How Hard is Inference for Structured Prediction?

Amir Globerson GAMIR@CS.HUJI.AC.IL
Tim Roughgarden TIM@CS.STANFORD.EDU
David Sontag DSONTAG@CS.NYU.EDU
Cafer Yildirim CAFERTYILDIRIM@GMAIL.COM

Abstract

Structured prediction tasks in machine learning involve the simultaneous prediction of multiple labels. This is often done by maximizing a score function on the space of labels, which decomposes as a sum of pairwise elements, each depending on two specific labels. The goal of this paper is to develop a theoretical explanation of the empirical effectiveness of heuristic inference algorithms for solving such structured prediction problems. We study the minimum-achievable expected Hamming error in such problems, highlighting the case of 2D grid graphs, which are common in machine vision applications. Our main theorems provide tight upper and lower bounds on this error, as well as a polynomial-time algorithm that achieves the bound.

1. Introduction

In recent years, an increasing number of problems in machine learning are being solved using structured prediction (Collins, 2002; Lafferty et al., 2001; Taskar et al., 2003). Examples of structured prediction include dependency parsing for natural language processing, part-of-speech tagging, named entity recognition, and protein folding. In this setting, the input X is some observation (e.g., an image, a sentence) and the output is a labeling Y, such as an assignment of each pixel in the image to foreground or background, or the parse tree for the sentence. The advantage of performing structured prediction is that one can use local features to infer global structure. For example, one could include a feature that encourages two neighboring pixels to be assigned to different segments (e.g., one to foreground and one to background) whenever there is a large difference in their colors. The feature vector can then be used within an exponential family distribution over the space of labels, conditioned on the input. The parameters are learned using maximum likelihood estimation, as with conditional random fields (Lafferty et al., 2001), or using structured SVMs (Altun et al., 2003; Taskar et al., 2003).

Both marginal and MAP inference in many of these model families are well known to be NP-hard. Despite this, the inference task seems to be much easier than the theoretical worst case. In particular, approximate inference algorithms can be extremely effective, often obtaining state-of-the-art results for these structured prediction tasks. Examples of heuristic MAP inference algorithms that work well in practice include those based on linear programming relaxations and dual decomposition (Koo et al., 2010; Sontag et al., 2008), policy-based search (Daumé et al., 2009), graph cuts (Kolmogorov & Rother, 2007), and branch-and-bound (Sun et al., 2012).

Real-world instances presumably have structure not possessed by worst-case instances, making the corresponding inference tasks relatively tractable. What would a theoretical explanation of this hypothesis look like? The first step in tackling such a problem is to decide on a performance measure for inference. In all of the applications above, performance is naturally quantified as the discrepancy between the correct "ground truth" labels Y and the predicted labels $\widehat{Y}$. The most common performance measure, which we also focus on here, is Hamming error (i.e., the number of disagreements between Y and $\widehat{Y}$).

The current theoretical understanding of the minimum-achievable Hamming error is limited. What makes the problem interesting and challenging is that it involves both a *statistical* and a *computational* perspective. The statistical question is whether there exists *any* algorithm that can predict the true labels with high accuracy, when ignoring computational constraints. Of course, in practice we cannot afford to wait arbitrarily long for each prediction, which motivates the need to understand the computational and statistical trade-offs for structured prediction. This is an increasingly important question for machine learning, discussed in a different context in Chandrasekaran & Jordan (2013) and other recent papers (see Section 3).

The goal of our paper is to initiate the theoretical study of

Proceedings of the 32^{nd} *International Conference on Machine Learning*, Lille, France, 2015. JMLR: W&CP volume 37. Copyright 2015 by the author(s).

structured prediction in terms of the possibility of obtaining small Hamming error. Such an analysis must define a generative process for the X, Y pairs, in order to properly define expected Hamming error. We consider a simplified model in which the observed X is a noisy version of local observations of a binary labeling Y. In particular, X_i is a noisy version of the true Y_i and for specific pairs i, j, $X_{i,j}$ is a noisy version of the indicator $\mathcal{I}\left[Y_i = Y_j\right]$. The posterior for Y given X is then very similar to the *data* and *smoothness* terms used for structured prediction in machine vision (Geman & Geman, 1984).

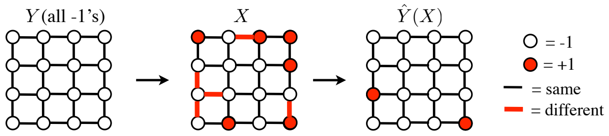
Motivated by these machine vision applications, we highlight the case where the i, j pairs correspond to the edges of a two-dimensional grid graph. The corresponding inference task corresponds to a grid-structured Ising model whose parameters are drawn randomly from this generative model. In addition to its relevance to real-world applications, the grid graph is particularly interesting because it is one of the simplest settings where the statistical and computational questions are non-trivial. In particular, both MAP and marginal inference in planar Ising models with an external field are well-known to be NP-hard and #P-hard in the worst case (Barahona, 1982).

After presenting our generative model, we proceed to study how well Y can be recovered from X. Optimal prediction requires calculating marginals of an Ising model and is intractable for worst case X. We introduce a polynomial time algorithm, and analyze its expected Hamming error. The algorithm is a two step procedure which ignores the node evidence in the first step, solving a MaxCut problem on a grid (which can be done in polynomial time), and at a second step uses node observations to break symmetry. Despite the simplicity of the algorithm, we show that it is in fact optimal (up to constants) for a natural regime of the problem parameters. Finally, our analysis is validated via experimental results on 2D grid graphs.

Taken together, our results provide the first theoretical analysis for structured prediction using approximate inference. In particular, we show that approximate inference can provide optimal results under natural modeling assumptions.

