
@inproceedings{Nguyen:Hu:Boyd-Graber-2014,  
Author = {Thang Nguyen and Yuening Hu and Jordan Boyd-Graber},  
Url = {docs/2014_acl_anchor_reg.pdf},  
Booktitle = {Association for Computational Linguistics},  
Location = {Baltimore, MD},  
Year = {2014},  
Title = {Anchors Regularized: Adding Robustness and Extensibility to Scalable Topic-Modeling Algorithms},  
}

Links:
- Talk https://www.youtube.com/watch?v=zN4fRz8xwK4&list=UUIXpj1xPL50wP0_gOHISQ

Downloaded from http://cs.colorado.edu/~jbg/docs/2014_acl_anchor_reg.pdf
Anchors Regularized: Adding Robustness and Extensibility to Scalable Topic-Modeling Algorithms

Thang Nguyen
iSchool and UMIACS, University of Maryland
and National Library of Medicine, National Institutes of Health
daithang@umiacs.umd.edu

Yuening Hu
Computer Science
University of Maryland
ynhu@cs.umd.edu

Jordan Boyd-Graber
iSchool and UMIACS
University of Maryland
jbg@umiacs.umd.edu

Abstract

Spectral methods offer scalable alternatives to Markov chain Monte Carlo and expectation maximization. However, these new methods lack the rich priors associated with probabilistic models. We examine Arora et al.’s anchor words algorithm for topic modeling and develop new, regularized algorithms that not only mathematically resemble Gaussian and Dirichlet priors but also improve the interpretability of topic models. Our new regularization approaches make these efficient algorithms more flexible; we also show that these methods can be combined with informed priors.

1 Introduction

Topic models are of practical and theoretical interest. Practically, they have been used to understand political perspective (Paul and Girju, 2010), improve machine translation (Eidelman et al., 2012), reveal literary trends (Jockers, 2013), and understand scientific discourse (Hall et al., 2008). Theoretically, their latent variable formulation has served as a foundation for more robust models of other linguistic phenomena (Brody and Lapata, 2009).

Modern topic models are formulated as a latent variable model. Like hidden Markov models (Rabiner, 1989; HMM), each token comes from one of $K$ unknown distributions. Unlike a HMM, topic models assume that each document is an admixture of these hidden components called topics. Posterior inference discovers the hidden variables that best explain a dataset. Typical solutions use MCMC (Griffiths and Steyvers, 2004) or variational EM (Blei et al., 2003), which can be viewed as local optimization: searching for the latent variables that maximize the data likelihood.

An exciting vein of new research provides provable polynomial-time alternatives. These approaches provide solutions to hidden Markov models (Anandkumar et al., 2012), mixture models (Kannan et al., 2005), and latent variable grammars (Cohen et al., 2013). The key insight is not to directly optimize observation likelihood but to instead discover latent variables that can reconstruct statistics of the assumed generative model. Unlike search-based methods, which can be caught in local minima, these techniques are often guaranteed to find global optima.

These general techniques can be improved by making reasonable assumptions about the models. For example, Arora et al. (2012b)’s approach for inference in topic models assume that each topic has a unique “anchor” word (thus, we call this approach anchor). This approach is fast and effective; because it only uses word co-occurrence information, it can scale to much larger datasets than MCMC or EM alternatives. We review the anchor method in Section 2.

Despite their advantages, these techniques are not a panacea. They do not accommodate the rich priors that modelers have come to expect. Priors can improve performance (Wallach et al., 2009), provide domain adaptation (Daumé III, 2007; Finkel and Manning, 2009), and guide models to reflect users’ needs (Hu et al., 2013). In Section 3, we regularize the anchor method to trade-off the reconstruction fidelity with the penalty terms that mimic Gaussian and Dirichlet priors.

