Joint Prediction of Topics in a URL Hierarchy

Michael Großhans[1], Christoph Sawade[2], Tobias Scheffer[1], and Niels Landwehr[1]

[1] University of Potsdam, Department of Computer Science, August-Bebel-Straße 89,
14482 Potsdam, Germany
{grosshan,landwehr,scheffer}@cs.uni-potsdam.de
[2] SoundCloud Ltd., Greifswalderstraße 212, 10405 Berlin, Germany
christoph@soundcloud.com

Abstract. We study the problem of jointly predicting topics for all web pages within URL hierarchies. We employ a graphical model in which latent variables represent the predominant topic within a subtree of the URL hierarchy. The model is built around a generative process that infers how web site administrators hierarchically structure web site according to topic, and how web page content is generated depending on the page topic. The resulting predictive model is linear in a joint feature map of content, topic labels, and the latent variables. Inference reduces to message passing in a tree-structured graph; parameter estimation is carried out using concave-convex optimization. We present a case study on web page classification for a targeted advertising application.

1 Introduction

Web page classification of entire web domains has numerous applications. For instance, topic labels can be used to match individual web pages to related advertisements; topic labels can be aggregated over pages that a user has visited in order to create a profile of the user's interests. Classifying the child suitability of web pages is another typical use case.

There is a rich body of research on topic classification of web pages based on page content [8]. Classification does not have to rely on page content alone. For instance, collective classification schemes that exploit the hyperlink structure within the world wide web have also been widely studied [6,11]. Collective classification approaches define probabilistic models over web page content and the observed hyperlink structure; inference in the models yields the most likely joint configuration of topic labels. Typically, discriminative models over topics given page content and link structure are studied, in order to avoid having to estimate high-dimensional distributions over page content. For example, maximum margin Markov networks have been shown to give excellent results in hypertext classification domains [11,2].

In this paper, we study models that exploit the information inherent in the URL hierarchy of a web domain, rather than the information contained in the hyperlink structure. Many web domains organize their individual pages in a meaningful hierarchy in which subtrees tend to contain web pages of similar topics. The predominant topic within a particular subtree constitutes a latent variable from the learner's perspective; correctly inferring these latent variables and propagating the topic information to pages within the subtree has the potential to boost predictive accuracy if topic correlation within subtrees is strong.

The presence of latent variables constitutes a key difference of our problem setting in comparison to collective classification based on hyperlinks; models developed for collective hypertext classification are typically not applicable as they cannot deal with latent variables during learning. The problem could instead be modeled using latent variable structured output models, such as latent variable SVMstruct [14]. The challenge when using this approach is to correctly model the interaction of latent and observed variables using a joint feature map and specifying appropriate loss functions while ensuring that decoding remains tractable.

We instead follow an approach in which we formulate a generative model of how URL trees are populated with topic labels and content is generated for web pages within that URL tree. The model can be formulated conveniently using topic-correlation models and standard exponential-family distributions for page content given a topic. The model is then trained discriminatively by maximizing the conditional distribution of topic labels given page content and the URL tree. This conditional distribution has the form of a linear model with a joint feature map of the URL hierarchy, page content, and the (observable and latent) topic labels. Efficient decoding is possible by message passing in a tree-structured graph. In this formulation, the feature map, decoding algorithm, and optimization criterion directly result from the probabilistic modeling assumptions.

The rest of this paper is organized as follows. Section 2 states the problem setting and introduces notation. Section 3 presents the probabilistic model, and Section 4 discusses parameter learning and inference. Section 5 reports on an empirical study on web classification for a targeted advertising application. Section 6 discusses related work, Section 7 concludes.

2 Notation and Problem Setting

A URL tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ consists of vertices and edges. The vertices $\mathcal{V}$ include n leaves that we will write as $v_1, \ldots, v_n$, and k inner nodes, written as $v_{n+1}, \ldots, v_{n+k}$. The leaves correspond to URLs of individual web pages (such as *washingtonpost.com/local/crime/murder.htm*); inner nodes corresponds to prefixes of these URLs that end in a separator—typically, the slash symbol. An inner node, such as *washingtonpost.com/local/*, thus represents a subtree in the URL hierarchy. Figure 1 (left) shows an imaginary URL tree for a web domain.

The content of each of the leaf nodes is encoded as a vector $\mathbf{x}_i \in \mathbb{R}^m$; we denote the content of the entire web portal as matrix $\mathbf{X} = (\mathbf{x}_1 \ldots \mathbf{x}_n) \in \mathbb{R}^{m \times n}$. The topic space is denoted by $\mathcal{Y}$; the vector $\mathbf{y} = (y_1, ..., y_n)^\mathsf{T} \in \mathcal{Y}^n$ denotes topic labels for the n leaf nodes that correspond to web pages.

Web domains are generated by an unknown distribution $p(\mathbf{y}, \mathbf{X}, \mathcal{T})$; we will make specific modeling assumptions about this distribution in the next section. Distinct web domains are drawn independently and from an identical distribu-

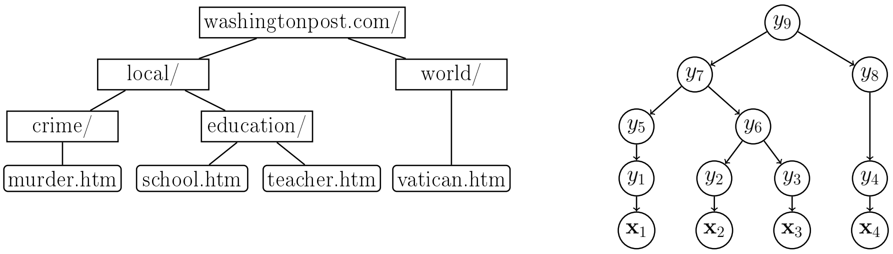


Fig. 1. URL tree (left) and graphical model (right) of an exemplary news domain.

tion. Note, however, that the random variables $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)$ that represent content and topic information for any single domain are not assumed to be independent. Typically, the variables will correlate as a function of their position in the URL hierarchy $\mathcal{T}$. A training sample of several labeled web domains is drawn according to this distribution.

Finally, the URL tree $\mathcal{T}^*$ and the content matrix $\mathbf{X}^*$ of a target web domain are drawn. In addition, topic labels y_i^* for a limited number (possibly zero) of leaf nodes are disclosed. The goal is to infer the most likely complete vector of topic labels $\mathbf{y}^*$ for this target domain.

3 Graphical Model

In order to define an appropriate probabilistic model for the problem stated in Section 2, we first define a model of a generative process that we assume to have generated the observable data in Section 3.1. In Section 3.2, we express this model as a member of the exponential family. Deriving the conditional distribution $p(\mathbf{y}|\mathbf{X}; \mathcal{T})$ in this model results in a linear structured-output model. In our application, inference can be carried out efficiently using message passing inference in the tree-structured model (Section 4.1). Parameters can be estimated according to maximal conditional likelihood using CCCP (Section 4.2).

3.1 A Generative Process for Web Domains

In this section, we define a generative process for populating a given URL tree $\mathcal{T}$ with topic labels and word count information (we will make no modeling assumptions about the distribution $p(\mathcal{T})$ over URL trees).