2. Preliminaries

We consider the setting of predicting a set of N labels $Y = Y_1, \ldots, Y_N$, where $Y_i \in \{-1, +1\}$, from a set of observations X. The observations X are assumed to be generated from Y by the following process. The generative process is defined via a graph $G = (V, E)$ and two parameters, an edge noise $p \in [0, .5]$ and a node noise $q \in [0, .5]$. For each edge $(u, v) \in E$, the edge observation X_{uv} is independently sampled to be $Y_u Y_v$ with probability $1 - p$ (called a *good* edge), and $-Y_u Y_v$ with probability p (called a *bad* edge). Similarly, for each node $v \in V$, the node observation X_v is independently sampled to be Y_v with probability $1 - q$ (*good* nodes), and $-Y_v$ with probability q (*bad* nodes). The process is illustrated in Figure 1 (a,b). Thus, the observed X provide noisy information about the labels Y and their pairwise relations.



(a) Ground truth (b) Observed evidence (c) Approximate recovery

Figure 1. Statistical recovery on a grid graph. **(a)** Ground truth, which we want to recover. **(b)** Noisy node and edge observations. **(c)** Approximate recovery (prediction), in this case with average Hamming error 2/16.

A labeling algorithm is a function $\mathcal{A} : \{-1, +1\}^E \times \{-1, +1\}^V \to \{-1, +1\}^V$ from graphs with labeled edges and nodes (the observation X) to a labeling of the nodes V (the unobserved label Y). We measure the performance of $\mathcal{A}$ by the expectation of the Hamming error (i.e., the number of mispredicted labels) over the observation distribution induced by Y. By the *error* of an algorithm, we mean its worst-case (over Y) expected error, where expectation is over the process generating the observations X from Y. Formally, we denote the error of the algorithm given a value $Y = y$ by $e_y(\mathcal{A})$ and define it as:

$$e_y(\mathcal{A}) = \mathbf{E}_{X|Y=y}\left[\tfrac{1}{2}\left\|\mathcal{A}(X) - y\right\|_1\right] \qquad (1)$$

The overall error is then:

$$e(\mathcal{A}) = \max_y e_y(\mathcal{A}). \qquad (2)$$

Note that the definition above does not involve a generative model for Y. However, in the analysis it will be useful to consider a model where Y is generated according to the uniform distribution $p_U(Y)$, resulting in a joint distribution:

$$p(X, Y) = p_U(Y)p(X|Y) \qquad (3)$$

2.1. MAP and Marginal Estimators

Given the above definitions, our goal is to find an algorithm $\mathcal{A}$ with low error $e(\mathcal{A})$. It is easy to show that the optimal strategy in this case is to label node Y_i with $\arg\max_{Y_i} p(Y_i|X)$ where the conditional is calculated from the joint $p(X, Y)$.[1]

[1] The optimality follows from the fact that this strategy is Bayes optimal for the uniform distribution, and furthermore has the same expected Hamming error for all Y. It is therefore minimax optimal (Berger, 1985).

Unfortunately, the above strategy is not tractable in our case since it requires the calculation of marginals of an Ising model on a grid with an external field (Barahona, 1982).

Our labeling algorithm will be loosely based on the maximum likelihood (ML) estimator, which returns the label $\arg\max_Y p(X,Y)$ This is equivalent to solving:

$$\max_{Y} \quad \sum_{uv\in E} \frac{1}{2} X_{uv} Y_u Y_v \log \frac{1-p}{p} + \sum_{v\in V} \frac{1}{2} X_u Y_u \log \frac{1-q}{q}, \tag{4}$$

or simply $\max_Y \sum_{uv\in E} X_{uv} Y_u Y_v + \gamma \sum_{v\in V} X_u Y_u$, where $\gamma = \log\frac{1-q}{q} / \log\frac{1-p}{p}$. We note that the ML estimator is also NP hard to compute for 2D grid graphs (Barahona, 1982).[2] However when $q = 0.5$ there is a polynomial time algorithm for solving it (Hadlock, 1975), and we shall make use of this in what follows.

Approximate Recovery: The interesting regime for structured prediction is when the node noise q is close to 0.5. In this regime there is no correlation decay, and correctly predicting a label requires a more global consideration of the node observations. The intriguing question — and the question that reveals the importance of the structure of the graph G — is whether or not there are algorithms with small error when the edge noise p is a small constant. Precisely, for a family of graphs $\mathcal{G}$, we say that *approximate recovery is possible* if there is a function $f : [0,1] \to [0,1]$ with $\lim_{p\downarrow 0} f(p) = 0$ such that, for every sufficiently small p and all $N \geq N_0(p)$ for some constant $N_0(p)$, the minimum-possible error of an algorithm on a graph $G \in \mathcal{G}$ with N vertices is at most $f(p) \cdot N$.

3. Related Work

Our goal is to recover a set of unobserved variables Y from a set of noisy observations X. As such it is related to various statistical recovery settings, but distinct from those in several important aspects. Below we review some of the related problems.

Channel Coding: This is a classic recovery problem (e.g., see Arora et al., 2009) where the goal is to exactly recover Y (i.e., with zero error). Here Y is augmented with a set of "error-correcting" bits, and the complete set of bits is sent through a noisy channel. In our model, $X_{i,j}$ is a noisy version of the parity of Y_i and Y_j. Thus our setting may be viewed as communication with an error correcting code where each error-correcting bit involves two bits of the original message Y, and each Y_i appears in d_i check bits, where d_i is the number of edge observations involving Y_i. Such codes cannot be used for errorless transmission (e.g., see our lower bound in Section 5). As a result, the techniques and results from channel coding do not appear to apply to our setting.

Correlation Clustering (CC): In the typical setting of CC, Y is a partition of N variables into an unknown number of clusters and $X_{u,v}$ specifies whether Y_u and Y_v are in the same cluster, with some probability of error as in Joachims & Hopcroft (2005) or adversarially as in Mathieu & Schudy (2010). The goal is to find Y from X. Our results apply to the case of two clusters. The most significant difference is that most of the CC works study the objective of minimizing the number of edge disagreements. It is not obvious how to translate the guarantees provided in these works to a non-trivial bound on Hamming error for our analysis framework. Stable instances of CC were studied by Balcan & Braverman (2009), who gave positive results when G is the complete graph and stated the problem of understanding general graphs as an open question.

