Discovering Different Types of Topics: Factored Topic Models

Yun Jiang and Ashutosh Saxena
Department of Computer Science, Cornell University.
{yunjiang,asaxena}@cs.cornell.edu

Abstract

In traditional topic models such as LDA, a word is generated by choosing a topic from a collection. However, existing topic models do not identify different *types* of topics in a document, such as topics that represent the content and topics that represent the sentiment. In this paper, our goal is to discover such different *types* of topics, if they exist. We represent our model as several parallel topic models (called topic factors), where each word is generated from topics from these factors jointly. Since the latent membership of the word is now a vector, the learning algorithms become challenging. We show that using a variational approximation still allows us to keep the algorithm tractable. Our experiments over several datasets show that our approach consistently outperforms many classic topic models while also discovering fewer, more meaningful, topics. [1]

1 Introduction

Topic models [Hofmann, 1999; Blei *et al.*, 2003; Teh *et al.*, 2006] have been extensively used to discover 'topics'—distributions over words in a vocabulary—shared across different documents in a corpus. However in many real-world datasets, there are different *types* of topics. For example, consider movie reviews that have: the sentiment (whether the review is positive or negative), the movie genre, different components of movie such as acting, directing or script, etc. Each of these would constitute a type of topic.

In some cases, such types (factors) of topics may be intuitive to us (e.g., for the movie reviews), but in other cases we want our algorithm to *discover* them—if they exist. While most prior work (e.g., [Zhao *et al.*, 2010; Eisenstein *et al.*, 2011; Paul and Girju, 2010], see §3) assumes knowledge of such topic factors in the data and also uses informative priors especially designed for the task, our goal is to do so without using any informative priors. Even given only a 'flat' bag-of-words representation from a document corpus, we want our algorithms to discover not only the topics but also the different factors in the topics. In fact, if such a factorization of topics does not exist for some dataset, then we want our algorithm to discover that.

In this work, we present factored topic models in which every data point is drawn from a set of different types (factors) of topics, and thus each sampled data point has an L-tuple latent membership. Topics from each factor would be drawn from a model similar to an LDA model [Blei *et al.*, 2003]. Our model would thus have L types of topics with $K^1, \ldots, K^L$ topics each. In the example about movie reviews, one type of topics could be about the sentiment and the other about the content. Thus, a movie review would now be drawn from a 2-tuple: (sentiment,content). The key effect of our modeling approach is that we can now model $\prod_\ell K_\ell$ effective topics with only $\sum_\ell K_\ell$ parameters. This allows us to discover parsimonious, more meaningful, topics.

Our method can be viewed as allowing a new type of generic structure on the topics—where the effective topics are composed from individual factors in the form of a tuple. There are other types of structures presented in the previous works that were also found to be quite effective, such as hierarchical tree-like structures [Blei *et al.*, 2004; Teh *et al.*, 2006; Williamson *et al.*, 2009]. These methods are complementary to ours in that they model different aspects found in the data. We will compare our model to these in § 4.

In our extensive experiments over *five* different datasets, we show that our approach consistently outperforms other models such as LDA, hierarchical DP and focused topic models. Experiments show that our model achieves lower perplexity on the hold-out documents while having fewer topics than the baselines. The extracted topics from the different factors are meaningful and also reflect certain orthogonality in the *types* of the topics extracted.

We then present a supervised version of our approach, where we show that one topic factor is more useful for prediction tasks, and the other is more useful for generating the document. We also show that our model outperforms the sLDA model significantly.

The contributions of this paper are:

- We present a new model for discovering different *types* of topics, if they exist:

  (a) Our topics are a tuple, with each component from an independent topic model. This allows us to effectively

[1] A first version of this work was made available on ArXiv [Jiang *et al.*, Aug 2012].

have $\prod_\ell K_\ell$ topics with $\sum_\ell K_\ell$ parameters.
(b) The "importance" of a factor is modeled and estimated from data.

- Our approach does not require any informative prior for particular datasets or tasks.
- The coupling of multiple topic models through observed data makes the parameter estimation challenging. We present a variational approximation with which it remains tractable.
- We present a supervised version of our algorithm.
- We show consistent improvements over several datasets, both qualitatively and quantitatively.

The rest of the paper is organized as follows. Section 2 presents the high-level idea, followed by the details of the algorithm in Section 2.1. Section 2.3 presents a variant of our method applicable to supervised learning setting. Section 3 describes the related work. Section 4 presents the experiments and Section 5 concludes.

2 Finite Factored Topic Models

In this section, we present the general idea of our factored topic models. We then describe the details in the next section.

Background. Latent Dirichlet allocation (LDA, see Fig. 1a) [Blei *et al.*, 2003] consists of K mixture components (or 'topics'), each of which is a multinomial distribution parameterized by θ_k, denoted as $F(\theta_k)$. The generative process is described as follows: first, a topic proportion $\boldsymbol{\pi}$ over K topics is drawn from a symmetric Dirichlet distribution with prior α; second, a topic $z \in \{1, \ldots, K\}$ is chosen for each word according to the mixing proportions $\boldsymbol{\pi}$, and finally, the word x is drawn from $F(\theta_z)$.

$$z|\boldsymbol{\pi} \sim \boldsymbol{\pi}; \qquad x|z, \theta \sim F(\theta_z).$$

For a given K, the parameters of the model, α and $\Theta = (\theta_k)_{k=1\ldots K}$, are learned from an unlabeled document corpus. Thus, LDA allows words from the same document share similar topic distributions while documents share finite topics.

