Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations

Wei Li and Andrew McCallum {WEILI,MCCALLUM}@CS.UMASS.EDU
University of Massachusetts, Dept. of Computer Science

Abstract

Latent Dirichlet allocation (LDA) and other related topic models are increasingly popular tools for summarization and manifold discovery in discrete data. LDA does not capture correlations between topics, however. Recently Blei and Lafferty have proposed the correlated topic model (CTM) in which the off-diagonal covariance structure in a logistic normal distribution captures pairwise correlations between topics. In this paper we introduce the *Pachinko Allocation model* (PAM), which uses a directed acyclic graph (DAG) to capture arbitrary, nested, and possibly sparse correlations. The leaves of the DAG represent individual words in the vocabulary, while each interior node represents a correlation among its children, which may be words or other interior nodes (topics). Using text data from UseNet, historic NIPS proceedings, and other research paper corpora, we show that PAM improves over LDA in document classification, likelihood of heldout data, ability to support finer-grained topics, and topical keyword coherence.

1. Introduction

Statistical topic models have been successfully used in many areas to analyze large amounts of textual information, including language modeling, document classification, information retrieval, automated document summarization and data mining. In addition to textual data such as newswire articles, research papers and personal emails, topic models have also been applied to images, biological information and other multi-dimensional data.

Latent Dirichlet allocation (LDA) (Blei et al., 2003) is one of the most popular models for extracting topic information from large text corpora. It represents each document as a mixture of topics, where each topic is a multinomial distribution over a word vocabulary. To generate a document, LDA first samples a multinomial distribution over topics from a Dirichlet distribution. Then it repeatedly samples a topic from the multinomial and samples a word from the topic. By applying LDA to a text collection, we are able to organize words into a set of semantically coherent clusters.

Several LDA variations have been proposed to deal with more complicated data structures. For example, the hierarchical LDA model (hLDA) (Blei et al., 2004) assumes a given hierarchical structure among topics. To generate a document, it first samples a topic path from the hierarchy and then samples words from those topics. The advantage of hLDA is the ability to discover topics with various levels of granularity. Another variation of LDA is the HMMLDA model (Griffiths et al., 2005), which combines a hidden Markov model (HMM) with LDA to extract word clusters from sequential data. It distinguishes between syntactic words and semantic words, and simultaneously organizes them into different clusters. HMMLDA has been successfully applied to sequence modeling tasks such as part-of-speech tagging and Chinese word segmentation (Li & McCallum, 2005). Another work related to LDA is the author-topic model by Rosen-Zvi et al. (2004), which associates each author with a mixture of topics. It can be applied to text collections where author information is available, such as research papers.

Topic models like LDA can automatically organize words into different clusters that capture their correlations in the text collection. However, LDA does not directly model the correlations among topic themselves.

Appearing in *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA, 2006. Copyright 2006 by the author(s)/owner(s).

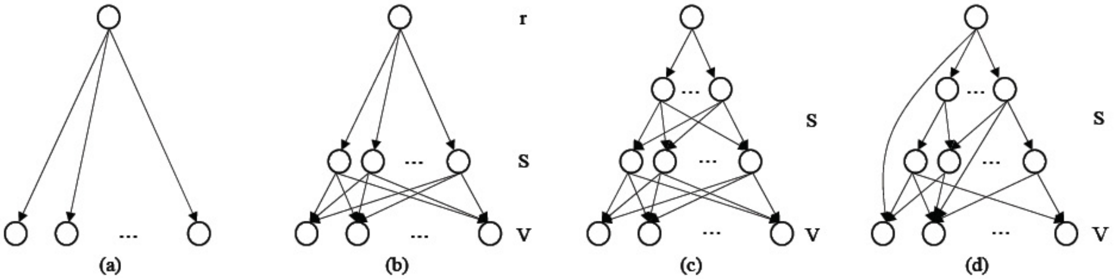


Figure 1. Model Structures for Four Topic Models (a) Dirichlet Multinomial: For each document, a multinomial distribution over words is sampled from a single Dirichlet. (b) LDA: This model samples a multinomial over topics for each document, and then generates words from the topics. (c) Four-Level PAM: A four-level hierarchy consisting of a root, a set of super-topics, a set of sub-topics and a word vocabulary. Both the root and the super-topics are associated with Dirichlet distributions, from which we sample multinomials over their children for each document. (d) PAM: An arbitrary DAG structure to encode the topic correlations. Each interior node is considered a topic and associated with a Dirichlet distribution.

