CSE250B Project 3

Mehmet Koc A53035914
Zeynep Su Kurultay A53034989

February 28, 2014

Abstract

In this project, we employ latent Dirichlet allocation (LDA) models generated via Gibbs sampling for the purpose of learning topics on a dataset. The first dataset is Classic400, acquired from [1]. The second dataset is the Yelp dataset, obtained from [3]. Our trained model is able to learn distinct topics on the first dataset, centered around physics, biology and chemistry related themes. Also, the results of the Yelp dataset show topics that can be categorized as positive sentiment reviews, negative sentiment reviews, food items and general attributes of restaurants. The goodness-of-fit of LDA models generated from some pre-selected parameters is measured using the ground truth that is provided. A supervised model is trained on a training portion of the dataset and the accuracy is measured on the test set. On Classic400 dataset, 98.8% accuracy rate is reached in predicting true labels from the generated LDA model.

1 Introduction

For the purpose of this project, we implement topic modeling, which is an important application of text mining that derives a list of topics for each document in a corpus. We generate LDA models with different initial parameters utilizing Gibbs Sampling on the datasets, obtained from [1] and [3]. LDA is a generative model that attributes occurence of each word in the observed text to a number of topics that are present latently in this text. Before we go into detail about the specific learning algorithms, we mention the data representation we use. Since the sequence of words is irrelevant in the trained model that we use, we employ a lossy representation, namely bag of words (BOW). In this representation, the data is a sparse matrix of size MxV where M is the number of documents and V is the size of the vocabulary extracted from all of the training documents. The sparsity of this matrix comes from the fact that most entries of words for individual documents contain 0. This approach is adopted in the given dataset and the second dataset that we obtain is transformed into this representation as well.

In the next section we describe in detail the theory of LDA model and how they are generated with the help of Gibbs sampling. Next, we report the results with the word lists that accentuate each topic the most and the graphs of topic segregation for each document in 3D space.

2 Algorithm

2.1 Latent Dirichlet Allocation

Before analyzing LDA in depth, it makes sense to state the Dirichlet distribution which is a probability density function ranging over all multinomial parameter vectors. The equation for the Dirichlet distribution is given below [4]. Note that D is a normalizing constant that is going to be useful when computing the components of each iteration of Gibbs sampling, which is discussed in the next subsection. Also, Γ function is the continuous generalization of the factorial over its input.

$$p(\gamma \mid \alpha) = \frac{1}{D(\alpha)} \prod_{s=1}^{m} \gamma_s^{\alpha_s - 1},\ D(\alpha) = \frac{\prod_{s=1}^{m} \Gamma(\alpha_s)}{\Gamma(\sum_{s=1}^{m} \alpha_s)}$$

LDA is an unsupervised learning algorithm that assumes that the data observed is generated by a probabilistic process and tries to infer the parameters of this process. Specifically, given K topics and M documents, LDA assumes a distribution over the topics for each document and then for each word in the document, a topic is chosen and a word is drawn. Mathematically, the probability of the final model can be formulated with the following equation [5]. In this equation, α denotes the parameter of the Dirichlet distribution for topics corresponding to each document with length K, β denotes the parameter of the Dirichlet distribution for words corresponding to each topic with length V, which is the cardinality of the vocabulary that can be extracted from the training set. The total probability also makes use of W, which is a set of M vectors of size N (having cardinality V) of all words in all documents and which is the only observable portion of the equation. Moreover, Z is a set of M vectors of size N (having cardinality K) representing a topic for all words in all documents, Φ is the distribution of words for each topic, of size K and Θ is the joint distribution of topics in each document, of size M. Note that the cardinality of each value that Z contains is K.

$$p(W, Z, \Theta, \Phi; \alpha, \beta) = \prod_{i=1}^{K} p(\phi_i; \beta) \prod_{j=1}^{M} p(\theta_j; \alpha) \prod_{t=1}^{N} p(z_{jt} \mid \theta_j) p(w_{jt} \mid \phi_{z_{jt}})$$

As can be seen in the above equation, instead of finding the optimal α and β values via maximum likelihood; these are treated as fixed and instead the hidden variables are tried to be estimated. The hidden variables in question are Θ, the document specific multinomial and Φ, the topic distribution. The learning algorithm that we use is explained next, which is collapsed Gibbs sampling.

