn. This time constraint makes such models inappropriate for large scale real-world problems, such as segmenting brain tumors.

In this chapter, we have proposed a Decoupled CRF (DCRF) to improve the efficiency of a discriminative Random Field method for finding regions in an image. Our proposed model first learns the two potentials (Association and Local-consistency) *independently*, each based on a variant of Support Vector Machines. Afterwards, to segment regions in a novel image, it uses a new potential that is the simple sum of these potentials, using ICM (with respect to this combined potential) to produce a labelling. One main drawback in the DCRF is the independently learned potentials

Table 6.4: Jaccard scores (percentage) for Gross tumor areas

Studies	Gross Tumor Area						
	ML	MRF	LR	DRF	SVM	SVRF	DCRF
1-1	19.3	19.5	39.4	**40.9**	40.7	40.5	41.1
2-1	35.4	35.7	65.1	66.1	**78.2**	76.9	78.0
3-1	44.4	46.1	72.9	73.4	77.9	78.7	**78.2**
3-2	51.2	51.3	76.3	76.2	78.1	78.8	**80.2**
4-1	37.4	38.7	39.4	40.1	41.4	41.2	**42.1**
4-2	38.0	40.2	39.7	39.4	62.1	**64.9**	62.1
4-3	66.0	68.5	73.3	**73.5**	64.4	64.5	64.1
4-4	46.7	45.8	83.8	83.5	86.0	**87.0**	86.2
5-1	50.1	50.9	65.3	68.3	82.8	**84.8**	83.4
6-1	46.6	47.6	79.6	79.4	87.6	**88.2**	87.8
7-1	66.4	66.3	71.9	73.2	74.6	74.1	**74.7**
7-2	49.6	52.4	68.3	67.9	72.7	**72.9**	72.5
7-3	43.4	43.7	73.5	72.7	81.6	81.2	**82.0**
Average	45.7	46.7	60.6	65.7	71.4	**71.8**	71.7

do not guarantee "optimality" in modelling spatial correlations.

Our empirical results — on both synthetic and real-world data — show that our DCRF approach is virtually as accurate as the most accurate random field for this task (SVRF), but the learning time is many times faster (here, by a factor over 14 in one case, and over 30 in another). In addition, our model produces effective classification results, even when data sets are heavily imbalanced.

We currently use only (a variant of) *linear* SVMs; we expect further accuracy improvements by using other kernels. We also use only a very simple approach for combining the two potentials; again we anticipate other combination rules may produce yet better results.

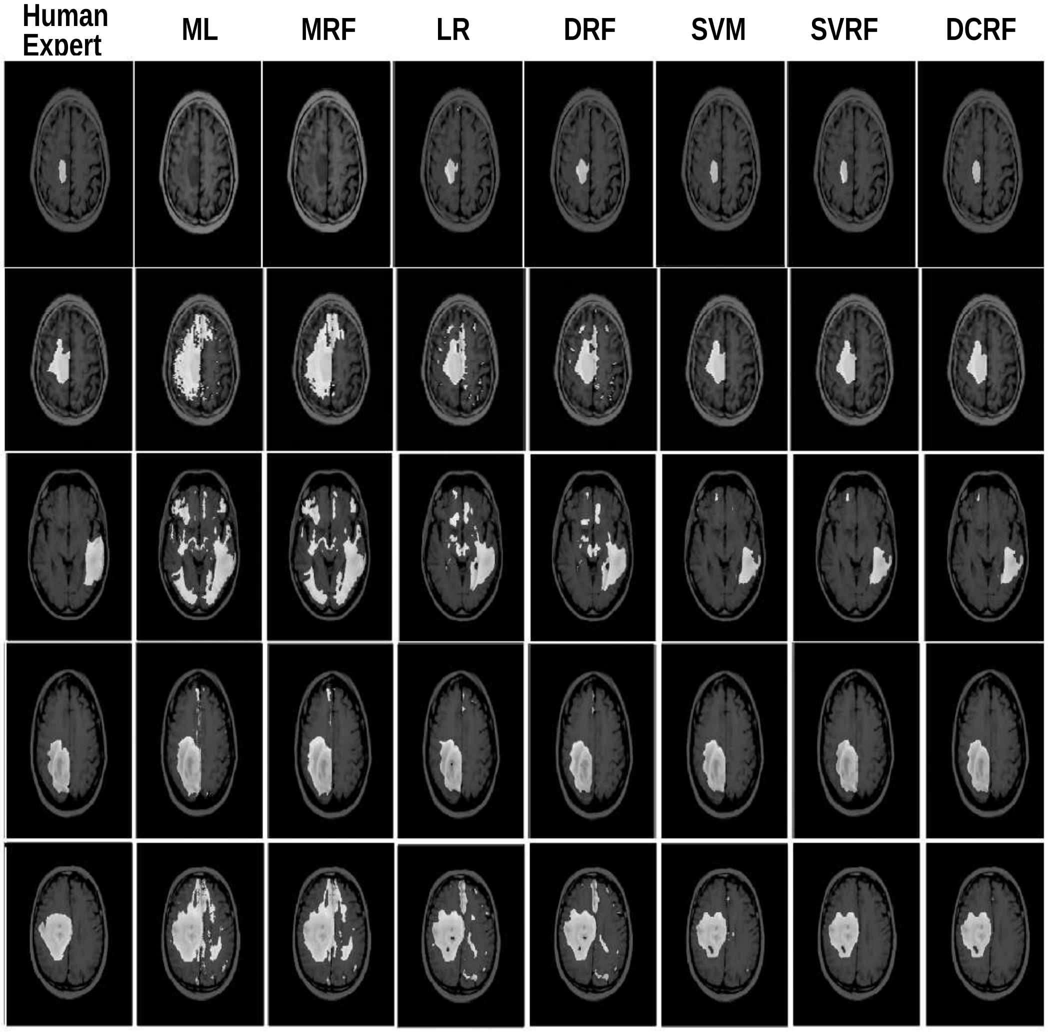


Figure 6.3: Classification results of seven methods on five different test slices, compared with human expert segmentation

Chapter 7

Pseudo Conditional Random Fields – PCRFs

7.1 Introduction

As with much of related work, there are a range of approximation methods available to learn parameters in the framework of general random fields including DCRFs [39] and Pairwise training [64, 65]. As an alternative to such approximations, we present a novel *efficient* supervised learning framework, Pseudo Conditional Random Fields (PCRFs), to model spatial compatibility among data instances. Although DCRFs learn two sets of model parameters including explicit learning parameters for edge potentials, our PCRFs can be viewed as a *regularized* iid discriminative classifier, where the classification task is performed with a regularization term that explicitly incorporates correlated dependencies. Specifically, a classifier is first *trained* under the iid assumption, and then relaxes its iid assumption during *inference* step. In other words, we regularize a decision of an iid classifier for a pixel by considering its neighboring pixels' labels as well as their feature characteristics.

