Semi-supervised Learning via Generalized Maximum Entropy

by

Ayşe Naz Erkan

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

New York University

September 2010

Yann LeCun

This thesis is dedicated to my parents,

Fatma and Muhammet Erkan.

Acknowledgements

My graduate studies have been a long journey during which I have met some truly amazing people. I consider myself very lucky as these friends, colleagues and mentors have both shaped my perspective and enhanced my career goals - many of whom broadened my horizons beyond what I had previously thought was possible.

To begin, there are no words to sufficiently express my gratitude to two exceptional women, Dr. Yasemin Altun and Prof. Margaret Wright. In May 2008, Dr. Altun invited me to Max Planck Institute (MPI) in Tübingen, Germany, which has had a profound effect on the direction of my research. She has shared her experience, time and resources generously since then. Prof. Wright has made all of this possible, helping me navigate the dense bureaucracy required to complete a PhD thesis in two different institutions, on two continents. I would like to also thank Rosemary Amico, who has provided enormous help, often beyond her responsibilities as the assistant director of the department of computer science at New York University.

I am very grateful to my advisor Prof. Yann LeCun, for all the inspiration he radiates as the head of the CBLL lab and for his confidence in me.

My interest in semi-supervised learning methods was sparked by the work and guidance of my project supervisors, Dr. Jason Weston and Dr. Ronan Collobert during my internship at NEC Research labs.

During my visit to MPI, Dr. Jan Peters and Dr. Gustavo Camps-Valls gave me support as collaborators, and were very generous with their resources. I would like to thank them for providing me with the data sets used in the experiment sections of Chapters 4 and 7. Apart from being a collaborator, Dr. Peters has provided me with all sorts of career advisement as well as feedback on my research. I would also

like to thank Prof. Dr. Bernhard Schölkopf, the director of the Empirical Inference Department at MPI, for the 18-month long MPI fellowship and for making me feel at home as an AGBS member from the very first day. Vielen Dank, Jan und Bernhard!

This thesis would not have been possible without the support of my dearest friends who were there for me whenever needed; Lyuba Chumakova, Mercedes Duff, Matthew Grimes, Despina Hadjikyriakou, Oliver Kroemer, Mehmet Küçükgöz, Pierre Sermanet, Elif Tosun and Atilla Yılmaz. They are the best listeners in the world.

Finally, I would like to thank my parents and my brother Tankut for their unconditional love and support.

Abstract

The maximum entropy (MaxEnt) framework has been studied extensively in the supervised setting. Here, the goal is to find a distribution p that maximizes an entropy function while enforcing data constraints so that the expected values of some (pre-defined) features with respect to p match their empirical counterparts approximately. Using different entropy measures, different model spaces for p, and different approximation criteria for the data constraints, yields a family of discriminative supervised learning methods (e.g., logistic regression, conditional random fields, least squares and boosting) (Dudík & Schapire, 2006; Friedlander & Gupta, 2006; Altun & Smola, 2006). This framework is known as the *generalized maximum entropy framework*.

Semi-supervised learning (SSL) is a promising field that has increasingly attracted attention in the last decade. SSL algorithms utilize unlabeled data along with labeled data so as to increase the accuracy and robustness of inference algorithms. However, most SSL algorithms to date have had trade-offs, e.g., in terms of scalability or applicability to multi-categorical data.

In this thesis, we extend the generalized MaxEnt framework to develop a family of novel SSL algorithms using two different approaches:

- *Introducing Similarity Constraints*

  We incorporate unlabeled data via modifications to the primal MaxEnt objective in terms of additional potential functions. A potential function stands for a closed proper convex function that can take the form of a constraint and/or a penalty representing our structural assumptions on the data geometry. Specifically, we impose similarity constraints as additional penalties

based on the *semi-supervised smoothness assumption*, i.e., we restrict the MaxEnt problem such that similar samples have similar model outputs. The motivation is reminiscent of that of Laplacian SVM (Sindhwani et al., 2005) and manifold transductive neural networks (Karlen et al., 2008), however, instead of regularizing the loss function in the dual we integrate our constraints directly to the primal MaxEnt problem which has a more straight-forward and natural interpretation.

- *Augmenting Constraints on Model Features*

  We incorporate unlabeled data to enhance the moment matching constraints of the generalized MaxEnt problem in the primal. We improve the estimates of the model and empirical expectations of the feature functions using our assumptions on the data geometry.

In particular, we derive the semi-supervised formulations for three specific instances of the generalized MaxEnt framework on conditional distributions, namely logistic regression and kernel logistic regression for multi-class problems, and conditional random fields for structured output prediction problems. A thorough empirical evaluation on standard data sets that are widely used in the literature demonstrates the validity and competitiveness of the proposed algorithms. In addition to these benchmark data sets, we apply our approach to two real-life problems, *vision based robot grasping*, and *remote sensing image classification* where the scarcity of the labeled training samples is the main bottleneck in the learning process. For the particular case of grasp learning, we also propose a combination of semi-supervised learning and *active learning*, another sub-field of machine learning that is focused on the scarcity of labeled samples, when the problem setup is

suitable for incremental labeling.

To conclude, the novel SSL algorithms proposed in this thesis have numerous advantages over the existing semi-supervised algorithms as they yield *convex*, scalable, inherently *multi-class* loss functions that can be kernelized naturally.

Contents

List of Figures

List of Tables

List of Appendices

Chapter 1

Introduction

Broadly speaking, machine learning algorithms aim to learn a mapping from observations $x \in \mathcal{X}$, to outputs $y \in \mathcal{Y}$. In classification, y consists of discrete values corresponding to the categories that the inputs are associated with, whereas in regression y can take arbitrary continuous values. Supervised learning algorithms infer such a mapping using completely labeled data, where the training set consists of pairs of inputs and their desired outputs. Unsupervised learning algorithms, on the other hand, deal with entirely unlabeled training sets.

Unlike the traditional supervised and unsupervised techniques, semi-supervised learning (SSL) is a relatively new sub-field of machine learning which has become a popular research topic throughout the last decade. SSL aims to make use of both labeled and unlabeled data during training. The scarcity of labeled training samples in a wide spectrum of applications ranging from natural language processing to bio-informatics has motivated the research on SSL algorithms.

A closely related concept to semi-supervised learning is *transduction*. Transductive inference refers to reasoning from observed training samples to unlabeled

but observed data, as opposed to *induction* where one aims to extract general rules from observed training samples so as to perform inference on completely novel data. Therefore, if an algorithm is designed to use labeled and unlabeled samples for training, yet if it is limited to assess its performance specifically on unlabeled samples, it is considered to be transductive.

Machine learning techniques can also be categorized as one of the two main paradigms, namely generative and discriminative learning, with respect to their underlying principles. Generative approaches attempt to model $p(x, y)$, the joint distribution of the inputs and outputs whereas discriminative models aim to learn a prediction function directly from the inputs to outputs, e.g., a discriminant function as in SVMs (Bishop, 2006; Schölkopf & Smola, 2001) or conditional probabilities, $p(y|x)$ as in logistic regression. For the purposes of this thesis, we focus on discriminative semi-supervised learning models only. Discriminative learning models have the following advantages over generative models (Bishop, 2006; Bishop & Lasserre, 2007):

- They can incorporate arbitrary feature representations more flexibly.
- Due to the conditional training, they are not affected by any modeling error of the data distribution.

In many learning problems, the output variables have structural or temporal dependencies such as class hierarchies, sequences, lattices or trees (Altun, 2005). When that is the case, we can not predict the outputs in an isolated manner for individual instances. *Structured output prediction* algorithms aim to capture such dependencies, e.g., Structured SVMs (Tsochantaridis et al., 2005) and Conditional Random Fields (Lafferty et al., 2001).

In the context of statistical machine learning, the maximum entropy (MaxEnt) principle (Jaynes, 1957) has long been used in the supervised setting (Berger et al., 1996; Rosenfeld, 1996). Here, one aims to find a distribution p that maximizes an entropy function while the data constraints are met, that is the expected values of some (pre-defined) features with respect to p match their empirical counterparts approximately. Using various entropy measures, model spaces for p or approximation criteria yields a family of discriminative supervised learning methods (e.g., logistic regression, least squares and boosting) including structured output prediction algorithms (e.g., Conditional Random Fields) (Altun & Smola, 2006; Dudík & Schapire, 2006). This framework is known as the *generalized maximum entropy framework*.

This thesis presents a novel semi-supervised approach that incorporates unlabeled data into the generalized maximum entropy framework. Using unlabeled data in the primal MaxEnt objective with conditional probabilities yields multiclass, convex, discriminative loss functions in a principled manner, allowing natural interpretation of the motivation. Moreover, our approach provides an intuitive way of imposing balanced label proportions on labeled and unlabeled samples, which has been successfully used in the earlier semi-supervised learning literature (Collobert et al., 2006; Chapelle & Zien, 2005; Karlen et al., 2008).

1.1 Semi-supervised Learning

Semi-supervised learning (SSL) methods aim to employ unlabeled data together with labeled data to improve performance. The motivation is the scarcity of labeled training samples in real life problems, particularly in situations where labels can not

be generated automatically and/or human effort is required during data collection. An extensive literature survey and a taxonomy of the existing techniques can be found in (Zhu, 2007) and (Chapelle et al., 2006) respectively. We also provide an overview of the relevant SSL algorithms in Section 1.2.

Generally speaking, unlabeled data gives us a better estimate of the marginal data distribution $p(x)$. Accordingly, there has to be some relation between $p(x)$ and the target function that we learn so that we can benefit from unlabeled samples (Seeger, 2001). This anticipation leads to structural assumptions on the geometry of the data. For instance, the intuition behind many of the semi-supervised learning algorithms is that the outputs should be smooth with respect to the structure of the data, i.e., the labels of two inputs that are similar with respect to the intrinsic geometry of data are likely to be the same. Most algorithms perform better on data which conforms to the assumptions they are based on. To date, there is no SSL algorithm that is universally superior (Chapelle et al., 2006). In a real life scenario one has to pick a model that matches the problem structure at hand which is often application specific or data-dependent. The basic structural assumptions that we employ in this thesis, namely the cluster and manifold assumptions and how they are integrated in our framework are discussed in Chapter 3.

Below we give a summary of the important criteria for SSL methods.

Convexity

Convex loss functions are often desirable since they guarantee a unique solution, enable easier theoretical justification and reduce complications such as the need for heuristics to avoid local minima. Many methods from the SSL literature have non-convex loss functions, e.g., transductive Support Vector Machines (TSVM) (Vapnik, 1998), CCCP-TSVM (Collobert et al., 2006),

low density separation (LDS) (Chapelle & Zien, 2005) and transductive neural networks (TNN) (Karlen et al., 2008). As it will become clear in Chapters 4 and 6, our approach yields convex loss functions as it is based on convex duality.

Capability to incorporate prior knowledge

One desirable feature of an SSL algorithm is its capability to integrate prior information or domain knowledge without ad-hoc manipulation. Our method can naturally incorporate such information, e.g., class-proportions or expectations on specific features known a priori by expressing them as constraints in the primal problem.

Generality

The semi-supervised learning methods introduced in this thesis are easily applicable to structured prediction problems (Chapter 4), allows kernelization (Chapter 5) and can be extended as a general framework using various information theoretic measures.

Scalability

The main motivation of SSL is to be able to make use of large amounts of unlabeled data. Therefore, scalability is an immediate concern for semi-supervised algorithms. However, until recently very few methods could scale up to millions of samples especially using non-linear models (Karlen et al., 2008; Fergus et al., 2009; Quadrianto et al., 2009).

1.2 A Survey of Semi-Supervised Learning

In this thesis, we focus on discriminative semi-supervised learning models. A general taxonomy of discriminative SSL methods can be given as follows: semi-supervised and transductive SVM variants, graph-based algorithms, spectral methods, transductive neural networks, information theoretic approaches and constraint based methods.

In the rest of this chapter, we provide an overview of the semi-supervised methods in the literature that are relevant to our work. We aim to investigate the advantages and disadvantages of these algorithms for a better evaluation of the experimental results.

1.2.1 Semi-supervised SVM Variants

Transductive SVM (TSVM)

The supervised Support Vector Machine (SVM) solves the optimization problem below for binary classification. Minimize:

$$\mathbb{R}(\lambda; D) = \gamma \|\lambda\|^2 + \sum_{i=1}^{l} \Delta\left(f(x_i), y_i\right)$$

with hinge-loss,

$$\Delta\left(f(x), y\right) = \max\left(0, 1 - yf(x)\right).$$

where $D = \{(x, y)_{i=1\ldots l}\}$ consists of labeled samples and $f(x) = \langle \lambda, x \rangle + b$. The Transductive Support Vector Machine (TSVM) (Vapnik, 1998) aims to assign labels to the unlabeled samples such that the SVM decision function maximizes the

margin from the separating hyperplane for both labeled and unlabeled samples. However, solving the original TSVM formulation is NP-hard requiring a search over all possible labelings. Accordingly, many heuristics have been proposed to reduce the computational cost of TSVM. In (Bennett & Demiriz, 1998), a mixed integer programming was proposed to find the labeling with the lowest objective function. The optimization, however, is intractable for large data sets. Joachims propose a heuristic that iteratively solves a convex SVM objective function with alternate labeling of unlabeled samples (Joachims, 1999). However, the algorithm is capable of dealing with a few thousand samples only.

TSVM loss can be seen as a regularized extension of SVM

$$\mathbb{R}(\lambda; D) = \gamma\|\lambda\|^2 + \sum_{i=1}^{l} \Delta\left(f(x_i), y_i\right) + \alpha \sum_{i=l}^{l+u} \Delta^*\left(f(x_i)\right).$$

The additional term on unlabeled samples acts as a regularizer that pushes the unlabeled samples far from the decision boundary using the symmetric hinge loss

$$\Delta^*\left(f(x)\right) = \max\left(0, 1 - |f(x)|\right).$$

However, this yields a non-convex objective function which arises difficulties in the optimization procedure. CCCP-TSVM regards the TSVM loss as a sum of convex and concave parts and solves it using a concave-convex procedure (Collobert et al., 2006). This method accommodates up to $60,000$ unlabeled samples in the experiments.

Low Density Separation (LDS)

Chapelle and Zien propose LDS (Chapelle & Zien, 2005) which is a combination of two stages. The former, namely the *graph* SVM, aims to learn an embedding exploiting the cluster assumption. Then, the ∇TSVM solves the TSVM loss using gradient descent in the primal in this new embedding space. The authors also incorporate the following label balancing constraint

$$\frac{1}{u}\sum_{i=1}^{u} f(x_i) = \frac{1}{l}\sum_{i=1}^{l} y_i,$$

enforcing that all unlabeled data are not assigned to a single class.

Laplacian SVM (LapSVM)

Laplacian SVM, optimizes an objective function of the following form (Sindhwani et al., 2005),

$$\mathbb{R}(\lambda; D) = \sum_{i=1}^{l} \Delta(f(x_i), y_i) + \gamma\|\lambda\|^2 + \frac{\alpha}{u}\sum_{i,j=1}^{u} W_{ij}\|f(x_i^*) - f(x_j^*)\|^2,$$

which introduces an additional regularization term defined over both labeled and unlabeled samples reflecting the geometry of the data. Several variations have been proposed for the LapSVM, e.g., by using a sparse manifold regularizer (Tsang & Kwok, 2006).

1.2.2 Transductive Neural Networks

Karlen et al. train transductive neural networks augmented with a manifold regularizer (Karlen et al., 2008). Their (non-convex) objective, which aims to simulta-

neously minimize a loss function and learn an embedding of the unlabeled samples, is given by

$$\mathbb{R}(\lambda; D) = \frac{1}{l}\sum_{i=1}^{l}\Delta\left(f(x_i), y_i\right) + \frac{\lambda}{u^2}\sum_{i,j=1}^{u} W_{ij}\Delta\left(f(x_i^*), y^*(\{i,j\})\right),$$

where

$$y^*(N) = \text{sign}\left(\sum_{k\in N} f(x_k^*)\right),$$

and the weights W_{ij} correspond to the similarity relationships between unlabeled examples x^*. The authors solve this optimization problem using stochastic gradient descent. Therefore, this is a highly scalable online semi-supervised method despite the fact that it is a non-linear model. A balancing constraint is also integrated during training.

1.2.3 Graph Based SSL Algorithms

Graph based SSL methods impose the smoothness assumption over a graph where the nodes represent observations and the edges are associated with weights corresponding to their pairwise similarities. Two commonly used similarity graphs are the k-nearest neighborhood graph ($W_{ij} = 1$ if x_i is among the k-nearest neighbors of x_j or vice-versa and $W_{ij} = 0$ otherwise) and the similarity graph with respect to the RBF kernel

$$W_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right).$$

These graphs have been used in iterative algorithms where each node starts to propagate its label to its neighbors, and the process is repeated until convergence (Zhu & Ghahramani, 2002; Bengio et al., 2006). Algorithm 1 details the steps.[1]

Algorithm 1 Label Propagation (Zhu & Ghahramani, 2002)

Compute affinity matrix W.
Compute the diagonal degree matrix D by $D_{ii} = \sum_j W_{ij}$
Initialize labels $\hat{Y}^{(0)} \leftarrow (y_1, \ldots, y_l, 0, 0, \ldots, 0)$
Iterate

1. $\hat{Y}^{(t+1)} \leftarrow D^{-1}W\hat{Y}^{(t)}$
2. $\hat{Y}^{(t+1)}_{1,\ldots,l} \leftarrow Y_l$

until convergence to $\hat{Y}^{(\infty)}$
Label point x_i by the sign of $\hat{y_i}^{(\infty)}$.

1.2.4 Spectral Methods in Semi-Supervised Learning

In the context of unsupervised data clustering, the spectral properties of the Laplacian matrix have long been used and analyzed in detail (Ng et al., 2001). The Laplacian matrix is defined as follows

$$L = I - D^{-1/2}WD^{-1/2},$$

where W is the similarity matrix and D is the diagonal matrix $D_{ii} = \sum_j W_{ij}$.

Chapelle et al. propose cluster kernels (Chapelle et al., 2003), to enforce the semi-supervised cluster assumption. They modify the eigenspectrum of the Laplacian for the original kernel matrix, $L = I - D^{-1/2}KD^{-1/2}$, where K is computed over both labeled and unlabeled samples, such that the distance induced by the

[1] from (Bengio et al., 2006).

resulting kernel is smaller for samples in the same cluster.

Sinha and Belkin analyze the eigenfunctions of the convolution operator (Sinha & Belkin, 2009), which is the continuous counterpart of the Gram matrix (Schölkopf & Smola, 2001), computed over both labeled and unlabeled data. Motivated by the fact that high density areas correspond to representative eigenvectors when the cluster assumption holds, the authors treat linear combinations of these bases as semi-supervised classifiers.

1.2.5 Information Theoretic Approaches

Various techniques with information theoretic justification have been previously proposed in the SSL literature. Expectation Regularization (ER) (Mann & McCallum, 2007) augments the negative conditional log-likelihood loss with a regularization term, enforcing the model expectation on features from unlabeled data to match either user-provided or empirically computed expectations. The authors provide experimental results for label features minimizing the KL divergence between the expected class distribution and the desired class proportions.

Information regularization (IR) (Grandvalet & Bengio, 2005) minimizes the conditional entropy of the label distribution predicted on unlabeled data, favoring minimal class overlap, along with the negative conditional log-likelihood of the labeled data

$$\mathbb{R}(\lambda; D) = \sum_{i=1}^{l} \log p_\lambda(y_i|x_i) + \gamma \sum_{i=l}^{n} \sum_{y} p_\lambda(y|x_i) \log p_\lambda(y|x_i),$$

where γ is the trade-off parameter to control the impact of unlabeled data. Even though experimental evidence shows high performance, IR is criticized for its sen-

sitivity to hyper-parameter tuning to balance the loss and regularization terms. Furthermore, if the labeled data is very scarce, IR tends to assign all unlabeled data to the same class.

In their information regularization framework, Szummer and Jaakola enforce that the labels should be uniform in regions of high density where they regard mutual information as a measure of label complexity (Szummer & Jaakkola, 2002a).

1.2.6 Constraint Driven Semi-supervised Learning

Recently constraint driven SSL approaches have attracted attention, (Bellare et al., 2009; Chang et al., 2007; Liang et al., 2009; Mann & McCallum, 2007). Chang et al. were one of the first to guide semi-supervised algorithms with constraints (Chang et al., 2007). Their model is trained via an EM like procedure with alternating steps. The authors impose constraints on the outputs y rather than the model distribution $p(y|x)$, as proposed in this thesis. They also have a constraint violation mechanism where the hyper-parameters are manually set.

Graca et al. inject auxiliary expectation constraints to the EM algorithm (Graca et al., 2007). The authors replace the E step with I-projection so that the posterior distribution of the latent variables of a graphical model respects the desired constraints deliberately chosen for structured output learning.

Bellare et al. impose expectation constraints on unlabeled data (Bellare et al., 2009). They define an auxiliary distribution that respects general convex constraints and has low divergence with the model distribution. The fundamental difference with our approach is that the authors impose the penalty functions on the dual objective of the MaxEnt framework. This in turn yields a non-convex optimization problem which is solved by alternating projections. In contrast, we

impose constraints on the target distribution directly to the primal problem which yields convex loss functions.

Measurements

Liang et al. propose *measurements* (Liang et al., 2009), a mechanism that allows partial supervision which unifies labels and constraints through constraints. A measurement is the expectation of some function over the outputs of the unlabeled samples, e.g., label proportions or output preferences of the user such as at least 90% of the classes should be classified as category A. This approach allows fully-labeled examples, partially-labeled examples and general constraints on the model predictions to be treated similarly as these can be cast as instances of measurements. The authors define a utility function indicating an expected reward or satisfaction measure for having included a certain type of measurement in the learning process. Using this utility function, they propose a sequential active selection mechanism over a large set of potential measurements. However, the measurement computations become intractable as their expected values require integration over the parameter space and approximate inference methods are required.

The authors expand the maximum entropy objective using additional penalties defined on the measurements and then solve the dual of this extended optimization problem. Our approach shares the principle to enforce constraints on the predicted model distribution using Fenchel's duality and the maximum entropy framework. Yet, we use such constraints to integrate prior information about the geometry of the data over local regions using a similarity metric which can also be interpreted as matching predicted moments of similarity features. Moreover, we analyze the primal-dual relations of model features in RHKS along with similarity

features. Also, our loss functions are tractable and can be solved via gradient descent methods.

Distribution Matching

Quadrianto et al. constrain the learning problem such that the distributions of the predictor functions on labeled and unlabeled data have the same distribution (Quadrianto et al., 2009). They solve a problem of the following form

$$R_{train}[f, X, Y] + \gamma g(f(X), f(X')),$$

where X and X' are sampled from the training and test samples respectively, g refers to a distance function between the two distributions in an RKHS and γ is a balancing term. They choose Maximum-Mean discrepancy which in the case of characteristic kernels define the data distribution uniquely. The authors provide an online approximation which renders the algorithm highly scalable. However, the optimization objective is non-convex and requires elaboration. Also the improvement over supervised learning across various data sets is not consistent.

1.2.7 Semi-supervised Learning in Structured Output Prediction

Even though semi-supervised learning has been a very active field for over a decade, research on semi-supervised structured prediction is relatively recent. Mann and McCallum extend their Expectation Regularization method on standard data (Mann & McCallum, 2007), which has been mentioned previously in Section 1.2.5, to linear-chain CRFs in (Mann & McCallum, 2008). The authors use partially

labeled sequences and enforce the expectations to hold on individual features between the target values $\hat{\psi}$ and the model expectations $E[\psi(x)]$.

For the semi-supervised training of CRFs, Jiao et al. (Jiao et al., 2006) use the information regularization (IR) method by Grandvalet and Bengio (Grandvalet & Bengio, 2005) mentioned earlier in Section 1.2.5. The following constraint driven SSL approaches previously mentioned in Section 1.2.6 also apply to CRFs (Bellare et al., 2009; Chang et al., 2007; Graca et al., 2007; Liang et al., 2009; Quadrianto et al., 2009).

Other than semi-supervised CRFs, some previous work focus on combining semi-supervised kernels and standard algorithms for structured output prediction. Altun et al. propose a max-margin semi-supervised classification method (with hinge loss) for linear-chain sequences using nonlinear graph-kernels (Altun et al., 2006). Note that these methods have scalability problems.

Chapter 2

Background

2.1 Basics

2.1.1 Maximum Entropy and Maximum Likelihood

Probability density estimation using maximum entropy dates back to the late 1950s. Jaynes was the first to point out a correspondence between statical inference and information theory (Jaynes, 1957). He postulated the maximum entropy principle which states that, given some testable information on a distribution, e.g., in terms of empirical observations, the true distribution is the one that maximizes the entropy among all the distributions that conform to the available information, or equivalently, satisfy the desired constraints pertaining to this information. In other words, the best possible estimate is the one that is as uninformative as possible apart from the information we are already given and thus does not make any assumptions about what we do not know.

In statistical inference, the testable information is provided in terms of real valued feature functions for a given sample and we aim to find the best distri-

bution that explains the data given the constraints on the expectations of these features. For the supervised learning scenario, a convex-dual solution of MaxEnt using Lagrange Multipliers gives the Gibbs distribution. In the machine learning community, the existence of a primal-dual relation between maximum likelihood Gibbs distribution and maximum entropy has long been known (Pietra et al., 1997). Before proceeding further, we introduce some basic concepts in information theory that will be used in the following sections.

Definition 1 *Let X be a discrete random variable with alphabet $\mathcal{X}$ and with probability mass distribution $p(x) = Pr\{X = x\}, x \in \mathcal{X}$.* ***Shannon's Entropy*** *(Cover & Thomas, 2006) is given by*

$$\mathbb{H}(X) \stackrel{\text{def}}{=} -\sum_{x \in \mathcal{X}} p(x) \log p(x). \tag{2.1}$$

Definition 2 ***Conditional Entropy*** *quantifies the uncertainty of a random variable Y given that the value of a second random variable X is known,*

$$\begin{aligned} \mathbb{H}(Y|X) &\stackrel{\text{def}}{=} \sum_{x \in \mathcal{X}} p(x)\, \mathbb{H}(Y|X = x), \\ &= -\sum_{x \in \mathcal{X}} p(x) \sum_{y \in \mathcal{Y}} p(y|x)\, \log\, p(y|x). \end{aligned} \tag{2.2}$$

Definition 3 ***Csiszár Divergence*** *A divergence is a function $\mathbb{D}(p \parallel q)$ that measures the difference between two probability distributions p and q. Let p and q be two probability distributions over a space $\mathcal{P}$. Then, for a convex function f such that $f(1) = 0$, the Csiszár or f-divergence of q from p is*

$$\mathbb{D}_f(p \parallel q) = \int_{\mathcal{P}} f\left(\frac{dp}{dq}\right)\, dq \tag{2.3}$$

Different f functions lead to different divergence measures. For instance, taking $f(x) = x\ln(x)$ gives the special case of Kullback-Leibler (KL) divergence for the discrete case

$$\mathbb{D}_{\mathrm{KL}}(p \parallel q) = \sum_x p(x) \ln \frac{p(x)}{q(x)}. \tag{2.4}$$

Definition 4 (Convex Conjugate) *Denote $\mathcal{X}$ to be a Banach space and let $\mathcal{X}^*$ be its dual. The convex conjugate or the Legendre-Fenchel transformation of a function $h : \mathcal{X} \to \Re$ is $h^* : \mathcal{X}^* \to \Re$ where h^* is defined as*

$$h^*(x^*) \stackrel{\text{def}}{=} \sup_{x \in \mathcal{X}} \{\langle x, x^* \rangle - h(x)\}.$$

2.1.2 Relation of MaxEnt Regularization and Priors

The duality between maximum entropy and log-likelihood has been well-known for decades. In its simplest form, the MaxEnt objective is to find a distribution that satisfies moment matching constraints, yielding an optimization problem of the following form

$$\max_{p \in \mathcal{P}} \mathbb{H}(Y|X) = \min_{p \in \mathcal{P}} \sum_x \tilde{\pi}(x) \sum_y p(y|x) \log p(y|x)$$

such that

$$\begin{aligned} \forall_i \quad & E_{x \sim \tilde{\pi}(x)} E_{y \sim p(y|x)}[\psi_i(x,y)] = E_{x,y \sim \tilde{\pi}(x,y)}[\psi_i(x,y)], \\ & \sum_y p(y|x) = 1. \end{aligned} \tag{2.5}$$

Equality constrained MaxEnt via Lagrangian Duality

In order to show the equivalence of the problem given by Equation (2.5) and log-likelihood maximization, we are going to derive the Lagrangian dual of this problem. The Lagrangian corresponding to Equation (2.5) is given by,

$$\begin{aligned}\mathcal{L}(p,\lambda,\gamma;D) = &\sum_x \tilde{\pi}(x) \sum_y p(y|x) \log p(y|x) \\ &+ \sum_i \lambda_i \left(E_{x,y\sim\tilde{\pi}(x,y)}[\psi_i(x,y)] - E_{x\sim\tilde{\pi}(x)} E_{y\sim p(y|x)}[\psi_i(x,y)] \right) \\ &+ \gamma_x (\sum_y p(y|x) - 1).\end{aligned}$$

In the equation above $p(y|x)$ is the primal variable and λ_is are the Lagrange multipliers corresponding to the moment matching constraints.[1]

Solving $\partial\mathcal{L}(p,\lambda,\gamma;D)/\partial p(y|x) = 0$ gives us optimal p^*

$$p^*(y|x) = \frac{\exp(<\lambda, \psi(x,y)>)}{Z(x;\lambda)} = \frac{\exp(F(x,y;\lambda))}{Z(x;\lambda)}. \tag{2.6}$$

Plugging p^* back in $\mathcal{L}$, we obtain the Lagrange dual,

$$\begin{aligned}\mathbb{Q}(\lambda;D) = &\sum_x \tilde{\pi}(x) \sum_y p^*(y|x) \log p^*(y|x) \\ &+ \sum_i \lambda_i \left(E_{x,y\sim\tilde{\pi}(x,y)}[\psi_i(x,y)] - E_{x\sim\tilde{\pi}(x)}[\sum_y p^*(y|x)\psi_i(x,y)] \right) \\ &= -\sum_x \tilde{\pi}(x) \sum_y \frac{\exp(F(x,y;\lambda))}{Z(x)} \log Z(x;\lambda) + \sum_i E_{x,y\sim\tilde{\pi}(x,y)}[\psi_i(x,y)\lambda_i] \\ &= -\frac{1}{N}\sum_{j=1}^{N} \log(\sum_y \exp\langle\lambda, \psi(x_j,y)\rangle) + \frac{1}{N}\sum_{j=1}^{N} \langle\lambda, \psi(x_j,y_j)\rangle . \end{aligned} \tag{2.7}$$

[1]Refer to Table A.1 for further details on the notation.

