Predicting Congressional Bill Outcomes

Barry Chen, Vivian Hu, David Wu

February 17, 2015

1 Introduction

Congressional bills are as varied as they are dense. There are personal and private bills, substantive and procedural bills, and resolutions and concurrent resolutions. Out of the many bills that are proposed in the House and Senate, only a fraction pass the congressional committees, and only a fraction of *those* are called for a house vote. Most of the bills that are put up for a vote by the entire House or Senate pass. [1]

Our project hopes to predict with high accuracy whether or not a bill will pass based on the text of the bill. We extract topics from legislative text using the Latent Dirichlet Allocation (LDA) topic model with Collapsed Gibbs Sampling. We use a collection of many bills to generate the topic model, and then use the topic distributions generated by our LDA model to classify new bills into pass and fail groups.

In this project, we say a bill has "passed" if it is explicitly passed in both the House and the Senate and is enrolled (passed to the President for a final signature). We say a bill has "failed" if it has been explicitly voted down, or it has not made it to the voting floor (by either not passing through the committee step or being ignored).

For the milestone, we have implemented our LDA topic model and trained it on a training set of pre-processed bills. Using the extracted topic distributions, we can classify bills in our test set with 76% accuracy using a logistic regression classifier. Our implementation can be found at `https://github.com/dxwu/mlCongress`.

2 Data processing

For our preliminary testing, we scraped all legislation from the 105th Congress from the database at `http://govtrack.us`. For the milestone, we stuck with bills from just one Congress for ease and speed of testing the initial implementation of our algorithm.

Using the most recent statuses of each piece of legislation, we then classified each bill as either "pass" or "fail" based on its voting status and our definitions of pass and fail in section 1. Then we extracted the latest version of its text, making sure to process out trivial stop-words and non-alphanumeric characters. From there, we built the vocabulary by lemmatizing the words and assigning each word a unique ID, resulting in a "bag-of-words" whose size is the number of word IDs.

3 Topic Models

3.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a topic model that we use to extract topics from our collection of bills. LDA basically represents documents (in our case, bills) as a mix of topics, with each topic being a distribution of word probabilities for that topic [4]. Using Collapsed Gibbs sampling, we randomly assign words to topics and then iteratively improve those assignments, as explained in the next subsection [5].

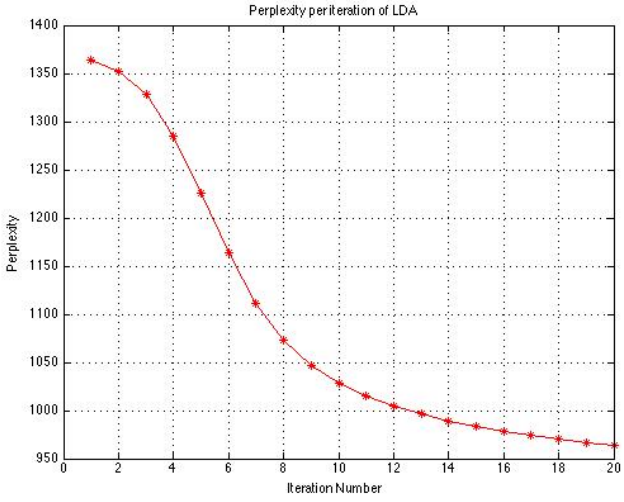
Our LDA model takes as input the number of topics we want to find, the number of iterations of Gibbs sampling, and the set of bills, from which a vocabulary is generated. We receive as output an assignment for each word in a document to a topic with some probability. This is expanded upon and visualized in section 5.1.

Based on qualitative analysis, the number of topics that we look for, K, is set to five because we found that the top words for five topics appear related and meaningful. Setting the number of topics too large slows down the algorithm and leads to topics that contain seemingly uninteresting and unrelated words.

3.2 Gibbs Sampling

Starting from a random assignment of words to topics, Collapsed Gibbs Sampling will repeatedly improve this assignment for a fixed number of iterations. A word is assigned a topic by assuming that all the other words have been assigned correctly. More precisely, a word is assigned to a topic based on a multinomial distribution given by equation 7 in [3]. The equation shows that the selected topic for a given word in a document is based on the prevalence of that topic in the document and the topics assigned to other occurrences of the word.

