Conditional Topical Coding: an Efficient Topic Model Conditioned on Rich Features

Jun Zhu[†] Ni Lao[†] Ning Chen[‡] Eric P. Xing[†]
[†]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213
[‡]Department of Computer Science and Technology, Tsinghua University, Beijing, China 100084
{junzhu, nlao, epxing}@cs.cmu.edu; chenn07@mails.thu.edu.cn

ABSTRACT

Probabilistic topic models have shown remarkable success in many application domains. However, a probabilistic conditional topic model can be extremely inefficient when considering a rich set of features because it needs to define a normalized distribution, which usually involves a hard-to-compute partition function. This paper presents conditional topical coding (CTC), a novel formulation of conditional topic models which is non-probabilistic. CTC relaxes the normalization constraints as in probabilistic models and learns non-negative document codes and word codes. CTC does not need to define a normalized distribution and can efficiently incorporate a rich set of features for improved topic discovery and prediction tasks. Moreover, CTC can directly control the sparsity of inferred representations by using appropriate regularization. We develop an efficient and easy-to-implement coordinate descent learning algorithm, of which each coding substep has a closed-form solution. Finally, we demonstrate the advantages of CTC on online review analysis datasets. Our results show that conditional topical coding can achieve state-of-the-art prediction performance and is much more efficient in training (one order of magnitude faster) and testing (two orders of magnitude faster) than probabilistic conditional topic models.

Categories and Subject Descriptors

I.5.1 [**Pattern Recognition**]: Models - Statistical

General Terms

Algorithms, Experimentation

Keywords

sparse coding, topic models, conditional models

1. INTRODUCTION

Probabilistic topic models, such as latent Dirichlet allocation (LDA) [4] and probabilistic latent semantic indexing (pLSI) [12], have been widely used for inferring a low dimensional representation that captures the latent semantics of text documents or images. Such low dimensional representations can be used for classifying, clustering, or structurally browsing large corpora. However, most existing topic models only consider document-word appearance (e.g., bag-of-word counts) and they are essentially not "feature-based" models. As discussed in [41], using the word information itself could be insufficient in resolving the word's meaning ambiguity. For example, in a hotel review, the word "friendly" can be used to describe a positive aspect, while it can also be used to describe a negative aspect when used with a denying word (e.g., "not"), such as in the sentence "the front desk was not very friendly". Another example is that documents are usually associated with meta-data (e.g., authors, dates and publication venues), which could serve as a rich information source for inferring semantically meaningful topical patterns [26]. Therefore, exploring a rich set of input features (e.g., non-local contextual or summary features in an article or image) can be expected to yield better models in terms of their discovered latent topics [41, 26, 29] and performance on prediction tasks (e.g., regression [41]). One can also find convincing arguments of instead preferring a feature-based model from the celebrated work on conditional random fields (CRF) [21] and their various applications on information extraction [30], gene prediction [8] and etc. However, it is non-trivial to incorporate rich features in probabilistic topic models. One reason is that a fully generative topic model specifies a joint distribution of all the introduced variables, which prevents flexible incorporation of nontrivial features in the data, because directly modeling such features as random variables would result in a prohibitively large state space that makes inference and learning very difficult, if at all possible. Recent progress has been made on developing conditional topic models [41], which can in principle incorporate arbitrary features to discover meaningful topical representations. However, due to its probabilistic nature, such a conditional topic model involves a normalization factor when defining the topic-assignment distribution given input features (please see Section 2), which can make inference and learning extremely hard in latent variable models. Various approximation techniques has been developed [41, 37, 19], but they are usually very inefficient.

Another limitation of probabilistic topic models is that they lack a mechanism to directly control the sparsity of the inferred posterior representations. Although using a sparse prior can indirectly influence the posterior sparsity, an arguably better way is to directly impose posterior regular-

KDD'11, August 21–24, 2011, San Diego, California, USA.

ization (e.g., using moment constraints [11] or entropic priors [31]). However, due to the smoothness of the entropic regularizer, such methods often do not yield truly sparse posterior representations in practice. A technical reason for the difficulty in achieving sparsity in probabilistic topic models (e.g., LDA) is that the mixing proportions or topics take the form as a normalized distribution. Thus it is unhelpful to directly use a sparsity inducing ℓ_1-regularizer as in lasso [35, 25] and sparse coding [28].

In this paper, we present conditional topical coding (CTC), a novel non-probabilistic formulation of conditional topic models. CTC is not subject to the strict normalization constraints as in probabilistic topic models and we show that it can efficiently incorporate a rich set of features for improved topic discovery and better prediction performance. Moreover, the non-probabilistic formulation can enforce a direct control on the sparsity of the inferred representations by using appropriate regularizers which have been widely studied in sparse coding [28] and lasso [35, 25] methods. In addition, the non-probabilistic CTC can be seamlessly integrated with a convex prediction loss measure (e.g., ϵ-insensitive loss [33] for learning a large-margin regression model) in order to incorporate the widely available supervised side information, such as review rating scores, for discovering predictive representations. We develop an efficient and easy-to-implement coordinate descent learning algorithm, of which each coding substep has a closed-form solution, and the feature weights and dictionary learning can be efficiently done with high-performance techniques. Finally, we extensively evaluate CTC on online review analysis data. Our results show that CTC can achieve state-of-the-art prediction performance and is much more efficient in training (one order of magnitude) and testing (two orders of magnitude faster) than probabilistic conditional topic models.

Related work: CTC represents a new extension of sparse coding (SPC) [28], which provides an elegant framework to achieve sparsity on the usually unnormalized code vectors or dictionary by using the theoretically sound ℓ_1-norm or other composite regularizers [20, 17, 16, 2]. Although much work has been done on learning a structured dictionary [17, 2], existing SPC methods typically discover flat representations, such as single-layer sparse codes of small image patches or word terms [17, 2]. In order to achieve a representation of an entire image or document, a post-processing such as average or max pooling [39] is needed. This two-step procedure can be rather sub-optimal because it lacks a channel to provide direct correlations between individual component representations [15], or to leverage the possibly available high-level weak supervision (e.g., document categories) to discover predictive representations [40] or learn a supervised dictionary [24].

The rest of the paper is structured as follows. Section 2 reviews probabilistic conditional topic models. Section 3 presents conditional topical coding together with a learning algorithm. Section 5 presents empirical studies and Section 6 concludes with future directions discussed.

2. PROBABILISTIC CONDITIONAL TOPIC MODELS

We briefly review the probabilistic conditional topic model (CdTM) [41] without Markov dependency among different topic assignments and motivate the development of CTC.

Figure 1 (a) illustrates the graphical structure of CdTM, where we have ignored the prior of θ for clarity. Let N be the number of terms in a vocabulary $V = \{1, \cdots, N\}$ and let β be a topical matrix, of which each row β_k is a unigram distribution over the terms in V, that is, $\beta_k \in \mathcal{P}_N$, where $\mathcal{P}_N$ is a $(N-1)$-simplex. In CdTM and standard LDA, a document is represented as a *sequence* of words $\mathbf{w} = (w_1, \cdots, w_M)$, where M is document length and w_m is the word that appears in position m of the document. Each position m is associated with a topic assignment variable Z_m, and the topics of all the words in a document are sampled from the same document-specific distribution, which is the mixing proportion θ in standard LDA. In CdTM, in order to incorporate rich features, which are denoted by A (an instance denoted by $\mathbf{a}$, e.g., POS tags or contextual features), the topic-assignment distribution is defined as a softmax function

$$p(z_m|\theta, \mathbf{a}) = \frac{1}{B(\theta)} \exp(\theta^\top \mathbf{f}(z_m, \mathbf{a})), \quad (1)$$

where $\mathbf{f}$ is a vector of feature functions and θ is the vector of weights, which follows a prior distribution (e.g., normal prior). The normalization factor (or partition function) $B(\theta) = \sum_z \exp(\theta^\top \mathbf{f}(z, \mathbf{a}))$ is a *sum-exp* function, whose logarithm is also known as a *log-sum-exp* function [19].

