Topic Modeling For Associated Press Articles Using Latent Dirichlet Allocation [LDA]

UNIVERSITY OF ARKANSAS AT LITTLE ROCK

Authored by: Samer Al-Khateeb

Instructor: Dr. Xiaowei Xu

Due Date: May 11, 2014

Class: Information Science Principal/Theory (IFSC 7321)

Contents

Topic Modeling For Associated Press Articles Using Latent Dirichlet Allocation [LDA]

I. Introduction

Topic modeling is one of many widely used tasks of machine learning and data mining such as classification and clustering. Latent topic models are statistical models for discovering the "topics" that occur in a collection of documents [1] [2]. They have been successful in many applications such as natural language processing, information retrieval, and filtering. The first probabilistic topic model technique to analyze documents and the words that they contain is called *Latent Semantic Analysis (LSA)*. After LSA was introduced, Hofmann proposed the *Probabilistic Latent Semantic Analysis (PLSA)* which was an evolution of the previous model. Afterwards, the *Latent Dirichelet Allocation (LDA)* was proposed and it assumed that both word-topic and topic-document distributions have a Dirichelet prior [2].

Latent Dirichlet allocation (LDA) is a generative probabilistic model for collections of discrete data such as text corpora. It is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities [3]. LDA overcomes problems previously encountered in the Probabilistic Latent Semantic Analysis model (PLSA). LDA needs training in order to converge to a generative model. After training the model it can be used in applications [2]. LDA has three parameters that need to be selected in order for it to generate documents in the form of a collection of words (w). These parameters are:

- The hyper parameters α and β which identify the Dirichelet priors.
- K which represent the number of topics in the corpus.

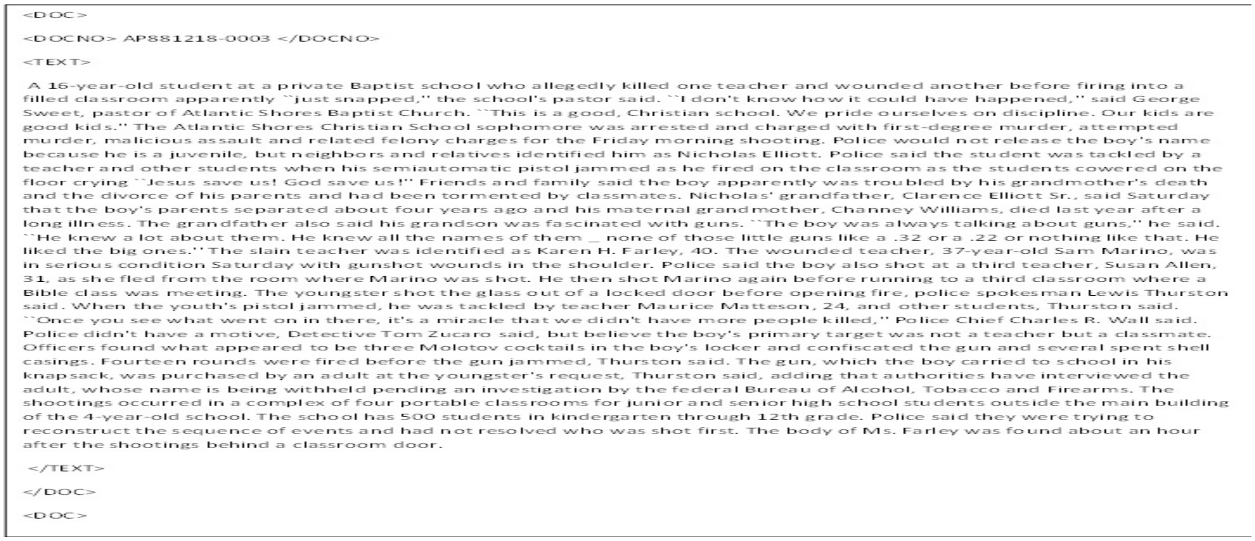
This report is structured as follows. The next section describes the methodology I used which includes: data collection, software used, and data preprocessing steps I performed. Section three describes the experiment I have made: selecting the optimal model parameters, training and testing, and the results I have obtained. Section four briefly describes related work. Section five provides a conclusion as well as a suggestion of plans for future work.

II. Methodology

This section describes the dataset and software that I used during the experiment.