Recovery Algorithms in Other Settings: The high-level goal of recovering ground truth from a noisy input has been studied in numerous other application domains. In the overwhelming majority of these settings, the focus is on maximizing the probability of exactly recovering the ground truth, a manifestly impossible goal in our setting. This is the case with, for example, planted cliques and graph partitions (e.g. Condon & Karp, 2001; Feige & Kilian, 2001; McSherry, 2001), detecting hidden communities (Anandkumar et al., 2013), and phylogenetic tree reconstruction (Daskalakis et al., 2006). We note that works on community detection also typically use observations on a complete graph (Massoulié, 2013; Mossel et al., 2013), whereas our interest is in observations restricted to a given graph (e.g., planar). Another relevant line of work is Braverman & Mossel (2008), who analyze sorting from noisy information. They give polynomial-time algorithms for the approximate recovery of a ground truth total ordering given noisy pairwise comparisons. Their approach, similar to the present work, is to compute the maximum likelihood ordering given the data, and prove that the expected distance between this ordering and the ground truth ordering is small.

Recovery on Random Graphs: Two very recent works (Abbe et al., 2014; Chen & Goldsmith, 2014) have addressed the case where noisy pairwise observations of Y are obtained for edges in a graph. In both of these, the focus is mainly on guarantees for random graphs (e.g., Erdos Renyi). Furthermore, the analysis is of perfect recovery (in the limit $N \to \infty$) and its relation to the graph ensemble. The goal of our analysis is considerably more challenging, as we are interested in the Hamming error for finite N. Abbe et al. (2014) explicitly state partial (as opposed to exact) recovery for sparse graphs with constant degrees as

[2] There are approximation results for such problems (e.g., Goemans & Williamson, 1995). However the approximation is with respect to the value of the maximized function, and not the Hamming error. It is thus not applicable in our context.

an open problem, which we partially solve in this paper.

Structured prediction: When learning a structured prediction model (i.e., learning the score function $s(X,Y)$ used in inference), a natural question is how train and test errors are related. Several works have provided generalization bounds for this setting (e.g., see Daniely & Shalev-Shwartz, 2014). This analysis is very different from our focus here, since they analyze the variance of the generalization error and not its expected value (known as the "bias"). Other works have considered learning with approximate inference (Finley & Joachims, 2008; Kulesza & Pereira, 2007), but provide no theoretical insights into whether real-world structured prediction tasks might result in low error when using these.

Percolation: Some of the technical ideas in our study of grid graphs in Section 5 are inspired by arguments in percolation, the study of connected clusters in random (often infinite) graphs (e.g., see p. 286 in Grimmett, 1999). We directly adapt results from statistical physics to give precise constants for our theoretical results.

4. Foreground-Background Segmentation

We begin with an empirical study of the structure of real-world probabilistic inference problems. Whereas there has been a wealth of empirical evidence that real-world inference tasks are easy to solve, there have been few investigations into *why* the corresponding inference tasks are easy.

We use the Weizmann horse dataset (Borenstein & Ullman, 2002) and consider the inference problems arising from using a conditional random field (CRF) to perform foreground-background segmentation. The data is shown in Fig. 2, where each image is accompanied with its ground truth segmentation. The parameters of the model were learned by Domke (2013, Sec. 8.3) using 200 training images. The CRF is a pairwise Markov random field with binary variables for each pixel (foreground vs. background), and is grid structured as in Fig. 1. The features used for the pairwise potentials consist of edge filters and various functions of the difference in colors between the pixels.

The MPLP algorithm (Sontag et al., 2008) was used for inference, and provably found the MAP assignment for the cases shown (via an optimality certificate). The Hamming error between the recovered segmentations in (e),(j) and true ones in (b),(g) is low (only 1.1%,1.6% wrong pixels).

Next, we turn to a more quantitative analysis of the inference problems that arise in this model. Denote by Z an input image. Then the learned model results in a set of image dependent weights $\beta_{uv} = f_{uv}(Z;\theta), \beta_u = f_u(Z;\theta)$ where θ are the learned parameters and f is a linear function of features of Z and θ. The posterior of the CRF is then:

$$\Pr(\hat{Y}|Z) \propto \exp(\sum_{uv\in E} \beta_{uv}\hat{Y}_u\hat{Y}_v + \sum_{u\in V}\beta_u\hat{Y}_u) \qquad (5)$$

The above is similar to (4), with β_{uv}, β_u replacing $0.5X_{uv}\log\frac{1-p}{p}, 0.5X_u\log\frac{1-q}{q}$. Of course in the CRF we are not observing a X_{uv}, X_u directly, but the weights β play the same role of providing information on the value of Y (singleton and pairwise).[3]

The above equivalence can be used to infer the p, q noise levels that correspond to a given CRF and image ensemble. To estimate q simply find the fraction of times where $Y_u = \text{sgn}\beta_u$, and similarly for p. This is illustrated in Figure 2. In (a,f) we consider two of the images from the test set.

Figure (c,h) shows pixels for which $\text{sgn}\,(\beta_u Y_u) = -1$, separating into cases where $Y_i = -1$ (red) and $Y_i = 1$ (blue). Similarly Figure (d,i) shows pixels for which $\text{sgn}\,(\beta_{uv}Y_uY_v) = -1$ (ignoring cases where $|\beta|$ is below a threshold), shown in red for $Y_uY_v = 1$ and blue otherwise. These can be used to calculate the corresponding p, q which turn out to be $p = 0.03$ and $q = 0.2$.