Another shortcoming is that these models have not been scrutinized using standard NLP evaluations. Because these approaches emerged from the theory community, anchor’s evaluations, when present, typically use training reconstruction. In Section 4, we show that our regularized models can generalize to previously unseen data—as measured by held-out likelihood (Blei et al., 2003)—and are more interpretable (Chang et al., 2009; Newman et al., 2010). We also show that our extension to the anchor method enables new applications: for
From Anchors to Topics

After we have the anchor words, we need to find the coefficients that best explain the data. The difference between anchor and conventional inference is that while conventional methods take a collection of documents as input, anchor takes word co-occurrence statistics. Given a vocabulary of size $V$, we represent this joint distribution as $Q_{i,j} = p(w_1 = i, w_2 = j)$, each cell represents the probability of words appearing together in a document.

Like other topic modeling algorithms, the output of the anchor method is the topic word distributions $A$ with size $V \times K$, where $K$ is the total number of topics desired, a parameter of the algorithm. The $k^{th}$ column of $A$ will be the topic distribution over all words for topic $k$, and $A_{w,k}$ is the probability of observing type $w$ given topic $k$.

Anchors: Topic Representatives

The anchor method (Arora et al., 2012a) is based on the separability assumption (Donoho and Stodden, 2003), which assumes that each topic contains at least one namesake “anchor word” that has non-zero probability only in that topic. Intuitively, this means that each topic has unique, specific word that, when used, identifies that topic. For example, while “run”, “base”, “fly”, and “shortstop” are associated with a topic about baseball, only “shortstop” is unambiguous, so it could serve as this topic’s anchor word.

Let’s assume that we knew what the anchor words were: a set $S$ that indexes rows in $Q$. Now consider the conditional distribution of word $i$, the probability of the rest of the vocabulary given an observation of word $i$; we represent this as $Q_{i,:}$, as we can construct this by normalizing the rows of $Q$. For an anchor word $s_a \in S$, this will look like a topic; $Q_{\cdot \cdot s_a}$, will have high probability for words associated with baseball.

The key insight of the anchor algorithm is that the conditional distribution of polysemous non-anchor words can be reconstructed as a linear combination of the conditional distributions of anchor words. For example, $Q_{\cdot \cdot \text{fly}}$, could be reconstructed by combining the anchor words “insecta”, “boeing”, and “shortshop”. We represent the coefficients of this reconstruction as a matrix $C$, where $C_{i,k} = p(z = k | w = i)$. Thus, for any word $i$,

$$Q_{i,:} \approx \sum_{s_a \in S} C_{i,k} Q_{s_a,:}$$

The coefficient matrix is not the usual output of a topic modeling algorithm. The usual output is the probability of a word given a topic. The coefficient matrix $C$ is the probability of a topic given a word. We use Bayes rule to recover the topic distribution $p(w = i | z = k) \equiv A_{i,k} \propto p(z = k | w = i) p(w = i) = C_{i,k} \sum_j \overline{Q}_{i,j}$

where $p(w)$ is the normalizer of $Q$ to obtain $Q_{w,:}$.

The geometric argument for finding the anchor words is one of the key contributions of Arora et al. (2012a) and is beyond the scope of this paper. The algorithms in Section 3 use the anchor selection subroutine unchanged. The difference in our approach is in how we discover the anchor coefficients $C$.

From Anchors to Topics

After we have the anchor words, we need to find the coefficients that
best reconstruct the data \( \bar{Q} \) (Equation 1). Arora et al. (2012a) chose the \( C \) that minimizes the KL divergence between \( Q_{i,*} \) and the reconstruction based on the anchor word’s conditional word vectors \( \sum_{s_k \in S} C_{i,k} Q_{s_k,*} \).

\[
C_{i,*} = \text{argmin}_{C_{i,*}} D_{\text{KL}} \left( Q_{i,*} \| \sum_{s_k \in S} C_{i,k} Q_{s_k,*} \right).
\]  

(3)

The anchor method is fast, as it only depends on the size of the vocabulary once the co-occurrence statistics \( Q \) are obtained. However, it does not support rich priors for topic models, while MCMC (Griffiths and Steyvers, 2004) and variational EM (Blei et al., 2003) methods can. This prevents models from using priors to guide the models to discover particular themes (Zhai et al., 2012), or to encourage sparsity in the models (Yao et al., 2009). In the rest of this paper, we correct this lacuna by adding regularization inspired by Bayesian priors to the anchor algorithm.

