Improving Topic Model Stability for Effective Document Exploration

Yi Yang*
University of Illinois
at Urbana-Champaign

Shimei Pan
University of Maryland
Baltimore County

Yangqiu Song
West Virginia University

Jie Lu
IBM T. J. Watson Research Center

Mercan Topkara
JW Player

Abstract

Topic modeling has become a ubiquitous topic analysis tool for text exploration. Most of the existing works on topic modeling focus on fitting topic models to input data. They however ignore an important usability issue that is closely related to the end user experience: stability. In this study, we investigate the stability problem in topic modeling. We first report on the experiments conducted to quantify the severity of the problem. We then propose a new learning framework to mitigate the problem by explicitly incorporating topic stability constraints in model training. We also perform user study to demonstrate the advantages of the proposed method.

1 Introduction

In an age of information explosion, statistical topic modeling techniques have frequently been used to structuralize large text collections by grouping documents into coherent topics. Latent Dirichlet Allocation (LDA) [Blei *et al.*, 2003] is one of the most commonly used approaches to topic modeling due to its simplicity and its capability to uncover hidden thematic patterns in text with little human supervision.

Much of the prior research on topic modeling has focused on enhancing the predictive accuracy of the learned topic models, i.e., by reducing perplexity. So far, not enough attention has been paid to the end user experience. In this research, we investigate topic model stability, a key usability factor that may significantly affect the experience of users.

Let us begin with an example that illustrates the problem we are addressing. Consider the example shown in Figure 1. Suppose that Alice uses a common LDA-based topic modeling tool to organize news articles on Middle East conflicts into several topics. Each topic is represented by the top keywords and the top representative documents [Alexander *et al.*, 2014; Lai *et al.*, 2014; Chaney and Blei, 2012]. Alice decides to focus on two topics: "Syria War" and "Iraq Violence." Due to a system upgrade, Alice needs to retrain the topic model on the same dataset. After the model is retrained, Alice notices that the topics have changed. For example, the top keywords of the two topics she is interested in now contain both "Syria War" and "Iraq Violence" keywords. In addition, the article she read previously about the killings in Iraq now appears together with "Syria War" articles. In a way, the mental map Alice has built for the news articles is disrupted, resulting in confusion and distrust.

The above example illustrates the instability of topic models when they are retrained on the same dataset. In reality, updating topic models regularly is also a common practice for many content collections due to their highly dynamic and constantly growing inventory and the need to capture the changes of topics over time. The instability gets even worse when the model is updated after new documents are added to the system. In this paper, we address the stability problem in a topic mode by first quantifying its severity, and then proposing a new learning framework to mitigate the problem by incorporating topic stability constraints in model training.

2 Stability Analysis

The stability of machine learning algorithms has been studied previously. A learning algorithm is said to be *unstable* if it is sensitive to small changes in the training data. Turney describes how in one study Decision Tree algorithms failed to be adopted by process engineers to help them understand the sources of low yield in a manufacturing process [Turney, 1995]: "... the engineers are disturbed when different batches of data from the same process result in radically different decision trees. The engineers lose confidence in the decision trees, even when we can demonstrate that the trees have high predictive accuracy...." Recent work on recommender systems [Adomavicius and Zhang, 2012] also shows that unstable recommendations have a negative impact on the user's trust and acceptance of recommender systems.

The potential impact of topic model instability on end user experience is multifold. (1) *Topic Comprehension*. The top N topic keywords inferred by a topic model are often used by end users to interpret the meaning of a topic. When the top N topic keywords are changed during model retraining or update, a user's capability in interpreting the semantics of a topic may be disrupted. (2) *Document Recall*. Topic models are often used to organize documents into coherent topics. Any changes in the topics assigned to the documents may disrupt a user's capability in locating existing documents since the user may still expect to find them under the old topics. (3)

