Gibbs Sampling And Non-Negative Matrix Factorization For Topic Modeling and Text Classification

Rotem Bar

Afeka College of Engineering,

Department Of Intelligent Systems

Tel-Aviv, Israel

rotembaruch@gmail.com

Uria Levkovich

Afeka College of Engineering

Department of Intelligent Systems

Tel-Aviv, Israel

urialevko@gmail.com

Abstract Topic modeling for text classification is an unsupervised machine learning approach for finding abstract topics in an extensive collection of documents. It scans a set of documents, detects word and phrase patterns within them, and automatically clusters word groups and similar expressions that best characterize a set of documents. It is an essential concept of the traditional Natural Language Processing (NLP) approach because of its potential to obtain semantic relationships between words in the document clusters. Gibbs and non-negative matrix factorization (NMF) are common frameworks for detecting topics. Gibbs uses a probabilistic approach, whereas NMF uses a matrix factorization approach. This article will describe topic modeling, how Gibbs sampling and NFM models work, and implementation of the models.

Keywords—topic modeling, nlp, gibss, nmf, natural language processing

1 Introduction

Topic modeling is a branch of unsupervised natural language processing used to represent a text document with the help of several topics [1] that can explain the underlying information in a particular document [2]. A topic represents a collection of words that are grouped. A trained model may then be used to decide which of these topics occur in new documents. A first step in identifying the content of a document is determining which topics that document addresses. Very commonly used models in topic classification are latent Dirichlet allocation (LDA) [3] using Gibbs sampling [4] and non-negative matrix factorization (NMF). The aim of LDA is to find topics a document belongs to, based on the words in it. This algorithm finds the weight of connections between documents and topics and between topics and words. Using the Gibbs sampling method, the correct weights of the data will be identified. NMF is a statistical method for reducing the dimension of the input corpus using the factor analysis method.

1. TOPIC MODELS

1.1 LDA and Gibbs sampling

The task of the Bayesian inference is to compute [13] the posterior distribution over the latent topic indices **z**, the mixing proportions θ, and the topics φ, given the observed words. Collapsed Gibbs sampling [4] is an efficient inference procedure, where the mixing proportions and topics are marginalized out, while only the latent variables **z** are sampled. After the sampler has burned in, we can calculate and estimate the latter parameters given latent variables. Gibbs sampling works because the limiting distribution of θ and φ is the desired posterior distribution. However, it is only so after many iterations. For this reason, it is necessary to discard the initial observations (i.e., the burn-in).

1.2 Collapsed Gibbs sampling algorithm

Let summations of the data being $N_{wjk} = \#\{i: X_{ij} = w, z_{ij} = k\}$ and use the convention that missing indices are summed out $N_{jk} = \sum_w N_{wjk}$ and $N_{wk} = \sum_j N_{wjk}$ (N_{jk} is the number of times a word in document j has been assigned to topic k and N_{wk} is

the number of times the word w is assigned to the topic k). Given the current state of all but one variable z_{ij}, the conditional probability is then:

$$p(z_{ij} = k | \boldsymbol{z}^{\neg ij}, \boldsymbol{x}, \alpha, \beta) = \frac{1}{Z} \alpha_{jk} \beta_{wk} \text{ where}$$

$$\alpha_{kj} = N_{kj}^{\neg ij} + \alpha \qquad \beta_{wk} = \frac{N_{wk}^{\neg ij} + \beta}{N_{k}^{\neg ij} + W\beta`}$$

Z is the normalization constant

$$Z = \sum_k \alpha_{kj} \beta_{wk}$$

the superscript ¬ij indicates that the corresponding datum has been excluded in the count summations N_{wjk}.

for $i \leftarrow 1$ **to** N
do
$u \leftarrow$ draw from Uniform$[0, 1]$
for $k \leftarrow 1$ **to** K
do
$P[k] \leftarrow P[k-1] + \frac{(N_{kj}^{\neg ij} + \alpha)(N_{x_{ij}k}^{\neg ij} + \beta)}{(N_{k}^{\neg ij} + W\beta)}$
for $k \leftarrow 1$ **to** K
do
if $u < P[k]/P[K]$
then $z_{ij} = k, stop$

Figure 1. Gibbs sampling algorithm

An iteration proceeds by drawing a sample for z_{ij} for each word i in each document j:

