Budgeted Online Collective Inference

Jay Pujara
University of Maryland
jay@cs.umd.edu

Ben London
University of Maryland
blondon@cs.umd.edu

Lise Getoor
University of California, Santa Cruz
getoor@soe.ucsc.edu

Abstract

Updating inference in response to new evidence is a fundamental challenge in artificial intelligence. Many real problems require large probabilistic graphical models, containing millions of interdependent variables. For such large models, jointly updating the most likely (i.e., MAP) configuration of the variables each time new evidence is encountered can be infeasible, even if inference is tractable. In this paper, we introduce *budgeted online collective inference*, in which the MAP configuration of a graphical model is updated efficiently by revising the assignments to a subset of the variables while holding others fixed. The goal is to selectively update certain variables without sacrificing quality with respect to full inference. To formalize the consequences of partially updating inference, we introduce the concept of *inference regret*. We derive inference regret bounds for a class of graphical models with strongly-convex free energies. These theoretical insights, combined with a thorough analysis of the optimization solver, motivate new approximate methods for efficiently updating the variable assignments under a budget constraint. In experiments, we demonstrate that our algorithms can reduce inference time by 65% with accuracy comparable to full inference.

1 INTRODUCTION

A key challenge of many artificial intelligence problems is that the evidence grows and changes over time, requiring updates to inferences. Every time a user rates a new movie on Netflix, posts a status update on Twitter, or adds a connection on LinkedIn, inferences about preferences, events, or relationships must be updated. When constructing a knowledge base, each newly acquired document prompts the system to update inferences over related facts and resolve mentions to their canonical entities. Problems such as these benefit from collective (i.e., joint) reasoning, but incorporating new evidence into a collective model is particularly challenging. New evidence can affect multiple predictions, so updating inference typically involves recomputing all predictions in an expensive global optimization. Even when a full inference update is tractable—which, using the best known methods, can be linear in the number of factors—it may still be impractical. For example, updating a knowledge graph with millions of facts can take hours (Pujara *et al.*, 2013), thereby requiring some compromise, either in the form of a deferment strategy or approximate update. In this work, we consider the task of efficiently updating the *maximum-a-posteriori* (MAP) state of a probabilistic graphical model, conditioned on evolving evidence. We refer to this problem as *online collective inference*.

In online collective inference, a single graphical model, describing the conditional distribution of a set of random variables with fixed dependency structure, is given. Over a series of epochs, the true assignments (i.e., labels) of certain variables are revealed, introducing new evidence with which we can update the assignments to the remaining unknowns. We constrain the problem by adding a budget, such that only a fixed percentage of variables can be updated in each epoch, necessitating some approximation to full inference. This constraint distinguishes our work from the vast body of literature on belief revision (e.g., Gardenfors, 1992), Bayesian network updates (e.g., Buntine, 1991; Friedman and Goldszmidt, 1997; Li *et al.*, 2006), models for dynamic (Murphy, 2002) or sequential (Fine *et al.*, 1998) data, and adaptive inference (e.g., Acar *et al.*, 2008), which deal with exact updates to inference. We analyze budgeted online collective inference from both the theoretical and algorithmic perspectives, addressing two fundamental questions: How do we choose which variables to update? How "close" is the approximate inference update to the full inference update?

To formalize the latter question, we introduce the concept of *inference regret*. Informally, inference regret measures the amount of change induced by fixing (i.e., condi-

tioning on) certain variables in the inference optimization. We specifically analyze the inference regret of continuous graphical models whose inference objective is strongly convex. One instantiation of this class of models is hinge-loss Markov random fields (Bach *et al.*, 2013), which have been broadly applied and demonstrate state-of-the-art performance in many applications. Using the duality between strong convexity and stability, we upper-bound the inference regret. Our bound is proportional to the distance from the fixed variables to the optimal values of the full inference problem, scaled by a function of several model-specific properties. We use our inference regret bound to quantify the effect of approximate inference updates in response to new evidence (in this case, revealed labels). The bound highlights two terms affecting the regret: the accuracy of the original predictions and the amount that the original predictions change. This latter insight informs our approximate update methods with a simple intuition: fix the predictions that are unlikely to change in a full inference update.

To efficiently determine which variables are least likely to change, we turn to the optimization algorithm used during inference. The alternating direction method of multipliers (ADMM) (Boyd *et al.*, 2011) is a popular convex optimization technique that offers convergence guarantees while remaining highly scalable. We analyze the optimization process and catalog the features that allow us to determine which variables will change the most. Using these features to generate a score for each variable, we establish a ranking capturing the priority of including the variables in subsequent inference. Since the variable scores are produced using the state of the optimization algorithm, our method does not incur computational overhead. By ranking variables, we approximate full inference with an arbitrary budget and support an anytime online inference algorithm.

We evaluate the theoretical guarantees and approximation quality in experiments on a synthetic collective classification task and a real-world collaborative filtering task. These experiments validate our theoretical bounds by measuring the stability and quality of the MAP state as new evidence is revealed. To connect theoretical guarantees with empirical performance, we compare approximate inference to computing the full MAP state at each epoch of graph evolution. We find that our approach to online inference allows a substantial decrease in computation and running time while maintaining the quality of the inferred values. In our experiments, our methods consistently reduce running time by 65% to 70%, show diminishing inference regret, and, in some cases, have lower test error than full inference.

1.1 RELATED WORK

Updating inference is a longstanding problem in artificial intelligence. The classic problem of belief revision (Gardenfors, 1992) considers revising and updating a set of propositional beliefs using a set of axiomatic guarantees to consistency. Diverse research has considered updating the parameters or structure of Bayesian networks in response to evolving evidence (Buntine, 1991; Friedman and Goldszmidt, 1997; Li *et al.*, 2006, e.g.,). Finally, many models address dynamic or sequential data, such as Dynamic Bayesian Networks (Murphy, 2002) and hierarchical hidden Markov models (Fine *et al.*, 1998). Our work addresses the specific problem of approximating full MAP inference in the online setting when a model is given and provides formal guarantees for the approximation quality.

Making efficient updates to the full inference result is the goal of a related area of research, *adaptive inference*. Adaptive marginal inference (Acar *et al.*, 2008; Sümer *et al.*, 2011) can update the marginal probability of a query in $\mathrm{O}\left(2^{\mathrm{tw}(G)} \log n\right)$-time, where $\mathrm{tw}(G)$ is the tree-width of the graph and n is the number of variables. Adaptive MAP inference (Acar *et al.*, 2009) can update the MAP state in $\mathrm{O}\left(m + m \log(n/m)\right)$-time, where m is the number of variables that change their state. Though the algorithm does not need to know m beforehand, a model change could result in changes to all n variables' states, with cost equivalent to exact inference. These adaptive inference techniques do not currently support partial updates to the MAP state or accommodate budgeted updates.