*Corresponding Author: yiyang@illinois.edu

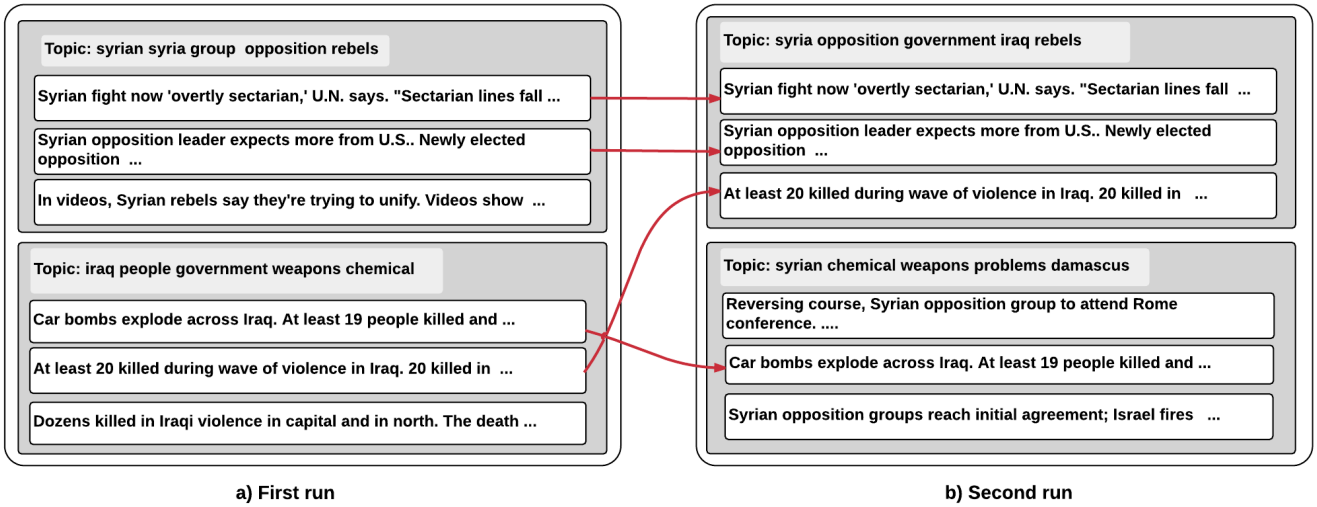


Figure 1: Topic modeling results trained on the same dataset in two different runs. The gray box represents each topic, which consists of a topic keywords list, and three representative documents. After the model is retrained, both the keywords and representative documents change. Red arrow shows the misplaced documents.

Trust. When different runs of the algorithm on the same input data give different results, users may consider the system unreliable and thus untrustworthy.

We define three different **topic model stability** measures to formally quantify the severity of the problem. Here we denote D_1 as the dataset used to train the old topic model M_1 with K topics, and D_2 as the new dataset. D_2 is empty if the model is retrained on the same data. $D_1 \cup D_2$ is used to train the new topic model M_2 with K topics. $I(.)$ is the indicator function. Here, we assume the topic indexes in M_1 and M_2 have been aligned.

1. Document topic assignment stability S_d. Here, l_{1i} and l_{2i} are the topic labels assigned to the ith document based on M_1 and M_2.

$$S_d = (1 - \frac{\sum_{d_i \in D_1} I(l_{1i} \neq l_{2i})}{|D_1|}) * 100\% \quad (1)$$

2. Topic keywords stability S_k. It measures the average percentage of unchanged topic keywords in M_1 and M_2. Each topic is represented by a set of N keywords (N is 10 in our experiments). K_{1t} and K_{2t} denote the keywords set of the tth topic in M_1 and M_2 respectively.

$$S_k = \frac{\sum_t^K |K_{1t} \cap K_{2t}|}{K * N} * 100\% \quad (2)$$

3. Token topic assignment stability S_t. Here, l_{1ij} and l_{2ij} are the topic labels assigned to the jth token in the ith document by M_1 and M_2.

$$S_t = (1 - \frac{\sum_{d_i \in D_1} \sum_{j \in d_i} I(l_{1ij} \neq l_{2ij})}{\sum_{d_i \in D_1} \sum_{j \in d_i}}) * 100\% \quad (3)$$

According to the definitions, stability is measured on a non-negative continuous scale, larger values indicate more stable topic model update.

To quantitatively measure the stability of a topic model, we experiment with two standard datasets, 20 Newsgroup [1] and NIPS [2]. We preprocess the datasets and train LDA models using Mallet [McCallum, 2002].We vary the number of new documents in D_2 from zero ($\emptyset$) to half of D_1 ($|D_2| = |D_1|/2$) and finally to the same as D_1 ($|D_2| = |D_1|$). As shown in Table 1, multiple runs of LDA on exactly the same dataset can produce significantly different topic assignments for documents. On average, only 56% of the documents are assigned the same topic labels, and 64% of the topic keywords stay unchanged. The number is even worse in topic model update scenario when new documents are trained together with old documents.