- calculate the normalization constant Z
- sample z_{ij} according to its normalized probability.
- Given the value sampled for z_{ij}, update the counts N_{jk} , N_k, N_{wk}
- Given a sample, we can then get an estimate for $\hat{\theta}_j$ and $\hat{\varphi}_k$

$$p(z_{d,n} = k | \vec{z}_{-d,n}, \vec{w}, \alpha, \lambda) = \frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i}$$

Figure 2. Gibbs sampling mathematical expression

Gibbs mathematical expression (figure 4) is explained as follows:

- n(d,k): Number of times document d use topic k
- v(k,w): Number of times topic k uses the given word
- αk: Dirichlet parameter for a document to topic distribution
- λw: Dirichlet parameter for a topic to word distribution

There are two parts two this equation. The first part tells us how much each topic is present in a document and the second part describes how much each topic likes a word. For each word, a vector of probabilities will explain how likely this word belongs to each of the topics. Finally, a topic will be randomly picked and will be assigned to a word, and then this step will be repeated for all other words as well.

1.3 NMF

Non-negative matrix factorization is a *linear*, *non-negative* approximate data representation [17]. Let's assume that our data consists of *T* measurements of *N* non-negative scalar variables. Denoting the (*N*-dimensional) measurement vectors $\mathbf{v}^t (t = 1, \ldots, T)$ a linear approximation of the data is given by:

$$V^t \approx \sum_{i=1}^{M} w_i h_i^t = \mathrm{W}\mathrm{h}^t$$

Where **W** is an *N* ×*M* matrix containing the *basis vectors* **w***i* as its columns. The *M* basis vectors **w***i* can be thought of as the 'building blocks of the data, and the (*M*-dimensional) coefficient vector $\mathbf{h}^t$ describes how strongly each building block is present in the measurement vector $\mathbf{v}^t$. Arranging the measurement vectors **v***t* into the columns of an *N*×*T* matrix **V**, we can now write: V ≈ WH

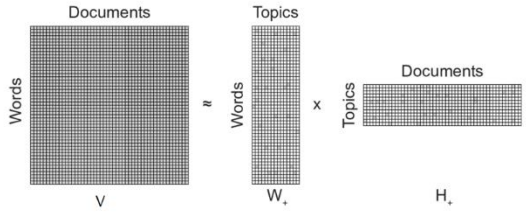

Figure 3: Non-negative matrix factorization diagram

Where each column of **H** contains the coefficient vector $\mathbf{h}^t$ corresponding to the measurement vector $\mathbf{v}^t$. Written in this form, it becomes apparent that a linear data representation is simply a factorization of the data matrix. Principal component analysis(PCA), independent component analysis, vector quantization, and non-negative matrix factorization can all be seen as matrix factorization, with different choices of the objective function. Whereas PCA does not in any way restrict the signs of the entries of **W** and **H**, NMF requires all entries of

both matrices to be non-negative. This means that the data is described by using additive components only. Matrix V is factorized into an n*t document-topic matrix and a t*m topic-term matrix, where t is the number of topics produced [19]. Equation $V \approx WH$ is achieved by finding W and H that minimize the error function $||V - HW||$ where W>0 , H>0 . A common method of measuring how good the approximation W, H is by the Frobenius norm:

$$||V - WH||^2 \approx \sum (V - WH)^2$$

(Hoyer PATRIKHOYER, 2004)There are several ways in which the W and H may be found, Lee and Seung's multiplicative update rule algorithm is:

- initialize W and H non-negative.
- Then update the values in W and H by computing the following, with n as an iteration index.

$$\mathbf{H}^{n+1}_{[i,j]} \leftarrow \mathbf{H}^{n}_{[i,j]} \frac{((\mathbf{W}^n)^T\mathbf{V})_{[i,j]}}{((\mathbf{W}^n)^T\mathbf{W}^n\mathbf{H}^n)_{[i,j]}}$$

and

$$\mathbf{W}^{n+1}_{[i,j]} \leftarrow \mathbf{W}^{n}_{[i,j]} \frac{(\mathbf{V}(\mathbf{H}^{n+1})^T)_{[i,j]}}{(\mathbf{W}^n\mathbf{H}^{n+1}(\mathbf{H}^{n+1})^T)_{[i,j]}}$$

Figure 4: H , W Matrix iterative update

Where n is the iteration index, and i,j are the indexes in the matrix.