The dual objective given by Equation (2.7) is equivalent to the conditional log-likelihood

$$\mathbb{L} \stackrel{\text{def}}{=} \log \prod_{j=1}^{N} p_\lambda(y_j|x_j). \tag{2.8}$$

Inequality constrained MaxEnt via Lagrangian Duality

Over-fitting is a common issue for problems with high dimensional features especially when the sample size is small. Among several remedies one is to use relaxed constraints rather than enforcing them to hold exactly as in Equation (2.5), since this is often an unrealistic goal for real-world data. Kazama et al. proposed a relaxation of MaxEnt using inequality constraints (Kazama & Tsujii, 2005). This form corresponds to an l_1 regularized version of the maximum likelihood objective.

MaxEnt objective with inequality type constraints is below

$$\begin{aligned} &\max_{p\in\mathcal{P}} \mathbb{H}(Y|X) \\ &= \min_{p\in\mathcal{P}} \sum_x \tilde{\pi}(x) \sum_y p(y|x) \log p(y|x) \end{aligned}$$

such that

$$\begin{aligned} \forall_i \quad & |E_{x\sim\tilde{\pi}(x)} E_{y\sim p(y|x)}[\psi_i(x,y)] - E_{x,y\sim\tilde{\pi}(x,y)}[\psi_i(x,y)]| < \epsilon \\ \forall_x \quad & \sum_y p(y|x) = 1\,. \end{aligned} \tag{2.9}$$

The Lagrangian for the problem is given by

$$\begin{aligned}\mathcal{L}_{(p,\lambda^+,\lambda^-,\gamma;D)} = &\sum_x \tilde{\pi}(x) \sum_y p(y|x) \log p(y|x) \\ &+ \sum_i \lambda_i^+ \left(\sum_x \tilde{\pi}(x,y)\psi_i(x,y) - \sum_x \tilde{\pi}(x) \sum_y p(y|x)\psi_i(x,y) - \epsilon \right) \\ &+ \sum_i \lambda_i^- \left(-\sum_x \tilde{\pi}(x,y)\psi_i(x,y) + \sum_x \tilde{\pi}(x) \sum_y p(y|x)\psi_i(x,y) - \epsilon \right) \\ &+ \gamma_x (\sum_y p(y|x) - 1) \,.\end{aligned}$$

Here, the need for two sets of constraints arises due to the absolute value on the inequality constraints. We obtain $p^*(y|x)$ by differentiating the Lagrangian with respect to the primal variable and setting $\partial\mathcal{L}(p, \lambda^+, \lambda^-, \gamma; D)/\partial p(y|x) = 0$. Plugging the optimal $p^*(y|x)$, which is of the same form as given in Equation (2.6), we derive the following dual objective

$$\begin{aligned}\mathbb{Q}_{(\lambda^+,\lambda^-;D)} = &\sum_x \tilde{\pi}(x) \sum_y p^*(y|x) \log \left(p^*(y|x)\right) \\ &+ \sum_i (\lambda_i^+ - \lambda_i^-) \left(\tilde{\psi}_i - \sum_x \tilde{\pi}(x) \sum_y p^*(y|x)\psi_i(x,y) \right) \\ &- \sum_i (\lambda_i^+ - \lambda_i^-)\epsilon_i \,.\end{aligned}$$

With several terms canceling out, this yields the following convex dual

$$\mathbb{Q}_{(\lambda^+,\lambda^-;D)} = -\sum_x \tilde{\pi}(x) \log Z(x; \lambda^+, \lambda^-) + \left\langle \lambda, \tilde{\psi} \right\rangle - \sum_i (\lambda_i^+ - \lambda_i^-)\epsilon_i.$$

which is an l_1 regularized logistic regression loss function when ϵ_i terms are identical. Goodman pointed out that this inequality type relaxation is equivalent to a

Laplacian prior (Goodman, 2004) on the estimated distribution. Earlier, Chen et al. showed that l_2^2 regularization corresponds to Gaussian prior (Chen & Rosenfeld, 2000). However, unlike l_1 regularization an l_2^2 regularized dual corresponds to penalty functions in the primal problem instead of inequality constraints. We will need Fenchel's duality to show that, as will be introduced in the next section.

Dudík et al. provide a theoretical analysis of the relaxed formulation with box constraints and in later work (Dudík et al., 2004), the same authors propose a general treatment for l_1, l_2, l_2^2 and $l_1 + l_2^2$ style regularization (Dudík & Schapire, 2006). Also an analysis for l_p norm is provided by (Friedlander & Gupta, 2006).

2.1.3 Generalized Maximum Entropy

When the target distribution is defined on a finite dimensional space and with specific forms of the constraints, the maximum entropy problem can be solved using Lagrangian techniques. See (Kazama & Tsujii, 2005; Dudík et al., 2004). However, in the generalized MaxEnt framework with non-differentiable penalty functions as proposed by (Dudík & Schapire, 2006) or with infinite dimensional spaces as (Altun & Smola, 2006) pointed out, we need Fenchel's duality for a proper analysis of the primal-dual space relations. This section briefly introduces the key concepts related to Fenchel's duality sufficient to follow the rest of this thesis. For a broader introduction to Fenchel's duality for the machine learning audience and a detailed reference the reader may refer to (Rifkin & Lippert, 2007) and (Rockafellar, 1996) respectively.

Definition 5 ***Core*** *The core of a set C,* $\mathrm{core}(C)$ *is the set of points x in C such that for any direction d in an arbitrary Euclidean space, $\mathbf{E}$, $x + td$ lies in C for all*

small t. This set contains the interior of C, although it may be larger, (Borwein & Lewis, 2006).

Theorem 1 (Fenchel's Duality, Theorem (4.4.3) of (Borwein & Zhu, 2005)) *Let $\mathcal{X}$ and $\mathcal{X}^*$ be Banach spaces, $f : \mathcal{X} \to \Re \cup \{+\infty\}$ and $g : \mathcal{X}^* \to \Re \cup \{+\infty\}$ be convex functions and $A : \mathcal{X} \to \mathcal{X}^*$ be a bounded linear map. Define t and d as follows* [2]*,*

$$t = \inf_{x \in \mathcal{X}} \{f(x) + g(Ax)\} \text{ and}$$
$$d = \sup_{x^* \in \mathcal{X}^*} \{-f^*(A^*x^*) - g^*(-x^*)\}.$$

Assume that f, g and A satisfy one of the following constraint qualifications,

1. $0 \in \operatorname{core}(\operatorname{dom} g - A \operatorname{dom} f)$ *and both f and g are lower semi continuous,*
2. $A \operatorname{dom} f \cap \operatorname{cont} g \neq \emptyset$*,*

where $s \in \operatorname{core}(S)$ if $\bigcup_{\lambda>0} \lambda(S - s) \subseteq \mathcal{X}$, $\mathcal{X}$ is a Banach space and $S \subseteq X$. Then $t = d$, where the dual solution d is attainable if it is finite.

In our context A is an observation operator, e.g., a map from distributions into a set of moments. The generalized maximum entropy framework incorporates various forms of constraints and penalty functions as a potential, $h : \Re \to (-\infty, \infty]$ to the maximum entropy objective. (Dudík, 2007) provides a theoretical analysis of the generalized maxent for marginal distributions in supervised settings. In this thesis however, we focus on loss functions for conditional distributions in the semi-supervised setting and their empirical evaluation. Therefore, in Section 2.2, we reformulate the maxent objective for conditional distributions.

[2] The adjoint transformation A^* is given by $\langle Ap, \lambda \rangle = \langle A^*\lambda, p \rangle$.

2.2 Duality of Maximum Entropy for Conditional Distributions

In this section, we outline a brief summary of duality relation between generalized Maximum Entropy on class conditional distributions and various supervised discriminative learning methods[3]. We focus on modeling conditional distributions given by

$$\mathcal{P} = \{p \,|\, p(y|x) \geq 0,\ \sum_{y \in \mathcal{Y}} p(y|x) = 1,\ \forall x \in \mathcal{X},\ y \in \mathcal{Y}\},$$

where $\mathcal{Y}$ and $\mathcal{X}$ are output and input spaces respectively.

The goal in generalized MaxEnt is to minimize the divergence of the target distribution p from a reference distribution q while penalizing the discrepancy between observed values $\tilde{\psi}$ of some pre-defined feature functions $\psi : \mathcal{X} \times \mathcal{Y} \to \mathcal{B}$ and their expected values with respect to the target distribution. Here, $\tilde{\psi}$ can be derived from a sample, e.g., $\tilde{\psi} = 1/n \sum_{i=1}^{n} \psi(x_i, y_i)$. The conditional expectation is defined as

$$\mathbb{E}_p[\psi] \stackrel{\text{def}}{=} \sum_x \tilde{\pi}(x) \mathbb{E}_{y \sim p(y|x)}[\psi(x, y)]. \tag{2.10}$$

Hence, the conditional expectation operator imposes a weighting with respect to

[3]We use entropy maximization and divergence minimization interchangeably since they are equivalent up to a constant for a fixed reference distribution.

the marginal distribution of $\tilde{\pi}(x)$ as shown below

$$\mathbf{Ep} = \begin{bmatrix} \tilde{\pi}(x_1)\psi_1(x_1,y_1) & \dots & \tilde{\pi}(x_n)\psi_1(x_n,y_c) \\ \vdots & \ddots & \\ \tilde{\pi}(x_1)\psi_i(x_1,y_1) & \dots & \tilde{\pi}(x_n)\psi_i(x_n,y_c) \\ \vdots & \ddots & \\ \tilde{\pi}(x_1)\psi_d(x_1,y_1) & \dots & \tilde{\pi}(x_n)\psi_d(x_n,y_c) \end{bmatrix} \begin{bmatrix} p(y_1|x_1) \\ \vdots \\ p(y_c|x_1) \\ \vdots \\ p(y_1|x_n) \\ \vdots \\ p(y_c|x_n) \end{bmatrix}$$

$$= \begin{bmatrix} E_{\sim\tilde{\pi}(x)}E_{\sim p(y|x)}\psi_1(x,y) \\ \vdots \\ E_{\sim\tilde{\pi}(x)}E_{\sim p(y|x)}\psi_i(x,y) \\ \vdots \\ E_{\sim\tilde{\pi}(x)}E_{\sim p(y|x)}\psi_d(x,y) \end{bmatrix}, \tag{2.11}$$

where d refers to the dimensionality of the feature space. The following lemma shows the duality of generalized MaxEnt for conditional distributions and various discriminative supervised learning methods.

Lemma 2 (MaxEnt Duality for conditionals) *Let $p, q \in \mathcal{P}$ be conditional distributions and $\mathbb{D}$ be a divergence function that measures the discrepancy between two distributions*

$$\mathbb{D}(p|q) = \sum_x \tilde{\pi}(x)\mathbb{D}_x\left(p_x|q_x\right). \tag{2.12}$$

Moreover, let $\psi : \mathcal{X}\times\mathcal{Y} \to \mathcal{B}$ be a feature map to a Banach space $\mathcal{B}$ (with dual space $\mathcal{B}^*$), g be lower semi-continuous (lsc) convex and $\mathbb{E}_p$ is the conditional expectation

operator in Equation (2.10)*. Define*

$$t := \min_{p \in \mathcal{P}} \{\mathbb{D}(p|q) + g\left(\mathbb{E}_p[\psi]; \tilde{\psi}, \epsilon\right)\}, \tag{2.13}$$

$$d := \max_{\lambda \in \mathcal{B}^*} \{-\sum_x \tilde{\pi}(x) \mathbb{D}^*_x(\langle \psi(x, .), \lambda \rangle); q_x) - g^*(\lambda; \tilde{\psi}, \epsilon)\}, \tag{2.14}$$

where q is a reference distribution (reflecting the prior knowledge for target distribution). Then, $d = t$.

Proof Let $f_q(p) = \mathbb{D}(p|q)$, $A_x p_x = \mathbb{E}_{p_x}[\psi]$ and $Ap = \mathbb{E}_p[\psi]$. Fenchel's Duality (Borwein & Zhu, 2005, Theorem (4.4.3)) states that

$$\inf_{p \in \mathcal{P}} \{f_q(p) + g(Ap)\} = \sup_{\lambda \in \mathcal{B}^*} \{-f^*(A^*\lambda) - g^*(-\lambda)\}$$

via strong duality. For the expectation operator,

$$\left\langle \sum_x \tilde{\pi}(x) \sum_y p(y|x) \psi(x, y), \lambda \right\rangle = \sum_x \tilde{\pi}(x) \sum_y p(y|x) \langle \psi(x, y), \lambda \rangle = \sum_x \tilde{\pi}(x) \langle A^*_x \lambda, p_x \rangle$$

for $A^*_x \lambda = \langle \lambda, \psi(x, .) \rangle$. Then,

$$\begin{aligned} f^*(A^*\lambda) &= \sup_p \{\langle p, A^*\lambda \rangle - f(p)\} \\ &= \sup_{\{p_x\}} \{\sum_x \tilde{\pi}(x) \langle A_x p_x, \lambda \rangle - \sum_x \tilde{\pi}(x) f(p_x)\} \\ &= \sum_x \tilde{\pi}(x) \sup_{p_x} \{\langle A^*_x p_x, \lambda \rangle - f(p_x)\}, \end{aligned}$$

for independent x. This is in turn equal to $\sum_x \tilde{\pi}(x) f^*(A^*_x \lambda)$. Plugging values to Fenchel's duality completes the proof. ■

$h_1(b;a) = \int_t b(t) \ln b(t)/a(t)$	$h_1^*(b^*;a) = \int_t a(t) \exp(b^*(t) - 1)$
$h_2(b;a,\epsilon) = \mathbb{I}(\|b-a\|_{\mathcal{B}} \leq \epsilon)$	$h_2^*(b^*;a,\epsilon) = \epsilon\|b^*\|_{\mathcal{B}^*} + \langle b^*, a\rangle$
$h_3(b;a,\epsilon) = \|b-a\|_{\mathcal{B}}^2/(2\epsilon)$	$h_3^*(b^*;a,\epsilon) = \epsilon\|b^*\|_{\mathcal{B}^*}^2/2 + \langle b^*, a\rangle$

Table 2.1: Examples of convex conjugacy used in this thesis are KL divergence, approximate norm constraints and and norm-square penalty functions.

2.2.1 A Unified MaxEnt Framework

Altun and Smola show that divergence measures other than KL-divergence lead to various algorithms and provide a framework to unify divergence minimization and statistical inference (Altun & Smola, 2006). Therefore, when p is the conditional distribution of an output variable $y \in \mathcal{Y}$ given the input $x \in \mathcal{X}$ in (2.13), the dual problems (2.14) correspond to various discriminative learning methods.

Important special cases for classification are listed below:

1. **Logistic regression** Taking p as the conditional probability distribution, $\mathbb{D}$ as KL divergence and plugging in the Fenchel's duality machinery gives the special case of logistic regression loss

$$\mathbb{R}(\lambda; D) = \sum_{j=1}^{n} \tilde{\pi}(x_j) \log \sum_{y} \exp\left(\langle \lambda, \psi(x_j, y)\rangle\right) - \sum_{j=1}^{n} \tilde{\pi}(x_j, y_j) \langle \lambda, \psi(x_j, y_j)\rangle + \Omega(\lambda), \tag{2.15}$$

where $\Omega(\lambda)$ is given by $\|\lambda\|_{\mathcal{B}}$ when $g = h_2$, and $\|\lambda\|_{\mathcal{B}}^2$ if $g = h_3$ from Table 2.1.

The relation between the primal p and dual λ variables is given by

$$p(y|x;\lambda) \propto \exp(\langle \lambda, \psi(x,y) \rangle). \tag{2.16}$$

2. **Kernel logistic regression**

   When $\mathbb{D}$ is KL divergence, p is a conditional probability distribution, $g = h_3$ (see Table 2.1) and $\mathcal{B}$ is a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$, the convex dual of the MaxEnt objective gives kernel logistic regression (KLR). The kernel K defining $\mathcal{H}$ is given by

$$k(x,y,x',y') = \delta(y,y')K(x,x') = \langle \psi(x,y), \psi(x',y') \rangle ,$$

   where $\psi(x,y)$ is the kernel induced (joint) feature space of possibly infinite dimensionality. Representer Theorem (Schölkopf & Smola, 2001) states that each minimizer of (2.15) admits the form

$$\lambda^* = \textstyle\sum_{i=1}^{n} \sum_y \alpha_{i,y} \psi(x_i, y). \tag{2.17}$$

   Accordingly, when we substitute the solution given by Equation (2.17) to (2.15), we obtain the KLR loss

$$\mathbb{R}(\alpha; D) = \sum_{i=1}^{n} \tilde{q}_m(x_i) \log \sum_y \exp(F(x_i, y; \alpha)) - \frac{1}{n} \sum_{i=1}^{n} F(x_i, y_i; \alpha) + \epsilon\, \alpha^T K \alpha, \tag{2.18}$$

   where $F(x,y) = \langle \lambda^*, \psi(x,y) \rangle$ for λ^* defined above.

3. **Boosting** when p is an unnormalized conditional distribution (UCD) and $\mathbb{D}$

is KL divergence. See (Collins et al., 2000).

4. A family of structured prediction algorithms, e.g., Conditional Random Fields (Lafferty et al., 2001), kernel Conditional Random Fields (Lafferty et al., 2004), and Boosted Random Fields (Torralba et al., 2005) if feature functions ψ decompose with respect to a graphical model and the conditions in items 1 or 3 hold respectively. See Section 5.3 for the convex dual derivations of the l_2^2 regularized linear chain CRFs.

5. **Least squares** when p is a UCD and $\mathbb{D}$ is l_2 divergence. Kernel least squares, when $\mathcal{B}$ is RKHS. See (Altun & Smola, 2006).

6. **Support Vector Machines** when p is a UCD, $\mathbb{D}$ is the total variation distance and $\mathcal{B}$ is RKHS. See (Altun & Smola, 2006).

Chapter 3

A Word on Similarity

3.1 Introduction

In both supervised and semi-supervised learning exploiting the similarities among pairs of samples is a common approach. Similarity (or inversely distance) metrics vary from one application domain to another. Depending on the application domain, one can come up with various similarity relations among instances such as the WordNet distance which is a semantic distance for English words (Fellbaum, 1998), edit distance for strings, cosine similarity for high dimensional sparse binary vectors, the cardinality of the intersection of two sets, etc. On the other hand, it may also not always be possible to treat all the attributes of an instance uniformly. In such situations, composite distance metrics may be required. For instance, the visual descriptors for a robot's grasp configuration (see Section 7.6.2) are comprised of attributes corresponding to the location of the actuator and quaternions for its rotation. Accordingly, the similarity metric we use is a combination of Euclidean and angular distances.

Similarities can be incorporated in the learning process as features or one can regard them as functions embedding the model features into a reproducing kernel Hilbert space (RKHS) (Schölkopf & Smola, 2001). Chen et al. provide an overview of classification algorithms using similarities as features and compare the use of *similarity features* versus kernels for various kernel based discriminative algorithms (Chen et al., 2009). The fundamental difference between these two approaches is that the former searches for a solution in the Euclidean space whereas the latter maps the inputs to the associated RHKS and requires regularization in this RHKS. Other than the regularizer, the objective functions are essentially the same.

Note that not all similarity definitions correspond to positive definite kernels. Indefinite kernel matrices require careful handling as they might lead to non-convex optimization problems. See (Chen et al., 2009) for remedies to overcome this technicality. Balcan analyzes theoretical properties of discriminative semi-supervised algorithms with similarity features (Balcan, 2008).

In this thesis, we utilize similarity functions with two different approaches. In Chapter 4, we incorporate similarity features into the learning process and impose additional penalties on the expectations of these features to the generalized MaxEnt objective. Alternatively, in Chapter 6, we use similarities to get better estimates of empirical feature expectations, i.e., instead of imposing additional constraints, we improve the existing constraints on model features. Although based on different motivations, both approaches use various forms of the *smoothness assumption*.

3.2 Prior Knowledge on Intrinsic Data Geometry

For the purposes of this thesis, we focus on the fundamental assumption on the relation between the outputs of the learned model and the geometry of the training samples, which is commonly referred to as smoothness assumption. Generally speaking, the smoothness assumption states that if two points x_1, x_2 are close, their outputs y_1, y_2 should be close as well. For the semi-supervised setting this is a bit more specific; if x_1, x_2 in a high-density region are close, so are y_1 and y_2. In other words, the outputs are smoother in denser regions. The question though, of how to determine whether x_1, x_2 are close arises naturally, and the answer requires further elaboration.

The manifold assumption states that the data lies on an embedded lower dimensional non-linear manifold within its actual high dimensional space. The cluster assumption, on the other hand, states that if two data samples are in the same cluster, they are more likely to belong to the same class. Low density separation is a closely related concept which states that a decision boundary preferably goes through low-density regions. The conceptual equivalence between the cluster assumption and low density separation has been observed by Chapelle et al. (Chapelle et al., 2006). One interpretation of low density separation is that such algorithms penalize changes in dense regions. Many discriminative methods such as the transductive SVM and information regularization exploit the low-density separation. On the other hand, graph-based SSL methods (Zhu, 2005) are typically based on the manifold assumption. They utilize the underlying manifold structure by constructing a similarity graph on the entire data and then diffusing the labels with various label propagation algorithms on this graph.

3.3 Defining similarities

In this section, we will describe how the geometry assumptions discussed in the previous section are integrated into the SSL algorithms developed in this thesis.

Cluster Assumption

In Chapter 6, we impose smoothness assumption, by using the similarities to get a weighted average of the attributes of a labeled sample with the unlabeled sample around its vicinity. This way, we get an enhanced estimate of the empirical feature expectations. To achieve this, following other SSL methods, e.g., (Zhu & Ghahramani, 2002; Bengio et al., 2006), we construct a k-nearest neighbor graph over labeled and unlabeled data where the neighbors are restricted to unlabeled data. The edge weights between node x_i and x_j are given by a chosen distance d, e.g., Euclidean distance $d_{ij} = \|x_i - x_j\|$. The similarity is defined by the Gaussian kernel $s(x_i, x_j) = \exp(-d_{ij}^2/\sigma^2)$.

In Chapter 4, similarity functions are defined likewise, however, unlike the formulation given in Chapter 6, they are defined either between pairs of training samples among all data points or local regions centered on both labeled and unlabeled samples.

Manifold Assumption

The manifold assumption enforces two input points that can be connected via a path on the data manifold to have the same label. The shorter the path is, the higher the pairwise similarity becomes. To achieve this, we construct a graph from the data where the nodes correspond to the samples and each edge is associated with a weight equal to the distance between the nodes it connects. Then the

k-nearest neighbors of a labeled instance x_i over the manifold are found using the minimum spanning tree (MST) (Cormen et al., 2001) that connects x_i and k unlabeled nodes in its vicinity. Accordingly the distance d between two points is the sum of the edge weights along the path. The similarity s is defined by the Gaussian kernel with respect to d. Here, one can use other criteria instead of MST, e.g., integrating the volume of paths using Markov random walks (Szummer & Jaakkola, 2002b). We restrict ourselves to this definition for simplicity.

Chapter 4

Semi-supervised Learning via Similarity Constraints

4.1 Introduction

In this chapter, we propose a novel approach to integrate unlabeled data to the entropy maximization problem via additional penalty functions that restrict the model outputs to be consistent within local regions. As discussed in Chapter 3, these local regions can be defined with respect to the assumed data geometry. In this chapter, we investigate two types of penalty functions. *Pairwise penalties* aim to minimize the discrepancy of the conditional class distributions for each sample pair with respect to their proximity. *Expectation penalties*, on the other hand, are a relaxed variant of the former, where the conditional output distribution of an instance is enforced to match the weighted average of the conditional distribution over local regions. The proximity of two samples is defined according to a similarity function that reflects our prior knowledge on the geometry of the data. Augment-

ing the primal maximum entropy problem and applying convex duality techniques yields convex semi-supervised objective functions, which we refer as the dual problems. In particular, we describe two special cases, namely semi-supervised logistic regression and kernel logistic regression in detail.

The rest of the chapter is organized as follows: Section 4.2 provides the details of our approach. An experimental evaluation of these algorithms on benchmark data sets is presented in Section 4.3.1. Comparison to a large number of semi-supervised learning methods shows that our method performs competitively.

4.2 Similarity Constrained Generalized MaxEnt

In semi-supervised learning, we are given a sample D that consists of labeled data $L = \{(x_i, y_i)\}_{i=1}^{l}$ drawn i.i.d. from the probability distribution on $\mathcal{X} \times \mathcal{Y}$ and unlabeled data $U = \{x_i\}_{i=l+1}^{n}$ drawn i.i.d. from the marginal distribution $p(x)$. Throughout this chapter, we focus on multi-class problems where $\mathcal{Y} = \{1, \ldots, C\}$. Hence $\{(x, y)_{i=1\cdots l}, (x)_{i=l+1\cdots n}\}$ denotes all the (labeled and unlabeled) observations in the sample.

If the optimal classification function is smooth with respect to $p(x)$, i.e., the outputs of two similar input points x_j and x_k are likely to be the same, one can utilize unlabeled data points to impose the predictive function to be smooth. Various approaches to enforce this smoothness assumption have lead to a large collection of semi-supervised learning methods. For example, (Sindhwani et al., 2005) implement this assumption by adding a new regularizer

$$\sum_{x_j, x_k} s(x_j, x_k) \sum_{y} (f(x_j, y) - f(x_k, y))^2,$$

to various objective functions where $f(x, y)$ is the predictive function and $\mathrm{s}(x_j, x_k)$ is the similarity between the samples x_j, x_k. With the same motivation, we extend the primal generalized MaxEnt problem to minimize the discrepancy between conditional probability distributions of similar instances. This yields new optimization methods favoring model outputs that are smooth with respect to the underlying marginal distribution.

