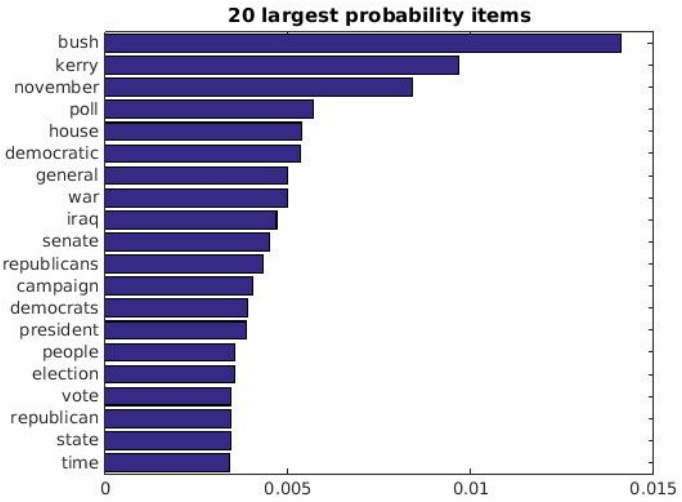
1 Training and Test Set

Set	Nb. of Documents	nb. of Words	nb. of Unique Words
A	2000	271892	6892
B	1430	195816	6870
A U B	3430	467714	6906

Table 1: Nb. of documents, words and unique words for sets A, B and A U B

2 Maximum Likelihood Multinomial Over Words

For the multinomial, each word is drawn from a discrete categorical distribution with parameters β where $p(w_i|\vec{\beta}) = \beta_i$. The MLE solution for β is $\frac{c_m}{N}$ where c_m is the total count of vocabulary word m and N is the total number of words in the set A.



(a)

Figure 1: Maximum multinomial over words for the 20 largest probability items.

3 Test Set Log Probability

$P(B) = P(w_1 w_2 ... w_n)$ for $w_n \in B$. For unigrams:

$$P(B|\vec{\beta}) = \prod_{m=1}^{N} P(w_m|\vec{\beta}) \tag{1}$$

$$log(P(B|\vec{\beta})) = \sum_{m=1}^{N} log(P(w_m|\vec{\beta}))$$

$$log(P(B|\vec{\beta})) = \sum_{m=1}^{N} log(\frac{c_m}{N})$$

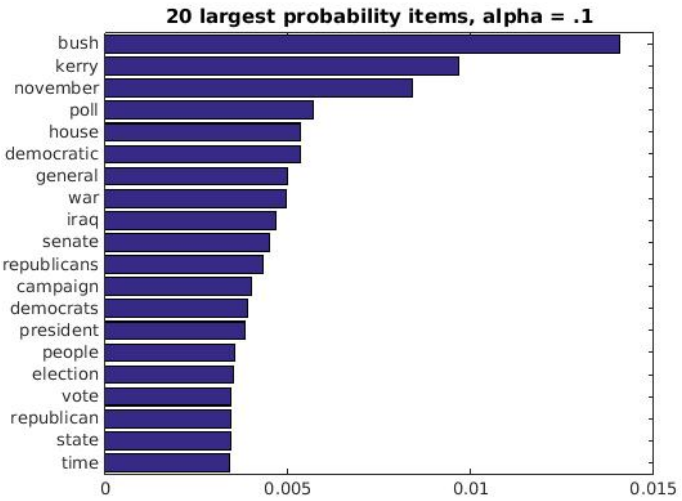
If the test set B contains a word which is not contained in the training set A, i.e. $c_m = 0$, the test set log probability will be $P(B) = -\inf$. One implication of sparse data, is that on inference, the likelihood of a sequence (given the words and the topic or class) that includes a word that is not in the training data will vanish to 0 (log probability goes to -inf) even though this sequence is valid. If the inference problem is classification, the likelihood of a class given a document will vanish to 0 for a document that includes a word that is not in any of the training documents labeled with that class (but included in the vocabulary) even though that class might be the true class.

4 Symmetric Dirichlet Prior

Each word is now drawn from a a symmetric Dirichlet prior. From page 15 of the lecture notes on Latent Dirichlet Allocation, the predictive distribution for a word j is given by:

$$p(j) = \int p(j|\pi)p(\pi|\vec{c})d\pi = E[\pi|\vec{c}] = \frac{\alpha_j + c_j}{\sum_{i=1}^{m} \alpha_i + c_i} \tag{2}$$

where $\vec{(c)} = [c_1, ..., c_k]$ are observed counts of each word j. Using a symmetric Dirichlet prior with a concentration parameter $\alpha = .1$. Increasing α in this case would increase the effect of the prior. This is the same as Laplace smoothing where α is one and where we assume we have seen every word at least once. This has the effect of avoiding 0 counts, and as a result avoiding a set log probability of -inf.