Approximate adaptive inference was considered by Nath and Domingos (2010), who proposed *expanding frontier belief propagation* (EFBP), a belief propagation algorithm that only updates messages in the vicinity of the updated potentials. They showed that the beliefs generated by EFBP lower- and upper-bound the beliefs of full BP, thereby providing guarantees on the quality of the approximation. This result differs from ours in that it bounds the individual marginal probabilities, whereas we bound the L^1 distance between MAP states. Unlike our approximation algorithm, EFBP does not explicitly limit computation and, in the worst case, may need to update all variables to achieve convergence conditions.

The quantity we call inference regret is conceptually similar to *collective stability* (London *et al.*, 2013a). Collective stability measures the amount of change in the output of a structured predictor induced by local perturbations of the evidence. London *et al.* (2013a, 2014) analyzed the collective stability of marginal inference in discrete graphical models, concluding that (approximate) inference with a strongly convex entropy function enhances stability. Our technical approach is similar, in that we also leverage strong convexity. However, the types of perturbations we consider—fixing target variables—are not covered by their analysis. Stability analysis is closely related to *sensitivity analysis*. Since the terms are used interchangeably in the literature, we distinguish them as follows: sensitivity analysis examines *if* and *when* the solution changes; stability analysis examines *how much* it changes by. Laskey

analyzed the sensitivity of queries (which can be used for marginal inference) in Bayesian networks. Chan and Darwiche studied the sensitivity of queries (2005) and MAP inference (2006) in Markov networks. Their 2005 paper also analyzes the stability of queries.

2 PRELIMINARIES

The theory and methods introduced in this paper apply to any continuous-valued MRF with a strongly convex MAP inference objective function. One case of particular interest is a class of graphical models known as *hinge-loss Markov random fields* (HL-MRFs) (Bach *et al.*, 2013). An HL-MRF is a continuous-valued Markov network in which the potentials are hinge functions of the variables. Our choice of HL-MRFs comes from technical considerations: we reason about the strength of convexity of the inference objective, and *maximum a posteriori* (MAP) inference in HL-MRFs can be strongly convex. However, from a practical standpoint, HL-MRFs have many benefits. MAP inference in HL-MRFs is provably and empirically efficient, in theory growing $O(N^3)$ with the number of potentials, N, but in practice often converging in $O(N)$ time. Models built using HL-MRFs achieve state-of-the-art performance for a variety of applications (Bach *et al.*, 2013; Beltagy *et al.*, 2014; Chen *et al.*, 2014; Fakhraei *et al.*, 2014; London *et al.*, 2013b; Ramesh *et al.*, 2014). Finally, HL-MRFs are easily specified through *probabilistic soft logic* (PSL) (Bach *et al.*, 2015), a probabilistic programming language with a first-order logical syntax.

To better understand HL-MRFs and PSL, consider a model for collective classification of network data, in which the goal is to assign labels to nodes, conditioned on some local evidence and network structure. Let $G \triangleq (\mathcal{V}, \mathcal{E})$ denote an undirected graph on $n \triangleq |\mathcal{V}|$ nodes. Each node $i \in \mathcal{V}$ is associated with a set of local observations, X_i, and an unknown label, L_i. (In some settings, a subset of the labels are revealed.) In general, the observations and labels can be real-valued; but for simplicity of exposition, let us assume that each observation is binary-valued, and each label is categorical. The following logical rules define a PSL program for a typical collective classification model:

$$w_{x,\ell} : \textsc{Feature}(N, x) \Rightarrow \textsc{Label}(N, \ell)$$
$$w_{e,\ell} : \textsc{Edge}(N_1, N_2) \wedge \textsc{Label}(N_1, \ell) \Rightarrow \textsc{Label}(N_2, \ell)$$

Variables N, N_1 and N_2 denote nodes; x indexes a local feature; and ℓ denotes a label. The rules are weighted by $w_{x,\ell}$ and $w_{e,\ell}$ respectively. Given G and $\mathbf{X} \triangleq (X_1, \ldots, X_n)$ (and possibly some subset of the labels), the rules are *grounded out* for all possible instantiations of the predicates. The groundings involving unknown variables—in this case, groundings of the LABEL predicate—are represented by $[0, 1]$-valued variables, $\mathbf{Y} \triangleq (Y_{i,\ell})_{i,\ell}$. Using a relaxation of the MAX-SAT problem to continuous domains (Globerson and Jaakkola, 2007), each grounding is converted to a convex hinge function of the form

$$f(\mathbf{X}, \mathbf{Y}) = \left(\max\{0, \varphi(\mathbf{X}, \mathbf{Y})\}\right)^q,$$

where φ is a linear function of $(\mathbf{X}, \mathbf{Y})$, and $q \in \{1, 2\}$ is an exponent that is set *a priori* for the given rule. Each hinge function becomes a potential in an HL-MRF.

The resulting HL-MRF enables probabilistic inference over the set of PSL rules. Fix a set of r PSL rules, with corresponding weights $\mathbf{w} \triangleq (w_1, \ldots, w_r)$. For the i^{th} rule, let $\mathcal{G}(i)$ denote its set of groundings in G, and let f_j^i denote the j^{th} grounding of its associated hinge function. To compactly express the weighted sum of grounded rules, we let

$$\mathbf{f}(\mathbf{X}, \mathbf{Y}) \triangleq \left[\sum_{j=1}^{|\mathcal{G}(1)|} f_j^1(\mathbf{X}, \mathbf{Y}), \ldots, \sum_{j=1}^{|\mathcal{G}(r)|} f_j^r(\mathbf{X}, \mathbf{Y}) \right]^\top$$

denote the aggregate of the grounded hinge functions. We can thus write the weighted sum of groundings as $\mathbf{w} \cdot \mathbf{f}(\mathbf{X}, \mathbf{Y})$. This inner product defines a distribution over $(\mathbf{Y} \,|\, \mathbf{X})$ with probability density function $p(\mathbf{Y} = \mathbf{y} \,|\, \mathbf{X} = \mathbf{x}; \mathbf{w}) \propto \exp\left(-\mathbf{w} \cdot \mathbf{f}(\mathbf{X}, \mathbf{Y})\right)$. The maximizer of the density function (alternately, the minimizer of $-\mathbf{w} \cdot \mathbf{f}(\mathbf{X}, \mathbf{Y})$) is the MAP state. The values of $\mathbf{Y}$ in the MAP state can be interpreted as confidences. Additionally, we can define a prior distribution over each $\mathbf{Y}$. In this case, we will use an L^2, or Gaussian, prior. This can be accomplished using the rule $w_{p,\ell} : \neg\, \textsc{Label}(N, \ell)$, with a squared hinge (i.e., $q = 2$). Let us assume, without loss of generality, that each prior rule has weight $w_{p,\ell} = w_p/2$, for some $w_p > 0$. Thus, the corresponding hinge function for grounding $\textsc{Label}(i, \ell)$ is simply $(Y_{i,\ell})^2$; the aggregate features for the prior are $\|\mathbf{Y}\|_2^2$. So as to simplify notation, let $\dot{\mathbf{w}} \triangleq (\mathbf{w}, w_p)$ and define an *energy* function,

$$E(\mathbf{y} \,|\, \mathbf{x}; \dot{\mathbf{w}}) \triangleq \mathbf{w} \cdot \mathbf{f}(\mathbf{x}, \mathbf{y}) + \frac{w_p}{2} \|\mathbf{y}\|_2^2 . \qquad (1)$$