We demonstrate our framework's performance by applying it to classify pixels using synthetic and real world problem of MR image analysis. In each case, we have obtained significant accuracy improvement over baselines – LR and SVM. In addition, the PCRF is as accurate as state-of-the-art CRF variants. Note that our training only involves learning an iid learner, and therefore its learning is much more efficient than CRF-variants.

Section 7.2 briefly reviews related work highlighting our motivation. Section 7.3 then introduces our novel framework – PCRFs – describing the two major steps in typical supervised learning – Learning and Inference. Section 7.3.2 discusses one major contribution to relax the iid assumptions made from base classifiers. Section 7.4 shows empirical experiments for efficiency and effectiveness of our model. In Section 7.5, we summarize our PCRF, also comparing with DCRFs.

7.2 Related Works

Extensions to CRFs such DRFs and SVRFs were designed to overcome these disadvantages of MRFs by relaxing conditional independency and incorporating observations when formulating spatial dependency. In Equation (2.17), the typical CRF's formulation is strictly based on *conditional* probability distribution, while an MRF is formulated on *joint* distribution of $\mathbf{X}$ and $\mathbf{Y}$. In addition, the CRF variants incorporate observations of data instances using the $\Psi(y_i, y_j, \mathbf{X})$. Empirically, CRF variants have shown better accuracy over spatially correlated classification problems than MRFs [35, 37].

The effectiveness of DRFs and SVRFs is compromised by the computational complexity of computing $Z(\mathbf{X})$ (refer to Equations (2.17) and (4.1)). Typically, their learning algorithms involve maximizing conditional likelihood which requires computing the derivatives of their objective. This in turn involves computing the conditional expectation of feature [34, 35, 40]. DRFs and SVRFs use approximations to avoid intractable computations associated with the conditional expectation. Recently, an alternative technique to deal with the computation of $Z(\mathbf{X})$, matrix-tree theorem, was applied to non-projective dependency parses: that is, dependency parses involves the exponential number of structured possibilities in sentence parsing tasks [32]. However, the naïve application of the theorem yields time complexity $O(n^4)$ for n words in a sentence.

The Decoupled Conditional Random Fields (DCRF) (discussed in Chapter 6) was introduced to improve the efficiency of CRF-based formulations by *decoupling the two potential functions* when learning parameters. Specifically, the DCRF system views a CRF as the combination of two “independently learned” potentials [39]. PCRFs differ from DCRFs since PCRFs only require learning parameters for a single potential. Coordination Classifiers [22], as an ensemble classifier, marginalizes its local consistency potential to compute the singleton potential. This means that two potentials are dependent which differs from our PCRFs.

In the next section, we propose a novel system, PCRF that efficiently learn parameters that model spatial correlation, efficiently learning model parameters.

This also produces effective classification results.

7.3 Pseudo Conditional Random Fields – PCRFs

While it is less expensive to estimate the maximum likelihood MRF parameters than the CRF parameters, CRF (and its variants) are more accurate. We introduce our Pseudo Conditional Random Fields (PCRF) system to take advantage of both approaches.

Our PCRF seeks the most-likely labelling, viewed as

$$P_\theta(\,\mathbf{Y}\,|\,\mathbf{X}\,) = \prod_{i \in S} P_\theta(\,y_i\,|\,\mathbf{X}, \mathbf{Y} - y_i\,)$$

Given feature vectors (observations) – $\mathbf{x}_i$ and $\mathbf{x}_{N_i}$ for each pixel i and its neighboring pixels N_i – as well as the class label y_j for each neighboring pixel $j \in N_i$, the PCRF formulation then defines

$$P_\theta(\,y_i\,|\,\mathbf{x}_i, \mathbf{x}_{N_i}, y_{N_i}\,) \quad = \quad \psi_\theta(\mathbf{x}_i, y_i) \times \rho_{N_i}, \tag{7.1}$$

where the potential function $\psi_\theta(.)$ is parameterized by θ and ρ_{N_i} is a regularization term that helps minimize uncertainty of $\psi_\theta(.)$ by incorporating spatial dependencies. If we simply define $\psi_\theta = p_\theta(y_i|\mathbf{x}_i)$ and $\rho_{N_i} = 1$, $i \in S$, we obtain the typical local conditional probabilistic model that corresponds to an iid classifier – for instance, logistic regression. However, the challenges to represent regularization term ρ_{N_i} still remain: (1) it explicitly needs to model spatial dependency; (2) it needs to be data dependent, implying that spatial correlations should consider observation similarity. Therefore, we define ρ_{N_i} as a product of two functions, considering neighboring pixels N_i.

$$\rho_{N_i} = \prod_{j \in N_i} \phi^o(\mathbf{x}_i, \mathbf{x}_j) \times \phi^c(y_i, y_j) \tag{7.2}$$

Note that $\phi^o(\mathbf{x}_i, \mathbf{x}_j)$ is a potential function that quantifies how much observations of pixels at i and j are comparable. The $\phi^c(y_i, y_j)$ function measures interactions between two class labels – y_i and y_j – and specifies how continuity with respect to class labels can be determined. In other words, if $\phi^c(y, y')$ gives a large score when $y \equiv y'$, then it prefers to have neighboring pixels being the same class label.

7.3.1 Learning

Typical CRF variant models are slow as they try to compute exact expectations, when learning parameters [34, 35, 37, 41]. To approximate the computation, one can use pseudo-likelihood, contrastive divergence, and pseudo-marginal approximation [34, 35, 41]. However, none of them consistently outperforms the others [34].

In the PCRF system, the parameter to be learned is associated with $\psi_\theta(\mathbf{x}_i, y)$

$$\psi_\theta(\mathbf{x}_i, y) = \sigma(\theta^T h(\mathbf{x}_i)), \tag{7.3}$$

where $\sigma(t) = \frac{1}{1+exp(-t)}$ corresponds to a local discriminative classifier (i.e. logistic regression), and $h(\mathbf{x}_i)$ is a feature function. This explicitly quantifies the probability being class y given observation $\mathbf{x}_i$. Note that we mainly focus on discriminative approaches rather than generative ones due to their robustness over generative approaches [49].

PCRF's learning algorithm is simple, and more efficient than CRF variants due to its formulation (Equation (7.3)); we only need to find parameter θ^* for a local potential function $\psi(.)$ by maximizing conditional log likelihood,

$$\theta^* = \arg\max_\theta \sum_{i \in S} \Big[y_i \log \sigma(\theta^T h(\mathbf{x}_i)) + (1 - y_i) \log(1 - \sigma(\theta^T h(\mathbf{x}_i))) \Big], \tag{7.4}$$

where y_i is a class label of observation $\mathbf{x}_i$.