Although the conditional model (1) can effectively consider rich features, the inference and learning of the resulting conditional topic model is usually hard because of the sum-exp function $B(\theta)$ especially when θ is a latent variable as in CdTM. Much research has been conducted to obtain a good approximate inference method, such as [37, 41, 19], but these methods are usually very inefficient.

Below, we present a novel non-probabilistic formulation of conditional topic models, which can efficiently incorporate a rich set of features to achieve improved topic representations and prediction performance. Another potential advantage of the non-probabilistic formulation is that it can explicitly control the sparsity of the learned representations. In contrast, a probabilistic topic model in general can be very inflexible to explicitly control the sparsity of inferred latent representations, as we have discussed in Section 1.

3. CONDITIONAL TOPICAL CODING

For clarity, we first consider the simplified CTC model without using features, which will be called sparse topical coding (STC) [42]. Slightly different from LDA and CdTM, we represent a document as a vector $\mathbf{w} = (w_1, \cdots, w_{|I|})^\top$, where I is the index set of words that appear and each w_n ($n \in I$) represents the number of times that word n appears in this document. Like other topic models, we present STC as a technique to project the input $\mathbf{w}$ into a semantic latent space that is spanned by a set of automatically learned bases (a basis set is also called a *dictionary* [17]) and achieve a high-level representation of the entire document. To remove the strict normalization constraints, we formulate STC as regularized loss minimization [35, 25, 28]. However, purely for the ease of understanding, we start with describing a probabilistic generative procedure.

3.1 A Probabilistic Generative Process

Let β denote a dictionary with K bases. As in LDA, we assume that each row β_k is a topic, i.e., a unigram distribution over the terms in V. We will use $\beta_{\cdot n}$ to denote the nth column of β. Graphically, STC is a hierarchical latent vari-

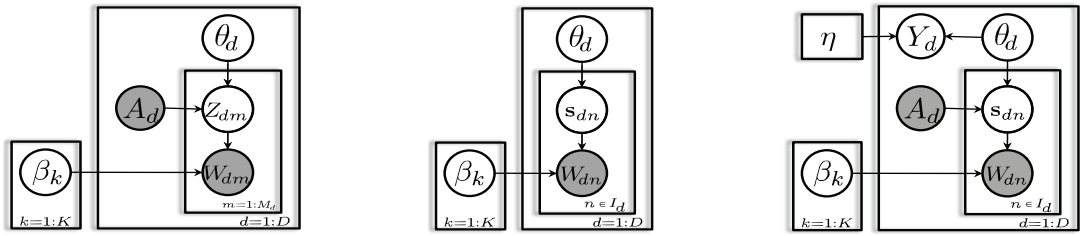


Figure 1: Graphical representations for (a) probabilistic conditional topic models [41]; (b) sparse topical coding; and (c) conditional topical coding with supervised information Y. We omit the priors on θ and β.

able model, as shown in Figure 1 (b), where $\theta \in \mathbb{R}^K$ is the latent representation of an entire document and $\mathbf{s}_n \in \mathbb{R}^K$ is a latent representation of the individual word n. We call $\mathbf{s}_n$ *word code* and θ *document code*. Then, a document can be described as arising from the following process

1. sample the document code θ from a prior $p(\theta)$.
2. for each observed word $n \in I$
   (a) sample the word code $\mathbf{s}_n$ from a conditional distribution $p(\mathbf{s}_n|\theta)$
   (b) sample the observed word count w_n from a distribution with the mean being $\mathbf{s}_n^\top \beta_{\cdot n}$.

Since we have relaxed the codes θ and $\mathbf{s}$ to be real-valued vectors, we need to define a new distribution for generating word counts. Here, we adopt the ideas of sparse coding [28]. Specifically, we treat $\mathbf{s}_n$ as a coefficient vector and use the linear combination $\mathbf{s}_n^\top \beta_{\cdot n}$ to reconstruct the observed word count w_n, under some loss measure as explained below; and the document code θ is obtained via an aggregation (e.g., truncated average) of the individual codes of all its terms. To specify a STC model, the choices of $p(\theta)$ and $p(\mathbf{s}|\theta)$ reflect our bias on the discovered latent representations and how θ and $\mathbf{s}$ are connected. We will discuss them in the next section. For the last step of generating observed data, we adopt the broad class of exponential family distributions to make STC applicable to rich forms of data. Formally, we use the linear combination $\mathbf{s}_n^\top \beta_{\cdot n}$ as the *mean* parameter of an exponential family distribution that generates the observation w_n. In other words, we define $p(w_n|\mathbf{s}_n, \beta)$ as an exponential family distribution that satisfies

$$\mathbb{E}_{p(w_n|\mathbf{s}_n,\beta)}[T(w_n)] = \mathbf{s}_n^\top \beta_{\cdot n}, \tag{2}$$

where $T(w_n)$ are sufficient statistics[1] of w_n. We choose to use the linear combination as *mean* parameter instead of *natural* parameter [23] because: 1) it is natural to constrain the feasible domain of codes for good interpretation, as detailed below. As shown in [22], imposing appropriate (e.g., non-negativity for modeling word counts) constraints could result in sparser and more interpretable patterns; and 2) many distributions (e.g., Poisson and Gaussian) are commonly expressed with mean parameters. [6] uses a similar method as ours to define exponential family distributions.

3.2 STC for Sparse MAP Estimation

The generating process defines a joint distribution of codes and word counts $p(\theta, \mathbf{s}, \mathbf{w}|\beta) = p(\theta)\prod_{n\in I} p(\mathbf{s}_n|\theta)p(w_n|\mathbf{s}_n, \beta)$. By imposing a prior on β, we define STC as finding the MAP estimate on a given training set $\{\mathbf{w}_d\}$, that is, STC solves

$$\max_{\Theta} \sum_{d,n\in I_d} (\log p(w_{dn}|\mathbf{s}_{dn}, \beta) + \log p(\mathbf{s}_{dn}|\theta_d)) + \sum_d \log p(\theta_d) + \log p(\beta)$$
$$\text{s.t.} : \theta_d \geq 0,\ \forall d;\ \mathbf{s}_{dn} \geq 0,\ \forall d, n \in I_d;\quad \beta_k \in \mathcal{P}_N,\ \forall k,$$

where we have used Θ to denote the parameters $(\{\theta_d\}, \{\mathbf{s}_d\}, \beta)$. Although we can have many choices, we concentrate on the following instantiation of STC for modeling word counts:

1. We define the Poisson distribution[2], i.e., $p(w_n|\mathbf{s}_n, \beta) = Poisson(w_n; \mathbf{s}_n^\top \beta_{\cdot n})$, where $Poisson(x; \nu) = \frac{\nu^x e^{-\nu}}{x!}$.
2. We choose the Laplace prior $p(\theta) \propto \exp(-\lambda\|\theta\|_1)$, and we define $p(\mathbf{s}_n|\theta)$ as a composite distribution $p(\mathbf{s}_n|\theta) \propto \exp(-\frac{\gamma}{2}\|\mathbf{s}_n - \theta\|_2^2 - \rho\|\mathbf{s}_n\|_1)$, which is supergaussian [14]. The ℓ_1-norm will bias toward finding sparse codes.
3. We use a uniform prior of β.