A. Data Collection

The data I used in this project are the Associated Press Articles that can be found in GitHub [4][5]. The data file is in text format and contain 2250 documents. The file contains markup tags, document IDs, and the articles themselves (a sample of the data is shown in figure 1). This data set has been used previously by David M.Blei in [3].



```
<DOC>

<DOCNO> AP881218-0003 </DOCNO>

<TEXT>

 A 16-year-old student at a private Baptist school who allegedly killed one teacher and wounded another before firing into a
filled classroom apparently ``just snapped,'' the school's pastor said. ``I don't know how it could have happened,'' said George
Sweet, pastor of Atlantic Shores Baptist Church. ``This is a good, Christian school. We pride ourselves on discipline. Our kids are
good kids.'' The Atlantic Shores Christian School sophomore was arrested and charged with first-degree murder, attempted
murder, malicious assault and related felony charges for the Friday morning shooting. Police would not release the boy's name
because he is a juvenile, but neighbors and relatives identified him as Nicholas Elliott. Police said the student was tackled by a
teacher and other students when his semiautomatic pistol jammed as he fired on the classroom as the students cowered on the
floor crying ``Jesus save us! God save us!'' Friends and family said the boy apparently was troubled by his grandmother's death
and the divorce of his parents and had been tormented by classmates. Nicholas' grandfather, Clarence Elliott Sr., said Saturday
that the boy's parents separated about four years ago and his maternal grandmother, Channey Williams, died last year after a
long illness. The grandfather also said his grandson was fascinated with guns. ``The boy was always talking about guns,'' he said.
``He knew a lot about them. He knew all the names of them _ none of those little guns like a .32 or a .22 or nothing like that. He
liked the big ones.'' The slain teacher was identified as Karen H. Farley, 40. The wounded teacher, 37-year-old Sam Marino, was
in serious condition Saturday with gunshot wounds in the shoulder. Police said the boy also shot at a third teacher, Susan Allen,
31, as she fled from the room where Marino was shot. He then shot Marino again before running to a third classroom where a
Bible class was meeting. The youngster shot the glass out of a locked door before opening fire, police spokesman Lewis Thurston
said. When the youth's pistol jammed, he was tackled by teacher Maurice Matteson, 24, and other students, Thurston said.
``Once you see what went on in there, it's a miracle that we didn't have more people killed,'' Police Chief Charles R. Wall said.
Police didn't have a motive, Detective Tom Zucaro said, but believe the boy's primary target was not a teacher but a classmate.
Officers found what appeared to be three Molotov cocktails in the boy's locker and confiscated the gun and several spent shell
casings. Fourteen rounds were fired before the gun jammed, Thurston said. The gun, which the boy carried to school in his
knapsack, was purchased by an adult at the youngster's request, Thurston said, adding that authorities have interviewed the
adult, whose name is being withheld pending an investigation by the federal Bureau of Alcohol, Tobacco and Firearms. The
shootings occurred in a complex of four portable classrooms for junior and senior high school students outside the main building
of the 4-year-old school. The school has 500 students in kindergarten through 12th grade. Police said they were trying to
reconstruct the sequence of events and had not resolved who was shot first. The body of Ms. Farley was found about an hour
after the shootings behind a classroom door.

 </TEXT>

</DOC>

<DOC>
```

Figure 1 Sample of raw data

B. Software Used

For topic modeling task there are many software packages that can be used to train an LDA model such as Mallet, Matlab, R package etc. In my project I used software called "Stanford Topic Modeling Toolbox" or "TMT". This software has been developed at the Stanford NLP group by Daniel Ramage and Evan Rosen in 2009 [6]. The scripts of this software are written in Scala [7]. I did minor changes to the scripts they provided so it can do the task the way I want. The reason I used this software package after trying some other packages, is because it is very powerful and generate an excellent output. It gives you the ability to calculate the perplexity of the model with respect to α, β, and K which are the model parameters. In addition to that, it gives the user the ability to choose between two ways of approximate inference methods: Variational Bayes (VB) and Gibbs Sampling (GS). It is also very well documented software. The only negative aspect of TMT is its slow processing (require powerful processor and a lot of memory).

C. Data Pre-Processing

In this project I used Microsoft Excel to remove the tags and format the data file to contain 2250 rows and two columns (which are the document IDs and the articles themselves). Then I saved this file as .CSV so it can be used by TMT. There are many preprocessing steps that can be performed using TMT. First, I used the "SimpleEnglishTokenizer()" to remove punctuation from the ends of words and then split up the input text by whitespace characters. Second, I used the "Case Folder" which will make lower and upper case words as one word i.e. "The=tHE=ThE=thE=tHe" will become "the". Third, I applied six filters which are:

1. *WordsAndNumbersOnlyFilter:* this filter is used to ignore non-words and non-numbers.
2. *MinimumLengthFilter (3):* this filter is used to take terms which are greater than or equal to 3 characters.
3. *TermMinimumDocumentCountFilter(4):* this filter will remove terms that appear in less than 4 documents.
4. *TermDynamicStopListFilter(30):* this filter will remove the 30 most common terms.
5. *StopWordFilter ("en"):* this filter will remove all English stop words.

6. *DocumentMinimumLengthFilter (5):* this filter will take only documents with length greater than or equal to 5 terms.

I did not use stemming, although TMT provide PorterStemmer(), because stemming doesn't always add value in a topic modeling context. Stemming sometimes combines terms that would best be considered distinct, and variations of the same word will usually end up in the same topic anyway [6].

III. Experiment and Evaluation

After passing the data file through the preprocessing steps that are mentioned above, I did the data processing that are required for a topic modeling task. First, I selected the optimal model parameters by calculating the perplexity. Second, I trained the model using the optimal parameters. Finally, I tested the trained model and find out the top 20 words of each topic.

