Sparse Word Graphs: A Scalable Algorithm for Capturing Word Correlations in Topic Models

Ramesh Nallapati, Amr Ahmed, William Cohen and Eric Xing
Machine Learning Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213, USA
{nmramesh,amahmed,wcohen,epxing}@cs.cmu.edu

Abstract

Statistical topic models such as the Latent Dirichlet Allocation (LDA) have emerged as an attractive framework to model, visualize and summarize large document collections in a completely unsupervised fashion. One of the limitations of this family of models is their assumption of exchangeability of words within documents, which results in a 'bag-of-words' representation for documents as well as topics. As a consequence, precious information that exists in the form of correlations between words is lost in these models.

In this work, we adapt recent advances in sparse modeling techniques to the problem of modeling word correlations within topics and present a new algorithm called **Sparse Word Graphs**. *Our experiments on AP corpus reveal both long-distance and short-distance word correlations within topics that are semantically very meaningful. In addition, the new algorithm is highly scalable to large collections as it captures only the most important correlations in a sparse manner.*

"ARTS"	"BUDGET"	"CHILDREN"	"EDUCATION"
New	Million	Children	School
Film	Program	Women	Students
Show	Tax	People	Schools
Music	Budget	Child	Education
Movie	Billion	Years	Teachers
Play	Federal	Families	High
Musical	Year	Work	Public
Best	Spending	Parent	Teacher
Actor	New	Says	Bennett
First	State	Family	Manigat
York	Plan	Welfare	Namphy
Opera	Money	Men	State
Theater	Programs	Percent	President
Actress	Government	Care	Elementary
Love	Congress	Life	Haiti

Figure 1. Most likely words from 4 topics in LDA from the AP corpus: the topic titles in quotes are not part of the algorithm. (Courtesy: Blei *et al*, [5])

1 Introduction

In the recent past, statistical topic modeling has become very popular as a completely unsupervised method to help summarize and visualize the contents of large document collections [5, 7, 9, 3, 6, 14]. These models use simple surface features such as word occurrences within documents to reveal suprisingly meaningful semantic content of documents in terms of multinomial distributions over the vocabulary, known as 'topics' [4]. The basic version of this family of models is called *Latent Dirichlet Allocation* (LDA) [5]. Some of the topics discovered automatically by LDA from the AP corpus are displayed in figure 1.

In the recent past, there have been several extensions to LDA. Notable among them are the Dynamic Topic Model, [6] which models the evolution of topic content with time; HMM-LDA [7] in which semantic analysis of LDA is combined with the syntax analysis of HMMs; Pachinko Allocation, [9] that models a hierarchy of topics; Correlated Topic Model [3] which captures the correlations between topics by replacing the Dirichlet prior with a logistic-normal distribution; and finally the Dirichlet Process Mixture Model [2] which discovers the number of topics K automatically.

Despite their additional features in comparison to LDA, one component remains the same among all these models, namely, *exchangeability* of words. In other words, word occurrences are treated conditionally independent of each other, given their topics. In information retrieval parlance, this is referred to as modeling documents as "bags of words". In doing so, much of the valuable information that exisits in terms of correlations between words is lost. For example, a topic representation that simply assigns high weight to the words 'white' and 'house', while completely ignoring the phrase 'white-house' may not immediately reveal to the user that the topic is about the president

of the United States. Apart from phrases, there could also be interesting long-distance correlations that occur between words. However, the existing topic models completely ignore both phrasal as well as long-distance correlations between words. Capturing such relationships explicitly in the model would go a long way in better visualization and summarization of topics. The aim of this work is to capture such correlations between words in topic models. We present a new scalable algorithm called *Sparse Word Graphs* that addresses this problem.

The rest of the paper is organized as follows. In section 2, we present some background and the new *Sparse Word Graphs* algorithm that casts the word correlations problem as that of structure learning problem in Markov Random Fields and applies the l_1 norm minimization technique to solve the problem. Section 3 presents some of the sparse word graphs generated for a few representative topics and compares them with the LDA topic representation. In section 4, we compare and contrast the new algorithm with other word-correlation models and finally chart out directions for future work in section 5.