This limitation comes from the assumption that the topic proportions in each document are all sampled from a single Dirichlet distribution. As a result, LDA has difficulty describing a scenario in which some topics are more likely to co-occur and some other topics are rarely found in the same document. However, we believe such correlations are common in real-world text data and are interested in topic models that can account for them.

Blei and Lafferty (2006) propose the correlated topic model (CTM) to address this problem. Its main difference from LDA is that for each document, CTM randomly draws the topic mixture proportions from a logistic normal distribution instead of a Dirichlet. The logistic normal distribution is parameterized by a covariance matrix, in which each entry specifies the correlation between a pair of topics. While topics are no longer independent in CTM, only pairwise correlations are modeled. Additionally, the number of parameters in the covariance matrix grows as the square of the number of topics.

In this paper, we introduce the *Pachinko Allocation model* (PAM). As we will see later, LDA can be viewed as a special case of PAM. In our model, we extend the concept of topics to be not only distributions over words, but also other topics. We assume an arbitrary DAG structure, in which each leaf node is associated with a word in the vocabulary, and each interior node corresponds to a topic. There is one root in the DAG, which has no incoming links. Interior nodes can be linked to both leaves and other interior nodes. Therefore, we can capture both correlations among words like LDA, and also correlations among topic themselves.

For example, consider a document collection that discusses four topics: *cooking*, *health*, *insurance* and *drugs*. *Cooking* only co-occurs often with *health* , while *health*, *insurance* and *drugs* are often discussed together. We can build a DAG to describe this kind of correlation. The four topics form one level that is directly connected to the words. Then there are two more nodes at a higher level, where one of them is the parent of *cooking* and *health*, and the other is the parent of *health*, *insurance* and *drugs*.

In PAM, we still use Dirichlet distribution to model topic correlations. But unlike LDA, where there is only one Dirichlet to sample all the topic mixture components, we associate each interior node with a Dirichlet, parameterized by a vector with the same dimension as the number of children. To generate a document, we first sample a multinomial from each Dirichlet. Then based on these multinomials, the Pachinko machine samples a path for each word, starting from the root and ending at the leaf node.

The DAG structure in PAM is completely flexible. It can be as simple as a tree, a hierarchy, or an arbitrary DAG with edges skipping levels. The nodes can be fully or sparsely connected. We can either fix the structure beforehand or learn it from the data. It is easy to see that LDA can be viewed as a special case of PAM; the DAG corresponding to LDA is a three-level hierarchy consisting of one root at the top, a set of topics in the middle and a word vocabulary at the bottom. The root is fully connected with all the topics and each topic is fully connected with all the words. Furthermore, LDA represents topics as multinomial distributions over words, which can be seen as Dirichlet distributions with variance 0.

We present improved performance of PAM over LDA in three different experiments, including topical word coherence by human judgement, likelihood on heldout test data and accuracy of document classification. A preliminary favorable comparison with CTM is also presented.

2. The Model

In this section, we define the Pachinko allocation model (PAM) and describe its generative process, inference algorithm and parameter estimation method. To provide a better understanding of PAM, We first give a brief review of latent Dirichlet allocation.

Latent Dirichlet allocation (LDA) (Blei et al., 2003) is a generative probabilistic model based on a three-level hierarchy including:

$V = \{x_1, x_2, ..., x_v\}$: a vocabulary over words.

$S = \{t_1, t_2, ..., t_s\}$: a set of topics. Each topic is represented as a multinomial distribution over words and sampled from a given Dirichlet distribution $g(\beta)$.

r: the root, which is the parent of all topic nodes and is associated with a Dirichlet distribution $g(\alpha)$.

The model structure of LDA is shown in Figure 1(b).

To generate a document, LDA samples a multinomial distribution over topics from $g(\alpha)$, and then repeatedly samples a topic from this multinomial and a word from the topic.

Now we introduce notation for the Pachinko allocation model:

$V = \{x_1, x_2, ..., x_v\}$: a word vocabulary.

$S = \{t_1, t_2, ..., t_s\}$: a set of topics, including the root. Each of them captures some correlation among words or topics.

D: an arbitrary DAG that consists of nodes in V and S. The topic nodes occupy the interior levels and the leaves are words.