A. Selecting the Optimal Model Parameters:

For topic modeling task the optimal model parameters need to be selected. These parameters will minimize the model's perplexity on the held-out data. To do this task, I modified the provided script (example-5-lda-select.scala) so it splits a document into two subsets: one used for training models (70%), the other used for evaluating their perplexity on unseen data (30%). Perplexity is scored on the evaluation documents by first splitting each document in half. The per-document topic distribution is estimated on the first half of the words. The toolbox then computes an average of how surprised it was by the words in the second half of the document, where *surprise* is measured in the number of equiprobable word choices, on average. The value is written to the console, with lower numbers meaning a surer model [6]. I used Collapsed Gibbs Sampler with 500 iterations. After that, I selected the optimal number of topics (K). So I set a fixed value for α =0.05 and β= 0.01 to find the optimal (k) as shown in table 1 and figure 2.

Collapsed Gibbs Sampler α=0.05 β=0.01	
Number of topics	**Perplexity Training and Testing**
25	3388.148814
75	3207.115652
100	3155.092468
125	3116.040479
150	3089.820897
175	3076.42743
200	3060.202799
225	3042.451751
250	3027.010929
275	3008.070549
300	2992.074674
350	2986.988786
375	2939.712206
400	2954.645789
450	2967.345053

Table 1 Perplexity with respect to K values

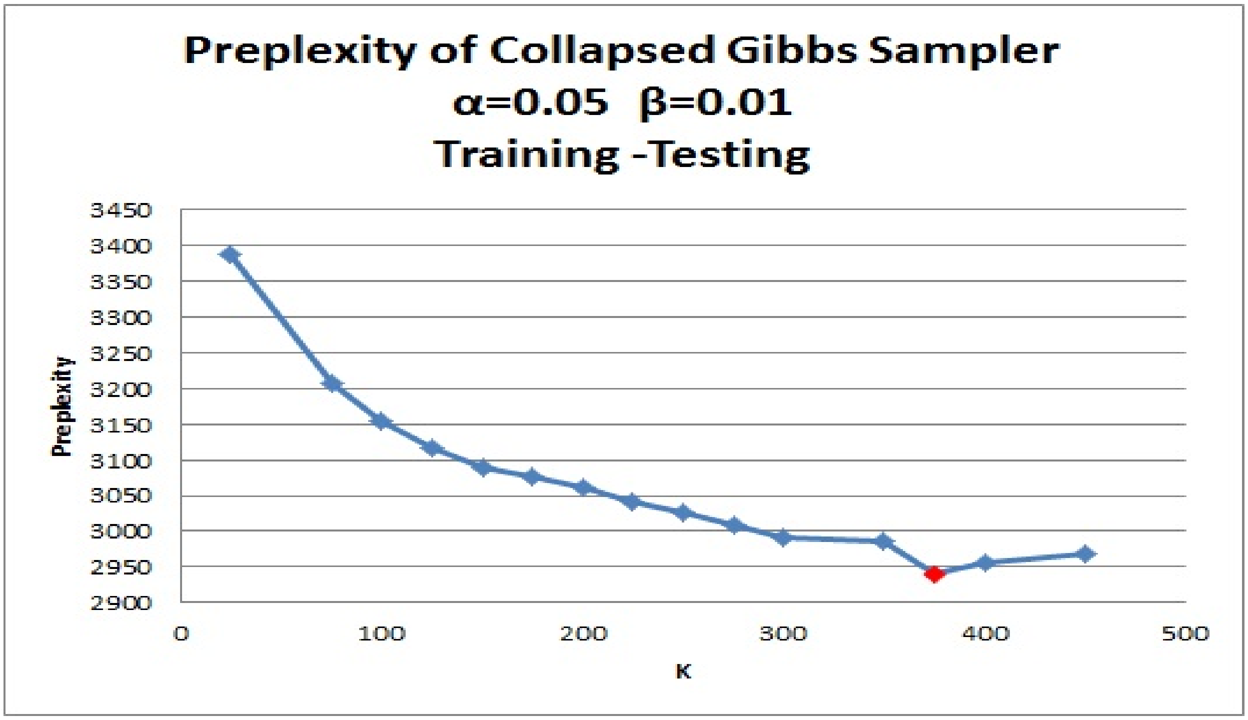


Figure 2 Perplexity with respect to K values

The minimum perplexity was obtained when k= 375, so I set this as fixed value and set β = 0.01 to find out the optimal value for α as shown in table 2 and figure 3 below.

Collapsed Gibbs Sampler k=375 β=0.01	
α	Perplexity Training and Testing
0.01	3107.94417710047
0.02	3011.87163298104
0.03	3004.17339272091
0.04	2988.93257677148
0.05	2947.06485469331
0.06	2964.5571451255
0.07	2969.99379381958
0.08	2961.39538548076
0.09	2979.88512363916
0.1	2958.06744200003
0.2	2999.20955447563
0.3	3057.32565221099
0.4	3126.01828105708
0.5	3180.55510509294
0.6	3249.67171373214
0.7	3307.38589407926
0.8	3362.773372349
0.9	3411.53131691701
1.0	3467.90782326943