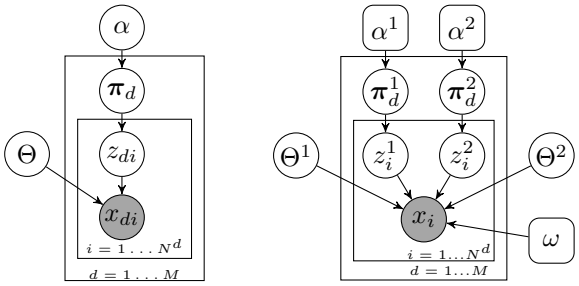
Overview of our model. Our model assumes a word is generated jointly by several independent topics. Particularly, a finite factored topic model (FFTM) with L factors has K^ℓ finite topics parameterized by θ_k^ℓ in each factor. Now, generating a word x involves choosing a topic $z^\ell \in \{1, \ldots, K^\ell\}$ for *each* of the L factors. Given $\Theta = (\theta_k^\ell)_{k=1\ldots K^\ell}^{\ell=1\ldots L}$ and $\boldsymbol{z} = (z^1, \ldots, z^L)$, we then draw x from the distribution parameterized by the selected L topics:

$$z^\ell|\boldsymbol{\pi}^\ell \sim \boldsymbol{\pi}^\ell,\ \ell = 1, \ldots, L; \qquad x|\boldsymbol{z}, \Theta \sim F(\theta^1_{z^1}, \ldots, \theta^L_{z^L}).$$

Note that the domain of the density function F is now a Cartesian product of the domains of L models.

Note that a FFTM with L factors is not the same as L independent models, as they are *linked through the observations*. This coupling makes the inference challenging, such as when optimizing parameters of the L mixture models jointly or sampling from their joint posterior distribution.

Furthermore, the standard topic model is a special case of our FFTM with $L = 1$. When $L > 1$, we have $\mathbf{K} = \prod_{\ell=1}^L K^\ell$ effective topics, each being a tuple $\theta'_k = (\theta^1_{j_1}, \ldots, \theta^L_{j_L})$ where $j_\ell \in \{1, \ldots, K^\ell\}$ from each of the factors. When L or K^ℓ is large, a standard model with actual $\mathbf{K}$ topics would not only be expensive to compute but also tend to over-fit the data. On the other hand, our FFTMs only construct $\sum_{\ell=1}^L K^\ell$ topics. While this is parsimonious, our method relies on the assumption that the data is generated from independent processes. *What happens when this is not the case?*



(a) LDA (FFTM with $L = 1$)

(b) FFTM with $L = 2$

Figure 1: Graphical representation of our FFTMs.

A continuum between the extremes. Is there a way to automatically discover if there are really L factors in the data, and if they are actually independent? The key idea in our work is to estimate the independence from the data itself, measured by an entangling parameter, ω (see Eq. (1)), for each of the topic factors.[2] We will show that when $\omega \to 0$ that factor becomes irrelevant, and thus letting the data decide the value of ω helps us from making a hard mistake of choosing some arbitrary value of L.

2.1 FFTMs

In our FFTM, we have L different types of topics. An example of a FFTM with $L = 2$ and its comparison to the classic LDA model are shown in Fig. 1. We define that our FFTM with L factors generates a document in the following process:

Algorithm 1 FFTM's generative process of a document.

for $\ell = 1, \ldots L$ **do**
  Draw a topic proportion $\pi^\ell|\alpha^\ell \sim Dir(\alpha^\ell)$.
end for
for $i = 1, \ldots$,#words **do**
  for $\ell = 1, \ldots, L$ **do**
    Draw a topic $z^\ell|\pi^\ell \sim \pi^\ell$.
  end for
  Draw a word $x_i|\boldsymbol{z}, \Theta^{1:L}, \boldsymbol{\omega} \sim$ Multi$(\sum_{\ell=1}^L \omega_\ell \theta^\ell_{z^\ell})$
end for

Specifically, now a word is drawn from a topic synthesized by L topics—one from each independent topic space—with the probability of

$$p(x|\boldsymbol{z}, \Theta^{1:L}, \boldsymbol{\omega}) = \textstyle\sum_{\ell=1}^L \omega_\ell \theta^\ell_{z^\ell, x}, \qquad (1)$$

[2] One direction for future work would be to use priors on ω.

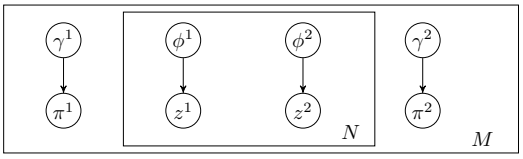


Figure 2: Graphical model representation of the variational distribution $q(\cdot)$.

where $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_L)$ represents the weight of the L topic models in forming a new topic. Note that when $L = 2$, we will only have a scalar ω. Here are a few interesting properties to note:

- It satisfies $\sum_{\ell=1}^{L} \omega_\ell = 1$ so that the synthesized topic, i.e., $\sum_{\ell=1}^{L} \omega_\ell \theta^\ell_{z^\ell}$, is a proper multinomial distribution. It is this constraint that makes learning challenging.
- As $\omega_\ell \to 0$, the ℓ^{th} topic factor becomes irrelevant.

Given a corpus and given $K_1, \ldots, K_L$, our goal is to estimate the parameters $\alpha^{1:L}, \Theta^{1:L}$ and $\boldsymbol{\omega}$ that maximizes the likelihood of documents.

Likelihood Function. After integrating out $\boldsymbol{\pi}^{1:L}$ and $\boldsymbol{z}$, we obtain the likelihood of a document $\mathbf{w} = (x_1, \ldots, x_N)$ conditioned on the model as,

$$\begin{aligned} & p(\mathbf{w}|\alpha^{1:L}, \Theta^{1:L}, \boldsymbol{\omega}) \\ & \propto \int \cdots \int \left(\prod_{i=1}^{K^1} (\pi_i^1)^{\alpha^1 - 1} \right) \cdots \left(\prod_{i=1}^{K^L} (\pi_i^L)^{\alpha^L - 1} \right) \times \\ & \prod_{n=1}^{N} \sum_{z^1=1}^{K^1} \cdots \sum_{z^L=1}^{K^L} \left(\prod_{i=1}^{L} \pi^i_{z^i} \right) \left(\sum_{i=1}^{L} \omega_i \theta^i_{z^i, x_n} \right) d\boldsymbol{\pi}^1 \cdots d\boldsymbol{\pi}^L. \end{aligned} \tag{2}$$

This distribution is intractable to compute in general. We therefore approximate it using variational inference, following the ideas used in LDA [Blei *et al.*, 2003].