$G = \{g_1(\alpha_1), g_2(\alpha_2), ..., g_s(\alpha_s)\}$: g_i, parameterized by α_i, is a Dirichlet distribution associated with topic t_i. α_i is a vector with the same dimension as the number of variables in t_i. It specifies the correlation among its children. In the more general case, g_i is not restricted to be Dirichlet. In fact, it could be any distribution over discrete children, such as the logistic normal. But in this paper, we focus only on Dirichlet and derive the inference algorithm under this assumption.

As we can see, PAM can be viewed as an extension to LDA. It incorporates all the topics and words into an arbitrary DAG structure, while LDA is limited to a special three-level hierarchy. Two possible model structures of PAM are shown in Figure 1(c) and (d).

To generate a document d, we follow a two-step process:

1. Sample $\theta_{t_1}^{(d)}$, $\theta_{t_2}^{(d)}$, ..., $\theta_{t_s}^{(d)}$ from $g_1(\alpha_1)$, $g_2(\alpha_2)$, ..., $g_s(\alpha_s)$, where $\theta_{t_i}^{(d)}$ is a multinomial distribution of topic t_i over its children.

2. For each word w in the document,
   - Sample a topic path $\mathbf{z}_w$ of length L_w: $< z_{w1}, z_{w2}, ..., z_{wL_w} >$. z_{w1} is always the root and z_{w2} through z_{wL_w} are topic nodes in S. z_{wi} is a child of $z_{w(i-1)}$ and it is sampled according to the multinomial distribution $\theta_{z_{w(i-1)}}^{(d)}$.
   - Sample word w from $\theta_{z_{wL_w}}^{(d)}$.

Following this process, the joint probability of generating a document d, the topic assignments $\mathbf{z}^{(d)}$ and the multinomial distributions $\theta^{(d)}$ is

$$P(d, \mathbf{z}^{(d)}, \theta^{(d)}|\alpha) = \prod_{i=1}^{s} P(\theta_{t_i}^{(d)}|\alpha_i) \times \prod_{w} (\prod_{i=2}^{L_w} P(z_{wi}|\theta_{z_{w(i-1)}}^{(d)})) P(w|\theta_{z_{wL_w}}^{(d)})$$

Integrating out $\theta^{(d)}$ and summing over $\mathbf{z}^{(d)}$, we calculate the marginal probability of a document as:

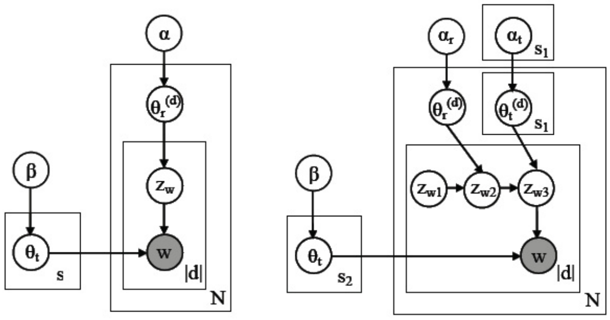
$$P(d|\alpha) = \int \prod_{i=1}^{s} P(\theta_{t_i}^{(d)}|\alpha_i) \times \prod_{w} \sum_{\mathbf{z}_w} (\prod_{i=2}^{L_w} P(z_{wi}|\theta_{z_{w(i-1)}}^{(d)})) P(w|\theta_{z_{wL_w}}^{(d)}) \mathrm{d}\theta^{(d)}$$

Finally, the probability of generating a whole corpus is the product of the probability for every document:

$$P(\mathbf{D}|\alpha) = \prod_{d} P(d|\alpha)$$

2.1. Four-Level PAM

While PAM allows arbitrary DAGs to model the topic correlations, in this paper, we focus on one special structure in our experiments. It is a four-level hierarchy consisting of one root topic, s_1 topics at the second level, s_2 topics at the third level and words at the bottom. We call the topics at the second level super-topics and the ones at the third level sub-topics. The



(a) LDA

(b) Four-Level PAM

Figure 2. Graphical Models for (a) LDA and (b) Four-Level PAM

root is connected to all super-topics, super-topics are fully connected to sub-topics and sub-topics are fully connected to words. We also make a simplification similar to LDA; i.e., the multinomial distributions for sub-topics are sampled once for the whole corpus, from a single Dirichlet distribution $g(\beta)$. The multinomials for the root and super-topics are still sampled individually for each document. As we can see, both the model structure and generative process for this special setting are similar to LDA. The major difference is that it has one additional layer of super-topics modeled with Dirichlet distributions as the root, which is the key component capturing topic correlations here. We show the model structure of this simplified PAM in Figure 1(c). We also present the corresponding graphical models for LDA and PAM in Figure 2.