Perplexity is a measurement of how well the probability model assigns words to topics. The perplexity is given by equation 11 in [2]. After each iteration of Sampling, the assignment of words to topics should improve, and the perplexity should decrease:



4 Classification

4.1 Features

The features of each bill are given by the proportion of words in the bill corresponding to each topic. More specifically, a bill consisting of words $(w_1, w_2, ..., w_n)$ is represented as a feature vector, X, of length K, where:

$$X_k = P(topic = k|w_1, w_2, ..., w_n) = \frac{(\prod_{i=1}^{n} P(w_i|topic = k))P(topic = k)}{P(w_1, w_2, ..., w_n)}$$

Since the denominator is a constant for a given bill, we can compute the numerator and normalize the features to sum to 1. The terms in the numerator can be calculated in the following way:

$$P(w_i|topic = k) = \frac{\text{frequency of word} w_i}{\text{frequency of words assigned to topic k}}$$

$$P(topic = k) = \frac{\text{frequency of words assigned to topic k}}{\text{total number of words}}$$

If a test document contains a word that is not assigned a topic by LDA (i.e. the word is not contained in any of the training documents), then the word is skipped.

4.2 Logistic Regression

Once we have represented each bill as a feature vector, we can use logistic regression to classify the bill as passed or failed. The logistic regression algorithm uses gradient ascent for optimization to find a linear decision boundary that separates the two labels. Our logistic regression classifier uses a training set of 54 bills that passed and 290 bills that failed and a testing set of 37 bills that passed and 119 bills that failed.

5 Results

5.1 Topic distributions

Here we show some visualizations of the word distributions and topic distributions generated by the LDA model. Here we looked at 500 bills and 5 topics. As mentioned in section 3.1, our LDA model outputs an assignment for each word in a document to a topic.

From this, we can generate a word distribution for each topic, which represents the probability that each word appears in a certain topic. We can also generate a topic distribution for all bills, which represents the probability, for each bill, that the words in the bill can be categorized into each topic. Using the word distributions for each topic, we can also intuit names or concepts for each topic based on the top words represented in that topic.

We show in Table 1 the top eight words for each topic (excluding stop words). Figures 1 through 5 show the word distributions for our five topics. We only show the word distribution for the top 50 words overall over all topics. Figure 6 shows the overall topic distribution for 50 randomly chosen bills.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
paragraph	***land***	***health***	state	state
deleted	secretary	***plan***	act	united
year	act	***service***	***person***	***congress***
subsection	***area***	***care***	***agreement***	***house***
act	***water***	secretary	***action***	***bill***
amended	state	***child***	federal	***senate***
amount	***forest***	state	subsection	***representative***
striking	national	act	title	committee

Table 1: Top eight words for each topic. Intuitive words italicized.