2.2 Variational Inference.

Following the classic LDA method, we use the variational distribution,

$$q(\boldsymbol{\pi}^{1:L}, z^{1:L}_{1:N} | \boldsymbol{\gamma}^{1:L}, \phi^{1:L}_{1:N}) = \prod_{\ell=1}^{L} q(\boldsymbol{\pi}^\ell | \boldsymbol{\gamma}^\ell) \prod_{n=1}^{N} \prod_{\ell=1}^{L} q(z^\ell_n | \phi^\ell_n),$$

as an approximation to the true posterior distribution $p(\boldsymbol{\pi}^{1:L}, \boldsymbol{z}^{1:L} | \mathbf{w}, \alpha^{1:L}, \Theta^{1:L})$. Here, $\boldsymbol{\gamma}^{1:L}$ and $\phi^{1:L}_{1:N}$ are the free variational parameters (see Fig. 2). This allows us to obtain an adjustable lower bound on the log likelihood [Jordan *et al.*, 1999]. The difference between the two is quantified by the KL divergence which is,

$$\log p(\mathbf{w}|\alpha^{1:L}, \Theta^{1:L}, \boldsymbol{\omega}) - \mathcal{L}(\boldsymbol{\gamma}^{1:L}, \phi^{1:L}; \alpha^{1:L}, \Theta^{1:L}, \boldsymbol{\omega}),$$

where

$$\begin{aligned} \mathcal{L} \;=\; & \textstyle\sum_{\ell=1}^{L} E_q[\log p(\boldsymbol{\pi}^\ell|\alpha^\ell)] + \sum_{\ell=1}^{L} E_q[\log p(\boldsymbol{z}^\ell|\boldsymbol{\pi}^\ell)] \\ & - \textstyle\sum_{\ell=1}^{L} E_q[\log q(\boldsymbol{\pi}^\ell|\boldsymbol{\gamma}^\ell)] - \sum_{\ell=1}^{L} E_q[\log q(\boldsymbol{z}^\ell|\phi^\ell)] \\ & - E_q[\log p(\mathbf{w}|\boldsymbol{z}^{1:L}, \Theta^{1:L})]. \end{aligned} \tag{3}$$

We provide the exact form of $\mathcal{L}$ in the appendix.

Since KL divergence is always non-negative, $\mathcal{L}$ above is the lower bound of $p(\mathbf{w}|\alpha^{1:L}, \Theta^{1:L}, \boldsymbol{\omega})$. Therefore, our goal is to maximize $\mathcal{L}$ so that the likelihood $p(\mathbf{w}|\alpha^{1:L}, \Theta^{1:L}, \boldsymbol{\omega})$ can be large as well. During inference, the goal is to optimize $\mathcal{L}$ with respect to $\phi^{1:L}$ and $\boldsymbol{\gamma}^{1:L}$ for each document. During training, given M documents, our goal is to find the model's parameters that maximize $\mathcal{L}$. We solve it by estimating $\alpha^{1:L}, \Theta^{1:L}$ and $\boldsymbol{\omega}$ given the rest and iteratively inferring $(\phi_d^{1:L}, \boldsymbol{\gamma}_d^{1:L})$ for each document $\mathbf{w}_d$. The variational inference becomes more challenging than LDA due to the entanglement of the multiple topics through $\boldsymbol{\omega}$.

Parameter Estimation. We now show how to estimate $\alpha^{1:L}, \Theta^{1:L}$ and $\boldsymbol{\omega}$. When the variational distribution is fixed, the terms involving one particular α^ℓ in $\mathcal{L}$ are,

$$\begin{aligned} \mathcal{L}_{\alpha^\ell} \;=\; & \log\Gamma(K^\ell \alpha^\ell) - K^\ell \log\Gamma(\alpha^\ell) + \\ & (\alpha^\ell - 1) \textstyle\sum_{i=1}^{K^\ell} \Big(\Psi(\gamma_i^\ell) - \Psi\Big(\sum_{j=1}^{K^\ell} \gamma_j^\ell \Big) \Big), \end{aligned}$$

where $\Gamma(\cdot)$ is the Gamma function, and $\Psi(\cdot)$ is the digamma function. Since $\alpha^1, \ldots, \alpha^L$ are independent to each other and to $\boldsymbol{\omega}$ and $\Theta^{1:L}$ as well, we update them separately.

For M training documents and N_d as the number of words in document d, the terms involving $\Theta^{1:L}$ and $\boldsymbol{\omega}$ in $\mathcal{L}$ are,

$$\mathcal{L}_{\Theta^{1:L}, \boldsymbol{\omega}} = \sum_{d=1}^{M} \sum_{n=1}^{N_d} \sum_{\boldsymbol{z}} \left(\prod_{\ell=1}^{L} \phi^\ell_{dnz^\ell} \right) \log \left(\sum_{\ell=1}^{L} \omega_\ell \theta^\ell_{z^\ell, \mathbf{w}_{dn}} \right).$$

We therefore need to minimize the following after including the constraints of summing to 1:

$$\begin{aligned} \min_{\Theta^{1:L}, \boldsymbol{\omega}} \quad & -\mathcal{L}_{\Theta^{1:L}, \boldsymbol{\omega}} + \frac{1}{2} \eta \Big(\sum_{\ell=1}^{L} \omega_\ell - 1 \Big)^2 \\ & + \tfrac{1}{2} \textstyle\sum_{\ell=1}^{L} \sum_{i=1}^{K^\ell} \lambda_i^\ell \Big(\sum_{j=1}^{V} \theta_{ij}^\ell - 1 \Big)^2, \end{aligned} \tag{4}$$

where η and $\{\lambda_i^\ell\}$ impose a positive penalty for violating the constraint. The derivatives of (4) (denoted by G) with respect to Θ and $\boldsymbol{\omega}$ are

$$\begin{aligned} \frac{\partial G}{\partial \theta_{ij}^\ell} \;=\; & -\sum_{d,n,\boldsymbol{z}} \frac{\omega_\ell \prod_{t=1}^{L} \phi^t_{dnz^t}}{\sum_{t=1}^{L} \omega_t \theta^t_{z^t j}} + \lambda_i^\ell \Big(\sum_{k=1}^{V} \theta_{ik}^\ell - 1 \Big) \\ \frac{\partial G}{\partial \omega_\ell} \;=\; & -\sum_{d,n,\boldsymbol{z}} \frac{\theta^\ell_{z^\ell n} \prod_{t=1}^{L} \phi^t_{dnz^t}}{\sum_{t=1}^{L} \omega_t \theta^t_{z^t n}} + \eta \Big(\sum_{k=1}^{L} \omega_k - 1 \Big) \end{aligned}$$

We can see that the closed-form solutions are hard to obtain. Therefore, the optimal $\Theta^{1:L}$ and $\boldsymbol{\omega}$ are computed by the limited-memory BFGS algorithm, a standard quasi-Newton method, which only requires the functions of derivatives and the objective score.