2.2. Inference and Parameter Estimation

The hidden variables in PAM include the sampled multinomial distributions θ and topic assignments $\mathbf{z}$. Furthermore, we need to learn the parameters in the Dirichlet distributions $\alpha = \{\alpha_1, \alpha_2, ..., \alpha_s\}$. We could apply the Expectation-Maximization (EM) algorithm for inference, which is often used to estimate parameters for models involving hidden variables. However, EM has been shown to perform poorly for topic models due to many local maxima.

Instead, we apply Gibbs Sampling to perform inference and parameter learning. For an arbitrary DAG, we need to sample a topic path for each word given other variable assignments enumerating all possible paths and calculating their conditional probabilities. In our special four-level PAM structure, each path contains the root, a super-topic and a sub-topic. Since the root is fixed, we only need to jointly sample the super-topic and sub-topic assignments for each word, based on their conditional probability given observations and other assignments, integrating out the multinomial distributions θ. The following equation shows the joint probability of a super-topic and a sub-topic. For word w in document d, we have:

$$P(z_{w2} = t_k, z_{w3} = t_p | \mathbf{D}, \mathbf{z}_{-w}, \alpha, \beta) \propto$$
$$\frac{(n_{1k}^{(d)} + \alpha_{1k})}{(n_1^{(d)} + \sum_{k'} \alpha_{1k'})} \times \frac{(n_{kp}^{(d)} + \alpha_{kp})}{(n_k^{(d)} + \sum_{p'} \alpha_{kp'})} \times \frac{(n_{pw} + \beta_w)}{(n_p + \sum_m \beta_m)}.$$

Here we assume that the root topic is t_1. z_{w2} and z_{w3} correspond to super-topic and sub-topic assignments respectively. $\mathbf{z}_{-w}$ is the topic assignments for all other words. $n_x^{(d)}$ is the number of occurrences of topic t_x in document d; $n_{xy}^{(d)}$ is the number of times topic t_y is sampled from its parent t_x in document d; n_x is the number of occurrences of sub-topic t_x in the whole corpus and n_{xw} is the number of occurrences of word w in sub-topic t_x. Furthermore, α_{xy} is the yth component in α_x and β_w is the component for word w in β.

Note that in the Gibbs sampling equation, we assume that the Dirichlet parameters α are given. While LDA can produce reasonable results with a simple uniform Dirichlet, we have to learn these parameters for the super-topics in PAM since they capture different correlations among sub-topics. As for the root, we assume a fixed Dirichlet parameter. To learn α, we could use maximum likelihood or maximum a posteriori estimation. However, since there are no closed-form solutions for these methods and we wish to avoid iterative methods for the sake of simplicity and speed, we approximate it by moment matching. In each iteration of Gibbs sampling, we update

$$\begin{aligned}
mean_{xy} &= \frac{1}{N} \times \sum_d \frac{n_{xy}^{(d)}}{n_x^{(d)}}; \\
var_{xy} &= \frac{1}{N} \times \sum_d (\frac{n_{xy}^{(d)}}{n_x^{(d)}} - mean_{xy})^2; \\
m_{xy} &= \frac{mean_{xy} \times (1 - mean_{xy})}{var_{xy}} - 1; \\
\alpha_{xy} &\propto mean_{xy}; \\
\sum_y \alpha_{xy} &= \frac{1}{5} \times exp(\frac{\sum_y log(m_{xy})}{s_2 - 1}).
\end{aligned}$$

For each super-topic x and sub-topic y, we first calculate the sample mean $mean_{xy}$ and sample variance var_{xy}. $n_{xy}^{(d)}$ and $n_x^{(d)}$ are the same as defined above. Then we estimate α_{xy}, the yth component in α_x from sample mean and variance. N is the number of documents and s_2 is the number of sub-topics.

Smoothing is important when we estimate the Dirichlet parameters with moment matching. From the equations above, we can see that when one sub-topic y does not get sampled from super-topic x in one iteration, α_{xy} will become 0. Furthermore from the Gibbs sampling equation, we know that this sub-topic will never have the chance to be sampled again by this super-topic. We introduce a prior in the calculation of sample means so that $mean_{xy}$ will not be 0 even if $n_{xy}^{(d)}$ is 0 for every document d.