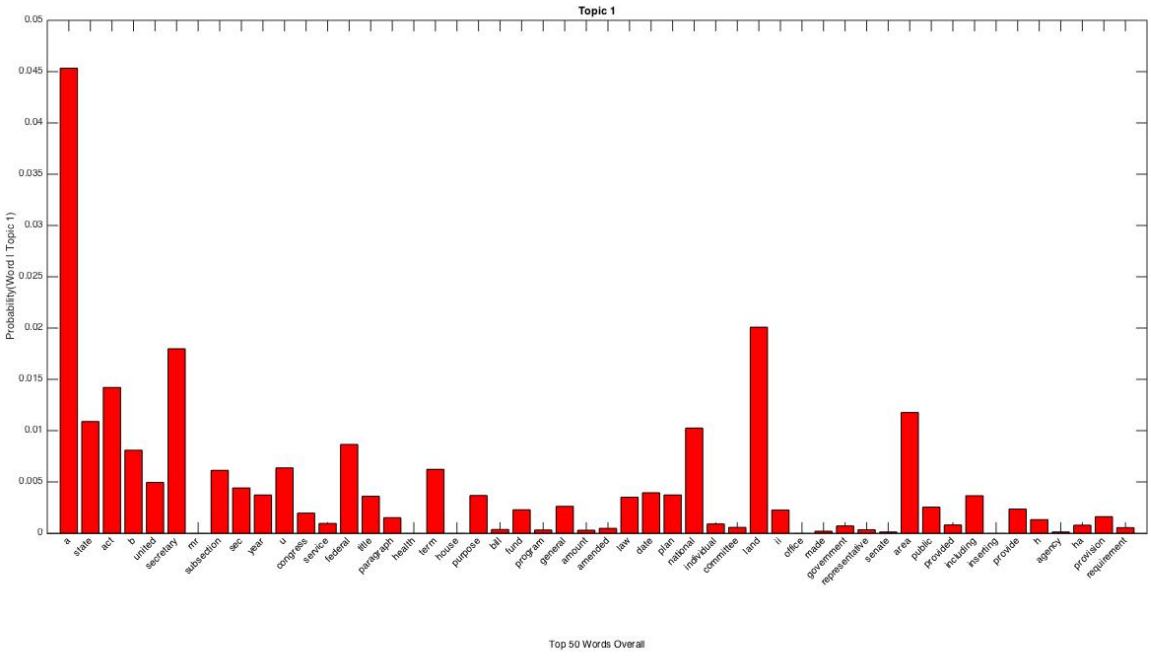


Figure 1: Word probability distribution for Topic 1

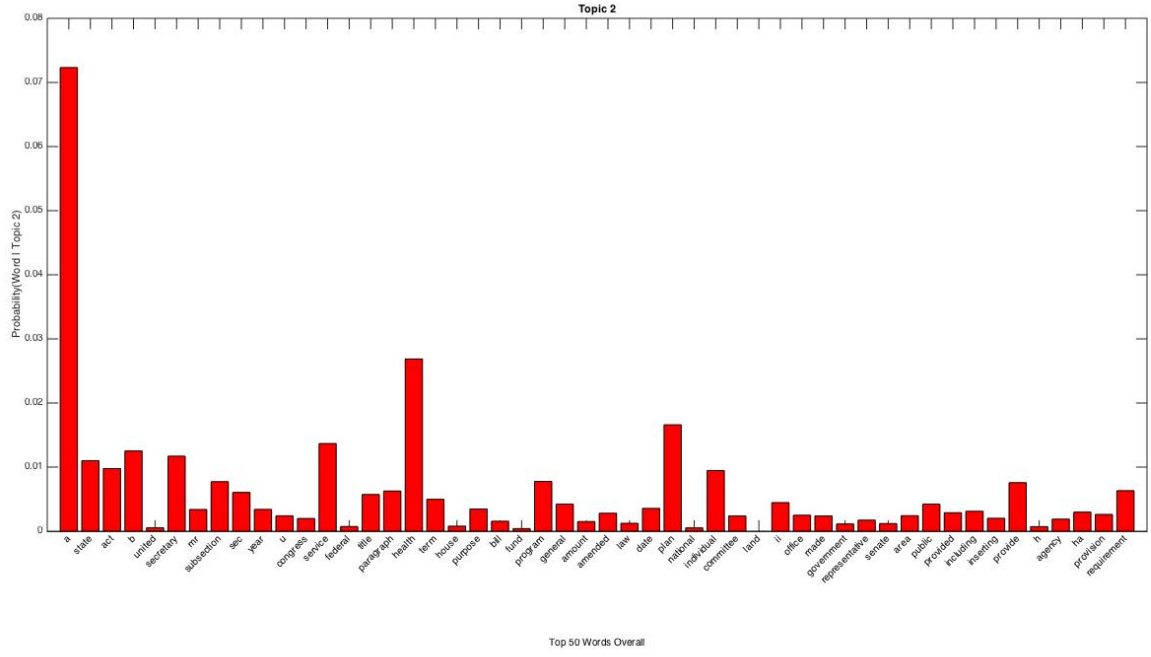


Figure 2: Word probability distribution for Topic 2

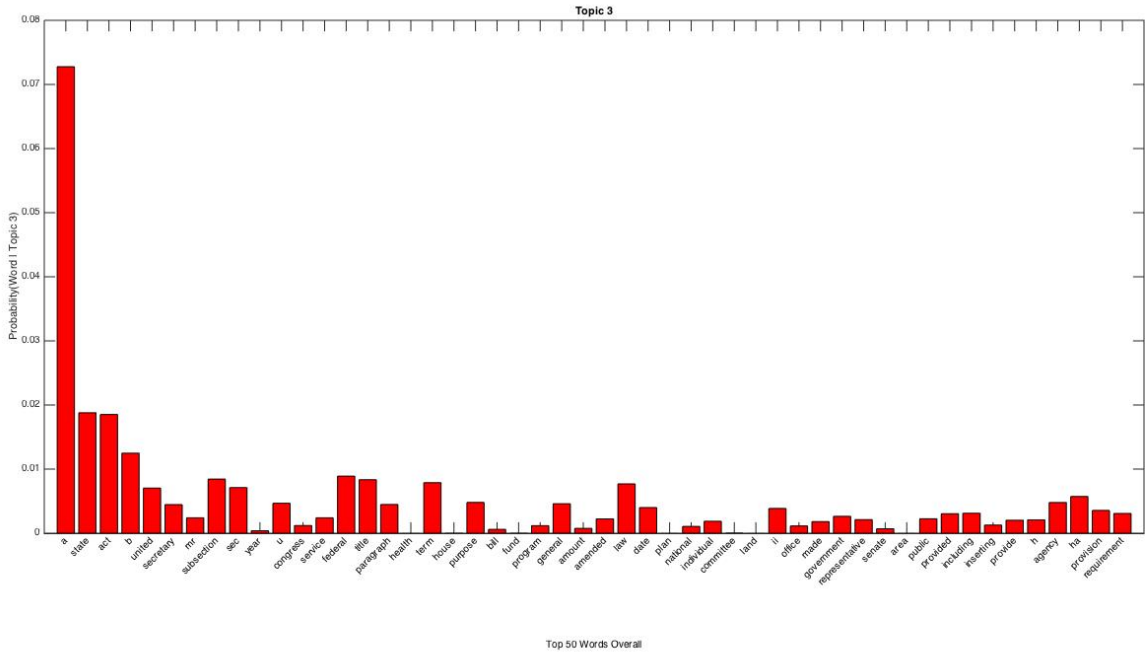


Figure 3: Word probability distribution for Topic 3

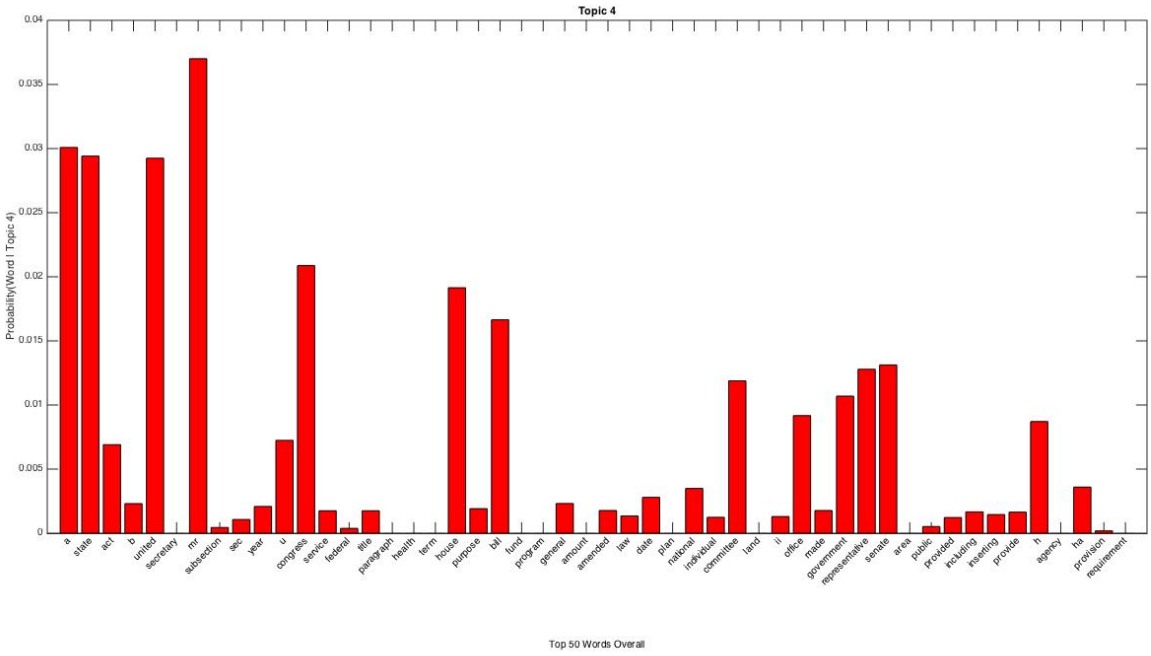


Figure 4: Word probability distribution for Topic 4

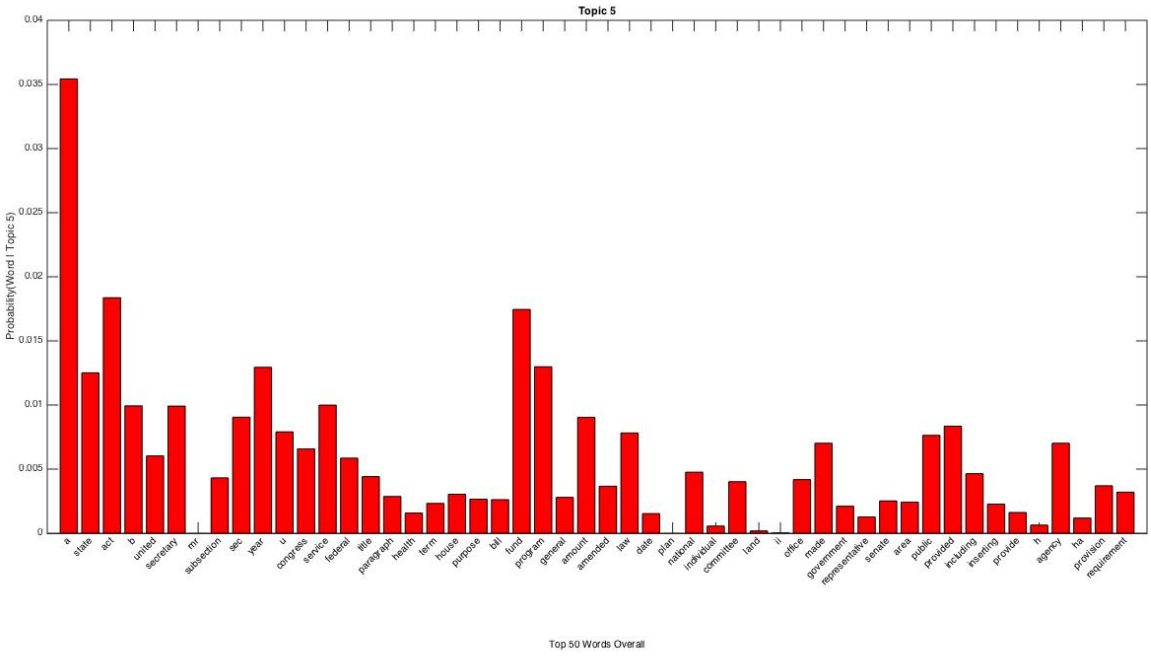


Figure 5: Word probability distribution for Topic 5

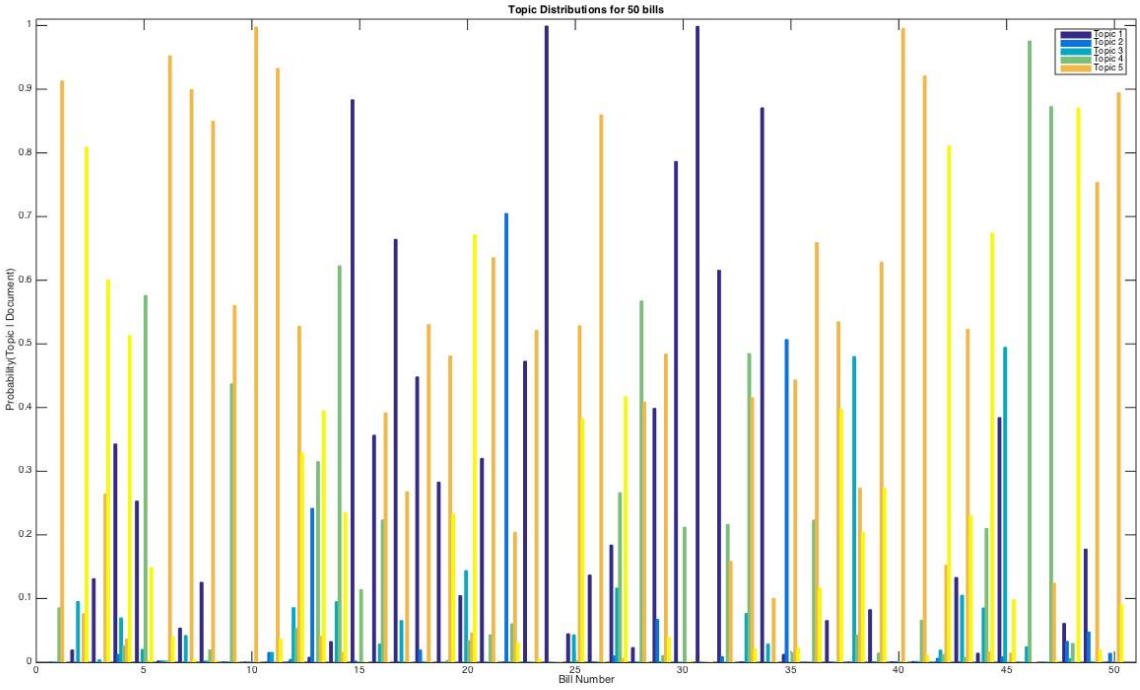
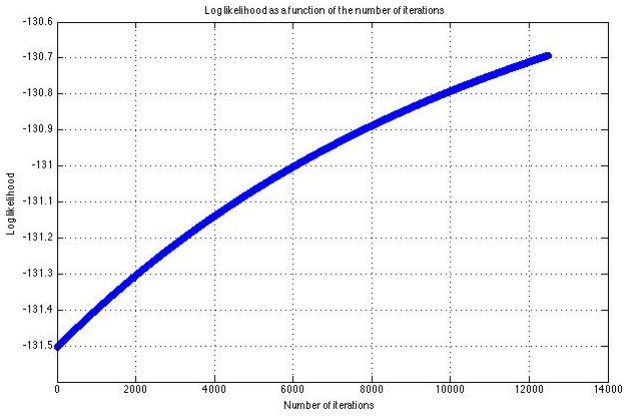


Figure 6: Topic distributions for 50 randomly chosen bills

5.2 Prediction

Our logistic regression algorithm achieves a misclassification rate of 15.7% on the training set, and a misclassification rate of 23.72% on the testing set. After each iteration of

gradient ascent, we plot the log-likelihood of the training data:



The log-likelihood increases after each iteration, supporting the accuracy of our implementation. We plan to implement random forests (discussed further in section 6) and compare the results of that classifier with our Logistic Regression baseline.

6 Going Forward

Going forward, there are several issues that need to be addressed. One such issue, introduced in section 1, is the complexity involving bill statuses themselves. Although our original assumption was that a piece of legislation would only ever either pass or fail, the range of status codes is not in fact binary.

A bill can be killed or vetoed at different stages in its life, but we've found that, more often than not, legislation ends up being "referred" and never really dealt with. We plan to take another look at more effective and accurate classifications, which would hopefully provide us with more balanced class sizes between our training and test sets, as well as better prediction results. We're also planning to reexamine our stop words to filter out additional noise for a more topic relevant "bag-of-words".

In addition, it is ultimately our goal to implement our binary classifier using random forests. We used logistic regression in our preliminary trials because it was more familiar to us, but it could also serve as a baseline comparison for whatever outcomes we might get with random forests. Right now, we are working on determining optimum thresholds for splitting with continuous features.

Each decision tree in our random forest classifier will have a decision node based on topic presence, with a leaf node representing the proportion of bills that passed with a certain set of topics. This would represent the confidence one tree has in its classification. We plan to use cross-validation to determine the optimal depth of the tress.

We're also looking to increase the size of our training set by adding legislation from multiple Congresses, which would naturally result in increased vocabulary and topic number. We're currently using Gibbs sampling as our inference method, but, in order to deal with a larger data set, it might be worthwhile, if time allows, to look into variational methods such as online LDA or extensions like hierarchical Dirichlet process (HDP) for choosing optimal topic numbers.

References

[1] `http://www.house.gov/content/learn/legislative_process/`

[2] `http://www.cis.jhu.edu/~xye/papers_and_ppts/ppts/LDA_study_notes_Xugang.pdf`

[3] `https://people.cs.umass.edu/~wallach/courses/s11/cmpsci791ss/readings/griffiths02gibbs.pdf`

[4] `http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf`

[5] https://people.cs.umass.edu/~wallach/courses/s11/cmpsci791ss/readings/griffiths02gibbs.pdf