Inference. We optimize $\mathcal{L}$ with respect to the variational distribution by setting the derivative to zero. This gives us the closed-form solutions for ϕ^ℓ_{nz} and γ^ℓ_z:

$$\phi^\ell_{nz} \propto \exp\Big(\Psi(\gamma^\ell_z) - \Psi\Big(\sum_{j=1}^{K^\ell} \gamma_j^\ell \Big) + \sum_{\boldsymbol{z}} \Big(\prod_{t \neq \ell} \phi^t_{nz^t} \Big) \log\Big(\sum_{t=1}^{L} \omega_t \theta^t_{i, \mathbf{w}_n} \Big) \Big)$$

$$\gamma^\ell_z = \alpha^\ell_z + \sum_{n=1}^{N} \phi^\ell_{nz}$$

2.3 Supervised FFTMs

Blei and McAuliffe [2007] presented supervised LDA (sLDA), where they showed that the extracted topics were useful for prediction tasks. A supervised version of our FFTM would give topics along multiple factors, where one topic factor could be directly relevant to the prediction task. We consider a supervised learning setup for our FFTM: during training, one factor of the topic assignments (i.e., z^ℓ) is observed. In this case, the variational inference remains the same except that we do not need to sum over the known assignments anymore.

In many applications, we only know the class of a document, say y_d, instead of per-word topic assignments $\boldsymbol{z}_n$. In a traditional topic model, setting $z_n = y_d$ for every word w_n would be too restrictive. Therefore, sLDA assumes y_d is the result of a linear regression over $\boldsymbol{z}_n$. On the other hand, our FFTM has multiple factors, and we can simply set one factor of topic assignment to y_d, since the terms from the same document still can choose different topics in other factors.

3 Related Work

There is a huge body of work employing topic models. Here we only name a few and refer the reader to [Blei, 2011; Steyvers and Griffiths, 2007] for a more general survey.

There are some recent works consider modeling different types of topics. For example, Zhao et al. [2010] propose specialized models with hand-designed functions for separating opinion and aspect topics in online reviews. Sparse additive generative models (SAGE) [Eisenstein *et al.*, 2011] assume a word is drawn from a regular topic and a 'background' topic. Similarly, Topic-aspect models (TAM) [Paul and Girju, 2010] also consider two types of topics, however, each word can only be from one type. Thus it only divides, not factorizes, topics into two types. Unlike these works, our model does not require informative priors, factorizes topics into different types and determines from data how important each type is.

In contemporary work, Factorial LDA [Paul and Dredze, 2012] considers factorizations of topic priors instead of topics directly. While it also captures different types of topics, it needs to explicitly learn $\prod_{\ell=1}^{L} K^\ell$ number of topics and thus is less parsimonious than our FFTM.

Many topic models relax the assumptions in LDA by modeling word non-exchangeability [Wallach, 2006; Griffiths *et al.*, 2004], or by modeling the correlations among topics [Blei and Lafferty, 2007; Kim and Sudderth, 2011; Putthividhya *et al.*, 2009]. These ideas are complementary to ours, and similar techniques may be applied to FFTMs. There has also been work on incorporating other meta-data such as authors [Rosen-Zvi *et al.*, 2004; Dai and Storkey, 2011], citations [Nallapati *et al.*, 2008], and tags [Das *et al.*, 2011]. Our FFTM does not require such meta-data. More importantly, none of these extensions consider the factorization of topics into different types.

There are previous works in matrix factorization [Ding *et al.*, 2008], factored models [Ranzato *et al.*, 2010] and parameter sharing [Kim and Xing, 2010; Jalali *et al.*, 2010; Li *et al.*, 2011; Mei *et al.*, 2008; Newman *et al.*, 2011], where a lower dimensional representation of the parameters is used. Even though these approaches are for completely different domains (although some connections were explored in [Arora *et al.*, 2012]), they are relevant to our work since at a high-level FFTM also uses a compact "factored" representation for the parameters.

Our model takes ideas from multidimensional clustering [Chen *et al.*, 2012], two-way groupings [Savia *et al.*, 2009; Hofman and Puzicha, 1999], some biclustering models [Madeira and Oliveira, 2004] and collaborative filtering [Si and Jin, 2003]. In these models, data (user preferences or rating scores) is organized in multiple dimensions (such as user groups and object groups). In this work, we are interested in modeling the posterior density and topics instead of clustering. Some recent works have also applied interacting LDA models for multi-modal data [Porteous *et al.*, 2008], however their input comes from different modalities. Cross-cat [Shafto *et al.*, 2006] proposed partitioning the binary features for categorization based on different criterion. This is quite different from our FFTM that partitions the parameters. As an example, if Crosscat were to be applied to topic modeling, Crosscat would find topics in each partition of the vocabulary, while topics in our FFTM share the whole vocabulary. These fundamental differences make our FFTM unique in finding different types of topics.

Topic models have been widely applied to several applications such as building image hierarchy [Li *et al.*, 2010], object detection [Sudderth *et al.*, 2006], annotation and segmentation [Li *et al.*, 2009] and robotic scene arrangement [Jiang *et al.*, 2012; Jiang and Saxena, 2012]. However, none of the models presented in these works consider generating data points from factored topics. Our recent work [Jiang *et al.*, 2013], in the application of 3D object detection, generalizes FFTM to infinite factored topic models (IFTMs) where the number of topics in each factor is not known in advance but learned from data. We model each type of topics using Dirichlet process mixture model, a nonparametric Bayesian method that is used to model unknown number of mixture components. IFTMs are advantageous in physical scene modeling where aspects such as object appearance, latent human poses, human activities and human-object interactions can be incorporated into one model and with no constraining on the number of components in each aspect.