7.3.2 Inference

Inference in our PCRF system explicitly incorporates spatial correlations. Our objective in inference is to find $\mathbf{Y}^*$ maximizing $P(\mathbf{Y}|\mathbf{X})$, written in log scale as,

$$\begin{aligned} \mathbf{Y}^* &= \arg\max_Y \; \log P(\mathbf{Y}|\mathbf{X}) \\ &= \arg\max_Y \sum_{i \in S} \Big(\log \psi(\mathbf{x}_i, y_i) + \log \rho_{N_i} \Big), \end{aligned} \tag{7.5}$$

where $\mathbf{X} = \{\mathbf{x}_i\}_{i \in S}$ and $\mathbf{Y} = \{y_i\}_{i \in S}$. Note that Equation (7.5) requires considering an exponential number of possibilities (i.e. $2^{|S|}$ for binary case) to find an optimal $\mathbf{Y}^*$. To efficiently solve Equation (7.5), we express it as,

$$\begin{aligned} &\log P(\mathbf{Y}|\mathbf{X}) = \\ &\sum_{i \in S} \log \psi(\mathbf{x}_i, y_i) \quad + \quad \sum_{j \in N_i} \Big(\log \phi^o(\mathbf{x}_i, \mathbf{x}_j) + \log \phi^c(y_i, y_j) \Big) \end{aligned} \tag{7.6}$$

Here, we see Equation (7.6) as an energy minimization problem, and therefore use graph cuts as they are designed to solve the pixel classification problem [8].

We solve graph cuts using linear programming to seek max-flow/min-cut, where a graph is represented with nodes corresponding pixels and edges connecting neighboring pixels. The weight between nodes i and j is determined by $\phi^o(.)$ and $\phi^c(.)$.

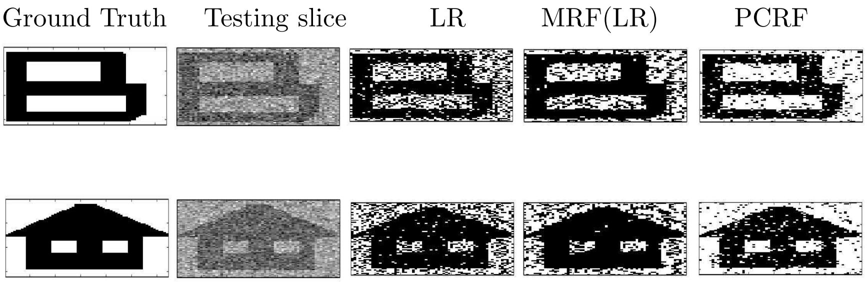


Figure 7.1: Synthetic data examples

Here, we need to introduce two auxiliary nodes: s and t denoting tumor and non-tumor class labels, respectively. The weight between node s and node i is weighted with $\psi(\mathbf{x}_i, s)$, and $\psi(\mathbf{x}_i, t)$ for node t and i.

7.4 Experiments

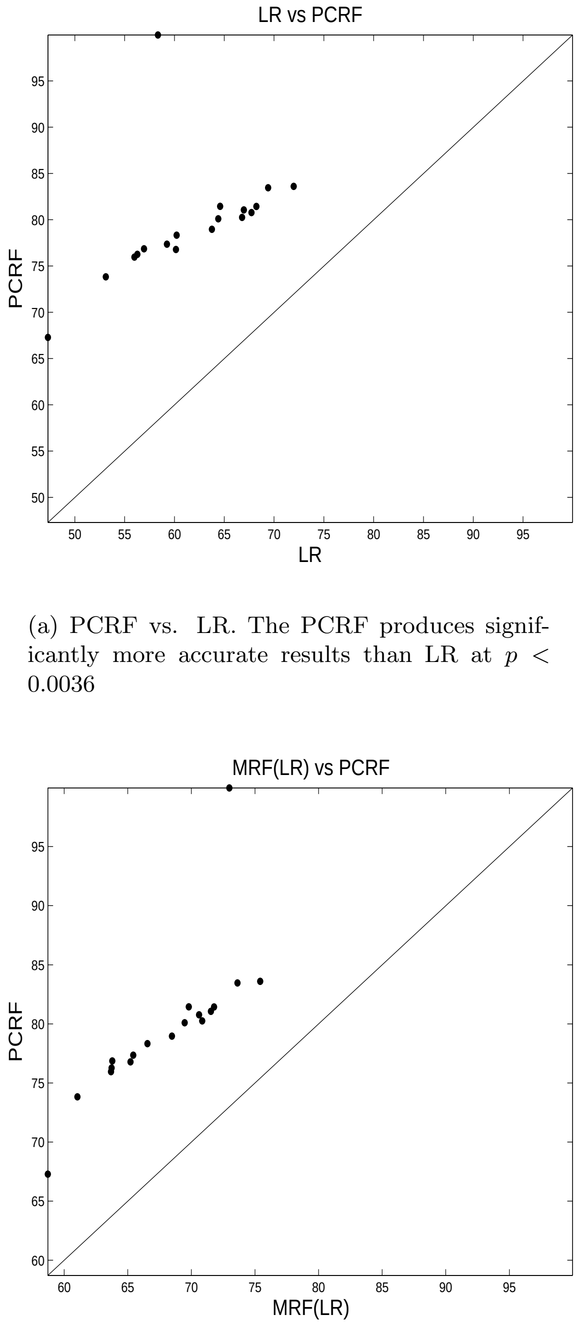
In this section, we present empirical results on synthetic and real world problem – magnetic resonance image analysis – using our novel PCRF. In order to evaluate our model, we first compare the results with baseline models – typical iid classifiers. Since the PCRF can be viewed as a regularized discriminative iid classifier, we want to highlight the effective performance of our PCRF in comparison with its corresponding iid classifier. Second, we also perform experiments by augmenting a typical MRF using a discriminative iid classifier that relaxes an MRF's local likelihood assumption. That is, we use a local *conditional* probability models – logistic regression and support vector machine – in a typical MRF. They are denoted as MRF(LR) and MRF(SVM), respectively.

We use $\phi^o(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$ which produces a scalar: the cosine measure of the similarity. Note this produces its largest value as the two vectors match one another. We also set $\phi^c(y_i, y_j) = \alpha$, if $y_i \equiv y_j$, otherwise $1 - \alpha$, where α weighs the continuity of same class labels. Here, we set $\alpha = 0.6$

7.4.1 Synthetic image sets

This section demonstrates our PCRF performance as a binary classification over a 2-D lattice comparing with base models using eighteen synthetic data sets introduced in Chapter 3.

Examples in Figure 7.1 illustrate classification results from two of synthetic sets. The first two columns are the ground truth and its testing image. The classification



(a) PCRF vs. LR. The PCRF produces significantly more accurate results than LR at $p < 0.0036$

(b) PCRF vs. MRF(LR). The PCRF is significantly more accurate than MRF(LR) at $p < 0.0013$

Figure 7.2: Jaccard scores (percentage) from synthetic data sets

results are presented from third columns: Logistic Regression(LR), MRF(LR), and PCRF. It is clear that LR produces the worst classification results, even though it accurately retrieves shape boundaries. Background pixels in LR are classified as target labels since a classification decision for a pixel is made only by considering the observation of the pixel. MRF(LR) produces better accuracy than LR since it simply relaxes LR's decisions by considering neighboring pixels' label distributions. This results in smoothing effects, but still suffers from under estimates of background pixel labels.