3 Adding Regularization

In this section, we add regularizers to the anchor objective (Equation 3). In this section, we briefly review regularizers and then add two regularizers, inspired by Gaussian (\( L_2 \), Section 3.1) and Dirichlet priors (Beta, Section 3.2), to the anchor objective function (Equation 3).

Regularization terms are ubiquitous. They typically appear as an additional term in an optimization problem. Instead of optimizing a function just of the data \( x \) and parameters \( \beta, f(x, \beta) \), one optimizes an objective function that includes a regularizer that is only a function of parameters: \( f(w, \beta) + r(\beta) \). Regularizers are critical in staid methods like linear regression (Ng, 2004), in workhorse methods such as maximum entropy modeling (Dudík et al., 2004), and also in emerging fields such as deep learning (Wager et al., 2013).

In addition to being useful, regularization terms are appealing theoretically because they often correspond to probabilistic interpretations of parameters. For example, if we are seeking the MLE of a probabilistic model parameterized by \( \beta, p(x|\beta) \), adding a regularization term \( r(\beta) = \sum_{i=1}^{L} \beta_i^2 \) corresponds to adding a Gaussian prior

\[
f(\beta_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{\beta_i^2}{2\sigma^2} \right\}
\]  

(4)

and maximizing log probability of the posterior (ignoring constant terms) (Rennie, 2003).

### 3.1 \( L_2 \) Regularization

The simplest form of regularization we can add is \( L_2 \) regularization. This is similar to assuming that probability of a word given a topic comes from a Gaussian distribution. While the distribution over topics is typically Dirichlet, Dirichlet distributions have been replaced by logistic normals in topic modeling applications (Blei and Lafferty, 2005) and for probabilistic grammars of language (Cohen and Smith, 2009).

Augmenting the anchor objective with an \( L_2 \) penalty yields

\[
C_{i,*} = \text{argmin}_{C_{i,*}} D_{\text{KL}} \left( Q_{i,*} \| \sum_{s_k \in S} C_{i,k} Q_{s_k,*} \right) + \lambda \| C_{i,*} - \mu_{i,*} \|^2_2,
\]  

(5)

where regularization weight \( \lambda \) balances the importance of a high-fidelity reconstruction against the regularization, which encourages the anchor coefficients to be close to the vector \( \mu \). When the mean vector \( \mu \) is zero, this encourages the topic coefficients to be zero. In Section 4.3, we use a non-zero mean \( \mu \) to encode an informed prior to encourage topics to discover specific concepts.

### 3.2 Beta Regularization

The more common prior for topic models is a Dirichlet prior (Minka, 2000). However, we cannot apply this directly because the optimization is done on a row-by-row basis of the anchor coefficient matrix \( C \), optimizing \( C \) for a fixed word \( w \) for and all topics. If we want to model the probability of a word, it must be the probability of word \( w \) in a topic versus all other words.

Modeling this dichotomy (one versus all others in a topic) is possible. The constructive definition of the Dirichlet distribution (Sethuraman, 1994) states that if one has a \( V \)-dimensional multinomial \( \theta \sim \text{Dir}(\alpha_1 \ldots \alpha_V) \), then the marginal distribution

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Train</th>
<th>Dev</th>
<th>Test</th>
<th>Vocab</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIPS</td>
<td>1231</td>
<td>247</td>
<td>262</td>
<td>12182</td>
</tr>
<tr>
<td>20NEWS</td>
<td>11243</td>
<td>3760</td>
<td>3726</td>
<td>81604</td>
</tr>
<tr>
<td>NYT</td>
<td>9255</td>
<td>2012</td>
<td>1959</td>
<td>34940</td>
</tr>
</tbody>
</table>
of \( \theta_w \) follows \( \theta_w \sim \text{Beta}(\alpha_w, \sum_{i \neq w} \alpha_i) \). This is the tool we need to consider the distribution of a single word’s probability.