(a)

Figure 2: Maximum multinomial with a symmetric Dirichlet prior over words for the 20 largest probability items.

5 Log Probability and Perplexity

When we increase α, the prior has a bigger weight on the log probability of a document compared to the counts from the training set. The model generalizes better (up to a certain value of α after which we reach an almost uniform multinomial) and the perplexity is smaller. The perplexity is given by $PP = exp(\frac{-1}{n} log(p(w_1 w_2 ... w_n))$. The combinatorial factor is not used because we do not want the probability of every possible permutation with the words in a document (since the order in the document is fixed).

Document	α	Log Probability	Perplexity
2001	Mutlinomial ($\alpha = 0$)	-3691.5	4402
2001	0.1	-3691.2	4399
2001	1	-3688.6	4373
B	Mutlinomial ($\alpha = 0$)	$-\inf$	inf
B	0.1	-1546900	2697
B	1	-1546000	2684

Table 2: Log Probability and perplexity of document 2001 and set B with different values of concentration parameter.

6 Perplexity of a Uniform Multinomial

With a uniform multinomial we have

$$p(w_1 w_2 ... w_N) = (\frac{1}{N})^N \tag{3}$$
$$PP = exp(\frac{-1}{n} log((\frac{1}{N})^N) = N$$

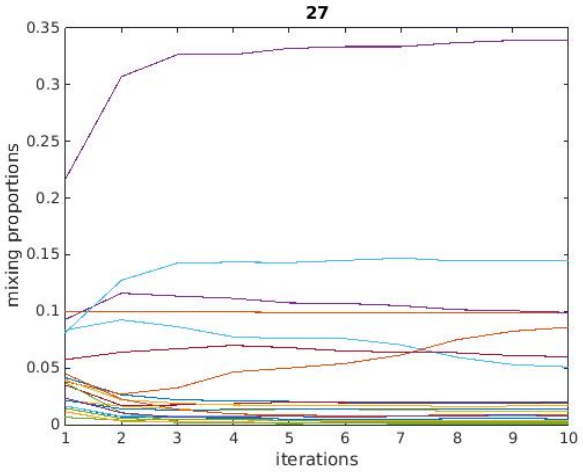
With a uniform multinomial the perplexity of all documents in B is equal to the number of unique words in B, $PP = 6879$. This is equivalent to choosing at random a word from the collection. Further, if the perplexity is higher than the number of

words in the data set, then using a uniform distribution instead would be a better approximation of the distribution. The uniform multinomial is hence a baseline where any smaller perplexity would indicate an improvement in the performance.

The perplexity in e) is smaller than the uniform multinomial distribution for the test set B. Using the likelihood of observed data in training set would give a better representation of the distribution of words in set B that simply assuming distributions of words are uniform. However, as explained in c), unseen words in the training set cause the perplexity to be infinity and assuming a 0 probability for these unseen words is not representative of their distribution (since they do exist). Adding a symmetric Dirichlet prior resolves this issue and gives lower misrepresentation of words in test documents by the training documents.

7 g. Gibbs Sampling for a Mixture of Multinomials Model

The perplexity after 10 sweeps is 2127 which is lower than the perplexity in e) and f). **Comparison**: With the uniform multinomial, the perplexity is highest because all words have equal probabilities and uncertainty is maximized. When considering counts in the training set, this undertainty decreases, which explains the lower perplexity for the multinomial with a Dirichlet prior. With the mixture of multinomials model, each document is generated by first choosing a topic z and then generating N words independently from the conditional multinomial $P(w|z)$. By using topics, the uncertainty in the model and the perplexity are decreased by considering the probabilities of words within a topic assigned to the document (the probabilities of these words within the topic better represent the document than what the probabilities of these words would have been without the different topics) .



(a)

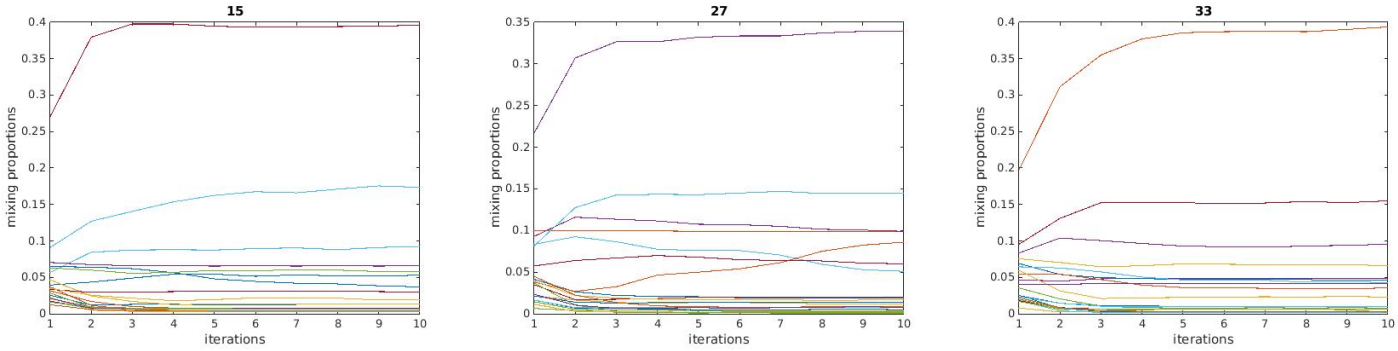
Figure 3: Mixing proportions after 10 Gibbs sweeps. (Random seed 27).