The resulting probability density function is

$$p(\mathbf{Y} = \mathbf{y} \,|\, \mathbf{X} = \mathbf{x}; \dot{\mathbf{w}}) \propto \exp\left(-E(\mathbf{y} \,|\, \mathbf{x}; \dot{\mathbf{w}})\right).$$

MAP inference, henceforth denoted $h(\mathbf{x}; \dot{\mathbf{w}})$, is given by

$$h(\mathbf{x}; \dot{\mathbf{w}}) = \arg\min_{\mathbf{y}} E(\mathbf{y} \,|\, \mathbf{x}; \dot{\mathbf{w}}).$$

3 INFERENCE REGRET

The notion of *regret* has often been used to measure the loss incurred by an online learning algorithm relative to the optimal hypothesis. We extend this concept to online *inference*. Fix a model. Suppose we are given evidence, $\mathbf{X} = \mathbf{x}$, from which we make a prediction, $\mathbf{Y} = \mathbf{y}$, using MAP inference. Then, some subset of the unknowns are

revealed. Conditioning on the new evidence, we have two choices: we can recompute the MAP state of the remaining variables, using full inference; or, we can fix some of the previous predictions, and only update a certain subset of the variables. To understand the consequences of fixing our previous predictions we must answer a basic question: how much have the old predictions changed?

We formalize the above question in the following concept.

Definition 1. Fix a budget $m \geq 1$. For some subset $\mathcal{S} \subset \{1, \ldots, n\}$, such that its complement $\overline{\mathcal{S}} \triangleq \{1, \ldots, n\} \setminus \mathcal{S}$, has size $\left|\overline{\mathcal{S}}\right| = m$, let $\mathbf{Y}_{\mathcal{S}}$ denote the corresponding subset of the variables, and let $\mathbf{Y}_{\overline{\mathcal{S}}}$ denote its complement. Assume there is an operator Γ that concatenates $\mathbf{Y}_{\mathcal{S}}$ and $\mathbf{Y}_{\overline{\mathcal{S}}}$ in the correct order. Fix a model, $\dot{\mathbf{w}}$, and an observation, $\mathbf{X} = \mathbf{x}$. Further, fix an assignment, $\mathbf{Y}_{\mathcal{S}} = \mathbf{y}_{\mathcal{S}}$, and let

$$h(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}}) \triangleq \Gamma\left(\mathbf{y}_{\mathcal{S}}, \operatorname*{arg\,min}_{\mathbf{y}_{\overline{\mathcal{S}}}} E\left(\Gamma(\mathbf{y}_{\mathcal{S}}, \mathbf{y}_{\overline{\mathcal{S}}}) \,|\, \mathbf{x}; \dot{\mathbf{w}}\right)\right)$$

denote the new MAP configuration for $\mathbf{Y}_{\overline{\mathcal{S}}}$ after fixing $\mathbf{Y}_{\mathcal{S}}$ to $\mathbf{y}_{\mathcal{S}}$. We define the *inference regret* for $(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}})$ as

$$\mathfrak{R}_n(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}}) \triangleq \frac{1}{n} \left\| h(\mathbf{x}; \dot{\mathbf{w}}) - h(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}}) \right\|_1 . \quad (2)$$

In general, the inference regret can be as high as 1 for variables in [0,1]. For example, consider network classification model in which probability mass is only assigned to configurations where all nodes have the same label. Fixing a variable corresponding to a single node label in this setting is tantamount to fixing the label for all nodes. In the presence of strong evidence for a different label, incorrectly fixing a single variable results in incorrectly inferring all variables.

In online inference, regret can come from two sources. First, there is the regret of not updating the MAP state given new evidence (in this case, revealed labels). If this regret is low, it may not be worthwhile to update inference, which can be useful in situations where updating inference is expensive (such as updating predicted attributes for all users in a social network). The second type of regret is from using an approximate inference update in which only certain variables are updated, while the rest are kept fixed to their previous values. We describe several such approximations in Section 4. In practice, one may have both types of regret, caused by approximate updates in response to new evidence. Note that the inference regret obeys the triangle inequality, so one can upper-bound the compound regret of multiple updates using the regret of each update.

3.1 REGRET BOUNDS FOR STRONGLY CONVEX INFERENCE

A convenient property of the L^2 prior is that it is *strongly convex*, by which we mean the following.

Definition 2. Let $\Omega \subseteq \mathbb{R}^n$ denote a convex set. A differentiable function, $f : \Omega \to \mathbb{R}$, is κ-*strongly convex* (w.r.t. the 2-norm) if, for all $\boldsymbol{\omega}, \boldsymbol{\omega}' \in \Omega$,

$$\frac{\kappa}{2} \left\| \boldsymbol{\omega} - \boldsymbol{\omega}' \right\|_2^2 + \langle \nabla f(\boldsymbol{\omega}), \boldsymbol{\omega}' - \boldsymbol{\omega} \rangle \leq f(\boldsymbol{\omega}') - f(\boldsymbol{\omega}). \quad (3)$$

Strong convexity has a well-known duality with stability, which we will use in our theoretical analysis.

The function $f(\boldsymbol{\omega}) \triangleq \frac{1}{2} \left\| \boldsymbol{\omega} \right\|_2^2$ is 1-strongly convex. Therefore, the prior, $\frac{w_p}{2} \left\| \mathbf{y} \right\|_2^2$, is at least w_p-strongly convex. We also have that the aggregated hinge functions, $\mathbf{f}(\mathbf{x}, \mathbf{y})$, are convex functions of $\mathbf{Y}$. Thus, it is easily verified that the energy, $E(\mathbf{y} \,|\, \mathbf{x}; \dot{\mathbf{w}})$, is at least a w_p-strongly convex function of $\mathbf{y}$. This yields the following upper bound on the inference regret.