4 Experiments and Results

In this section, we first evaluate our FFTM on the task of document modeling, on four different corpora and against three baselines. We then test it on an additional movie review dataset where we also compare the classification based on the learned topics.

4.1 Document Topic Modeling

We test our FFTM on four document corpora: **Dataset-1** contains processed NIPS 1-12 proceedings with 1447 papers organized into 9 sections and 5270 words after removing words appeared more than 4000 times or fewer than 50 times;[3] **Dataset-2** includes randomly selected 1000 documents from the 20 newsgroups with a total of 1498 words

[3] http://www.cs.nyu.edu/∼roweis/data.html

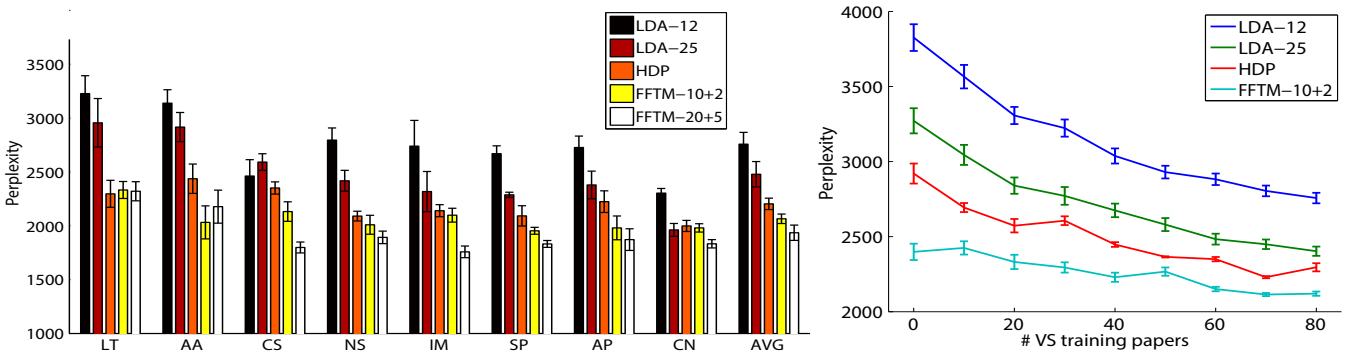


Figure 3: Results on Dataset-1. Perplexity of mixing VS and other 8 sections (left) and the average perplexity when changing the number of training documents from VS (right). The error bars are one standard error.

after removing stop-words and words in fewer than 5 documents;[4] **Dataset-3** selects 1000 encyclopedia articles with 1200 words;[5] **Dataset-4** takes 500 articles from Psychological Review with 1244 words.[6] All the results are based on 5-fold cross validation. Experiments on the first two datasets are performed with the same setup as in [Teh *et al.*, 2006] and [Williamson *et al.*, 2009] respectively for fair comparison.

We compare our model against LDA and HDP[Teh *et al.*, 2006], and focus on studying the affect of having additional factors in the topics with our FFTM. There are other methods such as correlated topic models (CTM) [Blei and Lafferty, 2007] that capture correlation between the topics,[7] or methods that use other prior information. These techniques are complementary to our idea of having factored topics. We also compare against focused topic models [Williamson *et al.*, 2009] on Dataset-2 that try to learn sparse topic mixture patterns.

Effect of having more than one factor of topics: To investigate how well our model can learn general topics and section-specific topics (as the two topic factors in our model), we train on 80 articles from the VS (vision science) section and 80 articles from one of the other 8 sections. We test on the other 47 VS papers. We use the perplexity [Blei *et al.*, 2003] of the on-hold documents to evaluate the learned topic model: $\exp(-(\sum_{d=1}^{D}\log p(\mathbf{w}_d))/\sum_{d=1}^{D}N^d)$. A lower perplexity indicates higher likelihood of the test data and thus better performance.

Fig. 3-left shows the perplexity obtained by LDA, HDP and our FFTM. In the comparison with LDA, we set the LDA's topic number K equal to the total sum of FFTM's topic numbers K^1+K^2, so that the two models have the same number of parameters. We see that our method performs significantly better than LDA across all eight sections for both 12 and 25 topics. This is due to that our FFTM effectively represents more topics than LDA with the same number of parameters. Such trends hold for different values of K, K^1 and K^2.[8] We show the results on the other three datasets in Fig. 4. Compared to the baselines, our FFTM obtains the lowest perplexity and demonstrates its robustness in different scenarios.

We noticed that when we have a large value of K^1 or K^2, the estimated value of ω was generally small for the corresponding factor. Furthermore, in certain data-sets such as Dataset-1 and Dataset-4, the value of ω was closer to 0.5, indicating that both factors were useful. On the other hand, for Dataset-2 and Dataset-3, the value of ω was closer to 0.

Robustness to size of the training corpus: In another experiment on Dataset-1, we change the number of training documents from VS from 0 to 80, but always test on the rest 47 VS documents. When the training set is small, the domain of the training and test dataset would be different and thus can be used to test the transfer of topic learning. Fig. 3-right shows the perplexity, averaged over all sections, with respect to different training documents. We can see that the performance of LDA largely depends on the number of VS papers, while the change in the perplexity of HDP and our model is less significant. Our FFTM not only beats all the baselines but also gives the most consistent results in all cases. This demonstrates that 1) our model can learn the common topics of two different sections, and 2) it is less sensitive to having a small training set since the factorization of topics encourages the sparsity in the learned topics which prevents over-fitting.

Qualitative study of the topics found: In Dataset-1, we found that one of the topic factors (one for which K was small, e.g., 2 or 5) learned the 'commonly shared' topics across the different sections (see Fig. 5). This ability to have shared topics is quite useful. This indicates by that both our FFTM and HDP outperform LDA. However, HDP does so only in a hierarchy so that a sub-tree shares similar topic proportions. Hence, it does not reduce the number of topics needed to model by factoring out the shared topics as another factor. In fact, the number of topics used in HDP is around 55, far more than 12 topics in our model.