The general assumption that underlies our model is that web site administrators hierarchically structure web site content according to topic, such that topics of pages within one subtree of the URL hierarchy correlate. To represent the predominant topic within specific subtrees, we associate a vector of latent topic variables $\bar{\mathbf{y}} = (y_{n+1}, ..., y_{n+k}) \in \mathcal{Y}^k$ to the k inner nodes $v_{n+1}, \ldots, v_{n+k}$ of the URL tree $\mathcal{T}$. These latent topic variables will couple the observable topic variables $\mathbf{y} = (y_1, ..., y_n)$ through the URL hierarchy. Throughout the paper, we denote by $v_{\to i} \in \mathcal{V}$ the unique parent of the node $v_i \in \mathcal{V}$ specified by the edge set $\mathcal{E}$ of URL tree $\mathcal{T}$. We extend this definition to topic labels as follows: for a topic variable $y_i \in \{y_1, ..., y_{n+k}\}$, we write $y_{\to i}$ to denote the latent topic variable associated with the node $v_{\to i}$.

We assume a top-down generative process for topic variables by modeling the dependency between a topic y_i and the topic $y_{\to i}$ of the parent node as a distribution $p(y_i|y_{\to i}; \lambda)$. The distribution is modeled as a *normalized exponential*

$$p(y_i|y_{\to i}; \lambda) = \frac{\exp\left(-\lambda \Delta(y_i, y_{\to i})\right)}{\sum\limits_{y' \in \mathcal{Y}} \exp\left(-\lambda \Delta(y', y_{\to i})\right)} \tag{1}$$

where the function $\Delta(y_i, y_{\to i})$ measures topic distance in $\mathcal{Y}$. A simple choice for topic distance would be $\Delta(y, y') = 0$ if $y = y'$ and $\Delta(y, y') = 1$ if $y \neq y'$; other distance functions may be employed to reflect a specific structure on the topic space. The parameter λ controls the degree of correlation expected between the topic variables y_i and $y_{\to i}$. This generative process corresponds to the assumption that when administrators add novel material that covers topic $y \in \mathcal{Y}$ to a web domain, they insert a corresponding URL subtree under a parent node that is associated with a topic $y_{\to i}$ close to y. We assume this process is carried out recursively up to and including the leaf nodes, that is, novel URL subtrees are again populated with subtrees and eventually web pages with topics that are close within the topic space $\mathcal{Y}$. The prior distribution $p(y_{n+k}|\boldsymbol{\tau})$ over the topics of the root is a categorical distribution over topics, parametrized by $\boldsymbol{\tau}$.

In order to complete the specification of the data-generating process we have to assume a distribution $p(\mathbf{x}|y)$ over word-count information $\mathbf{x}$ given the web page topic y. At this point, we only assume that this distribution is a member of the exponential family and follows the general form

$$p\left(\mathbf{x}|y; \boldsymbol{\eta}\right) = h(\mathbf{x}) \exp\left(\boldsymbol{\eta}^{\mathsf{T}} \phi(\mathbf{x}, y) - g_{\boldsymbol{\eta}}(\boldsymbol{\eta}, y)\right). \tag{2}$$

In Equation 2, $h(\mathbf{x})$ is called the base measure, $g_{\boldsymbol{\eta}}(\boldsymbol{\eta}, y)$ is the log-partition function that ensures correct normalization of the distribution, $\phi(\mathbf{x}, y)$ is a joint feature map of the web page $\mathbf{x}$ and topic y, and $\boldsymbol{\eta}$ is a parameter vector.

By defining a joint feature map of $\mathbf{x}$ and y, we subsume the case of modeling topic-specific parameter vectors for $\phi(\mathbf{x}, y) = \Lambda(y) \otimes \phi(\mathbf{x})$, where operator $\otimes$ denotes the Kronecker product and $\Lambda(y) = ([\![y = \bar{y}]\!])_{\bar{y} \in \mathcal{Y}}$ denotes the one-of-k encoding of y, but can also encode structural prior knowledge, for instance about a structured topic space.

By combining the generative process for topic variables based on Equation 1 and the categorical distribution $p(y_{n+k}|\boldsymbol{\tau})$ with the conditional distribution defined by Equation 2, we obtain a generative model $p(\mathbf{y}, \bar{\mathbf{y}}, \mathbf{X}|\mathcal{T}; \lambda, \boldsymbol{\eta}, \boldsymbol{\tau})$ of all topic variables given web page texts and the URL tree:

$$p(\mathbf{y}, \bar{\mathbf{y}}, \mathbf{X}|\mathcal{T}; \lambda, \boldsymbol{\eta}, \boldsymbol{\tau}) = p(y_{n+k}|\boldsymbol{\tau}) \left(\prod_{i=1}^{n+k-1} p\left(y_i|y_{\to i}; \lambda\right) \right) \prod_{i=1}^{n} p\left(\mathbf{x}_i|y_i; \boldsymbol{\eta}\right). \tag{3}$$

Figures 1 (right) shows the graphical model representation of this model for the example URL tree shown in Figure 1 (left).

3.2 A Discriminative Joint Topic Model

Starting from the generative process defined in Section 3.1, we now derive a discriminative model for page topics based on URL hierarchy $\mathcal{T}$ and page texts $\mathbf{X}$.

We begin the derivation by casting the generative process $p(\mathbf{y}, \bar{\mathbf{y}}, \mathbf{X}|\mathcal{T}; \lambda, \boldsymbol{\eta}, \boldsymbol{\tau})$ defined in Section 3.1 into an exponential-family model using a joint feature map $\varPhi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})$. Note that $p(y_i|y_{\to i}; \lambda)$ can be written in the canonical form of an exponential family by $p(y_i|y_{\to i}; \lambda) = \exp\left(-\lambda \varDelta(y_i, y_{\to i}) - g_\lambda(\lambda, y_{\to i})\right)$ where $g_\lambda(\lambda, y_{\to i}) = \log \sum_{y' \in \mathcal{Y}} \exp\left(-\lambda \varDelta(y', y_{\to i})\right)$, and $p(y_{n+k}|\boldsymbol{\tau})$ can be written in exponential family form by $p(y_{n+k}|\boldsymbol{\tau}) = \exp\left(\boldsymbol{\tau}^\mathsf{T} \varLambda(y) - g_{\boldsymbol{\tau}}(\boldsymbol{\tau})\right)$ where $g_{\boldsymbol{\tau}}(\boldsymbol{\tau}) = \log(\mathbf{1}^\mathsf{T} \exp(\boldsymbol{\tau}))$. Then, Equation 3 can be written as