This requires including the topic matrix as part of the objective function. The topic matrix is a linear transformation of the coefficient matrix (Equation 2). The objective for beta regularization becomes

\[
C_{i, \cdot} = \text{argmin}_{C_{i, \cdot}} D_{KL} \left( Q_{i, \cdot} \mid \mid \sum_{s_k \in S} C_{i,k} Q_{k, \cdot} \right) - \lambda \sum_{s_k \in S} \log(\text{Beta}(A_{i,k}; a, b)), \tag{6}
\]

where \( \lambda \) again balances reconstruction against the regularization. To ensure the tractability of this algorithm, we enforce a convex regularization function, which requires that \( a > 1 \) and \( b > 1 \). If we enforce a uniform prior—\( \mathbb{E}_{\text{Beta}(a,b)}[A_{i,k}] = \frac{1}{V} \)—and that the mode of the distribution is also \( \frac{1}{V} \),

\[
a = \frac{x}{V} + 1, \quad \text{and} \quad b = \frac{(V - 1)x}{V} + 1 \tag{7}
\]

for real \( x \) greater than zero.

### 3.3 Initialization and Convergence

Equation 5 and Equation 6 are optimized using L-BFGS gradient optimization (Galassi et al., 2003). We initialize \( C \) randomly from \( \text{Dir}(\alpha) \) with \( \alpha = \frac{1}{V} \) (Wallach et al., 2009). We update \( C \) after optimizing all \( V \) rows. The newly updated \( C \) replaces the old topic coefficients. We track how much the topic coefficients \( \Delta C \) change between two consecutive iterations \( i \) and \( i + 1 \) and represent it as \( \Delta C = \|C^{i+1} - C^i\|_2 \). We stop optimization when \( \Delta C \leq \delta \). When \( \delta = 0.1 \), the \( L_2 \) and unregularized anchor algorithm converges after a single iteration, while beta regularization typically converges after fewer than ten iterations (Figure 4).

### 4 Regularization Improves Topic Models

In this section, we measure the performance of our proposed regularized anchor word algorithms. We will refer to specific algorithms in bold. For example, the original anchor algorithm is \textbf{anchor}. Our \( L_2 \) regularized variant is \textbf{anchor-L2}, and our beta regularized variant is \textbf{anchor-beta}. To provide conventional baselines, we also compare our methods against topic models from variational inference (Blei et al., 2003, \textit{variational}) and MCMC (Griffiths and Steyvers, 2004; McCallum, 2002, \textit{MCMC}).

We apply these inference strategies on three diverse corpora: scientific articles from the Neural Information Processing Society (NIPS), Internet newsgroups postings (20NEWS), and New York Times editorials (Sandhaus, 2008, NYT). Statistics for the datasets are summarized in Table 2. We split each dataset into a training fold (70%), development fold (15%), and a test fold (15%): the training data are used to fit models; the development set are used to select parameters (anchor threshold \( M \), document prior \( \alpha \), regularization weight \( \lambda \)); and final results are reported on the test fold.

We use two evaluation measures, held-out likelihood (Blei et al., 2003, \textit{HL}) and topic interpretability (Chang et al., 2009; Newman et al., 2010, \textit{T1}). Held-out likelihood measures how well the model can reconstruct held-out documents that the model has never seen before. This is the typical evaluation for probabilistic models. Topic interpretability is a more recent metric to capture how useful the topics can be to human users attempting to make sense of a large datasets.