2.2 Training LDA with Collapsed Gibbs Sampling

In order to figure out the Φ and Θ values, Gibbs sampling first infers a z value for each word in the corpus. Remember that z is the main topic associated with each word, having cardinality K. At each word, Gibbs sampling assumes all of the other words have known z values and draws a z value for the current word according to this assumed distribution. This process is repeated for each word

as mentioned, and the resulting distribution converges to a correct distribution of z values for all words in the data. Mathematically notated, this process can be shown with the following equation. Note that the conditional probabilities should be computed for $z_i = 1$ to $z_i = K$.

$$p(z_i \mid \bar{z}', \bar{w}) = \frac{p(\bar{z}, \bar{w})}{p(\bar{z}', \bar{w})} = \frac{p(\bar{w} \mid \bar{z})p(\bar{z})}{p(w_i \mid \bar{z}')p(\bar{w}' \mid \bar{z}')p(\bar{z}')}$$

In this equation, the first factor in the denominator is ignored since it does not depend on z_i but the other 4 components are analyzed since they simplify to another equation. First, $p(\bar{z})$ and $p(\bar{z}')$ are analyzed. Note that these values depend on the Dirichlet prior α, so let's write in terms of that conditional probability, $p(\bar{z} \mid \alpha)$:

$$p(\bar{z} \mid \alpha) = \int_\theta p(\theta \mid \alpha)p(\bar{z} \mid \theta) \xrightarrow{\substack{by\ Dirichlet\ distribution \\ definition}} \frac{D(\bar{n}_m + \alpha)}{D(\alpha)}$$

In this equation, $\bar{n}_m$ is a vector of topic counts for document m. Similarly;

$$p(\bar{z}' \mid \alpha) = \frac{D(\bar{n}'_m + \alpha)}{D(\alpha)}$$

The total probability for whole corpus is just obtained with the product over all M documents. Now focusing on $p(\bar{w} \mid \bar{z})$ and $p(\bar{w}' \mid \bar{z}')$, we can see that these values depend on the Dirichlet prior β, so if we write in terms of that:

$$p(\bar{w} \mid \bar{z}, \beta) = \int_\Phi p(\Phi \mid \beta)p(\bar{w} \mid \bar{z}, \Phi)$$

Then, if we make use of the Dirichlet distribution again and grouping the words w_i according to their topic z_i, the equation becomes:

$$p(\bar{w} \mid \bar{z}, \beta) = \int_\Phi (\prod_{k=1}^{K} \frac{1}{D(\beta)} \prod_{t=1}^{V} \phi_{ky}^{\beta_t - 1})(\prod_{k=1}^{K} \prod_{t=1}^{V} \phi_{kt}^{q_{kt}}) = \prod_{k=1}^{K} \frac{1}{D(\beta)} D(\bar{q}_k + \beta)$$

Note that in this intermediary equation, q_{kt} is the total count of word t in topic k corpus-wide. A very similar equation is obtained for $p(\bar{w}' \mid \bar{z}')$. Combining these equations and using the definition of the Dirichlet distribution that was given in the beginning of the previous subsection, one obtains the final equation. Ultimately, Gibbs sampling resolves to the following equation:

$$p(z_i) = j \mid \bar{z}', \bar{w}) \propto \frac{q'_{jw_i} + \beta_{w_i}}{\sum_t q_{jt} + \beta_t} \frac{n'_{mj} + \alpha_j}{\sum_k n'_{mk} + \alpha_k}$$

which gives the intuition that the word i is likely to be assigned the topic j if this particular word occurs in association with topic j in the corpus frequently, or if j occurs frequently in the specific document containing i.

2.3 Implementation and Complexity of Gibbs sampling

Since the sequence of words inside a document do not affect LDA or Gibbs sampling, the bag-of-words (BOW) model is employed in the training phase. Note that the complexity of one epoch of Gibbs sampling amounts to $O(NK)$, since for each word in all documents, a sample is drawn by iterating through all topics. We improve this complexity by realizing the fact that in the final equation given in section 2.2 contains a component that does not depend on the current topic, j: $\sum_k n'_{mk} + \alpha_k$. By using a normalization constant Z that is the sum over all topics (and hence is computed once), we end up with the following optimized equation:

$$p(z_i = j \mid \bar{z}', \bar{w}) = \frac{1}{Z} \frac{q'_{jw_i} + \beta_{w_i}}{Q'_j + \beta} (n'_{mj} + \alpha_j)$$

3 Experiments

3.1 Parameters of Dirichlet Distribution

As explained in Section 2.2, LDA assumes prior distributions over topic proportions in each document $p(z; \alpha)$ and over word probabilities per topic $p(w|z; \beta)$. These priors are Dirichlet distributions since Dirichlet distribution is the conjugate prior for multinomial distribution which is used to model documents and vocabulary. Hence, these priors have fixed α and β parameters and the final result of LDA is dependent on them.