2. Coherence Measure

Coherence can be defined as a set of statements or facts connected by meaning and supporting each other. A coherent fact set can be interpreted in a context covering all or most of the points. There are different coherence measures [18], and each is based on another calculation method. We will focus on the most popular one, the CV. CV is based on a sliding window, a one-set segmentation of the top words, and an indirect confirmation measure that uses normalized pointwise mutual information (NPMI) and the cosine similarity. This CV coherence measure retrieves co-occurrence counts for the given words using a sliding window. The counts are used to calculate the NPMI of every top word to every other top word, thus, resulting in a set of vectors, one for every top word. The one-set segmentation of the top words leads to calculating the similarity between every top word vector and the sum of all top word vectors.

$$MPMI = \sum f(w,k|\alpha) = \alpha log(\varphi_{k,w}) + (1-\alpha) log(\frac{\varphi_{k,w}}{p_w})$$

Where $\varphi_{k,w}$ is the probability of word W in topic k, and p_w is the probability of the word w in the corpus. α is a value between 0 and 1 that determines the relative importance of first and second terms.

3. Recherche Questions

This paper has reviewed two models, Gibbs and NMF, used for topic modeling. We would like to examine several research questions:

I. How the structure of data affects the model's results. We will examine the models on two different types of datasets. One type contains a small number of documents, but each document contains a high number of words. A second type, on the other hand, has many documents. Still, each document contains a small number of words. According to the type of dataset, the quality of the models will be examined by the Coherence criterion.

II. Is there a correlation between the coherence criterion and human analysis. When we examine the models, the empirical measure we use is a Coherence criterion. However, we would like to perform additional human analysis. This analysis will be performed by our ability as humans to say whether we find a common topic between the words the models found in each topic. These results we will compare for the results obtained a Coherence criterion.

III. Will the models produce a similar optimal number of topics. One of the parameters that the model receives is the number of topics on which it performs the modeling. We will examine the optimal number of topics obtained from each model according to the Coherence criterion

IV. How the number of documents effect on the models. When we examine a data set that contains many samples, over a million in our case of the ABC dataset, we will want to take a smaller number due to computing power limits. We will examine the effect of the number of samples model results by the Coherence criterion.

V. Which of the models requires lower running time. We want to examine the time for each model to perform on the same data, and these will help us analyze the processing time required for each of the models.

4. Methodology

4.1 Data Set

In our work, we will use two datasets to examine the models, 'ABC-news' and 'CNN-news'. Both contain sentences from the media, so they have many topics. The main difference between them is the length of the average sample. Where in ABC dataset is 731.8 compared to 6.5 in CNN. The number of samples in the CNN dataset is 301 compared to over 1 million samples on the ABC dataset. We will examine up to 300,000 documents from the ABC dataset in our work.

4.2 Pre-Processing

To work with the raw data, we want to do two things, first is to transform the text to a suitable format for the model. Second, we want to clean the words from irrelevant information. Therefore, we will perform several actions such as splitting the sentences into tokens based on a word, dropping characters that are not letters, and dropping 'stop word', which are words with no meaningful semantic meaning, such as 'I', 'me', 'our'. Additionally, we will perform stemming, which conveys a word to the base form.

5. Results

After the pre-processing, we run each model, Gibbs, and NMF, on a topic range of 5 to 30.

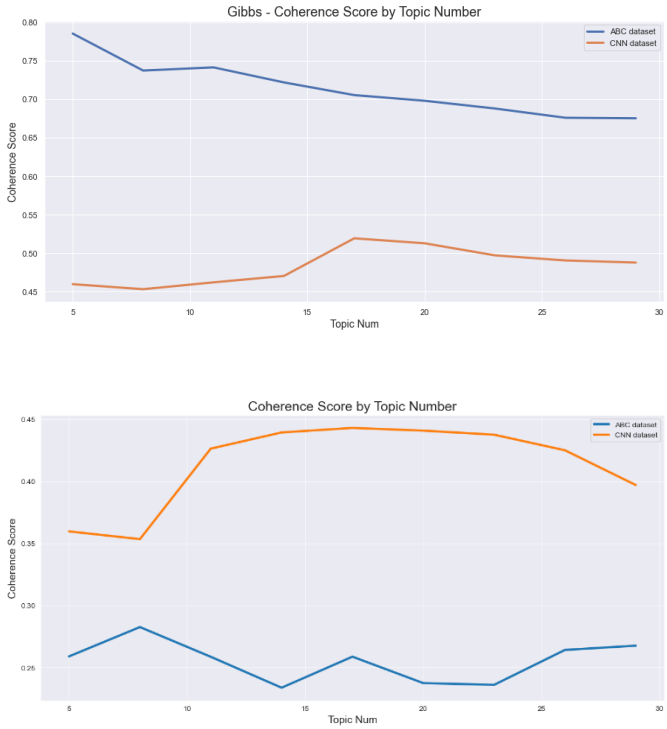

Figure 5: Gibbs (upper) and NMF(lower) Coherence score by topic number

We will run each of them on the optimal topic number that the Coherence measure found to test our models. We will present the first eight topics from each model on the CNN dataset.