Proposition 1. *Fix a model with weights $\dot{\mathbf{w}}$. Assume there exists a constant $B \in [0, \infty)$ such that, for any $\mathbf{x}$, and any $\mathbf{y}, \mathbf{y}'$ that differ at coordinate i,*

$$\left\| \mathbf{f}(\mathbf{x}, \mathbf{y}) - \mathbf{f}(\mathbf{x}, \mathbf{y}') \right\|_2 \leq B \left| y_i - y_i' \right| . \quad (4)$$

Then, for any observations $\mathbf{x}$, any budget $m \geq 1$, any subset $\mathcal{S} \subset \{1, \ldots, n\} : \left|\overline{\mathcal{S}}\right| = m$, and any assignments $\mathbf{y}_{\mathcal{S}}$, with $\hat{\mathbf{y}} \triangleq h(\mathbf{x}; \dot{\mathbf{w}})$, we have that

$$\mathfrak{R}_n(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}}) \leq \sqrt{\frac{1}{n} \left(\frac{3}{2} + \frac{B \left\| \mathbf{w} \right\|_2}{w_p} \right)} \left\| \mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}} \right\|_1 .$$

Proof Due to space restrictions, the proof is somewhat abbreviated. Let $\hat{\mathbf{y}} \triangleq h(\mathbf{x}; \dot{\mathbf{w}})$ denote the original MAP configuration, i.e., the minimizer of $E(\cdot \,|\, \mathbf{x}; \mathbf{w})$. Let $\hat{\mathbf{y}}' \triangleq h(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}})$ denote the updated MAP state after conditioning, and note that $\hat{\mathbf{y}}'_{\overline{\mathcal{S}}}$ is the minimizer of $E\left(\Gamma(\mathbf{y}_{\mathcal{S}}, \cdot) \,|\, \mathbf{x}; \dot{\mathbf{w}}\right)$. Since $\hat{\mathbf{y}}_{\mathcal{S}}$ may be different from $\mathbf{y}_{\mathcal{S}}$, we have that $\hat{\mathbf{y}}$ may not be in the domain of $E\left(\Gamma(\mathbf{y}_{\mathcal{S}}, \cdot) \,|\, \mathbf{x}; \dot{\mathbf{w}}\right)$. We therefore define a vector $\tilde{\mathbf{y}} \in [0, 1]^n$ that *is* in the domain, and has minimal Hamming distance to $\hat{\mathbf{y}}$. Let $\tilde{y}_i \triangleq y_i$ for all $i \in \mathcal{S}$, and $\tilde{y}_j \triangleq \hat{y}_j$ for all $j \notin \mathcal{S}$. It can be shown that

$$\left\| \hat{\mathbf{y}}' - \hat{\mathbf{y}} \right\|_2^2 = \left\| \hat{\mathbf{y}}' - \tilde{\mathbf{y}} \right\|_2^2 + \left\| \tilde{\mathbf{y}} - \hat{\mathbf{y}} \right\|_2^2 . \quad (5)$$

Further, since the domain of each Y_i is $[0, 1]$,

$$\left\| \tilde{\mathbf{y}} - \hat{\mathbf{y}} \right\|_2^2 = \left\| \mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}} \right\|_2^2 \leq \left\| \mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}} \right\|_1 . \quad (6)$$

Therefore, combining Equations 5 and 6,

$$\begin{aligned} \left\| \hat{\mathbf{y}} - \hat{\mathbf{y}}' \right\|_2^2 &= \frac{1}{2} \left(\left\| \hat{\mathbf{y}} - \hat{\mathbf{y}}' \right\|_2^2 + \left\| \hat{\mathbf{y}}' - \hat{\mathbf{y}} \right\|_2^2 \right) \\ &\leq \frac{1}{2} \left(\left\| \hat{\mathbf{y}} - \hat{\mathbf{y}}' \right\|_2^2 + \left\| \hat{\mathbf{y}}' - \tilde{\mathbf{y}} \right\|_2^2 + \left\| \mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}} \right\|_1 \right) . \quad (7) \end{aligned}$$

For any κ-strongly convex function, $\varphi : \Omega \to \mathbb{R}$, where $\hat{\boldsymbol{\omega}} = \arg\min_{\boldsymbol{\omega} \in \Omega} \varphi(\boldsymbol{\omega})$ is the minimizer, then $\forall\, \boldsymbol{\omega}' \in \Omega$,

$$\frac{1}{2} \left\| \hat{\boldsymbol{\omega}} - \boldsymbol{\omega}' \right\|_2^2 \leq \frac{1}{\kappa} \left(\varphi(\boldsymbol{\omega}') - \varphi(\hat{\boldsymbol{\omega}}) \right) . \quad (8)$$

Applying this identify to the first two terms in Equation 7, since $E(\cdot \mid \mathbf{x}; \dot{\mathbf{w}})$ is w_p-strongly convex, we have that

$$\frac{1}{2}\left\|\hat{\mathbf{y}} - \hat{\mathbf{y}}'\right\|_2^2 + \frac{1}{2}\left\|\hat{\mathbf{y}}' - \tilde{\mathbf{y}}\right\|_2^2 \leq \frac{1}{w_p}\left(E(\tilde{\mathbf{y}} \mid \mathbf{x}; \dot{\mathbf{w}}) - E(\hat{\mathbf{y}} \mid \mathbf{x}; \dot{\mathbf{w}})\right). \quad (9)$$

The $E(\hat{\mathbf{y}}' \mid \mathbf{x}; \dot{\mathbf{w}})$ terms cancel out. Expanding $E(\cdot \mid \mathbf{x}; \dot{\mathbf{w}})$,

$$\begin{aligned} &E(\tilde{\mathbf{y}} \mid \mathbf{x}; \dot{\mathbf{w}}) - E(\hat{\mathbf{y}} \mid \mathbf{x}; \dot{\mathbf{w}}) \\ &= \mathbf{w} \cdot (\mathbf{f}(\mathbf{x}, \tilde{\mathbf{y}}) - \mathbf{f}(\mathbf{x}, \hat{\mathbf{y}})) + \frac{w_p}{2}\left(\|\tilde{\mathbf{y}}\|_2^2 - \|\hat{\mathbf{y}}\|_2^2\right) \\ &\leq \|\mathbf{w}\|_2 \|\mathbf{f}(\mathbf{x}, \tilde{\mathbf{y}}) - \mathbf{f}(\mathbf{x}, \hat{\mathbf{y}})\|_2 + w_p \|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1 . \quad (10) \end{aligned}$$

The last inequality uses Cauchy-Schwarz and

$$\|\tilde{\mathbf{y}}\|_2^2 - \|\hat{\mathbf{y}}\|_2^2 \leq 2\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1 .$$

Finally, we construct a series of vectors, indexed by each $i \in \mathcal{S}$, that transform $\hat{\mathbf{y}}$ into $\tilde{\mathbf{y}}$, one coordinate at a time. For the following, let $\mathcal{S}(j)$ denote the j^{th} element in $\mathcal{S}$. First, let $\tilde{\mathbf{y}}^{(0)} \triangleq \hat{\mathbf{y}}$; then, for $j = 1, \ldots, m$, let $\tilde{\mathbf{y}}^{(j)}$ be equal to $\tilde{\mathbf{y}}^{(j)}$ with index $\mathcal{S}(j)$ replaced with value $\tilde{y}_{\mathcal{S}(j)}$. Note that $\tilde{\mathbf{y}}^{(m)} = \tilde{\mathbf{y}}$. Using the triangle inequality, one can show that

$$\begin{aligned} \|\mathbf{f}(\mathbf{x}, \tilde{\mathbf{y}}) - \mathbf{f}(\mathbf{x}, \hat{\mathbf{y}})\|_2 &\leq \sum_{j=1}^{m} \left\|\mathbf{f}(\mathbf{x}, \tilde{\mathbf{y}}^{(j)}) - \mathbf{f}(\mathbf{x}, \tilde{\mathbf{y}}^{(j-1)})\right\|_2 \\ &\leq B\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1 . \quad (11) \end{aligned}$$