In order to explore what orthogonal topics our FFTM discovered, we list one topic from each factor in Fig. 5. Topics

[4] http://people.csail.mit.edu/jrennie/20Newsgroups/

[5] http://www.cs.nyu.edu/∼roweis/data.html

[6] http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm

[7] In our experiments, we found CTM to be quite sensitive to the number of documents being trained on, and performed about the same as LDA in the cases it worked.

[8] We tested LDA with up to 900 topics, and the best perplexity is 2308 given by 100 topics, which is worse than 1934 given by our FFTM with 20+5 topics.

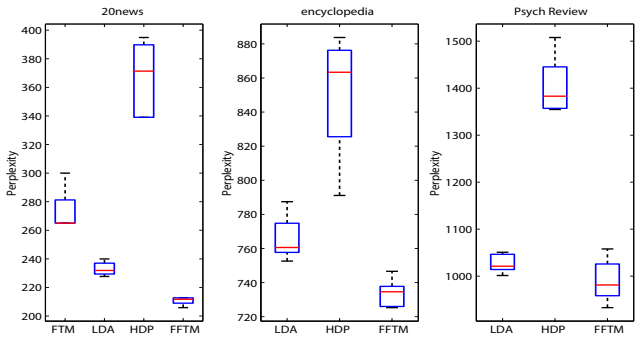


Figure 4: Results on Dataset-2, Dataset-3 and Dataset-4 , performed by focused topic model (FTM, reported in [Williamson *et al.*, 2009]), LDA, HDP and our FFTM. For LDA and FFTM, we only report the best performance among different number of topics.

	VS+AP	VS+CN	VS+SP
Factor 1	parameter	response	brain
	measure	murray	digits
	subthreshold	receptive	bifurcation
	versions	lower	force
	tuned	type	dimension
	bound	statistical	electrodes
Factor 2	work	work	response
	algorithms	section	form
	analysis	features	algorithms
	achieved	problems	rate
	form	obtained	low
	local	images	local

Figure 5: Topics of VS combined with three other sections found by our FFTM (in top words). Topics from first factor are section-specific and different from each other while the second one contains popular terms common in NIPS and the topics in it do not vary much.

from the first row are quite different from each other containing some keywords for specific sections, such as 'digits' for the SP (speech and signal processing) while the topics in the bottom row are mostly from popular words in NIPS such as 'work' and 'algorithms'. This indeed reflects that FFTM represents topics parsimoniously.

In FFTMs, K^1 and K^2 are the tunable parameters, and setting them would affect the performance. Similar to LDA, the optimal value for the number of topics ($K^1 + K^2$) varies with the size and heterogeneity of the corpus, and one may have to try different values. The ratio K^1/K^2 is interesting—in most datasets we found that an asymmetric value performs better, e.g., the result of setting $K^1 = 20, K^2 = 5$ is better than $K^1 = K^2 = 10$.

4.2 Movie Review Analysis: Supervised FFTMs

We test our FFTM on a movie review dataset, where our goal is to study the performance of our supervised FFTM in finding topics of interest and also in predicting of review ratings.

The movie review dataset (**Dataset-5**), introduced by [Pang and Lee, 2005], contains 5006 reviews paired with ratings from four reviewers.[9] The dataset is also used to test supervised LDA (sLDA) [Blei and McAuliffe, 2007]. Hence we adopt the same setup as in [Blei and McAuliffe, 2007]: 5000 words are chosen by tf-idf and experiments are performed using 5-fold cross validation.

In addition to evaluate the topic modeling by perplexity, we also consider regression on the ratings using the learned topics. The learned topics are treated as the feature space and $\phi^{1:L}$ from the variational distribution are the feature values that are used to infer the ratings [Blei and McAuliffe, 2007]. In particular, for each document d, we compute $\bar{\phi}_d^\ell = \frac{1}{N_d}\sum_{n=1}^{N_d}\phi_d^\ell n$, and then simply apply a linear regression, parameterized by β, to infer its rating: $\hat{y}_d = \beta_0 + \sum_{\ell=1}^{L}\sum_{z=1}^{K^\ell}\beta_\ell\bar{\phi}_{dz}^\ell$. β is learned from training data given true ratings y_d and tested out on the test data. The performance is evaluated by the coefficient of determination defined as $R^2 = 1 - \frac{\sum_d y_d - \hat{y}_d}{y_d - \bar{y}_d}$ where $\bar{y}_d = \sum_d y_d/M$.

The movie reviews are affected by many factors, such as the sentiment (whether the reviews is positive or negative), the movie genre, different components of movie such as acting, directing or script, etc. Among these, the sentiment is directly relevant to the ratings. Thus, identifying the topics of sentiment becomes very crucial. Since reviews that have the same ratings are more likely to share the same topics of sentiment, we use our supervised FFTM where we use the available labels, i.e., z^2 during training.

We compare our unsupervised and supervised FFTMs against LDA and sLDA respectively. To test the effect of the number of topics, we vary it from 5 to 50 for LDA and sLDA. For our models, we set K^2 to 4 and 10, and vary K^1, thus reporting four curves with x-axis equal to $K^1 + K^2$. The result for perplexity is shown in Fig. 6-left.[10] We can see that while sLDA performs better than LDA, our FFTMs *significantly* outperform both. This demonstrates the efficiency of our learned factored topics, under the same number of parameters.

We note that $K^2 = 4$ gives better result than $K^2 = 10$ and the performance drops quickly as K^1 increases. This means that this dataset does not have a large variety of topics and thus models with large $K^1 \times K^2$ values tend to over-fit.

The result of rating prediction using the topics as features is shown in Fig. 6-right. Our unsupervised FFTM outperforms the unsupervised LDA model and our supervised FFTM outperforms the sLDA model. We also performed linear regression *only* on the second topic factor (i.e., features are only $\bar{\phi}_{d1}^2, \ldots, \bar{\phi}_{dK^2}^2$, shown in magenta in the figure). We see that they achieve almost the same performance as using all the topics. (In fact, the performance of only using $\bar{\phi}^1$ is poor, around 0.1.) This verifies that supervised FFTMs are useful in modeling different factors of the topics where one factor may be more useful for supervised prediction tasks.