$$
\begin{aligned}
&p(\mathbf{y}, \bar{\mathbf{y}}, \mathbf{X}|\mathcal{T}; \boldsymbol{\theta}) \\
&\quad= \exp\left(\boldsymbol{\tau}^\mathsf{T} \varLambda(y) - g_{\boldsymbol{\tau}}(\boldsymbol{\tau})\right) \left(\prod_{i=1}^{n+k-1} \exp\left(-\lambda \varDelta(y_i, y_{\to i}) - g_\lambda(\lambda, y_{\to i})\right) \right) \\
&\quad\quad \prod_{i=1}^{n} h(\mathbf{x}_i) \exp\left(\boldsymbol{\eta}^\mathsf{T} \phi(\mathbf{x}_i, y_i) - g_{\boldsymbol{\eta}}(\boldsymbol{\eta}, y_i)\right) \\
&\quad= \exp\left(\boldsymbol{\tau}^\mathsf{T} \varLambda(y) - g_{\boldsymbol{\tau}}(\boldsymbol{\tau})\right) \exp\left(-\lambda \sum_{i=1}^{n+k-1} \varDelta(y_i, y_{\to i}) - \sum_{i=1}^{n+k-1} g_\lambda(\lambda, y_{\to i}) \right) \\
&\quad\quad \left(\prod_{i=1}^{n} h(\mathbf{x}_i) \right) \exp\left(\boldsymbol{\eta}^\mathsf{T} \sum_{i=1}^{n} \phi(\mathbf{x}_i, y_i) - \sum_{i=1}^{n} g_{\boldsymbol{\eta}}(\boldsymbol{\eta}, y_i) \right) \\
&\quad= h(\mathbf{X}) \exp\left(\boldsymbol{\theta}^\mathsf{T} \varPhi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}}) - g_{\boldsymbol{\tau}}(\boldsymbol{\tau})\right), \qquad (4)
\end{aligned}
$$

where we define a joint parameter vector $\boldsymbol{\theta} = (\boldsymbol{\eta}^\mathsf{T}, \boldsymbol{\tau}^\mathsf{T}, \lambda, \boldsymbol{\gamma}^\mathsf{T})^\mathsf{T}$, a feature map

$$
\varPhi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}}) = \begin{pmatrix} \sum_{i=1}^{n} \phi(\mathbf{x}_i, y_i) \\ \varLambda(y_{n+k}) \\ \sum_{i=1}^{n+k-1} \varDelta(y_i, y_{\to i}) \\ \sum_{i=1}^{n+k-1} \varLambda(y_{\to i}) \end{pmatrix}, \qquad (5)
$$

and base measure $h(\mathbf{X}) = \prod_{i=1}^{n} h(\mathbf{x}_i)$. Note that in Equation 4 we subsumed the sum of log-partition functions $\sum_{i=1}^{n} g_{\boldsymbol{\eta}}(\boldsymbol{\eta}, y_i)$ into the model parameter $\boldsymbol{\eta}$ by adding a constant feature to the feature map $\phi(\mathbf{x}, y)$ for each $y \in \mathcal{Y}$. Additionally we subsumed the sum of log-partition functions $\sum_{i=1}^{n+k-1} g_\lambda(\lambda, y_{\to i})$ into an additional model parameter $\boldsymbol{\gamma}$.

The conditional distribution $p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta})$ over observable and latent topic variables given web page content and the URL tree is now given by

$$\begin{aligned} p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta}) =& \frac{p(\mathbf{y}, \bar{\mathbf{y}}, \mathbf{X}|\mathcal{T}; \boldsymbol{\theta})}{\sum_{\mathbf{y}', \bar{\mathbf{y}}'} p(\mathbf{y}', \bar{\mathbf{y}}', \mathbf{X}|\mathcal{T}; \boldsymbol{\theta})} \\ =& \frac{h(\mathbf{X}) \exp(g_{\boldsymbol{\mathcal{T}}}(\boldsymbol{\tau})) \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right)}{h(\mathbf{X}) \exp(g_{\boldsymbol{\mathcal{T}}}(\boldsymbol{\tau})) \sum_{\mathbf{y}', \bar{\mathbf{y}}'} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}', \bar{\mathbf{y}}')\right)} \\ =& \frac{\exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right)}{\sum_{\mathbf{y}', \bar{\mathbf{y}}'} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}', \bar{\mathbf{y}}')\right)}. \end{aligned} \tag{6}$$

Note that Equation 6 defines a linear structured-output model in the joint feature map $\Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})$ because $\arg\max_{\mathbf{y}, \bar{\mathbf{y}}} \; p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta}) = \arg\max_{\mathbf{y}, \bar{\mathbf{y}}} \; \boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})$.

4 Inference and Parameter Estimation

We now turn toward the problem of inferring topic variables and obtaining maximum-a-posteriori estimates of model parameters from data.

4.1 Inferring Topics for New Web Portals

For a given new web domain, inference might target the most likely joint assignment of topics to web pages by summing out the latent variables $\bar{\mathbf{y}}$,

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} \sum_{\bar{\mathbf{y}}} p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta}). \tag{7}$$

Unfortunately, this problem is NP-hard even for tree-structured graphs [4]. Instead, we are able to infer the most likely topic assignment y_i of the i-th page by summing out latent variables $\bar{\mathbf{y}}$ and topic variables of all other web pages $\mathbf{y}_{\bar{i}}$,

$$y_i^* = \arg\max_{y} \sum_{\mathbf{y}_{\bar{i}}, \bar{\mathbf{y}}} p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta}). \tag{8}$$

Alternatively, we can infer the most likely joint state of all topic variables,

$$(\mathbf{y}^*, \bar{\mathbf{y}}^*) = \arg\max_{\mathbf{y}, \bar{\mathbf{y}}} p(\mathbf{y}, \bar{\mathbf{y}}|\mathbf{X}, \mathcal{T}; \boldsymbol{\theta}), \tag{9}$$

thereby also inferring topics for inner nodes in the URL hierarchy $\mathcal{T}$. For the application motivating this paper, the latter approach is advantageous if web sites are very dynamic: if novel pages are added to the URL tree and there is insufficient time to carry out a full inference, the topic assigned to the parent node of the added page can be used to label the novel page quickly. This is often the only feasible approach for real-time systems, and is in fact implemented in the targeted-advertisement company that we collaborate with.

Moreover, if topics $\mathbf{y}_{\bar{S}}$ of a subset of web pages $\bar{S} \subseteq \{1, \ldots, n\}$ are already observed, the most likely conditional joint assignment for the unobserved labels $\mathbf{y}_S$ where $S = \{1, \ldots, n\} \backslash \bar{S}$ and the latent variables $\bar{\mathbf{y}}$ given the observed labels is

$$(\mathbf{y}_S^*, \bar{\mathbf{y}}^*) = \operatorname*{arg\,max}_{\mathbf{y}_S, \bar{\mathbf{y}}} p(\mathbf{y}_S, \bar{\mathbf{y}} | \mathbf{X}, \mathbf{y}_{\bar{S}}, \mathcal{T}_U; \boldsymbol{\theta}). \tag{10}$$

Due to the tree-structured form of the model given by Equation 3, the optimization problems given by Equation 8, 9, and 10 can be solved efficiently using standard message passing algorithms [7].