The hyper-parameters (λ, γ, ρ) are non-negative and they can be selected via cross-validation or integrated out by introducing hyper-priors [7, 10]. Under such an instantiation, sparse topical coding solves the equivalent problem

$$\min_{\Theta} \sum_{d,n\in I_d} \ell(w_{dn}, \mathbf{s}_{dn}^\top \beta_{\cdot n}) + \lambda \sum_d \|\theta_d\|_1 + \sum_{d,n\in I_d} (\frac{\gamma}{2}\|\mathbf{s}_{dn} - \theta_d\|_2^2 + \rho\|\mathbf{s}_{dn}\|_1)$$
$$\text{s.t.} : \theta_d \geq 0,\ \forall d;\ \mathbf{s}_{dn} \geq 0,\ \forall d, n \in I_d;\quad \beta_k \in \mathcal{P}_N,\ \forall k, \tag{3}$$

where $\ell(w_n, \mathbf{s}_n^\top \beta_{\cdot n}) = -\log Poisson(w_n; \mathbf{s}_n^\top \beta_{\cdot n})$ is a loss function. Minimizing the log-Poisson loss is actually equivalent to minimizing an unnormalized KL-divergence between observed word counts w_n and their reconstructions $\mathbf{s}_n^\top \beta_{\cdot n}$ [34]. Since word counts are non-negative, a negative θ or $\mathbf{s}$ will lose interpretability. Therefore, we constrain θ and $\mathbf{s}$ to be non-negative as in [13]. A non-negative code can be interpreted as representing the relative importance of topics.

3.3 Conditional Topical Coding

Now, we formally present CTC, which is a novel non-probabilistic formulation of conditional topic models and is very efficient in training and inference, as we shall see. A conditional topical coding (CTC) is graphically illustrated in Figure 1 (c). To be consistent with the above generating process, we define the conditional probability

$$p(\mathbf{s}_n|\theta, \mathbf{U}, \mathbf{a}) \propto \exp\big(-\frac{\gamma}{2}(\|\mathbf{s}_n - \theta\|_2^2 + \|\mathbf{s}_n - \mathbf{U}\mathbf{f}(\mathbf{a})\|_2^2) - \rho\|\mathbf{s}_n\|_1\big),$$

where $\mathbf{f}(\mathbf{a})$ is a L-dimensional vector of real-valued feature functions; $\mathbf{U}$ is a $K \times L$ weight matrix; and $\mathbf{a}$ is again a set of

[1] In general, $\mathbf{s}_n$ will be a matrix if T is a vector.

[2] It is usually numerically safer to introduce a small offset $b \in \mathbb{R}_+$ and define $p(w_n|\mathbf{s}_n, \beta) = Poisson(w_n; \mathbf{s}_n^\top \beta_{\cdot n} + b)$ to avoid taking the logarithm of zero. In all our experiments, we fix b at a very small positive value.

global or local features. We have omitted the normalization factor. Again, $p(\mathbf{s}_n|\theta, \mathbf{U}, \mathbf{a})$ is supergaussian [14]. We define CTC as solving the constrained problem

$$\min_{\Theta,\mathbf{U}} \sum_{d,n\in I_d} \ell(w_{dn}, \mathbf{s}_{dn}^\top \beta_{\cdot n}) + \lambda \sum_d \|\theta_d\|_1 + \sum_{d,n\in I_d} \big(\frac{\gamma}{2}\|\mathbf{s}_{dn} - \theta_d\|_2^2 + \frac{\gamma}{2}\|\mathbf{s}_{dn} - \mathbf{U}\mathbf{f}(\mathbf{a}_d)\|_2^2 + \rho\|\mathbf{s}_{dn}\|_1\big)$$
$$\text{s.t.}: \ \theta_d \geq 0, \ \forall d; \ \mathbf{s}_{dn} \geq 0, \ \forall d, n \in I_d; \quad \beta_k \in \mathcal{P}_N, \ \forall k, \qquad (4)$$

which is related to but not exactly the MAP estimation because we have ignored the normalization factor of $p(\mathbf{s}_n|\theta, \mathbf{U}, \mathbf{a})$. Note that if we include the normalization factor, the non-probabilistic CTC can still be efficient (e.g., using gradient descent techniques) because we do not need to deal with the expectation of a *log-sum-exp* function, which is usually hard to compute in latent variable models [19, 41].

Supervised side information: we have presented conditional topical coding as an *unsupervised* technique that discovers latent representations. However, with the increasing availability of free on-line information, various forms of side-information such as rating scores for hotel reviews on TripAdvisor[3] and object categories for images in the LabelMe dataset[4] can potentially offer "free" supervision. This has led to a need for new models and training schemes that can make effective use of such information to achieve better results, such as more discriminative representations of documents and more accurate image classifiers. In order to incorporate such side information, we consider the general case where supervised information Y is provided in training data $\mathcal{D} = \{(\mathbf{w}_d, y_d)\}_{d=1}^D$, and we define the supervised CTC as solving the following problem for learning a prediction model η and a conditional topic discovery model $(\Theta, \mathbf{U})$

$$\min_{\Theta,\mathbf{U},\eta} \ h(\Theta, \mathbf{U}) + C\mathcal{R}_\mathcal{D}(\{\theta_d\}, \eta) + \frac{1}{2}\|\eta\|_2^2$$
$$\text{s.t.}: \ \theta_d \geq 0, \ \forall d; \ \mathbf{s}_{dn} \geq 0, \ \forall d, n \in I_d; \quad \beta_k \in \mathcal{P}_N, \ \forall k, \qquad (5)$$

where $h(\Theta, \mathbf{U})$ is the objective function of problem (4) and $\mathcal{R}_\mathcal{D}$ is an error measure of the prediction model η on the training data $\mathcal{D}$. Since CTC is non-probabilistic, we can seamlessly incorporate any convex loss that does not necessarily arise from a probabilistic model and thus can avoid dealing with normalization factors. Here, we consider the linear regression model (i.e., the prediction $y^* = \eta^\top \theta$) and use the ϵ-insensitive loss [33] as used in support vector regression (SVR) for large-margin learning, that is, $\mathcal{R}_\mathcal{D} = \frac{1}{D}\sum_d \max(0, |\eta^\top \theta_d - y_d| - \epsilon)$, where ϵ is a small positive precision parameter. For classification, we can use the SVM hinge loss [42], which again avoids dealing with a normalization factor that can make inference and learning hard in a probabilistic topic model (please see [37, 40] for discussions).