The example above demonstrates several principles which motivate our analysis: first, inference problems on grid graphs that arise from practical statistical recovery seem to be solvable in practice and with low Hamming error. Second, the node noise turns out to be considerably larger than the edge noise. Indeed, we will show that recovery for low edge noise is indeed possible using an even simpler algorithm than MPLP.

5. Inference in Grid Graphs

This section studies grid graphs. We devote a lengthy treatment to them for several reasons. First, grid graphs are central in applications such as machine vision (see Section 4). Second, the grid is a relatively poor expander (Hoory et al., 2006) and for this reason poses a number of interesting technical challenges. Third, our algorithm for the grid and other planar graphs is computationally efficient. Finally, our grid analysis yields matching upper and lower bounds of $\Theta(p^2N)$ on the information-theoretically optimal error.

5.1. Upper Bound

We study the algorithm $\bar{A}$ given in Algorithm 1, which has two stages. The first stage ignores the node observations and computes a labeling $\widehat{Y}$ that maximizes the agreement with respect to edge observations only, i.e.

[3] In the CRF, β_{uv}, β_u are edge and node dependent. In our model (4) the p and q are constants. This is just to simplify the analysis, and can be changed.

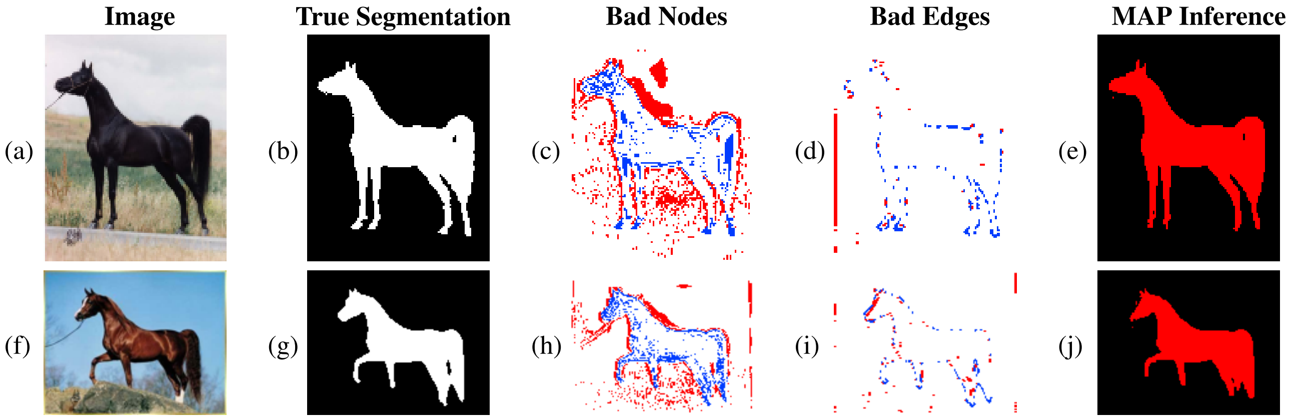


Figure 2. Weizmann horse dataset. See discussion in Section 4. Note that the MAP inference is found using the MPLP algorithm (Sontag et al., 2008), and is the exact MAP assignment in this case (via optimality certificates). It is different from the true segmentation, as the CRF model does make some errors. See text for explanation of bad edges and nodes.

Algorithm 1 $\bar{A}(X)$ for inference in grids.

```
input Edge and node observations $X$
  $\widehat{Y} \leftarrow \arg\max_Y \sum_{uv\in E} X_{uv}Y_uY_v$
  if $\sum_{v\in V} X_v\widehat{Y}_v < 0$ then
    $\widehat{Y} \leftarrow -\widehat{Y}$
  end if
output $\widehat{Y}$
```

$$\widehat{Y} \leftarrow \arg\max_{Y} \sum_{uv\in E} X_{uv}Y_uY_v. \tag{6}$$

Note that $\widehat{Y}$ and $-\widehat{Y}$ agree with precisely the same set of edge observations, and thus both maximize Eq. 6. The second stage of algorithm $\bar{A}$ outputs $\widehat{Y}$ or $-\widehat{Y}$, according to a "majority vote" by the node observations. Namely, it outputs $-\widehat{Y}$ if $\sum_{v\in V} X_v\widehat{Y}_v < 0$, and $\widehat{Y}$ otherwise.

When the graph G is a 2D grid, or more generally a planar graph, this algorithm can be implemented in polynomial time by a reduction to the maximum-weight matching problem (see Fisher, 1966; Hadlock, 1975; Barahona, 1982). We shall prove the following theorem, which shows that approximate recovery on grids is possible.[4]

Theorem 5.1 *$\bar{A}$ achieves an error $e(\bar{A}) = O(p^2N)$.*

Analysis of First Stage: We first show that after the first stage, the expected error of the better of $\widehat{Y}, -\widehat{Y}$ is $O(p^2N)$. We then extend this error bound to the output of the second stage of the algorithm.

We begin by highlighting a simple but key lemma characterizing a structural property of the maximizing assignment of Eq. 6, i.e. the MAP assignment without node observations. Recall from Section 2 that an edge is good if $X_{uv} = Y_uY_v$, and bad otherwise. The intuition is that if the maximizing assignment is wrong on some connected region of the grid, there must be many bad edge observations on the boundary of this region. Since the probability p of a bad edge observation is assumed to be small, this tells us that it is highly unlikely that there can be a large region of the graph for which the maximizing assignment disagrees with the ground truth.

We use $\delta(S)$ to denote the boundary of $S \subseteq V$, i.e. the set of edges with exactly one endpoint in S.