Table 2 Perplexity with respect to α values

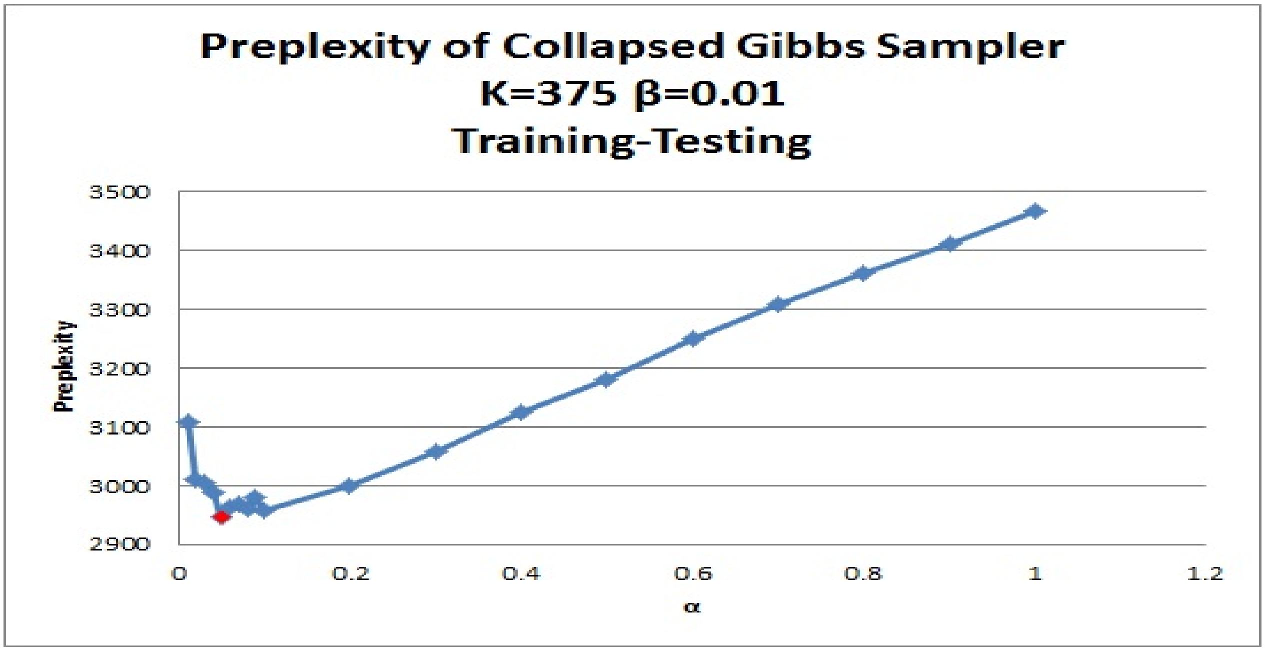


Figure 3 Perplexity with respect to α values

The minimum perplexity was obtained when α= 0.05, so I set this as fixed value and set k = 375 to find out the optimal value for β as shown in table 3 and figure 4 below.

Collapsed Gibbs Sampler k=375 α= 0.05	
β	**Perplexity Training and Testing**
0.01	2947.06485469331
0.02	2876.53574227394
0.03	2857.41009440494
0.04	2878.45721279308
0.05	2884.66531703094
0.06	2922.91330352572
0.07	2936.94286866542
0.08	2959.36743846578
0.09	2995.51550944521
0.1	2985.1512440778
0.2	3187.89763264937
0.3	3306.49393892208
0.4	3427.05646118863
0.5	3481.78289003124
0.6	3574.99640642755
0.7	3640.14403275955
0.8	3724.49025152197
0.9	3769.18900669437
1.0	3771.8287636388