2 Sparse Word Graphs

2.1 Background

Wainwright *et al* [13] showed how to estimate a sparse graph structure of a discrete pairwise Markov Random Field (MRF) wherein, the neighborhood of any vertex in the graph is estimated by performing an l_1-regularized logistic regression on the rest of the vertices. Their algorithm is as follows:

Let $G = (\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ of size $|\mathcal{V}| = p$ and edge set $\mathcal{E}$. Let $\mathbf{X} = (X_1, \cdots, X_p)$ be a set of binary random variables associated with the vertices of the graph. Let the joint probability of the random variables be given by the Ising model as follows:

$$P(\mathbf{x}|\boldsymbol{\lambda}) = \exp(\sum_{s \in \mathcal{V}} \lambda_s x_s + \sum_{(s,t) \in \mathcal{E}} \lambda_{st} x_s x_t - A(\boldsymbol{\lambda})) \quad (1)$$

where the parameters $\{\lambda_{st}\}_{(s,t) \in \mathcal{E}}$ capture the correlation between the variables X_s and X_t. $A(\boldsymbol{\lambda})$ is the log-normalizing constant of the distribution. Given n samples $(\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(n)})$ such that each $\mathbf{x}^{(i)} \in \{0,1\}^p$ drawn from an unknown distribution $P(\mathbf{x}|\boldsymbol{\lambda}^*)$, the goal is to estimate the structure of the graph, that is to estimate $\hat{\mathcal{E}}$ such that $\lim_{n \to \infty} P(\hat{\mathcal{E}} = \mathcal{E}) = 1$.

The authors show that the following algorithm asymptotically converges to the true structure as the data size n increases: we maximize the l_1 regularized conditional likelihood of each variable X_s conditioned on all the other variables $\mathbf{X}_{-s}$. For a pairwise MRF with binary variables, this leads to the following l_1 regularized logistic regression problem:

$$\begin{aligned} \hat{\boldsymbol{\lambda}}_s &= \arg\max_{\boldsymbol{\lambda}_s} \sum_{i=1}^{n} \log P(x_s^{(i)} | \mathbf{x}_{-s}^{(i)}, \boldsymbol{\lambda}_s) - \rho \|\boldsymbol{\lambda}_{-s}\|_1 \\ &= \arg\max_{\boldsymbol{\lambda}_s} \sum_{i=1}^{n} x_s^{(i)} \boldsymbol{\lambda}_s^T \mathbf{x}_{-s}^{(i)} - \log(1 + \exp(\boldsymbol{\lambda}_s^T \mathbf{x}_{-s}^{(i)})) \\ &\quad -\rho \|\boldsymbol{\lambda}_{-s}\|_1 \end{aligned} \quad (2)$$

where $\boldsymbol{\lambda}_s = (\lambda_{s1}, \cdots, \lambda_{sp})$ are the parameters of the l_1 logistic regression and $\mathbf{x}_{-s}$ denotes the set of all variables with x_s replaced by 1 while $\boldsymbol{\lambda}_{-s}$ denotes the vector $\boldsymbol{\lambda}_s$ with the component λ_{ss} removed. Similar to the case of Gaussian Random Fields, the estimated set of neighbors is given by:

$$\hat{\mathcal{N}}(s) = \{t : \hat{\lambda_{st}} \neq 0\} \quad (3)$$

One could then define the set of edges $\mathcal{E}$ as a union or an intersection of neighborhood sets $\{\mathcal{N}(s)\}_{s \in \mathcal{V}}$ of all the vertices. Wainwright *et al* [13] showed that both definitions would converge to the true structure asymptotically.

In the next subsection, we will cast our problem as that of structure learning of a pairwise Markov Random field, which allows us to apply the algorithm of Wainwright *et al* [13] described above.