Held-out likelihood cannot be computed with existing \textbf{anchor} algorithms, so we use the topic distributions learned from \textbf{anchor} as input to a reference variational inference implementation (Blei et al., 2003) to compute \textit{HL}. This requires an additional parameter, the Dirichlet prior \( \alpha \) for the per-document distribution over topics. We select \( \alpha \) using grid search on the development set.

To compute \textit{T1} and evaluate topic coherence, we use normalized pairwise mutual information (NPMI) (Lau et al., 2014) over topics’ twenty most probable words. Topic coherence is computed against the NPMI of a reference corpus. For coherence evaluations, we use both intrinsic and extrinsic text collections to compute NPMI. Intrinsic coherence (Ti-i) is computed on training and development data at development time and on training and test data at test time. Extrinsic coherence (Ti-e) is computed from English Wikipedia articles, with disjoint halves (1.1 million pages each) for distinct development and testing Ti-e evaluation.

\textsuperscript{1}For \( a, b < 1 \), the expected value is still the uniform distribution but the mode lies at the boundaries of the simplex. This corresponds to a sparse Dirichlet distribution, which our optimization cannot at present model.
Evaluating Regularization

With document frequency $M$ and regularization weight $\lambda$ selected from the development set, we will explain other words’ cooccurrence poorly because all regularizations benefit equally from higher-quality anchor words, we use cross-validation to select the document frequency cutoff $M$ using the unregularized anchor algorithm. Figure 1 shows the performance of anchor with different $M$ on our three datasets with 20 topics for our two measures $HL$ and $TI$.

Because all regularizations benefit equally from higher-quality anchor words, we use cross-validation to select the document frequency cutoff $M$ using the unregularized anchor algorithm. Figure 1 shows the performance of anchor with different $M$ on our three datasets with 20 topics for our two measures $HL$ and $TI$.

Figure 1: Grid search for document frequency $M$ for our datasets with 20 topics (other configurations not shown) on development data. The performance on both $HL$ and $TI$ score indicate that the unregularized anchor algorithm is very sensitive to $M$. The $M$ selected here is applied to subsequent models.

Figure 2: Selection of $\lambda$ based on $HL$ and $TI$ scores on the development set. The value of $\lambda = 0$ is equivalent to the original anchor algorithm; regularized versions find better solutions as the regularization weight $\lambda$ becomes non-zero.

4.1 Grid Search for Parameters on Development Set

Anchor Threshold A good anchor word must have a unique, specific context but also explain other words well. A word that appears only once will have a very specific cooccurrence pattern but will explain other words’ cooccurrence poorly because the observations are so sparse. As discussed in Section 2, the anchor method uses document frequency $M$ as a threshold to only consider words with robust counts.

Because all regularizations benefit equally from higher-quality anchor words, we use cross-validation to select the document frequency cutoff $M$ using the unregularized anchor algorithm. Figure 1 shows the performance of anchor with different $M$ on our three datasets with 20 topics for our two measures $HL$ and $TI$.

Regularization Weight Once we select a cutoff $M$ for each combination of dataset, number of topics $K$ and a evaluation measure, we select a regularization weight $\lambda$ on the development set. Figure 2 shows that beta regularization framework improves topic interpretability $TI$ on all datasets and improved the held-out likelihood $HL$ on 20NEWS. The $L_2$ regularization also improves held-out likelihood $HL$ for the 20NEWS corpus (Figure 2).

In the interests of space, we do not show the figures for selecting $M$ and $\lambda$ using $TI$-$e$, which is similar to $TI$-$i$: anchor-beta improves $TI$-$e$ score on all datasets, anchor-$L_2$ improves $TI$-$e$ on 20NEWS and NIPS with 20 topics and NYT with 40 topics.

4.2 Evaluating Regularization

With document frequency $M$ and regularization weight $\lambda$ selected from the development set, we
compare the performance of those models on the test set. We also compare with standard implementations of Latent Dirichlet Allocation: Blei’s LDAC (variational) and Mallet (mcmc). We run 100 iterations for LDAC and 5000 iterations for Mallet.