3.4 Optimization with Coordinate Descent

Now, we briefly present how to solve the most general problem (5), which subsumes problem (3) and problem (4). Let $f(\Theta, \mathbf{U}, \eta)$ be its objective. When using log-loss under the exponential family of distributions, $f(\Theta, \mathbf{U}, \eta)$ is bi-convex, i.e., convex over either $(\{\theta_d\}, \{\mathbf{s}_d\})$ or $(\beta, \mathbf{U}, \eta)$ when fixing the other one. Moreover, the feasible set is a convex set. A natural algorithm to solve this bi-convex problem is coordinate descent, as typically used in sparse coding methods [23, 2]. Our result relies on the following proposition

Proposition 1. *Let $\psi(x)$ be a convex function on $\mathbb{R}$. If x_0 is a solution of the unconstrained problem $P_0: \ \min_x \psi(x)$, then $x^\star = \max(0, x_0)$ is an optimal solution of the constrained problem $P_1: \ \min_{x\geq 0} \psi(x)$.*

Proof: [Sketch] If $x_0 \geq 0$, then by definition, $x^\star$ is a solution of P_1; otherwise (i.e., $x_0 < 0$ and $x^\star = 0$), let's assume $x^\star$ is not a solution of P_1. Let x_1 be any one solution of P_1 and define $\alpha = \frac{x_1}{x_1 - x_0}$. Then, we have $x_1 > 0$, $0 < \alpha < 1$ and $0 = \alpha x_0 + (1-\alpha)x_1$. By definition, we have $\psi(x_0) \leq \psi(0)$ and $\psi(x_1) < \psi(0)$. Therefore, we can get $\alpha\psi(x_0) + (1-\alpha)\psi(x_1) < \psi(0) = \psi(\alpha x_0 + (1-\alpha)x_1)$. This contradicts the convexity of $\psi(x)$. □

Then, the procedure alternatively performs *hierarchical topical coding*, *dictionary learning*, *prediction model learning* and *feature weights learning* until convergence, as outlined in Algorithm 1 and detailed below.

Algorithm 1 for learning supervised CTC

1: **Input:** corpus $\mathcal{D} = \{(\mathbf{w}_d, \mathbf{a}_d, y_d)\}_{d=1}^D$, regularization constants $(\lambda, \gamma, \rho, C)$ and precision parameter ϵ.
2: **Output:** codes $\{\theta_d\}_{d=1}^D$ and $\{\mathbf{s}_d\}_{d=1}^D$, dictionary β, feature weights $\mathbf{U}$ and prediction model η
3: **repeat**
4: hierarchical topical coding using Alg. 2.
5: dictionary learning by solving problem (8)
6: learning prediction model η using SVMLight
7: learning feature weights $\mathbf{U}$ using gradient descent
8: **until** convergence

Hierarchical topical coding: this step solves for the optimal codes $\{\theta_d\}$ and $\{\mathbf{s}_d\}$ when $(\beta, \mathbf{U}, \eta)$ are fixed. Since documents are not coupled, we can perform this step for each document separately by solving the ℓ_1-regularized problem

$$\min_{\theta,\mathbf{s}} \sum_{n\in I} \ell(w_n, \mathbf{s}_n^\top \beta_{\cdot n}) + \lambda\|\theta\|_1 + \sum_{n\in I} \frac{\gamma}{2}(\|\mathbf{s}_n - \theta\|_2^2 + \|\mathbf{s}_n - \mathbf{U}\mathbf{f}(\mathbf{a})\|_2^2) + \rho\|\mathbf{s}_n\|_1))$$
$$\text{s.t.}: \ \theta \geq 0; \quad \mathbf{s}_n \geq 0, \ \forall n \in I.$$

While previous work used either local quadratic approximation [23] or a specialized Poisson likelihood estimation method [34], we solve this problem with coordinate descent (outlined in Alg. 2), which has a closed-form at each substep. Moreover, our method have a simple structure, which is similar to the variational EM method for probabilistic LDA, as we shall see. Specifically, we alternatively solve:

Optimize over $\mathbf{s}$: when θ is fixed, $\mathbf{s}_n$ are not coupled. For each $\mathbf{s}_n$, we solve the convex problem

$$\min_{\mathbf{s}_n \geq 0} \ell(w_n, \mathbf{s}_n^\top \beta_{\cdot n}) + \frac{\gamma}{2}(\|\mathbf{s}_n - \theta\|_2^2 + \|\mathbf{s}_n - \mathbf{U}\mathbf{f}(\mathbf{a})\|_2^2) + \rho\sum_k s_{nk}$$

where we have explicitly written the ℓ_1-norm of $\mathbf{s}_n$ under the non-negativity constraint. Let $g(\mathbf{s}_n)$ be the objective function. By Proposition 1, the solution of component k is

$$s_{nk} = \max(0, \nu_k), \qquad (6)$$

where ν_k is the solution of minimizing $g(\mathbf{s}_n)$ over s_{nk} without the constraint. By setting the gradient $\nabla_{s_{nk}} g(\mathbf{s}_n)$ equal to zero, it is easy to derive that ν_k is the solution of the equation

$$\gamma\beta_{kn}\nu_k^2 + (\gamma\mu + \beta_{kn}\tau)\nu_k + \mu\tau - w_n\beta_{kn} = 0,$$

[3] http://www.tripadvisor.com/
[4] http://labelme.csail.mit.edu/

Algorithm 2 for hierarchical topical coding

Input: corpus $\mathcal{D}$, regularization constants (λ, γ, ρ), dictionary β, feature weights $\mathbf{U}$ and prediction model η.
Output: codes $\{\theta_d\}_{d=1}^D$ and $\{\mathbf{s}_d\}_{d=1}^D$
repeat
  for $d = 1$ **to** D **do**
    for each word $n \in I_d$ **do**
      Update word code $\mathbf{s}_{dn}$ using Eq. (6).
    end for
    Update document code θ_d using Eq. (7).
  end for
until convergence

where $\mu = \sum_{j \neq k} s_{nj}\beta_{jn}$ and $\tau = \beta_{kn} + \rho - \gamma(\theta_k + \mathbf{U}_k\mathbf{f}(\mathbf{a}))$. Therefore, if $\beta_{kn} = 0$, the solution is $\nu_k = (\theta_k + \mathbf{U}_k\mathbf{f}(\mathbf{a})) - \frac{\rho}{\gamma}$. Otherwise (i.e., $\beta_{kn} > 0$), we need to solve a quadratic equation, which again always has real solutions because the discriminant $\nabla \triangleq (\gamma\mu + \beta_{kn}\tau)^2 - 4(\gamma\beta_{kn})(\mu\tau - w_n\beta_{kn}) = (\gamma\mu - \beta_{kn}\tau)^2 + 4\gamma w_n\beta_{kn}^2$ is guaranteed to be positive. We choose ν_k to be the larger one of the two possible solutions.

Optimize over θ: when $\mathbf{s}$ is fixed, this step solves

$$\min_{\theta \geq 0} \ \lambda\|\theta\|_1 + \frac{\gamma}{2}\sum_{n \in I}\|\mathbf{s}_n - \theta\|_2^2 + \frac{C}{D}\max(0, |\eta^\top\theta - y| - \epsilon).$$

We alternatively solve for each θ_k. By Proposition 1, we first solve an unconstrained one dimensional problem and then use the max operator to get the optimum solution. It is easy to show that one of the subgradient of this objective (with the ℓ_1-norm explicitly written as $\sum_k \theta_k$) is $\lambda + \gamma\sum_{n \in I}(\theta_k - s_{nk}) + \frac{C}{D}\mathbb{I}(|\eta^\top\theta - y| > \epsilon)Sign(\eta^\top\theta - y)\eta_k$. By setting the subgradient equal to zero, we have one solution

$$\forall k, \ \theta_k = \max(0, \bar{s}_k - \frac{\lambda}{\gamma|I|}), \tag{7}$$

where $\bar{s}_k = \frac{1}{|I|}\sum_{n \in I} s_{nk} - \frac{C}{D|I|}\mathbb{I}(|\eta^\top\theta - y| > \epsilon)Sign(\eta^\top\theta - y)\eta_k$ and $\mathbb{I}(\cdot)$ is an indicator function.