2.2 Algorithm

In order to be able to define a binary pairwise MRF structure learning problem, we first convert the topic assignments generated by LDA into topic-specific binary data as follows:

The Latent Dirichlet model assigns a topic to each word-occurrence in a document. Given a document collection, these latent topic assignments to all the words can be computed using a variational algorithm or Gibbs sampling. The starting point of our algorithm is the topic-assignment data generated by LDA. In LDA, each document d is represented as a vector of words $\mathbf{w}_d = (w_1, \cdots, w_{N_d})$, where $w_i \in \{1, \cdots, V\}$ is one of the V unique words in the vocabulary and N_d is the document length. LDA generates corresponding topic assignment vector $\mathbf{z}_d = (z_1, \cdots, z_{N_d})$ where each $z_i \in \{1, \cdots, K\}$ is one of K topics.

First, we convert the LDA topic-assignment vector $\mathbf{z}_d$ to a set of K binary vectors $\{\mathbf{x}_k^{(d)}\}_{k=1}^K$, where each $\mathbf{x}_k^{(d)} = (x_{k1}^{(d)}, \cdots, x_{kV}^{(d)})$ is of length V as follows:

$$\begin{aligned} &\forall\ k = 1, \cdots, K;\ \ v = 1, \cdots, V \\ &x_{kv}^{(d)} = \begin{cases} 1 & \text{if } \exists\, j \in \{1, \cdots, N_d\} \\ & \quad \text{s.t. } w_j = v \text{ and } z_j = k \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (4)$$

In other words, each variable $x_{kv}^{(d)}$ associated with a word v for a topic k is assigned a value of 1 if the word occurs

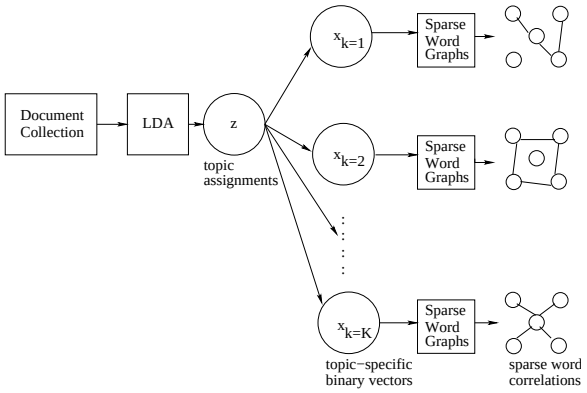

Figure 2. Flow chart of the Sparse Word Graphs algorithm

in the document and is assigned the topic k by LDA and 0 otherwise. Note that each word can occur multiple times in a document. Technically, LDA permits a different topic assignment to each occurrence of the word in the document. However it turns out that, in the maximum likelihood estimation setting, LDA assigns the same topic to all occurrences of a given word. Hence the assignment of $x_{kv}^{(d)}$ in eqn. (4) is a well-defined function.

Now we assume that each topic k is associated with a random vector $\mathbf{X}_k = (X_{k1}, \cdots, X_{kV})$ whose joint probability is defined under the following pairwise Markov Random Field defined as follows:

$$P(\mathbf{x}_k|\boldsymbol{\lambda}_k) = \exp(\sum_{s\in\mathcal{V}} \lambda_{ks}x_{ks} + \sum_{(s,t)\in\mathcal{E}} \lambda_{kst}x_{ks}x_{kt} - A(\boldsymbol{\lambda}_k)) \tag{5}$$

where the vertices $\mathcal{V}$ in the MRF correspond to the words in the vocabulary with an unknown sparse edge structure $\mathcal{E}$. The task is now to learn the sparse structure $\mathcal{E}$ of the topic-specific MRF using the data $\{\mathbf{x}_k^{(d)}\}_{d=1}^M$ where M is the number of documents in the collection. It is clear that one can directly apply the algorithm of Wainwright *et al* [13] to this problem.