Each result is averaged over three random runs and appears in Figure 3. The highly-tuned, widely-used implementations uniformly have better held-out likelihood than anchor-based methods, but the much faster anchor methods are often comparable. Within anchor-based methods, $L_2$-regularization offers comparable held-out likelihood as unregularized anchor, while anchor-beta often has better interpretability. Because of the mismatch between the specialized vocabulary of NIPS and the general-purpose language of Wikipedia, TI-e has a high variance.

4.3 Informed Regularization

A frequent use of priors is to add information to a model. This is not possible with the existing anchor method. An informed prior for topic models seeds a topic with words that describe a topic of interest. In a topic model, these seeds will serve as a “magnet”, attracting similar words to the topic (Zhai et al., 2012).

We can achieve a similar goal with anchor-$L_2$. Instead of encouraging anchor coefficients to be zero in Equation 5, we can instead encourage word probabilities to close to an arbitrary mean $\mu_{i,k}$. This vector can reflect expert knowledge.

One example of a source of expert knowledge is Linguistic Inquiry and Word Count (Pennebaker and Francis, 1999, LIWC), a dictionary of keywords related to sixty-eight psychological concepts such as positive emotions, negative emotions, and death. For example, it associates “excessive, estate, money, cheap, expensive, living, profit, live, rich, income, poor, etc.” for the concept materialism.

We associate each anchor word with its closest LIWC category based on the cooccurrence matrix $Q$. This is computed by greedily finding the anchor word that has the highest cooccurrence score for any LIWC category: we define the score of a category to anchor word $w_{a_i}$ as $\sum_i Q_{s_k,i}$, where $i$ ranges over words in this category; we compute the scores of all categories to all anchor words; then we find the highest score and assign the category to that anchor word; we greedily repeat this process until all anchor words have a category.

Given these associations, we create a goal mean $\mu_{i,k}$. If there are $L_i$ anchor words associated with LIWC word $i$, $\mu_{i,k} = \frac{1}{L_i}$ if this keyword $i$ is associated with anchor word $w_{s_k}$ and zero otherwise.

We apply anchor-$L_2$ with informed priors on NYT with twenty topics and compared the topics against the original topics from anchor. Table 3 shows that the topic with anchor word “soviet”, when combined with LIWC, draws in the new words “bush” and “nuclear”; reflecting the threats of force during the cold war. For the topic with topic word “arms”, when associated with the LIWC category with the terms “agree” and “agreement”, draws in “clinton”, who represented a more conciliatory foreign policy compared to his republican predecessors.

5 Discussion

Having shown that regularization can improve the anchor topic modeling algorithm, in this section we discuss why these regularizations can improve the model and the implications for practitioners.

Efficiency Efficiency is a function of the number of iterations and the cost of each iteration. Both anchor and anchor-$L_2$ require a single iteration, although the latter’s iteration is slightly more expensive. For beta, as described in Section 3.2, we update anchor coefficients $C$ row by row, and then repeat the process over several iterations until it converges. However, it often converges within ten iterations (Figure 4) on all three datasets: this requires much fewer iterations than MCMC or variational inference, and the iterations are less expensive. In addition, since we optimize each row $C_{i}$ independently, the algorithm can be easily parallelized.

Sensitivity to Document Frequency While the original anchor is sensitive to the document frequency $M$ (Figure 1), adding regularization makes this less critical. Both anchor-$L_2$ and anchor-beta are less sensitive to $M$ than anchor.

To highlight this, we compare the topics of anchor and anchor-beta when $M = 100$. As Table 4 shows, the words “article”, “write”, “don” and “doe” appear in most of anchor’s topics. While anchor-$L_2$ also has some bad topics, it still can find reasonable topics, demonstrating anchor-beta’s greater robustness to suboptimal $M$.