Table 1: Topic model stability in different scenarios.

Dataset		$D_2 = \emptyset$	$\|D_2\| = \|D_1\|/2$	$\|D_2\| = \|D_1\|$
20 NG	S_d	52.7%	48.3%	44.5%
	S_k	66.4%	62.4%	58.0%
	S_t	42.2%	39.6%	38.2%
NIPS	S_d	60.3%	47.3%	43.3%
	S_k	61.5 %	58.6%	57.4%
	S_t	39.2%	33.6%	30.8%

We speculate that there might be several reasons behind the instability problem in topic modeling: *different local optima*, *model convergence* and *new data*. First, in topic modeling, since computing the posterior distributions of model parameters is computationally intractable, approximate inference methods such as Gibbs sampling are often used. Since LDA is a non-convex model, when initialized with different random seeds, different runs of these methods may converge

[1] http://qwone.com/ jason/20Newsgroups/

[2] https://archive.ics.uci.edu/ml/datasets/Bag+of+Words

to different local optima. Thus, even with the same input data, the same inference algorithm may produce different results on two separate runs. Second, for Gibbs Sampling, frequently there is no specific criterion to test the convergence of the model. Thus in practice, we often use a pre-determined iteration number (e.g., 1000). Since with different random seeds, different runs of the same inference algorithm may have different convergence speeds, using a pre-determined iteration number may cause some runs to end prematurely. Models that ended prematurely may produce very different results, which may cause the instability of the inference results. Third, since fitting the input data is the main optimization criterion in topic model training, when new data are added into the input data, it is expected that the model would need to adjust its parameters to fit the new data.

3 Stable Topic Model Update

The stability analysis results show that traditional topic modeling outcome is unstable both when the model is retrained on the same input documents and when it is updated with new documents. Since retraining is a special case of model update (when $D_2 = \emptyset$), here we focus on improving topic model stability during model update.

We identified two ways of keeping topic model stable: (1) keep the top keywords associated with each topic unchanged; (2) keep the topic labels assigned to existing documents unchanged. We believe the first option is too rigid. Since the change of topic keywords is the natural result of topic evolution, keeping top topic keywords unchanged is not a viable solution.

In this work, we focus on maintaining the stability of topic assignments of existing documents (e.g., even though the keywords representing the 2016 presidential election evolve over time, existing documents about the election should still be labeled with the same topic). Some topic model extensions such as Labeled LDA [Ramage *et al.*, 2009], can be used to force the documents' topic labels to remain unchanged. However, due to the dynamic nature of the topics in topic models, the concept of a topic label is not well-defined. The topic index in a topic model is also interchangeable and not directly associated with any meaning. Thus, in this work, we focus on keeping documents sharing similar topics together to maintain stability. Specifically, we employ document pairwise constraints to encode document topic assignment stability. Two types of document pairwise constraints are used: *must-link*, which suggests that two documents are likely to share the same topics, and *cannot-link*, which suggests that two documents are about different topics. The pairwise document constraints do not force documents to stay in fixed topics. They only encourage documents sharing similar topics to stay together during model update.

So far, only a few methods are reported to be capable of incorporating document pairwise constraints in topic models [Andrzejewski *et al.*, 2011; Yang *et al.*, 2014; Xie *et al.*, 2015; Yang *et al.*, 2015]. All of these methods take a document collection D, as well as a set of document must-link constraints $\mathcal{M}$ and cannot-link constraints $\mathcal{C}$ as input and infer topic models that fit the input data while satisfying as many constraints as possible. In this study, we adopt [Yang *et al.*, 2014]'s approach.