We run V l_1 regularized logistic regression problems for each topic as shown in table 1. We define the set of edges as the union of all the neighborhoods of all vertices. As a heuristic, we also estimate the strength of the correlation between two words as the sum of the two parameter values obtained from the logistic regression problems corresponding to the two words, as shown in table 1. The algorithm is also illustrated in the form of a flow chart in figure 2.

2.3 Scalability

As described above, for each topic, we run V l_1 regularized logistic regression problems, each of which is of size V. Hence it appears that the complexity of the problem is still $\mathcal{O}(V^2K)$. This is technically true in terms of a loose upper-bound, but in practice, the new algorithm is very efficient in terms of both computational time as well as storage costs compared to a traditional bigram model for the following reasons:

1. Although the size of data vectors $\mathbf{x}_k^{(d)}$ is V, the number of non-zero components is strictly upper-bounded by $N_{d_{max}}$, the maximum document length in the collection, which is typically much less than V. Thus, the input data to logistic regression is exteremely sparse, making the learning very efficient.

2. A bigram model typically needs to estimate and store $KV(V-1)$ parameters. The new algorithm estimates and stores only the edge weights of the sparse structure, which needs much less computation and smaller storage space in practice.

3. Since each of the problems is independent, it is possible to run the V problems in parallel, resulting in a speed-up of computation.

4. Recent work by Koh *et al* [8] proposed a new, fast interior point solution for the l_1-regularized logistic regression problem, which makes it very scalable and practical for large dimensional problems.

3 Experiments

For our experiments, we used the small AP corpus[1] consisting of $M = 2,246$ documents and $V = 10,473$ unique words. We ran a 10 topic LDA model [2] on this document set to obtain the topic-assignment data $\{\mathbf{z}_d\}_{d=1}^M$. Next, for each topic, we generated binary data $\{\mathbf{x}_k^{(d)}\}_{d=1}^M$. We filtered out those documents for which the mixing proportion for this topic θ_{dk} is less than 0.25, to remove noisy data. The only parameter in the algorithm is the regularization weight ρ (see table 1) which can be used to control the degree of sparsity: higher values of ρ will result in more sparsity. We used $\rho = 0.1$ in our experiments. Then, for each topic, we ran the fast, scalable, interior point implementation of l_1 regularized logistic regression[3] [8] for each of the 10,473 words and merged the resulting sparse neighborhoods by a union operation. On an average, it took us just about 45 minutes per topic to compute the sparse graph structure on an Intel Xeon 1.86GHz processor with 4GB of RAM.

[1] Downloadable from http://www.cs.princeton.edu/∼blei/lda-c/index.html

[2] We used an efficient C-implementation downloaded from http://www.cs.princeton.edu/∼blei/lda-c/

[3] Downloadable from http://www.stanford.edu/∼boyd/l1_logreg

- For each topic $k \in \{1, \cdots, K\}$
  - **Input:** LDA binary data $\{\mathbf{x}_k^{(d)}\}_{d=1}^M$
  - For each word $v \in \{1, \cdots, V\}$
    - Compute $\boldsymbol{\lambda}_{kv} = \arg\max_{\boldsymbol{\lambda}_{kv}} \sum_{d=1}^M x_{kv}^{(d)} \boldsymbol{\lambda}_{kv}^T \mathbf{x}_{k,-v}^{(d)} - \log(1 + \exp(\boldsymbol{\lambda}_{kv}^T \mathbf{x}_{k,-v}^{(d)})) - \rho \|\boldsymbol{\lambda}_{k,-v}\|_1$
  - For each word pair (v, v') s.t. $\lambda_{kvv'} \neq 0$ or $\lambda_{kv'v} \neq 0$; $\phi_{kvv'} = \lambda_{kvv'} + \lambda_{kv'v}$
  - **Return:** Topic specific edge weights $\boldsymbol{\phi}_k$

Table 1. Sparse Word Graphs Algorithm: $\boldsymbol{\lambda}_{kv} = (\lambda_{kv1}, \cdots, \lambda_{kvV})$ is the parameter vector associated with the l_1 logistic regression problem of word v in topic k, $\mathbf{x}_{k,-v}^{(d)}$ is the binary vector $\mathbf{x}_{kv}^{(d)}$ with $x_{kvv}^{(d)}$ set to 1 and $\boldsymbol{\lambda}_{k,-v}$ is obtained by removing the component λ_{kvv} from the vector $\boldsymbol{\lambda}_{kv}$.