In Fig. 1, the Dirichlet distributions with different parameter values are depicted. In Fig. 1, the parameter is called α where:

$$\alpha = \alpha_0 * ones(K, 1) \quad where \quad \alpha_0 \in \{0.1, 1, 10\}$$

In Fig. 1, θ can be considered as the probability of $K = 3$ mixture components (topics) which are used to model a multinomial probability with 3 components. Although in the plot, there are 3 components, only 2 of them are shown (θ_1 and θ_2) since $\sum_i \theta_i = 1$ (i.e. meaning DOF=2). The vertical axis of Fig 1 shows the prior probability of each feasible θ vector.

In Fig. 1, it is seen that α_0 values close to 0 (the left plot) encodes the prior belief that each example has a dominant component. On the other hand, $\alpha_0 = 1$ (the center plot) encodes the belief that any combination of components is equally likely. From the viewpoint of Bayesian Learning, this prior is not informative, but it can be used if we have no prior belief or information about the component proportion of each example. Lastly, α_0 much greater than 1 (the right plot) encodes the belief that each example is expected to have component proportions which are close to each other. Note that although the discussion here concentrates on α, the same comments also apply to the Dirichlet distribution with β parameter.

It is possible that one may want to come up with a more general α parameter for Dirichlet distribution. In this project, since the components are hidden and also for convenience, the α_i and β_j for every component i and every word j are

α_0	β_0
50/3	0.01
5	0.5
1	1
0.5	5
0.01	50/3

Table 1: α_0 and β_0 values used in the project

treated the same. Thus, the following is the form for α and β vectors (V is the number of words in vocabulary).

$$\alpha = \alpha_0 * ones(K, 1) \quad \beta = \beta_0 * ones(V, 1)$$

For the first part of experiments, the combinations of α_0 and β_0 vectors in Table 1 are used with $K = 3$ following the recommendations in [6] and extending on them. The results of these five pairs are included in Section 4. The parameters that perform well on the initial dataset are used on the Yelp Dataset.

3.2 Goodness-of-fit of LDA Model

Following the procedure described in Sections 2.2 and 3.1, one can obtain an LDA model for each document. However, since the topics are latent, how can one measure qualitatively or quantitavely is an important question that needs to be answered since there is no guarantee that the priorly selected K, α and β are suitable for the documents at hand.

Since LDA is unsupervised model, ground truth is needed to answer the question. Assuming ground truth (category) is available for all documents, one can learn a supervised model over some training/validation examples and test it on a separate test set. Here, the examples are θ vectors that result from LDA process so that we can quantify the goodness-of-fit of initial parameters from the accuracy of the supervised classifier. Note that using a supervised classifier for categorization is superior than selecting the topic with maximum proportion for each document since supervised learning can also distinguish the examples in the same category, but with quite different topic compositions.

For supervised learning, LIBSVM toolbox [7] is used and the results of different classifiers learned from LDA models with different initial parameters are presented in Section 4. Since the number of topics selected is much less than the number of documents (i.e. $K << M$), LIBSVM with Gaussian kernel is used [8].

To save from computation time, first of all, the best LDA model is chosen by using $K = 3$ and trying α_0 and β_0 values in Table 1. After that, diferent values for K are tested with the best α_0 and β_0 values obtained at $K = 3$ where $K \in \{3, 5, 10\}$.

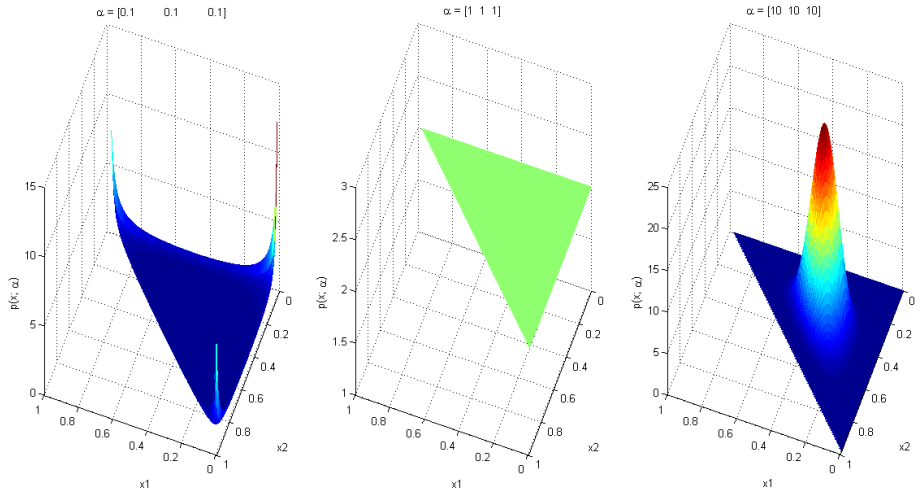

Figure 1: The Dirichlet distribution with different α parameters

3.3 Preprocessing on Data

The first dataset used comes from [1] whose actual source is [2]. In this dataset, there are $M = 400$ documents and $V = 6205$ words and the provided data is their BOW representation, the actual label for each document and the words in the vocabulary. Since the dataset does not include stop words, only preprocessing done is eliminating the words with single occurance over the corpus. After that, there are 3341 words left in the vocabulary. The topics for each word are assigned randomly from a uniform $K \times 1$ distribution and Gibbs Sampling with $S = 500$ is started where S is the number of epochs used.