The last inequality uses Equation 4, since $\tilde{\mathbf{y}}^{(j)}$ and $\tilde{\mathbf{y}}^{(j-1)}$ differ at a single coordinate, $\mathcal{S}(j)$. Combining Equations 7 and 9 to 11, we have that

$$\left\|\hat{\mathbf{y}} - \hat{\mathbf{y}}'\right\|_2^2 \leq \left(\frac{3}{2} + \frac{B\|\mathbf{w}\|_2}{w_p}\right)\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1 .$$

We then multiply both sides of the inequality by $1/n$ and take the square root. Using $\frac{1}{n}\left\|\hat{\mathbf{y}} - \hat{\mathbf{y}}'\right\|_1 \leq \frac{1}{\sqrt{n}}\left\|\hat{\mathbf{y}} - \hat{\mathbf{y}}'\right\|_2$ finishes the proof. ■

Proposition 1 states that the inference regret is proportional to the L^1 distance from $\mathbf{y}_{\mathcal{S}}$ to $\hat{\mathbf{y}}_{\mathcal{S}}$, multiplied by a model-dependent quantity, $\mathrm{O}\left(\frac{B\|\mathbf{w}\|_2}{n\, w_p}\right)$. Later in this section, we discuss how to bound the features' Lipschitz constant, B, demonstrating that it is typically a small constant (e.g., 1). Thus, assuming $\|\mathbf{w}\|_2$ is bounded from above, and the weight on the prior, w_p, is bounded from below, the model-dependent term should decrease with the number of variables, n. For variables bounded in $[0, 1]$, the Hamming distance upper-bounds the L^1 distance. Using this identity, a pessimistic upper bound for the distance term is $\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1 \leq |\mathcal{S}|$. In this case, the regret is proportional to $\mathrm{O}\left(\sqrt{|\mathcal{S}|/n}\right)$; i.e., the square root of the fraction of the variables that are fixed. While this yields a uniform, analytic upper bound, we gain more insight by considering the specific contexts.

For instance, suppose $\mathbf{y}_{\mathcal{S}}$ is a set of labels that has been revealed. Then $\mathfrak{R}_n(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}})$ is the regret of not updating inference conditioned on new evidence, and $\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1$ is the L^1 error of the original predictions w.r.t. the true labels. Now, suppose $\mathbf{y}_{\mathcal{S}}$ is a set of labels that are fixed from a previous round of inference. Then $\mathfrak{R}_n(\mathbf{x}, \mathbf{y}_{\mathcal{S}}; \dot{\mathbf{w}})$ is the regret of an approximate inference update, and $\|\mathbf{y}_{\mathcal{S}} - \hat{\mathbf{y}}_{\mathcal{S}}\|_1$ is the L^1 distance between the old predictions and the new predictions in the full inference update. Thus, to minimize this regret, we must fix values that are already close to what we think they will be in the updated MAP state. This criteria motivates our approximate update methods in Section 4.

3.1.1 The Lipschitz Constant of the Features

In this section, we give some intuition on how to bound the Lipschitz constant of the features, B, by considering a specific example. Suppose the model has a single rule: $X \Rightarrow Y$. The corresponding hinge is $f(X, Y) \triangleq \max\{0, X - Y\}$. Using the fact that $|\max\{0, a\} - \max\{0, b\}| \leq |a - b|$, one can show that $\|\mathbf{f}(\mathbf{x}, \mathbf{y}) - \mathbf{f}(\mathbf{x}, \mathbf{y}')\|_2 \leq |y_i - y_i'| \leq 1$, so $B = 1$.

PSL models typically use rules of this nature, with varying arity (i.e., diadic, triadic, etc.). In general, B should grow linearly with the number of groundings involving any single variable (i.e., the maximum degree of the factor graph). The number of groundings generated by each rule depends on its arity and the data. For instance, the relational rule in Section 2 will ground out once for each edge and each label; if there are 2 labels, and the maximum degree is bounded by a constant, Δ, then the number of groundings generated by this rule for any single variable is at most 2Δ. Thus, in many practical models, B will be a small constant.

4 BUDGETED ONLINE INFERENCE

The bounds presented in Section 3.1 suggest that online collective inference under budget constraints is close to the full inference update when one is able to successfully choose and fix variables whose inferred values will have little or no change. We refer to the complementary process of selecting which variables to infer as *activation*. In practice, designing an activation algorithm is difficult. The optimization problem required to choose a set of variables, each with heterogeneous regret and optimization cost, that do not exceed an optimization budget is an instance of the NP-hard knapsack problem. Given the intrinsic intractability of selecting an optimal set of variables, we present two algorithms that employ theoretical insights from the previous section and show promise in empirical experiments.

4.1 BACKGROUND: ADMM OPTIMIZATION

To develop activation algorithms, we turn to the optimization technique used to determine the MAP state in HL-MRFs. Bach *et al.* (2012) have shown that applying consensus optimization using the Alternating Direction Method of Multipliers (ADMM) (Boyd *et al.*, 2011) provides scalable inference for HL-MRFs. For clearer exposition, we express the inference in terms of the set of ground rules, $\mathcal{G}$ and rewrite the energy function in Section 2 as:

$$E(\mathbf{y} \mid \mathbf{x}; \dot{\mathbf{w}}) \triangleq \sum_{g \in \mathcal{G}} w_g f_g(\mathbf{x}, \mathbf{y}) + \frac{w_p}{2} \|\mathbf{y}\|_2^2$$

Here, $w_g f_g(\mathbf{x}, \mathbf{y})$ is a weighted potential corresponding to a single ground rule. ADMM substitutes the global optimization problem with local optimizations for each potential using independent copies of the variables. For each grounding $g \in \mathcal{G}$, let $\mathbf{y}_g$ denote the variables involved in g and $\tilde{\mathbf{y}}_g$ indicate the local copy of those variables. To reconcile the local optimizations, ADMM introduces a constraint that local variable copies agree with the global "consensus" for each variable i involved in the grounding; that is, $\mathbf{y}_g[i] = \tilde{\mathbf{y}}_g[i]$. This constraint is transformed into an augmented Lagrangian with penalty parameter $\rho > 0$ and Lagrange multipliers $\boldsymbol{\alpha}_g$:

$$\min_{\tilde{\mathbf{y}}_g} \; w_g \, f_g(\mathbf{x}, \tilde{\mathbf{y}}_g) + \frac{\rho}{2} \left\| \tilde{\mathbf{y}}_g - \mathbf{y}_g + \frac{1}{\rho} \boldsymbol{\alpha}_g \right\|^2 \qquad (12)$$

ADMM iteratively alternates optimizing the local potentials, then updating the consensus estimates and associated Lagrange multipliers for each variable, as such:

$$\tilde{\mathbf{y}}_g \leftarrow \text{argmin}_{\tilde{\mathbf{y}}_g} w_g \, f_g(\mathbf{x}, \tilde{\mathbf{y}}_g) + \frac{\rho}{2} \left\| \tilde{\mathbf{y}}_g - \mathbf{y}_g + \frac{1}{\rho} \boldsymbol{\alpha}_g \right\|^2 ;$$
$$\mathbf{y}[i] \leftarrow \text{mean}_g(\tilde{\mathbf{y}}_g[i]) \; ; \;\; \boldsymbol{\alpha}_g[i] \leftarrow \boldsymbol{\alpha}_g[i] + \rho(\tilde{\mathbf{y}}_g[i] - \mathbf{y}_g[i]) \, .$$

A key element of this optimization is the interplay of two components: the weighted potential corresponding to a grounding and the Lagrangian penalty for deviating from the consensus estimate. As optimization proceeds, the Lagrange multipliers are updated to increase the penalty for deviating from the global consensus. At convergence, a balance exists between the two components, reconciling the local minimizer and the aggregate of global potentials.