Topic	Gibbs Top Words	NMF Top Words
1	amazon social facility virus distancing quarantine	instacart worker shopper custom gig order
2	mask ventilator production produce patient face	oil saudi price arabia russia barrel
3	plan announce furlough step part executive	loan payment small busi bank mortgag
4	cut low production global world barrel	trump fox news brief hanniti presid
5	game life family lot school event	airlin flight passeng carrier fli travel
6	order delivery state essential stay local	app user zoom data googl inform
7	financial stock share investor fund dollar	ventil ford devic patient product design
8	video release live zoom film studio	store retail close open maci tjx

Table 1: Top words in each topic

In addition, we will examine the runtime of the two models. The time is measured according to the time it took for the models to produce the graphs in figures (4) and (5).

	Gibbs model	**NMF model**
CNN dataset	589 [s]	185 [s]
ABC dataset	745 [s]	630 [s]

Table 2: Running time comparison

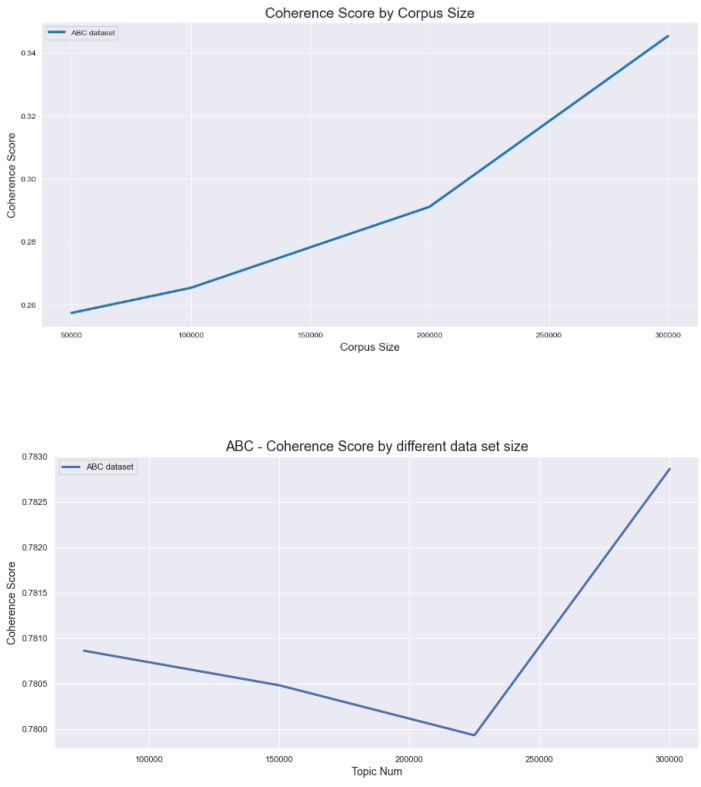

Figure 6: Gibbs (upper) and NMF(lower) Coherence score by documents number

6. Results Analysis

6.1. Comparing the two models in figure (5), NMF and Gibbs, according to the Coherence measure, we see that the models presented considerably similar results in the CNN database, which has a small number of documents (300). In contrast, in the ABC database with a large number of documents (300,000), the results in the NMF model decreased significantly compared to the Gibbs model, which presents better results over the CNN dataset.

6.2. When we analyze the results in table (1), we will examine the models' words in each topic and see if we can understand what they have in common in the real world. We can see that the NMF model presents topics that are easy to identify. For example, we can say that topics 1-6 are: Oil & Gas industry, Economy, Politic, Flights, Technology, Cars. Gibbs topics top words, we see that it is more difficult for us to find a common topic between the words. These results contradict the results obtained according to the Coherence criterion.