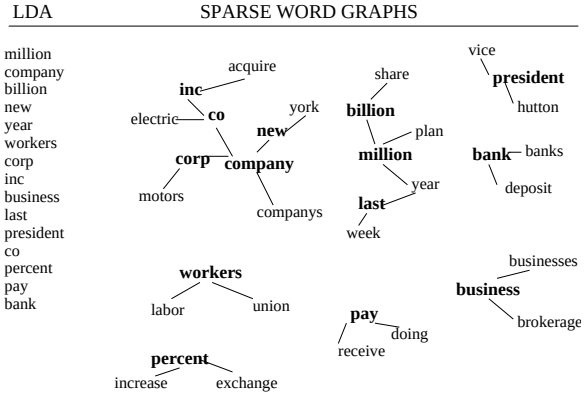

Figure 3. Comparison of LDA topic representation with truncated neighborhoods of top ranking LDA words for topic "Business": top ranked LDA words are bold-faced.

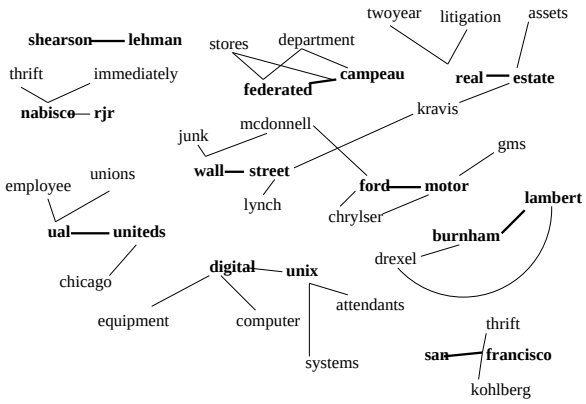

Figure 4. Truncated neighborhoods of top ranked edges in the topic "Business": top ranked edges are displayed in bold-faced with thick lines

We present the sparse word correlation structure of two of the topics we labeled "Business" and "War" respectively in figures 3 through 6. The topic "Business" resulted in 128,074 edges, while the topic " War" has only 35,588 edges (as compared to the total 109,683,729 possible edges). Since it is practically impossible to display all the edges in each topic, we presented two views of each topic. In figures 3 and 5, we presented truncated neighborhoods of the top ranking words in LDA. We define the truncated neighborhood of a word as the top two edges in its neighborhood, where the ranking is done in the descending order of the edge-strength ϕ_k (see table 1). In figures 4 and 6, we display the truncated neighborhoods of the top ranked edges from the full set of edges.

It is clear from figures 3 and 5 that the Sparse Word Graphs representation of topics is more expressive and meaningul than the LDA representation. The Sparse Word Graphs algorithm succeeds in not only capturing phrases

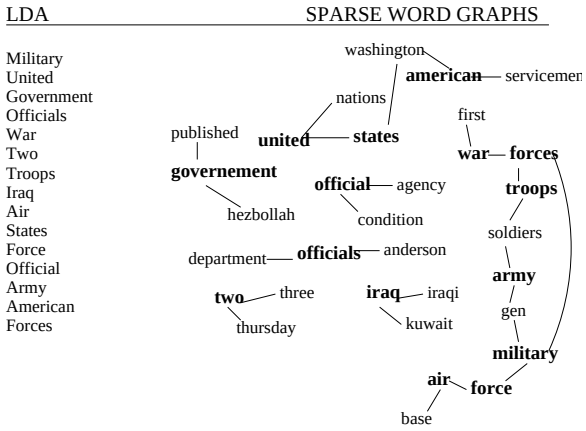

Figure 5. Comparison of LDA topic representation with truncated neighborhoods of top ranking LDA words for topic "War": top ranked LDA words are bold-faced.