8 Convergence

Convergence of a Markov Chain is when values of the Markov chain provide samples from the posterior, i.e. the sample values have the same distribution as if they were sampled from the true posterior joint distribution. For a fixed random seed, the Gibbs sampler converges after few iterations (the perplexity is the same after few iterations).

However, the posterior probabilities of the mixture components are different for different random seeds. We notice that the perplexity of the final state changes when we change the random seed.

This shows that the Gibbs sampler converges to a local optimum but does not explore all of the posterior distribution in a limited number of iterations. The sampler is converging to a different local optima every time. In fact, the mixture of multinomials model assigns one topic to every document. The sampler explores part of the posterior distribution. If we run the sampler for an infinite amount of iterations, the sampler will explore the posterior distribution.

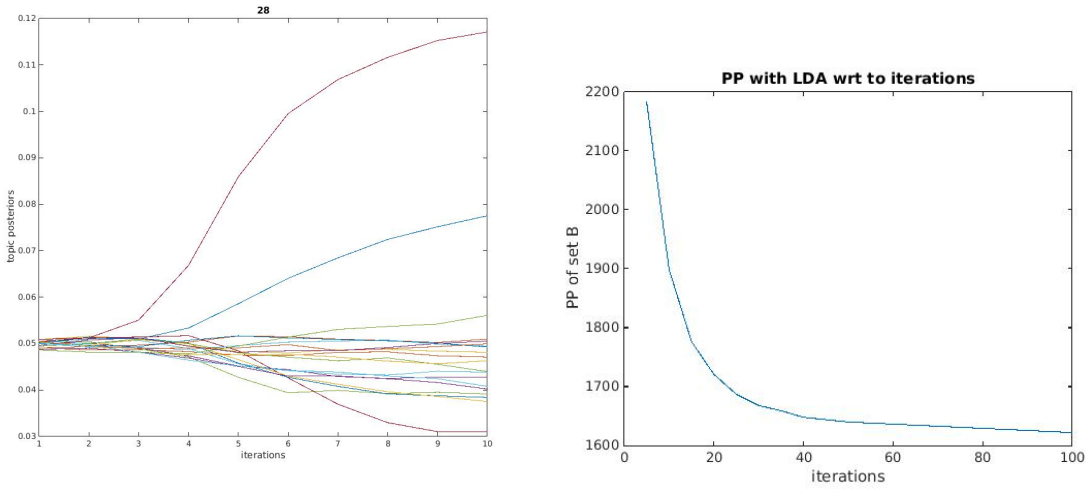


(a) (b) (c)

Figure 4: Mixing proportions for different random seed (a) 15, (b) 27, (c) 33.

9 Latent Dirichlet Allocation

Perplexity: The perplexity of documents in B for the state after 10 Gibbs sweeps is 1872 for a random seed of 33. **Comparison**: This perplexity is much lower that the one obtained with all previous models. Mixture of models is a representation of topics with the assumptions that each document is generated by 1 topic whereas LDA allows documents to exhibit multiple topics (and incorporates uncertainty related to the topic assigned to a document). All words in each document are drawn from one specific topic distribution which would work if each document is exclusively about one topic. **Number of sweeps**: The mixing proportions do not converge completely after 10 sweeps. When looking at the perplexity of doc B after each iteration, we notice that it converges after about 50 sweeps. 20 Gibbs sweeps are not adequate.