The last column demonstrates effects of considering neighboring pixels with respect to their labels as well as observation similarity. Our PCRF distinguishes boundaries clearly, and the background pixels are relatively more corrected comparing with LR and augmented MRF. This is because the PCRF formulation avoids under estimates of spatial compatibility. Figure 7.2 supports robustness of PCRF from eighteen data sets. Each point above the diagonal line in Figure 7.2 indicates PCRF producing higher Jaccard scores for a data set.

7.4.2 Brain Tumor Segmentation

We first applied three models – LR, MRF(LR), and PCRF – to the classification of eleven studies from brain tumor MR images, where an MR image (a.k.a. slice) has three modalities available. Refer to Figure 3.3 for the examples of three modality. We segmented the "enhancing" tumor area, the region that appears hyper-intense after injecting the contrast agent, and we also included non-enhancing areas contained within the enhancing contour. Figure 7.3 shows examples of classification results including the ground truth and testing slice at the first and second column, respectively. It is clear that the visual identification of tumor areas is not a trivial task.

From examples on the first row in Figure 7.3, LR correctly classifies pixels from the slice with an outlined tumor contour, but it also incorrectly produces many small blobs as false positives. As shown in Figure 7.3, MRF(LR) further smooths rough boundaries of LR result. However, it still suffers from many false positives. PCRFs show better smoothed tumor areas. From the second and third rows examples, it is clear that PCRF is robustly effective. Overall, PCRF's accuracy is higher than the other two models — LR and MRF(LR) — at the $p < 0.0045$ and $p < 0.0048$ level on a paired example t-test, respectively. Figure 7.4 presents Jaccard scores

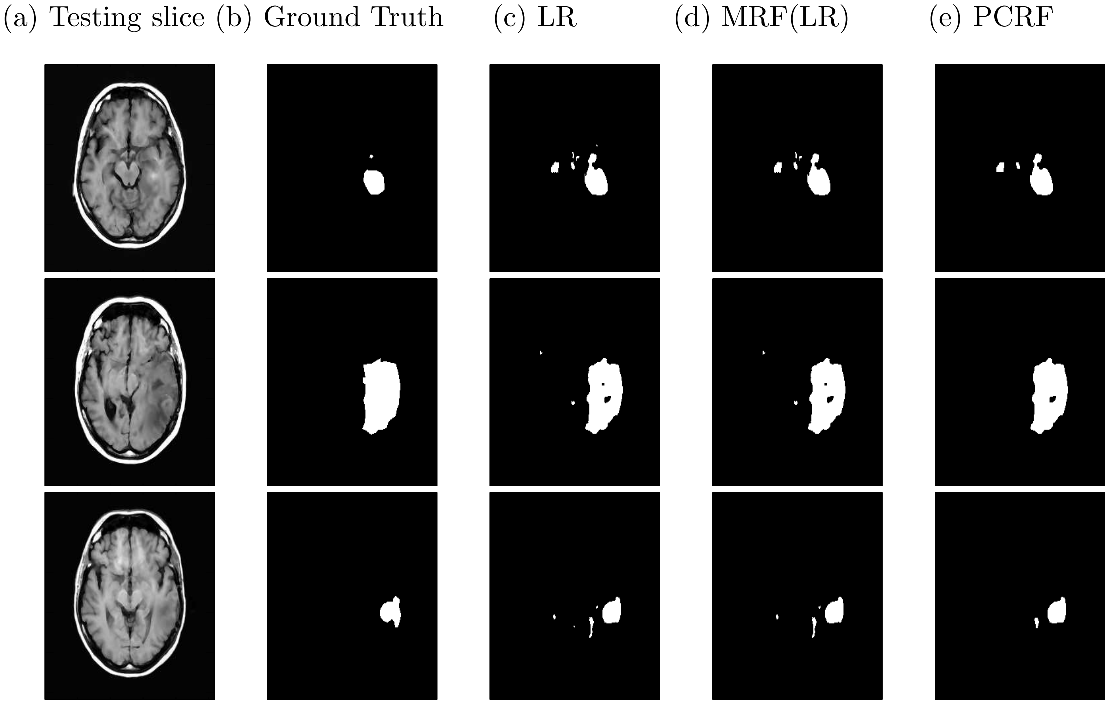


Figure 7.3: Classification results from various models. PCRF reduces false positives, resulting in better smoothed tumor shapes

(percentage) from all testing results.

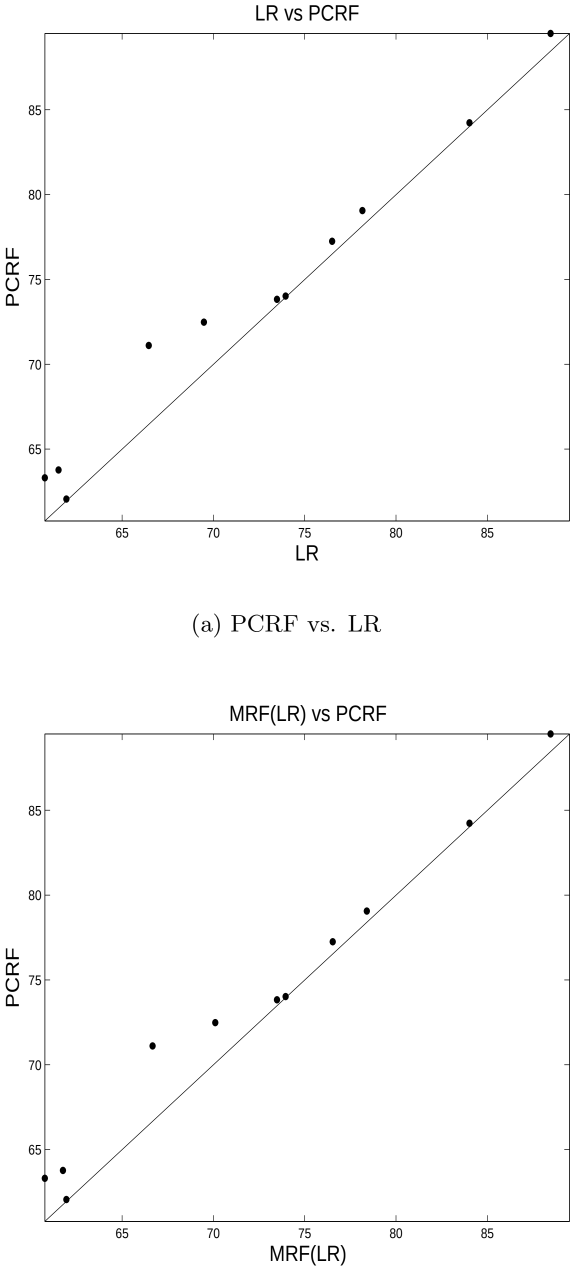
Figure 7.5 presents other segmentation results, showing the results from segmenting the entire *Edema Area* associated with the tumor. This is known to be more challenging because of the high degree of similarity between the intensities of edema areas and normal cerebrospinal fluid in the various modalities.