$L_2$ (Sometimes) Improves Generalization As Figure 2 shows, anchor-$L_2$ sometimes improves held-out development likelihood for the smaller
Algorithm ● anchor anchor−beta anchor−L2 MCMC variational
20NEWS ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●●
<table>
<thead>
<tr>
<th>Topic</th>
<th>anchor</th>
<th>anchor-beta</th>
</tr>
</thead>
<tbody>
<tr>
<td>frequently</td>
<td>article write don doe make time people good file question</td>
<td>article write don doe make people time good email file</td>
</tr>
<tr>
<td>debate</td>
<td>write article people make don doe god key government time</td>
<td>people make god article write don doe key point government</td>
</tr>
<tr>
<td>wings</td>
<td>game team write wings article win red play hockey year</td>
<td>game team wings win red hockey play season player fan</td>
</tr>
<tr>
<td>stats</td>
<td>player team write game article stats year good play doe</td>
<td>stats player season league baseball fan team individual playoff nhl</td>
</tr>
<tr>
<td>compile</td>
<td>program file write email doe windows call problem run don</td>
<td>compile program code file ftp advance package error windows sun</td>
</tr>
</tbody>
</table>

Table 4: Topics from anchor and anchor-beta with $M = 100$ on 20NEWS with 20 topics. Each topic is identified with its associated anchor word. When $M = 100$, the topics of anchor suffer: the four colored words appear in almost every topic. anchor-beta, in contrast, is less sensitive to suboptimal $M$.

Figure 4: Convergence of anchor coefficient $C$ for anchor-beta. $\Delta C$ is the difference of current $C$ from the $C$ at the previous iteration. $C$ is converged within ten iterations for all three datasets.

and “trade” in the regularized topic. These words are less ambiguous and more likely to make sense to a consumer of topic models.

Figure 5 shows why this happens. Compared to the unregularized topics from anchor, the beta regularized topic steals from the rich and creates a more uniform distribution. Thus, highly frequent words do not as easily climb to the top of the distribution, and the topics reflect topical, relevant words rather than globally frequent terms.

6 Conclusion

A topic model is a popular tool for quickly getting the gist of large corpora. However, running such an analysis on these large corpora entail a substantial computational cost. While techniques such as anchor algorithms offer faster solutions, it comes at the cost of the expressive priors common in Bayesian formulations.

This paper introduces two different regularizations that offer users more interpretable models and the ability to inject prior knowledge without sacrificing the speed and generalizability of the underlying approach. However, one sacrifice that this approach does make is the beautiful theoretical guarantees of previous work. An important piece of future work is a theoretical understanding of generalizability in extensible, regularized models.

Incorporating other regularizations could further improve performance or unlock new applications. Our regularizations do not explicitly encourage sparsity; applying other regularizations such as $L_1$ could encourage true sparsity (Tibshirani, 1994), and structured priors (Andrzejewski et al., 2009) could efficiently incorporate constraints on topic models.

These regularizations could improve spectral algorithms for latent variables models, improving the performance for other NLP tasks such as latent variable PCFGs (Cohen et al., 2013) and HMMs (Anandkumar et al., 2012), combining the flexibility and robustness offered by priors with the speed and accuracy of new, scalable algorithms.

Acknowledgments

We would like to thank the anonymous reviewers, Hal Daumé III, Ke Wu, and Ke Zhai for their helpful comments. This work was supported by NSF Grant IIS-1320538. Boyd-Graber is also supported by NSF Grant CCF-1018625. Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the view of the sponsor.
Figure 5: How beta regularization influences the topic distribution. Each topic is identified with its associated anchor word. Compared to the unregularized anchor method, anchor-beta steals probability mass from the “rich” and prefers a smoother distribution of probability mass. These words often tend to be unimportant, polysemous words common across topics.

Table 5: Comparing topics—labeled by their anchor word—from anchor and anchor-beta. With beta regularization, relevant words are promoted, while more general words are suppressed, improving topic coherence.

References