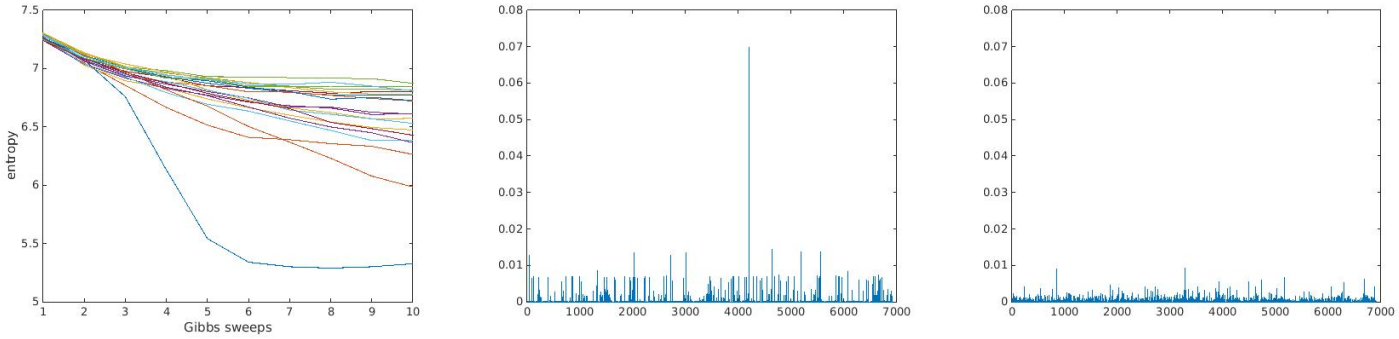
(a) (b)

Figure 5: (a) Topic posteriors with LDA after 10 sweeps. (b) Perplexity of set B after eahch Gibbs sweep.

10 Word Entropy

The (Shannon Differential) entropy is $H(P) = \sum_i p(x_i)log(p(x_i))$. The entropy of the distribution of topics captures how much a word is shared across several topics. We notice the entropy for each topic decreases after each iteration. This is because the probability of most relevant words associated with each of the topics increases after each iteration (and the probability of less relevant words decreases i.e. the model is getting more distinguishable by increasing the probability of the words associated to it).

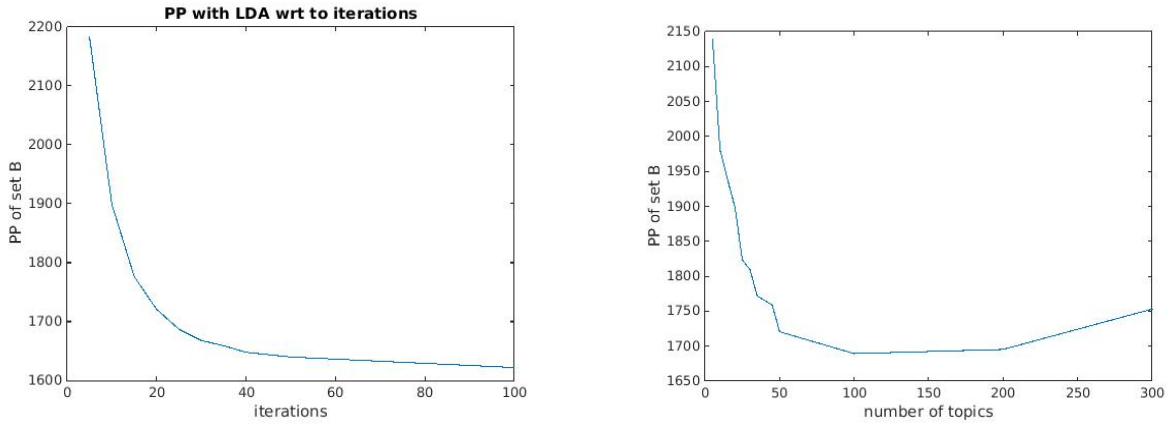
Further we notice some topics have higher entropy than the others. Looking at the top words for these topics, we notice that topics with higher entropy are more similar than the ones with lowest entropy, which aligns well with the observation that topics with lower entropy are more 'specific'. If we look at two topics (see figure 6 (b) and (c), one with low entropy and one with high entropy, we notice that distributions of words for the lower entropy topic are less uniform (we notice peaks where the model is very certain about some words compared to others) than the distributions of words for the higher entropy topic. This explains further why the uncertainty is lower for low entropy topics.



(a) (b) (c)

Figure 6: (a) Word entropy for each of the topics a s function of the number of Gibbs sweeps. (b) The word distributions for a low entropy topic and (c) high entropy topic.

11 LDA Performance with Different Gibbs Sweeps and Topics



(a) (b)

Figure 7: Perplexity of set B with respect to the number of (a) Gibbs sweeps and (b) number of topics, with LDA.

Performance with number of topics K: The perplexity of set B decreases with larger K up to K=100. The model generalizes better when there are more topics. This is because there are more parameters to fit. The perplexity of 50 topics and 100 topics LDA is similar. Hence, 50 latent topics should be enough to represent the data. However, for the training set B, the perplexity starts to increase after 100 topics. This is because a larger K is causing over-fitting and the number of parameters is too large for the training data. **Performance with number of Gibbs sweeps**: The perplexity of set B decreases with larger number of Gibbs sweeps and plateaus after about 50 sweeps. The perplexity decreases with the number of iterations because after each iteration, the uncertainty of the the distribution of topics over words decreases. As explained in k, the probabilities of words most relevant to each topic increase after each iteration.