4.2 ADMM FEATURES

The goal of activation is to determine which variables are most likely to change in a future inference. From the analysis in the previous section, we can identify several basic elements for each variable in the model that serve as features for an activation algorithm. For each variable, we have its value at convergence ($\mathbf{y}[i]$), and for each grounding g, the weight (w_g), the value of the potential ($f_g(\mathbf{x}, \tilde{\mathbf{y}}_g)$), and the Lagrange multipliers ($\boldsymbol{\alpha}_g[i]$) measuring the aggregate deviation from consensus. We discuss each of these features to motivate their importance in an activation algorithm.

The value of a variable at convergence can provide a useful signal in certain situations, where a model has clear semantics. For example, the formulation of HL-MRFs often lends itself to a logical interpretation with binary outcomes, as in the cases of collective classification of attributes that are either present or absent. In this setting, assignments in the vicinity of 0.5 represent uncertainty, and therefore provide good candidates for activation. Unfortunately, this feature is not universal. Many successful HL-MRF models adopt semantics that use continuous values to model continuous variables, such as pixel intensity in image completion tasks or Likert-scale ratings in recommender systems. In this case, the semantics of the variable's consensus value may provide an ambiguous signal for activation.

The weighted potentials of each variable contribute directly to the probability of the MAP configuration. Since the log-probability is proportional to the *negated* energy, $-E$, high weights and high potential values decrease the probability of the assignment. Intuitively, activating those variables that contribute high weighted potentials provides the best mechanism for approaching the full inference MAP state. A complication to this approach is that each weighted potential can depend on many variables. However, the potential value is a scalar quantity and there is no general mechanism to apportion the loss to the contributing variables.

In contrast, the Lagrange multipliers provide a granular perspective on each variable's effect on Equation 12. For each variable copy ($\tilde{\mathbf{y}}_g$), the Lagrange multiplier aggregates the difference between the copy and the global consensus across iterations. High Lagrange multipliers signal discord between the local minimizer and the global minimizer, indicating volatility. Activating variables with high Lagrange multipliers can resolve this discord in future inference using updated evidence. However, updated evidence may also resolve the disagreement between the local and global minimum, obviating an update to the variable.

4.3 ACTIVATION ALGORITHMS

Building on our analysis of ADMM optimization, we introduce two activation algorithms for online collective inference, "agnostic activation" and "relational activation". Both algorithms produce a ranking that prioritizes each variable for inclusion in inference. The key difference between these algorithms is whether new or updated evidence is an input to the algorithm. Agnostic activation scores variables concurrently with inference, based on their susceptibility to change in future inferences. In contrast, relational activation runs prior to inference, with scores based primarily on relationships between variables and updated evidence in the factor graph.

Each approach has different advantages. Agnostic activation scores variables during inference, providing a performance advantage since the scoring algorithm does not delay a future run of inference. However, this technique has a slower response to new evidence since scoring occurs before such evidence is available. Relational activation can respond to newly-arrived evidence and choose variables related to new evidence, but this requires delaying scoring which can add a computational overhead to inference.

Both activation algorithms output a ranking of the variables, which requires a scoring function. We introduce two scoring functions that use the ADMM features described Section 4.2. Our first scoring function, VALUE, captures the intuition that uncertain variables are valuable activation candidates using the function $1 - |0.5 - \mathbf{y}[i]|$, where $\mathbf{y}[i]$ is the consensus value for variable i. The second scoring function, WLM, uses both the weight and Lagrange multipliers of each potential. For each variable, we define a set of weighted Lagrange multiplier magnitudes, $\mathcal{A}_w[i] \triangleq \{|w_g \boldsymbol{\alpha}_g[i]|\}$. To obtain a single scalar score, we take the maximum value of $\mathcal{A}_w[i]$.

The agnostic activation algorithm simply ranks each variable by their score from a scoring function, irrespective of the new evidence. The RELATIONAL algorithm combines the score with information about the new evidence. Using the existing ground model, RELATIONAL first identifies all ground potentials dependent on the new evidence. Then, using these ground potentials as a seed set, the algorithm performs a breadth-first search of the factor graph adding the variables involved in each factor it encounters to the frontier. Traversing the factor graph can quickly identify many candidate variables, so we prioritize variables in the frontier by $\frac{S}{2^d}$ where S is the score assigned by a scoring function and d is the minimum distance between the variable and an element of the seed set in the factor graph.

The ranking output by either agnostic or relational activation lets us prioritize which variables to activate. Given a budget for the number or percentage of variables to infer, we activate a corresponding number of variables from the ranking. The remaining variables are constrained to their previously inferred values. We selectively ground the model, including only those rules that involve an activated variable. Following inference on the ground model, we use the updated optimization state to produce new scores.

When an inactive variable is treated as a constant, it does not have any associated Lagrange multipliers, and lacks features for the WLM scoring function. Therefore, instead of treating fixed variables as constants, we introduce them as constrained variables in the optimization. This allows us to generate features by capturing the discrepancy between a variable's constrained value and the value of its local copies in groundings involving activated variables.

Our implementation of the agnostic activation algorithm is extremely efficient; all necessary features are byproducts of the inference optimization. Once scores are computed and the activated atoms are selected, the optimization state can be discarded to avoid additional resource commitments. In relational activation, scoring is similarly efficient, but there is an additional overhead of preserving the ground model to allow fast traversal of the factor graph. By selectively grounding the model, we replace queries that scan the entire database, potentially many times, with precise queries that exploit indices for faster performance. Finally, selectively activating atoms produces an optimization objective with fewer terms, allowing quicker optimization.

5 EVALUATION

To better understand the regret bounds and approximation algorithms for online inference, we perform an empirical evaluation on two online collective inference settings. The first setting is a synthetic online collective classification task where the data generator allows us to modulate the importance of collective dependencies and control the amount of noise. The second evaluation setting is a real-world collaborative filtering task, where user preferences are incrementally revealed and the outputs of a recommender system are correspondingly updated.

In both evaluation settings, we measure regret relative to full inference and inference error relative to ground truth. The results demonstrate that empirical regret follows the form of our regret bounds. We also evaluate the approximation algorithms presented in Section 4.3, to investigate whether features from the optimization algorithm can reliably determine which variables to activate. The results show that our approximation algorithms are able to reduce running time by upwards of 65%, with inference regret relative to full inference.