3. Experimental Results

In this section, we present example topics that PAM discovers from real-world text data and evaluate against LDA using three measures: topic clarity by human judgement, likelihood of heldout test data and accuracy of document classification.

In the experiments we discuss below, we use a fixed four-level hierarchical structure for PAM, which includes a root, a set of super-topics, a set of sub-topics and a word vocabulary. For the root, we always assume a fixed Dirichlet distribution with parameter 0.01. We can change this parameter to adjust the variance in the sampled multinomial distributions. We choose a small value so that the variance is high and each document contains only a small number of super-topics, which tends to make the super-topics more interpretable. We treat the sub-topics in the same way as LDA, i.e., assume they are sampled once for the whole corpus from a given Dirichlet with parameter 0.01. So the only parameters we need to learn are the Dirichlet parameters for the super-topics.

For both PAM and LDA, we perform Gibbs sampling to train the model. We start with 2000 burn-in iterations, and then draw a total of 10 samples in the following 1000 iterations. The total training time for the NIPS dataset (as described in Section 3.2) is approximately 20 hours on a 2.4 GHz Opteron machine with 2GB memory.

3.1. Topic Examples

The first dataset we use comes from data used by Rexa, a search engine over research papers. We randomly choose a subset of abstracts from its large collection. In this dataset, there are 4000 documents, 278438 word tokens and 25597 unique words. In Table 1 and Figure 3, we show some of the topics PAM discover using 50 super-topics and 100 sub-topics. Table 1 displays five sub-topic examples. Each column corresponds to one sub-topic and lists the top 10 words and their probabilities. Figure 3 shows a subset of super-topics in the data, and how they capture correlations among sub-topics. For each super-topic x, we rank the sub-topics $\{y\}$ based on the learned Dirichlet parameter α_{xy}. In Figure 3, each circle corresponds to one super-topic and links to a set of sub-topics as shown in the boxes, which are selected from its top 10 list. The numbers on the edges are the corresponding α values. As we can see, all the super-topics share the same sub-topic in the middle, which is a subset of stopwords in this corpus. Some super-topics also share the same content sub-topics. For example, the topics about *scheduling* and *tasks* co-occur with the topic about *agents* and also the topic about *distributed systems.* Another example is *information retrieval.* It is discussed along with both the *data mining* topic and the *web, network* topic.

3.2. Human Judgement

We provided each of five independent evaluators a set of topic pairs. Each pair contains one topic from PAM and another from LDA. Evaluators were asked to choose which one has stronger sense of semantic coherence and specificity.

These topics were generated using the NIPS abstract dataset (NIPS00-12), which includes 1647 documents, a vocabulary of 11708 words and 114142 word tokens. We use 100 topics for LDA, and 50 super-topics and 100 sub-topics for PAM. The topic pairs are created based on similarity. For each sub-topic in PAM, we find its most similar topic in LDA and present them as a pair. We also find the most similar sub-topic in PAM for each LDA topic. Similarity is measured by the KL-divergence between topic distributions over words. After removing redundant pairs and dissimilar pairs that share less than 5 out of their top 20 words,, we provide the evaluators with a total of 25 pairs. We present four example topic pairs in Table 2. There are 5 PAM topics that every evaluator agrees to be the better ones in their pairs, while LDA has none. And out of 25 pairs, 19 topics from PAM are chosen by the majority ($\geq$ 3 votes). We show the full evaluation results in Table 3.