Here, we implement PCRF(SVM), which differs from the PCRF system by using an SVM to compute the $\psi(\mathbf{x}, y)$ (from Equation (7.5)) which models the relationship between a voxel's feature vector and its label. An SVM produces the distance between a hyperplane and a data instance as its decision value $f_{SVM}(\mathbf{x}_i) \in (-\infty, +\infty)$. To normalize this unbounded range, we fit this value to a sigmoid function:

$$g_{\beta_0,\beta_1}(f_{SVM}(\mathbf{x})) = P(\, y = +1 \,|\, f_{SVM}(\mathbf{x})\,) = \frac{1}{1 + \exp(\beta_0 + \beta_1(f_{SVM}(\mathbf{x})))}, \qquad (7.7)$$

estimating the parameters β_0 and β_1 from the training data $\{(f_{SVM}(\mathbf{x}_i),\, y_i)\}_i$. Refer to Section 4.2.1 for details.

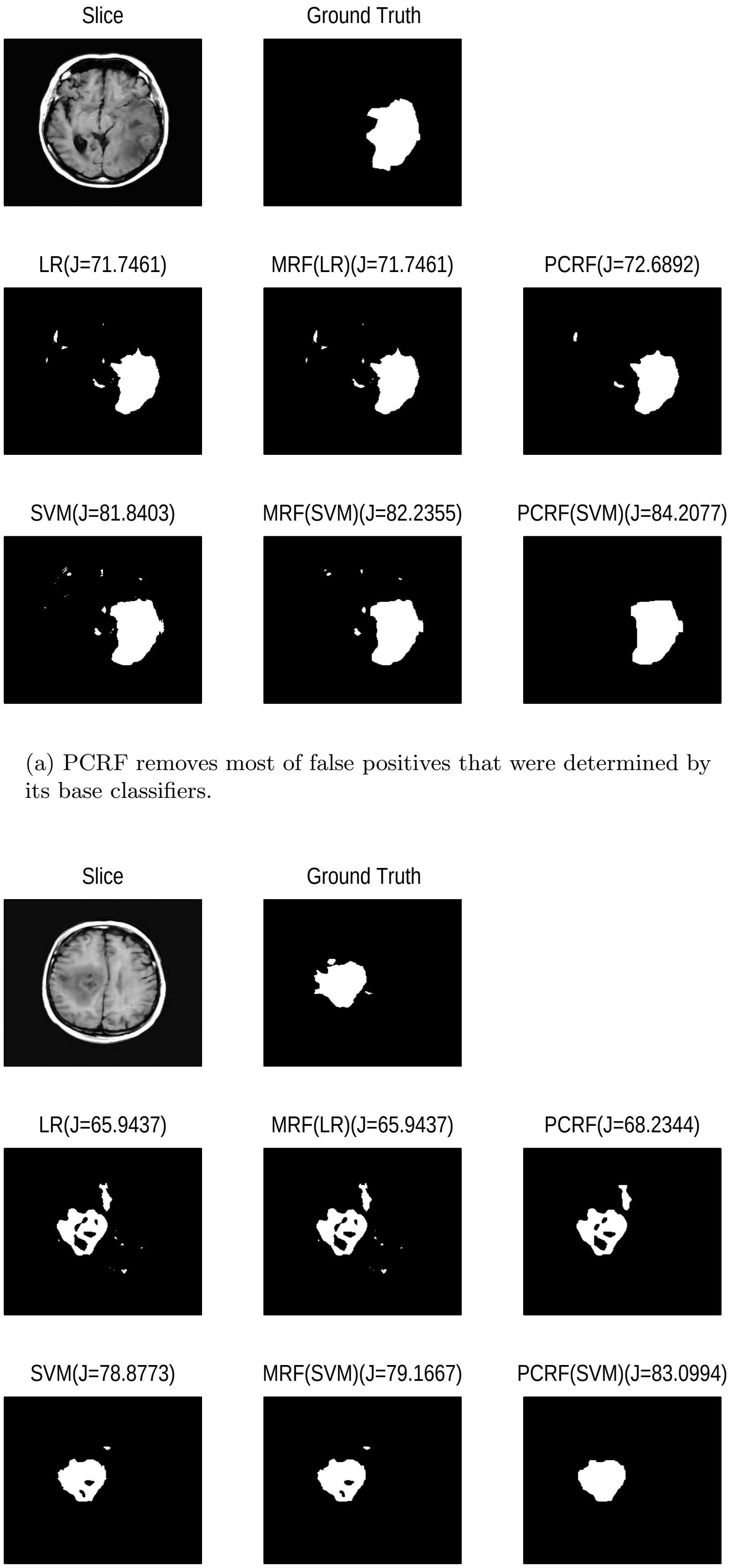
Figure 7.6 compares the percentage Jaccard scores of PCRF(SVM) vs SVM to classify enhancing, edema, and gross tumor areas. We see that PCRF(SVM) outperforms its base classifier SVM at $p < 0.0001$.



(a) PCRF vs. LR

(b) PCRF vs. MRF(LR)

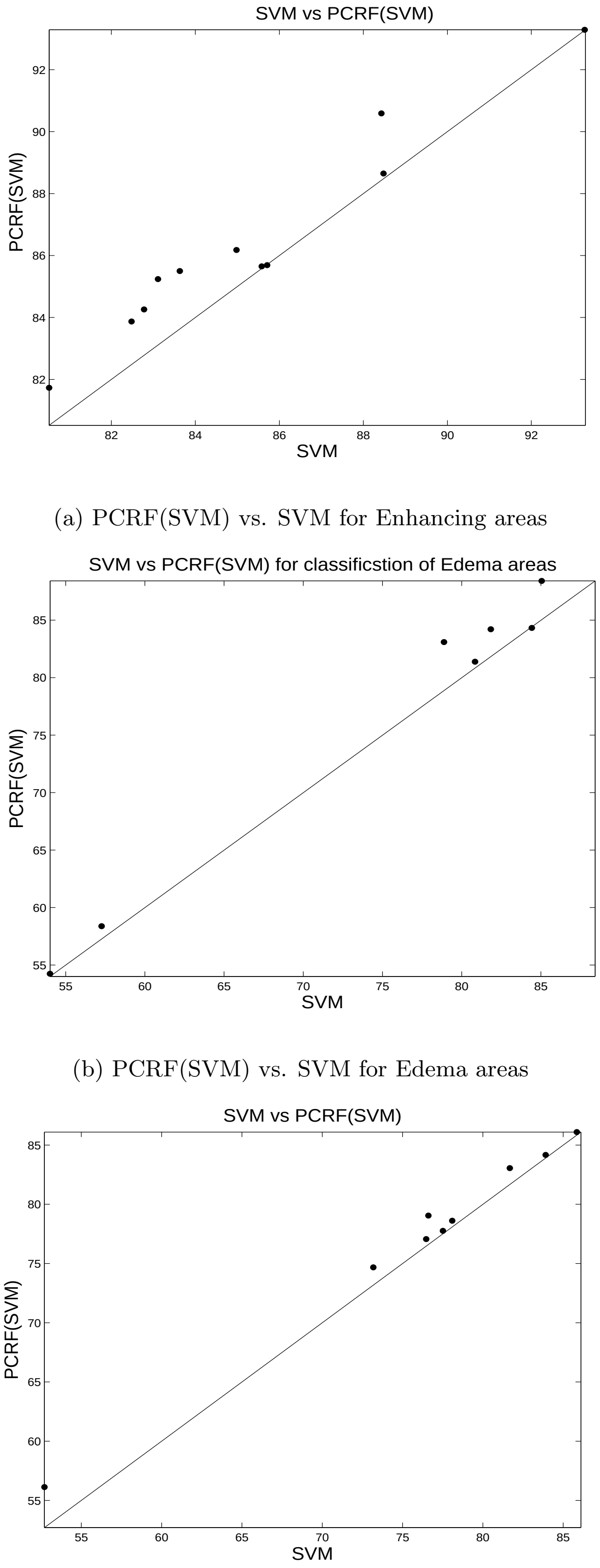
Figure 7.4: Jaccard scores (percentage) for Enhancing areas



(a) PCRF removes most of false positives that were determined by its base classifiers.