6.3. In figure (5), we see that the optimal number of topics obtained in each model is relatively similar. The CNN dataset received 18,17, and in the ABC dataset, 5, between Gibbs and NMF, respectively. These results and reflect the distribution and structure of each dataset.

6.4. From looking at figure (2), we see that in the NMF model, there is a consistent increase as the number of documents increases. However, in the Gibbs model, we see a decrease and then an increase. These results do not reconcile with logic since we would expect results similar to the graph in the NMF model, i.e., the increase in the documents produces a greater value of the coherence criterion. We can assume that this anomaly is due to a local minimum with a less good result than in a run with a smaller number of documents.

6.5. Table (2) shows that NMF has a significantly lower runtime for the CNN dataset with few documents. On the other hand, the runtime increases considerably in the ABC dataset that contains many documents. However, in the Gibbs model, the runtime does not vary significantly between the two types of data sets. These results are consistent with the literature and theory. In the NMF model, an increase in the number of documents causes an increase in the V matrix, which increases the complexity of the problem. Compared to Gibbs, which can handle a large number of documents, its runtime depends on the total amount of data (when we assume constant parameters) .

7. Conclusions

In this paper, we reviewed two different methods used for topic modeling, and we examined the models using several ways. By comparing the results, we obtained in the Coherence measure and the human analysis, and we see a contradiction between the results. That shows the importance of human analysis and the problem of relying on the Coherence measure. In addition, the results are consistent with the literature we have reviewed. Generally, a dataset is unsuitable for topic modeling for short-length documents, too small datasets, or many topics within a collection. Fine-tuning those values ranges is an important task involving a combination of human analysis and measure methods (as coherence score). There are guidelines [15] for handling this topic. Sufficiently minimum size of documents (M) is crucial for good modeling. However, Once a viable M value is achieved, further increasing it may not significantly improve performance. It is also recommended to avoid selecting an overly large k value – It might cause inference to become inefficient. It can be said that the Gibbs model works better on a more extensive database compared to NMF, which manages to present topics that are better humanly discernible, especially on databases that contain a small number of documents. In that sense, each of these models can adapt to different problems.

REFERENCES

1. Onan, A., Korukoglu, S., & Bulut, H. (2016). LDA-based Topic Modelling in Text Sentiment Classification: An Empirical Analysis. *Int. J. Comput. Linguistics Appl.*, *7*(1), 101-119.

2. Darling, W. M. (2011, December). A theoretical and practical implementation tutorial on topic modeling and gibbs sampling. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 642-647).

3. Tong, Z., & Zhang, H. (2016, May). A text mining research based on LDA topic modelling. In *International Conference on Computer Science, Engineering and Information Technology* (pp. 201-210).

4. Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences*, *101*(suppl 1), 5228-5235.

5. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, *3*, 993-1022.

6. Resnik, P., & Hardisty, E. (2010). *Gibbs sampling for the uninitiated*. Maryland Univ College Park Inst for Advanced Computer Studies.

7. Suri, P., & Roy, N. R. (2017, February). Comparison between LDA & NMF for event-detection from large text stream data. In *2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT)* (pp. 1-5). IEEE.

8. M'sik, B., & Casablanca, B. M. (2020). Topic modeling coherence: A comparative study between lda and nmf models using covid'19 corpus. *International Journal*, *9*(4).

9. *A.M. Johansen, in International Encyclopedia of Education (Third Edition), 2010, Pages 245-252*

10. *J.L. Tobias, in Encyclopedia of Health Economics, 2014, Pages 146-154*

11. *D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. Journal of Machine Learning Research, 3:993--1022, 2003.*

12. *TomasHrycej , in Artificial Intelligence Volume 46, Issue 3, December 1990, Pages 351-363*

13. *KDD '08: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data miningAugust 2008 Pages 569–577*

14. *Dave Binkley, in Understanding LDA for Software Engineering. Loyola University Maryland Baltimore MD, 21210, USA*

15. *ICML'14: Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32June 2014 Pages I-190–I-198*

16. *Michael Röder, Exploring the Space of Topic Coherence Measures*

17. *Hoyer PATRIKHOYER, P. O. (2004). Non-negative Matrix Factorization with Sparseness Constraints. In Journal of Machine Learning Research (Vol. 5).*

18. *O'Callaghan, D., Greene, D., Carthy, J., & Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. Expert Systems with Applications, 42(13), 5645–5657*

19. *Xu, W., Liu, X., & Gong, Y. (2001). Document Clustering Based On Non-negative Matrix Factorization*