The second dataset that we use is taken from The Yelp Dataset [3]. This dataset is in the form of Json so we use Gson library to parse the given data. The reviews are extracted from the Json and stored in a file. The associated labels, which are ratings for businesses ranging from 1 to 5, are stored as well. Then, since using the whole ˜350000 provided examples would be computationally too burdening, we use the linux command "shuf -n N input > output" to make sure that we pick our training, validation and test sets randomly. Next, we employ Porter stemmer's Java library in order to stem the reviews, and we employ a prepared list of stop words and punctuation to clean the data of these. We also get rid of the words that occur only once in the training set, assuming these to be spelling mistakes. Then, we construct a dictionary from the training set. Using the constructed dictionary, we transform each of the sets into vectors of counts, as befitting of the BOW model. The ratio of training, validation and test sets are chosen as 5:2:3, respectively.

4 Results

4.1 Classic400 Dataset Results

As described in Section 3.3, LDA models with parameter values in Table 1 and $K = 3$ are generated and these models are shown from Fig 2 to 6. Each point in these figures correspond to a document and their topic proportions. Using the ground truth provided, the figures are colored (note that true labels information are used only in the last supervised learning phase to numerically measure goodness-of-fit) and this gives a qualitive idea about the separation of document into distinct categories (i.e. goodness-of-fit).

Fig. 2 to 6 are shown in four subplots where θ is displayed as Gibbs Sampling iterates and slowly converges to a probability distribution. The subplots are nice to observe how Gibbs Sampling converges to the actual distribution when initial LDA parameters are selected properly as in Fig. 2 to 5. These four figures seem to be well separated whereas Fig. 6 is not well-separated and a lot of θ vectors are on top of each other at the corners in Fig. 6. This might be attributed to the fact that $\alpha_0 = 0.01$ and $\beta_0 = 50/3$ are not good initial parameters and the result is expected since $\alpha_0 = 0.01$ being close to 0 encodes our belief that each document is expected to have one dominant topic.

Now that LDA models are acquired with $K = 3$, their goodness-of-fit is evaluated quantitavely with LIBSVM as explained in Section 3.2. First, dataset is split into training and test sets in 7:3 ratio and 4-fold cross-validation (CV) is applied to select the best Gaussian kernel SVM parameters. The results are displayed in Table 2. According to this table, the best LDA parameters are $\alpha_0 = 0.5$ and $\beta_0 = 5$ with 98.8% accuracy.

Having obtained, best $\alpha_0 = 0.5$ and $\beta_0 = 5$ for $K = 3$, we vary K value and see its effect on the goodness-of-fit. SVM classification results for three K values and $\alpha_0 = 0.5$ and $\beta_0 = 5$ values are shown in Table 3. Observe that the best goodness-of-fit is observed at $K = 3$ and this might an indicator of the fact that the documents are coming from a small set of categories since intuitively it makes sense to believe that best K value and the number of categories are proportional.

Table 4 and 5 show the 20 highest-probability words from the best and worst LDA models with $K = 3$ (according to Table 2). One can see that in Table 4, the words from hidden components look like representing words from the topics; Chemistry, Biology and Physics. On the other hand, Table 5 have some mixed-topic and general words (i.e. the words in Table 4 are more semantically-related). Therefore, it is not surprising that the LDA model in Table 5 has relatively poor statistics in Table 2.

In Table 6, the 20 highest-probability words with $K = 5$ are displayed. In this table $k = 1$ seems to show words from Chemistry, $k = 4$ from Biology and $k = 5$ from Physics. On the other hand, $k = 2$ and $k = 3$ include more general and mixed words. This might be a sign that the LDA model started to overfit and this might be the reason of the relatively low performance of $K = 5$ in Table 3.

α_0	β_0	0/1 accuracy
50/3	0.01	90.0%
5	0.5	95.8%
1	1	95.8%
0.5	5	98.8%
0.01	50/3	81.7%

Table 2: Goodness-of-fit for α_0 and β_0 values with $K = 3$ in Table 1

K	0/1 accuracy
3	98.8%
5	96.7%
10	92.5%

Table 3: Goodness-of-fit for different K values with $\alpha_0 = 0.5, \beta_0 = 5$

This overfitting effect is more obvious for the words $K = 10$, but its results are omitted here for briefness.