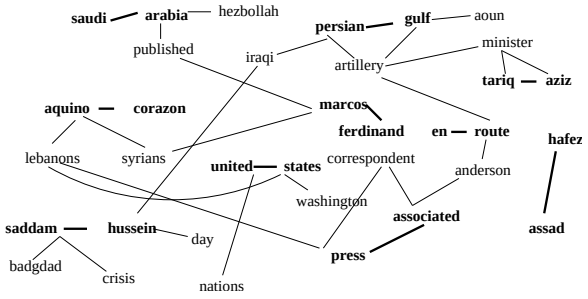

Figure 6. Truncated neighborhoods of top ranked edges in the topic "War": top ranked edges are displayed in bold-faced with thick lines

(such as 'New-York' and 'vice-president' in topic "Business" and 'United-States' and 'air-force' in topic "War"), but also semantically coherent, long distance correlations (such as 'inc-acquire' and 'pay-receive' in "Business" and 'Washington-American' and 'Iraq-Kuwait' in "War"). We also notice that each of the connected components in the figures corresponds to a distinct "concept" within the topic.

The top ranking edges from the two topics and their truncated neighborhoods, displayed in 4 and 6, contain mostly phrases and full names of people. This is not surprising, since these are the word pairs that display strongest correlations within topics. This information is quite useful as well, since a quick glance tells us who are the major players in the respective topics. For example 'Shearson-Lehman', 'Wall-Street', 'Ford-Motor' certainly play a major role in "Business" while 'Saudi-Arabia', 'Saddam-Hussein', United-States', 'Ferdinand-Marcos', 'Contras-Sandinista' are all associated with one war or the other. It is also interesting that the name of an anti-war proponent such as 'Corazon-Aquino' has cropped up in the topic. Also, the truncated neighbors of the top ranked edges exhibit interesting and meaningful associations that are not necessarily phrases (e.g.:' Wall-street-McDonnell', 'UAL-Uniteds-unions', 'Ford-motors-Chrysler', 'San-Francisco-Kohlberg', *etc.* in "Business" and 'Saudi-Arabia-Hezbollah', 'Saddam-Hussein-Baghdad', 'United-States-Washington', 'Tariq-Aziz-minister', *etc.* in "War").

4 Discussion

4.1 Relation to other word-correlation models

We note that there are other approaches in the past that addressed the problem of modeling word-correlations.

Most notable among them is the Hyperspace Analog to Language (HAL) model [11, 10]. This technique models correlation between a pair of words as a weighted count of the number of times they co-occur within a window of fixed length. The weight of each co-occurrence is given by the inverse of the number of words between them. The model enforces sparsity by not considering word pairs that never co-occur within the window length in the entire collection. In [10], the authors show that this algorithm can be implemented on document collections with vocabulary as large as 70,000 words. Our work is very similar to the HAL algorithm in spirit, but the main difference is the following: HAL unearths *global* correlations between words, while *Sparse Word Graphs* can capture topic-specific, *semantic* word correlations. For example, the words 'bank' and 'river' may exhibit high correlation in the topic of "Geography" but will exhibit almost no correlations in "Business" ('bank' is a common word in "Business" but 'river' is not). HAL does not recognize this distinction, but *Sparse Word Graphs* can. Note that *Sparse Word Graphs* can approximately produce HAL output by using document binary vectors as input instead of the LDA topic-assignments data.

Another technique that is similar in spirit to our work is the popular idea of query expansion in information retrieval [1]. In this approach, the original short query from the user is first issued to the database to fetch top ranking documents. Words that highly co-occur with the query words in these documents are returned as candidates for query expansion. When the original short query is replaced by the new expanded query and re-issued to the system, the performance typically improves significantly. One can think of this approach as a *dynamic* version of Sparse Word Graphs, in which the neighborhood of query words in the query-specific topic graph are generated as the output. However, query expansion is not a document summarization tool as it requires the queries (topics) to be pre-specified.