4.2.1 Pairwise Penalties

One way of encoding the smoothness criteria is by augmenting the supervised MaxEnt problem in (2.13) with a discrepancy for all similar x_j, x_k pairs.

$$t_s := \min_{p \in \mathcal{P}} \{\mathbb{D}(p|q) + g\left(\mathbb{E}_p[\psi]; \tilde{\psi}, \epsilon\right) + \bar{g}(p)\}, \tag{4.1}$$

where

$$\bar{g}(p) = \hat{h}(\sum_{x,x'} h(p_x, p_{x'}))$$

for $h, \hat{h}$ such that $\bar{g}$ is lsc convex.

Corollary 3 *The dual of the semi-supervised MaxEnt objective with pairwise similarities in* (4.10)*, is given by*

$$d_s := \max_{\lambda \in \mathcal{B}^*} \{-g^*(\lambda; \tilde{\psi}, \epsilon) - \sum_x \tilde{\pi}(x)(\mathbb{D} + \bar{g})^*_x(\langle \psi(x, .), \lambda \rangle; q_x)\}. \tag{4.2}$$

The equality of t_s given in Equation (4.10) *and d_s in Equation* (4.2) *follows from Fenchel's duality and Lemma* (2) *by defining $f_q(p) = \mathbb{D}(p|q) + \bar{g}(p)$. Note $(\mathbb{D} + \bar{g})^* =$*

$\mathbb{D}^* \square \bar{g}^*$, *where* $\square$ *denotes the infimal convolution function*[1]. *This term can be solved when* $\mathbb{D}$ *and* g *functions are specified.*

Theoretically, a penalty function can be any convex proper lsc function. However, one should consider efficiency, feasibility and compatibility with the divergence function $\mathbb{D}$ when choosing $\bar{g}(p)$. For instance,

$$\bar{g}(p) = \mathbb{I}\left(s(x_j, x_k)\|p_{x_j} - p_{x_k}\| \leq \epsilon, \forall j, k\right)$$

may lead to infeasible solutions for small ϵ values or may render unlabeled data ineffective for large ϵ values. Adjusting ϵ for each x_j, x_k pair, on the other hand, leads to a very large number of hyper-parameters rendering optimization intractable.

An interesting setting of t_s is when $g = \bar{g}$ is a norm. In this case, the difference between the model outputs weighted with similarities can be written as a linear operator Φ which can then be combined with $\mathbb{E}_p[\psi]$ given in Equation (2.10). Let Φp be the expectation operator over *similarity feature functions* ϕ,

$$\phi_{j,k,y}(x_m, y') = \begin{cases} s(x_m, x_k) & \text{if } x_m = x_j,\ x_j \neq x_k \text{ and } y = y', \\ -s(x_j, x_m) & \text{if } x_m = x_k,\ x_j \neq x_k \text{ and } y = y', \\ 0 & \text{otherwise,} \end{cases} \tag{4.3}$$

for $j, k \in \{1, \ldots, n\}$. Then,

$$(\Phi p)_i = s(x_j, x_k)(p(y|x_j) - p(y|x_k)),$$

[1]The infimal convolution of two functions f and g is defined as

$$(f \,\square\, g)(x) \stackrel{\text{def}}{=} \inf_y \{f(x - y) + g(y) \,|\, y \in \Re^n\}.$$

where $i = (j \times |\mathcal{X}| \times |\mathcal{Y}|) + (k \times |\mathcal{Y}|) + y$, i.e.,

$$\Phi\mathbf{p} = \begin{bmatrix} \phi_1(x_1,y_1) & \dots & \phi_1(x_1,y_c) & \dots & \phi_1(x_n,y_1) & \dots & \phi_1(x_n,y_c) \\ \vdots & \ddots & & & & & \\ \phi_i(x_1,y_1) & \dots & \phi_i(x_1,y_c) & \dots & \phi_i(x_n,y_1) & \dots & \phi_i(x_n,y_c) \\ \vdots & \ddots & & & & & \\ \phi_d(x_1,y_1) & \dots & \phi_d(x_1,y_c) & \dots & \phi_d(x_n,y_1) & \dots & \phi_d(x_n,y_c) \end{bmatrix} \begin{bmatrix} p(y_1|x_1) \\ \vdots \\ p(y_c|x_1) \\ \vdots \\ p(y_1|x_n) \\ \vdots \\ p(y_c|x_n) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_x E_{\sim p(y|x)}\phi_1(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_i(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_d(x,y) \end{bmatrix}$$

$$= \begin{bmatrix} \vdots \\ s(x_j,x_k)(p(y|x_j) - p(y|x_k)) \\ \vdots \end{bmatrix}, \tag{4.4}$$

where $d = |\mathcal{X}| \times |\mathcal{X}| \times |\mathcal{Y}|$ and $c = |\mathcal{Y}|$. Concatenating Φp to $\mathbb{E}_p[\psi]$ and $\mathbf{0}$ vector (of size n^2C) to $\tilde{\psi}$, by Lemma (2), we get the dual of the semi-supervised MaxEnt as

$$d_s := \max_{\lambda,\gamma} \{-g^*((\lambda,\gamma);(\tilde{\psi},\mathbf{0}),\epsilon) - \sum_x \tilde{\pi}(x)\mathbb{D}_x^*(\langle\psi(x,.),\lambda\rangle + \langle\phi(x,.),\gamma\rangle\,;q_x)\}. \tag{4.5}$$

The semi-supervised MaxEnt formulation in (4.10) promotes target distributions that are smooth with respect to the similarity measure $s(x_j, x_k)$ and remains indifferent to distant instance pairs. As mentioned in Chapter 3, s can be defined with respect to the manifold distances in order to impose the manifold assumption, with respect to the Euclidean distances over high density regions in order to impose the smoothness assumption or with respect to data clusters in order to impose cluster assumption. We assume that $s(x_j, x_k) \geq 0, \forall j, k$.

Investigating the difference between the dual supervised and semi-supervised formulations given in Equations (2.14) and (4.5) respectively, we observe that $\mathbb{D}_x^*$ term is evaluated on both labeled and unlabeled data in the semi-supervised case, since the empirical marginal distribution $\tilde{\pi}$ is now measured with respect to D. Furthermore, the expectation term $\mathbb{E}_{p_x}$ is evaluated on the similarity features ϕ as well as the original model features ψ. This requires n^2C additional parameters in the optimization problem, where n is the total size of the data and C is the number of classes.

It is important to note that inconsistent constraints are likely to occur, as similar samples might have different labels especially when they are close to decision boundaries. To alleviate the conflicting constraints, we prefer to relax the restrictions on the model probabilities with penalty functions instead of box constraints.

The increase in the number of parameters may be prohibitively expensive for very large data sets. One solution to this problem is to define a sparse similarity function as the parameters for x_j and x_k become redundant if $\mathrm{s}(x_j, x_k) = 0$. Hence, the number of parameters can be reduced significantly. We now present two special cases of (4.5), namely Pairwise Semi-Supervised Logistic Regression and Pairwise Semi-Supervised Kernel Logistic Regression.

Semi-Supervised Logistic Regression with Pairwise Similarity Penalty

The semi-supervised logistic regression with ℓ_2^2 regularization for pairwise semi-supervised penalty can be derived by setting the divergence function to KL, $\mathbb{D}_x = h_1$ with uniform q, and g to norm-square penalty function h_3 (See Table 2.1)

$$\min_{p \in \mathcal{P}} \mathrm{KL}(p||q) + \frac{1}{\epsilon}\|\tilde{\psi} - \mathbb{E}_p[\psi]\|_2^2 + \frac{1}{\epsilon}\|\Phi p\|_2^2. \tag{4.6}$$

Note that

$$\|\Phi p\|_2^2 = \sum_{j,k} \sum_{y} \left(s(x_j, x_k)(p(y|x_j) - p(y|x_k))\right)^2, \tag{4.7}$$

as shown in Equation (4.4). Using Corollary (3), we plug the convex conjugates of the corresponding functions to Equation (4.5) and negate the result. This gives us the minimization problem of

$$\mathbb{R}(\lambda, \gamma; D) = \sum_{x \in D} \tilde{\pi}(x) \log Z_x(\lambda; \gamma) - \left\langle \lambda, \tilde{\psi} \right\rangle + \epsilon\|\lambda\|_2^2 + \epsilon\|\gamma\|_2^2, \tag{4.8}$$

where

$$Z_x(\lambda, \gamma) = \sum_{y} \exp\left(F(x, y; \lambda, \gamma)\right),$$

$$F(x, y; \lambda, \gamma) = \langle \lambda, \psi(x, y) \rangle + \sum_{\hat{x}} \mathrm{s}(\hat{x}, x)\gamma_{\hat{x}xy} - \sum_{\bar{x}} \mathrm{s}(x, \bar{x})\gamma_{x\bar{x}y},$$

and the relation between the primal parameter p and the dual parameters λ, γ is given by

$$p(y|x) = \exp(F(x, y))/Z_x. \tag{4.9}$$

Here $\mathbb{R}(\lambda, \gamma; D)$ is no longer the negative log-likelihood term. First, there is no inner product term on similarity parameters. Second, the log-partition function is computed for both labeled and unlabeled data. The similarity terms in F can be seen as a flow problem, where the weighted average of incoming flow from neighbors $s(\hat{x}, x)\gamma_{\hat{x}xy}$ is matched to the outgoing flow $\mathrm{s}(x, \bar{x})\gamma_{x\bar{x}y}$.

It is important to note that $p(y|x)$ is well-defined for all x, hence it can be applied to out-of-sample data. From this perspective, this is a proper semi-supervised learning method. However, for out-of-sample data the similarity features are all 0 according to the similarity feature definition given in Equation (4.3). Hence, the penalty function remains ineffective for these instances. From this perspective, this is a transduction method since the performance is expected to improve from supervised to semi-supervised optimization only on the in-sample unlabeled data.

The gradients of the objective function with respect to the dual variables are given by

$$\frac{\partial\, \mathbb{R}(\lambda, \gamma; D)}{\partial \lambda} = \mathbb{E}_{p_x}[\psi(x, y)] - \tilde{\psi} + 2\epsilon\lambda \,,$$

and

$$\frac{\partial\, \mathbb{R}(\lambda, \gamma; D)}{\partial\, \gamma_{\hat{x},\bar{x},y}} = -\, p(y|\bar{x})\, \mathrm{s}(\hat{x}, \bar{x}) + p(y|\hat{x})\, \mathrm{s}(\hat{x}, \bar{x}) + 2\epsilon\gamma_{\hat{x},\bar{x},y} \,.$$

Any gradient based optimization method that minimizes $\mathbb{R}$ can be applied to find λ, γ. In practice, we use the quasi-Newton method BFGS.

Semi-Supervised Kernel Logistic Regression with Pairwise Penalty

In order to treat the penalty functions on model features and similarity features uniformly in our MaxEnt objective

$$t_s := \min_{p \in \mathcal{P}} \{\mathbb{D}(p|q) + g\left(\mathbb{E}_p[\psi]; \tilde{\psi}, \epsilon\right) + \bar{g}\left(\mathbb{E}_p[\phi]; \epsilon\right), \tag{4.10}$$

we define the following combined penalty function U that acts on the concatenation of these two feature spaces

$$U = \|Ax - b\|_2^2 = \| \begin{bmatrix} \mathbb{E}_p[\psi]p - \tilde{\psi} \\ \mathbb{E}_p[\phi]p - \mathbf{0} \end{bmatrix} \|_2^2.$$

When each of the individual feature spaces $\xi_1 = \psi$ and $\xi_2 = \phi$ are defined on RKHS's $\mathcal{H}_1$ and $\mathcal{H}_2$ (with kernels K_1 and K_2 respectively), U takes the following form

$$U = \| \begin{bmatrix} \mathbb{E}_p[\psi]p - \tilde{\psi} \\ \mathbb{E}_p[\phi]p - \mathbf{0} \end{bmatrix} \|_{\mathcal{H}}^2$$

where $\mathcal{H} = (\mathcal{H}_1, \mathcal{H}_2)$ with kernel $K = K_1 + K_2$ (Bach et al., 2004; Sonnenburg et al., 2006). For our similarity constrained formulations, $\mathcal{H}_1$ is any RKHS where the model features are mapped to and $\mathcal{H}_2$ is the linear RKHS where the similarity features are defined. We can restate our primal and dual objectives accordingly as

below

$$t = \inf_{x \in \mathcal{X}} \{\mathbb{D}(p|q) + \| \begin{bmatrix} \mathbb{E}_p[\psi]p - \tilde{\psi} \\ \mathbb{E}_p[\phi]p - \mathbf{0} \end{bmatrix} \|_{\mathcal{H}}^2\} \text{ and}$$

$$d = \sum_{x \in \mathcal{D}} \tilde{\pi}(x) \log \sum_y \exp\left(\langle \lambda, \psi(x,y) \rangle + \langle \gamma, \phi(x,y) \rangle\right) - \left\langle \lambda, \tilde{\psi} \right\rangle + \epsilon \| \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} \|_{\mathcal{H}}^2.$$

For the ease of notation, we combine the feature spaces and rename it as ξ, also rename the joint vector of optimization parameters as β

$$d = \sum_{x \in \mathcal{D}} \tilde{\pi}(x) \log \sum_y \exp\left(\langle \beta, \xi(x,y) \rangle\right) - \left\langle \beta, \tilde{\xi} \right\rangle + \epsilon \|\beta\|_{\tilde{\mathcal{H}}}^2 \tag{4.11}$$

where $\tilde{\xi} = \begin{bmatrix} \tilde{\xi}_1 \\ 0 \end{bmatrix}$. Representer theorem states that our optimal β^* is of the following form,

$$\beta^* = \sum_{x=1}^{N} \sum_y \alpha_{xy} \xi(x,y). \tag{4.12}$$

Plugging this optimal β^* back into Equation (4.11) we get

$$\begin{aligned} \mathcal{Q}(\alpha; D) = & \sum_{j=1}^{N} \tilde{\pi}(x) \log \sum_{y'} \exp \left\langle \sum_{i=1}^{N} \sum_y \alpha_{x_i,y} \xi(x_i, y), \xi(x_j, y') \right\rangle \\ & - \left\langle \sum_{i=1}^{N} \sum_y \alpha_{x_i,y} \xi(x_i, y), \tilde{\xi} \right\rangle \\ & + \epsilon \| \sum_{x=1}^{N} \sum_y \alpha_{xy} \xi(x,y) \|_{\mathcal{H}}^2 \end{aligned}$$

$$
\begin{aligned}
&= \sum_{j=1}^{N} \tilde{\pi}(x) \log \sum_{y'} \exp \sum_{i=1}^{N} \sum_{y} \alpha_{x_i,y} \left\langle \xi(x_i, y), \xi(x_j, y') \right\rangle \\
&\quad - \left\langle \sum_{i=1}^{N} \sum_{y} \alpha_{x_i,y} \xi(x_i, y), \tilde{\xi} \right\rangle \\
&\quad + \epsilon \left\langle \sum_{i=1}^{N} \sum_{y} \alpha_{x_i,y} \xi(x_i, y), \sum_{j=1}^{N} \sum_{\tilde{y}} \alpha_{x_j,\tilde{y}} \xi(x_j, \tilde{y}) \right\rangle . \qquad (4.13)
\end{aligned}
$$

Substituting the kernel property

$$
K((x_a, y_a), (x_b, y_b)) = \left\langle \xi(x_a, y_a), \xi(x_b, y_b) \right\rangle ,
$$

the fact that $K = K_1 + K_2$ and the definition of $\tilde{\xi}$

$$
\tilde{\xi} = \sum_{l=1}^{L} \tilde{\pi}(\bar{x}, y) \begin{bmatrix} \tilde{\xi}_1(\bar{x}_l, y_l) \\ 0 \end{bmatrix},
$$

yields the following kernelized loss function

$$
\begin{aligned}
\mathbb{Q}(\alpha; D) = &\sum_{j=1}^{N} \tilde{\pi}(x) \log \sum_{y'} \exp \sum_{i=1}^{N} \sum_{y} \alpha_{x_i,y} \left[(K_1(x_i, y), (x_j, y')) + K_2 \left((x_i, y), (x_j, y') \right) \right] \\
&- \sum_{l=1}^{L} \tilde{\pi}(\bar{x}_l, y) \sum_{i=1}^{N} \sum_{y} \alpha_{x_i,y} K_1 \left((x_i, y), (\bar{x}_l, y_l) \right) \\
&+ \epsilon \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{y} \sum_{\tilde{y}} \alpha_{x_i,y} \alpha_{x_j,\tilde{y}} \left[(K_1(x_i, y), (x_j, \tilde{y})) + K_2 \left((x_i, y), (x_j, \tilde{y}) \right) \right] .
\end{aligned} \qquad (4.14)
$$

4.2.2 Expectation Penalties

As mentioned earlier the number of parameters for pairwise penalties can get intractable with the increasing size of data. In order to reduce the number of parameters, we consider a relaxed version of the pairwise penalties. Here, instead of minimizing the discrepancy of conditional distributions across all x_j, x_k pairs, we minimize the discrepancy of distributions over local regions. In particular, we impose minimization of various norms of the following discrepancy

$$\sum_j \left(\mathrm{s}(x_j, x_k)p(y|x_j) - \mathrm{s}(x_j, x_k)p(y|x_k)\right), \tag{4.15}$$

over (x, y) pairs (See Equation (4.17)). This enforces the conditional output probabilities of an instance x_k to be close to the weighted average of the model outputs of the instances x_j within its vicinity.

As in the case of pairwise penalties, we can express the additional penalty term given in (4.15) in terms of a linear operator Φp over similarity feature functions ϕ given by

$$\phi_{k,y}(x_m, y') = \begin{cases} s(x_m, x_k) & \text{if } x_m \neq x_k \text{ and } y = y', \\ -\sum_j s(x_j, x_m) & \text{if } x_m = x_k \text{ and } y = y', \\ 0 & \text{otherwise,} \end{cases} \tag{4.16}$$

for $m \in \{1, \ldots, n\}$. Then $(\Phi p)_i$ yields (4.15) for $i = k \times |\mathcal{Y}| + y$. Therefore, Φp

can be given as

$$\Phi\mathbf{p} = \begin{bmatrix} \phi_1(x_1,y_1) & \dots & \phi_1(x_1,y_c) & \dots & \phi_1(x_n,y_1) & \dots & \phi_1(x_n,y_c) \\ \vdots & \ddots & & & & & \\ \phi_i(x_1,y_1) & \dots & \phi_i(x_1,y_c) & \dots & \phi_i(x_n,y_1) & \dots & \phi_i(x_n,y_c) \\ \vdots & \ddots & & & & & \\ \phi_d(x_1,y_1) & \dots & \phi_d(x_1,y_c) & \dots & \phi_d(x_n,y_1) & \dots & \phi_d(x_n,y_c) \end{bmatrix} \begin{bmatrix} p(y_1|x_1) \\ \vdots \\ p(y_c|x_1) \\ \vdots \\ p(y_1|x_n) \\ \vdots \\ p(y_c|x_n) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_x E_{\sim p(y|x)}\phi_1(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_i(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_d(x,y) \end{bmatrix}$$

$$= \begin{bmatrix} \vdots \\ \sum_j \left(s(x_j,x_k)p(y|x_j)\right) - \left(\sum_j s(x_j,x_k)\right) p(y|x_k) \\ \vdots \end{bmatrix}, \tag{4.17}$$

where $d = |\mathcal{X}| \times |\mathcal{Y}|$ and $c = |\mathcal{Y}|$.

We penalize the primal MaxEnt problem with some norm of Φp. Imposing expectation penalties requires at most nC additional parameters as opposed to pairwise penalties which requires n^2C additional parameters. In addition to the reduction of optimization parameters, another advantage of these relaxed constraints is that they are more robust to conflicting (true but hidden) labels of

similar samples. Therefore, box constraints or equivalently an l_1 regularized dual objective is less likely to give an infeasible solution.

The semi-supervised logistic regression with ℓ_2^2 regularization for expectation semi-supervised penalty is given by (4.8) with F defined as

$$F(x, y; \lambda, \gamma) = \langle \lambda, \psi(x, y) \rangle + \sum_{\hat{x}} s(\hat{x}, x)\gamma_{xy} - \sum_{\bar{x}} s(x, \bar{x})\gamma_{\bar{x}y}.$$

The gradients of γ are given by

$$\frac{\partial \mathbb{R}(\lambda, \gamma; D)}{\partial \gamma_{xy}} = \sum_{x'} p(y|x')s(x', x) - \sum_{\hat{x}} p(y|x)s(\hat{x}, x).$$

The kernel version follows as in Section 4.2.1.

4.3 Experiments

4.3.1 Experiments on Benchmark Data Sets

Similarity Metric

For the empirical evaluation we use the following similarity definition

$$s(x_i, x_j) = \begin{cases} K(x_i, x_j) & \text{if } x_j \in N_{\kappa_{x_i}}, \\ 0 & \text{otherwise.} \end{cases} \tag{4.18}$$

where K is a Mercer kernel and $N_{\kappa_{x_i}}$ is the κ-nearest neighborhood of x_i with respect to K. Note that this similarity metric is sparse and non-symmetric.

Data Sets

We present experiments on data sets that have been extensively analyzed in previous SSL work for fair and extensive comparison. We chose Digit1, USPS_2 and COIL data sets among the benchmarks data sets from (Chapelle et al., 2006), USPS_{10} and text data sets from (Chapelle & Zien, 2005) and MNIST (LeCun et al., 1998). Appendices B.1 and B.2 describe the essential properties of the data sets. For further details see (Chapelle et al., 2006; Chapelle & Zien, 2005).

Model Selection

The hyper-parameters of our algorithm are the neighborhood size κ in (4.18), the regularization constant ϵ_1 for the model feature parameters and ϵ_2 for the similarity feature parameters and finally the kernel bandwidth α in the case of a RBF kernel. We performed cross validation on a subset of labeled samples for model selection. From each data split we moved 25% of the labeled samples to the corresponding unlabeled data split and found the model parameters that give the best average transduction performance on these samples only. In other words, *model selection is completely blind to the true labels of the unlabeled samples* in order to reflect the real-life scenario as closely as possible. We considered a range of hyper-parameters for model selection, $\kappa \in \{5, 15, 20, 30\}$ and $\epsilon_1, \epsilon_2 \in \{e^{-1}, e^{-2}, e^{-3}, e^{-4}\}$. We set $\alpha = \eta^{-2}$ where η is the median of pairwise distances. Subsequently, we retrained the algorithm with these parameters on the original set of labeled and unlabeled samples. In the following section, we report the transductive error on the unlabeled samples averaged over all splits. Following previous work, we used the cosine kernel, $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \mathbf{x_i}, \mathbf{x_j} \rangle / \|\mathbf{x_i}\| \|\mathbf{x_j}\|$ for *text* and the RBF kernel, $K(\mathbf{x_i}, \mathbf{x_j}) = \exp(-\alpha \|\mathbf{x_i} - \mathbf{x_j}\|^2)$ for all other data sets. In all experiments, the same kernel is

used for the kernel logistic regression (KLR) and the similarity metric.

Results

Tables 4.1, 4.2, 4.3 and 4.4 summarize the empirical evaluation of our algorithm. In Table 4.1, we report transduction error on Digit 1, $USPS_2$ and COIL data sets from (Chapelle et al., 2006) for logistic regression (LR) and kernel logistic regression (KLR) both augmented with pairwise (PW) and expectation (EP) penalties. All results are averages over all splits with the model parameters selected via cross validation as discussed above. The first four lines correspond to the supervised methods, namely 1-nearest neighborhood (1-NN), Support Vector Machine (SVM), LR and KLR, where the algorithms are trained only on the labeled samples. At the bottom of the table, the performances of the most competitive semi-supervised algorithms reported in (Chapelle et al., 2006), namely Transductive SVM (TSVM) (Vapnik, 1998), Cluster Kernel (Chapelle et al., 2003), Discrete Regularization (Chapelle et al., 2006), Data Dependent Regularization (Chapelle et al., 2006) and Low Density Separation (LDS) (Chapelle & Zien, 2005). The reader may refer to (Chapelle et al., 2006) for a comparison with a wider selection of algorithms.

A comparison of the results of our framework to supervised learning methods indicates a consistent improvement for all data sets. This is not the case for many semi-supervised learning methods. Regarding the relative performance with respect to other SSL methods, we observe that our approach is very competitive. In particular, it yields the best performance in Digit1 data set with 20% error reduction. For the other data sets, the method achieves the second and third best results. Interestingly the linear logistic regression algorithm is as good as the

kernel logistic regression algorithm in most cases, indicating that using similarity features captures the non-linearities sufficiently. Investigating the differences between pairwise and expectation penalties, we observe that except for the Digit1 data set, pairwise constraints are almost always more informative.

Table 4.2 reports the 10 class USPS data set and the text data. Performances of ∇TSVM, a variant of TSVM (Chapelle & Zien, 2005), Laplacian SVM (Sindhwani et al., 2005), LDS (Chapelle & Zien, 2005), Label Propagation (Zhu & Ghahramani, 2002), Transductive Neural Network (TNN) (Karlen et al., 2008) and Manifold Transductive Neural Network (Karlen et al., 2008) (ManTNN) algorithms are provided for comparison. The comparative analysis yields a similar pattern to Table 4.1. On text data, the performance of our approach is not as good as the most competitive methods reported for this data set.

Finally Tables 4.3, 4.4, 4.5 and 4.6 report the transduction performance of our algorithm on a randomly chosen subset of the MNIST data set with up to $70,000$ samples. The classification error for logistic regression (LR) with pairwise (PW) and expectation (EP) penalties for increasing number of unlabeled samples is shown. The supervised test error is given for comparison. Pairwise constrained LR performs better however, as the number of unlabeled data exceeds $25,000$, it requires a number of optimization parameters on the scale of millions. The performance on USPS_{10}, COIL, MNIST data sets indicates that our algorithm can successfully handle multi-class problems.

	Digit1	$USPS_2$	COIL
1-NN	3.89	5.81	17.35
SVM	5.53	9.75	22.93
LR	7.31	12.83	35.17
KLR	6.02	9.20	24.63
LR+EP	2.35	5.69	15.33
LR+PW	2.27	**5.18**	12.37
KLR+EP	**1.94**	6.44	15.22
KLR+PW	2.26	5.54	**11.34**
TSVM	6.15	9.77	25.80
MVU + 1-NN	3.99	6.09	32.27
LEM + 1-NN	2.52	6.09	36.49
QC + CMN	3.15	6.36	10.03
Discrete Reg.	2.77	**4.68**	**9.61**
SGT	2.61	6.80	-
Cluster-Kernel	3.79	9.68	21.99
Data-Dep. Reg.	**2.44**	5.10	11.46
LDS	3.46	4.96	13.72
Laplacian RLS	2.92	**4.68**	11.92
CHM (normed)	3.79	7.65	-

Table 4.1: Transduction error on benchmark data sets averaged over all splits. Here we report only the most competitive results from previous work, for the full comparison table see the analysis of benchmarks chapter in (Chapelle et al., 2006). 1-NN: 1-nearest neighborhood.

	$USPS_{10}$	text
SVM	23.18	18.86
NN	24.57	15.87
LR	26.07	15.64
KLR	28.81	15.70
LR+EP	20.02	13.03
LR+PW	**14.96**	12.87
KLR+EP	19.76	13.20
KLR+PW	16.15	**12.06**
SVMLight-TSVM	26.46	7.44
CCCP-TSVM	16.57	7.97
∇TSVM	17.61	5.71
LapSVM	12.70	10.40
LDS	15.80	**5.10**
Label Propagation	21.30	11.71
Graph	16.92	10.48
TNN	16.06	6.11
ManTNN	**11.90**	5.34

Table 4.2: Transduction error averaged over all splits of $USPS_{10}$ and text data sets. Supervised training error for single layer neural network and SVM and other semi-supervised methods have been provided for comparison. NN stands for neural network. Results of previous work obtained from (Karlen et al., 2008).

$\|U\|$	$SUP.$	5000	10000	15000	20000	25000
LR_{EP}	27.84	19.53	19.74	19.98	20.47	21.08
LR_{PW}	27.84	16.42	14.05	12.98	12.17	11.61

Table 4.3: Transduction error on MNIST data set with $|L| = 100$ averaged over 10 partitions for logistic regression with pairwise (PW) and expectation constraints (EP). The neighborhood size, κ is taken as 20 for EP and 10 for PW. SUP. indicates supervised LR results on all unlabeled data used as test samples.