Table 3 Perplexity with respect to β values

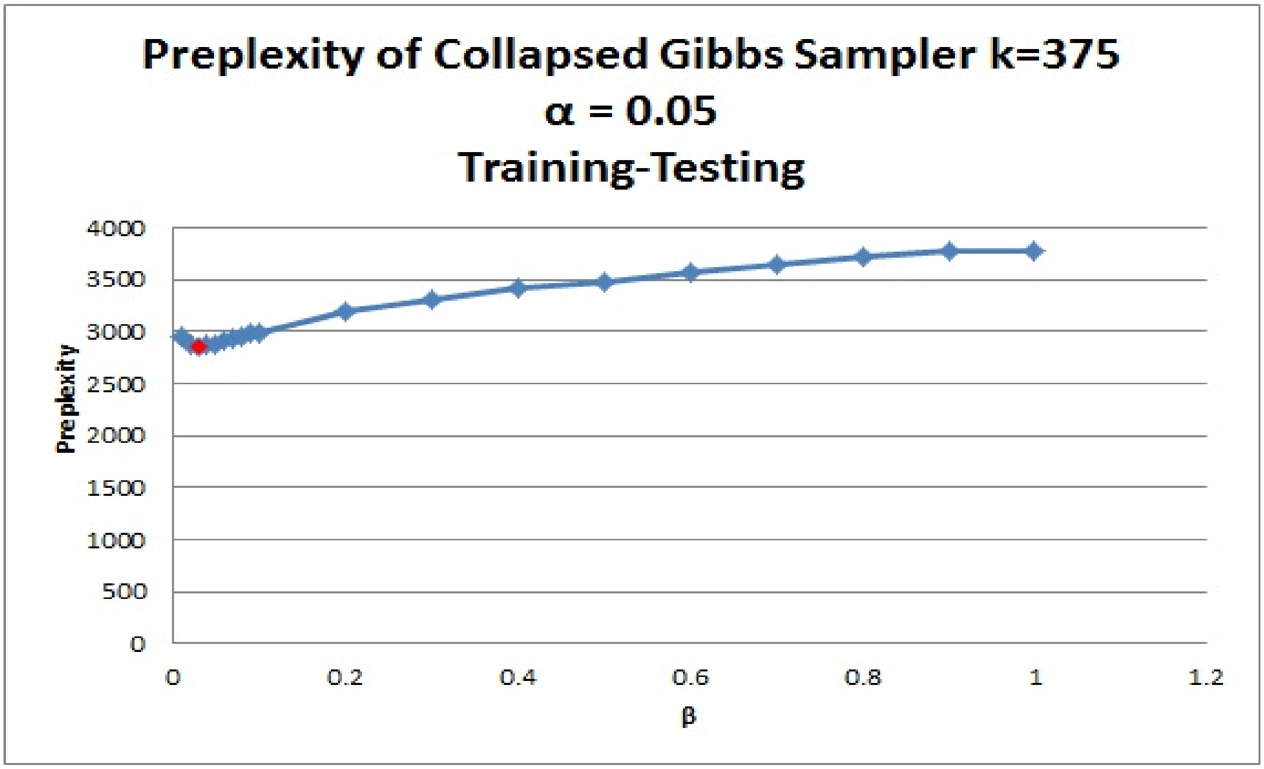


Figure 4 Perplexity with respect to β values

As a result, I found out that the minimum perplexity values were obtained when k= 375, α = 0.05, and β = 0.03 (the optimal parameters).

B. Training:

After obtaining the optimal parameters:

1. I trained the model by modifying the provided code (example-2-lda-select.scala).
2. I trained the model using the optimal model parameters (k= 375, α=0.05, β= 0.03).
3. I used 1500 iterations for training. That made the model converge as we can see in figure 5, the model's estimation of the probability of the data while training made a curve that tapers off (at 200 iterations).
4. I obtained the document-topic distributions for the trained model which will be used for testing.

C. Testing:

After I trained the model:

1. I modified the provided code (example-3-lda-select.scala) to do the inference.
2. I used the trained model to analyze the same body of text (different dataset can be used here too).
3. This process is called “inference” and I used Collapsed Gibbs Sampler because it was faster than the other method (Collapsed Variational Bayes Approximation).
4. I obtained the document-topic distributions for the testing dataset and the top 20 words of each topic.