Lemma 5.2 (Flipping Lemma) *Let S denote a maximal connected subgraph of G with every node of S incorrectly labelled by $\hat{Y}$. Then at least half the edges of $\delta(S)$ are bad.*[5]

Proof: First, note that the output label $\widehat{Y}$ satisfies $X_{uv}\hat{Y}_u\hat{Y}_v = 1$ (or equivalently $X_{uv} = \hat{Y}_u\hat{Y}_v$) on at least half the edges of $\delta(S)$. Otherwise, flipping $\widehat{Y}$ of all nodes in S would strictly increase the objective in (6), contradictig the optimality of $\hat{Y}$. On the other hand, since S is maximal, for every edge $e \in \delta(S)$ exactly one endpoint of e is correctly labeled. Namely, for each such $e = uv$ we have: $Y_uY_v \neq \hat{Y}_u\hat{Y}_v$. The above two statements are only compatible if at least half the edges of $\delta(S)$ are bad. ■

Call a set S *bad* if at least half its boundary $\delta(S)$ is bad. The Flipping Lemma motivates bounding the probability that a given set is bad, and then bounding the Hamming error by enumerating over sets S. This approach can be made to work only if the collection of sets S is chosen carefully — otherwise, there are far too many sets and this approach fails to yield a non-trivial error bound.

[4] Big-O notation describes the behavior as p goes to zero.

[5] Result holds for $\pm\hat{Y}$, since only relies on $\hat{Y}$ maximizing (6).

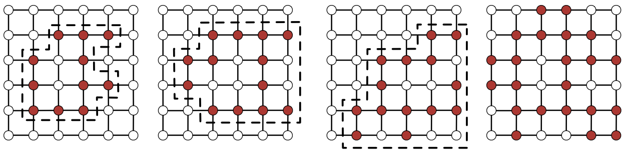


Figure 3. Examples of type 1, 2, 3, and 6 regions, left-to-right.

For any subset S of V, denote by $G[S]$ the subgraph of G induced by S. Thus, $G[S]$ is the subset of edges in G whose endpoints are in S. Let $\mathcal{C}$ denote the subsets $S \subseteq V$ such that the induced subgraph $G[S]$ is connected. We classify subsets S of $\mathcal{C}$ into 6 categories depending on whether S contains (see Figure 3):

1. No vertices on the perimeter of G
2. Vertices from exactly one side of the perimeter of G
3. Vertices from exactly two sides of the perimeter of G, and these two sides are adjacent
4. Vertices from exactly two sides of the perimeter of G, and these two sides are opposite
5. Vertices from exactly three sides of the perimeter of G
6. Vertices from all four sides of the perimeter of G.

Let $\mathcal{C}_{<6}$ denote the set of all $S \subset V$ from one of the first 5 categories. For a set $S \in \mathcal{C}_{<6}$, we define a corresponding *filled in* set $F(S)$. An illustration of the filling in procedure is given in the supplementary. Consider the connected components $C_1, \dots, C_k$ of $G[V \setminus S]$ for such a subset S. Call such a connected component *3-sided* if it includes vertices from at least three sides of the grid G. We define $F(S)$ as the union of S with all the connected components of $G[V \setminus S]$ except for a single 3-sided one. Observe that $F(S) \supseteq S$, and $F(S)$ is not defined for type 6 components S. Let $\mathcal{F} = \{F(S) : S \in \mathcal{C}_{<6}\}$ denote the set of all such filled-in components.

Let H denote the Hamming error of our algorithm on a random input. We introduce a simpler-to-analyze upper bound on H. In particular, define a new random variable T by

$$T = \sum_{F \in \mathcal{F}} |F| \cdot 1_{F \text{ is bad}} \tag{7}$$

The next two Lemmas will imply that $H \leq T$ with probability 1. First, we show that enumerating over filled in regions can only be an overestimate of the error.

Lemma 5.3 *If S_1, S_2 are disjoint and not type 6, then $F(S_1), F(S_2)$ are distinct and not type 6.*

Proof: Since $F(S)$ excludes a 3-sided component, it cannot be type 6. Also, for a set S that is not type 6, $\delta(F(S))$ is a non-empty subset of $\delta(S)$. Thus, the non-empty set of endpoints of $\delta(F(S))$ that lie in $F(S)$ also lie in S. This implies that if $F(S_1) = F(S_2)$, then $S_1 \cap S_2 \neq \emptyset$. ■

Next we show for each mislabeled region, its filled in region is bad. This applies to whichever of $\widehat{Y}, -\widehat{Y}$ does not have a type-6 set of mislabeled vertices (there is at most one type-6 set in the connected components of mislabeled vertices, so at least one of $\widehat{Y}, -\widehat{Y}$ has this property). Let B denote the mislabeled vertices of such a labeling, and let $B_1, \dots, B_k$ denote the connected components (of types 1–5) of $G[B]$.

Lemma 5.4 *For every B_i, the filled-in set $F(B_i)$ is bad.*

The proof of Lemma 5.4 is an extension of the Flipping Lemma; see supplementary material for the details.

We now upper bound the easier-to-analyze quantity T. The proofs for the next three Lemmas can be found in the supplementary. The first is straightforward to prove, and it provides an upper bound on the probability that a set S is bad, as a function of its boundary size $|\delta(S)|$.

Lemma 5.5 *For every set S with $|\delta(S)| = i$, it holds that* $\mathbf{Pr}[S \textit{ is bad}] \leq (3\sqrt{p})^i$.

Our probability bound is naturally parameterized by the number of boundary edges. Because of this, we face two tasks in upper bounding T. First, T counts the number of *nodes* of bad filled-in sets $F \in \mathcal{F}$, not boundary sizes. The next lemma states that the number of nodes of such a set cannot be more than the square of its boundary size.

Lemma 5.6 *For all $F \in \mathcal{F}$: (1) $|F| \leq |\delta(F)|^2$; (2) if F is a type-1 region, then $|F| \leq \frac{1}{16}|\delta(F)|^2$.*

The second task in upper bounding T is to count the number of filled-in sets $F \in \mathcal{F}$ that have a given boundary size. We do this by counting simple cycles in the dual graph.