$\|U\|$	$SUP.$	5000	10000	15000	20000	25000
LR_{EP}	19.93	13.92	12.84	12.28	12.44	12.28
LR_{PW}	19.93	11.63	9.88	8.98	8.41	8.01

Table 4.4: Transduction error on MNIST data set with $|L| = 250$.

$\|U\|$	$SUP.$	5000	10000	15000	20000	25000
LR_{EP}	14.41	8.08	7.01	6.33	5.96	5.70
LR_{PW}	14.41	7.45	6.71	6.13	5.79	5.50

Table 4.5: Transduction error on MNIST data set with $|L| = 1000$.

	L=100	L=1000
SVM	23.44	7.77
NN	25.81	10.70
CNN	22.98	6.45
LR	26.99	14.36
KLR	26.65	9.60
LR+EP_{70K}	20.21	**4.87**
LR+EP_{25K}	21.08	5.70
LR+PW_{25K}	**11.61**	5.50
TNN	18.02	6.66
ManTNN	**7.30**	2.88
TCNN	13.01	3.50
ManTCNN	6.65	**2.15**
CCCP-TSVM	16.81	5.38

Table 4.6: A comparison of our methods on MNIST with 100 and 1000 labeled samples to the results reported in the literature. Results obtained from (Karlen et al., 2008) use an unlabeled sample set of size 70,000.

4.3.2 Remote Sensing Image Classification Experiments

Joint work with Dr. Gustavo Camps-Valls. [2]

In this section, we present the experimental results for the semi-supervised logistic regression algorithm (with expectation constraints) in several remote sensing image classification problems. Remote sensing is a discipline that studies and models the processes occurring on the Earth's surface and their interaction with the atmosphere (Lillesand et al., 2004). Images acquired by airborne or satellite optical sensors measure the emergent radiation at different wavelengths, while active sensors measure the back-scattered energy emitted by the on-board antenna. In both cases, a pixel in the image can be defined as a potentially very high-dimensional vector characterizing the observed material. This information allows the characterization, identification, and classification of the land-cover classes. While image segmentation is the main product in remote sensing data analysis, its success is limited by the scarcity (and also the quality) of the labeled pixels. Collecting a sufficient amount of reliable labels requires a very costly terrestrial campaign, both in terms of time and human resources. As in other application domains, unlabeled remote sensing data are relatively easier to obtain as it does not require human or time resources: one can simply select a set of the unlabeled pixels in an image.

The remote sensing images used in the experiments are selected from the following categories: hyperspectral (Salinas, KSC) and multispectral (Naples). The RGB compositions for the considered scenes are given in Figure 4.1.

[2]Image Processing Laboratory, Universitat de València, València, Spain, email: gustavo.camps@uv.es, www: http://www.uv.es/gcamps.

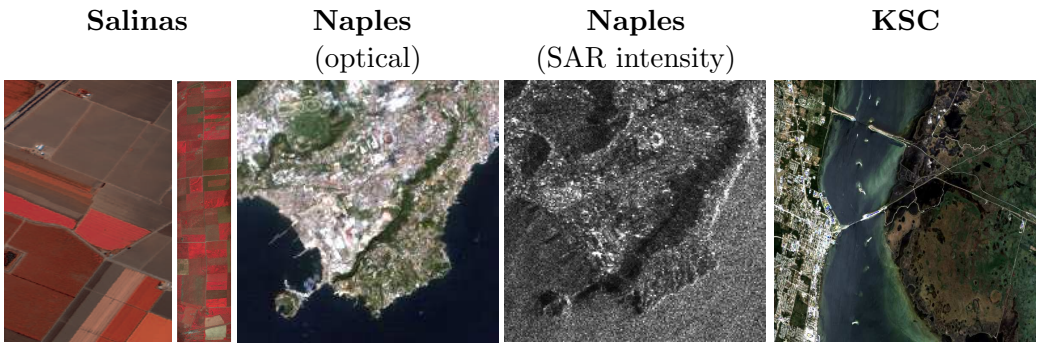


Figure 4.1: RGB composition of the considered data sets, ranging from multispectral to hyperspectral, radar and very high spatial resolution imagery.

Experimental setup

In our experiments, we have worked with 3 data sets (See Figure 4.1). The properties of the data sets and the data generation process details are provided in Appendix B.3. For all considered classification problems, we generated three sets, training, validation, and unlabeled sets. Training and validation sets contain the same number of labeled samples (variable in the $[100, 500]$ range) whereas the unlabeled data set contained a total of 2000 samples (500 for the KSC data). The data is partitioned into 10 different splits and we report the overall accuracy, OA[%] averaged over these splits. Inductive error is computed on the validation set whereas the transductive error is computed over the unlabeled sets. Data was scaled in the $[0, 1]$ range before training.

For the model selection for SLR, we follow Section 4.3.1. We compare SLR with standard methods in the literature: classical SVM, regularized least squares SVM (RLSC), Laplacian SVM (LapSVM), and the Laplacian RLSC.

For all the methods mentioned above, we used the RBF kernel. The graph Laplacian consists of labeled and unlabeled nodes connected using κ nearest neigh-

bors, and the edge weights are computed using the Euclidean distance among samples. Two more free parameters are tuned in Laplacian methods: γ_L is the standard regularization parameter for the decision function and γ_M controls the complexity of the intrinsic data geometry. Both parameters were tuned in the range $[10^{-4}, 10^4]$, the number of neighbors κ used to compute the graph Laplacian varies from 3 to 9, and the kernel width was tuned in the range $\sigma = \{10^{-2}, \ldots, 10\}$. The selection of the best subset of free parameters was carried out by cross-validation on the training set.

Results

Figures 4.2, 4.3 and 4.4 illustrate the results for the *inductive* (prediction on the validation set) and *transductive* (prediction on the unlabeled set) settings for Salinas, KSC and Naples data sets respectively. For the Salinas data set, we observe that a clear gain is obtained with respect to all other semi-supervised methods in the inductive setting, and SLR outperforms the rest with an average gain of +2% (Salinas). The gain over supervised approaches is more significant with smaller numbers of labeled training samples. In the transductive setting, a significant improvement is observed with smaller labeled data sets ($n < 300$), however performance saturates for $n > 300$. With a sufficient amount of labeled training samples, the output distribution is modeled fairly well and the introduction of unlabeled samples may harm rather than help.

In the case of Naples, the SLR transduction error is lower than 1% and largely outperforms the rest of the methods while the induction accuracy is low. Both results match with the data characteristics: we are merging features of different nature (optical and radar) so we observe that, first, the data set is very sensitive

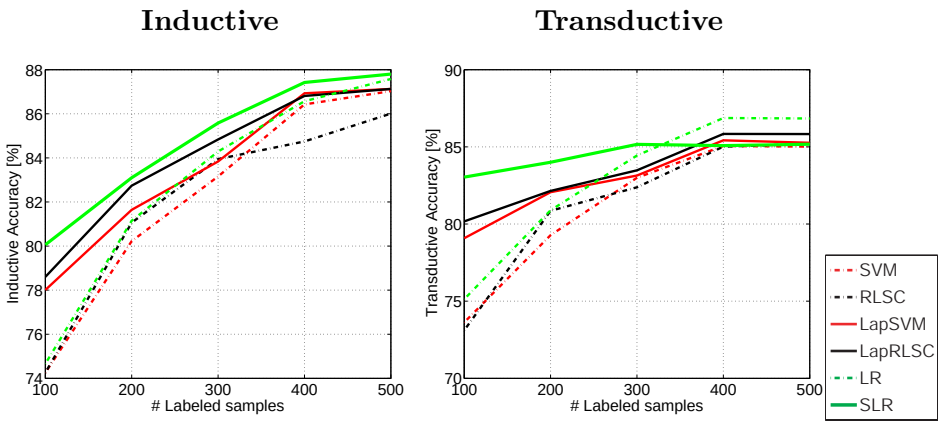


Figure 4.2: Salinas dataset. Overall accuracy, [%]OA, both inductive (left) and transductive (right) settings.

to the non-linear similarity features, and second, that a linear logistic regression may not be sufficient to solve the problem.

Finally, in the case of the hyper-spectral KSC image, we observe poor performance in the inductive setting (using unlabeled samples here may even harm the solution) but significant improvement is reported in transduction, with an average gain over the (nonlinear) Laplacian methods of around +1.5%.

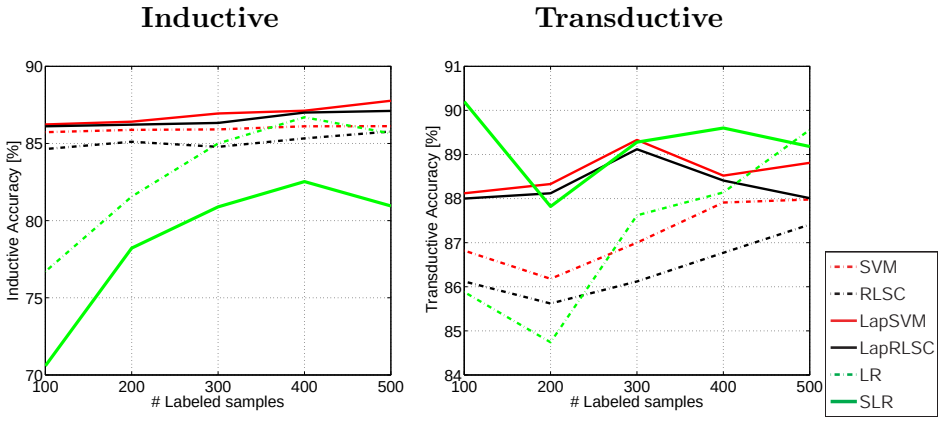


Figure 4.3: KSC data set. Overall accuracy, [%]OA, for the considered images in both inductive (left) and transductive (right) settings.

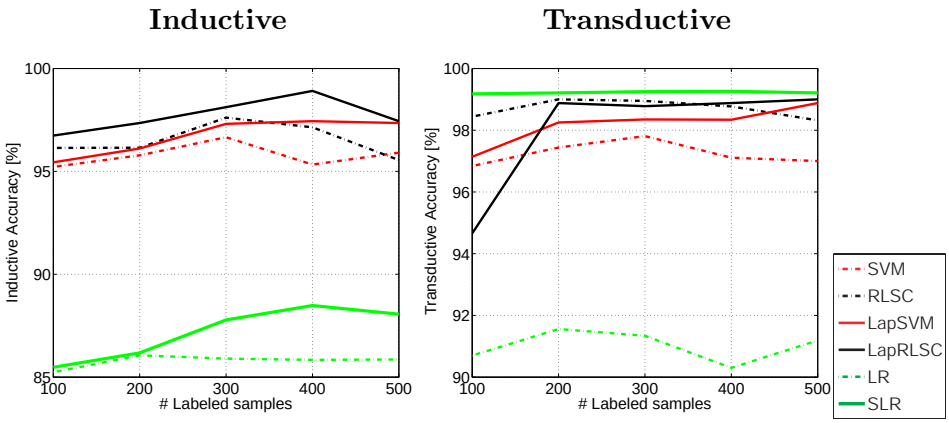


Figure 4.4: Naples data set. Overall accuracy, [%]OA, for the considered images in both inductive (left) and transductive (right) settings.

Chapter 5

Semi-supervised Structured Output Prediction

5.1 Introduction

In structured output (SO) learning the goal is to learn a mapping from arbitrary input spaces to output spaces whose elements are *structured objects* such as sequences, trees, strings and graphs. In other words, SO prediction is the task of predicting a vector of inter-dependent output variables $\mathbf{y} = (y^1, \ldots, y^r)$ given a vector of observations $\mathbf{x} = (x^1, \ldots, x^t)$.

Prior to SO learning, traditional discriminative machine learning algorithms used to decompose complex outputs into isolated entities and train independent classifiers on them, losing the knowledge inherent in the output inter-dependencies (Abney et al., 1999). However, these inter-dependencies or interactions between different components of complex data might be in fact very rich and informative. Taking these interactions into account contradicts the assumption of the indepen-

dent and identically distributed data instances of the majority of classical learning algorithms. Simply put, a structured model implies restrictions on the possible outputs reducing the search space. This can improve performance in comparison to classifiers that are agnostic to the dependency of structures.

The alternative to discriminative sequence learning methods is generative models such Hidden Markov Models (HMMs) and its variations. However, HMMs are known to have to major shortcomings, first they attempt to model the joint distribution of the observations and the labels due to their generative nature. Secondly, they impose independence assumptions on past and future observations which is often violated in real-life applications. See (Altun, 2005) for a thorough discussion on discriminative versus generative SO prediction. Recently SO learning has attracted increasing interest with many potential applications particularly in natural language processing, bio-informatics and computer vision.

5.2 Background

5.2.1 Conditional Random Fields

A canonical example of SO learning is sequence labeling where the dependency structure is a simple chain. SO prediction problem can be represented by a Markov network $G = (V, E)$ where V denotes the variables of $\mathbf{x}$ and $\mathbf{y}$, and E represents the dependencies of the variables. Let $\mathcal{C}$ be the set of cliques of G and y_c denote the output variables restricted to the clique $c \in \mathcal{C}$ (x_c defined similarly). We define $\mathcal{P}$ be the set of conditional probability distributions $p(\mathbf{y}|\mathbf{x})$ of structured input-output objects, $\mathbf{x} \in \mathcal{X}$, $\mathbf{y} \in \mathcal{Y}$. Then, finding the $p \in \mathcal{P}$ that has the maximum conditional entropy while respecting the moment matching constraints for features defined over

the cliques yields Conditional Random Fields (CRFs) (Lafferty et al., 2001) as we will show in Section 5.3. Therefore, Conditional Random Fields (CRF) are a discriminative probabilistic model for structured data. For simplicity, we focus on linear chain CRFs for the purposes of this thesis. CRFs model the conditional probability of sequences of feature vectors (observations) and their associated label sequences (output or target values we predict).

As CRFs directly model $p(\mathbf{y}|\mathbf{x})$, they do not have to model the marginal distribution $p(\mathbf{x})$ which is often a much harder task due to higher dimensionality of the data and the complex inter-dependencies between the features. In addition, CRFs do not have to make any unrealistic independence assumptions among the data since they are indifferent to $p(\mathbf{x})$. In contrast, Hidden Markov Models (HMMs) which can be thought of as a generative counterpart for CRFs, enforce independency assumptions on the data as they model $p(x, y)$. HMM graphs are illustrated with directed edges connecting nodes such as in Figure 5.1(a), indicating that the outputs generate the inputs. In contrast, a CRF model is depicted using undirected edges as in Figure 5.1(b).

In a Markov random field, i.e., an undirected graphical model, the *Markov blanket*, $MB(v)$ of a graph node v consists of v's neighbors. All other nodes in the network are conditionally independent of v when conditioned on $MB(v)$. Therefore, for nodes v_i and v_j,

$$p(v_i|MB(v_i), v_j) = p(v_i|MB(v_j)), \text{ if } i \neq j \text{ and } v_j \notin MB(v_i).$$

In other words, apart from the set of nodes in v's Markov blanket, one's knowledge on the rest of the network becomes irrelevant in terms of predicting v's behavior.

Accordingly, in CRFs, when conditioned on the observations, X, the random variables Y obey the Markov property.

With the graphical model formalism, the probability of a label sequence can be approximated as the normalized product of potential functions of the cliques in the graph

$$p(\mathbf{x}, \mathbf{y}) = \prod_{c \in \mathcal{C}(\mathbf{x})} \Psi(\mathbf{x}, \mathbf{y})_c. \tag{5.1}$$

For instance, for the linear chain CRF model in Figure 5.1(b),

$$\prod_{c \in \mathcal{C}(\mathbf{x})} \Psi(\mathbf{x}, \mathbf{y})_c = \prod_t \exp\left(\sum_j \gamma_j \Lambda_j(y_t, y_{t-1}) + \sum_k \mu_k \psi_k(y_t, x_t)\right) \tag{5.2}$$

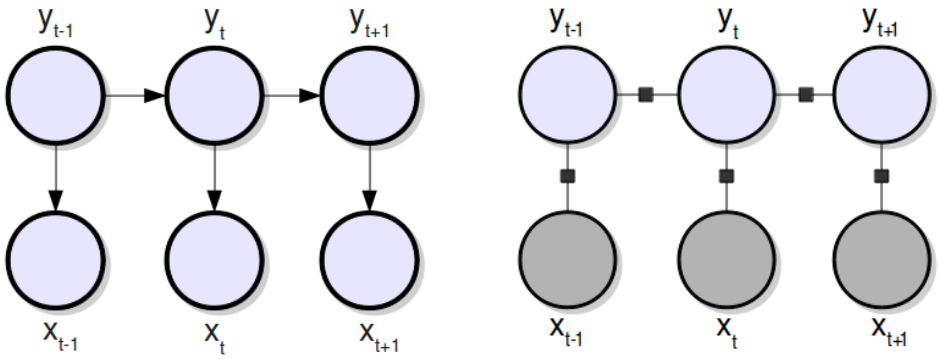
$$= \prod_t \exp\left(\langle \gamma, \Lambda(y_t, y_{t-1}) \rangle + \langle \mu, \psi(y_t, x_t) \rangle\right), \tag{5.3}$$

where Λ are transitional features, ψ are state feature functions, and γ and μ are the model parameters. Then,

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_{t=1}^{T} \left(\langle \gamma, \Lambda(y_t, y_{t-1}) \rangle + \langle \mu, \psi(y_t, x_t) \rangle\right)\right), \text{ and}$$

$$Z(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{Y}(\mathbf{x})} \exp\left(\sum_{t=1}^{T} \left(\langle \gamma, \Lambda(y_t, y_{t-1}) \rangle + \langle \mu, \psi(y_t, x_t) \rangle\right)\right).$$

$Z(\mathbf{x})$ is the normalization function (also known as log-partition function) that is computed per instance $\mathbf{x}$ over all possible $\mathbf{y}$ values for each state.



(a) HMM (b) CRF

Figure 5.1: Linear-Chain Conditional Random Fields and Hidden Markov Models illustrated as graphical models. CRFs are discriminative whereas HMMs are generative.

5.2.2 Parameter Estimation and Inference for Linear Chain CRFs

In this section, we will provide the details of parameter estimation and inference for the specific form of linear chain CRFs. In order to train a CRF, the conditional log-likelihood of the data

$$\begin{aligned}\mathbb{L}(\gamma,\mu) &= \sum_{i=1}^{n} \log p(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}) \\ &= \sum_{i=1}^{n}\left[\sum_{t=1}^{T}\left(\left\langle\gamma, \Lambda({y_t}^{(i)}, {y_{t-1}}^{(i)})\right\rangle + \left\langle\mu, \psi({y_t}^{(i)}, {x_t}^{(i)})\right\rangle\right)\right] - \sum_{i=1}^{n}\log Z(\mathbf{x}^{(i)}),\end{aligned} \tag{5.4}$$

is maximized.

The gradients of the log-likelihood with respect to the optimization parameters

are given by

$$\begin{aligned}\frac{\partial \mathbb{L}}{\partial \gamma_j} &= \sum_{i=1}^{n}\sum_{t=1}^{T}\Lambda_j(y_{t-1}^{(i)}, y_t^{(i)}) - \sum_{i=1}^{n}\sum_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}\sum_{t=1}^{T}\sum_{\sigma_1,\sigma_2\in\mathcal{O}}\Lambda_j(y_{t-1}=\sigma_1, y_t=\sigma_2)p(\mathbf{y}|\mathbf{x}^{(i)}) \\ &= \sum_{i=1}^{n}\sum_{t=1}^{T}\Lambda_j(y_{t-1}^{(i)}, y_t^{(i)}) - \sum_{i=1}^{n}\sum_{t=1}^{T}\sum_{\sigma_1,\sigma_2\in\mathcal{O}}\Lambda_j(y_{t-1}=\sigma_1, y_t=\sigma_2)p(y_{t-1}=\sigma_1, y_t=\sigma_2|\mathbf{x}^{(i)}),\end{aligned} \tag{5.5}$$

and

$$\begin{aligned}\frac{\partial \mathbb{L}}{\partial \mu_k} &= \sum_{i=1}^{n}\sum_{t=1}^{T}\psi_k(y_t^{(i)}, x_t^{(i)}) - \sum_{i=1}^{n}\sum_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}\sum_{t=1}^{T}\sum_{\sigma\in\mathcal{O}}\psi_k(y_t=\sigma, x_t)p(\mathbf{y}|\mathbf{x}^{(i)}) \\ &= \sum_{i=1}^{n}\sum_{t=1}^{T}\psi_k(y_t^{(i)}, x_t^{(i)}) - \sum_{i=1}^{n}\sum_{t=1}^{T}\sum_{\sigma\in\mathcal{O}}\psi_k(y_t=\sigma, x_t)p(y_t=\sigma|\mathbf{x}^{(i)}),\end{aligned} \tag{5.6}$$

where $\mathcal{O}$ is the output alphabet, d is the number of model features, $i \in \{1, 2 \ldots n\}$, $j \in \{1, 2 \ldots |\mathcal{O}| \times |\mathcal{O}|\}$, $k \in \{1, 2 \ldots d \times |\mathcal{O}|\}$ and t are the indices over the data samples, transition features, state features and different states respectively. The first term of the gradient is the empirical expectation of the corresponding feature. The second term is the expectation of the feature under the model distribution $p(\mathbf{y}|\mathbf{x})$.

The log partition function and the expected values of the features under the model distribution can be efficiently calculated using a dynamic programming method, namely **the forward-backward algorithm** (Rabiner, 1989). Further details on inference in CRFs can be found in (Sutton & McCallum, 2007).

5.3 Duality of Chain CRFs

Theorem 4 (Duality of Chain CRFs) *Using $U_1 = U_2 = h_2$ from Table 2.1, the minimization problem of $\min_{p \in P} KL(p||q_0) + U_1 + U_2$ such that*

$$U_1 = \frac{1}{\epsilon} \| E_{\mathbf{x} \sim p(\mathbf{x})} E_{\mathbf{y} \sim p(\mathbf{y}|\mathbf{x})} \left[\Lambda(\mathbf{x}, \mathbf{y}) \right] - E_{\mathbf{x}, \mathbf{y} \sim \tilde{p}(\mathbf{x}, \mathbf{y})} \left[\Lambda(\mathbf{x}, \mathbf{y}) \right] \|_{\mathcal{B}}^2$$
$$U_2 = \frac{1}{\epsilon} \| E_{\mathbf{x} \sim p(\mathbf{x})} E_{\mathbf{y} \sim p(\mathbf{y}|\mathbf{x})} \left[\psi(\mathbf{x}, \mathbf{y}) \right] - E_{\mathbf{x}, \mathbf{y} \sim \tilde{p}(\mathbf{x}, \mathbf{y})} \left[\psi(\mathbf{x}, \mathbf{y}) \right] \|_{\mathcal{B}}^2,$$

for Banach Space $\mathcal{B}$ has the dual given by,

$$\max_{\gamma, \mu} \; (\langle \Lambda(\mathbf{x}, \mathbf{y}), \gamma \rangle + \langle \psi(\mathbf{x}, \mathbf{y}), \mu \rangle) - \log Z(\mathbf{x}|\gamma, \mu) - \epsilon \|\gamma\|_{\mathcal{B}^*}^2 - \epsilon \|\mu\|_{\mathcal{B}^*}^2 \qquad \textit{where,}$$
$$Z(\mathbf{x}|\gamma, \mu) = \sum_{\mathbf{y} \in \mathcal{Y}(\mathbf{x})} q_0(\mathbf{y}|\mathbf{x}) \exp\left(\langle \Lambda(\mathbf{x}, \mathbf{y}), \gamma \rangle + \langle \psi(\mathbf{x}, \mathbf{y}), \mu \rangle \right).$$

Proof *Sketch* The convex conjugate of $KL(p\|q_0) = \sum_{\mathcal{Z}} q(z) = \log(\frac{p(z)}{q_0(z)})dz$ where p is a probability distribution, is given by $KL^*(p^*_{\mathbf{x}}) = \log(\sum_{\mathcal{Z}} q_0(\mathbf{y}|\mathbf{x}) \exp(p^*_{\mathbf{x}})) + e^{-1}$, (Rockafellar, 1996). The convex dual of the the potential functions $U^* = (U_1 + U_2)^*$ is

$$U^* = \left\langle \tilde{\Lambda}, \gamma \right\rangle + \left\langle \tilde{\psi}, \mu \right\rangle - \epsilon \|\gamma\|_{\mathcal{B}^*}^2 - \epsilon \|\mu\|_{\mathcal{B}^*}^2 \tag{5.7}$$

Substituting these relations into Fenchel's duality yields the claim. ■

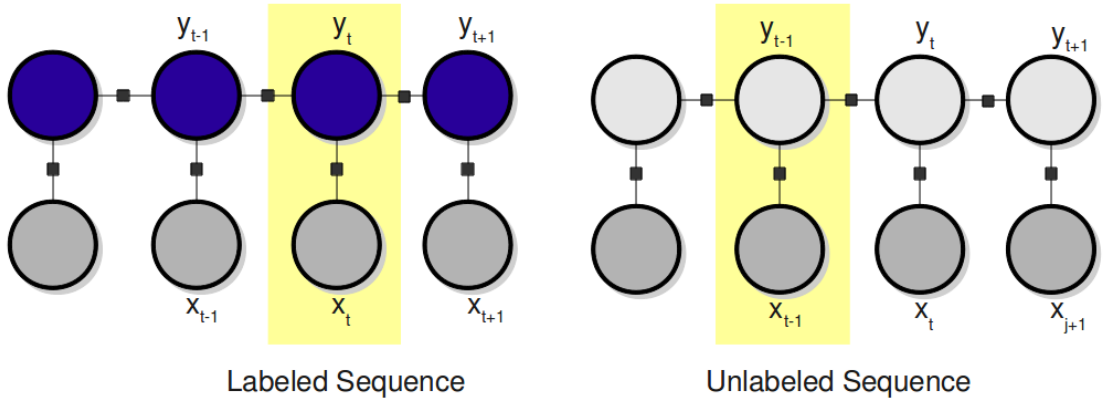


Figure 5.2: We define similarity constraints over pairs of cliques, i.e., we impose the semi-supervised smoothness assumption such that the marginalized conditional probabilities of cliques with similar features are likely to be the same. This can be achieved via additional constraints in the form given in Equation (5.9) or penalties as in Equation (5.10) which lead to different regularization schemes in the dual. In this example, two similar cliques from different sequences are indicated with yellow shading.