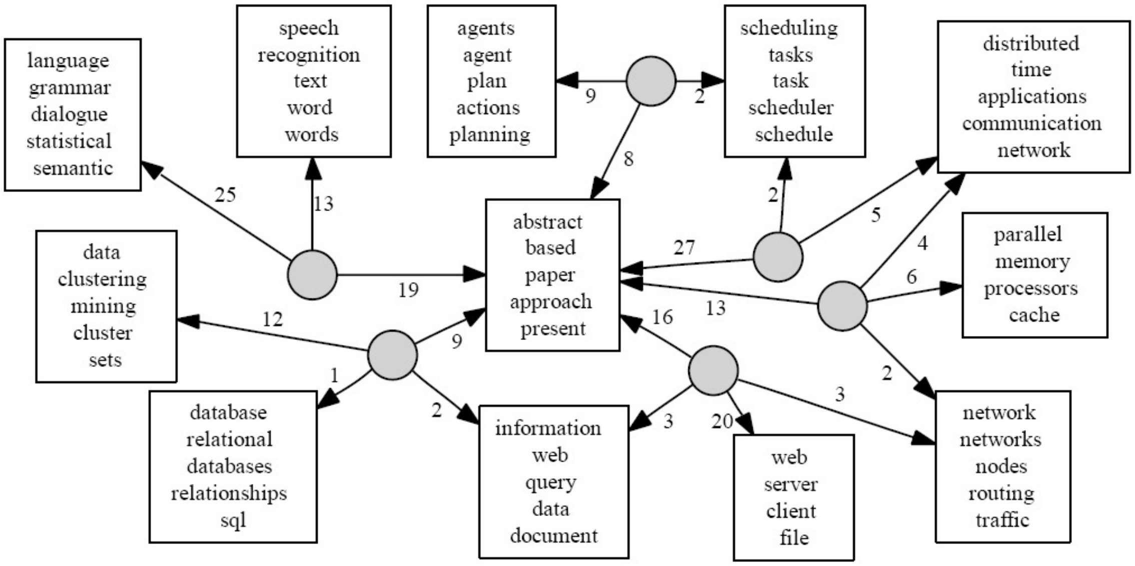
3.3. Likelihood Comparison

Besides human evaluation of topics, we also provide quantitative measurements to compare PAM with LDA. In this experiment, we use the same NIPS dataset and split it into two subsets with 75% and 25% of the data respectively. Then we learn the models from the larger set and calculate likelihood for the smaller set. We still use 50 super-topics for PAM, but the number of sub-topics varies from 20 to 180.

In order to calculate the heldout data likelihood accurately, we need to integrate out the sampled multi-

speech	0.0694	agents	0.0909	market	0.0281	students	0.0619	performance	0.0684
recognition	0.0562	agent	0.0810	price	0.0218	education	0.0445	parallel	0.0582
text	0.0441	plan	0.0364	risk	0.0191	learning	0.0332	memory	0.0438
word	0.0315	actions	0.0336	find	0.0145	training	0.0309	processors	0.0211
words	0.0289	planning	0.0260	markets	0.0138	children	0.0281	cache	0.0207
system	0.0194	communication	0.0246	information	0.0126	teaching	0.0197	parallelism	0.0169
algorithm	0.0194	world	0.0198	prices	0.0123	school	0.0185	execution	0.0169
task	0.0183	decisions	0.0194	equilibrium	0.0116	student	0.0180	programs	0.0156
acoustic	0.0183	situation	0.0165	financial	0.0116	educational	0.0146	machine	0.0150
training	0.0173	decision	0.0151	evidence	0.0111	quality	0.0129	message	0.0147

Table 1. Example Topics in PAM from Rexa dataset



language grammar dialogue statistical semantic
speech recognition text word words
agents agent plan actions planning
scheduling tasks task scheduler schedule
distributed time applications communication network
data clustering mining cluster sets
abstract based paper approach present
parallel memory processors cache
database relational databases relationships sql
information web query data document
web server client file
network networks nodes routing traffic
25
13
19
9
2
8
2
27
5
4
6
13
12
9
1
2
16
3
20
3
2

Figure 3. Topic Correlation in PAM. Each circle corresponds to a super-topic each box corresponds to a sub-topic. One super-topic can connect to several sub-topics and capture their correlation. The numbers on the edges are the corresponding α values for the (super-topic, sub-topic) pair.

nomials and sum over all possible topic assignments. This has no closed-form solution. Previous work that uses Gibbs sampling for inference approximates the likelihood of a document d by taking the harmonic mean of a set of conditional probabilities $P(d|\mathbf{z}^{(d)})$, where the samples are generated using Gibbs sampling (Griffiths & Steyvers, 2004). However, this approach is a poor approximation across different models because it measures only the best topic assignments, not the full distribution over topic assignments.