4.2 Parameter Estimation

To estimate model parameters, we minimize the regularized discriminative negative log-likelihood over all URL trees:

$$\begin{aligned} \boldsymbol{\theta}^* &= \operatorname*{arg\,min}_{\boldsymbol{\theta}} \ \mathit{\Omega}(\boldsymbol{\theta}) - \log \prod_{j=1}^{u} p(\mathbf{y}^j | \mathbf{X}^j, \mathcal{T}_j; \boldsymbol{\theta}) \\ &= \operatorname*{arg\,min}_{\boldsymbol{\theta}} \ \mathit{\Omega}(\boldsymbol{\theta}) + \sum_{j=1}^{u} \ell_{log}(\boldsymbol{\theta}, \mathbf{X}^j, \mathbf{y}^j), \end{aligned} \tag{11}$$

where the loss function is given by

$$\ell_{log}(\boldsymbol{\theta}, \mathbf{X}, \mathbf{y}) = \log \sum_{\mathbf{y}', \bar{\mathbf{y}}'} \exp\left(\boldsymbol{\theta}^\mathsf{T} \mathit{\Phi}(\mathbf{X}, \mathbf{y}', \bar{\mathbf{y}}')\right) - \log \sum_{\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^\mathsf{T} \mathit{\Phi}(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right).$$

In order to specify the regularizer $\mathit{\Omega}(\boldsymbol{\theta}) = \mathit{\Omega}_{\boldsymbol{\eta},\gamma}(\boldsymbol{\eta}, \gamma) + \mathit{\Omega}_{\boldsymbol{\tau}}(\boldsymbol{\tau}) + \mathit{\Omega}_\lambda(\lambda_j)$, we assume a zero-mean Gaussian prior $(\boldsymbol{\eta}, \gamma) \sim \mathcal{N}(\mathbf{0}, \sigma^2_{\boldsymbol{\eta},\gamma}\mathbf{I})$ with variance $\sigma^2_{\boldsymbol{\eta},\gamma}$ over $\boldsymbol{\eta}$ and $\boldsymbol{\gamma}$, a Dirichlet prior over the topic distribution $p(y_{n+k}|\boldsymbol{\tau})$ at the root node, and an inverse gamma prior $\lambda \sim \mathrm{InvGam}(1, \sigma^2_\lambda)$ over the coupling parameter λ, where the inverse gamma distribution is parameterized using mean and variance. Given these prior distributions the regularizing terms are defined by:

$$\mathit{\Omega}_{\boldsymbol{\eta},\gamma}(\boldsymbol{\eta}, \gamma) = \frac{1}{2\sigma^2_{\boldsymbol{\eta},,\gamma}} \left(\|\boldsymbol{\eta}\|_2^2 + \|\gamma\|_2^2\right), \quad \mathit{\Omega}_\lambda(\lambda) = \frac{\sigma^{-2} + 1}{\lambda} + (\sigma_\lambda^{-2} + 3) \log(\lambda),$$
$$\mathit{\Omega}_{\boldsymbol{\tau}}(\boldsymbol{\tau}) = \log(\mathbf{1}^\mathsf{T} \exp(\boldsymbol{\tau}))\mathbf{1}^\mathsf{T}(\boldsymbol{\alpha} - \mathbf{1}) - (\boldsymbol{\alpha} - \mathbf{1})^\mathsf{T}\boldsymbol{\tau}.$$

In Equation 11 we determine the minimizing argument of a sum of a convex and a concave function $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} f_\cup(\boldsymbol{\theta}) + f_\cap(\boldsymbol{\theta})$ where

$$f_\cup(\boldsymbol{\theta}) = \mathit{\Omega}_\cup(\boldsymbol{\theta}) + \sum_{j=1}^{u} \log \sum_{\mathbf{y}', \bar{\mathbf{y}}'} \exp\left(\boldsymbol{\theta}^\mathsf{T} \mathit{\Phi}(\mathbf{X}^j, \mathbf{y}', \bar{\mathbf{y}}')\right)$$
$$f_\cap(\boldsymbol{\theta}) = \mathit{\Omega}_\cap(\boldsymbol{\theta}) - \sum_{j=1}^{u} \log \sum_{\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^\mathsf{T} \mathit{\Phi}(\mathbf{X}^j, \mathbf{y}^j, \bar{\mathbf{y}})\right),$$

Algorithm 1 Concave-Convex Procedure

```
1: Initialize $\boldsymbol{\theta}^*$
2: repeat
3:    Construct upper bound on $f_\cap(\boldsymbol{\theta})$ for some $c$: $f_/(\boldsymbol{\theta}) \leftarrow \boldsymbol{\theta}^\mathsf{T}\left[\nabla f_\cap(\boldsymbol{\theta})\right]_{\boldsymbol{\theta}=\boldsymbol{\theta}^*}+\mathrm{c}$
4:    $\boldsymbol{\theta}^* \leftarrow \arg\min_{\boldsymbol{\theta}} f_\cup(\boldsymbol{\theta}) + h(\boldsymbol{\theta})$
5: until $\boldsymbol{\theta}^*$ converges
6: return $\boldsymbol{\theta}^*$
```

and where

$$\varOmega_\cup(\boldsymbol{\theta}) = \varOmega_{\boldsymbol{\eta},\boldsymbol{\gamma}}(\boldsymbol{\eta},\boldsymbol{\gamma}) + \varOmega_{\boldsymbol{\tau}}(\boldsymbol{\tau}) + \frac{\sigma^{-2}+1}{\lambda}, \qquad \varOmega_\cap(\boldsymbol{\theta}) = (\sigma_\lambda^{-2}+3)\log(\lambda)$$

subsume the convex and the concave part of the regularization term. Thus, the optimization problem given by Equation 11 is in general not convex—although it is convex in $\boldsymbol{\eta}$, since $f_\cap(\boldsymbol{\theta})$ is a linear function in $\boldsymbol{\eta}$. In order to solve the optimization problem, we use the Concave-Convex Procedure, which is guaranteed to converge to a local optimum [15]. Algorithm 1 iteratively upper-bounds the concave part $f_\cap(\boldsymbol{\theta})$ by the linear function $f_/(\boldsymbol{\theta})$ (see Line 4) and solves the resulting convex optimization problem in Line 5 using standard gradient descend methods. The gradients with respect to $\boldsymbol{\theta}$ are given by:

$$\nabla f_\cup(\boldsymbol{\theta}) = \nabla \varOmega_\cup(\boldsymbol{\theta}) + \sum_{j=1}^{u} \mathbb{E}^j_{\mathbf{y}',\bar{\mathbf{y}}}\left[\varPhi(\mathbf{X}^j,\mathbf{y}',\bar{\mathbf{y}})\right],$$
$$\nabla f_\cap(\boldsymbol{\theta}) = \nabla \varOmega_\cap(\boldsymbol{\theta}) + \sum_{j=1}^{u} \mathbb{E}^j_{\bar{\mathbf{y}}}\left[\varPhi(\mathbf{X}^j,\mathbf{y}^j,\bar{\mathbf{y}})\right],$$

where expectation $\mathbb{E}^j_{\mathbf{y}',\bar{\mathbf{y}}}$ bases on distribution $p(\mathbf{y}',\bar{\mathbf{y}}|\mathbf{X}^j,\mathcal{T}_j;\boldsymbol{\theta})$ and expectation $\mathbb{E}^j_{\bar{\mathbf{y}}}$ bases on $p(\bar{\mathbf{y}}|\mathbf{y}^j,\mathbf{X}^j,\mathcal{T}_j;\boldsymbol{\theta}) = p(\mathbf{y}^j,\bar{\mathbf{y}}|\mathbf{X}^j,\mathcal{T}_j;\boldsymbol{\theta})/\sum_{\bar{\mathbf{y}}'} p(\mathbf{y}^j,\bar{\mathbf{y}}'|\mathbf{X}^j,\mathcal{T}_j;\boldsymbol{\theta})$. The gradients of the regularization parts are given by

$$\nabla \varOmega_\cup(\boldsymbol{\theta}) = \sigma_{\boldsymbol{\eta},\boldsymbol{\gamma}}^{-2}\boldsymbol{\eta} + \sigma_{\boldsymbol{\eta},\boldsymbol{\gamma}}^{-2}\boldsymbol{\gamma} + \frac{\mathbf{1}^\mathsf{T}(\boldsymbol{\alpha}-\mathbf{1})}{\mathbf{1}^\mathsf{T}\exp(\boldsymbol{\tau})}\exp(\boldsymbol{\tau}) - (\boldsymbol{\alpha}-\mathbf{1}) - \frac{\sigma_\lambda^{-2}+1}{\lambda^2}$$
$$\nabla \varOmega_\cap(\boldsymbol{\theta}) = \frac{\sigma_\lambda^{-2}+3}{\lambda}.$$