(b) PCRF successfully recovers false negatives by filling in holes.

Figure 7.5: Classification results for edema areas. Jaccard scores(percentage) are presented along with classification results.

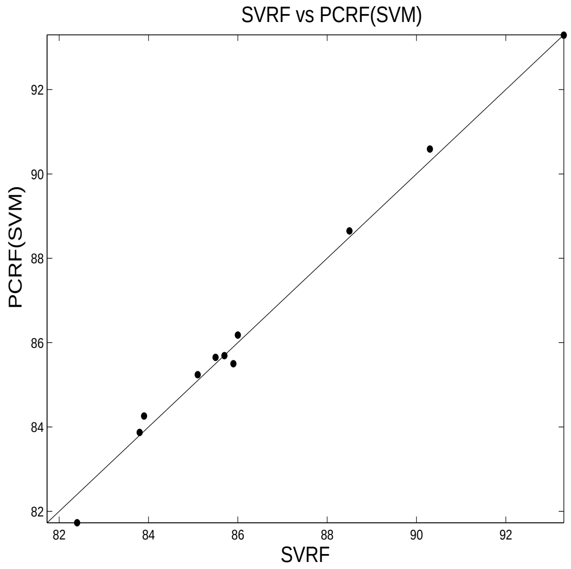


(a) PCRF(SVM) vs. SVM for Enhancing areas

(b) PCRF(SVM) vs. SVM for Edema areas

(c) PCRF(SVM) vs. SVM for Gross tumor areas

Figure 7.6: Jaccard scores(percentage) from Enhancing, Edema, and Gross Tumor areas



(a) PCRF(SVM) vs. SVRF

Figure 7.7: Jaccard scores(percentage) from Enhancing areas

We also compared our PCRF system with the state-of-the-art CRF variant, the Support Vector Random Field (SVRF [40]), whose potential functions are based on Support Vector Machines (SVMs). Figure 7.7 shows that PCRF(SVM) is comparable with SVRF.

We also perform efficiency tests for the PCRF: how efficiently the PCRF is learned. As our PCRF did not need to learn parameters for modelling its spatial correlation, we anticipated it would be significantly faster during the learning stage. The learning times (average across 11 patients, in seconds) confirm this:

Table 7.1: Average elapsed learning time (seconds)

	DRF	SVRF	DCRF	PCRF
Tumor segmentation	1697	1276	63	**38**

Our PCRF was over 40 times faster than the DRF and over 30 times faster than the SVRF ($p < 10^{-37}$ and $p < 10^{-29}$, paired-samples t-tests for DRFs and SVRFs, respectively). Even DCRF, known as the fastest CRF variant, is significantly slower than our PCRF ($p < 10^{-26}$).

7.5 Conclusion

We found that the PCRF(SVM) system, which uses a linear SVM to map from a data instance to label, worked effectively. We might be able to obtain further performance improvements by using a non-linear kernel function. We are extending this work to develop effective systems to overcome the limitations of patient-specific training, by taking advantage using semi-supervised learning principles.

This chapter has presented the Pseudo Conditional Random Field (PCRF) model, a CRF-inspired formulation that incorporates a specified potential function to model the relationships between neighboring data instances. Our PCRF is efficient to train as it does not need to fit parameters that model the neighbor relationships. This in turn allows PCRF to be trained much faster than DCRFs. Both PCRF and DCRF are designed to be efficient. Which is better? If one has sufficient domain knowledge to express the two PCRF potentials, then we recommend using PCRFs. As data distributions can be changed, the hand-tuned potentials, which may require multiple trials but taking advantage of domain expertise, could have an advantage of accurately reflecting data characteristics. Our PCRF can be viewed as a regularized iid classifier, which relaxes its decisions by considering neighboring instances with respect to labels and observations. Thus, during inference, PCRF relaxes the iid assumption. We demonstrate that PCRF is effective by showing it can effectively segment brain tumors from MR images, achieving state-of-the-art segmentation results, but at a small fraction of the training time.

Chapter 8

Conclusions and Future directions

This dissertation presents several novel models that incorporate spatial correlations, to produce systems that are effective segmenters, and that can be learned efficiently. They are extensions to conditional random fields (CRFs), often based on discriminative iid classifiers such as Logistic Regression and Support Vector Machines.

Support Vector Random Fields (SVRFs) and Semi-Supervised Discriminative Random Fields (SSDRFs) produce accurate classification results both on synthetic and real world problems, outperforming their degenerate iid classifiers as well as several random fields. Decoupled Conditional Random Fields (DCRFs) and Pseudo Conditional Random Fields (PCRFs) are designed *to efficiently* learn models for spatial correlations, while remaining as effective as typical CRF variants.

8.1 Future Directions

There are several future directions that can lead to yet other interesting theoretical and empirical results. One of major challenges in incorporating spatial correlations in 2-D lattice is dealing with the computational complexity of the CRF framework, which involves an intractable computation for computing the normalizing factor $Z(X)$. This challenge forces practitioners to use approximations, including pseudo likelihood, which have key impacts on classification results [34]. We believe that "effective" approximations produce highly accurate classification results with "efficient" learning procedures.

8.1.1 Model

Here, we discuss several future directions in designing a model that incorporates spatial correlations, which can produce accurate classification results.

1. The typical CRF formulations involves two potentials – one to express the local conditional probability for a class label given features of the individual pixel, and the other to express the local spatial compatibility among local neighborhood. By exploring different ways to express the local spatial compatibility, we may be able to find some approaches that can produce yet other accurate classification results on challenging tasks.

2. As mentioned above, a CRF can be seen as a combination of two potentials. A DCRF uses only a very simple approach to combine the two potentials; again we anticipate other combination rules may produce yet better results.

3. For the local conditional probability, our current methods use only *linear* SVMs. We anticipate further improvements by using other kernels, although this may require using extensive prior knowledge about data sets.

4. Dietterich *et al.* [15] propose learning a CRF by applying Friedman's gradient tree boosting method; their empirical experiments demonstrate that a CRF can be learned efficiently, achieving high accurate classification results. We also want to extend Friedman's gradient tree boosting to deal with spatial correlations in a 2-D structure, learning a model by stage-wise optimizations, similar to the boosting process [59].