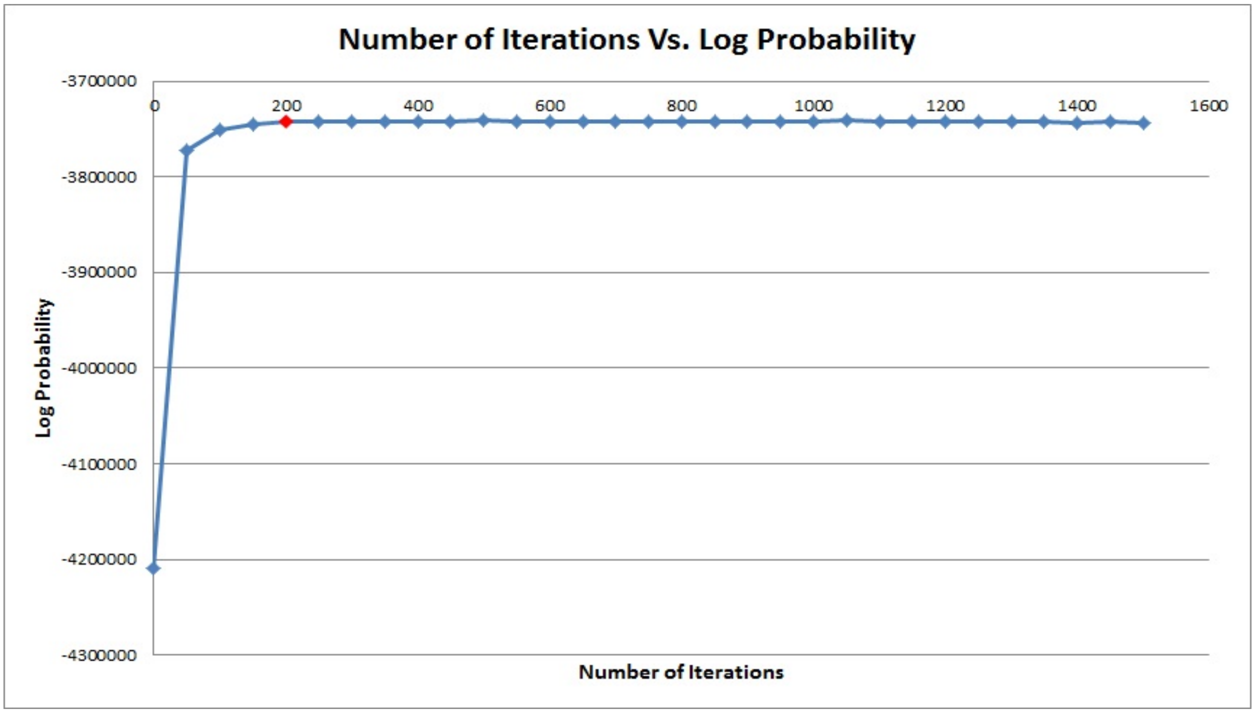


Figure 5 Number of iterations Vs. Log Probability

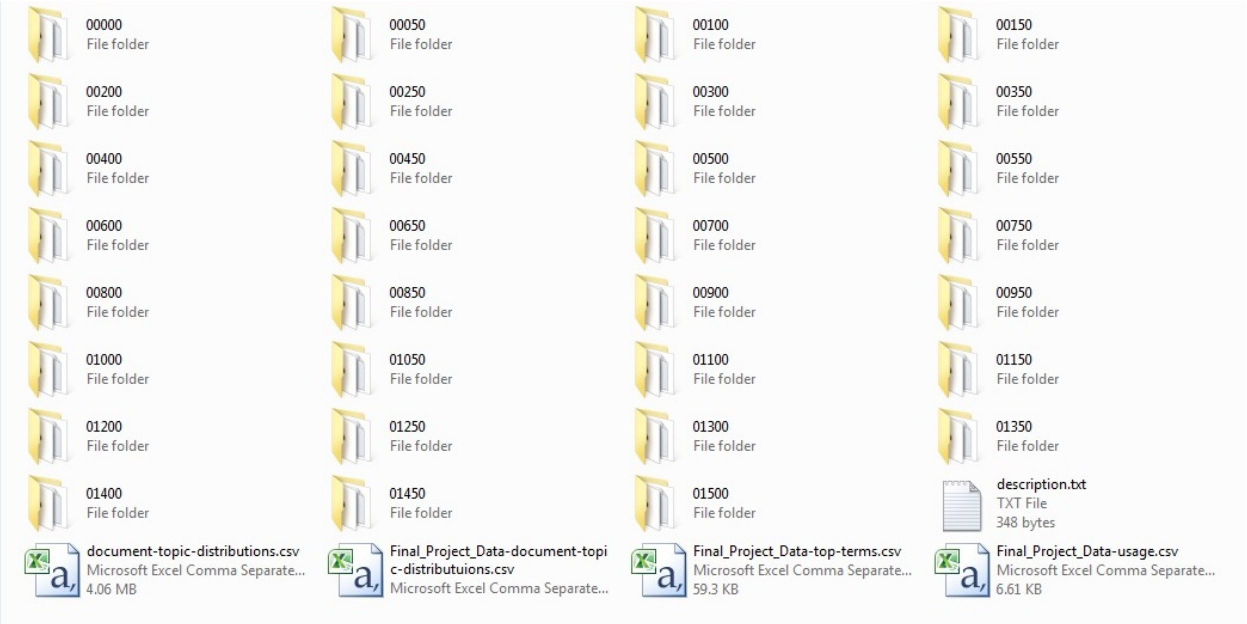
D. Results:

When I finished the three steps of topic modeling task which are:

1. obtaining the model optimal parameters,
2. training the model with the optimal parameters,
3. testing (inference) on the same dataset (or another dataset can be used too),

I checked the result folder (as shown in figure 6) which is called “lda-6ec966ca-375-b1e6781f”. This folder contains:

1. *Document-topic-distributions.csv:* it contains information about the document topic distributions of the trained model.
2. *Final_Project_Data_document_topic_distributions.csv*: it contains the information about the document topic distributions of the tested model.
3. *Final_Project_Data_top_terms.csv*: it contains the top 20 words for each topic.
4. *Description.txt*: it contains a description of the trained model.
5. 31 Folder: for each 50 iterations one folder. Each folder contains 7 files inside it:
   1. *Descrtiption.txt*: a description of the model saved in this folder
   2. *Log-probability-estimate.txt*: an estimate of the log probability of the dataset at this iteration (as shown in figure 5).
   3. *Params.tx*: the model parameters used during training.
   4. *Summary.txt*: human readable summary of the topic model, with top-20 terms per topic and how many words instances of each have occurred.
   5. *Term-index.txt:* mapping from terms in the corpus to ID numbers.
   6. *Tokenizer.txt:* tokenizer used to tokenize text for use with this model.
   7. *Topic-term-distributions.csv:* for each topic, the probability of each term in that topic.



00000 File folder
00050 File folder
00100 File folder
00150 File folder
00200 File folder
00250 File folder
00300 File folder
00350 File folder
00400 File folder
00450 File folder
00500 File folder
00550 File folder
00600 File folder
00650 File folder
00700 File folder
00750 File folder
00800 File folder
00850 File folder
00900 File folder
00950 File folder
01000 File folder
01050 File folder
01100 File folder
01150 File folder
01200 File folder
01250 File folder
01300 File folder
01350 File folder
01400 File folder
01450 File folder
01500 File folder
description.txt TXT File 348 bytes
document-topic-distributions.csv Microsoft Excel Comma Separate... 4.06 MB
Final_Project_Data-document-topic-distributuions.csv Microsoft Excel Comma Separate...
Final_Project_Data-top-terms.csv Microsoft Excel Comma Separate... 59.3 KB
Final_Project_Data-usage.csv Microsoft Excel Comma Separate... 6.61 KB

Figure 6 Output folder

After I did data analysis for the *Final_Project_Data_document_topic_distributions.csv* the summation of topics distribution for each document equals 1. I checked the two documents with ID “AP881122-0030” and “AP900509-0037” as shown in figure 7, and found out that both of them have 14 topics with different percentages. The highest percentage in both documents is topic 218.