Lemma 5.7 *Let i be a positive integer. (a) If i is odd or 2, then there are no type-1 sets $F \in \mathcal{F}$ with $|\delta(F)| = i$; (b) If i is even and at least 4, then there are at most $\frac{N \cdot 4 \cdot 3^{i-2}}{2i} = N \cdot \frac{2 \cdot 3^{i-2}}{i}$ type 1 sets $F \in \mathcal{F}$ with $|\delta(F)| = i$; (c) If i is at least 2, then there are at most $2\sqrt{N} \cdot 3^{i-2}$ type 2–5 sets $F \in \mathcal{F}$ with $|\delta(F)| = i$.*

A computation (see supplementary) now shows that

$$\mathbf{E}[T] \leq cp^2 N + O(p\sqrt{N}) \tag{8}$$

for a constant $c > 0$ that is independent of p and N. The intuition for why this computation works out is that Lemma 5.7 implies that there is only an exponential number of relevant regions to sum over; Lemma 5.6 implies that the Hamming error is quadratically related to the (bad) boundary size; and Lemma 5.5 implies that the probability of a bad boundary is decreasing exponentially in i (with base $3\sqrt{p}$). Provided p is at most a sufficiently small constant (independent of N), the probability term dominates and so the expected error is small.

In the supplementary, we show that the constant c in (8) is 8 for $p \leq 0.017$. To derive the constant, we use results from statistical physics on the connectivity constant of square lattices (Clisby & Jensen, 2012; Madras & Slade, 1993), and explicit computations of the number of self-avoiding polygons (which correspond to our filled in regions) of a particular boundary length and area (Jensen, 2000).

Analysis of Second Stage: Our analysis so far shows that the better of $\widehat{Y}, -\widehat{Y}$ has small error with respect to the ground truth Y. In the second phase, we use the node labels to choose between them via a "majority vote." Straightforward Chernoff bounds imply that, provided q is slightly below $\frac{1}{2}$, the better of $\widehat{Y}, -\widehat{Y}$ is chosen in the second stage with high probability. This implies that the approximate recovery bound of the original algorithm without node labels carries over to the two-phase algorithm with node labels, which proves Theorem 5.1. The formal proof of the second stage appears in the supplementary.

5.2. Lower Bound

In this section, we prove that every algorithm suffers worst-case expected error $\Omega(p^2 N)$ on 2D grid graphs, matching the upper bound for the 2-step algorithm in Theorem 5.1.

Theorem 5.8 *Every algorithm $\mathcal{A}$ must have error $e(\mathcal{A})$ which is $\Omega(p^2 N)$.*

We use the fact that marginal inference is minimax optimal for Eq. 2 (see Section 2.1). The expected error of marginal inference is independent of the ground truth (by symmetry), so we can lower bound its expected error for the case $Y_i = 1$ for all i. Also, its error only decreases if it is given part of the ground truth.

Consider an alternating black and white coloring of the nodes of the grid, like a chess board. Suppose we give the inference algorithm the true labels of all the black nodes. Call a white node *ambiguous* if exactly two of the four incident edges are labeled "+1" in X. A white node is ambiguous with probability $6p^2(1-p)^2 \geq 5p^2$ for $p \leq 0.078$.

For an ambiguous node, marginal inference would predict the label corresponding to the node observation, which is wrong with probability q. Hence, the expected (over the input) error of marginal inference is at least $\frac{N}{2} \cdot 5p^2 \cdot q$, which proves our result. The full proof is in the supplementary.

5.3. Empirical Study

Our theoretical analysis suggests that statistical recovery on 2D grid graphs can attain an error that scales with p^2. Furthermore, we showed that this error is achieved using the two-step algorithm in Section 5. Here we describe a synthetic experiment that compares the two-step algorithm to other recovery procedures. We consider a 20×20 grid, with high node noise of 0.4 and variable edge noise levels. In addition to the two-step algorithm we consider the following:[6]

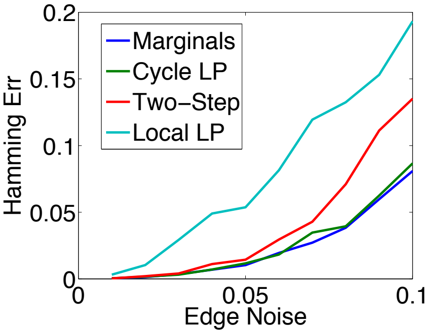


Figure 4. Average Hamming error for different recovery algorithms. Data is generated from a 20×20 grid with node noise $q = 0.4$ and variable edge noise p. The true Y is the assignment of all -1. Results are averaged over 100 repetitions.

- *Marginal inference* – predicting according to $p(Y_i|X)$. As mentioned in Section 2 this is the optimal procedure. Although it is generally hard to calculate, for the graph size we use it can be done in 20 minutes per model using the junction tree algorithm.
- *Local LP relaxation* – Instead of calculating $p(Y_i|X)$ one can resort to approximation. One possibility is to calculate the mode of $p(Y|X)$ (also known as the MAP problem). However, since this is also hard, we consider LP relaxations of the MAP problem. The simplest such relaxation assumes locally consistent psuedo-marginals.
- *Cycle LP relaxation* – A tighter version of the LPs uses cycle constraints instead of pairwise. In fact, for planar graphs with no external field (as in the first step of our algorithm) this relaxation is tight. It is thus of interest to study in our context. For both the cycle and local relaxations we use the code from Sontag et al. (2012).

Fig. 4 shows the expected error for the different algorithms, as a function of edge noise. It can be seen that the two step procedure almost matches the accuracy of the optimal marginal algorithm for low noise levels. As the noise increases the gap grows. Another interesting observation is that the local relaxation performs significantly worse than the other baselines, but the cycle relaxation is close to optimal. The latter observation is likely to be due to the fact that with high node noise and low edge noise, the MAP problem is "close" to the no node-noise case, where the cycle relaxation is exact. However, an analysis of the Hamming

[6] We also tried hill climbing, but results were poor.

error in this case remains an open problem. Finally, Fig. 4 is in agreement with our theoretical results, which predict that for low edge noise the two step procedure is optimal.