Dictionary learning: after we have inferred the latent codes $\{\theta_d\}$ and $\{\mathbf{s}_d\}$, we update the dictionary β by solving

$$\min_{\beta} \sum_{d,n \in I_d} \ell(w_n, \mathbf{s}_{dn}^\top\beta_{\cdot n}), \ \text{s.t.}: \ \beta_k \in \mathcal{P}_N, \ \forall k, \tag{8}$$

which can be efficiently done with projected gradient descent. After each step of gradient descent, a projection to the simplex $\mathcal{P}_N$ is performed with a linear algorithm [9].

Learning prediction model η: when the document codes are given, we solve a standard support vector regression (SVR) problem [33], which can be efficiently done with a high-performance package, e.g., SVMLight[5].

Learning feature weights U: in practice, we need a regularization term to avoid over-fitting. Here, we constrain the ℓ_2-norm of each row to be less than a constant (e.g., 16 used in our experiments). Then, we solve for $\mathbf{U}$ using projected gradient descent.

From the update rule (7), we can see the regularization effects introduced by considering supervised side information. Specifically, if the current prediction $y^* = \eta^\top\theta$ differs much from the truth y (e.g., ℓ_1-distance is larger than ϵ), then the last term of $\bar{s}_k$ will be non-zero. Moreover, if the prediction y^* is larger than y, the last term will be of the same sign as η_k, which means the new θ will tend to be biased toward yielding a smaller new prediction. We have similar bias effects when the current prediction y^* is smaller than y. Thus we can expect to discover predictive document codes θ by considering supervised information, which usually leads to improved prediction performance as we shall see.

[5]http://svmlight.joachims.org/

3.5 Comparison with Probabilistic CdTM

Table 1 summarizes the difference between CTC and probabilistic CdTM [41]. Briefly, CdTM doesn't explicitly define word and document code. An equivalence to word code can be defined as the *empirical* word-topic assignment distribution $\tilde{p}(z(n) = k) \propto \sum_m \mathbb{I}(w_m = n)p(z_{mk} = 1|\mathbf{w})$, where $z(n)$ is the topic of word n, which needs to be inferred using variational methods [41]. Then, an equivalence to document code is the average aggregation of *empirical* word-topic assignment distributions. Second, CdTM or a probabilistic topic model in general lacks an explicit sparcification procedure on the inferred representations as discussed in Section 1.

4. EXPERIMENTS

In this section, we present empirical studies on online review data. We report quantitative evaluation on rating score prediction and time efficiency, as well as qualitative analysis of the discovered representations. Our results demonstrate that the non-probabilistic CTC can achieve state-of-the-art prediction performance and is much more efficient than probabilistic topic models. All our datasets and code are available at http://www.cs.cmu.edu/∼junzhu/ctc.htm. As we have stated, STC and CTC could explicitly control the sparsity of latent representations using appropriate regularization. But a systematical analysis of the sparsity is beyond the scope of this paper. Please see the companion paper [42] for details.

4.1 Data and Features

In order to evaluate the effects of features, we use online review data, in which the documents are hotel reviews downloaded from the TripAdvisor website in 2009. Each review is associated with a global rating score and five aspect rating scores for *Value*, *Rooms*, *Location*, *Cleanliness*, and *Service*. This dataset is interesting for many data mining tasks, for example, extracting the textual mentions of each aspect [36, 18], using the guidance of side information to discover semantic information [5], or discovering latent rating aspects [38]. In these experiments, we focus on predicting the global rating score, which ranks from 1 to 5, and revealing some underlying structures.

Besides the small dataset [41], which contains 1000 reviews for each rating category or 5000 reviews in total, we also build a new dataset for evaluating the scalability of CTC, which contains 97,948 reviews in total (about 20 times larger than the small one). To avoid too short and too long reviews, we only keep those reviews whose character length is between 1000 and 6000. For each review, we use NLProcessor[6] to do part-of-speech (POS) tagging and noun phrase (NP) chunking, and we extract the following features:

. **POS-Tag**: We distinguish four types of POS tags, that is, Adjective, Noun, Adverb, and Verb. Each type includes all its subcategories, e.g., Adjective includes "JJ" (Adjective), "JJR" (comparative Adjective), and "JJS" (superlative Adjective).

. **WordNet**: WordNet[7] is a large lexical database of English.

[6]http://www.infogistics.com/textanalysis.html
[7]http://wordnet.princeton.edu/

Table 1: Comparison between CTC and probabilistic conditional topic model (CdTM) [41]

	CTC	CdTM
formulation	non-probabilistic	probabilistic
document code	$\theta \in \mathbb{R}_+^K$	no explicit definition, need to be inferred
word code	$\mathbf{s} \in \mathbb{R}_+^K$	no explicit definition, need to be inferred
dictionary	$\forall k: \ \beta_k \in \mathcal{P}_N$	$\forall k: \ \beta_k \in \mathcal{P}_N$
estimation	regularized loss minimization	(empirical) Bayesian inference
algorithm	coordinate descent	mean field dealing with expectation of log-sum-exp function [41, 19]
sparsity	direct control by using regularizers	indirect control by using sparse priors

We navigate it with some seeds of positive (e.g., good, excellent, etc) and negative (e.g., bad, painful, etc) words, and identify whether a word is positive or negative based on the synonym and antonym relationship. Words without strong relationship with the seeds are treated as neutral. For a positive or negative word, we also identify whether a denying word (e.g., not, no, etc.) appears before it within a word distance of 4.

. **NP-Chunking**: We define pairwise feature functions of conditional topical random fields (CTRF) [41] for those words that are in the same noun or verb phrase, or the conjunction "and" or "or" appears between them.

By removing a standard list of stopping words and those terms whose count frequency is less than 5, we build a dictionary with 12000 terms.

4.2 Results on the Small Dataset

We first carefully investigate the properties of CTC on the small dataset, on which a number of benchmark methods have been evaluated in [41]. For saving space, we compare the supervised CTC with other supervised topic models, including supervised CTRF [41] and CdTM [41], MedLDA [40], supervised LDA (sLDA) [3], and supervised STC (i.e., supervised CTC with none features), which usually outperform their unsupervised counterpart models as shown in various previous work [3, 37, 40, 41]. We follow the same setting as in [41] with a uniform train/test partition of the data. For hyperparameters, we set $\rho = 5e^{-4}$, $\epsilon = 1e^{-3}$ and $\lambda = \gamma$; and search for C and λ via cross-validation on training data.