The following proposition states that the gradients can be evaluated efficiently using message passing.

Proposition 1. *Let $\phi(\mathbf{x},y) = \varLambda(y)\otimes\mathbf{x}$. Then the expectations $\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\varPhi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ and $\mathbb{E}_{\bar{\mathbf{y}}}\left[\varPhi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ can be computed in time*

$$\mathcal{O}(|\mathcal{Y}|^3(n+k)^2 + |\mathcal{Y}|nm),$$

where m denotes the number of features, n the number of leaf nodes, and k the number of inner nodes of the URL tree.

A proof of Proposition 1 can be found in the appendix. Computations are based on variations of standard message passing; additional computational savings are realized by reusing specific messages within the overall computation of the gradient. These savings depend on the URL structure under study and thus do not influence asymptotic complexity but significantly influence empirical execution time for the web domains that we have studied.

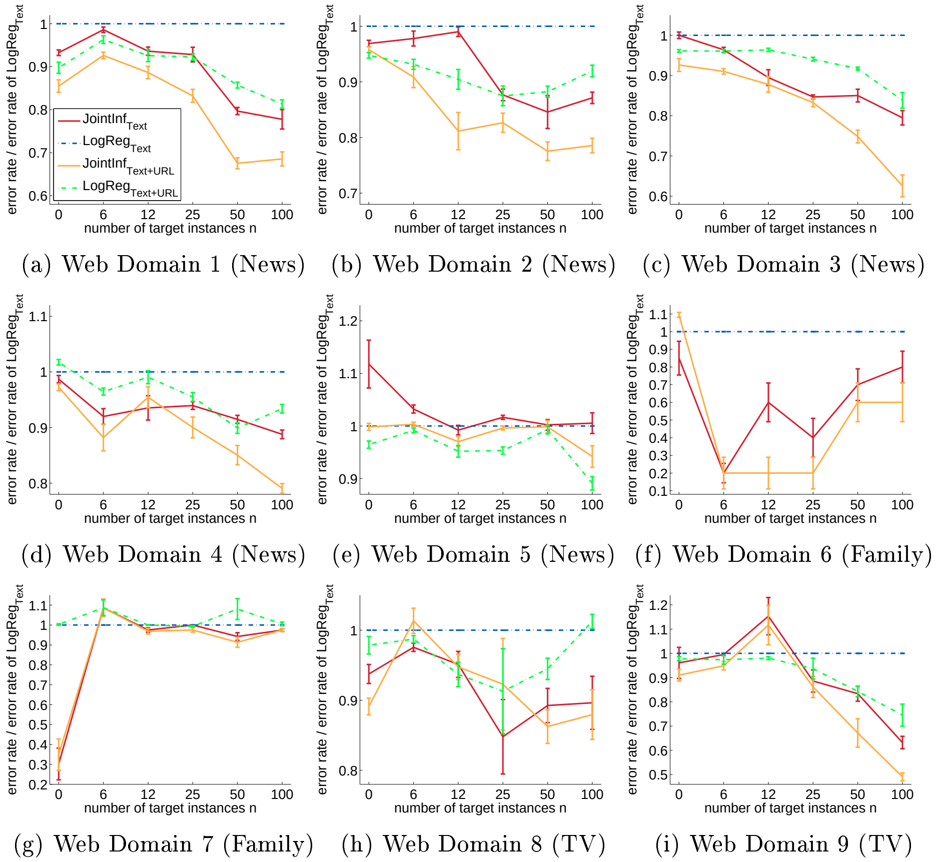
5 Empirical Study

We empirically investigate the predictive performance of the proposed joint topic model using data from a large targeted advertisement company. The data set contains 36,579 web pages within nine web domains that have been manually annotated with topic labels by human labelers employed by the targeted advertising company. We use a total of $|\mathcal{Y}| = 30$ labels. Out of the nine web domains, five are general news portals run by large newspaper publishers, two are more topic-specific web portals run by family magazines, and two are web portals run by TV stations. Web page content is represented using a binary bag-of-words encoding. Words that occur fewer than ten times in the training data are removed from the dictionary, this results in 94,624 distinct bag-of-words features.

We study the model proposed in Sections 3 and 4 (denoted $\mathit{JointInf}_{Text}$), where we choose a linear feature map $\phi(\mathbf{x}, y) = \Lambda(y) \otimes \mathbf{x}$. The topic distance $\Lambda(y, y')$ in Equation 1 is one if $y = y'$ and zero otherwise. As a baseline, we obtain predictions from a logistic regression model that independently predicts topics for individual web pages (denoted LogReg_{Text}). As a further baseline, we study a logistic-regression model based on an augmented feature representation that concatenates the text features with a binary bag-of-words representation of the page's URL where numbers or special characters are used as separator between words (denoted $\mathit{LogReg}_{Text+URL}$). We also study our model using this augmented feature representation (denoted $\mathit{JointInf}_{Text+URL}$). Such URL features have been shown to be predictive for web page classification; for example, Baykan et al. have studied web page classification based on URL features only [1].

5.1 Topic Classification Performance

We study topic prediction in each of the nine web domains (the *target domain*). Training data includes instances from the remaining eight web domains (the *training domains*) as well as a varying number of instances from the target domain. Specifically, a training set is obtained by sampling 100 labeled web pages from each of the training domains and between $n = 0$ and $n = 100$ labeled web pages from the target domain. At $n = 0$ this corresponds to a setting in which predictions for novel web domains have to be obtained given a training set of existing web domains. At $n > 0$ this corresponds to a setting in which a small set of manually labeled seed pages is available from the target domain, and topic labels for the remaining pages need to be predicted. For $\mathit{JointInf}_{Text}$ and $\mathit{JointInf}_{Text+URL}$, the labeled pages from the target domain constitute observed



(a) Web Domain 1 (News) (b) Web Domain 2 (News) (c) Web Domain 3 (News)

(d) Web Domain 4 (News) (e) Web Domain 5 (News) (f) Web Domain 6 (Family)

(g) Web Domain 7 (Family) (h) Web Domain 8 (TV) (i) Web Domain 9 (TV)

Fig. 2. Average error ratio for different target portals and different number of labeled web sites from target portal.

variables, inference is carried out conditioned on these observations (see Equation 10). Predictive performance is evaluated on a sample of 500 pages from the remaining web pages of the target domain.