4.2 Yelp Dataset Results

For Yelp dataset, LDA model is generated from the initial parameters $K = 4$, $\alpha_0 = 12.5$, $\beta_0 = 0.01$ and the 17 highest probability words from each topic are shown in Table 7. As can be seen from Table 7, even though the topics chosen in the Yelp dataset are not as distinct as those in the initial dataset, we can see the main patterns emerging. It can be seen that at $k = 1$, we have positive feedback words emerging like 'good', 'nice', 'well', 'best', pretti(y)', 'tast(y)', 'top', 'awes(ome)', 'flavor' and 'again'. On the other hand, for $k = 3$, we have negatively connotated words emerging like 'no', 'never', 'dont', 'didnt', 'try', 'little'. We see food items for $k = 2$ and restaurant items for $k = 4$. Note that the existence of the word 'happi' in $k = 4$ should not be alarming since the model is picking up the latent association between this sentiment and the current topic. The reviews may have a clear distinction sentiment-wise, but they do not have a clear distinction when it comes to these two topics in relation with the two remaining topics. In other words, the reviews have very close probability densities for two topics at a time.

The reason for the visible noise might be that Yelp dataset is not as clean and focused as the initial dataset. Nevertheless, as can be seen, our model is able to learn the most recurrent themes, namely the positive and negative sentiments associated with each review and topics regarding food and restaurants, which is basically the domain of our dataset. The language of reviews are much informal and even though we get rid of words which occur only once (which are mostly intentional or unintentional misspellings of words), there is still some noise in the data. One thing that also comes to mind is that we stemmed the Yelp dataset while the initial dataset is unstemmed (as can be seen by the fact that 'wing' and 'wings' exist at the same time in the results table). This might cause the results of the initial dataset to seem more focused than the Yelp dataset and this is not necessarily the case. We have kept this part brief since an extended overview of parameters and their meanings are given in the results obtained on