All experiments are implemented using the open-source PSL framework and our code is available on GitHub[1].

5.1 ONLINE COLLECTIVE CLASSIFICATION

Our evaluation data simulates a collective classification problem of inferring labels for users in a social network as new evidence is incrementally revealed. Each user is assigned one of two mutually exclusive labels. Some portion of the users have observed labels, while the labels of the remaining users are inferred. At each epoch, the label of one more user is revealed, so the model must update the inferred labels for the remaining users with unknown labels.

For each user, we generate local and collective features correlated with the user's label. Local features are generated for each user and label by drawing from a Gaussian distribution conditioned on the label, such that the mean is t for

[1] https://github.com/puuj/uai15-boci-code

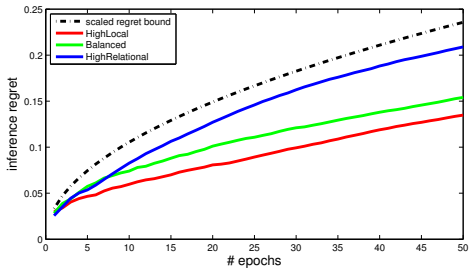
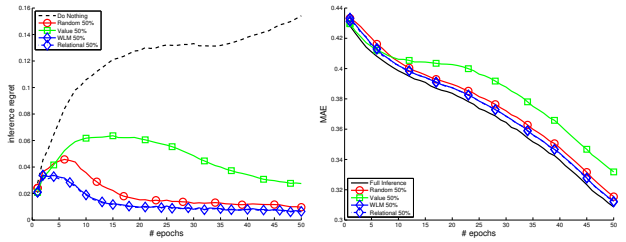


Figure 1: Inference regret, w.r.t. full inference, of fixing the original MAP state (i.e., no updates) in the HIGHLOCAL, HIGHCOLLECTIVE and BALANCED data models.



(a) Inference Regret (b) MAE

Figure 2: Inference regret (w.r.t. full inference) and MAE (w.r.t. ground truth) using various approximation algorithms, with 50% activation, in the COMPLEX data model.

the true label and $1-t$ for the incorrect label. The collective features are links between users, generated randomly using the following process: for each pair of users with the same label, a link is generated with probability p; for each pair of users with different labels, a link is created with probability $1-p$. We refer to p as the affinity of the network.

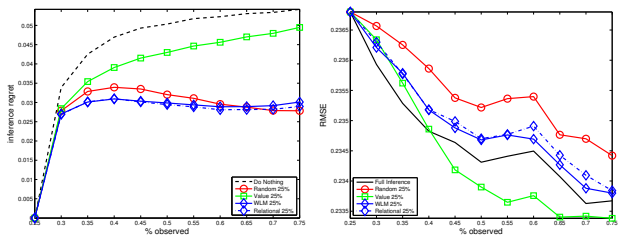
We model the data using the PSL rules described in Section 2 and learn weights for the model. Varying the parameters of the data generator impacts inference in the learned model, since the learned weights are proportional to the discriminative power of their associated rules. For example, varying the distance between the conditional means of the local features controls the importance of the local evidence rule: when the means are far apart, local evidence has high discriminative power; however, when the means are close, local evidence does not provide much signal.

We introduce three data models: HIGHLOCAL ($t = .8, p = .75$), HIGHCOLLECTIVE ($t = .55, p = .9$), and BALANCED ($t = .7, p = .75$). We combine these three conditions in a fourth data model, COMPLEX, which samples uniformly from the three settings on a per-user basis resulting in heterogeneous evidence. For each condition, we generate 10 trials, each with a training social network used to learn the model parameters and a separate test social network to evaluate inference quality. Both the training and test graph have 100 users, with 60 observed user labels in the training graph and 10 observed user labels in the test graph. To infer user attributes, we use the simple collective classification model introduced in Section 2. We simulate the process of online inference by creating a sequence of observations consisting of 50 epochs. In each epoch, the true label of a previously unknown user is revealed, resulting in 60 observed user labels at the end of the sequence. For each trial, we generate 10 such sequences from a breadth-first traversal of the network from a randomly chosen user, resulting in a total of 5000 inferences.

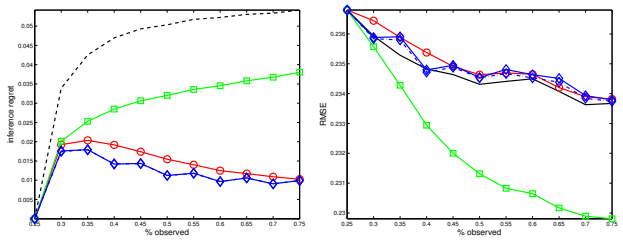
In the first experiment, shown in Figure 1 we measure the inference regret of fixing variables to the initial MAP state (i.e., not updating inference) over 50 epochs, comparing the HIGHLOCAL, HIGHCOLLECTIVE and BALANCED conditions. Our theoretical analysis predicts that the worst-case regret grows at rate $\mathrm{O}\left(1/\sqrt{\text{epoch}}\right)$. The experimental results exhibit the same growth rate, which is very pronounced for the HIGHCOLLECTIVE data model, where variables are strongly interdependent, and less so for HIGHLOCAL, where variables are largely independent. The key insight is that the collective nature of the inference task determines the regret of online updates.

In the second experiment (Figure 2), we compare the approximate scoring algorithms with a budget of 50% of unknowns to running full inference on the COMPLEX network. We measure significance across 100 total sequences using a paired t-test with rejection threshold .05. For inference regret, we compare against the static algorithm, DONOTHING, which does not update the MAP state, and a random baseline, RANDOM, that fixes an arbitrary subset of 50% of the variables. We compare these to three approximation algorithms described in Section 4.1: VALUE, which uses the value assigned to the variable; WLM, which uses the maximum of the weighted Lagrange multipliers; and RELATIONAL, which uses WLM to prioritize exploration.

All methods exhibit low regret relative to full inference, contrasting the high regret of the static algorithm, although VALUE exhibits somewhat higher regret. The WLM and RELATIONAL methods have significantly lower regret relative to RANDOM, in 98% and 100% of epochs, respectively. We also compare the mean average error (MAE), with respect to ground truth, of using full inference vs. the approximations. This illustrates that the approximation algorithms remain competitive with full inference, although VALUE again lags in accuracy. Here, the WLM and RELATIONAL methods have significantly lower error than RANDOM in 80% and 100% of epochs, respectively. Comparing the running times highlights the computational benefit of using the approximation algorithms. The average running time for a single trial (which includes training and 10 random sequences of revealed variables) using full inference is 3076 seconds, while approximate inference requires only 955 seconds, a reduction of 69%, with inference time varying less than 3% across methods.



(a) Inference Regret (25% act.) (b) RMSE (25% act.)



(c) Inference Regret (50% act.) (d) RMSE (50% act.)