We also qualitatively examine the topics found along the first and the second factor. Fig. 7 lists the topics as represented by five top-ranked terms. The top four topics are

[9] http://www.cs.cornell.edu/people/pabo/movie-review-data/

[10] To be consistent with previous experiment, we use the perplexity instead of the per-word held out log-likelihood used in [Blei and McAuliffe, 2007], which can be converted from the perplexity by taking its negative logarithm.

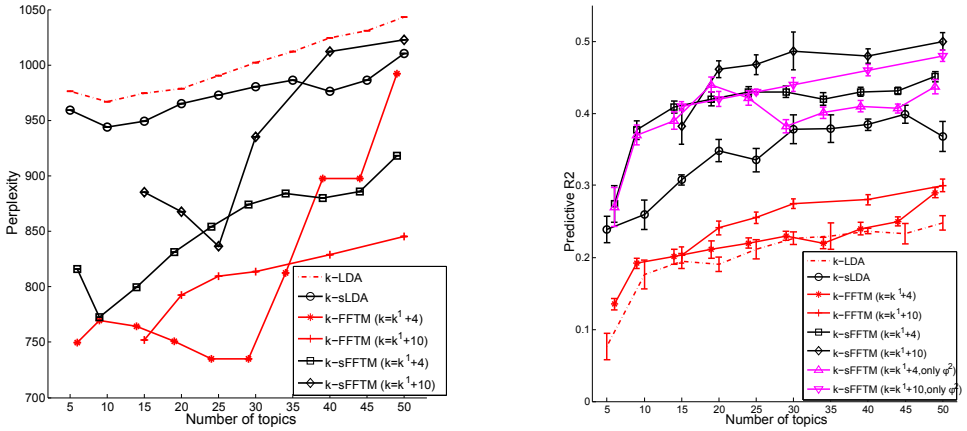


Figure 6: Perplexity and predictive R^2 on movie reviews (Dataset-5). We compare our supervised FFTMs (sFFTM) and unsupervised FFTMs (FFTM) against LDA and sLDA. X-axis refers to k, the number of topics in different models. For FFTMs, $K = K^1 + K^2$ where K^2 is set to 4 or 10 and K^1 varies.

Factor 1	presentation	violence	large	script
	plot	sexual	budget	director
	cinema	dark	material	screen
	british	animations	waste	pictures
	lines	comedy	core	music
Factor 2	instead	painfully	hilarious	efforts
	acrimonious	barely	effective	marvelous
	trouble	hard	perfectly	pleasant
	old	obvious	epic	consistently
	creepy	poor	beautiful	like

Figure 7: Two types of movie-review topics found by our FFTM. The first factor contains terms about the content of the movie, and the second one contains terms indicating the sentiment.

about different aspects of a movie, such as its genre, production, components. The bottom four topics are clearly related to sentiments: The first two correspond to negative reviews while the last two for positive ones.

We have made the code available at:
`http://pr.cs.cornell.edu/factoredtopicmodel/`

5 Conclusion

In this paper, we presented the factored topic models for discovering different *types* of topics. Each word now has a *vector* of latent topic assignments (as compared to one topic assignment in the classic topic models) indicating that it is generated by topics from multiple factors. Our model also estimated discovered the importance of different types (factors) of topics. With multiple factors, the parameter estimation becomes challenging. We presented a variational approximation that results in a tractable algorithm. We then also presented a supervised FFTM. Over five different datasets, we showed that our model outperforms the standard topic models while producing fewer, but meaningful, topics.

In summary, the key insight in this work is that having L factors with $K^1, \ldots, K^L$ topics each allows us to model $\prod_\ell K_\ell$ effective topics with only $\sum_\ell K_\ell$ parameters. This would only work when the real-world datasets are produced by different *types* of topics. In our experiments on five different datasets, we did find that that was the case.

Acknowledgments

We thank Marcus Lim and Elaheh Momeni for helpful discussions. This research was funded by Microsoft Faculty Fellowship and Sloan Fellowship to Saxena.

Appendix

The expanded form of $\mathcal{L}$ for one document is (see §2.2):

$$
\begin{aligned}
\mathcal{L} \;=\; & \sum_{\ell=1}^{L}\Big(\log\Gamma(\sum_{j=1}^{K^\ell}\alpha_j^\ell) - \sum_{j=1}^{K^\ell}\log\Gamma(\alpha_j^\ell) \\
& + \sum_{i=1}^{K^\ell}(\alpha_i^\ell - 1)(\Psi(\gamma_i^\ell) - \Psi(\sum_{j=1}^{K^\ell}\gamma_j^\ell)) \\
& + \sum_{n=1}^{N}\sum_{i=1}^{K^\ell}\phi_{ni}^\ell(\Psi(\gamma_i^\ell) - \Psi(\sum_{j=1}^{K^\ell}\gamma_j^\ell)) - \log\Gamma(\sum_{j=1}^{K^\ell}\gamma_\ell^\ell) \\
& + \sum_{j=1}^{K^\ell}\log\Gamma(\gamma_j^\ell) - \sum_{i=1}^{K^\ell}(\gamma_i^\ell - 1)(\Psi(\gamma_i^\ell) + \Psi(\sum_{j=1}^{K^\ell}\gamma_j^\ell)) \\
& - \sum_{n=1}^{N}\sum_{i=1}^{K^\ell}\phi_{ni}^\ell\log\phi_{ni}^\ell\Big) \\
& + \sum_{n=1}^{N}\sum_{z^1=1}^{K^1}\cdots\sum_{z^L=1}^{K^L}\left(\prod_{\ell=1}^{L}\phi_{nz^\ell}^\ell\right)\log\left(\sum_{\ell=1}^{L}\omega_\ell\theta_{z^\ell,\mathbf{w}_n}^\ell\right).
\end{aligned}
$$

For M documents, the equation above would be summed over each document ($d = 1, \ldots, M$), and N would be replaced by N_d (the number of words in the document d).

References

[Arora *et al.*, 2012] Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond SVD. In *FOCS*, 2012.

[Blei and Lafferty, 2007] D.M. Blei and J.D. Lafferty. A correlated topic model of science. *The Annals App Stats*, 1(1):17–35, 2007.

[Blei and McAuliffe, 2007] David M. Blei and Jon D. McAuliffe. Supervised topic models. In *NIPS*, 2007.