Hyperparameters of all models are tuned using grid search and a leave-one-domain-out cross-validation on the training data. News domains generally exhibit more diverse topic labels than the topic-specific web portals run by TV stations and family magazines; for tuning we therefore evaluate the models only on domains from the same of these two groups as the target domain. For $\mathit{JointInf}_{Text}$ and $\mathit{JointInf}_{Text+URL}$ hyperparameters are the coupling parameter σ^2_λ and the regularization parameter $\sigma^2_{\boldsymbol{\eta},\boldsymbol{\gamma}}$; for LogReg_{Text} and $\mathit{LogReg}_{Text+URL}$, the standard regularization parameter. The hyperparameter of the Dirichlet prior is set to $\boldsymbol{\alpha} = (2, ..., 2)^\top$ (Laplace smoothing).

Figure 2 shows predictive performance for all web domains as a function of n, averaged over five resampling iterations of web pages from the training

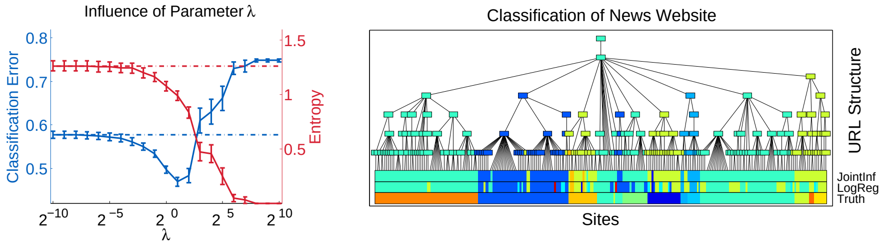


Fig. 3. Classification error and empirical entropy for different choices of parameter λ on *News Portal 1* (left). Labeled URL structure (right) for sample of *News Portal 1* for $\lambda = 1$ and zero labeled web sites from target domain.

and target domains. Specifically, we report the ratio of the mean zero-one error rate of each method to the mean zero-one error of the baseline *LogReg*$_{Text}$. If both methods incur an error rate of zero, the quotient is defined to be one. In one experiment (Web Domain 6, *LogReg*$_{Text+URL}$) the quotient was undefined because the error of *LogReg*$_{Text}$ was zero while the error of *LogReg*$_{Text+URL}$ was nonzero. Thus the curve for *LogReg*$_{Text+URL}$ is missing from Figure 2(f).

From Figure 2 we observe that, on average, the methods *JointInf*$_{Text}$ and *JointInf*$_{Text+URL}$ predict topic labels more accurately than their corresponding baseline *LogReg*$_{Text}$ or *LogReg*$_{Text+URL}$. Additionally we observe that the inclusion of URL features in the feature representation on average improves predictive accuracy. Performance varies for different web domains and values of n, with the best case being a reduction in error rate of approximately 80% and the worst case an increase in error rate by approximately 20% compared to the baseline model.

5.2 Effect of the Model Parameter $\boldsymbol{\lambda}$

We also study the influence of λ—which controls the structural homogeneity of the classifier prediction (Equation 1)—for *News Portal 1* and $n = 0$. Figure 3 (left) shows the error rate of *JointInf*$_{Text}$ (blue solid line) and the *LogReg*$_{Text}$ (blue dashed line) as a function of λ. In these experiments, the regularization parameter $\sigma^2_{\boldsymbol{\eta},\boldsymbol{\gamma}} = 1$ is fixed and only model parameters $\boldsymbol{\eta}$, $\boldsymbol{\gamma}$ and $\boldsymbol{\tau}$ are optimized; results are averaged over 20 resampling iterations. Figure 3 (left) also shows the corresponding empirical entropy of the predicted labels (red curves). If λ converges to zero, the model assumes no correlation between topics of nodes and their parents in the URL tree; in this case, *JointInf*$_{Text}$ reduces to *LogReg*$_{Text}$. High values of λ couple topic labels strongly within the URL tree, therefore more uniform topic labels are assigned and the empirical entropy of the predicted labels is reduced. Predictive accuracy is maximal for intermediate values of λ.

Figure 3 (right) shows the label tree inferred by the joint topic model, predictions of the logistic regression baseline, and the ground truth for *News Portal 1*, $\lambda = 1$, and $n = 0$ labeled web pages of the target domain. We used a color

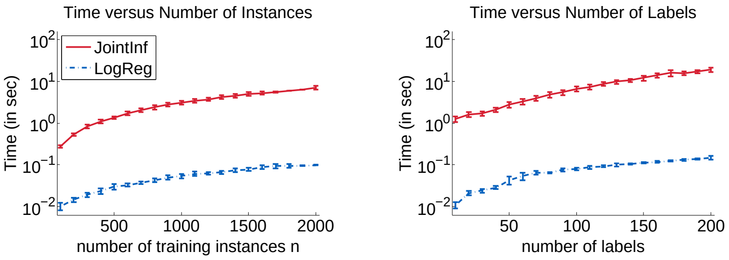


Fig. 4. Execution time for computation of expectation $\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ (*JointInf*$_{\text{Text}}$) and $\mathbb{E}_{\mathbf{y}}\left[\Phi(\mathbf{X},\mathbf{y})\right]$ (*LogReg*$_{\text{Text}}$) for different number of instances (left) and different number of labels (right). Error bars show standard errors.

scheme that maps related topics to related colors. It shows that topic labels are more uniform when using *JointInf*$_{Text}$ instead of *LogReg*$_{Text}$.

5.3 Execution Time

In our experiments, both the logistic regression model and the structured model are optimized using a gradient descent approach. Thus, the main computational difference is caused by the gradients of their loss functions: The computational time for the gradient of the structured loss $\ell_{\log}$ is dominated by evaluating the quantities $\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ and $\mathbb{E}_{\bar{\mathbf{y}}}\left[\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ (see Proposition 1). The gradient of the logistic loss function can be written as $\mathbb{E}_{\mathbf{y}}\left[\Phi(\mathbf{X},\mathbf{y})\right] - \Phi(\mathbf{X},\mathbf{y})$, where

$$\mathbb{E}_{\mathbf{y}}\left[\Phi(\mathbf{X},\mathbf{y})\right] = \sum_{i=1}^{n} \frac{\exp(\boldsymbol{\eta}^{\mathsf{T}}\phi(\mathbf{x}_i,y_i))\phi(\mathbf{x}_i,y_i)}{\sum_{y'}\exp(\boldsymbol{\eta}^{\mathsf{T}}\phi(\mathbf{x}_i,y'_i))}. \quad (12)$$

We compare the execution time for computation of expectation $\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right]$ for the joint topic model and the corresponding quantity given by Equation 12 for the logistic regression model. Figure 4 (left) shows the execution time for different number of training instances and a fixed number of labels $|\mathcal{Y}| = 30$. Figure 4 (right) shows the execution time for different number of labels—randomly assigned to instances—and a fixed number of instances $n = 500$. We found that the noticeable difference in time complexities—$\mathcal{O}(|\mathcal{Y}|^3(n+k)^2 + |\mathcal{Y}|nm)$ for joint topic model and $\mathcal{O}(|\mathcal{Y}|nm)$ for logistic regression—reduces approximately to a constant factor, when we reuse messages over different variations of the message passing scheme.