Figure 3: Inference regret (w.r.t. full inference) and RMSE (w.r.t. ground truth) for the Jester dataset.

5.2 COLLABORATIVE FILTERING

Our second evaluation task is a collaborative filtering task that employs a collective model to infer the preferences of users. We use the Jester dataset (Goldberg *et al.*, 2001) which includes ratings from 24,983 users on a set of 100 jokes. The task in this setting is to infer the user's rating of each joke. We use the model from Bach *et al.* (2013) which assigns ratings to jokes based on the joke's similarity to other jokes rated highly by the user. Joke similarity is measured using the mean-adjusted cosine similarity of the observed ratings of two jokes. (Refer to Bach *et al.* (2013) for further model details.) We sample 200 users who have rated all 100 jokes and split them into 100 training users and 100 testing users. We generate 10 sequences, each of which consists of a training and testing phase. Model weights are learned using 75% of the training users' ratings observed. During testing, we incrementally reveal $[25\%, 30\%, 40\%, \ldots, 75\%]$ of the testing users' ratings, performing online collective inference at each epoch.

We compare inference regret, relative to full inference, for the RANDOM, VALUE, WLM and RELATIONAL approximate methods. We also plot the RMSE, relative to ground truth, for full inference and all approximate methods. Figure 3a-b show results for 25% activation, and Figure 3c-d show 50% activation. Inference regret follows a similar pattern for both budgets, with VALUE showing increasing regret over epochs, and the remaining methods exhibiting level or diminishing regret after the first few epochs. The high regret for VALUE can be explained by considering the RMSE—VALUE actually *improves* the results of full inference, incurring high regret but low RMSE. Our intuition for this improvement is that VALUE fixes polarized user ratings and allows these ratings to have greater influence on other unknown ratings, while full inference produces more moderate ratings for the entire set. The other approximation algorithms remain close to the full inference RMSE (at 50% activation) or perform slightly worse (at 25% activation). Comparing the running times, we find a similar improvement in speed. The average time for a sequence using full inference is 137 seconds, while the approximate methods require only 46 seconds, yielding a speedup of 66%. Approximation methods had consistent timing, varying less than 6%.

6 CONCLUSION

In this paper, we introduce a new problem, *budgeted online collective classification*, which addresses a common problem setting where online inference is necessary but full inference is infeasible, thereby requiring approximate inference updates. Our contributions are: (1) a formal analysis of online collective inference, introducing the concept of inference regret to measure the quality of the approximation; (2) analytic upper bounds on the inference regret incurred by strongly convex inference; and (3) several algorithms to address the practical problem of activation (i.e., choosing which variables to infer at each epoch), through a close analysis of the MAP inference optimization. Our empirical results demonstrate that our activation algorithms exhibit low inference regret and error that is competitive with full inference, while reducing the time required for inference by 65% or more.

This work inspires many exciting areas of future research. One open question is whether one can derive a tighter regret bound using the mechanics of the activation strategy, thus characterizing how performance degrades as a function of the budget. We are also interested in training an "optimal" activation policy that is trained using the variables whose values change the most during full inference. Finally, a crucial assumption in our analysis is that the model structure is fixed, but it is useful to consider the setting in which the set of variables change over time, allowing us to address situations such as new users joining a social network.

Acknowledgments This work was partially supported by National Science Foundation (NSF) grant IIS1218488 and by the Intelligence Advanced Research Projects Activity (IARPA) via DoI/NBC contract D12PC00337. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF, IARPA, DoI/NBC, or the U.S. Government.

References

U. Acar, A. Ihler, R. Mettu, and Ö. Sümer. Adaptive inference on general graphical models. In *UAI*, 2008.

U. Acar, A. Ihler, R. Mettu, and Ö. Sümer. Adaptive updates for MAP configurations with applications to bioinformatics. In *IEEE Statistical Signal Processing (SSP)*, pages 413–416. 2009.

S. H. Bach, M. Broecheler, L. Getoor, and D. P. O'Leary. Scaling MPE inference for constrained continuous markov random fields with consensus optimization. In *NIPS*, 2012.

S. H. Bach, B. Huang, B. London, and L. Getoor. Hinge-loss Markov random fields: Convex inference for structured prediction. In *UAI*, 2013.

S. H. Bach, M. Broecheler, B. Huang, and L. Getoor. Hinge-loss Markov random fields and probabilistic soft logic. arXiv:1505.04406 [cs.LG], 2015.

I. Beltagy, K. Erk, and R. J. Mooney. Probabilistic soft logic for semantic textual similarity. In *ACL*, 2014.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends Machine Learning*, 3(1):1–122, 2011.

W. Buntine. Theory refinement on bayesian networks. In *UAI*, 1991.

H. Chan and A. Darwiche. Sensitivity analysis in Markov networks. In *IJCAI*, 2005.

H. Chan and A. Darwiche. On the robustness of most probable explanations. In *UAI*, 2006.

P.-T. Chen, F. Chen, and Z. Qian. Road traffic congestion monitoring in social media with hinge-loss Markov random fields. In *ICDM*, 2014.

S. Fakhraei, B. Huang, L. Raschid, and L. Getoor. Network-based drug-target interaction prediction with probabilistic soft logic. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2014.

S. Fine, Y. Singer, and N. Tishby. The hierarchical hidden Markov model: Analysis and applications. *Machine Learning*, 32(1):41–62, 1998.

N. Friedman and M. Goldszmidt. Sequential update of Bayesian network structure. In *UAI*, 1997.

Peter Gardenfors, editor. *Belief Revision*. Cambridge University Press, New York, NY, USA, 1992.

A. Globerson and T. Jaakkola. Fixing max-product: convergent message passing algorithms for MAP LP-relaxations. In *NIPS*, 2007.

K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

K. Laskey. Sensitivity analysis for probability assessments in Bayesian networks. In *UAI*, 1993.

W. Li, P. van Beek, and P. Poupart. Performing incremental Bayesian inference by dynamic model counting. In *AAAI*, 2006.

B. London, B. Huang, B. Taskar, and L. Getoor. Collective stability in structured prediction: Generalization from one example. In *ICML*, 2013.

B. London, S. Khamis, S. H. Bach, B. Huang, L. Getoor, and L. Davis. Collective activity detection using hinge-loss markov random fields. In *CVPR Workshop on Structured Prediction: Tractability, Learning and Inference*, 2013.

B. London, B. Huang, B. Taskar, and L. Getoor. PAC-Bayesian collective stability. In *AIStats*, 2014.

K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

A. Nath and P. Domingos. Efficient belief propagation for utility maximization and repeated inference. In *AAAI*, 2010.

J. Pujara, H. Miao, L. Getoor, and W. Cohen. Knowledge graph identification. In *ISWC*, 2013.

A. Ramesh, D. Goldwasser, B. Huang, Hal Daumé III, and L. Getoor. Learning latent engagement patterns of students in online courses. In *AAAI*, 2014.

Ö. Sümer, U. Acar, A. Ihler, and R. Mettu. Adaptive exact inference in graphical models. *JMLR*, 12:3147–3186, 2011.