In our experiment, we calculate the data likelihood by a robust method based on extensive sampling. The first step is to randomly generate 1000 documents from the trained model, based on its own generative process. Then we estimate a new simple mixture model from the synthetic data, where each topic perfectly explains one document. Finally, we calculate the probability for the new model to generate the test data. Since the mixture model assumes only one topic for each test document, we can easily calculate the exact probability by summing over all possible topics. We show the log-likelihood on the test data in Figure 4, including the means and standard errors over 10 samples. As we can see, PAM always produces higher likelihood for different numbers of sub-topics. The advantage is especially obvious for large numbers of topics. LDA performance peaks at 40 topics and decreases as the number of topics increases. On the other hand, PAM supports larger numbers of topics and has its best performance at 160 sub-topics. We also present the likelihood for different numbers of training documents in Figure 5. Since PAM has more parameters to estimate than LDA, it does not perform as well when there is limited amount of training data.

Blei and Lafferty also compared CTM with LDA using likelihood on heldout test data. While we both

PAM	LDA	PAM	LDA
control	control	motion	image
systems	systems	image	motion
robot	based	detection	images
adaptive	adaptive	images	multiple
environment	direct	scene	local
goal	con	vision	generated
state	controller	texture	noisy
controller	change	segmentation	optical
5 votes	0 vote	4 votes	1 vote
PAM	LDA	PAM	LDA
signals	signal	algorithm	algorithm
source	signals	learning	algorithms
separation	single	algorithms	gradient
eeg	time	gradient	convergence
sources	low	convergence	stochastic
blind	source	function	line
single	temporal	stochastic	descent
event	processing	weight	converge
4 votes	1 vote	1 vote	4 votes

Table 2. Example Topic Pairs in Human Judgement

	LDA	PAM
5 votes	0	5
$\geq$ 4 votes	3	8
$\geq$ 3 votes	9	16

Table 3. Human Judgement Result. For all the categories, 5 votes, $\geq$ 4 votes and $\geq$ 3 votes, PAM has more topics to be considered better than LDA.

used the NIPS dataset, the result of PAM is not directly comparable to CTM because of different likelihood estimation procedures, (their variational and our sampling-based method each used on both LDA and the new model). However, we indirectly make a comparison. As reported by Blei and Lafferty (2006), CTM obtains an 0.40% increase over the best result of LDA in log-likelihood. On the other hand, PAM increases log-likelihood over the best LDA by as much as 1.18%. This comparison is admittedly preliminary, and more parallel comparisons are forthcoming.

3.4. Document Classification

Another evaluation comparing PAM with LDA is document classification. We conduct a 5-way classification on the comp subset of the 20 newsgroup dataset. This contains 4836 documents with a vocabulary size of 35567 words. Each class of documents is divided into 75% training and 25% test data. We train a model for each class and calculate the likelihood for the test data. A test document is considered correctly classified if its corresponding model produces the highest likelihood. We present the classification accuracies for both PAM and LDA in Table 4. According to the sign test, the improvement of PAM over LDA is statistically significant with a p-value < 0.05.

class	# docs	LDA	PAM
graphics	243	83.95	86.83
os	239	81.59	84.10
pc	245	83.67	88.16
mac	239	86.61	89.54
windows.x	243	88.07	92.20
total	1209	84.70	87.34

Table 4. Document Classification Accuracies (%)

4. Related Work

Previous work in document summarization has explored the possibility of building a topic hierarchy with a probabilistic language model (Lawrie et al., 2001). They capture the dependence between a topic and its children with relative entropy and represent it in a graph of conditional probabilities. Unlike PAM, which simultaneously learns all the topic correlations at different levels, this model incrementally builds the hierarchy by identifying topic terms for individual levels using a greedy approximation to the Dominating Set Problem.

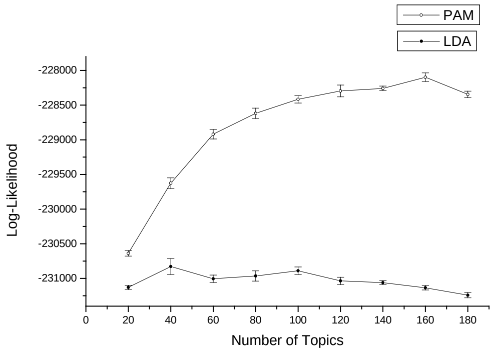


Figure 4. Likelihood Comparison with Different Numbers of Topics

Hierarchical LDA (hLDA) is a variation of LDA that assumes a hierarchical structure among topics. Topics at higher levels are more general, such as stopwords, while the more specific words are organized into topics at lower levels. To generate a document, it samples a topic path from the hierarchy and then samples every word from those topics. Thus hLDA can well explain a document that discusses a mixture of *computer science*, *artificial intelligence* and *robotics*. However, for example, the document cannot cover both *robotics* and *natural language processing* under the more general