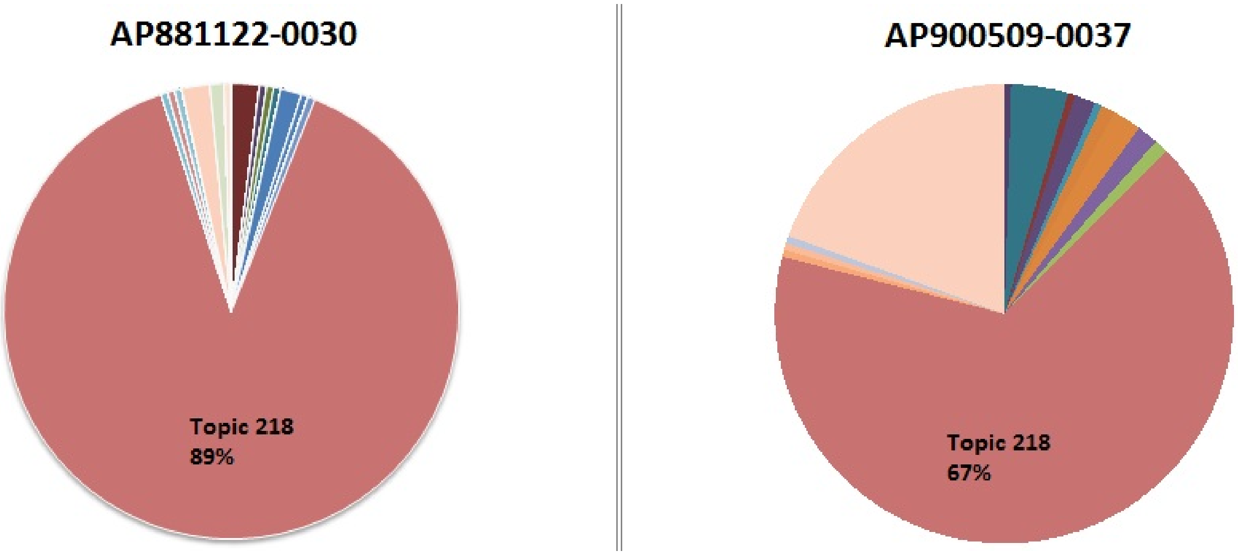
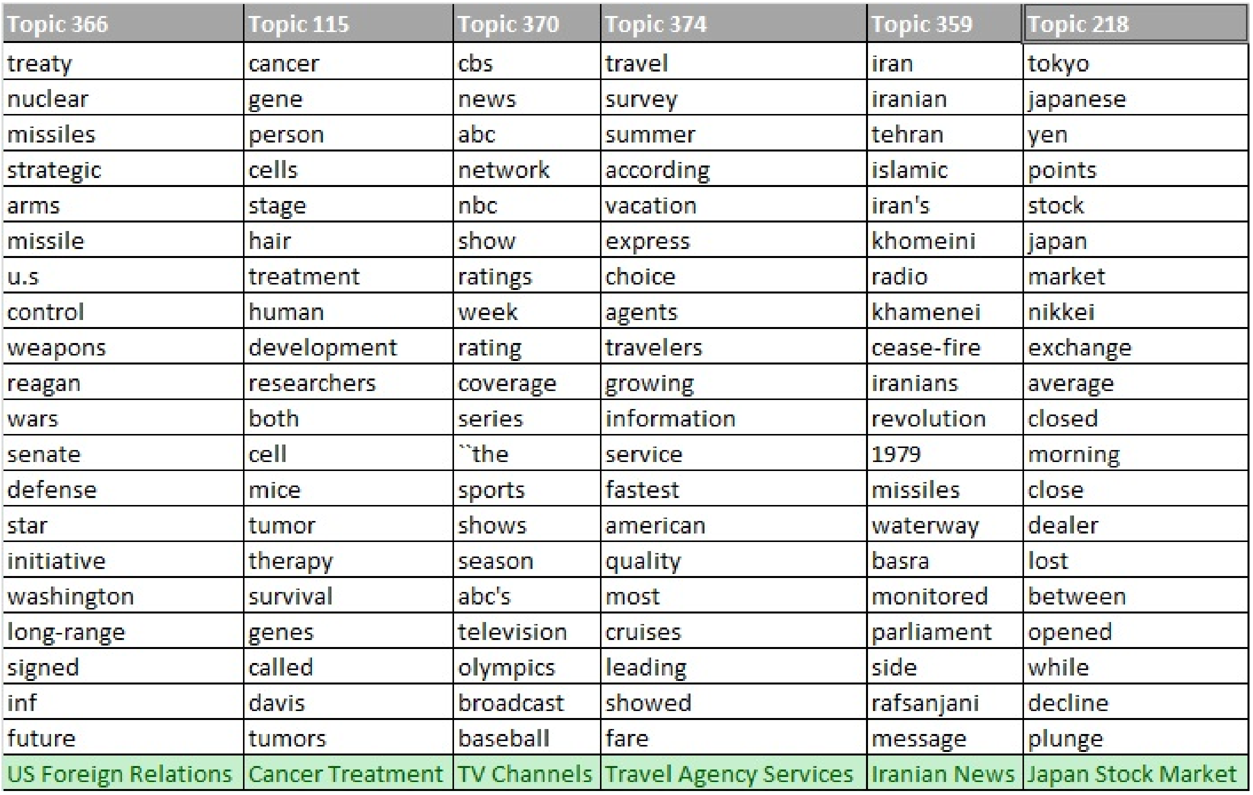