8.1.2 Applications

We anticipate being able to apply our models to several other applications.

1. In this dissertation, our experiments on brain tumor segmentation task are based on patient-specific scenario, where training and testing are performed on a specific patient. We can continue to extend our models to deal with *non patient-specific scenario*, where we can train a model on k patients $\{A_1, \ldots, A_k\}$, and test the learned model on novel patient A_{k+1}. We anticipate this approach will still yield effective results.

2. The problem we have investigated so far is “classification” where each pixel in a given image is categorized into a class. For the brain tumor segmentation task, it would be interesting and useful to develop a framework that produces a probability map; that is, mapping each voxel $\mathbf{x}_i$ into the probability that it is a tumor. Currently, we use graph cuts [8] for inference (i.e. to produce the class labels for voxels); graph cuts can be modified to produce a probability map as an alternative to the classification result.

3. The empirical evaluations on challenging real world problems show encouraging results, and hence we can extend the models to a wide range of applications such as 3-D classification problems.

8.2 Summary of Contributions

This dissertation extends typical iid classification to a 2-D lattice structure that incorporates spatial correlations of class labels. Essentially, the primary results in this thesis are:

- Our SVRF system, as a novel type of CRFs, is an *effective* segmenter.
- Our SSDRF system *incorporates unlabelled* as well as *labelled* data in a supervised learning framework produces an effective segmenter.
- Our DCRF and PCRF systems, which *efficiently learn models that incorporates spatial correlations*, can achieve effective classification results.

The first statement is addressed in Chapter 4 which defines Support Vector Random Fields (SVRFs), that exploit the ideas underlying Support Vector Machines. The SVRF is based on a typical supervised learning framework. Its classification accuracy is significantly better than degenerate iid basis classifier and other random fields.

To explore the challenge of the second statement, we propose a semi-supervised learning framework to learn a model that incorporates spatial compatibility in Chapter 5. Empirical results demonstrate that by incorporating unlabelled data into the learning procedure, we can produce a conditional random field that is more accurate than the one learned without the unlabelled data.

While working on these two challenges, we noticed that the learning efficiency was one of the critical issues in CRF-variants especially when the graph structure of class labels contains cycles (e.g. on grids). Chapter 6 first presents a "decoupled" approach, that separately learns the two potentials. To further enhance the learning efficiency, Chapter 7 presents an alternative; use a standard iid discriminative classifier to learn the local conditional probability model without considering dependencies among class labels. This system uses a hand-tuned model of spatial dependencies in the inference steps. Our empirical results show that these learning approaches are significantly faster than the standard approaches, while achieving the classification results as effective as CRF-variants.

We anticipate that we will be able to use these ideas in other applications where correlations exist among data instances such as social network analysis and web information extraction [50, 66].

Bibliography

[1] Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. Applying support vector machines to imbalanced datasets. In *In Proceedings of the 15th European Conference on Machine Learning (ECML*, pages 39–50, 2004.

[2] Y. Altun, D. McAllester, and M. Belkin. Maximum margin semi-supervised learning for structured variables. In *Advances In Neural Information Processing Systems 18*. 2006.

[3] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22:39–71, 1996.

[4] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of Royal Statistical Society, Series B*, pages 36:192–236, 1974.

[5] J. Besag. On the statistical analysis of dirty pictures. *Journal of Royal Statistical Society. Series B*, 48:3:259–302, 1986.

[6] C. Bishop, N. Lawrence, T. Jaakkola, and M. Jordan. Approximating posterior distributions in belief networks using mixtures. In *Advances in Neural Information Processing Systems 10, MIT Press, Cambridge MA (1998).*, 1998.

[7] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, pages 92–100, 1998.

[8] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *ICCV*, pages 377–384, 1999.

[9] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[10] G. Celeux and G. Govaert. A classification EM algorithm for clustering and two stochastic versions. *Comput. Stat. Data Anal.*, 14(3):315–332, 1992.

[11] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

[12] Ting Chen and Dimitris N. Metaxas. Gibbs prior models, marching cubes, and deformable models: A hybrid framework for 3d medical image segmentation. In *International Society and Conference Series on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 703–710, 2003.

[13] A. Corduneanu and T. Jaakkola. Data dependent regularization. In O. Chapelle, B. Schoelkopf, and A. Zien, editors, *Semi-Supervised Learning*, pages 163–182. MIT Press, Cambridge, MA, 2006.

[14] Thomas G. Dietterich. Machine-learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.

[15] Thomas G. Dietterich, Adam Ashenfelter, and Yaroslav Bulatov. Training conditional random fields via gradient tree boosting. In *In Proceedings of the 21th International Conference on Machine Learning (ICML*, pages 217–224. ACM, 2004.

[16] Pedro Domingos and Michael J. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130, 1997.

[17] C. Garcia and J.A. Moreno. Kernel based method for segmentation and modeling of magnetic resonance images. *LNCS*, 3315:636–645, Oct 2004.

[18] D.T. Gering. *Recognizing Deviations from Normalcy for Brain Tumor Segmentation*. PhD thesis, MIT, 2003.

[19] Amir Globerson and Tommi S. Jaakkola. Approximate inference using planar graph decomposition. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 473–480. MIT Press, Cambridge, MA, 2007.

[20] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 529–536. MIT Press, Cambridge, MA, 2005.

[21] Russell Greiner and W. Zhou. Structural extension to logistic regression. *Proceedings of the Eighteenth Annual National Conference on Artificial Intelligence (AAI02)*, 2002.

[22] Yuhong Guo, Russell Greiner, and Dale Schuurmans. Learning coordination classifiers. In *IJCAI*, pages 714–721, 2005.

[23] Trevor Hastie, Robert Tibshirani, and Jerome Friedma. *The Elements of Statistical Learning*. Springer, New York, NY, 2002.

[24] John Henderson, Steven Salzberg, and Kenneth H. Fasman. Finding genes in dna with a hidden markov model. *Journal of Computational Biology*, 4:127–141, 1997.

[25] Richard Hughey and Anders Krogh. Hidden markov models for sequence analysis: Extension and analysis of the basic method. *CABIOS*, 12:95–107, 1996.

[26] F. Jiao, S. Wang, C. Lee, R. Greiner, and D Schuurmans. Semi-supervised conditional random fields for improved sequence segmentation and labeling. In *COLING/ACL*, Sydney, Austrailia, July 2006.

[27] T. Joachims. Making large-scale svm learning practical. In B. Scholkopf, C.J.C. Burges, and A.J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999.

[28] M. Jordan, editor. *Learning in Graphical Models*. MIT Press, 1998.

[29] Zoltan Kato and Ting Chuen Pong. A markov random field image segmentation model for color textured images. *Image and Vision Computing*, 24(10):1103–1114, 2006.

[30] M.R. Kaus, S.K. Warfield, A. Nabavi, P.M. Black, F.A. Jolesz, and R. Kikinis. Automated segmentation of MR images of brain tumors. *Radiology*, 218:586–591, 2001.