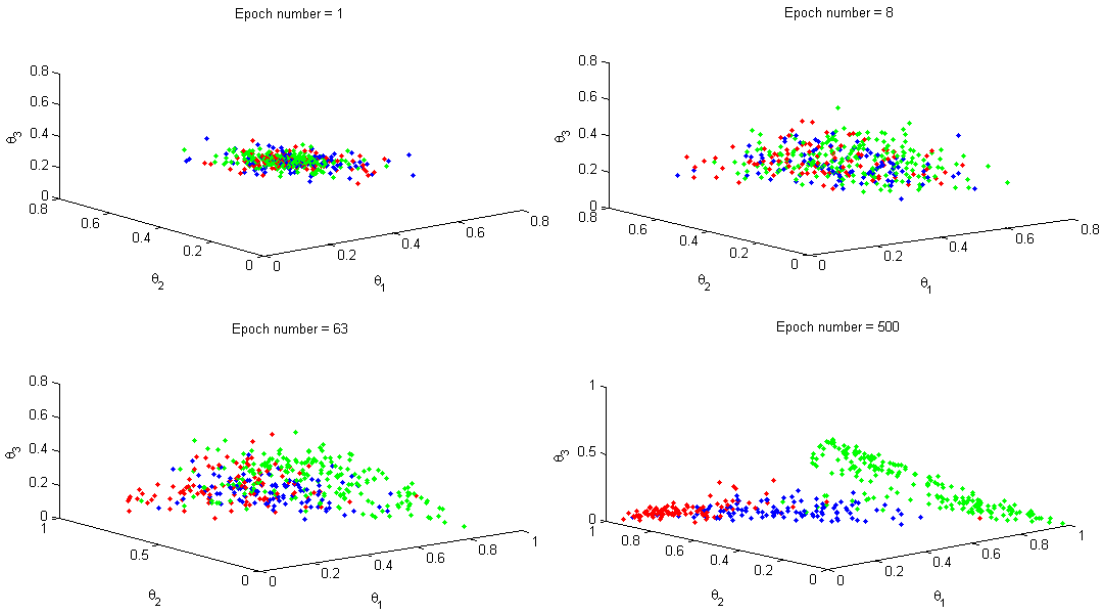

Figure 2: The evolution of θ with $K = 3, \alpha_0 = 50/3, \beta_0 = 0.01$

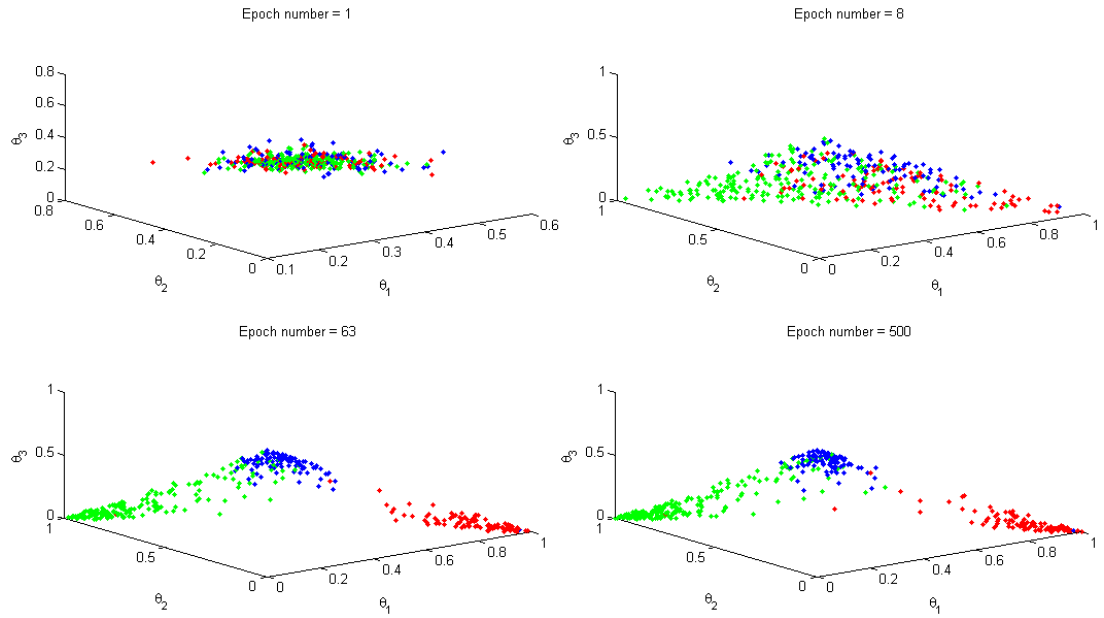

Figure 3: The evolution of θ with $K = 3, \alpha_0 = 5, \beta_0 = 0.5$

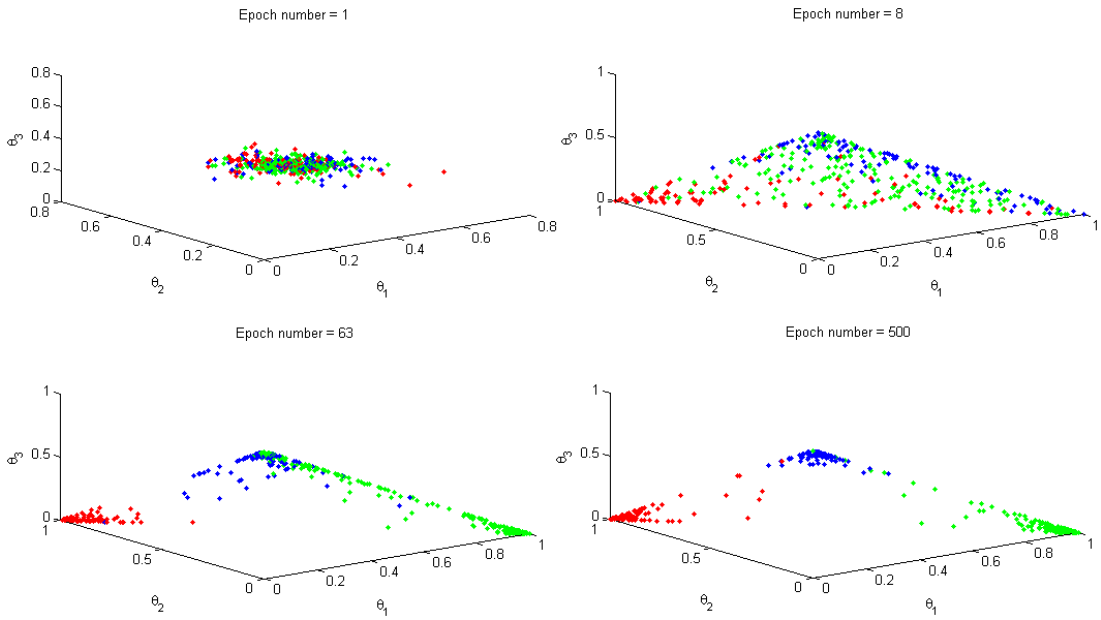

Figure 4: The evolution of θ with $K = 3, \alpha_0 = 1, \beta_0 = 1$

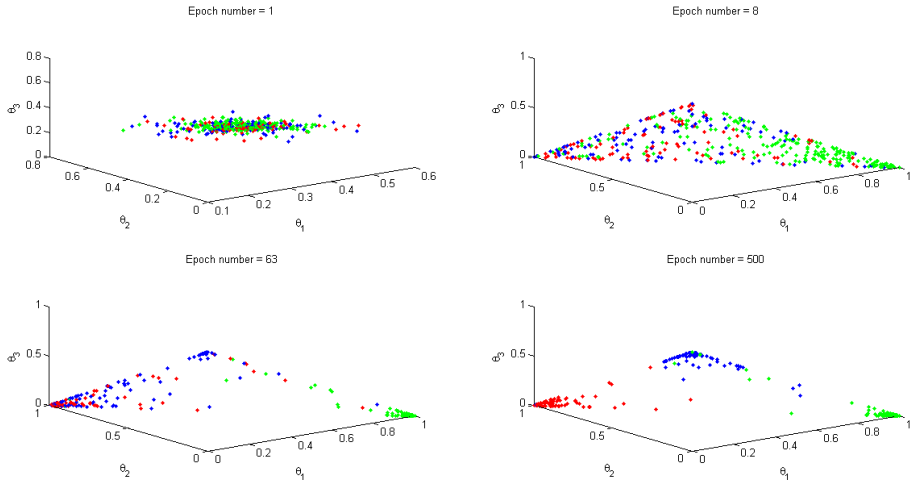

Figure 5: The evolution of θ with $K = 3, \alpha_0 = 0.5, \beta_0 = 5$

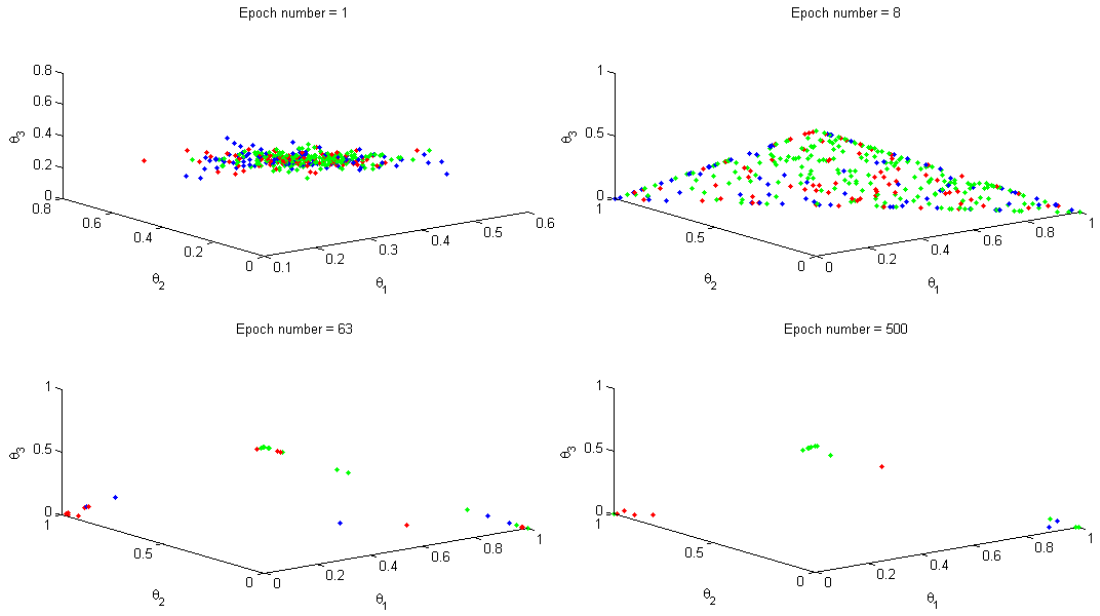

Figure 6: The evolution of θ with $K = 3, \alpha_0 = 0.01, \beta_0 = 50/3$

k = 1	k = 2	k = 3
'boundary'	'patients'	'wing'
'layer'	'cases'	'effects'
'solution'	'ventricular'	'ratio'
'plate'	'left'	'wings'
'field'	'fatty'	'mach'
'problem'	'time'	'high'
'heat'	'technique'	'lift'
'case'	'increase'	'numbers'
'transfer'	'due'	'characteristics'
'general'	'acids'	'edge'
'form'	'aortic'	'speeds'
'temperature'	'blood'	'drag'
'laminar'	'studied'	'supersonic'
'transition'	'levels'	'experimental'
'distribution'	'glucose'	'data'
'cylinder'	'volume'	'presented'
'shown'	'means'	'shock'
'wave'	'septal'	'aerodynamic'
'fluid'	'level'	'investigation'
'reynolds'	'system'	'subsonic'