4.2.1 Prediction Performance

Similar as in [3, 41], we take logarithm to make the response variables approximately normal and treat the problem of predicting review rating scores as a regression problem. Figure 2 shows the predictive R2 scores [3] for different models. We can see that conditional topical coding (CTC) achieves state-of-the-art prediction performance, which is comparable with the best performance of the conditional CTRF and a bit better than that of CdTM, which is a simplified CTRF without modeling the Markov dependency among topic assignments of neighboring words. As we shall see, the probabilistic CTRF and CdTM are much slower in training and testing than CTC. The reasons for this outstanding performance potentially come from three aspects. First, using the features (e.g., CTC and CdTM) can significantly improve the performance compared to the models that do not use features (e.g., STC and MedLDA). Second, the non-probabilistic formulation (e.g., STC) of topic models can potentially discover representations that have a better predictive power than the probabilistic formulation (e.g., MedLDA and sLDA). We will provide some insights about the properties of the discovered representations. Finally, the large-margin principle could potentially improve the performance, e.g., the large-margin based MedLDA achieves

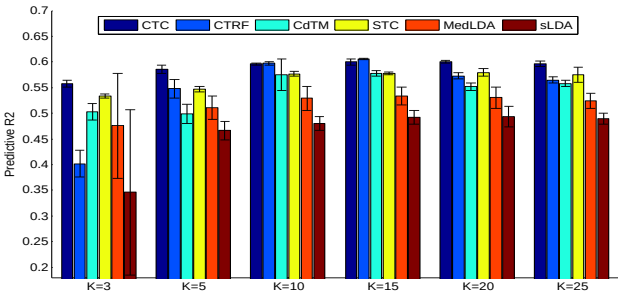


Figure 2: Predictive R2 values for different models.

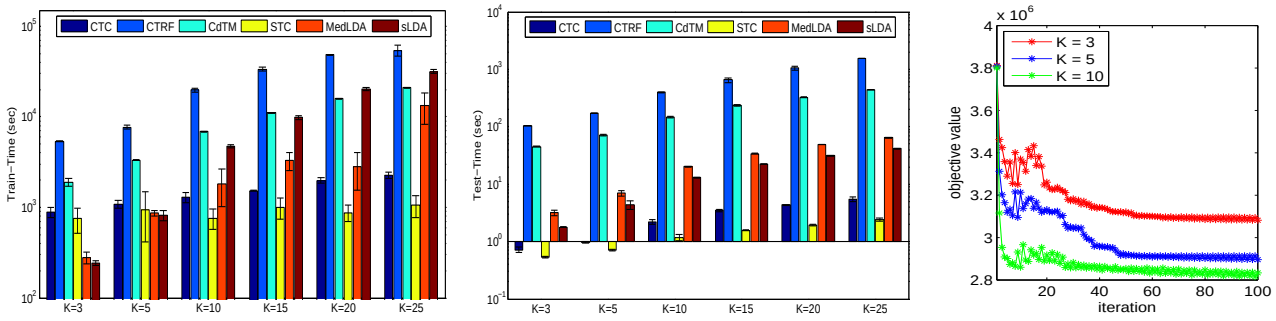
slightly better results than sLDA which uses maximum likelihood estimation to learn the predictive model.

4.2.2 Time Efficiency and Convergence

Figure 3 (a,b) shows the averaging time and standard deviation over five randomly initialized runs. All the models are implemented in C++ language without special optimization. For probabilistic models (e.g., MedLDA and sLDA), we use variational methods [3, 40] to do inference, which has a similar structure as the coordinate descent algorithm as we have discussed. We implement these methods using the same data structure, and run the experiments on a standard desktop with 2G RAM and a 2.66 GHz processor.

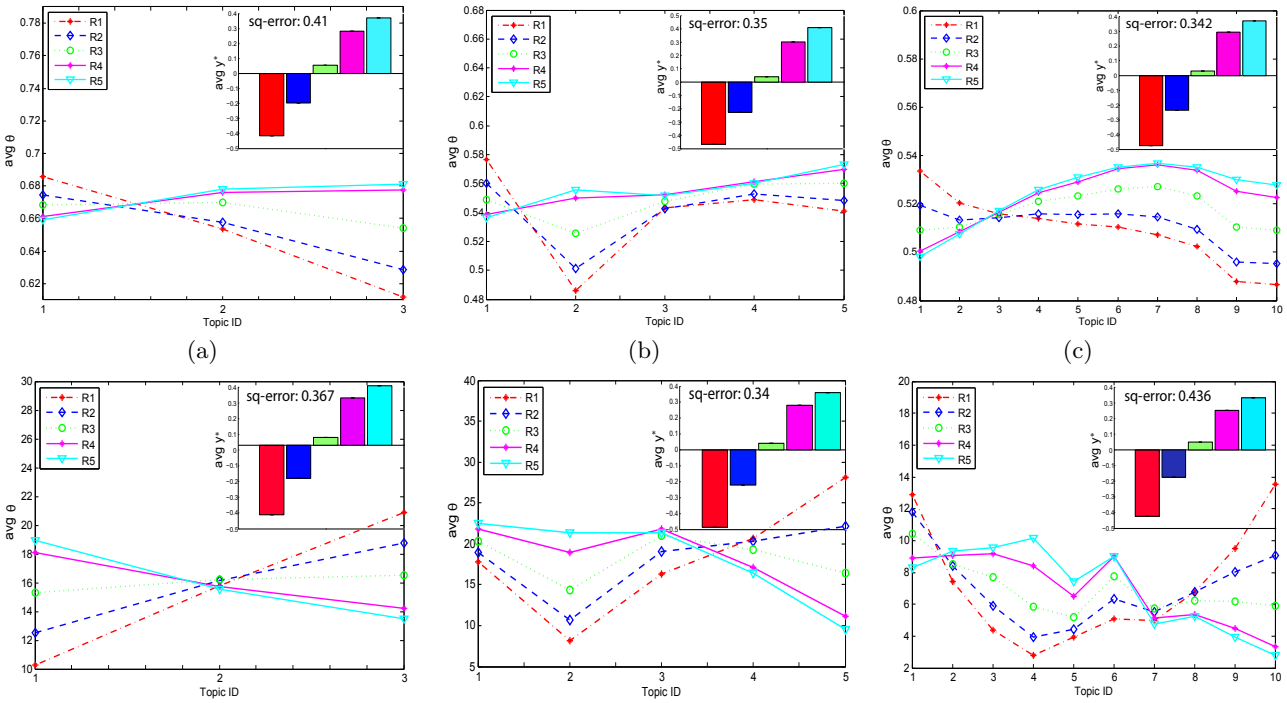
We can see that in testing, the conditional CTC is about 100 times more efficient than the probabilistic conditional topic models CdTM and CTRF. In training, CTC is about 10 times more efficient than CTRF. The reason for the smaller improvement in training is that most of the training time is spent on learning large-margin SVR, whose complexity is largely dependent on the number of training samples. For the same reason, we observe that the training time of topical coding models (i.e., STC and CTC) does not change much when the topic number increases, while the training time of probabilistic models (e.g., CTRF) usually scales linearly with the topic number. This suggests that the topical coding methods have a better scalability when the number of topics is large. If we compare the models without using features, we can see that the non-probabilistic STC is about 10 times more efficient than probabilistic topic models (e.g., sLDA and MedLDA) in testing and training (when the number of topics is large, e.g., 25). The reason for the improvements is that STC (or CTC in general) is not subject to the strict constraint as made in sLDA or MedLDA that θ is a normalized mixing proportion vector. Thus, STC has one additional dimension of freedom, which usually leads to faster convergence. Moreover, probabilistic models (e.g., sLDA) need many calls to the digamma function, which can cost additional computational resources [1].

Figure 3 (c) shows the convergence curves of training CTC



(a) (b) (c)

Figure 3: (a) training and (b) test time for different models on the small dataset; and (c) convergence curves of CTC with different numbers of topics.