Figure 7 Document-Topic Distribution for two documents

After looking at these two documents, I find out that they are about “Tokyo Stock Exchange”. Therefore, I checked what topic 218 is about, and I find out that it is all about Japan Stock Market as shown in figure 8 below.



Topic 366	Topic 115	Topic 370	Topic 374	Topic 359	Topic 218
treaty	cancer	cbs	travel	iran	tokyo
nuclear	gene	news	survey	iranian	japanese
missiles	person	abc	summer	tehran	yen
strategic	cells	network	according	islamic	points
arms	stage	nbc	vacation	iran's	stock
missile	hair	show	express	khomeini	japan
u.s	treatment	ratings	choice	radio	market
control	human	week	agents	khamenei	nikkei
weapons	development	rating	travelers	cease-fire	exchange
reagan	researchers	coverage	growing	iranians	average
wars	both	series	information	revolution	closed
senate	cell	``the	service	1979	morning
defense	mice	sports	fastest	missiles	close
star	tumor	shows	american	waterway	dealer
initiative	therapy	season	quality	basra	lost
washington	survival	abc's	most	monitored	between
long-range	genes	television	cruises	parliament	opened
signed	called	olympics	leading	side	while
inf	davis	broadcast	showed	rafsanjani	decline
future	tumors	baseball	fare	message	plunge
US Foreign Relations	Cancer Treatment	TV Channels	Travel Agency Services	Iranian News	Japan Stock Market

Figure 8 The Top 20 words for 6 selected topics

As we can see in figure 8, the terms for each topic are very related which confirm that the parameters I used were the optimal parameters.

IV. Related Work

Latent Dirichlet Allocation (LDA) is an important hierarchical Bayesian model which attracts global interest and touches on many important applications in computer vision, text mining and many other applications [8]. LDA is also widely used in e-commerce. It is used to provide an insight of the consumer behavior and effectively support an item recommender system. The recommender system compare the user profile to some reference characteristics then try to predict what kind of items the consumer might consider in the future [2]. There are many types of LDA-based topic models such as:

- *Document-Topic Model*.
- Author-Topic Model (ATM).
- Relational-Topic Models (RTM).
- Labeled LDA (LaLDA).

Latent Dirichlet Allocation has three efficient approximate inference methods which are:

1. Variational Bayes (VB)
2. *Collapsed Gibbs Sampling (GS)*
3. Belief Propagation (BP) [8].

This project is a Document-Topic Model and as an inference method I used Collapsed Gibbs Sampling (GS).

V. Conclusion and Future Work

In this project, I applied Latent Dirichlet Allocation (LDA) on 2250 associated press articles using the Stanford Topic modeling tool. As an inference method I used the collapsed Gibbs sampler. With that, I obtained the optimal model parameters (K=375, $\alpha = 0.05$, $\beta = 0.03$) because all the topic words seems to be related, as shown in figure 8. As a result, I got the document-topic-distributions and the top 20 words of each topic.

In the future, I will consider two lines of research. First, I can try another inference method such as Variational Bayes (VB) or Belief Propagation (BP) [8] using the same dataset and compare the optimal parameters obtained from all methods. Second, I will test the results of training and testing, to see what is the most accurate inference method and which one is faster in giving the results.

References

[1] "Topic model," *Wikipedia, the free encyclopedia*. 30-Mar-2014.

[2] K. Christidis, D. Apostolou, and G. Mentzas, "Exploring Customer Preferences with Probabilistic Topics Models."

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[4] "cjrd/TMA," *GitHub*. [Online]. Available: https://github.com/cjrd/TMA. [Accessed: 09-May-2014].

[5] "Latent Dirichlet Allocation in C," 2003. [Online]. Available: http://www.cs.princeton.edu/%7Eblei/lda-c/. [Accessed: 25-Apr-2014].

[6] D. Ramage and E. Rosen, "Stanford Topic Modeling Toolbox," *The Stanford Natural Language Processing Group*, Sep-2009. [Online]. Available: http://nlp.stanford.edu/software/tmt/tmt-0.3/. [Accessed: 25-Apr-2014].

[7] "Scala (programming language)," *Wikipedia, the free encyclopedia*. 28-Apr-2014.

[8] J. Zeng, "A topic modeling toolbox using belief propagation," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 2233–2236, 2012.