6. Extensions

Other Planar Graphs: Our main theorem in Sec. 5 uses properties of grids beyond planarity, but is robust in that it applies to all planar graphs that share two key features with grids. The first property, which fails in "thin" planar graphs like a path but holds in many planar graphs of interest, is the following weak expansion property: *(P1: Weak expansion.)* For some constants $c_1, c_2 > 0$, every filled-in set $F \in \mathcal{F}$ satisfies $|F| \leq c_1 |\delta(F)|^{c_2}$. (Filled-in sets can be defined analogously to the grid case.) The second key property is: *(P2: Bounded Dual Degree.)* Every face of G, except possibly for the outer face, has a constant number of boundary edges. Our proof of Theorem 5.1 shows that every family of planar graphs meeting (P1) and (P2) admits computationally efficient approximate recovery.

Expander Graphs: Another widely studied class of graphs is the family of d-regular expanders (Hoory et al., 2006). Recall that a d-regular graph with N nodes is an expander with constant $c > 0$ if, for every set $S \subseteq V$ with $|S| \leq N/2$, $|\delta(S)| \geq c \cdot d \cdot |S|$. We next show that the family $\mathcal{G}$ of d-regular expanders with constant c allows approximate recovery with $f(p) = 3p/c$.

The algorithm is the same as in Sec. 5 (it is not computationally efficient for expanders). As in Sec. 5, analyzing the two-stage algorithm reduces to analyzing the better of the two solutions produced by the first stage. We therefore assume that the output $\hat{Y}$ of the first stage has error H at most $N/2$.

Fix a noise parameter $p \in (0, \frac{1}{2})$, a graph $G \in \mathcal{G}$ with N sufficiently large, and a ground truth. Chernoff bounds imply that for all sufficiently large N, the probability that $|B| \geq 2p|E| = pdN$ is at most $1/N^2$. When $|B| > pdN$, we can trivially bound the error H by $N/2$. When $|B| \leq pdN$, we bound H from above as follows.

Let S denote the nodes of V correctly classified by the first stage $\widehat{Y}$ and $C_1, \ldots, C_k$ the connected components of the (misclassified) nodes of the induced subgraph $G[V \setminus S]$. Since $H \leq N/2$, $|C_i| \leq N/2$ for every i. Using $H = \sum_{i=1}^{k} |C_i|$, we have

$$H \leq \frac{1}{cd} \sum_{i=1}^{k} |\delta(C_i)| \leq \frac{2}{cd} \sum_{i=1}^{k} |\delta(C_i) \cap B| \leq \frac{2}{cd} |B|, \quad (9)$$

where the first inequality follows from the expansion condition, the second from Lemma 5.2, and the third from the fact that the $\delta(C_i)$'s are disjoint (since the C_i's are maximal). Thus, when $|B| \leq pdN$, $H \leq \frac{2p}{c} N$. Overall, we have $\mathbf{E}[H] \leq 3pN/c$ for N sufficiently large, as claimed.

7. Discussion

Structured prediction underlies many empirically successful systems in machine vision and NLP. In most of these (e.g., see Koo et al., 2010; Kappes et al., 2013) the inference problems are intractable, and approximate inference is used instead. However, there is little theoretical understanding of when structured prediction is expected to perform well, how its performance is related to the *structure* of the score function, which approximation algorithms are expected to work in which setting, etc.

Here we present a first step in this direction, analyzing the error of structured prediction for 2D grid models. One key finding is that a two-step algorithm attains the information theoretically optimal error in a natural regime of parameters. What makes this setting particularly interesting from a theoretical perspective is that exact inference (marginals and MAP) is intractable due to the intractability of planar models with external fields. It is thus surprising and encouraging that a tractable algorithm achieves optimal error.

Our positive results easily extend to many variations of our generative model. The reason is that our proofs only use the fact that the probability that a boundary $\delta(S)$ consists of at least half bad edges decays exponentially in the boundary size $|\delta(S)|$ (Lemma 5.5). As such, our results apply to a much broader class of generative models, which may explain our observations for foreground-background segmentation. We can also obtain an analysis that is closer to CRF models, by allowing variable edge and noise probabilities.

Our work opens the door for a number of new research directions, with both theoretical and practical implications. For instance, for grid models, our two step procedure uses both node and edge evidence, but it is clear that for small q, improved procedures are possible. In particular, the experiments in Section 5.3 show that decoding with cycle LP relaxations results in empirical performance that is close to optimal, even for large p. More generally, an exciting direction is to understand the statistical and computational properties of structured prediction for complex tasks such as dependency parsing (Koo et al., 2010) and non-binary variables (as in semantic segmentation). In these cases, it would be interesting to understand how the structure of the score function affects both the optimal expected accuracy and which algorithms can achieve it.

Acknowledgments

AG was supported by ISF Centers of Excellence grant 1789/11, TR by NSF grant CCF-1215965, and DS by ONR #N00014-13-1-0646 and NSF CAREER award #1350965.

References

Abbe, Emmanuel, Bandeira, Afonso S., Bracher, Annina, and Singer, Amit. Decoding binary node labels from censored edge measurements: Phase transition and efficient recovery. *CoRR*, abs/1404.4749, 2014.

Altun, Y., Tsochantaridis, I., and Hofmann, T. Hidden Markov support vector machines. In *ICML*, 2003.

Anandkumar, Anima, Ge, Rong, Hsu, Daniel, and Kakade, Sham M. A tensor spectral approach to learning mixed membership community models. In *COLT*, 2013.

Arora, Sanjeev, Daskalakis, Constantinos, and Steurer, David. Message passing algorithms and improved lp decoding. In *STOC*, pp. 3–12, 2009.