Table 4: 20 highest probability words with $K = 3, \alpha_0 = 0.5, \beta_0 = 5$

k = 1	k = 2	k = 3
'journals'	'patients'	'boundary'
'science'	'ventricular'	'layer'
'nicl'	'left'	'supersonic'
'scientific'	'aortic'	'mach'
'library'	'pulmonary'	'wings'
'systems'	'regurgitation'	'shock'
'action'	'ventricle'	'wing'
'kidney'	'cases'	'surface'
'processes'	'septal'	'edge'
'system'	'visual'	'velocity'
'activity'	'language'	'speeds'
'current'	'clinical'	'numbers'
'drug'	'cardiac'	'dimensional'
'potential'	'defect'	'ratio'
'homografts'	'artery'	'subsonic'
'rejected'	'heart'	'plate'
'libraries'	'catheterization'	'effects'
'status'	'regurgitant'	'transition'
'drugs'	'agnosia'	'reynolds'
'host'	'severe'	'laminar'

Table 5: 20 highest probability words with $K = 3, \alpha_0 = 0.01, \beta_0 = 50/3$

k = 1	k = 2	k = 3	k = 4	k = 5
'boundary'	'general'	'ratio'	'patients'	'supersonic'
'layer'	'problems'	'wing'	'cases'	'effects'
'velocity'	'research'	'shock'	'ventricular'	'high'
'field'	'part'	'lift'	'left'	'surface'
'plate'	'system'	'experimental'	'developed'	'numbers'
'solution'	'subject'	'body'	'aortic'	'characteristics'
'heat'	'science'	'wings'	'studied'	'speeds'
'transfer'	'considered'	'normal'	'septal'	'data'
'problem'	'scientific'	'values'	'defect'	'dimensional'
'temperature'	'authors'	'low'	'pulmonary'	'aerodynamic'
'case'	'development'	'force'	'studies'	'investigation'
'cylinder'	'state'	'order'	'regurgitation'	'subsonic'
'laminar'	'means'	'bodies'	'volume'	'reynolds'
'distribution'	'work'	'compared'	'curves'	'due'
'fluid'	'significant'	'fatty'	'response'	'transition'
'form'	'noise'	'drag'	'occurred'	'range'
'wall'	'basic'	'leading'	'visual'	'model'
'flat'	'reference'	'jet'	'defects'	'tests'
'equations'	'structure'	'acids'	'heart'	'tunnel'
'shown'	'systems'	'increase'	'analyzed'	'stability'

Table 6: 20 highest probability words with $K = 5, \alpha_0 = 0.5, \beta_0 = 5$

k = 1	k = 2	k = 3	k = 4
'good'	'one'	'no'	'order'
'love'	'salad'	'try'	'price'
'sweet'	'drink'	'even'	'fresh'
'nice'	'they'	'look'	'staff'
'well'	'chicken'	'want'	'wait'
'best'	'did'	'know'	'help'
'make'	'meat'	'never'	'give'
'pretti'	'made'	'first'	'tabl'
've'	'take'	'went'	'bar'
'thing'	'sauc'	'ask'	'servic'
'tast'	'custom'	'came'	'menu'
'few'	'lunch'	'need'	'use'
'top'	'hamburg'	'so'	'review'
'awes'	'walk'	'littl'	'dish'
'right'	'sandwich'	'don'	'place'
'flavor'	'chees'	'got'	'happi'
'again'	'see'	'didn'	'eat'

Table 7: 17 highest probability words in Yelp with $K = 4, \alpha_0 = 50/K, \beta_0 = 0.01$

the initial dataset.

5 Conclusion

In conclusion, we train an LDA model on two datasets after they are preprocessed in order to realize the topics in a given dataset. We utilize collapsed Gibbs sampling to generate our LDA model and we experiment with different values of α_0 and β_0.

The goodness-of-fit for the parameters is obtained with the help of provided ground truth values used to train a supervised model (SVM) on the generated LDA model. For example, the model for Classic400 dataset is able to learn physics, biology and chemistry related topics when $K = 3$. In Yelp dataset, the generated LDA model can distinguish between positive and negative sentiments in reviews and also learns food and restaurant related topics for the Yelp dataset.

It is seen although LDA model does not take into account the order of the words, it is interesting to note that it is quite successful with the propagation of topics over corpus and within each document after the stochastic initialization. Yet, one must be careful in selecting the good initial LDA parameters (i.e. K, α and β) to obtain the optimal performance. Otherwise, a non-practical or overfitted model might be inevitable.

References

[1] http://cseweb.ucsd.edu/users/elkan/151/classic400.mat

[2] http://www.dataminingresearch.com/index.php/tag/dataset-2

[3] https://www.yelp.com/academicdataset

[4] http://cseweb.ucsd.edu/ elkan/250B/topicmodels.pdf

[5] http://en.wikipedia.org/wiki/LatentDirichletallocation

[6] http://jmlr.org/papers/volume3/blei03a/blei03a.pdf

[7] http://www.csie.ntu.edu.tw/ cjlin/libsvm

[8] http://www.csie.ntu.edu.tw/ cjlin/papers/guide/guide.pdf