(a) (b) (c)

(d) (e) (f)

Figure 4: Average document codes discovered by (a) 3-topic CTC; (b) 5-topic CTC; (c) 10-topic CTC; (d) 3-topic STC; (e) 5-topic STC; and (f) 10-topic STC respectively. The up-right corner of each plot shows the average prediction for the documents in different rating categories together with the square distance from the true average.

Table 2: Top words in different topics by 10-topic CTC and 10-topic STC.

CTC										STC									
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
told	left	n't	back	resort	service	restaurant	hotel	great	n't	night	staff	good	great	make	pool	room	problem	desk	told
asked	find	food	room	place	clean	room	room	good	nice	back	day	nice	trip	dinner	beach	hotel	floor	door	asked
looked	booked	people	water	rooms	floor	hotel	beach	room	staff	stay	place	time	large	car	hotel	n't	reviews	left	called
manager	reception	small	made	night	walk	breakfast	area	hotel	good	people	breakfast	area	ocean	week	room	beds	hotels	front	manager
work	arrived	bar	hotel	bed	room	family	free	n't	front	bed	rooms	view	friendly	recommend	n't	open	person	arrived	put
bad	called	stayed	make	days	hotel	main	pool	friendly	day	bathroom	stayed	service	lovely	town	bar	internet	noise	husband	inn
inn	wanted	found	stay	dinner	lot	price	large	beautiful	view	made	days	clean	beautiful	bedroom	resort	expect	rate	looked	call
star	pay	rooms	kids	things	pool	times	restaurants	lovely	trip	morning	find	walk	pools	park	restaurant	late	needed	bad	checked
sleep	put	room	check	car	hot	children	big	plenty	bit	found	lot	water	plenty	island	food	offer	hard	gave	leave
half	money	guests	shower	coffee	desk	day	day	pools	pool	guests	experience	free	lots	year	kids	deal	cost	charge	give
phone	staying	hotel	minutes	week	evening	pool	town	lots	time	shower	food	bit	visit	property	coffee	fine	arrival	wedding	decided
returned	thought	stay	hotels	parking	night	bedroom	couple	comfortable	room	minutes	tv	location	comfortable	end	main	housekeeping	room	room	sleep
toilet	paid	door	towels	recommend	time	ocean	balcony	enjoyed	hotel	check	feel	family	drive	home	evening	standard	due	guest	phone
finally	call	experience	morning	bathroom	long	table	time	sea	perfect	reception	stay	parking	enjoyed	wife	small	fridge	number	card	toilet
card	leave	holiday	husband	room	full	area	drinks	quiet	beach	booked	eat	hot	unit	road	side	hear	business	website	returned
dirty	checked	felt	thing	side	reviews	year	location	excellent	island	towels	drink	restaurants	excellent	extra	times	helpful	order	problems	point
key	give	person	hour	open	day	end	pretty	huge	street	things	years	price	quiet	local	children	disappointed	english	cleaned	finally
cleaned	decided	expect	eat	beach	park	property	staff	wonderful	area	holiday	book	big	huge	kitchen	lobby	king	travel	review	dirty
walked	gave	drink	needed	tv	home	internet	chairs	unit	breakfast	nights	choice	balcony	enjoy	worth	full	check-in	working	rest	key
care	wait	arrival	late	lobby	part	time	buffet	fresh	shuttle	wanted	bath	street	wonderful	part	couple	issue	euros	wall	walked
smell	charge	due	return	feel	bathroom	early	road	loved	special	thing	served	chairs	loved	lunch	drinks	ready	hotel	booking	smell
poor	offered	served	years	hours	rate	nights	local	happy	helpful	long	menu	suite	fresh	places	table	decent	window	change	care
booking	point	order	guest	desk	hard	extra	suite	wine	fantastic	pay	time	buffet	dining	meal	pretty	past	entire	noticed	owner
euros	cold	review	book	hotel	fact	close	worth	views	river	money	making	close	photos	sea	early	job	weekend	cleaning	poor
window	business	rest	read	stay	front	night	airport	grounds	cabo	hours	group	small	located	house	building	tables	head	lady	move
double	room	reservation	offer	air	bed	building	lunch	ate	tip	work	stop	tub	views	walking	airport	pleasant	light	sign	management
change	website	buy	spent	noise	enjoy	wife	places	tour	san	hour	buy	taxi	grounds	quality	area	reason	previous	paying	waiting
move	working	past	wedding	walking	beds	kitchen	drive	easy	adults	staying	taking	club	ate	high	top	elevator	worked	didnt	sheets
previous	run	absolutely	door	morning	set	visit	meal	fun	sand	thought	white	vacation	hilton	spa	bus	opened	mind	charged	man
management	doors	case	bath	cost	place	top	high	perfect	resorts	rooms	simply	living	beach	return	size	eating	show	loud	moved

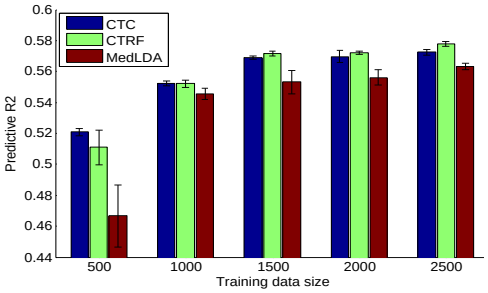


Figure 5: Predictive R2 scores for different models with 10 topical bases on the large dataset with different numbers of training data.

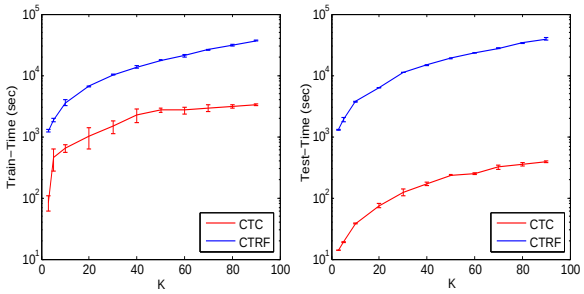


Figure 6: Training and test time for CTC and CTRF on the large dataset with different numbers of topical bases.

with different numbers of topics. We can see that the coordinate descent algorithm is quite stable and converges fast.

4.2.3 Characteristics of Code Representations

Now, we investigate the properties of the discovered representations. Figure 4 shows the average document codes θ discovered by the conditional CTC (plots a – c) and STC (plots d – f) that doesn't use features. The average document code is computed by averaging θ overall all the documents in each of the 5 rating categories, which are denoted by $R1$, $R2$, $\cdots$ and $R5$ from low to high rating values. We use lines to show the overall trend. On the up-right corner of each plot, we also show the average prediction y^* for the documents in each rating category, together with the square distance (i.e., square error) from the true average rating. Table 2 shows the top words of the learned dictionary (i.e., topics). Overall, we can see that both CTC and STC can discover very predictive code presentations (i.e., the average θ are quite different from each other), as also demonstrated by the outstanding prediction performance in Figure 2. However, using features can make CTC discover representations that are of a stronger regular pattern. For example, the documents with a high rating score (e.g., $R4$) have larger values on positive topics (e.g., T6 to T10) than the documents with a low rating score (e.g., $R2$), and the increasing trend from low rating to high rating is consistent. The reason for this strong regularity is that the discovered topics by CTC show strong positive, negative or neutral properties. For example, the topic T1 is obviously a negative topic; topics T3 to T5 tend to be neutral; while topics T6 to T10 tend to be positive, although they are about different aspects in detail. In contrast, STC which doesn't use features tends to discover topics (e.g., topic T7) that are mixtures of positive and negative words. On such topics, the corresponding average θ values usually don't show regular ordering, e.g., on topic T7 the value for $R1$ is smaller than that for $R3$, while the value for $R2$ is larger than that for $R4$.