5.4 Semi-supervised CRFs via MaxEnt

5.4.1 Pairwise Similarity Constrained Semi-supervised CRFs

In Section 5.3 we have demonstrated that CRFs are a specific instance of the generalized MaxEnt framework via Theorem (4). In Chapter 4, we have proposed an approach to extend the MaxEnt framework to the semi-supervised setting using additional penalty functions on the objective. Therefore, we can combine these ideas to derive semi-supervised CRFs. However, at this point the critical issue becomes the definitions of similarities on structured objects. Defining similarity relations on the entire sequences corresponds to restricting the conditional output probabilities on the entire sequence $p(y|x)$. However, this would require a very big set of samples as the cardinality of output space $\mathcal{Y}$ is the exponential in terms of the cardinality of the output alphabet and the length of the sequences. On the

other hand, restricting probabilities only on individual nodes, $p(y_t|x_t)$, would not be sufficiently informative, as this would ignore all sorts of dependencies between observations in the sequence. Therefore, we are constraining the marginalized conditional output probabilities

$$p(y_c = \sigma|\mathbf{x}) \quad = \sum_{\mathbf{y}\in\mathcal{Y}(x):y_c=\sigma} p(\mathbf{y}|\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})} p(\mathbf{y}|\mathbf{x})\delta(y_c,\sigma), \tag{5.8}$$

which is more informative yet computationally tractable. We are going to introduce similarity constraints over pairs of cliques as illustrated in Figure 5.2. To be specific, we would like to have smooth outputs, that is we want our model to favor pairs of cliques with similar model features to have similar marginalized conditional probabilities. One way to express this is as a constraint

$$\forall_{\hat{\mathbf{x}}\in D}\forall_{\hat{c}\in\mathcal{C}(\hat{\mathbf{x}})}\forall_{\bar{\mathbf{x}}\in D}\forall_{\bar{c}\in\mathcal{C}(\bar{\mathbf{x}})}\forall_{\sigma} \;\; |\sum_{\mathbf{y}\in\mathcal{Y}(\hat{\mathbf{x}})} p(\mathbf{y}|\hat{\mathbf{x}})\delta(\mathbf{y}_{\hat{c}},\sigma) - \sum_{\mathbf{y}\in\mathcal{Y}(\bar{\mathbf{x}})} p(\mathbf{y}|\bar{\mathbf{x}})\delta(\mathbf{y}_{\bar{c}},\sigma)| < \frac{\epsilon}{\mathrm{s}(\hat{\mathbf{x}}_{\hat{c}},\bar{\mathbf{x}}_{\bar{c}})}, \tag{5.9}$$

such that the discrepancy between conditional marginal output probabilities between similar cliques are constrained more. Alternatively, we can express this objective as a penalty function U as follows

$$U = \sum_{\hat{\mathbf{x}}\in D}\sum_{\hat{c}\in\mathcal{C}(\hat{\mathbf{x}})}\sum_{\bar{\mathbf{x}}\in D}\sum_{\bar{c}\in\mathcal{C}(\bar{\mathbf{x}})}\sum_{\sigma\in\mathcal{O}} \left(\mathrm{s}(\hat{\mathbf{x}}_{\hat{c}},\bar{\mathbf{x}}_{\bar{c}})p(y_{\hat{c}}=\sigma|\hat{\mathbf{x}}) - \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}},\bar{\mathbf{x}}_{\bar{c}})p(y_{\bar{c}}=\sigma|\bar{\mathbf{x}})\right)^2, \tag{5.10}$$

which enforces that the discrepancies between the model outputs for similar cliques are smaller. The crucial point here is that, as in the case of model features, we decompose the similarity features over cliques. The MaxEnt primal objective with

U as an additional penalty including unlabeled samples, can be reformulated as follows,

$$\min_{p\in\mathcal{P}} \mathrm{KL}(p||q) + \frac{1}{\epsilon}\|\tilde{\psi} - \mathbb{E}_p[\psi]\|_2^2 + \frac{1}{\epsilon}\|\tilde{\Lambda} - \mathbb{E}_p[\Lambda]\|_2^2 + \frac{1}{\epsilon}\|\Phi p\|_2^2. \tag{5.11}$$

where the similarity feature functions ϕ are given by

$$\phi_{\hat{\mathbf{x}}_{\hat{c}},\bar{\mathbf{x}}_{\bar{c}},\sigma}(\mathbf{x}_c, \mathbf{y}_c) = \begin{cases} s(\mathbf{x}_c, \bar{\mathbf{x}}_{\bar{c}})\delta(\mathbf{y}_c, \sigma) & \text{if } \mathbf{x}_c = \hat{\mathbf{x}}_{\hat{c}}, \\ -s(\hat{\mathbf{x}}_{\hat{c}}, \mathbf{x}_c)\delta(\mathbf{y}_c, \sigma) & \text{if } \mathbf{x}_c = \bar{\mathbf{x}}_{\bar{c}}, \\ 0 & \text{otherwise.} \end{cases} \tag{5.12}$$

Notice the analogies between Equations (4.8) and (5.11) as well as the similarity feature definitions, (4.3) and (5.19). Similarly, $\|\Phi p\|^2$, the norm-square of the vector resulting from operator Φ acting on our primal variable $p(\mathbf{y}|\mathbf{x})$ gives the penalty in Equation (5.10) since

$$(\Phi p)_{\hat{\mathbf{x}}_{\hat{c}},\bar{\mathbf{x}}_{\bar{c}},\sigma} = \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) \left(\sum_{\hat{\mathbf{y}}\in\mathcal{Y}(\hat{\mathbf{x}})} p(\hat{\mathbf{y}}|\hat{\mathbf{x}})\delta(\hat{\mathbf{y}}_{\hat{c}}, \sigma) - \sum_{\bar{\mathbf{y}}\in\mathcal{Y}(\bar{\mathbf{x}})} p(\bar{\mathbf{y}}|\bar{\mathbf{x}})\delta(\bar{\mathbf{y}}_{\bar{c}}, \sigma) \right). \tag{5.13}$$

$$\Phi\mathbf{p} = \begin{bmatrix} \phi_1\left(x_1, y_1(x_1)\right) & \ldots & \phi_1\left(x_1, y_{K_1}(x_1)\right) & \ldots & \phi_1\left(x_n, y_{K_n}(x_n)\right) \\ \vdots & \ddots & & & \\ \phi_i\left(x_1, y_1(x_1)\right) & \ldots & \phi_i\left(x_1, y_{K_1}(x_1)\right) & \ldots & \phi_i\left(x_n, y_{K_n}(x_n)\right) \\ \vdots & \ddots & & & \\ \phi_d\left(x_1, y_1(x_1)\right) & \ldots & \phi_d\left(x_1, y_{K_1}(x_1)\right) & \ldots & \phi_d\left(x_n, y_{K_n}(x_n)\right) \end{bmatrix} \begin{bmatrix} p(y_1(x_1)|x_1) \\ \vdots \\ p(y_{K_1}(x_1)|x_1) \\ \vdots \\ p(y_1(x_n)|x_n) \\ \vdots \\ p(y_{K_n}(x_n)|x_n) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_x E_{\sim p(y|x)}\phi_1(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_i(x,y) \\ \vdots \\ \sum_x E_{\sim p(y|x)}\phi_d(x,y) \end{bmatrix}$$

$$= \begin{bmatrix} \vdots \\ \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) \left(\sum_{\hat{\mathbf{y}} \in \mathcal{Y}(\hat{\mathbf{x}})} p(\hat{\mathbf{y}}|\hat{\mathbf{x}}) \delta(\hat{\mathbf{y}}_{\hat{c}}, \sigma) - \sum_{\bar{\mathbf{y}} \in \mathcal{Y}(\bar{\mathbf{x}})} p(\bar{\mathbf{y}}|\bar{\mathbf{x}}) \delta(\bar{\mathbf{y}}_{\bar{c}}, \sigma) \right) \\ \vdots \end{bmatrix}, \tag{5.14}$$

where the dimensionality of the feature space is given by

$$d = \sum_{\mathbf{x}} |\mathcal{C}(\mathbf{x})| \times \sum_{\mathbf{x}} |\mathcal{C}(\mathbf{x})| \times |\mathcal{O}|,$$

and the cardinality of the output space for each observation sequence $\mathbf{x}$ is $K_i = |\mathcal{Y}(\mathbf{x}_i)|$.

Hence Equation (5.11) can be restated as,

$$\begin{aligned}
&\min_{p\in\mathcal{P}} \sum_{\mathbf{x}\in D} \tilde{\pi}(\mathbf{x}) \sum_{\mathcal{Y}(\mathbf{x})} p(\mathbf{y}|\mathbf{x}) \log p(\mathbf{y}|\mathbf{x}) + \\
&\frac{1}{\epsilon} \| \sum_{\mathbf{x}\in L} \tilde{p}(\mathbf{x},\mathbf{y}) \psi_i(\mathbf{x},\mathbf{y}) - \sum_{\mathbf{x}\in D} \tilde{\pi}(\mathbf{x}) \sum_{\mathcal{Y}(\mathbf{x})} p(\mathbf{y}|\mathbf{x}) \psi_i(\mathbf{x},\mathbf{y}) \|^2 + \\
&\frac{1}{\epsilon} \| \sum_{\mathbf{x}\in L} \tilde{p}(\mathbf{x},\mathbf{y}) \Lambda_i(\mathbf{x},\mathbf{y}) - \sum_{\mathbf{x}\in D} \tilde{\pi}(\mathbf{x}) \sum_{\mathcal{Y}(\mathbf{x})} p(\mathbf{y}|\mathbf{x}) \Lambda_i(\mathbf{x},\mathbf{y}) \|^2 + \\
&\frac{1}{\epsilon} \sum_{\hat{\mathbf{x}}\in D} \sum_{\hat{c}\in\mathcal{C}(\hat{\mathbf{x}})} \sum_{\bar{\mathbf{x}}\in D} \sum_{\bar{c}\in\mathcal{C}(\bar{\mathbf{x}})} \sum_{\sigma} \quad (\mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) p(\mathbf{y}_{\hat{c}} = \sigma|\hat{\mathbf{x}}) - \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) p(\mathbf{y}_{\bar{c}} = \sigma|\bar{\mathbf{x}}))^2.
\end{aligned}$$

Deriving the convex dual yields the following dual objective,

$$\mathbb{Q}(\alpha,\beta,\gamma;D) = -\sum_{\mathbf{x}\in D} \tilde{\pi}(\mathbf{x}) \log Z_{\mathbf{x}} + \left\langle \alpha, \tilde{\psi} \right\rangle + \left\langle \beta, \tilde{\Lambda} \right\rangle + \epsilon\|\alpha\|_2^2 + \epsilon\|\beta\|_2^2 + \epsilon\|\gamma\|_2^2 \tag{5.15}$$

where

$$\begin{aligned}
F(\mathbf{x},\mathbf{y};\alpha,\beta,\gamma) =& \tilde{\pi}(\mathbf{x}) \langle \alpha, \psi(\mathbf{x},\mathbf{y}) \rangle + \tilde{\pi}(\mathbf{x}) \langle \beta, \Lambda(\mathbf{x},\mathbf{y}) \rangle \\
&- \sum_{\bar{\mathbf{x}}\in D} \sum_{c,\bar{c},\sigma} \mathrm{s}(\mathbf{x}_c, \bar{\mathbf{x}}_{\bar{c}}) \gamma_{\mathbf{x}_{\hat{c}}\bar{\mathbf{x}}_{\bar{c}}\sigma} \delta(\mathbf{y}_c, \sigma) + \sum_{\hat{\mathbf{x}}\in D} \sum_{\hat{c},c,\sigma} \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \mathbf{x}_c) \gamma_{\hat{\mathbf{x}}_{\hat{c}}\mathbf{x}_c\sigma} \delta(\mathbf{y}_c, \sigma),
\end{aligned}$$

and the log-partition function is given by

$$Z_{\mathbf{x}} = \sum_{\mathbf{y}\in\mathcal{Y}(\mathbf{x})} \exp(F(\mathbf{x},\mathbf{y};\alpha,\beta,\gamma).$$

Using the Viterbi algorithm, we find $\mathbf{y}^*$ that has the maximum model probability

for inference,

$$\mathbf{y}^* = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmax}}\ p(\mathbf{y}|\mathbf{x}) = \underset{\mathbf{y}\in\mathcal{Y}(\mathbf{x})}{\operatorname{argmax}}\ F(\mathbf{x},\mathbf{y};\alpha,\beta,\gamma).$$

Since the dual loss is convex, we can derive the gradients of the parameters and solve the optimization problem with any gradient descent algorithm. The gradients are as follows,

$$\frac{\partial \mathbb{Q}(\alpha,\beta,\gamma;D)}{\partial \alpha} = \mathbb{E}[\psi(\mathbf{x},\mathbf{y})] - \tilde{\psi}(\mathbf{x},\mathbf{y}) + \epsilon\alpha, \tag{5.16}$$

$$\frac{\partial \mathbb{Q}(\alpha,\beta,\gamma;D)}{\partial \beta} = \mathbb{E}[\Lambda(\mathbf{x},\mathbf{y})] - \tilde{\Lambda}(\mathbf{x},\mathbf{y}) + \epsilon\beta, \tag{5.17}$$

$$\frac{\partial \mathbb{Q}(\alpha,\beta,\gamma;D)}{\partial \gamma_{\mathbf{x}_c\bar{\mathbf{x}}_{\bar{c}}\sigma}} = \mathrm{s}(\mathbf{x}_c,\bar{\mathbf{x}}_{\bar{c}})p(\mathbf{y}_{\bar{c}}=\sigma|\bar{\mathbf{x}}) - \mathrm{s}(\mathbf{x}_c,\bar{\mathbf{x}}_{\bar{c}})p(\mathbf{y}_c=\sigma|\mathbf{x}) + \epsilon\gamma_{\mathbf{x}_c\bar{\mathbf{x}}_{\bar{c}}\sigma}. \tag{5.18}$$

5.4.2 Expectation Similarity Constrained Semi-supervised CRFs

With the same motivation as in Section 4.2.2 we can define a penalty function U that imposes smoothness functions over local regions as opposed to pairs of cliques. We redefine the similarity features and the associated operator as follows,

$$\phi_{\hat{\mathbf{x}}_{\hat{c}},\sigma}(\mathbf{x}_c,\mathbf{y}_c) = \begin{cases} s(\mathbf{x}_c,\hat{\mathbf{x}}_{\hat{c}})\delta(\mathbf{y}_c,\sigma) & \text{if } \mathbf{x}_c \neq \hat{\mathbf{x}}_{\hat{c}}, \\ -\sum_{\bar{\mathbf{x}}_{\bar{c}}} s(\bar{\mathbf{x}}_{\bar{c}},\mathbf{x}_c)\delta(\mathbf{y}_c,\sigma) & \text{if } \mathbf{x}_c = \hat{\mathbf{x}}_{\hat{c}}, \\ 0 & \text{otherwise.} \end{cases} \tag{5.19}$$

$$
\begin{aligned}
\Phi\mathbf{p} &= \begin{bmatrix}
\phi_1\left(x_1, y_1(x_1)\right) & \dots & \phi_1\left(x_1, y_{K_1}(x_1)\right) & \dots & \phi_1\left(x_n, y_{K_n}(x_n)\right) \\
\vdots & \ddots & & & \\
\phi_i\left(x_1, y_1(x_1)\right) & \dots & \phi_i\left(x_1, y_{K_1}(x_1)\right) & \dots & \phi_i\left(x_n, y_{K_n}(x_n)\right) \\
\vdots & \ddots & & & \\
\phi_d\left(x_1, y_1(x_1)\right) & \dots & \phi_d\left(x_1, y_{K_1}(x_1)\right) & \dots & \phi_d\left(x_n, y_{K_n}(x_n)\right)
\end{bmatrix}
\begin{bmatrix}
p(y_1(x_1)|x_1) \\
\vdots \\
p(y_{K_1}(x_1)|x_1) \\
\vdots \\
p(y_1(x_n)|x_n) \\
\vdots \\
p(y_{K_n}(x_n)|x_n)
\end{bmatrix} \\
&= \begin{bmatrix}
\sum_x E_{\sim p(y|x)}\phi_1(x,y) \\
\vdots \\
\sum_x E_{\sim p(y|x)}\phi_i(x,y) \\
\vdots \\
\sum_x E_{\sim p(y|x)}\phi_d(x,y)
\end{bmatrix} \\
&= \begin{bmatrix}
\vdots \\
\sum\limits_{\hat{\mathbf{x}}_{\hat{c}}} \left(\mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) \sum\limits_{\hat{\mathbf{y}} \in \mathcal{Y}(\hat{\mathbf{x}})} p(\hat{\mathbf{y}}|\hat{\mathbf{x}}) \delta(\hat{\mathbf{y}}_{\hat{c}}, \sigma) \right) - \sum\limits_{\hat{\mathbf{x}}_{\hat{c}}} \mathrm{s}(\hat{\mathbf{x}}_{\hat{c}}, \bar{\mathbf{x}}_{\bar{c}}) \left(\sum\limits_{\bar{\mathbf{y}} \in \mathcal{Y}(\bar{\mathbf{x}})} p(\bar{\mathbf{y}}|\bar{\mathbf{x}}) \delta(\bar{\mathbf{y}}_{\bar{c}}, \sigma) \right) \\
\vdots
\end{bmatrix},
\end{aligned}
\tag{5.20}
$$

where the dimensionality of the feature space is given by $d = \sum_{\mathbf{x}} |\mathcal{C}(\mathbf{x})| \times |\mathcal{O}|$, and the cardinality of the output space for each observation sequence $\mathbf{x}$ is $K_i = |\mathcal{Y}(\mathbf{x}_i)|$.

Incorporating the associated penalty to the primal loss function given in Equation (5.11) yields the loss form in Equation (5.15) where the definition of F, $p(\mathbf{y}|\mathbf{x})$

and $Z(\mathbf{x})$ are given by,

$$
\begin{aligned}
F(x, y; \alpha, \beta, \gamma) =& \tilde{\pi}(\mathbf{x}) \langle \alpha, \psi(x, y) \rangle + \tilde{\pi}(\mathbf{x}) \langle \beta, \Lambda(x, y) \rangle \\
& - \sum_{\bar{x} \in D} \sum_{c, \bar{c}, \sigma} \mathrm{s}(x_c, \bar{x}_{\bar{c}}) \gamma_{\bar{x}_{\bar{c}}} \delta(y_c, \sigma) + \sum_{\hat{x} \in D} \sum_{\hat{c}, c, \sigma} \mathrm{s}(\hat{x}_{\hat{c}}, x_c) \gamma_{x_c} \delta(y_c, \sigma), \\
p^*(\mathbf{y}|\mathbf{x}) =& \exp\left(F(\mathbf{x}, \mathbf{y}; \alpha, \beta, \gamma)\right) / Z_{\mathbf{x}}, \\
Z_{\mathbf{x}} =& \sum_{\mathcal{Y}(\mathbf{x})} \exp\left(F(\mathbf{x}, \mathbf{y}; \alpha, \beta, \gamma)\right).
\end{aligned}
$$

The gradients for $\partial \mathbb{Q} / \partial \alpha$ and $\partial \mathbb{Q} / \partial \beta$ are the same as in Equation 5.16 and 5.17. The gradients of parameters for the similarity features become

$$
\frac{\partial \mathbb{Q}(\alpha, \beta, \gamma; D)}{\partial \gamma_{\mathbf{x}_c}} = \sum_{\hat{\mathbf{x}}} \mathrm{s}(\mathbf{x}_c, \hat{\mathbf{x}}_{\hat{c}}) p(\mathbf{y}_{\hat{c}} = \sigma | \hat{\mathbf{x}}) - \sum_{\bar{\mathbf{x}}} \mathrm{s}(\mathbf{x}_c, \bar{\mathbf{x}}_{\bar{c}}) p(\mathbf{y}_c = \sigma | \mathbf{x}). \tag{5.21}
$$

5.5 Experiments

5.5.1 Parts of Speech Tagging

We evaluate our semi-supervised conditional random fields algorithm with the parts-of-speech (POS) tagging problem. POS tagging is the task of marking up the words in a text with particular parts of speech such as noun, verb, article, adjective, preposition, pronoun, adverb, conjunction, and interjection, using features of the word and its context, i.e., adjacent and related words in a phrase, sentence, or paragraph. In the supervised context, CRFs have been successfully used for this well-known structured output prediction problem (Altun, 2005).

For our experiments, we have used a subset of the Penn TreeBank corpus which originally consists of approximately 7 million words of Part-of-Speech tagged Wall

Street Journal articles. We used Sections 24 for labeled and unlabeled data, Section 23 for the test data in particular. A list of observation attributes are provided in Table 5.1. There are 45 different categories of speech tags. Table 5.2 shows a subset of these word level POS tags. The average number of words in a sentence is around 22.

5.5.2 Similarity Metric

Our data features are represented as sparse binary vectors. In order to reflect the word based similarities, we experiment with the RBF kernel as in our multi-class experiments in Chapters 4 and 6, and the *Tanimoto coefficient*, which is also known as the *extended Jaccard coefficient* as our distance metric. The Jaccard coefficient

$$J(A, B) = \frac{A \cap B}{A \cup B}, \tag{5.22}$$

is a statistic that indicates the similarity between two sets as the ratio of the common attributes and the union of the two sets. The Tanimoto coefficient extends this metric to binary vectors as below

$$T(A, B) = \frac{\langle A, B \rangle}{\|A\|^2 + \|B\|^2 - \langle A, B \rangle}, \tag{5.23}$$

which is reminiscent of the cosine similarity

$$\cos(\theta_{AB}) = \frac{\langle A, B \rangle}{\|A\| \|B\|}. \tag{5.24}$$

However, in our experiments we have observed that Tanimoto coefficient works better than cosine similarity in practice.

CAP INI SENT INI	ENDS WITH ING
CAP INI DOT END	ENDS WITH ED
CAP INI CONTAINS DOT	ENDS WITH EN
CAP INI CONTAINS HYPEN	ENDS WITH LY
CAP INI CONTAINS DIGIT	ENDS WITH ER
CONTAINS DOT CONTAINS DIGIT	ENDS WITH EST
CONTAINS DOT CONTAINS HYPEN	ENDS WITH TH
CONTAINS DIGIT CONTAINS HYPEN	BEGINS WITH WH
CONTAINS DIGIT	DOT END
CUR WORD	SENT INI
TYPE FIRST LETTER	ENDINGS ONE
ALL CAPS	CAP INI

Table 5.1: Attributes used in the parts-of-speech tagging experiments.

CC	Coordinating conjunction	CD	Cardinal number
DT	Determiner	EX	Existential there
FW	Foreign word	IN	Preposition
JJ	Adjective	JJR	Adjective, comparative
JJS	Adjective, superlative	LS	List item marker
MD	Modal	NN	Noun, singular or mass
NNS	Noun, plural	NNP	Proper noun, singular
NNPS	Proper noun, plural	PDT	Predeterminer
POS	Possessive ending	PRP	Personal pronoun
PRP	Possessive pronoun	RB	Adverb
RBR	Adverb, comparative	RBS	Adverb, superlative
RP	Particle	SYM	Symbol
TO	to	UH	Interjection
VB	Verb, base form	VBD	Verb, past tense
VBG	Verb, gerund	VBN	Verb, past participle
VBP	Verb, non-3rd person present	VBZ	Verb, 3rd person present

Table 5.2: A subset of the word level Penn Treebank POS labels.

	Sup. CRF				PW-SSCRF				EP-SSCRF			
L	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}
10	51.8	5.54	51.4	5.77	45.8	3.11	45.6	3.29	46.4	3.44	46.6	3.29
20	42.0	4.00	41.8	3.56	37.2	0.84	37.4	1.14	37.8	0.84	38.2	0.84
30	35.4	1.67	35.2	1.30	32.2	0.84	33.0	0.71	32.8	0.84	33.6	0.55
40	31.4	1.82	31.2	1.10	29.2	1.48	30.6	1.14	29.4	1.14	31.0	1.22
50	28.6	1.95	28.4	1.82	26.8	0.84	28.6	0.55	26.8	0.84	29.2	0.45
100	21.0	1.00	21.2	0.84	21.0	0.00	24.6	0.55	21.4	0.55	24.8	0.84

Table 5.3: Token error % for supervised, pairwise constrained (PW-SSL) and expectation constrained (EP-SSL) CRFs in parts of speech tagging experiments averaged over 5 realizations with **RBF similarity**. The neighborhood size is taken as $\kappa = 5$ and the number of unlabeled sentences are 1000. tst indicates error on the test set with 4293 sentences. td indicates the error on unlabeled sentences, i.e., transductive error for PW and EP.

	Sup. CRF				PW-SSCRF				EP-SSCRF			
L	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}
10	26.2	3.63	26.6	3.78	33.6	3.91	33.6	3.58	32.6	4.28	32.8	3.70
20	35.2	3.49	35.4	3.58	41.8	1.92	42.4	2.51	41.4	2.19	41.8	2.28
30	41.6	2.07	42.2	2.39	47.6	1.82	47.8	1.79	47.0	1.41	47.2	1.79
40	47.0	2.35	47.2	3.11	53.0	1.58	52.8	1.10	52.4	1.67	52.4	1.14
50	50.2	2.68	50.0	3.16	55.8	2.59	55.6	1.67	55.2	2.17	55.2	1.64
100	61.2	1.30	60.8	1.10	63.8	2.39	62.0	1.00	63.6	2.70	62.0	1.00

Table 5.4: Macro-averaged F1 score for RBF similarity

	Sup. CRF				PW-SSCRF				EP-SSCRF			
L	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}
10	51.8	5.54	51.4	5.77	46.0	3.24	46.0	3.32	45.6	3.36	45.8	3.03
20	42.0	4.00	41.8	3.56	37.0	0.71	37.8	0.84	37.0	0.71	37.8	0.84
30	35.4	1.67	35.2	1.30	32.0	0.71	33.2	0.84	31.8	0.84	33.4	0.55
40	31.4	1.82	31.2	1.10	28.6	1.14	30.4	0.89	28.6	1.14	30.6	0.89
50	28.6	1.95	28.4	1.82	25.8	1.10	28.4	0.55	26.4	1.14	28.6	0.55
100	21.0	1.00	21.2	0.84	20.2	0.45	24.2	0.84	20.2	0.45	24.2	0.84

Table 5.5: Token error % for supervised, pairwise constrained (PW-SSL) and expectation constrained (EP-SSL) CRFs in parts of speech tagging experiments averaged over 5 realizations with **Tanimoto Coefficient similarity**. The neighborhood size is taken as $\kappa = 5$ and the number of unlabeled sentences are 1000. tst indicates error on the test set with 4293 sentences. td indicates the error on unlabeled sentences, i.e., transductive error for PW and EP.

	Sup. CRF				PW-SSCRF				EP-SSCRF			
L	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}	tst	σ_{tst}	td	σ_{td}
10	26.2	3.63	26.6	3.78	32.0	3.81	32.6	3.91	33.0	3.81	33.2	3.70
20	35.2	3.49	35.4	3.58	40.8	1.92	40.8	2.28	41.4	2.30	41.8	2.59
30	41.6	2.07	42.2	2.39	46.6	1.95	47.0	1.58	47.2	1.79	47.0	1.58
40	47.0	2.35	47.2	3.11	52.4	2.07	52.4	1.95	53.0	2.00	52.6	1.14
50	50.2	2.68	50.0	3.16	55.6	2.61	54.8	2.17	55.8	2.39	55.0	1.58
100	61.2	1.30	60.8	1.10	63.8	2.17	62.0	1.22	64.0	2.55	62.2	1.30

Table 5.6: Macro-averaged F1 score for Tanimoto coefficient similarity

5.5.3 Results

Tables 5.3, 5.5, 5.4 and 5.6 illustrate the experiment results for supervised conditional random fields (CRF), pairwise constrained semi-supervised CRF (PW-SSCRF) and expectation constrained semi-supervised CRF (EP-SSCRF) algorithms. We demonstrate the improvement in terms of both token error and macro-averaged F1 measure.

Macro-averaged F1 measure is an overall average of the local F-measures for each class. Since it neglects the class frequencies, it gives equal importance to the infrequent classes unlike the token error. In case the class distributions are significantly unbalanced, e.g., majority of the true labels belong to a single or default category such as unknown, macro-F1 gives a better insight. Therefore, given

$$q_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i} \ , \ \rho_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \text{ and } F_i = \frac{2q_i\rho_i}{q_i + \rho_i}, \tag{5.25}$$

the macro-averaged F1 is computed as below

$$F(\text{macro-averaged}) = \frac{\sum_i^C F_i}{C}, \tag{5.26}$$

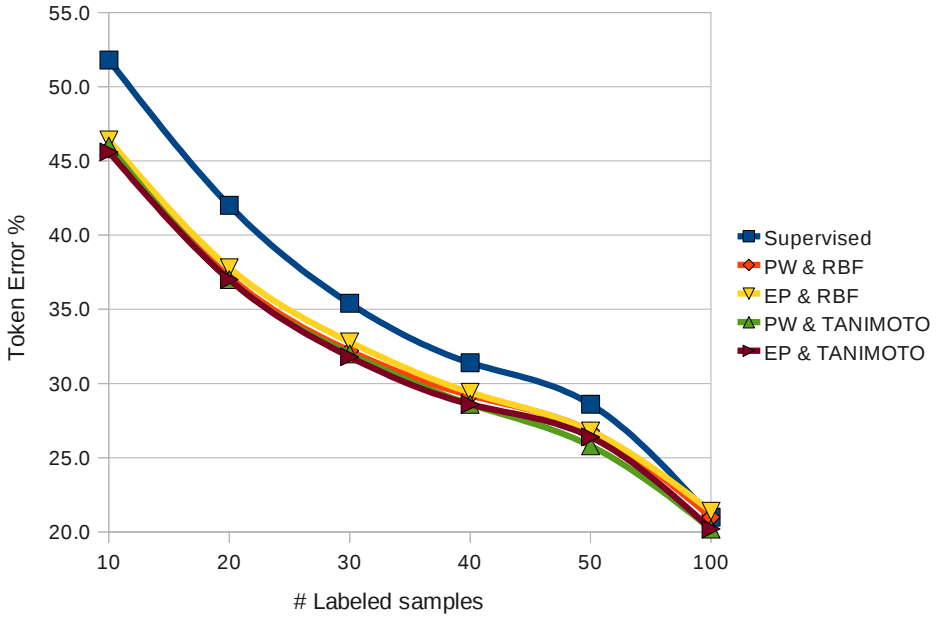
where the acronyms TP, FP and FN stand for true positives, false positives and false negatives respectively.