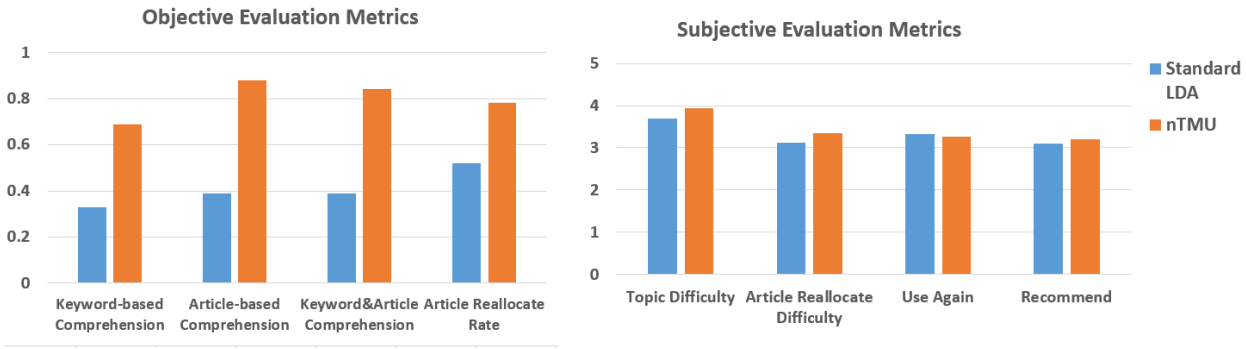
Specifically, we develop a **n**on-disruptive **T**opic **M**odel **U**pdate (nTMU) method to maintain stability during model update. Due to the interchangeability of topic indexes in topic models, even two equivalent models with exactly the same model parameters could have mis-matched topic indexes. To facilitate the comparison of two topic models, we use the Hungarian algorithm to align the topic indexes in two models. The details of nTMU is summarized in Algorithm 1. Given dataset D_1, a topic model is firstly trained with LDA (line 1). Based on the model, we assign a topic label for each document (line 2-4). Here, we adopt [Song *et al.*, 2009] method and define the topic label of document d as $l_d = \arg\max_{k \in K} \theta_{dk} / \sum_{i=0}^{D} \theta_{ik}$, where θ_i is the topic distribution of the ith document, and θ_{ik} is the probability mass of the kth topic. During model update, by default, we generate S document pairwise constraints. S is empirically set based on the dataset size, and in this work, we set it to be the size of D_1. We add must-link constraints to two documents if they have the same topic label and add cannot-link constraints if they do not have the same topic label (line 5-12). Optionally, we can let the end users to specify which topics to keep stable. The user's choice will then be converted to must-link and cannot-link constraints. Finally, the new topic model is trained on $D_1 \bigcup D_2$ with constraints.

Algorithm 1: non-disruptive Topic Model Update

Input: Dataset D_1 and D_2. Constraints Size S.
1 train topic model on D_1 with LDA.
2 **for** *each document* $d \in D_1$ **do**
3 topic label $l_d = \arg\max_{k \in K} \theta_{dk} / \sum_{i=0}^{|D_1|} \theta_{ik}$.
4 **end**
5 must-link set $\mathcal{M} = \emptyset$; cannot-link set $\mathcal{C} = \emptyset$; s=0.
6 **while** $s! = S$ **do**
7 randomly select two documents $u, v \in D_1$.
8 **if** $l_u == l_v$ **then** $\mathcal{M} = \mathcal{M} \bigcup (l_u, l_v)$;
9 `// it is a must-link constraint`
10 **else** $\mathcal{C} = \mathcal{C} \bigcup (l_u, l_v)$;
11 `// it is a cannot-link constraint`
$s = s + 1$;
12 **end**
13 train topic model on $D_1 \bigcup D_2$ with constraints $\mathcal{M}$ and $\mathcal{C}$.

4 Quantitative Evaluation

To evaluate the effectiveness of our method, we compare its performance with three baseline methods: 1). **Standard LDA**: jointly resamples all topics for both the old and the new documents from scratch. 2). **Fixed Fold-in**: jointly resamples topics for all the new documents, but keeps the old documents' topic samples fixed. Therefore, old documents' topic samples are used to initialize the topic model, but they are not updated. Note that although this method keeps all old documents' topic sample fixed, it does not necessarily mean that the topic labels for the old documents will be the same



(a) (b)

Figure 2: User study evaluation results.

before and after the update because document's topic distribution θ is a weighted distribution that is normalized over all the documents, including the new documents. 3). **Rejuvenated Fold-in**: in addition to jointly resampling topics for all new documents, updates old documents' topic samples. Therefore, after being used to initialize the topic model, old documents' topic samples will be updated in this method.