Wordnet[4] is another effort at constructing a semantic network of words. However, this is a completely human supervised effort, and as such is not directly related to our completely unsupervised algorithm.

4.2 Applications of *Sparse Word Graphs*

Our experiments demonstrated that the algorithm captures both short distance correlations such as bigrams and phrases as well as semantically meaningful long distance correlations in topics. Therefore this algorithm serves as a better visualization and summarization tool for document collections than LDA.

The algorithm could also be used for word sense disambiguation. Given a word such as 'bank', one could identify its different senses in terms of its neighborhoods in various topics such as "Geography", "Business", *etc*. Some preliminary work on this idea using the LDA model already shows

[4] http://wordnet.princeton.edu/

promise [12].

Another related application is query expansion: this technique can sometimes be misled by polysemous[5] words. One could use *Sparse Word Graphs* as an intermediate step to disambiguate the query as follows. Using the same running example, if the user types the query 'bank', then the query-expansion algorithm could first specify the neigborhoods of 'bank' from different topics. The user could pick one of neighborhoods, which could then be used to expand the query unambiguously and perform a second retrieval. Alternatively, the system could exploit the session contextual information to automatically pick the right topic and then expand the query based on its neighborhood in the topic.

5 Conclusions and Future Work

We have presented a new algorithm that combines LDA with sparse structural learning methods to successfully capture short and long distance within-topic correlations between words. The algorithm is highly scalable to large collections. For an interested data-mining practitioner, efficient implementations of its components (LDA and l_1 regularized logistic regression: URLs displayed in section 3 in the paper) are readily available.

We however note that, our algorithm is not a unified probabilistic model for capturing within-topic word correlations. Building a comprehensive topic model for this problem is a very challenging and complex problem, since these correlations are not explicitly observed (unlike in a model like HAL), but contained within latent variables called topics. Hence, in this work, we simplified the problem by using a two-step process of running LDA first and then using its output in learning the structure of the sparse MRF for each topic. We believe that our work is a significant first step towards solving this challenging problem of modeling sparse word correlations in the topic modeling framework.

As part of our future work, we hope to be able to construct a unified statistical topic model that addresses this problem. We also intend to evaluate the efficacy of this algorithm on specific tasks such as word-sense disambiguation and query-expansion for information retrieval.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*. Addison Wesley, 1999.

[2] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, 2004.

[3] D. Blei and J. Lafferty. Correlated topic models. In *Advances in Neural Information Processing Systems*, 2006.

[4] D. Blei and J. Lafferty. A correlated topic model of science. *Annals of Applied Statistics*, 2007.

[5] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3, 2003.

[6] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *International conference on Machine learning*, pages 113–120, 2006.

[7] T. L. Griffiths, M. Steyvers, D. M. Blei, and J. B. Tenenbaum. Integrating topics and syntax. In *Advances in Neural Information Processing Systems*, pages 537–544, 2005.

[8] K. Koh, S. Kim, and S. Boyd. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, 2007.

[9] W. Li and A. McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *International conference on Machine learning*, pages 577–584, 2006.

[10] K. Lund, C. Burges, and C. Audet. Dissociating semantic and associative relationships using high-dimensional semantic space. In *Cognitive Science Proceedings*, pages 603–608, 1996.

[11] K. Lund and C. Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Instruments and Computers*, 28:203–208, 1997.

[12] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of Latent Semantic Analysis*, 2007.

[13] M. J. Wainwright, P. Ravikumar, and J. Lafferty. High dimensional graphical model selection using l1-regularized logistic regression. In *Neural Information Processing Systems*, 2006.

[14] H. M. Wallach. Topic modeling: beyond bag-of-words. In *International conference on Machine learning*, pages 977–984, 2006.

[5] words with multiple meanings