Tables 5.3 and 5.5 demonstrate the token error for the RBF kernel similarity, $s(x_i, x_j) = e^{-\|x_i - x_j\|^2}$ and Tanimoto Coefficient respectively $s(x_i, x_j) = T(x_i, x_j)$. The results of CRF, PW-SSCRF and EP-SSCRF algorithms with respect to increasing numbers of labeled sentences over five random splits of training data are

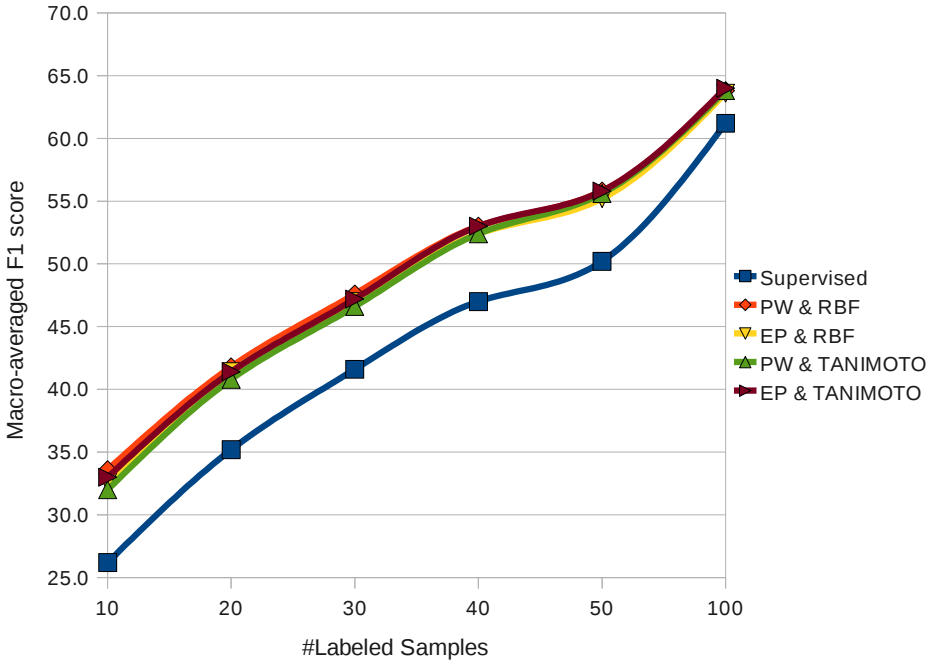
demonstrated for both unlabeled (indicated with td for transductive) samples and an out-of-sample test (indicated with tst) set of size 4293. In all semi-supervised experiments, five randomized sets of unlabeled sentence sets of size 1000 are used. We have taken the neighborhood size as five throughout all our experiments. The standard deviations, σ, over five splits are also provided.

Note that the number of labeled training instances is given in terms of sentences. As each sentence contains an average of 22 words, the unlabeled data set contains over $20,000$ unlabeled cliques. To reduce the computational burden, we have introduced very sparse similarities; we generated symmetric links only between labeled cliques and their five nearest neighbors. Therefore, for the case of 10 labeled sentences, around 200 labeled cliques are associated with similarity features with their neighboring five unlabeled cliques each. Accordingly, with the pairwise constrained formulation, enforcing symmetricity yields a total $\sim 200 \times 5 \times 2$ similarity features. As the table indicates, the semi-supervised formulations achieve better token error consistently. Note that there are 45 categories possible for each token. Tables 5.4 and 5.6 show the corresponding macro-averaged F1 scores. The improvement here is more significant as macro-averaged F1 score weighs rare categories equally. In other words, although the improvement in token error becomes less obvious as the number of labeled samples get higher, the improvement in F1 score stays consistent for higher values of L.

Note that these experiments are mainly for the proof of concept as NLP researchers can design more sophisticated feature representations and application specific similarity metrics for this complex problem. Figure 5.3 illustrates the results for all four variations of the semi-supervised algorithm against the supervised CRF.



(a) Token Error



(b) Macro-averaged F1

Figure 5.3: Token Error and Macro-averaged F1 score on test samples.

Chapter 6

Semi-supervised Learning via Constraint Augmentation

6.1 Introduction

In this chapter, we present an alternative approach to incorporate unlabeled data to the MaxEnt framework. Here, the main motivation is to enhance the data constraints of generalized MaxEnt directly in the primal rather than modifying the dual objective in an ad-hoc manner. Therefore, we propose improving our estimations on both the model and empirical expectations of the feature functions. In particular, the feature expectations predicted by the model are computed over both labeled and unlabeled data as these statistics do not require the labels. On the other hand, in order to improve the accuracy of the empirical feature expectations, we employ the smoothness assumption. We carry out this assumption by enhancing the empirical feature expectations obtained from labeled samples using unlabeled data that is in the vicinity of each labeled instance. Deriving the dual using these

new estimates in the data constraints yields a convex optimization problem. Hence, this framework provides a principled way of combining the ideas of information theoretic SSL methods with ideas of geometry based SSL method.

6.2 Generalized MaxEnt with Augmented Data Constraints

As stated in Chapter 2, the generalized MaxEnt objective is to minimize the divergence of the target distribution p from a reference distribution while penalizing the discrepancy between the empirical feature values $\tilde{\psi}$ of some pre-defined feature functions $\psi : \mathcal{X} \times \mathcal{Y} \to \mathcal{B}$ and their expected values with respect to the target distribution. We assume that we are given a sample D that consists of labeled data $\{(x_i, y_i)\}_{i=1}^{l}$ and unlabeled data $\{x_i\}_{i=l+1}^{n}$. In supervised learning, both the empirical joint distribution $\tilde{\pi}(x, y)$ and the marginal distribution $\tilde{\pi}(x)$ are derived from labeled data that is assumed to be sampled from a fixed but unknown distribution π^*. Intuitively, generalized MaxEnt can yield more accurate estimates of the target model distribution p, as empirical feature expectations given by $\tilde{\psi} = \mathbb{E}_{\sim\tilde{\pi}(x,y)}[\psi(x, y)]$, and model feature expectations, $\mathbb{E}_{\sim\tilde{\pi}(x)}\mathbb{E}_{y\sim p(.|x)}[\psi(x, y)]$ approach their counterparts with respect to the true underlying distributions, or equivalently as the empirical joint $\tilde{\pi}(x, y)$ and empirical marginal distributions $\tilde{\pi}(x)$ approach their true values. This can be seen by the risk bounds of the negative dual problem given by

$$\mathbb{R}(\lambda; D) := \min_{\lambda} \mathbb{E}_{x\sim\tilde{\pi}(x)} \left[b(x; \lambda)\right] - \left\langle \lambda, \tilde{\psi} \right\rangle + \epsilon\|\lambda\|_{\mathcal{B}^*}. \tag{6.1}$$

where $b(x;\lambda)$ is the convex conjugate of $\mathbb{D}_x$ and $g = h_2$ (see Table 2.1) and Lemma (2).

Lemma 5 *Let $\psi^* = \mathbb{E}_{(x,y)\sim\pi^*}[\psi(x,y)]$ denote the true statistics of the features, λ^* and $\tilde{\lambda}$ denote the minimizers of $\mathbb{R}(.;\pi^*)$[1] and $\mathbb{R}(.;D)$. The difference between the optimal and the empirical risk is given by*

$$\mathbb{R}(\tilde{\lambda};D) - \mathbb{R}(\lambda^*;\pi^*) \leq \|\lambda^*\|_{\mathcal{B}^*} \left\|\psi^* - \tilde{\psi}\right\|_{\mathcal{B}} + \sum_x |\pi^*(x) - \tilde{\pi}(x)| b(x;\lambda^*) \tag{6.2}$$

Proof

$$\begin{aligned}
&\mathbb{R}(\tilde{\lambda};D) - \mathbb{R}(\lambda^*;\pi^*) \\
&= \mathbb{R}(\tilde{\lambda};D) - \mathbb{R}(\lambda^*;\pi^*) + \mathbb{R}(\lambda^*;D) - \mathbb{R}(\lambda^*;D) \\
&\leq \mathbb{R}(\lambda^*;D) - \mathbb{R}(\lambda^*;\pi^*) \\
&= \left\langle \lambda^*, \psi^* - \tilde{\psi} \right\rangle + \mathbb{E}_{x\sim\tilde{\pi}(x)}[b(x;\lambda^*)] - \mathbb{E}_{x\sim\pi^*(x)}[b(x;\lambda^*)]
\end{aligned}$$

The equalities hold by basic algebra. The inequality holds by construction $\mathbb{R}(\tilde{\lambda};D) \leq \mathbb{R}(\lambda^*;D)$. We get the claim using Hölder's Inequality and $a \leq |a|$ for any a. ■

The risk bound can be further analyzed in terms of complexity measures such as Rademacher averages (Bartlett et al., 2003). However, Lemma 5 is sufficient to show that the empirical risk approaches to the optimal risk as empirical means $\tilde{\psi}$ and marginal distribution $\tilde{\pi}$ approach their true values. In the following sections, we propose improving these estimates by with the help of unlabeled data.

[1] with slight abuse of the notation for simplicity

6.2.1 Improving Expected Feature Values

In order to achieve better expected feature values $\mathbb{E}_{\sim\tilde{\pi}(x)}\mathbb{E}_{y\sim p(.|x)}[\psi(x,y)]$, we make the simple observation that these estimators do not require labeled data. Thus, one can simply use the marginal distribution obtained from both labeled, L and unlabeled U data $\{x_i\}_{i=1}^{n}$, denoted by $\tilde{q}_m$. Then, the empirical means of features on labeled data are forced to approximately match their expected values on the whole data set, both labeled and unlabeled. If L and U are coming from the same distribution, which is a basic yet commonly used assumption in inference problems, $\tilde{q}_m$ is closer to the true marginal than the empirical marginal distribution estimated using labeled data only. This yields more accurate expected feature values.

Studying the convex dual problem, we see that only the first term of (6.1) changes to $\mathbb{E}_{x\sim\tilde{q}_m}[b(x;w)]$ and the risk bound of Lemma 5 holds with respect to $\tilde{q}_m$ and can render empirical risk closer to the optimal risk.

In practice, instead of a uniform distribution over labeled and unlabeled data, we define $\tilde{q}_m$ by splitting the probability mass into two parts and assign uniform distribution over the labeled and unlabeled splits individually,

$$\tilde{q}_m(x_i) = \begin{cases} 1/2l & 1 \leq i \leq l \\ 1/2(n-l) & l < i \leq n \end{cases}.$$

Since number of labeled data is smaller than unlabeled data, this definition places higher importance to labeled data and the resulting optimization problem is not heavily driven by unlabeled data.

6.2.2 Improving Empirical Feature Values

In Section 6.2.1, we used unlabeled data in order to improve the expected values of the features. We now investigate employing unlabeled data to augment their empirical counterparts.

In semi-supervised learning, the size of the labeled data is typically quite small. If the empirical feature means $\tilde{\psi}$ is derived from only labeled data, the estimation error $\|\psi^* - \tilde{\psi}\|$ can be very large leading to a large difference between the empirical and optimal risk (6.2).

The estimation error of $\tilde{\psi}$ is especially problematic for semi-supervised learning methods that enforce empirical feature values to match their expected counterparts over unlabeled data either by imposing minimal divergence in the primal as outlined in Section 6.2.1 or by adding a regularization term in the dual problem as in (Mann & McCallum, 2007). In particular, if the features are binary and sparse, which is commonly observed in natural language processing and information retrieval, many features may never be observed in a small labeled sample. If the divergence is imposed via constraints, i.e., $g = h_2$ (see Table 2.1), then the approximate moment matching constraints of Section 6.2.1 can become infeasible. This in turn can lead to over-fitting in the dual problem. One possible solution is to relax these constraints by making ϵ larger. However, this may render the unlabeled data ineffective. Alternatively, one can augment the empirical values of the features using unlabeled data. To this extent, we propose employing the smoothness assumption which enforces observations that are close to each other with respect to the intrinsic geometry of the data to have the same label.

There are various ways to improve the empirical means. We employ the simplest option and aggregate a weighted average of the unlabeled instances that are

adjacent to each labeled instance and the labeled instance itself

$$\hat{x}(x_i) = \frac{n_i \sum_{j=l+1}^{n} s(x_i, x_j) x_j + x_i}{n_i \sum_{j=l+1}^{n} s(x_i, x_j) + 1}, \tag{6.3}$$

where n_i is the number of neighbors of x_i and $s(x_i, x_j)$ denotes the similarity of x_i and x_j.[2] Here the denominator enforces proper normalization and n_i guarantees that the unlabeled data is emphasized proportionally with the density of the neighborhood region. This corresponds to placing a distribution around each labeled instance with respect to the intrinsic geometry of the data and computing the empirical means from this new distribution. We call these new statistics *augmented means*.

We use the augmented means along with the new marginal distribution $\tilde{q}_m$ in the generalized MaxEnt framework as

$$t_u := \min_{p \in \mathcal{P}} \{ \mathbb{E}_{\underline{x \sim \tilde{q}_m}} [\mathbb{D}_x(p_x | q_x)] + g \left(\mathbb{E}_p[\psi]; \mathbb{E}_{x,y \sim \tilde{\pi}(x,y)} \left[\psi(\underline{\hat{x}(x)}), y) \right], \epsilon \right) \}.$$

If the smoothness assumption holds with respect to the employed similarity metric, it can be argued that augmented means are better estimates of ψ^* than the standard empirical means and this can lead to better generalization properties.

[2]Refer to Chapter 3 for a discussion on similarity functions.

6.2.3 Special Cases

Semi-supervised Logistic Regression

Using standard convex duality techniques yields the following optimization problem for logistics regression

$$\mathbb{R}(\lambda; D) = \sum_{j=1}^{n} \tilde{q}_m(x_j) \log \sum_{y} \left(\exp \langle \lambda, \psi(x_j, y) \rangle\right) - \frac{1}{l} \sum_{j=1}^{l} \langle \lambda, \psi(\hat{x}(x_j), y_j) \rangle + \Omega(\lambda), \tag{6.4}$$

as a special case of our semi-supervised learning framework. Equation (6.4) gives a convex optimization problem which can be solved using standard gradient methods. The relation between the primal p and dual λ variables is given by

$$p(y|x; \lambda) \propto \exp(\langle \lambda, \psi(x, y) \rangle). \tag{6.5}$$

It is worthwhile to point out that Equation (6.5) also holds for out-of-sample input points, thus our framework naturally extends to new test data unlike transductive algorithms. Once the augmented empirical feature expectations are computed, the computational complexity of optimizing $\mathbb{R}(\lambda; D)$ becomes the same as the complexity of logistic regression over both labeled and unlabeled data. Hence, SSLR can scale to large unlabeled data sets.

Semi-supervised Kernel Logistic Regression

Representer Theorem states that (6.4) (the sum of a loss function over labeled and unlabeled data and the RKHS norm) admits the form

$$\lambda^* = \sum_{i=1}^{n} \sum_y \alpha_{i,y} \psi(x_i, y) + \sum_{i=1}^{l} \bar{\alpha}_i \psi(\hat{x}(x_i), y_i).$$

When we substitute the solution to (6.4), we obtain our semi-supervised KLR (SSKLR) loss function given by

$$\mathbb{R}(\alpha; D) = \sum_{i=1}^{n} \tilde{q}_m(x_i) \log \sum_y \exp(h(x_i, y; \alpha)) - \frac{1}{l} \sum_{i=1}^{l} h(\hat{x}(x_i), y_i; \alpha) + \epsilon\, \alpha^T K \alpha, \tag{6.6}$$

where $h(x, y) = \langle \lambda^*, \psi(x, y) \rangle$ for λ^* defined above and K is the gram matrix over both labeled and unlabeled data. In general $k((x, y), (x', y')) = \delta_{y,y'} \bar{k}(x, x')$ for any Mercer kernel $\bar{k}$. Note that the kernel is evaluated over all data pairs in the sample. This enables capturing kernel-induced nonlinearity over unlabeled data as well as labeled data. This is an advantage over other information theoretic approaches where unlabeled data is employed only in an entropic regularization term. The disadvantage is the computational complexity of SSKLR, which is the same as the computational complexity of supervised KLR on data of size n.

6.3 Incorporating class distributions into generalized MaxEnt

Often one can assume that the class distributions in labeled and unlabeled data are similar. In fact, explicitly imposing balanced label proportions on labeled and unlabeled samples has been successfully used in semi-supervised learning literature (Collobert et al., 2006; Chapelle & Zien, 2005; Karlen et al., 2008). Such constraints can be naturally imposed in our maximum entropy framework by defining binary features of the form $\psi_\sigma(x, y) = \delta_{y=\sigma}$, that are only a function of the output variable as used in (Mann & McCallum, 2007). These features are commonly called *label features*. Adding them to the existing feature vector $\psi(x, y)$ corresponds to imposing the expected label distribution on unlabeled data to match the (augmented) empirical distribution on labeled data. We will refer to such constraints as *label balancing* constraints.

6.4 Experiments

In this section, we provide an empirical evaluation of the SSLR (Section 6.2.3) and SSKLR (Section 6.2.3) algorithms. In our experiments we use data from two different origins, that have been extensively analyzed in previous SSL work. The first two data sets (referred as *small data sets*), g50c and text, are provided by the authors of (Chapelle & Zien, 2005), whereas the rest of the data sets are from the benchmarks provided in (Chapelle et al., 2006). See Table B.1 for the geometry assumptions as well as other properties of each data set.

The hyper-parameters of our algorithm are the neighborhood size κ, the regu-

Table 6.1: MTE on small data sets.

	g50c	text
SVM	8.32	18.86
Neural Net	8.54	15.87
LR	7.96	16.64
SSLR	7.96	16.87
SSLR+LB	7.84	9.95
SSLR+Aug	5.58	13.82
SSLR+Aug+LB	**4.94**	**9.62**
SVMlight TSVM	6.87	7.44
CCCP-TSVM	5.62	7.97
TSVM	5.80	5.71
LapSVM	**5.40**	10.40
LDS	**5.40**	**5.10**
Label Propagation	17.30	11.71
Graph	8.32	10.48
TNN	6.34	6.11
ManTNN	5.66	5.34

larization constant ϵ_1 for the model feature parameters and ϵ_2 for the label feature parameters, and finally the kernel bandwidth σ in the case of a RBF kernel. We considered a range of hyper-parameters for model selection, $\kappa \in \{10, 25, 50, 100\}$ and $\epsilon_1, \epsilon_2 \in \{e^{-1}, e^{-2}, e^{-3}, e^{-4}\}$. We set $\alpha = \eta^{-2}$ where η is the median of pairwise distances. In order to reflect the real-life scenario as closely as possible, we performed cross validation on a subset of labeled samples following the experimental setup described in Section 4.3.1.

Table 6.1 and 6.2 present the experimental results from the small data sets and SSL benchmark data sets. We report the error both with (denoted by +Aug) and without augmented empirical feature expectations which are computed over $\{\hat{x}(x)_{i=1\ldots l}\}$ given by Equation (6.3) and $\{(x)_{i=1\ldots l}\}$ respectively. Additionally, we investigate the effect of label balancing constraints (denoted by +LB) which were introduced in Section 6.3. The first three lines report the supervised performance of

Table 6.2: MTE on SSL benchmark data sets.

	Digit1	USPS	COIL	g241c
1-NN	6.12	7.64	23.27	40.28
SVM	5.53	9.75	22.93	23.11
KLR	5.25	9.20	24.63	22.28
SSKLR	4.26	6.43	22.28	23.50
SSKLR+LB	4.32	6.30	21.48	21.88
SSKLR+Aug	3.69	7.71	16.48	21.03
SSKLR+Aug+LB	**3.59**	**6.18**	**16.08**	**15.31**
TSVM	6.15	9.77	25.80	18.46
MVU + 1-NN	3.99	6.09	32.27	44.05
LEM + 1-NN	2.52	6.09	36.49	42.14
QC + CMN	3.15	6.36	10.03	22.05
Discrete Reg.	2.77	**4.68**	**9.61**	43.65
SGT	2.61	6.80	-	17.41
Cluster-Kernel	3.79	9.68	21.99	**13.49**
Data-Dep. Reg.	**2.44**	5.10	11.46	20.31
LDS	3.46	4.96	13.72	18.04
Laplacian RLS	2.92	**4.68**	11.92	24.36
CHM (normed)	3.79	7.65	-	24.82

various methods, SVMs, single layer neural networks, 1-Nearest Neighborhood(1-NN) and (kernel) logistic regression. At the bottom of the tables, the performances of the most competitive semi-supervised algorithms reported in (Chapelle et al., 2006), namely Transductive SVM (TSVM) (Vapnik, 1998), Cluster Kernel (Chapelle et al., 2003), Discrete Regularization (Chapelle et al., 2006), Data Dependent Regularization (Chapelle et al., 2006), Low Density Separation (LDS) (Chapelle & Zien, 2005) among others. The reader may refer to (Chapelle et al., 2006) for a comparison with a wider selection of algorithms.

Analyzing results, we observe that our semi-supervised learning approach almost always yields improvement over supervised learning. Even without feature augmentation, we may observe significant error reduction, for example on Digit1

data set (20%). When both augmentations are used (when the optimization corresponds to (6.4) or (6.6).), our method becomes competitive ranking in the middle across most data sets. When label balancing constraints are also included the results improve further. Perhaps the most attractive property of this approach is its generality. In particular, it provides natural means to incorporate different geometry assumptions of the data and thus performing compatible across all data sets. This is not the case for other SSL methods since they incorporate a single geometry assumption. Comparing different settings of our algorithm, we conclude that imposing label balancing constraints always perform preferably. We observed that manifold data sets prefer κ to be small (25, 50) whereas the other data sets prefer larger κ. We conjecture that with large κ the MST can jump across classes, which might be improved with a less noisy estimate for example by using random walks.

Chapter 7

Combining Semi-Supervised and Active Learning Paradigms

Joint work with Oliver Kroemer[†]*, Renaud Detry*[‡]*, Yasemin Altun*[†]*, Justus Piater*[‡]*, and Jan Peters*[†]

7.1 Introduction

In this chapter, we propose an approach to combine *active learning* with our SSL algorithm with constraint augmentation introduced in Chapter 6. Active learning refers to the learning paradigm where an algorithm has the means to query the supervisor, which is often referred to as the *oracle*, for labels. Although it is motivated by the scarcity of labeled training data as SSL, active learning primarily focuses on the questions of which data to label, particularly the selection criteria.

† Max Plank Institute for Biological Cybernetics, Spemannstraße 38, Tuebingen Germany {oliverkro,altun,jan.peters}@tuebingen.mpg.de

‡ Department of Electrical Engineering and Computer Science Montefiore Institute, Université de Liège 4000 Liège Sart Tilman Belgium, {renaud.detry,justus.piater}@ulg.ac.be

Unlike SSL, active learning applies only to incremental learning setups as the system is expected to be retrained or updated with results of the queries.

We evaluate our synergistic method in the context of robot grasping, a real-life problem where the number of labeled grasps is extremely scarce. Therefore, our goal is to learn discriminative probabilistic models of object-specific grasp affordances despite limited supervision. Another concern is that the proposed method does not require an explicit 3D model but rather learns an implicit manifold defining a probability distribution over grasp affordances.

We obtain a large set of hypothetical grasp configurations from visual descriptors that are associated with the contours of an object. While these automatically generated hypothetical configurations are abundant, labeled configurations are very scarce as these are acquired via experiments carried out by the robot as executing and labeling grasps of novel objects is a time-consuming process that requires human monitoring as otherwise the robot may damage the objects. However, a vast number of hypothetical grasp configurations can be generated by a vision model, in our context the Early Cognitive Vision reconstructor. Even though they are suggested as potentially stable grasps based on heuristics, the hypothetical configurations can not be given any confident labels, as they have not been empirically tested, and are therefore effectively unlabeled.

Kernel logistic regression (KLR) via joint kernel maps is trained to map these hypothesis space of grasps into continuous class conditional probability values indicating their achievability. We propose a soft-supervised extension of KLR and a framework to combine the merits of semi-supervised and active learning approaches to tackle the scarcity of labeled grasps. Experimental evaluation shows that combining active and semi-supervised learning is favorable in the existence of

an oracle. Furthermore, semi-supervised learning outperforms supervised learning, particularly when the labeled data is very limited.

7.2 Motivation

Grasping is a fundamental skill for robots that need to interact with their environment in a flexible manner. A wide spectrum of tasks (e.g., emptying a dishwasher, opening a bottle, or using a hammer) depend on the capability to reliably grasp an object or tool as part of a larger planning framework. It is therefore imperative that the robot learns a task-independent model of an object's grasp affordances in an efficient manner. Given such a flexible model, a planner can be used to grasp and manipulate the object for a wide range of tasks. In this chapter, we investigate learning probabilistic models of grasp affordances for an autonomous robot equipped with a 3D vision system (see Figure 7.1).

Until recently, the most predominant approach to grasping has been constructing a full 3D model of the object and then employing various techniques such as friction cones (Mason & Salisbury, 1985) and form- and force- closures (Bicchi & Kumar, 2000). Given the difficulties of obtaining a 3D model with sufficient accuracy to reliably apply these techniques, designing statistical learning methods for grasping has become an active research field (Detry et al., 2009a; de Granville et al., 2006; Saxena et al., 2008a; Saxena et al., 2008b). These new learning methods often employ efficient representations and vision based models, without requiring full 3D reconstruction, in order to provide a more robust alternative to traditional approaches. Most of the previous work focuses only on learning successful grasps (Detry et al., 2009a; de Granville et al., 2006).

The investigation of discriminative learning methods for grasp affordances presented in this work continues on from previous approaches of probabilistic grasp affordance models, namely (Saxena et al., 2008a) and (Saxena et al., 2008b). In (Saxena et al., 2008a), the authors propose extracting a set of 2D image features and apply a discriminative supervised learning method to model grasp affordance probabilities given the 2D image. In (Saxena et al., 2008b), this approach is extended with a probabilistic classifier using a set of arm/finger kinematics features in order to identify physically impossible 2D points for the robot to reach. The strength of their approach comes from the combination of two important sources of information, image and kinematic features, in a probabilistic manner.

To incorporate unlabeled grasp configurations in the discriminative learning of grasp affordance probabilities, we use the semi-supervised kernel logistic regression (SSKLR) algorithm introduced in Section 6.2.3. SSKLR method provides a principled way of combining information from the object as well as from the robot hand via *joint kernels* (Bakir et al., 2003). By training a single classifier using a joint kernel, as opposed to training two separate classifiers, e.g., (Saxena et al., 2008b), our approach can capture the non-linear interactions of the morphology of the robot hand and the surface characteristics of the object implicitly. The system therefore does not have to rely on explicit representations such as closed form geometric descriptions or libraries of feasible grasps. We investigate using unlabeled data in our KLR approach to reduce the number of labeled grasps needed. In particular, we combine a novel semi-supervised KLR method with active learning in the context of robot grasping.

As mentioned earlier, active learning assumes the existence of an annotator, commonly referred to as the *oracle*, that can provide labels to *queries*. In a robotics

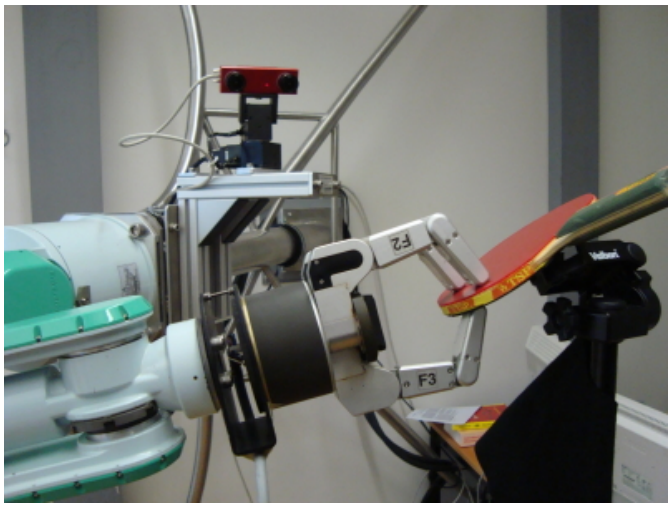


Figure 7.1: Three-finger Barrett hand equipped with a 3D vision system. A table tennis paddle is used in the experiments.

context, the annotator corresponds to the robot attempting to perform new grasps. The goal of active learning is to guide the robot to evaluate the most informative grasps so that the classification error is reduced with the fewest queries possible.