[Blei *et al.*, 2003] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[Blei *et al.*, 2004] David Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *NIPS*, 2004.

[Blei, 2011] D.M. Blei. Introduction to probabilistic topic models. *Communications of the ACM*, pages 1–16, 2011.

[Chen *et al.*, 2012] T. Chen, N.L. Zhang, T. Liu, and Y. Wang K.M. Poon. Model-based multidimensional clustering of categorical data. *Artificial Intelligence*, 176:2246–2269, 2012.

[Dai and Storkey, 2011] A. Dai and A. Storkey. The grouped author-topic model for unsupervised entity resolution. *ICANN*, 1:241–249, 2011.

[Das *et al.*, 2011] P. Das, R. Srihari, and Y. Fu. Simultaneous joint and conditional modeling of documents tagged from two perspectives. In *CIKM*, 2011.

[Ding *et al.*, 2008] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *CSDA*, 52(8), 2008.

[Eisenstein *et al.*, 2011] J. Eisenstein, A. Ahmed, and E.P. Xing. Sparse additive generative models of text. In *ICML*, 2011.

[Griffiths *et al.*, 2004] T.L. Griffiths, M. Steyvers, D.M. Blei, and J.B. Tenenbaum. Integrating topics and syntax. In *NIPS*, 2004.

[Hofman and Puzicha, 1999] T. Hofman and J. Puzicha. Latent class models for collaborative filtering. In *IJCAI*, 1999.

[Hofmann, 1999] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57, 1999.

[Jalali *et al.*, 2010] A. Jalali, P. Ravikumar, S. Sanghavi, and C. Ruan. A dirty model for multi-task learning. In *NIPS*, 2010.

[Jiang and Saxena, 2012] Y. Jiang and A. Saxena. Hallucinating humans for learning robotic placement of objects. In *ISER*, 2012.

[Jiang *et al.*, 2012] Y. Jiang, M. Lim, and A. Saxena. Learning object arrangements in 3d scenes using human context. In *ICML*, 2012.

[Jiang *et al.*, 2013] Yun Jiang, Hema Koppula, and Ashutosh Saxena. Hallucinated humans as the hidden context for labeling 3d scenes. In *CVPR*, 2013.

[Jiang *et al.*, Aug 2012] Yun Jiang, Marcus Lim, and Ashutosh Saxena. Multidimensional membership mixture models. In *ArXiv*, Aug 2012.

[Jordan *et al.*, 1999] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. Introduction to variational methods for graphical models. *Machine Learning*, 37:183233, 1999.

[Kim and Sudderth, 2011] D. Kim and E.B. Sudderth. The doubly correlated nonparametric topic model. In *NIPS 24*, 2011.

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

[Li *et al.*, 2009] L-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In *CVPR*, 2009.

[Li *et al.*, 2010] L.J. Li, C. Wang, Y. Lim, D.M. Blei, and L. Fei-Fei. Building and using a semantic visual image hierarchy. In *CVPR*, 2010.

[Li *et al.*, 2011] C. Li, A. Saxena, and T. Chen. θ-MRF: Capturing spatial and semantic structure in the parameters for scene understanding. In *NIPS*, 2011.

[Madeira and Oliveira, 2004] S.C. Madeira and A.L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *Comp Bio and Bioinformatics, IEEE Trans*, 1(1):24–45, 2004.

[Mei *et al.*, 2008] Qiaozhu Mei, Deng Cai, Duo Zhang, and ChengXiang Zhai. Topic modeling with network regularization. In *WWW*, 2008.

[Nallapati *et al.*, 2008] R.M. Nallapati, A. Ahmed, E.P. Xing, and W. Cohen. Joint latent topic models for text and citations. In *KDD*, 2008.

[Newman *et al.*, 2011] D. Newman, E. Bonilla, and W. Buntine. Improving topic coherence with regularized topic models. In *NIPS*, 2011.

[Pang and Lee, 2005] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL*, 2005.

[Paul and Dredze, 2012] M. Paul and M. Dredze. Factorial lda: Sparse multi-dimensional text models. In *NIPS*, 2012.

[Paul and Girju, 2010] M. Paul and R. Girju. A two-dimensional topic-aspect model for discovering multi-faceted topics. *Urbana*, 51:61801, 2010.

[Porteous *et al.*, 2008] Ian Porteous, Evgeniy Bart, and Max Welling. Multi-hdp: A nonparametric bayesian model for tensor factorization. In *AAAI*, 2008.

[Putthividhya *et al.*, 2009] D. Putthividhya, H. Attias, and S. Nagarajan. Independent factor topic models. In *ICML*, 2009.

[Ranzato *et al.*, 2010] M. Ranzato, A. Krizhevsky, and G. Hinton. Factored 3-way restricted boltzmann machines for modeling natural images. In *AISTATS*, 2010.

[Rosen-Zvi *et al.*, 2004] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *UAI*, 2004.

[Savia *et al.*, 2009] E. Savia, K. Puolamaki, and S. Kaski. Latent grouping models for user preference prediction. *Mach Learn*, 2009.

[Shafto *et al.*, 2006] P. Shafto, C. Kemp, V. Mansinghka, M. Gordon, and J. Tenenbaum. Learning cross-cutting systems of categories. In *28th Annual Conf Cog Sci Soc*, 2006.

[Si and Jin, 2003] L. Si and R. Jin. Flexible mixture model for collaborative filtering. In *ICML*, 2003.

[Steyvers and Griffiths, 2007] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

[Sudderth *et al.*, 2006] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky. Describing visual scenes using transformed dirichlet processes. In *NIPS 18*, 2006.

[Teh *et al.*, 2006] Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei. Hierarchical dirichlet processes. *J American Stat Association*, 101(476):1566–1581, 2006.

[Wallach, 2006] H.M. Wallach. Topic modeling: beyond bag-of-words. In *ICML*, pages 977–984, 2006.

[Williamson *et al.*, 2009] S. Williamson, C. Wang, K. A. Heller, and D. M. Blei. Focused topic models. In *NIPS Workshop on Applications of Topic Models: Text and Beyond*, 2009.

[Zhao *et al.*, 2010] Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *EMNLP*, 2010.