6 Related Work

There is a rich body of work on general web page classification [8]. In addition to textual information on pages, hyperlink structure is often used to improve classification accuracy [9,6].

Some earlier work has studied using URL tree information for web page classification. Kan and Thi [3] and Baykan et al. [1] use models over features of URLs to classify web pages based on URL information only. Shih and Karger [10] use URL trees and page layout information encoded in an HTML tree for ad blocking and predicting links that are of interest to a particular user. They employ a generative probabilistic model similar to the coupling model defined by Equation 1 to represent correlations within URL trees. In contrast to our approach, their model does not include web page text or other page content.

Tian et al. [12] study models based on URL tree information with the goal of assigning topics to entire web sites rather than individual web pages. Kumar et al. [5] study the problem of segmenting a web site into topically uniform regions based on the URL tree structure and predictions of a node-level topic classification algorithm. Their central result is that segmentations that are optimal according to certain cost measures can be computed efficiently using dynamic programming.

The prediction problem we study can be phrased as a structured output problem involving latent variables; such problems have been studied, for example, by Wang et al. [13] and Yu and Joachims [14]. The latter model, latent variable structured SVM, is also trained using CCCP. Its margin-based objective leads to a learning algorithm alternating between performing point estimates of latent variables and model parameters, while in our maximum conditional likelihood formulation latent variables are summed out during learning. In the application-specific model that we present, these summations as well as decoding for structured prediction can be carried out efficiently because both problems reduce to message passing in a tree-structured factor graph.

7 Conclusions

We have studied the problem of jointly predicting topic labels for all web pages within a URL hierarchy. Section 3.1 defines a generative process for web page content that captures our intuition about how web site administrators hierarchically structure web sites according to content; latent variables in this process reflect the predominant topic within a URL subtree. Section 3.2 shows that deriving the conditional distribution over topic labels given page content in this model results in a structured output model that is linear in a joint feature map of page content, topic labels, and latent topic variables. Parameter estimation can be carried out using a concave-convex procedure. Proposition 1 shows that parameter estimation and decoding in the model are efficient. An empirical study in a targeted advertisement domain shows that joint inference of topic labels with the proposed model is more accurate than inferring topic labels independently based on features derived from page content and the URL.

Acknowledgment

This work was carried out in collaboration with nugg.ad AG.

Appendix

Proof of Proposition 1

We first turn toward the quantity

$$\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right] = \frac{\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}}\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right)\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})}{\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}}\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right)}. \tag{13}$$

Let the unnormalized probabilities—the normalizing quantities $h(\mathbf{X})$ and $g_{\boldsymbol{\tau}}(\boldsymbol{\tau})$ can be canceled out in nominator and denominator—be denoted by

$$\psi_{\boldsymbol{\tau}}(y) = \exp(\tau_y) \propto p(y|\boldsymbol{\tau}) \qquad \psi_{\boldsymbol{\eta}}(y,\mathbf{x}) = \exp(\boldsymbol{\eta}^{\mathsf{T}}\phi(\mathbf{x},y)) \propto p(\mathbf{x}|y;\boldsymbol{\eta})$$
$$\psi_{\lambda,\boldsymbol{\gamma}}(y,y') = \exp(-\lambda\Delta(y,y') - \gamma_{y'}) = p(y|y';\lambda).$$

Since $\Delta(y,y')$ is a given problem-specific loss function, $\psi_{\lambda,\boldsymbol{\gamma}}(y,y')$ can be computed in time $\mathcal{O}(|\mathcal{Y}|^2)$ for all $y,y' \in \mathcal{Y}$. Furthermore, under the assumption that $\phi(\mathbf{x},y) = \Lambda(y) \otimes \mathbf{x}$, we can compute $\psi_{\boldsymbol{\eta}}(y,\mathbf{x}_i)$ for all $i = 1,\ldots,n$ and all $y \in \mathcal{Y}$ in time $\mathcal{O}(|\mathcal{Y}|nm)$. Given these quantities, the denominator in Equation 13 can be computed efficiently using standard message passing [7]. We therefore evaluate

$$\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}}\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right) = \sum_{y_{n+k}} \psi_{\boldsymbol{\tau}}(y_{n+k}) \prod_{v_{\rightarrow i}=v_{n+k}} \mu_i(y_{n+k}) \tag{14}$$

recursively, where the messages $\mu_i(y_{\rightarrow i})$ have the form

$$\sum_{y_i} \psi_{\lambda,\boldsymbol{\gamma}}(y_i,y_{\rightarrow i}) \begin{cases} \psi_{\boldsymbol{\eta}}(y_i,\mathbf{x}_i) & \text{, if } i \le n \\ \prod_{v_{\rightarrow j}=v_i} \mu_j(y_i) & \text{, otherwise.} \end{cases} \tag{15}$$

In order to evaluate Equation 15 for a given $i \in \{1,\ldots,n+k\}$ and a given $y_{\rightarrow i} \in \mathcal{Y}$, we have to compute all $|\mathcal{Y}|$ summands. Each summand contains at most $\max\{c_i+1,2\}$ factors, where c_i is the number of children of node v_i. Hence one message can be computed in time $\mathcal{O}(|\mathcal{Y}|c_i)$. Due to the tree structure $\mathcal{T}$, each node has a unique parent node and therefore $\sum_{i=1}^{n+k} c_i = n+k-1$ holds. Hence the computation for all $i \in \{1,\ldots,n+k\}$ and all $y_{\rightarrow i} \in \mathcal{Y}$ can be done in time $\sum_{i=1}^{n+k}\sum_{y_{\rightarrow i}\in\mathcal{Y}} \mathcal{O}(|\mathcal{Y}|c_i) = \mathcal{O}(|\mathcal{Y}|^2(n+k))$ using dynamic programming.

We now consider the numerator in Equation 13 and show that the parts of the joint feature map (see Equation 5) that refer to the parameters $\boldsymbol{\eta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\tau}$ and λ can be computed in time $\mathcal{O}(|\mathcal{Y}|^3 n(n+k) + |\mathcal{Y}|nm)$, in time $\mathcal{O}(|\mathcal{Y}|^3(n+k)^2)$, in time $\mathcal{O}(|\mathcal{Y}|^2(n+k))$, and in time $\mathcal{O}(|\mathcal{Y}|^2(n+k)^2)$, respectively. For $\boldsymbol{\eta}$, the numerator can be expressed as

$$\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}}\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right) \sum_{l=1}^{n} \phi(\mathbf{x}_l,y_l)$$
$$= \sum_{l=1}^{n} \left(\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}}\Phi(\mathbf{X},\mathbf{y},\bar{\mathbf{y}})\right) \Lambda(y_l) \right) \otimes \mathbf{x}_l. \tag{16}$$

In Equation 16, we reorder the sums and make use of $\phi(\mathbf{x}, y) = \Lambda(y) \otimes \mathbf{x}$. Additionally, we exploit the associativity of the Kronecker product. Note that $\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Lambda(y_l)$ is a vector of length $|\mathcal{Y}|$; each component is associated with the case that the l-th web site has a certain label y'. This quantity can be computed by applying the message passing for each label $y' \in \mathcal{Y}$, where the message $\mu_l(y_{\rightarrow l})$ is substituted by $\psi_{\lambda,\boldsymbol{\gamma}}(y_l, y_{\rightarrow l})\psi_{\boldsymbol{\eta}}(y_l, \mathbf{x}_l)$ if $y_l = y'$ and zero otherwise. In order to evaluate Equation 16, we need to apply standard message passing for all $y' \in \mathcal{Y}$ and for all $l = 1, \ldots, n$, which can be done in time $\mathcal{O}(|\mathcal{Y}|^3 n(n+k))$. The computation of the Kronecker product can be done in time $\mathcal{O}(|\mathcal{Y}|nm)$. Thus, the overall computational time is $\mathcal{O}(|\mathcal{Y}|^3 n(n+k) + |\mathcal{Y}|nm)$.