This framework enables the robot to learn grasp affordances by autonomously evaluating new grasps in an incremental manner. We provide comparisons between supervised, semi-supervised as well as a hybrid of semi-supervised and active learning setups, as minimizing the need for large amounts of labeled data is the primary concern. Our experimental evaluation shows not only that the proposed active learning and semi-supervised learning methods individually improve the system's performance, but the fact that the amount of necessary annotated data is also significantly reduced when supervised learning is combined with active learning.

This chapter is organized as follows. Section 7.3 gives a detailed explanation of the machine learning techniques evaluated in the context of robot grasping. Section 7.5 overviews relevant work in the literature. In Section 7.6, we describe the

details regarding the acquisition of the visual features and the joint kernel designed for this particular application. Then, we introduce the experimental setup, give empirical results and provide a comparison of supervised, semi-supervised and active learning approaches. Finally, Section 7.7 provides a discussion and directions for future work.

7.3 Learning Probabilistic Grasp Affordances Discriminatively under Limited Supervision

For training, we use the SSKLR loss given in Equation (6.6) so that the learning process can accommodate unlabeled data. SSKLR models the conditional distribution $p(y|x)$ enabling the probabilistic interpretation of the outputs. Such a natural interpretation is advantageous for the particular application of grasp affordance learning as the grasp affordances can be easily compared and easily mapped to policies by the robot's motion planner.

We use a joint kernel as a distance metric between pairs of grasp configurations. This kernel decomposes into separate distance measures on the position and rotation parameters as described in Section 7.6.2. We use this kernel both in the SSKLR algorithm and as the distance metric the compute the augmented empirical expectations. In the next section, we describe the uncertainty criterion to select grasps for the queries in the active learning setting.

7.4 Uncertainty based active learning

Active learning algorithms differ based on their assumptions on the potential merit of samples. For instance, in the *uncertainty sampling* approach which we employ here, the algorithm queries the sample for which the classifier generates the most uncertain outputs at a given time. On the other hand, *query-by-committee* approaches decide on the queries based on the disagreement of a set of classifiers on the unlabeled samples. The higher the disagreement measure on a sample, the more beneficial it is to know its actual label. Other approaches favor querying samples which have the biggest potential impact on the model parameters or highest estimated error reduction (Zhu et al., 2003) in case its label is acquired. A thorough literature survey can be found in (Settles, 2009).

We can employ active learning in scenarios where the robot has the means to choose what to learn. For the active selection of grasps, we use uncertainty sampling (Lewis & Catlett, 1994) which is straightforward for probabilistic models. In this method, the algorithm queries for the grasps on which it is the least confident. Therefore, at each iteration, the algorithm requests the true label for the grasp, x^* that has the highest class conditional entropy among the set of unlabeled grasps

$$x^* = \underset{x \in U}{\operatorname{argmax}} \ \mathbb{H}(p(y|x)).$$

In turn, the robot carries out the configuration that corresponds to x^* and labels it accordingly.

7.5 Related Work

Efficient representation and vision based modeling of grasp configurations is an active research field (Detry et al., 2009a; Saxena et al., 2008a). We follow the methodology in (Detry et al., 2009a) to obtain grasp pose candidates and orientations. However, the authors learn grasp densities using successful grasps only, whereas here, we model the class conditional probabilities of both successful and unsuccessful grasp configurations in a discriminative manner. Furthermore, we focus on the scarcity of the labeled data points and evaluate active and semi-supervised learning algorithms with the smallest number of annotated experiences possible.

De Granville et al. present a method where the robot learns a mapping from object representations to grasps from human demonstration (de Granville et al., 2006). They cluster the orientations of grasps and each cluster is associated with a canonical approach orientation. The authors indicate that limiting the encoding to orientations or excluding position knowledge, is due to their underlying assumption that orientation and position are independent.

As labeled data collection is expensive for most robotics tasks, active learning techniques have already been considered. (Salganicoff et al., 1996) proposed some of the earliest work on uncertainty based active learning for vision-based grasp learning by modifying the ID3, a decision tree algorithm. (Montesano & Lopes, 2009) also propose a method to learn local visual descriptors of good grasping points via self-experimentation. Their method associates the outputs with confidence values. Morales et al. propose an active learning method for grasp reliability (Morales et al., 2004). They use a k-nearest neighbors approach to learn grasp affordance probabilities whereas we propose an information theoretic approach and

kernel methods extended to semi-supervised learning.

In machine learning, various methods to combine semi-supervised and active learning have been proposed to exploit the merits of both approaches (Zhu et al., 2003; Tür et al., 2005). We attempt to be the first in the context of robotics. The active learning methodology proposed in (Tür et al., 2005) is similar to ours, as the authors employ confidence sampling for active learning based on the probabilistic outputs of a logistic regression classifier. Their method differs from ours since they perform semi-supervised learning via self-training, whereas we propose a soft-labeling approach motivated by the maximum entropy framework.

7.6 Empirical Evaluation

We have empirically evaluated the methods described in Section 7.3 on a 3-finger Barrett robot with simple objects such as a table tennis paddle and a watering can. For supervised learning, we have used a Kernel Logistic Regression classifier and the joint kernel defined on position and orientation features. The labels were collected by a human demonstrator. For the semi-supervised experiments we have used SSKLR loss given in Equation (6.6) in Section 6.2.3. Details on the experimental setup such as data collection, preprocessing, model selection and the results are given below.

7.6.1 Visual Feature Extraction For Grasping

The inputs of our learning algorithm are represented as grasp configurations generated from Early Cognitive Vision (ECV) descriptors (Krüger et al., 2004; Pugeault, 2008), which represent short edge segments in 3D space, as described in (Detry

et al., 2009a). Accordingly, an ECV reconstruction is performed. Next, pose hypotheses for potential grasps are generated from pairs of co-planar ECV descriptors. The grasp position is set to the location of one of the ECV descriptor pairs whereas the grasp orientation is computed from the normal of the plane on which these descriptors lie. The assumption is that two co-planar segments constitute a potential edge of the object that the robot hand can hold. However, this is quite optimistic as many infeasible edges and orientations will be included in the hypothesis space, see Figure 7.2. Hence, we need a learning algorithm to discriminate between the feasible and infeasible grasps contained in this set.

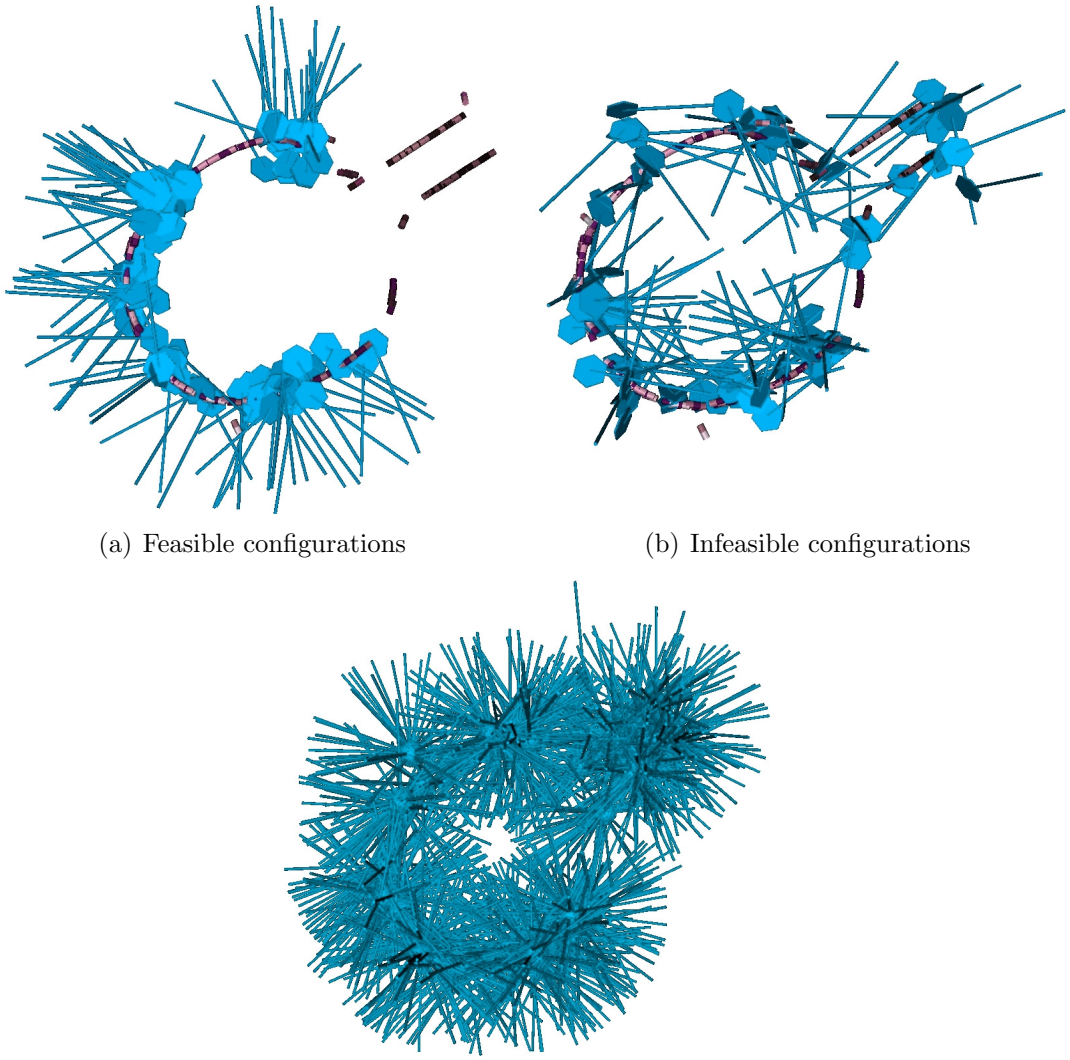
7.6.2 Joint Kernel

Each grasp configuration $x = (v, r)$ consists of seven parameters, three from the 3D position v of the robot hand in the object's reference frame,[1] and four from the unit quaternions r defining the rotation.[2] These values have different coordinate systems and have to be treated separately in order to obtain a proper distance metric. This distance metric, which indicates the similarity of two configurations, is employed for both the kernel computation and the similarity measure required by semi-supervised learning (see Section 6.2.2). We define the joint kernel as

$$K\left(x_a, x_b\right) = \exp\left(-\frac{\|v_a - v_b\|^2}{2\sigma_v^2} - \frac{f(\theta_{ab})^2}{2\sigma_{f(\theta)}^2}\right),$$

[1]The object relative reference frame is a coordinate system that is attached to the object such that any rigid body transformation applied to the object will also be applied to the coordinate system and objects therein.

[2]Unit quaternions are widely used in the context of robotics to represent spatial rotations in three dimensions. As well as their convenience as a mathematical notation, they are preferred as they avoid the problem of gimbal lock.



(a) Feasible configurations

(b) Infeasible configurations

(c) Hypothesis space

Figure 7.2: Kernel logistic regression algorithm is used to discriminate the successful 7.2(a) and unsuccessful grasps 7.2(b) lying on separable nonlinear manifolds. The entire hypothesis space 7.2(c) of potential grasp configurations extracted from pairs of ECV descriptors contains feasible grasps as well as infeasible configurations.

where $f(\theta_{ab})$ is the rotational distance between $x_a = (v_a, r_a)$ and $x_b = (v_b, r_b)$, and σ_v and $\sigma_{f(\theta)}$ are the standard deviation of the pose and rotation distances of all pairs of samples respectively. In order to cope with the double cover property (Kuffner, 2004) of quaternions, we compute the rotational distance $f(\theta_{ab})$, as the smaller angle between the two unit length quaternions r_a and r_b. This definition allows us to use a Gaussian distribution on this rotational distance metric. Here, θ_{ab} is the angle of the 3D rotation that moves r_a to r_b, i.e., $\theta_{ab} = \theta(r_a, r_b) = \arccos(r_a^T r_b)$, and

$$f(\theta_{ab}) = \min\{\theta(r_a, r_b), \theta(r_a, -r_b)\}.$$

For further details on distance computations between unit quaternions see (Kuffner, 2004). This joint kernel is similar to that in (Detry et al., 2009b) in the way it decomposes into kernels on position and rotation features. However, there the authors employ a Dimroth-Watson distribution to get the rotational kernel as whereas we use the Gaussian distribution, which is preferable due to the computational complexity of the former.

7.6.3 Experimental Setup

We collected 200 samples, 100 successful (positive labels) and 100 unsuccessful (negative labels) grasps. We preprocess the data by normalizing the position parameters to zero mean and unit variance. The unit quaternions do not require preprocessing. In all experiments, we fix the hyperparameters at the initial step using fourfold cross validation. The model variance in semi-supervised and active learning can be high as the training set is typically very small. In order to

compensate for the resulting high variance, we have generated 20 random training sets with equal numbers of positive and negative samples and corresponding test partitions. We report the average classification error on these random test sets. For the active learning scenario, we used a separate active learning pool.

Our framework has two hyper-parameters which are to be set during the model selection. The first parameter, κ is the size of the neighborhood. The second parameter, ϵ is the regularization constant of the kernel logistic regression algorithm. We sweep over a grid of values $\kappa = \{10, 20, 30, 50\}$, and $\epsilon = \{10^{-2}, 10^{-3}, 10^{-4}\}$.

7.6.4 Evaluation on collected data sets

We evaluate the supervised and semi-supervised models as the amount of labeled data increases. When additional data is selected with uncertainty sampling, we assess the active supervised and active semi-supervised performances. In all experiments, we train initial models with 10 randomly selected labeled samples. We perform model selection in this setup and fix the value of the hyper-parameters for the following experiments. The semi-supervised algorithm uses an additional unlabeled set of size 4000. All results are the averages over the models trained over 20 realizations of the training set and the fourfold cross validation.

First, we empirically evaluate the performance of semi-supervised learning versus supervised learning. Figure 7.3 shows the improvement of classification error as randomly selected samples are added to the training sets one at a time (i.e., classification error of KLR and SSKLR with respect to increasing labeled data size). As expected, when the size of the labeled data is small, semi-supervised learning is advantageous over supervised learning. The difference diminishes as the data set gets larger.

An alternative evaluation measure is the *perplexity* of the data,

$$2^{H(p)} = 2^{\left(\sum_x \sum_y -p(y|x)\log_2 p(y|x)\right)},$$

which measures the uncertainty of the predictions of the trained models. This information theoretic measure is commonly used for probabilistic models in fields such as speech recognition and natural language processing (Jurafsky & Martin, 2000). In Figure 7.5(a), we plot the perplexity of KLR and SSKLR. This figure shows that the semi-supervised model is more confident (smaller perplexity) of its predictions than the supervised model, and thus yields preferable results. We also note that the variance of perplexity across different validation sets are smaller in the case of SSKLR, when the dataset is small. This renders semi-supervised learning more robust compared to supervised learning in real-life scenarios.

Secondly, we comparatively demonstrate the impact of active learning. Figure 7.4 illustrates the performance of both KLR and SSKLR as they are incrementally retrained with uncertainty based sampling. The corresponding perplexity plots are shown in Figure 7.5(b). The comparison of KLR and SSKLR in the active learning setting shows a similar behavior to that of random selection, Figure 7.3 and 7.5(a).

Figure 7.6 illustrates the classification error rate for all four scenarios together. For the supervised classifier, the improvement rate is clearly faster with active learning than random selection. A 10% error rate is achieved with 17 samples whereas to get the same error rate 40 samples are required for the random selection case.

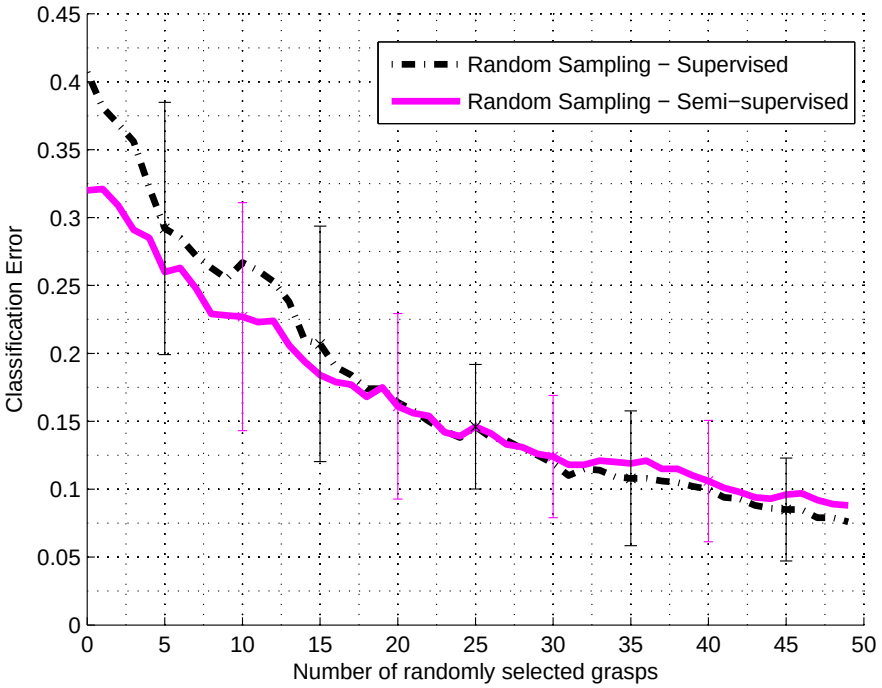


Figure 7.3: Supervised and semi-supervised logistic regression error on the validation sets versus the number of randomly selected labeled samples added to the initial training of size 10. Model selection is carried out at the initial step with 10 samples. 50 samples are added in an incremental manner and all models are retrained at each iteration. SSKLR uses an unlabeled training set of size 4000. The neighborhood size for the similarity based augmentation, κ, is set to 30. Semi-supervised is learning is advantageous at the initial stages.

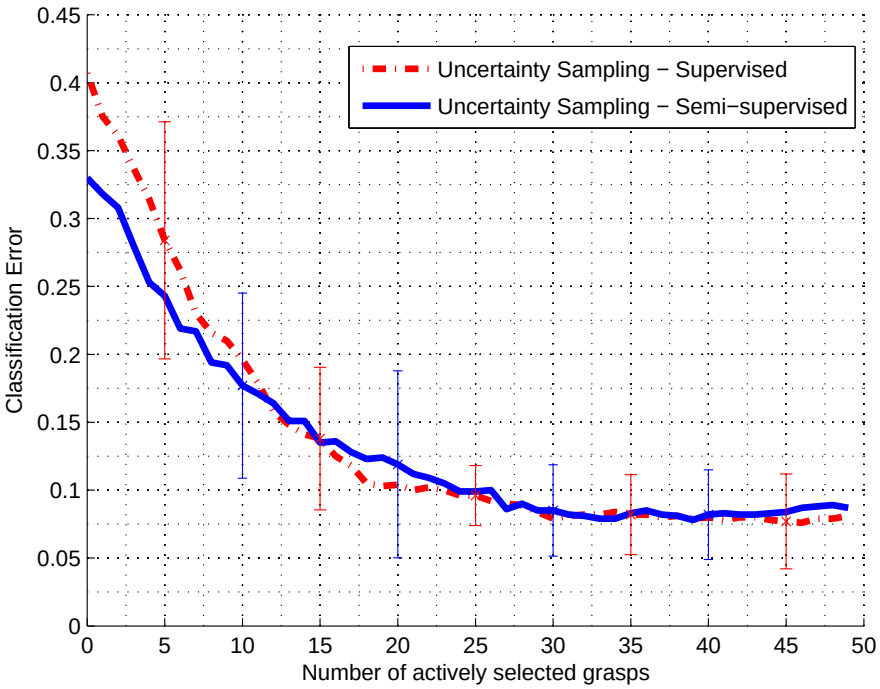
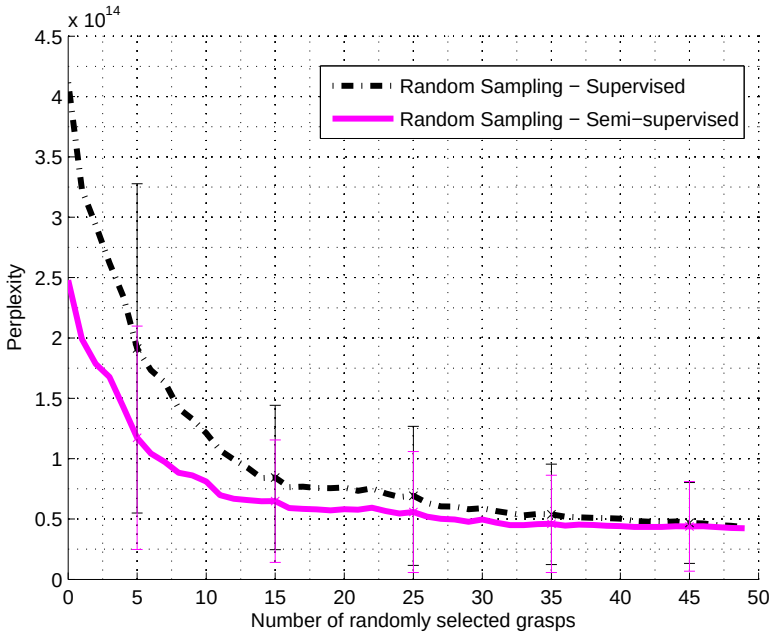
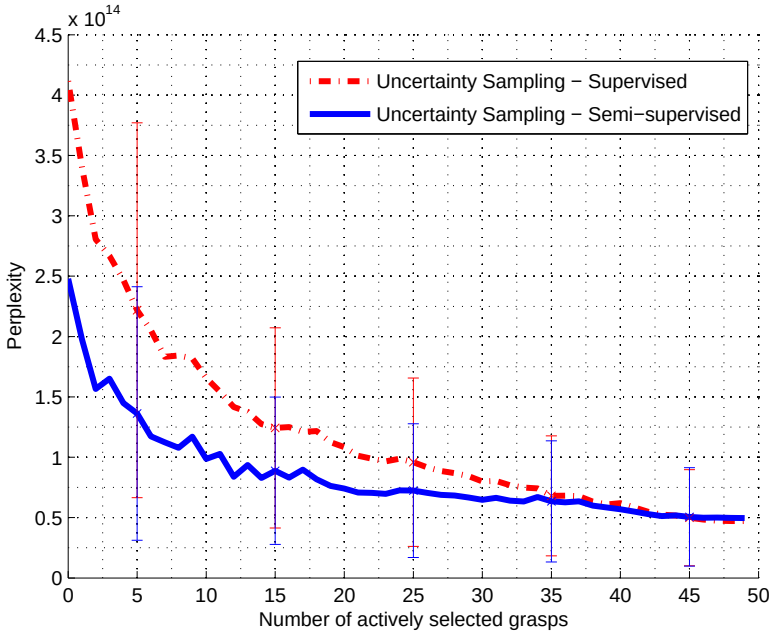


Figure 7.4: Supervised and semi-supervised classification error on the validation sets as actively selected samples are queried via uncertainty sampling. The error bars indicate one standard deviation over 20 realizations. The initial 10 labeled samples are randomly selected. Later, at each iteration the unlabeled sample with the highest class conditional entropy is queried from the active learning pool and inserted to the training set. The models are retrained with this augmented set. With active learning a 10% error is reached with 17 labeled samples in total whereas with random sampling 40 samples are needed to reach the same performance. The semi-supervised curve corresponds to the **hybrid of semi-supervised and active learning** approaches. SSKLR uses an unlabeled training set of size 4000. The neighborhood size for the similarity based augmentation, κ, is set to 30.



(a) Perplexity in random sampling.



(b) Perplexity in active sampling.

Figure 7.5: Perplexity versus the number of iterations shown for the random sampling in (a) and active sampling in (b). Semi-supervised learning reduces perplexity significantly in both settings. Error bars indicate one standard deviation of perplexity over 20 data splits.

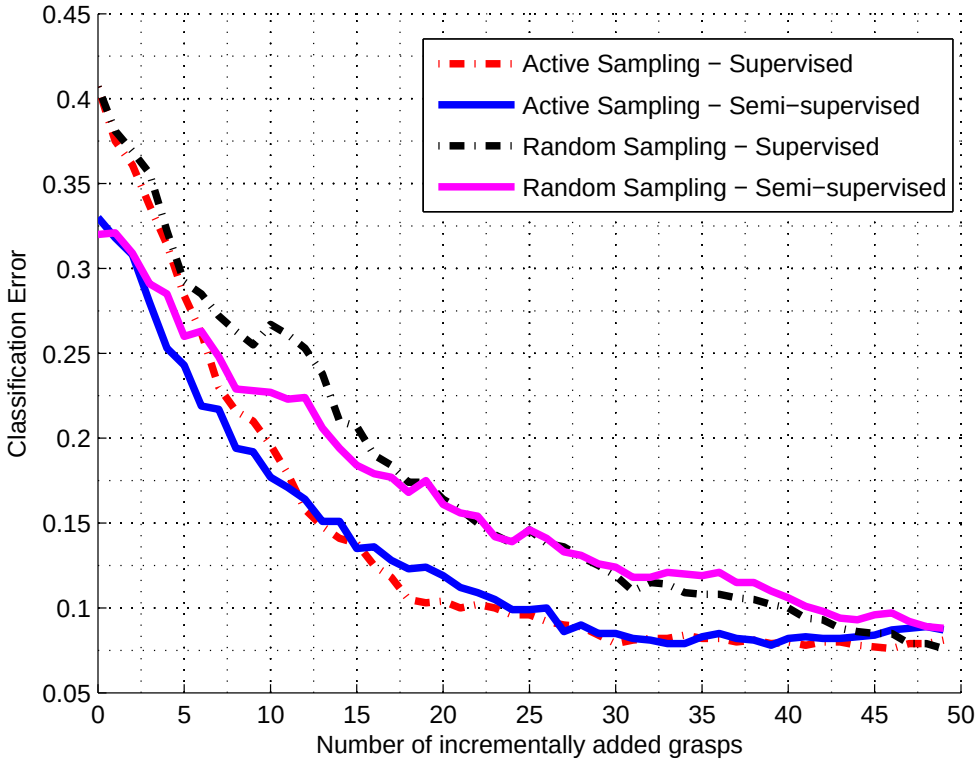


Figure 7.6: Classification error rate for KLR, SSKLR, active-KLR and active SSKLR.

7.6.5 On-Policy Evaluation

In order to test our approach on the robot, we have used a second object, the watering can shown in Figure 7.7(b). For the experiments we have collected a total of 20 labeled instances of 10 successful and 10 unsuccessful configurations. Figure 7.8(b) illustrates these initial training set of data samples where green indicates feasible grasps whereas red indicates infeasible ones. Next, we trained the system incrementally with 15 more samples twice, first with random (RS) and then with actively sampled (AS) data. After we stopped training we have identified 10 test configurations on which the AS and RS algorithms disagree the most. When we carried out these configurations on the robot, in 10 out of 10 configurations the decision of the AS was correct and RS failed validating that the AS is stronger in the decision boundaries.