4.3 Results on the Large Dataset

We now show the scalability of CTC on the large dataset. We compare with the conditional CTRF model and MedLDA which usually achieve better results than the other models (e.g., the standard LDA or sLDA). Also, we analyze the effect of the size of training data.

Figure 5 shows the predictive R2 scores of different models with the training set size being 500, 1000, 1500, 2000, and 2500. We build the training set from the large dataset of hotel reviews by selecting 100, 200, 300, 400 and 500 reviews from each rating category respectively. All the remaining reviews are used as test set. In this experiment, we set K at 10, which is sufficiently large for the review data, as demonstrated in Figure 2. For hyper-parameters, we fix $\rho = 5e^{-4}$, $\epsilon = 1e^{-3}$ and set $\lambda = \gamma$. Then, we select C and λ using the small dataset as validation set. We can see that the conditional models (i.e., CTC and CTRF) generally achieve better results than the unconditional model MedLDA, especially when the training set is small. For the two conditional models, again CTC and CTRF achieve comparable predictive R2 scores in all the five settings.

Finally, Figure 6 shows the training and test time on the large dataset with 2500 training samples and different numbers (e.g., $K = 3, 5, 10, 20, \cdots, 90$) of topical bases. Again, we can see that for each K CTC is about 100 times more efficient than CTRF in test and about 10 times more efficient than CTRF in training. Since the complexity of CTC in testing is linear in terms of the data size, we can expect that the efficient CTC can be scalable to very large datasets, e.g., processing tens of millions of documents in hours even when using a large number (e.g., 90) of topical bases. Moreover, the simple structure of the coordinate descent algorithm also makes it easy to be implemented in a distributed environment [32]. We leave this very large-scale implementation and evaluation as future work.

5. CONCLUSIONS AND FUTURE WORK

We have presented conditional topical coding (CTC), a novel non-probabilistic formulation of conditional topic models which can incorporate a rich set of features. By relaxing the strict normalization constraints, CTC learns non-negative code vectors and can avoid dealing with a hard-to-compute partition function. We develop an efficient and easy-to-implement coordinate descent learning algorithm. Our empirical results on online review data demonstrate that the non-probabilistic CTC can achieve state-of-the-art prediction performance and is much more efficient than probabilistic conditional topic models in both training and testing.

For future work, we plan to develop a parallel CTC and STC for very large-scale applications [27, 32] by using the simply structured coordinate descent algorithm. Also, we are interested in learning structured dictionaries [17] and extending CTC to incorporate various types of features, such as graph-based word similarity features [29] and document-level meta-data [26]. Finally, CTC raises a challenge to estimate the hyperparameters such as ρ and λ. Although the current restricted search works well in practice, it is worth of a systematical investigation to automatically estimate the hyperparameters, such as using the recent work [10, 7].

6. REFERENCES

[1] A. Asuncion, M. Welling, P. Smyth, and Y. Teh. On smoothing and inference for topic models. In *UAI*, 2009.

[2] S. Bengio, F. Pereira, Y. Singer, and D. Strelow. Group sparse coding. In *NIPS*, 2009.

[3] D. Blei and J. McAuliffe. Supervised topic models. In *NIPS*, 2007.

[4] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *JMLR*, (3):993–1022, 2003.

[5] S. Branavan, H. Chen, J. Eisenstein, and R. Barzilay. Learning document-level semantic properties from free-text annotations. In *ACL*, 2008.

[6] W. Buntine and A. Jakulin. Discrete components analysis. *Subspace, Latent Structure and Feature Selection Techniques*, 2006.

[7] G. Cawley, N. Talbot, and M. Girolami. Sparse multinomial logistic regression via bayesian ℓ_1 regularization. In *NIPS*, 2007.

[8] A. Culotta, D. Kulp, and A. McCallum. Gene prediction with conditional random fields. *Tech. Report UM-CS-2005-028, UMass, Amherst.*, 2005.

[9] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *ICML*, 2008.

[10] C.-S. Foo, C. Do, and A. Ng. A majorization-minimization algorithm for (multiple) hyperparameter learning. In *ICML*, 2009.

[11] J. Graça, K. Ganchev, B. Taskar, and F. Pereira. Posterior vs. parameter sparsity in latent variable models. In *NIPS*, 2009.

[12] T. Hofmann. Probabilistic latent semantic analysis. In *UAI*, 1999.

[13] P. Hoyer. Non-negative sparse coding. In *Neural Networks for Signal Processing*, 2002.

[14] A. Hyvärinen. Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. *Neural Computation*, (11):1739Ű–1768, 1999.

[15] A. Hyvärinen and P. Hoyer. A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, 41(18):2413–2423, 2001.

[16] L. Jacob, G. Obozinski, and J.-P. Vert. Group lasso with overlap and graph lasso. In *ICML*, 2009.

[17] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *ICML*, 2010.

[18] Y. Jo and A. Oh. Aspect and sentiment unification model for online review analysis. In *WSDM*, 2011.

[19] M. E. Khan, B. Marlin, G. Bouchard, and K. Murphy. Variational bounds for mixed-data factor analysis. In *NIPS*, 2010.

[20] S. Kim and E. P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. In *ICML*, 2010.

[21] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[22] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788 – 791, 1999.

[23] H. Lee, R. Raina, A. Teichman, and A. Ng. Exponential family sparse coding with applications to self-taught learning. In *IJCAI*, 2009.

[24] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *NIPS*, 2008.

[25] N. Meinshausen and B. Yu. Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics*, 37(1):246–270, 2009.

[26] D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *UAI*, 2008.

[27] D. Newman, A. Asuncion, P. Smyth, and M. Welling. Distributed algorithms for topic models. *JMLR*, (10):1801–1828, 2009.

[28] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

[29] J. Petterson, A. Smola, T. Caetano, W. Buntine, and S. Narayanamurthy. Word features for latent dirichlet allocation. In *NIPS*, 2010.

[30] D. Pinto, A. McCallum, X. Wei, and W. Croft. Table extraction using conditional random fields. In *SIGIR*, 2003.

[31] M. Shashanka, B. Raj, and P. Smaragdis. Sparse overcomplete latent variable decomposition of counts data. In *NIPS*, 2007.

[32] A. Smola and S. Narayanamurthy. An architecture for parallel topic models. In *VLDB*, 2010.

[33] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 2003.

[34] S. Sra, D. Kim, and B. Schölkopf. Non-monotonic poisson likelihood maximization. *Tech. Report, MPI for Biological Cybernetics*, 2008.

[35] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc.*, B(58):267–288, 1996.

[36] I. Titov and R. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *ACL*, 2008.

[37] C. Wang, D. Blei, and L. Fei-Fei. Simultaneous image classification and annotation. In *CVPR*, 2009.

[38] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: A rating regression approach. In *KDD*, 2010.

[39] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 2009.

[40] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: Maximum margin supervised topic models for regression and classification. In *ICML*, 2009.

[41] J. Zhu and E. P. Xing. Conditional topic random fields. In *ICML*, 2010.

[42] J. Zhu and E. P. Xing. Sparse topical coding. In *UAI*, 2011.