By reordering the sums, the numerator for $\boldsymbol{\gamma}$ can be expressed as

$$\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \sum_{l=1}^{n+k-1} \Lambda(y_{\rightarrow l}) = \sum_{l=1}^{n+k-1} \sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Lambda(y_{\rightarrow l}). \quad (17)$$

The term $\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Lambda(y_{\rightarrow l})$ is a vector of length $|\mathcal{Y}|$, where each component is associated with the case that the parent of the l-th node has certain label y'. For each label $y' \in \mathcal{Y}$, this quantity can be computed by standard message passing, where the message $\mu_l(y_{\rightarrow l})$ is substituted by

$$\psi_{\lambda,\boldsymbol{\gamma}}(y_l, y_{\rightarrow l}) \begin{cases} \psi_{\boldsymbol{\eta}}(y_l, \mathbf{x}_l) & \text{, if } l \leq n \text{ and } y_{\rightarrow l} = y' \\ \prod_{v_{\rightarrow j} = v_l} \mu_j(y_l) & \text{, if } l > n \text{ and } y_{\rightarrow l} = y' \\ 0 & \text{, otherwise.} \end{cases}$$

In order to evaluate Equation 17, we need to apply message passing for all $l = 1, \ldots, n+k-1$ and $y' \in \mathcal{Y}$, which can be done in time $\mathcal{O}(|\mathcal{Y}|^3 (n+k)^2)$.

For $\boldsymbol{\tau}$, the numerator can be evaluated by using message passing

$$\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Lambda(y_{n+k}) \quad = \sum_{y_{n+k}} \psi_{\boldsymbol{\tau}}(y_{n+k}) \Lambda(y_{n+k}) \prod_{v_{\rightarrow i} = v_{n+k}} \mu_i(y_{n+k}), \quad (18)$$

where $\mu_i(y_{\rightarrow i})$ is defined by Equation 15. Standard message passing as described in Equation 14 requires a summation over $|\mathcal{Y}|$ summands. Instead, in Equation 18 we save each of the $|\mathcal{Y}|$ summands. Hence, the computational time for Equation 18 is the same as for Equation 14, which is $\mathcal{O}(|\mathcal{Y}|^2 (n+k))$.

For λ, the numerator can be expressed as

$$\begin{aligned} &\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \sum_{l=1}^{n+k-1} \Delta(y_l, y_{\rightarrow l}) \\ &\qquad = \sum_{l=1}^{n+k-1} \left(\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Delta(y_l, y_{\rightarrow l}) \right) \quad (19) \end{aligned}$$

by reordering the sums. Again, we use the message passing algorithm in order to evaluate the quantity $\sum_{\mathbf{y},\bar{\mathbf{y}}} \exp\left(\boldsymbol{\theta}^{\mathsf{T}} \Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right) \Delta(y_l, y_{\rightarrow l})$ for $l = 1, \ldots, n+k-1$.

Therefore, we substitute the message $\mu_l(y_{\to l})$ by

$$\sum_{y_l} \Delta(y_l, y_{\to l}) \psi_{\lambda,\boldsymbol{\gamma}}(y_l, y_{\to l}) \begin{cases} \psi_{\boldsymbol{\eta}}(y_l, \mathbf{x}_l) & \text{, if } l \leq n \\ \prod\limits_{v_{\to j}=v_l} \mu_j(y_l) & \text{, otherwise.} \end{cases}$$

Hence, Equation 19 can be evaluated by applying standard message passing $n+k$ times which can be done in time $\mathcal{O}(|\mathcal{Y}|^2(n+k)^2)$. This completes the proof for the computational time of the expectation $\mathbb{E}_{\mathbf{y},\bar{\mathbf{y}}}\left[\Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right]$.

The proof for the expectation $\mathbb{E}_{\bar{\mathbf{y}}}\left[\Phi(\mathbf{X}, \mathbf{y}, \bar{\mathbf{y}})\right]$ can be done analogously by replacing the sum over $\mathbf{y}$ with the true labels. □

References

1. Baykan, E., Henzinger, M., Weber, I.: Web page language identification based on URLs. Proceedings of the VLDB Endowment 1(1), 176–187 (2008)
2. Huynh, T., Mooney, R.: Max-margin weight learning for Markov logic networks. In: Proceedings of the European Conference on Machine Learning (2009)
3. Kan, M., Thi, H.: Fast webpage classification using URL features. In: Proceedings of the 14th ACM international conference on information and knowledge management (2005)
4. Koller, D., Friedman, N.: Probabilistic Graphical Models, vol. 1. MIT Press (2009)
5. Kumar, R., Punera, K., Tomkins, A.: Hierarchical topic segmentation of websites. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining (2006)
6. McDowell, L., Gupta, K., Aha, D.: Cautious inference in collective classification. In: Proceedings of the 22nd AAAI Conference on Artificial Intelligence (2007)
7. Pearl, J.: Fusion, propagation, and structuring in belief networks. Artificial intelligence 29(3), 241–288 (1986)
8. Qi, X., Davidson, B.: Web page classification: Features and algorithms. ACM Computing Surveys 41(2), 12:1–12:31 (2009)
9. Shen, D., Sun, J., Yang, Q., Chen, Z.: A comparison of implicit and explicit links for web page classification. In: Proceedings of the 15th international World Wide Web conference (2006)
10. Shih, L., Karger, D.: Using URLs and table layout for web classification tasks. In: Proceedings of the 13th World Wide Web conference (2004)
11. Taskar, B., Guestrin, C., Koller, D.: Max-margin Markov models. In: Proceedings of the 17th Annual Conference on Neural Information Processing Systems (2004)
12. Tian, Y., Huang, T., Gao, W.: Two-phase web site classification based on hidden Markov tree models. Web Intelligence and Agent Systems 2(4), 249–264 (2004)
13. Wang, S., Quattoni, A., Morency, L., Demirdjian, D., Darrell, T.: Hidden conditional random fields for gesture recognition. In: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2006)
14. Yu, C.N.J., Joachims, T.: Learning structural SVMs with latent variables. In: Proceedings of the 26th Annual International Conference on Machine Learning. pp. 1169–1176. ACM (2009)
15. Yuille, A.L., Rangarajan, A.: The concave-convex procedure. Neural Computation 15(4), 915–936 (2003)