[31] R. Kindermann and J.L. Snell. Makrov random fields and their applications. *American Mathematical Society*, 1980.

[32] Terry Koo, Amir Globerson, Xavier Carreras, and Michael Collins. Structured prediction models via the matrix-tree theorem. In *In EMNLP-CoNLL*, 2007.

[33] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *CVPR*, 2003.

[34] Sanjiv Kumar, Jonas August, and Martial Hebert. Exploiting inference for approximate parameter learning in discriminative fields: An empirical study. In *EMMCVPR*, pages 153–168, 2005.

[35] Sanjiv Kumar and Martial Hebert. Discriminative fields for modeling spatial dependencies in natural images. *Advances in Neural Information Processing Systemsf 16*, 2003.

[36] Sanjiv Kumar and Martial Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. *Proceedings of the 2003 IEEE International Conference on Computer Vision (ICCV '03)*, pages 1150–1157, 2003.

[37] J. Lafferty, F. Pereira, and A. McCallum. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICMLProceedings of the 23rd international conference on Machine learning (ICML)*, 2001.

[38] Chi-Hoon Lee, Russ Greiner, and Shaojun Wang. Using query-specific variance estimates to combine bayesian classifiers. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 529–536, New York, NY, USA, 2006. ACM Press.

[39] Chi-Hoon Lee, Russ Greiner, and Osmar Zaïane. Efficient spatial classification using decoupled conditional random fields. In *10th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 272–283, 2006.

[40] Chi-Hoon Lee, Russell Greiner, and Mark Schmidt. Support vector random fields for spatial classification. In *European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 121–132, 2005.

[41] Chi-Hoon Lee, Shaojun Wang, Feng Jiao, Dale Schuurmans, and Russell Greiner. Learning to model spatial dependency: Semi-supervised discriminative random fields. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA, 2007.

[42] S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer-Verlag, Tokyo, 2001.

[43] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on platt's probabilistic outputs for support vector machine. Technical report, 2003.

[44] Richard Maclin, Edward W. Wild, Jude W. Shavlik, Lisa Torrey, and Trevor Walker. Refining rules incorporated into knowledge-based support vector learners via successive linear programming. In *AAAI*, pages 584–589, 2007.

[45] Olvi L. Mangasarian, W. Nick Street, and William H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. Technical Report MP-TR-1994-10, 1994.

[46] Andrew Mccallum, Dayne Freitag, and Fernando Pereira. Maximum entropy markov models for information extraction and segmentation. pages 591–598. Morgan Kaufmann, 2000.

[47] Medical image processing, analysis and visualization, http://mipav.cit.nih.gov/, Online.

[48] Tom Mitchell. *Machine Learning*. McGraw Hill, 1997.

[49] A. Ng and M. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *in Advances in Neural Information Processing Systems 14. Cambridge, MA: MIT Press*, 2002.

[50] Zaiqing Nie, Ji rong Wen, and Bo Zhang. 2d conditional random fields for web information extraction. In *Proc. of ICML*, pages 1044–1051. ACM Press, 2005.

[51] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.

[52] J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*, pages 185–208. MIT Press, 1999.

[53] J. Platt. *Probabilistic outpus for support vector mahcines and comparison to regulaized likelihood methods*. MIT Press, Cambridge, MA, 2000.

[54] Marcel Prastawa, Elizabeth Bullitt, Sean Ho, and Guido Gerig. A brain tumor segmentation framework based on outlier detection. *International Society and Conference Series on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2002.

[55] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *Advances In Neural Information Processing Systems 17*, 2004.

[56] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[57] L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, pages 4–15, January 1986.

[58] R.Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 1987.

[59] Robert E. Schapire. A brief introduction to boosting. In *Journal of Japanese Society for Artificial Intelligence*, pages 1401–1406, 1999.

[60] M.W. Schmidt. Automatic brain tumor segmentation. Master's thesis, University of Alberta, 2005.

[61] Bernhard Scholkopf and Alex Smola. Advances in kernel methods: Support vector learning. MIT Press, 1999.

[62] Shawe-Taylor and Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, UK, 2004.

[63] Statistical parametric mapping, http://www.fil.ion.bpmf.ac.uk/spm/, Online.

[64] Charles Sutton and Andrew McCallum. Fast, piecewise training for discriminative finite-state and parsing models. Technical Report IR-403, Center for Intelligent Information Retrieval, 2005.

[65] Charles Sutton and Andrew McCallum. Piecewise training of undirected models. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2005.

[66] Jie Tang, Jing Zhang, Limin Yao, and Juanzi Li. Extraction and mining of an academic social network. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 1193–1194, New York, NY, USA, 2008. ACM.

[67] B. Taskar, C. Guestrin, and D. Koller. Max margin markov networks. In *Advances In Neural Information Processing Systems*, 2003.

[68] Ben Taskar, Vassil Chatalbashev, and Daphne Koller. Learning associative markov networks. In *Proceedings of the 23rd international conference on Machine learning (ICML)*, page 102, New York, NY, USA, 2004. ACM Press.

[69] A. Torralba, K. Murphy, and W. Freeman. Contextual models for object detection using boosted random fields. In *Advances In Neural Information Processing Systems 17*, 2004.

[70] Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Contextual models for object detection using boosted random fields. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 17*. MIT Press, Cambridge, MA, 2005.

[71] Jayaram K. Udupa, Vicki R. Leblanc, Ying Zhuge, Celina Imielinska, Hilary Schmidt, Leanne M. Currie, Bruce E. Hirsch, and James Woodburn. A framework for evaluating image segmentation algorithms. *Computerized Medical Imaging and Graphics*, 30(2):75–87, March 2006.

[72] Bram van Ginneken, Mikkel B. Stegmann, and Marco Loog. Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Medical Image Analysis*, 10(1):19–40, February 2006.

[73] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, NY, November 1999.

[74] S.V.N. Vishwanathan, N. Schraudolph, M. Schmidt, and K. Murphy. Accelerated training of conditional random fields with stochastic gradient methods. In *Proceedings of the 23rd international conference on Machine learning (ICML)*, 2006.

[75] J. Yedidia, W. Freeman, and Y. Weiss. Generalized belief propagation. In *Advances In Neural Information Processing Systems 13*, pages 689–695, 2000.

[76] J. Zhang, K. Ma, M.H. Er, and V. Chong. Tumor segmentation from magnetic resonance imaging by learning via one-class support vector machine. *Int. Workshop on Advanced Image Technology*, pages 207–211, 2004.

[77] Yongyue Zhang, Stephen Smith, and Michael Brady. Hidden markov random field model and segmentation of brain mr images. *IEEE Transactions on Medical Imaging*, 20:45–57, 2001.

[78] D. Zhou, O. Bousquet, T. Navin Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *Advances In Neural Information Processing Systems 16*, 2004.

[79] D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 23rd international conference on Machine learning (ICML)*, 2005.

[80] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 23rd international conference on Machine learning (ICML)*, 2003.