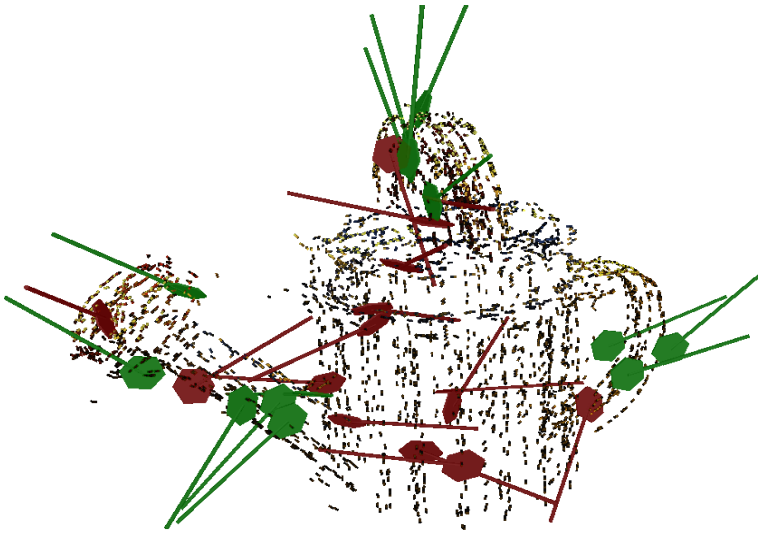
7.7 Conclusion

We have presented a probabilistic approach to model the success likelihoods of grasp configurations from a pool of hypothetical configurations extracted from ECV descriptors. The main bottleneck in the learning process is the scarcity of labeled data due to time-consumption of annotating grasps. Therefore, we have used semi-supervised and active learning approaches in the context of robot grasping. We have experimentally evaluated these approaches in two settings, in the former the data is provided only once as a batch whereas in the latter the agent has the means to query new labeled samples incrementally. We provided the results for three-finger Barrett hand and simple objects. Experimental evaluation demonstrates that combining semi-supervised and active learning approaches is

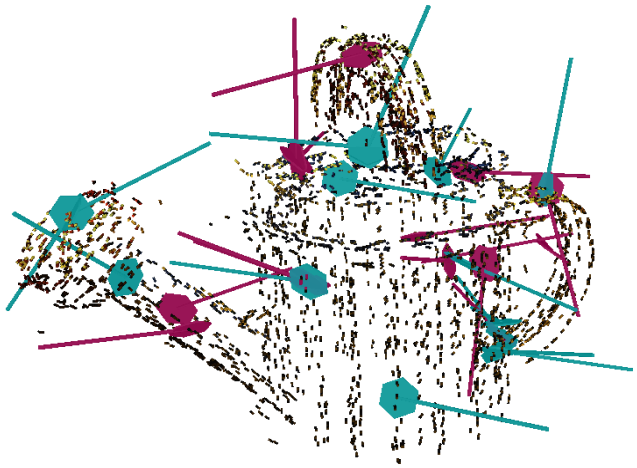
(a) Watering can

(b) Another feasible grasp

Figure 7.7: The watering can used for the on-policy evaluation is shown. There are various potential stable grasp points as demonstrated in (a) and (b).



(a) Initial set of training samples



(b) Iteratively selected training points

Figure 7.8: The training grasp configurations are demonstrated along with the 3D model of the watering can. We initiate the incremental algorithm with 20 labeled training data shown in (b) with 10 feasible and 10 infeasible grasp configurations illustrated in green and red respectively. (c) Iteratively added training samples; pink indicates randomly sampled, blue indicates actively sampled data.

effective in improving the robot's performance with limited supervision. However, it may not always be possible to incrementally train a system. In such situations, semi-supervised learning is advantageous.

The future direction is to learn visual cues that are shared among various objects so that the grasp affordance models are not object-specific but can be generalized to many object categories.

Chapter 8

Conclusion

In this thesis, we have presented two novel approaches to integrate unlabeled data within the generalized entropy framework. In the similarity constrained SSL approach, one incorporates unlabeled samples in the learning process through modifications to the potential functions. We demonstrated two such modifications through pairwise and expectation penalties on the MaxEnt objective. These penalties restrict the entropy maximization problem using the similarity relationships between data samples reflecting our prior knowledge. In the augmentation approach, one exploits unlabeled data to improve the estimates on the model and empirical expectations of the features. We have carried out an extensive empirical analysis of both approaches using standard benchmark data sets from the SSL literature as well as real-life problems from robotics and remote sensing. We have also analyzed the approach on sequential data deriving semi-supervised variants of the CRF algorithm.

Our approach offers a number of advantages over previous methods.

1. Using various entropy measures, we obtain a family of semi-supervised algo-

rithms.

2. These algorithms can be kernelized allowing the model to exploit unlabeled data in a nonlinear manner as opposed to other information theoretic semi-supervised learning methods such as (Grandvalet & Bengio, 2005; Mann & McCallum, 2007).

3. The resulting objective functions are convex since the unlabeled data is incorporated in the primal MaxEnt problem and the objective functions are then derived using convex duality techniques.

4. Another key advantage is that our method is inherently multi-class. This is often not the case for discriminative semi-supervised classifiers, e.g., Transductive Support Vector Machines (TSVMs), as in multi-class settings they require further elaboration in inference such as the one-vs-rest error assessment scheme.

5. The approach proposed in this thesis yields scalable SSL algorithms. For instance, in Chapter 4, we report experimental results on the MNIST data set with up to a total of 70,000 samples. This number can be improved by defining sparser similarity features.

6. In the case of the similarity constrained SSL introduced in Chapter 4, our motivation is reminiscent of the SSL methods in the literature that are based on the smoothness criterion. However, we treat similarities as features and associate them with unique optimization parameters, enabling the algorithm to choose which similarities are salient. This is not the case for many SSL algorithms that treat the similarities uniformly such as the Laplacian SVM.

This is a significant advantage of encoding the similarities in the primal problem via features, as opposed to encoding them within a regularization term in the dual.

8.1 Future Directions

In this thesis, we have derived and analyzed the semi-supervised formulations for three specific cases of the generalized MaxEnt framework. This approach can be extended to other instances mentioned in Section 2.2.1.

There is a lot of potential in terms of the applications of semi-supervised (kernel) CRFs for structured prediction problems in robotics, computer vision and natural language processing. One future direction that is the application of the semi-supervised loss functions to CRFs with general graphical models. This requires the integration of variational approximation methods to our current framework. Large scale variants of these algorithms are also a potential direction of research. Hierarchical methods proposed in (Cai & Hofmann, 2004) and (Seeger, 2008) are also instances of structured output prediction. Their convex duals are explained in (Altun, 2008) in detail. Therefore, investigating the semi-supervised extensions of hierarchical classification methods is also a promising direction.

Another potential direction of research is the definition and evaluation of similarity metrics for semi-supervised learning. Designing similarity metrics is particularly challenging for structured prediction problems. As the similarity metrics often represent our assumptions about the geometry on the data, the choice of the metric is crucial. Answering questions such as the following are promising research directions for SSL for structured problems: How accurate are the approximations

we can get if we assume the similarity features decompose over (combinations of) the cliques of a graphical model? With such a compromise can we still effectively perform semi-supervised structured prediction and if so, when? What kind of (geometry) assumptions do semantic similarity metrics impose, e.g., WordNet distance for natural language processing applications?

Combining semi-supervised learning and active learning effectively is a fundamental AI problem on the way towards self-sufficient autonomous systems that supervise their own learning.

Appendices

Appendix A

Notation and Terminology

Table A.1: Notation and Terminology

D	$\{(x,y)_{i=1\cdots l},(x)_{i=l+1\cdots n}\}$ where $y \in \{\sigma_1,\ldots,\sigma_k\}$
$\mathcal{X}$	Input space (set of observations)
$\mathcal{Y}$	Output space
$\mathcal{Y}(\mathbf{x})$	Output space for observation $\mathbf{x}$
$\mathcal{C}(\mathbf{x})$	The set of cliques of observation $\mathbf{x}$
L	Set of labeled samples, $\{(x_i,y_i)\}_{i=1}^{l}$
U	Set of unlabeled samples $\{x_i\}_{i=l+1}^{n}$
l	Number of labeled samples, $\|L\|$
u	Number of unlabeled samples, $\|U\|$
n	Number of all samples, $n = l + u$
$p(y\|x)$	Predicted (model) conditional distribution
$\tilde{\pi}(x)$	Empirical marginal distribution of the observations

$\tilde{\pi}(x, y)$	Empirical joint distribution of the observations and the labels over (only) labeled data, $\tilde{\pi}(x, y) = \frac{1}{l} \sum_{x_l \in L} \delta(x_l, x)\delta(y_l, y)$
$\psi(x, y)$	Real valued feature vector defined on x and y
$\tilde{\psi}$	Empirical expectation of the model features over labeled data, $\tilde{\psi} = \sum_{x_l \in L} \tilde{\pi}(x_l, y_l)\psi(x_l, y_l)$
$\mathrm{s}(x_i, x_j)$	Similarity between samples x_i and x_j
$\mathbb{R}$	Risk
$\mathbb{D}$	Divergence
$\mathbb{H}$	(Conditional) Entropy
$\mathbb{L}$	Log-likelihood
$\mathbb{E}$	Expectation
$\mathbb{Q}$	Dual objective
$\lambda, \phi, \gamma, \mu$	Dual variables
$K(x_i, x_j)$	A Mercer kernel
$K(x_i, y_i, x_j, y_j)$	Joint kernel representation $K(x_i, y_i, x_j, y_j) = \delta(y_i, y_j)K(x_i, x_j)$
$\mathcal{P}$	Set of (class) conditional probability distributions defined on $\mathcal{X} \times \mathcal{Y}$, $\mathcal{P} = \{p \mid p(y\|x) \geq 0, \sum_{y \in \mathcal{Y}} p(y\|x) = 1, \forall x \in \mathcal{X}, y \in \mathcal{Y}\}$
$\mathcal{L}$	Lagrange function
$\mathbb{I}_C(x)$	Indicator function of a convex set C, $\mathbb{I}_C(x) = 0$ on dom $\mathbb{I}_C = C$, $\mathbb{I}_C(x) = \infty$ otherwise
$\delta(a, b)$	Kronecker-δ, $\delta(a, b) = 0$ if $a \neq b$ and $\delta(a, b) = 1$ if $a = b$
$\langle a, b \rangle$	Dot product, $\sum_i a_i b_i$

Appendix B

Data Sets

B.1 Small Data Sets

These data sets are previously used by (Chapelle & Zien, 2005; Collobert et al., 2006; Karlen et al., 2008). A summary table of the performance values of previous methods are taken from Karlen et al.'s work (Karlen et al., 2008).

g50c

This is an artificial data set originally created by Chapelle and Zien (Chapelle & Zien, 2005). The data is generated such that it comes from two standard normal multi-variate (50 dimensional) Gaussians. The means are chosen such that the Bayes error is 5%. Therefore, the **cluster assumption holds** for this data set perfectly. We use 10 splits of 50 labeled and 500 unlabeled samples.

text

This data set consists of the **mac** and **mswindows** classes of the **News-**

	C	$\|U\|$	$\|L\|$	Dimensions	Splits
g50c	2	500	50	50	10
Digit1	2	1400	100	241	12
COIL	6	1400	100	241	12
USPS_2	2	1400	100	241	12
USPS_{10}	10	1957	50	256	10
text	2	1896	50	7511	10
MNIST	10	5000	100/250	784	10

Table B.1: Properties of multiclass data sets. See (Chapelle et al., 2006; Chapelle & Zien, 2005) for more details. C stands for the number of classes.

group20 data set. The data is sparse with 7511 features. There are 10 splits of 50 labeled and 1896 unlabeled samples.

USPS

This is the well known USPS data for hand digit recognition with 10 classes and 256 dimensions. The data consists of 10 splits of 50 labeled and 1957 unlabeled samples.

B.2 Benchmark Data Sets

These are the data sets used in the SSL benchmark experiments (Chapelle et al., 2006) which are available online.[1] The datasets are standardized so that they all have 241 data features and 100 labeled, 1400 unlabeled samples.

g241c

This is an artificial data set generated so that the cluster assumption holds. 750 points sampled from each of two Gaussians with unit variance such that

[1] http://www.kyb.tuebingen.mpg.de/ssl-book/benchmarks.html

$\|\mu_1 - \mu_2\| = 2.5$ where all dimensions are shifted and rescaled to zero mean and unit variance. The classes are the Gaussians themselves.

g241d

This is again an artificial data set drawn from two Gaussian distributions, however this time the data distribution is manipulated such that the **cluster assumption does not hold** and is in fact misleading. Details and a two dimensional projection can be found in (Chapelle et al., 2006). Manifold assumption does not hold as well.

Digit1

Artificially generated images of the digit 1 so that the data lies in a five dimensional manifold which correspond to different tilt angles of the digit. Details can be found in (Chapelle et al., 2006). The classification problem is to identify the upright digit vs the rest.

USPS

This data set is also derived from the well known USPS data set with two classes such that digits 2 and 5 constitute the first class and the rest of the 10 digits from the original data set form the second class.

COIL

This data comes from The Columbia object image library (COIL-100) so that the 24 original classes are grouped to form six classes of four objects each. For further details the reader may refer to (Chapelle et al., 2006).

The number of labeled and unlabeled samples, data dimensions, classes and data splits can also be found in (Table B.1). Note that while the labeled and unlabeled

	Salinas	**Naples** (optical)	**Naples** (SAR intensity)	**KSC**
Spatial resolution [m]	3.7	30	100	18
Spectral channels	224	7	3	176

Table B.2: Spatial (in meters) and spectral resolution (number of considered channels).

samples for a particular split are entirely distinct, there are overlaps between different splits for both small data sets and benchmark data sets.

B.3 Remote Sensing Data Sets

We considered different kinds of remotely sensed images in the experiments:

Salinas

The Salinas AVIRIS data set, collected over Salinas Valley, California. A total of 16 crop classes were labeled. However, we selected the 8 most representative classes ('Broccoli', 'Celery', 'Corn', 'Fallow', 'Lettuce', 'Soil', 'Stubble', and 'Vinyard') in the image to conduct the experiments. This is a high-resolution scene with pixels of 3.7 meters and the spectral similarity among the classes is also very high. This hyperspectral image is 217×512 and contains 224 spectral channels.

Naples99

Images from ERS2 synthetic aperture radar (SAR) and Landsat Thematic Mapper (TM) sensors were acquired in 1999 over Naples (Italy). The available features were the seven TM bands, two SAR backscattering intensities (0–35 days), and the SAR interferometric coherence. Since these features

come from different sensors, the first step was to perform a specific processing and conditioning of optical and SAR data, and co-registered them (Gómez-Chova et al., 2006). After pre-processing, all features were stacked at a pixel level.

KSC

The image was acquired by the AVIRIS instrument over the Kennedy Space Center (KSC), Florida, on March 23rd, 1996. A total of 224 spectral bands of 10 nm width with center wavelengths from 400-2500 nm is acquired. The image was acquired from an altitude of 20 km and has a spatial resolution of 18 m. After removing low SNR bands and water absorption, a total of 176 bands remains for analysis. A total of 13 classes of interest were labeled representing the various land cover types of the environment. Classes were highly unbalanced, and different marsh subclasses were labeled which makes it a difficult classification problem.

Appendix C

Optimization Software

C.1 Gradient-Based Optimization

To train our classifier we are using the Toolkit for Advanced Optimization(TAO) software (Benson et al., 2007) which is designed for large-scale optimization problems. For the l_2^2 regularized loss function we have used limited memory variable metric (LMVM) algorithm (also known as L-BFGS). The algorithm requires a loss function value and a gradient vector.

Bibliography

Abney, S., Schapire, R. E., & Singer, Y. (1999). Boosting applied to tagging and PP attachment. *In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora* (pp. 38–45).

Altun, Y. (2005). *Discriminative methods for label sequence learning.* Doctoral dissertation, Brown University.

Altun, Y. (2008). *Regularization schemes for structured prediction using convex analysis* (Technical Report). Max Planck Institute.

Altun, Y., McAllester, D., & Belkin, M. (2006). Maximum margin semi-supervised learning for structured variables. *Advances in Neural Information Processing Systems (NIPS)* (pp. 33–40). Cambridge, MA: MIT Press.

Altun, Y., & Smola, A. J. (2006). Unifying divergence minimization and statistical inference via convex duality. *Proceedings of the Annual Conference on Computational Learning Theory (COLT)* (pp. 139–153).

Bach, F. R., Lanckriet, G. R. G., & Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the SMO algorithm. *Proceedings of the International Conference on Machine Learning (ICML).*

Bakir, G., Weston, J., & Schölkopf, B. (2003). Learning to find pre-images. *Advances in Neural Information Processing Systems (NIPS)*. MIT Press.

Balcan, M.-F. (2008). *New theoretical frameworks for machine learning*. Doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA, USA.

Bartlett, P. L., Jordan, M. I., & Mcauliffe, J. D. (2003). *Convexity, classification, and risk bounds* (Technical Report). Journal of the American Statistical Association.

Bellare, K., Druck, G., & McCallum, A. (2009). Alternating projections for learning with expectation constraints. *Conference on Uncertainty in Artificial Intelligence*.

Bengio, Y., Delalleau, O., & Le Roux, N. (2006). Label propagation and quadratic criterion. *Semi-Supervised Learning* (pp. 193–216). MIT Press.

Bennett, K. P., & Demiriz, A. (1998). Semi-supervised Support Vector Machines. *NIPS 12*. Cambridge, MA, USA.

Benson, S., McInnes, L. C., Moré, J., Munson, T., & Sarich, J. (2007). *TAO user manual (revision 1.9)* (Technical Report ANL/MCS-TM-242). Mathematics and Computer Science Division, Argonne National Laboratory. http://www.mcs.anl.gov/tao.

Berger, A. L., Pietra, V. J. D., & Pietra, S. A. D. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, *22*, 39–71.

Bicchi, A., & Kumar, V. (2000). Robotic grasping and contact: A review. *International Conference on Intelligent Robots and Systems (IROS)*.

Bishop, C. M. (2006). *Pattern recognition and machine learning (Information Science and Statistics)*. Springer.

Bishop, C. M., & Lasserre, J. (2007). Generative or discriminative? Getting the best of both worlds. *In Bayesian Statistics 8*. Oxford University Press.

Borwein, J. M., & Lewis, A. S. (2006). *Convex analysis and nonlinear optimization theory and examples*. Springer.

Borwein, J. M., & Zhu, Q. (2005). *Techniques of variational analysis*. Springer.

Cai, L., & Hofmann, T. (2004). Hierarchical document categorization with Support Vector Machines. *CIKM '04: Proceedings of the thirteenth ACM International Conference on Information and Knowledge Management* (pp. 78–87). New York, NY, USA: ACM.

Chang, M.-W., Ratinov, L., & Roth, D. (2007). Guiding semi-supervision with constraint-driven learning. *Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 280–287). Prague, Czech Republic: Association for Computational Linguistics.

Chapelle, O., Schölkopf, B., & Zien, A. (Eds.). (2006). *Semi-supervised learning*. Cambridge, MA: MIT Press.

Chapelle, O., Weston, J., & Schölkopf, B. (2003). Cluster kernels for semi-supervised learning. *Advances in Neural Information Processing Systems (NIPS)* (pp. 585–592). MIT Press.

Chapelle, O., & Zien, A. (2005). Semi–supervised classification by low density

separation. *Proceedings of the International Workshop on Artificial Intelligence and Statistics* (pp. 57–64).

Chen, S. F., & Rosenfeld, R. (2000). A survey of smoothing techniques for ME models. *Speech and Audio Processing, IEEE Transactions on*, *8*, 37–50.

Chen, Y., Garcia, E. K., Gupta, M. R., Rahimi, A., & Cazzanti, L. (2009). Similarity-based classification: Concepts and algorithms. *Journal of Machine Learning Research*, *10*, 747–776.

Collins, M., Schapire, R. E., & Singer, Y. (2000). Logistic regression, Adaboost and Bregman distances. *Proceedings of the Annual Conference on Computational Learning Theory (COLT)* (pp. 158–169).

Collobert, R., Sinz, F., Weston, J., & Bottou, L. (2006). Large scale transductive SVMs. *Journal of Machine Learning Research*, *7*, 1687–1712.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction to algorithms*. MIT Press.

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory 2nd edition*. Wiley Series in Telecommunications and Signal Processing. Wiley-Interscience.

de Granville, C., Southerland, J., & Fagg, A. H. (2006). Learning grasp affordances through human demonstration. *International Conference on Development and Learning (ICDL)*.

Detry, R., Baseski, E., Popovic, M., Touati, Y., Kruger, N., Kroemer, O., Peters, J., & Piater, J. (2009a). Learning object-specific grasp affordance densities. *International Conference on Development and Learning (ICDL)*, *0*, 1–7.

Detry, R., Pugeault, N., & Piater, J. H. (2009b). A probabilistic framework for 3D visual object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*, 1790–1803.

Dudík, M. (2007). *Maximum entropy density estimation and modeling geographic distributions of species*. Doctoral dissertation, Princeton University, Princeton, NJ, USA.

Dudík, M., Phillips, S. J., & Schapire, R. E. (2004). Performance guarantees for regularized maximum entropy density estimation. *Proceedings of the Annual Conference on Computational Learning Theory (COLT)* (pp. 472–486). Springer Verlag.

Dudík, M., & Schapire, R. E. (2006). Maximum entropy distribution estimation with generalized regularization. *Proceedings of the Annual Conference on Computational Learning Theory (COLT)* (pp. 123–138).

Fellbaum, C. (Ed.). (1998). *WordNet an electronic lexical database*. Cambridge, MA ; London: The MIT Press.

Fergus, R., Weiss, Y., & Torralba, A. (2009). Semi-supervised learning in gigantic image collections. *Advances in Neural Information Processing Systems (NIPS)* (pp. 522–530).

Friedlander, M. P., & Gupta, M. R. (2006). On minimizing distortion and relative entropy. *IEEE Transactions on Information Theory*, *52*, 238–245.

Gómez-Chova, L., Fernández-Prieto, D., Calpe, J., Soria, E., Vila-Francés, J., & Camps-Valls, G. (2006). Urban monitoring using multitemporal SAR and multispectral data. *Patt. Rec. Lett.*, *27*, 234–243.

Goodman, J. (2004). Exponential priors for maximum entropy models. *In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004* (pp. 305–312).

Graca, J. V., Ganchev, K., & Taskar, B. (2007). Expectation maximization and posterior constraints. *Advances in Neural Information Processing Systems (NIPS)*.

Grandvalet, Y., & Bengio, Y. (2005). Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems (NIPS)* (pp. 529–536). Cambridge, MA: MIT Press.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical Review*, *106*, 620–630.

Jiao, F., Wang, S., Lee, C.-H., Greiner, R., & Schuurmans, D. (2006). Semi-supervised Conditional Random Fields for improved sequence segmentation and labeling. *Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 209–216). Morristown, NJ, USA: Association for Computational Linguistics.

Joachims, T. (1999). *Making large-scale Support Vector Machine learning practical*, 169–184. MIT Press.

Jurafsky, D., & Martin, J. (2000). *Speech and language processing an introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall.

Karlen, M., Weston, J., Erkan, A., & Collobert, R. (2008). Large scale manifold transduction. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 448–455).

Kazama, J., & Tsujii, J. (2005). Maximum entropy models with inequality constraints: A case study on text categorization. *Machine Learning, 60*, 159–194.

Krüger, N., Lappe, M., & Wörgötter, F. (2004). Biologically motivated multimodal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence the Simulation of Behavious, AISB Journal, 1(5)*, 417–427.

Kuffner, J. J. (2004). Effective sampling and distance metrics for 3D rigid body path planning. *In IEEE International Conference on Robotics and Automation* (pp. 3993–3998).

Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 282–289). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Lafferty, J., Zhu, X., & Liu, Y. (2004). Kernel Conditional Random Fields: representation and clique selection. *Proceedings of the International Conference on Machine Learning (ICML)*. New York, NY, USA: ACM.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE* (pp. 2278–2324).

Lewis, D. D., & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. *Proceedings of the International Conference on Machine Learning*

(ICML) (pp. 148–156). New Brunswick, US: Morgan Kaufmann Publishers, San Francisco, US.

Liang, P., Jordan, M. I., & Klein, D. (2009). Learning from measurements in exponential families. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 641–648). New York, NY, USA: ACM.

Lillesand, T. M., Kiefer, R. W., & Chipman, J. W. (2004). *Remote sensing and image interpretation*. New York: John Wiley. 5th edition.

Mann, G. S., & McCallum, A. (2007). Simple, robust, scalable semi-supervised learning via expectation regularization. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 593–600). Corvalis, Oregon.

Mann, G. S., & McCallum, A. (2008). Generalized expectation criteria for semi-supervised learning of Conditional Random Fields. *Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 870–878). Columbus, Ohio: Association for Computational Linguistics.

Mason, M. T., & Salisbury, J. K. (1985). *Manipulator grasping and pushing operations*. MIT Press.

Montesano, L., & Lopes, M. (2009). Learning object-specific grasp affordance densities. *International Conference on Development and Learning (ICDL)*.

Morales, A., Chinellato, E., Fagg, A., & del Pobil, A. (2004). An active learning approach for assessing robot grasp reliability. *International Conference on Intelligent Robots and Systems (IROS)* (pp. 485–491).

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems (NIPS)* (pp. 849–856). MIT Press.

Pietra, S. D., Pietra, V. D., & Lafferty, J. (1997). Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*, 380–393.

Pugeault, N. (2008). *Early cognitive vision: Feedback mechanisms for the disambiguation of early visual representation*. Verlag Dr. Muller, ISBN 978-3-639-09357-5.

Quadrianto, N., Petterson, J., & Smola, A. J. (2009). Distribution matching for transduction. advances in neural information processing systems. *Advances in Neural Information Processing Systems (NIPS)*.

Rabiner, L. R. (1989). A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*, 257–286.

Rifkin, R. M., & Lippert, R. A. (2007). Value regularization and Fenchel duality. *Journal of Machine Learning Research*, *8*, 441–479.

Rockafellar, R. T. (1996). *Convex analysis*. Princeton University Press.

Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modeling. *Computer, Speech and Language*, *10*, 187–228.

Salganicoff, M., Ungar, L. H., & Bajcsy, R. (1996). Active learning for vision-based robot grasping. *Machine Learning*, *23*, 251–278.

Saxena, A., Driemeyer, J., & Ng, A. Y. (2008a). Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, *27*, 157–173.

Saxena, A., Wong, L., & Ng, A. Y. (2008b). Learning grasp strategies with partial shape information. *AAAI*.

Schölkopf, B., & Smola, A. J. (2001). *Learning with kernels: Support Vector Machines, regularization, optimization, and beyond*. The MIT Press.

Seeger, M. (2001). *Learning with labeled and unlabeled data* (Technical Report). University of Edinburgh.

Seeger, M. W. (2008). Cross-validation optimization for large scale structured classification kernel methods. *Journal of Machine Learning Research*, *9*, 1147–1178.

Settles, B. (2009). *Active learning literature survey* (Technical Report 1648). University of Wisconsin-Madison.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005). Beyond the point cloud: from transductive to semi-supervised learning. *ICML* (pp. 824–831). New York, USA: ACM Press.

Sinha, K., & Belkin, M. (2009). Semi-supervised learning using sparse eigenfunction bases. *Advances in Neural Information Processing Systems (NIPS)*.

Sonnenburg, S., Rätsch, G., Schölkopf, B., & Rätsch, G. (2006). Large scale multiple kernel learning. *Journal of Machine Learning Research*, *7*.

Sutton, C., & McCallum, A. (2007). An introduction to Conditional Random

Fields for relational learning. In L. Getoor and B. Taskar (Eds.), *Introduction to statistical relational learning*. MIT Press.

Szummer, M., & Jaakkola, T. (2002a). Information regularization with partially labeled data. *Advances in Neural Information Processing Systems (NIPS)* (pp. 1025–1032). MIT Press.

Szummer, M., & Jaakkola, T. (2002b). Partially labeled classification with Markov random walks. *Advances in Neural Information Processing Systems (NIPS)* (pp. 945–952). MIT Press.

Torralba, A., Murphy, K., & Freeman, W. T. (2005). Contextual models for object detection using Boosted Random Fields. *Advances in Neural Information Processing Systems (NIPS)*. MIT Press.

Tsang, I. W., & Kwok, J. T. (2006). Large-scale sparsified manifold regularization. *NIPS* (pp. 1401–1408).

Tsochantaridis, I., Joachims, T., Hofmann, T., & Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *The Journal of Machine Learning Research*, *6*, 1453–1484.

Tür, G., Tür, D. H., & Schapire, R. (2005). Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, *45(2)*, 171–186.

Vapnik, V. (1998). *Statistical learning theory*. John Wiley & Sons.

Zhu, X. (2005). *Semi-supervised learning with graphs*. Doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA, USA.

Zhu, X. (2007). *Semi-supervised learning literature survey* (Technical Report). Carnegie Mellon University.

Zhu, X., & Ghahramani, Z. (2002). *Learning from labeled and unlabeled data with label propagation* (Technical Report). Carnegie Mellon University.

Zhu, X., Lafferty, J., & Ghahramani, Z. (2003). Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions. *ICML 2003 Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining* (pp. 58–65).