Balcan, Maria-Florina and Braverman, Mark. Finding low error clusterings. In *The 22nd Conference on Learning Theory*, 2009.

Barahona, Francisco. On the computational complexity of Ising spin glass models. *J. Phys. A*, 15(10):3241, 1982.

Berger, James O. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 1985.

Borenstein, Eran and Ullman, Shimon. Class-specific, top-down segmentation. In *Proceedings of the 7th European Conference on Computer Vision-Part II*, ECCV '02, pp. 109–124, London, UK, UK, 2002. Springer-Verlag. ISBN 3-540-43744-4.

Braverman, Mark and Mossel, Elchanan. Noisy sorting without resampling. In *SODA*, pp. 268–276, 2008.

Chandrasekaran, Venkat and Jordan, Michael I. Computational and statistical tradeoffs via convex relaxation. *Proceedings of the National Academy of Sciences*, 110(13): E1181–E1190, 2013. doi: 10.1073/pnas.1302293110.

Chen, Yuxin and Goldsmith, Andrea J. Information recovery from pairwise measurements. *CoRR*, abs/1404.7105, 2014.

Clisby, Nathan and Jensen, Iwan. A new transfer-matrix algorithm for exact enumerations: self-avoiding polygons on the square lattice. *J. Phys. A*, 45(11):115202, 15, 2012. ISSN 1751-8113.

Collins, M. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *EMNLP*, 2002.

Condon, Anne and Karp, Richard M. Algorithms for graph partitioning on the planted partition model. *Random Structures and Algorithms*, 18(2):116–140, 2001.

Daniely, Amit and Shalev-Shwartz, Shai. Optimal learners for multiclass problems. In *Proceedings of The 27th Conference on Learning Theory*, pp. 287–316, 2014.

Daskalakis, Constantinos, Mossel, Elchanan, and Roch, Sébastien. Optimal phylogenetic reconstruction. In *STOC*, pp. 159–168, 2006.

Daumé, Iii, Hal, Langford, John, and Marcu, Daniel. Search-based structured prediction. *Mach. Learn.*, 75 (3):297–325, June 2009. ISSN 0885-6125.

Domke, J. Learning graphical model parameters with approximate marginal inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10): 2454–2467, 2013. ISSN 0162-8828.

Feige, Uriel and Kilian, Joe. Heuristics for semirandom graph problems. *Journal of Computer and System Sciences*, 63(4):639–671, 2001.

Finley, T. and Joachims, T. Training structural SVMs when exact inference is intractable. In *International Conference on Machine Learning (ICML)*, pp. 304–311, 2008.

Fisher, Michael E. On the dimer solution of planar Ising models. *J. of Mathematical Phys.*, 7:1776, 1966.

Geman, S. and Geman, D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 1984.

Goemans, Michel X and Williamson, David P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

Grimmett, Geoffrey. *Percolation*. Springer, 1999.

Hadlock, F. Finding a maximum cut of a planar graph in polynomial time. *SIAM Journal on Computing*, 4(3): 221–225, 1975.

Hoory, Shlomo, Linial, Nathan, and Wigderson, Avi. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

Jensen, Iwan. Size and area of square lattice polygons. *J. Phys. A*, 33(18):3533–3543, 2000. ISSN 0305-4470.

Joachims, Thorsten and Hopcroft, John E. Error bounds for correlation clustering. In *Proceedings of the Twenty-Second International on Machine Learning (ICML)*, pp. 385–392, 2005.

Kappes, J.H., Andres, B., Hamprecht, F.A., Schnorr, C., Nowozin, S., Batra, D., Kim, Sungwoong, Kausler, B.X., Lellmann, J., Komodakis, N., and Rother, C. A comparative study of modern inference techniques for discrete energy minimization problems. In *CVPR*, pp. 1328–1335, June 2013.

Kolmogorov, Vladimir and Rother, Carsten. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1274–1279, July 2007. ISSN 0162-8828.

Koo, Terry, Rush, Alexander M, Collins, Michael, Jaakkola, Tommi, and Sontag, David. Dual decomposition for parsing with non-projective head automata. In *EMNLP*, pp. 1288–1298, 2010.

Kulesza, Alex and Pereira, Fernando. Structured learning with approximate inference. In *Advances in neural information processing systems*, pp. 785–792, 2007.

Lafferty, J., McCallum, A., and Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pp. 282–289, 2001.

Madras, Neal and Slade, Gordon. *The self-avoiding walk*. Probability and its Applications. Birkhäuser Boston Inc., Boston, MA, 1993. ISBN 0-8176-3589-0.

Massoulié, Laurent. Community detection thresholds and the weak ramanujan property. *CoRR*, abs/1311.3085, 2013. URL http://arxiv.org/abs/1311.3085.

Mathieu, Claire and Schudy, Warren. Correlation clustering with noisy input. In *SODA*, pp. 712–728, 2010.

McSherry, Frank. Spectral partitioning of random graphs. In *FOCS*, pp. 529–537, 2001.

Mossel, Elchanan, Neeman, Joe, and Sly, Allan. A proof of the block model threshold conjecture. *CoRR*, abs/1311.4115, 2013. URL http://arxiv.org/abs/1311.4115.

Sontag, David, Meltzer, Talya, Globerson, Amir, Weiss, Yair, and Jaakkola, Tommi. Tightening LP relaxations for MAP using message-passing. In *UAI*, pp. 503–510, 2008.

Sontag, David, Choe, Do Kook, and Li, Yitao. Efficiently searching for frustrated cycles in MAP inference. In *UAI*, pp. 795–804, 2012.

Sun, Min, Telaprolu, Murali, Lee, Honglak, and Savarese, Silvio. An efficient branch-and-bound algorithm for optimal human pose estimation. In *CVPR*, 2012.

Taskar, B., Guestrin, C., and Koller, D. Max-margin Markov networks. In *NIPS*, 2003.