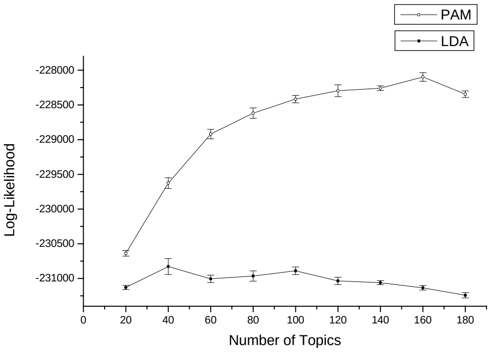


Figure 5. Likelihood Comparison with Different Amount of Training Data

topic *artificial intelligence*. This is because a document is sampled from only one topic path in the hierarchy. Compared to hLDA, PAM provides more flexibility because it samples a topic path for each word instead of each document. Note that it is possible to create a DAG structure in PAM that would capture hierarchically nested word distributions, and obtain the advantages of both models.

Another model that captures the correlations among topics is the correlated topic model introduced by Blei and Lafferty (2006). The basic idea is to replace the Dirichlet distribution in LDA with a logistic normal distribution. Under a single Dirichlet, the topic mixture components for every document are sampled almost independently from each other. Instead, a logistic normal distribution can capture the pairwise correlations between topics based on a covariance matrix. Although CTM and PAM are both trying to model topic correlations directly, PAM takes a more general approach. In fact, CTM is very similar to a special-case structure of PAM, where we create one super-topic for every pair of sub-topics. Not only is CTM limited to pairwise correlations, it must also estimate parameters for each possible pair in the covariance matrix, which grows as the square of the number of topics. In contrast, with PAM we do not need to model every pair of topics but only sparse mixtures of correlations, as determined by the number of super-topics.

In this paper, we have only described PAM with fixed DAG structures. However, we will explore the possibility of learning the structure in the future. This work is closely related to hierarchical Dirichlet process (HDP) (Teh et al., 2005). A Dirichlet prior is a distribution over parameter vectors of multinomial distributions. The Dirichlet process can be viewed as an extension where there is an infinite number of mixture components. It has been used to learn the number of topics in LDA. However, the Dirichlet prior is not restricted to parameters of multinomials only. It can also be used to sample parameter vectors for a Dirichlet distribution. Therefore, we can construct a hierarchical structure among the Dirichlets. One planned area of future work is to use Dirichlet processes to help learn the number of topics at different levels. But unlike HDP, where the documents are pre-divided into different subsets according to the hierarchical structure, we will automatically organize them in PAM.

5. Conclusion

In this paper, we have presented Pachinko allocation, a mixture model that uses a DAG structure to capture arbitrary topic correlations. Each leaf in the DAG is associated with a word in the vocabulary, and each interior node corresponds to a topic that represents a correlation among its children. A topic can be not only the parent of words, but also of other topics. The DAG structure is completely general and some topic models like LDA can be represented as special cases of PAM. Compared to other approaches that capture topic correlations such as hierarchical LDA and correlated topic model, PAM provides more expressive power to support complicated topic structures and adopts more realistic assumptions to generate documents.

References

Blei, D., Griffiths, T., Jordan, M., & Tenenbaum, J. (2004). Hierarchical topic models and the nested chinese restaurant process.

Blei, D., & Lafferty, J. (2006). Correlated topic models. In *Advances in neural information processing systems 18*.

Blei, D., Ng, A., & Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Griffiths, T., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences* (pp. 5228–5235).

Griffiths, T., Steyvers, M., Blei, D., & Tenenbaum, J. (2005). Integrating topics and syntax. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*, 537–544. Cambridge, MA: MIT Press.

Lawrie, D., Croft, W., & Rosenberg, A. (2001). Finding topic words for hierarchical summarization. *Proceedings of SIGIR'01* (pp. 349–357).

Li, W., & McCallum, A. (2005). Semi-supervised sequence modeling with syntactic topic models. *Proceedings of AAAI'05*.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., & Smyth, P. (2004). The author-topic model for authors and documents. *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*. Banff, Alberta, Canada.

Teh, Y., Jordan, M., Beal, M., & Blei, D. (2005). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*.