We conduct an experiment using the NIPS dataset. We simulate the update process by splitting the dataset into two halves based on the documents' timestamps. We compute the document topic assignment stability S_d using Equation 1. Results show that nTMU significantly outperforms the other methods in maintaining topic stability (43.3% for LDA, 60.2% for Fixed fold-in, 57.8% for Rejuvenated fold-in and 88.1% for nTMU). Its improvement over standard LDA is 103%. It also improves over the second best method (Fixed fold-in) by 46.3%.

5 User Experience Evaluation

We conducted a user study to evaluate the impact of topic stability on users when they perform a news exploration task. This news dataset includes 320 CNN news articles from October 2012 to November 2013, covering five prominent topics at the time including *"Fiscal Cliff"*, *"Hurricane Sandy"*, *"Violence in Iraq"*, *"Obamacare"*, and *"Syrian Civil War."* We split the articles into two halves based on their timestamps. The first half is used to train an initial topic model using LDA. Then we updated the topic model by adding the second half. We employed a between-subject design, testing two different update algorithms, one using Standard LDA, the other using nTMU. We recruited 80 participants using Amazon Mechanical Turk (MTurk) (40 users participated in each condition in this study).

Four objective metrics were used to assess how topic stability affects a participant's mental model of the topics. The first three metrics 'Keyword Based Comprehension,' 'Article Based Comprehension,' and 'Keyword & Article Based Comprehension' assess whether a participant is able to comprehend a topic after model update based on the topic keywords, or the representative articles, or both keywords and representative articles. The fourth metric 'Article Locate Rate' measures a participant's ability to locate the articles s/he visited before the update[3].

We also collected responses to four subjective metrics using a post-task questionnaire to measure end user experience. 'Topic Difficulty' measures how difficult it is for a participant to understand the system-derived topics. 'Article Locate Difficulty' measures how difficult it is to locate the articles that a participant visited before. 'Use Again' and 'Recommend' measure the likelihood of a participant to use the system again and the likelihood of recommending the system to others. All the survey questions are rated on a 5-point Likert scale with 1 being the least desirable and 5 the most desirable.

As shown in Figure 2(a), nTMU makes it easier for a participant to understand a topic and select a correct topic label, given topic keywords (0.688 with nTMU v.s. 0.338 with LDA). It also makes it easier to choose a correct topic label based on the representative articles (0.875 with nTMU versus 0.388 with LDA). Finally, keeping the topic model stable can significantly improve a participant's chance to locate an article again (0.775 with nTMU v.s. 0.525 with LDA). All the differences are statistically significant with $p < 0.001$ using independent sample t-test. In addition, as shown in Figure 2(b), nTMU also outperforms LDA in three out of the four subjective evaluation dimensions, although the differences are not as significant as those in objective evaluation.

6 Conclusion

The stability of a learning algorithm, if neglected, may significantly impact its usability. In this paper, we quantitatively measure the severity of the stability problem in topic models. We also present a method to directly address the problem by incorporating topic stability constraints in model training. We hope our work will help text mining practitioners to overcome the topic model instability problem while they develop real world applications. Moreover, we want to raise the awareness of the importance of usability in developing machine learning algorithms to facilitate their wide adoption in the real world.

[3]Before model update, users are asked to select an article to read and then answer a few questions. They are asked to find the article again after model update.

References

[Adomavicius and Zhang, 2012] Gediminas Adomavicius and Jingjing Zhang. Stability of recommendation algorithms. *ACM Transactions on Information Systems*, 30(4):23:1–23:31, November 2012.

[Alexander *et al.*, 2014] Eric Alexander, Joe Kohlmann, Robin Valenza, Michael Witmore, and Michael Gleicher. Serendip: Topic model-driven visual exploration of text corpora. In *Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 173–182. IEEE, October 2014.

[Andrzejewski *et al.*, 2011] David Andrzejewski, Xiaojin Zhu, Mark Craven, and Benjamin Recht. A framework for incorporating general domain knowledge into latent dirichlet allocation using first-order logic. In *International Joint Conference on Artificial Intelligence*, pages 1171–1177, 2011.

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[Chaney and Blei, 2012] Allison Chaney and David Blei. Visualizing topic models. In *International Conference on Weblogs and Social Media*, 2012.

[Lai *et al.*, 2014] Jennifer Lai, Jie Lu, Shimei Pan, Danny Soroker, Mercan Topkara, Justin Weisz, Jeff Boston, and Jason Crawford. Expediting expertise: Supporting informal social learning in the enterprise. In *ACM Conference on Intelligent User Interfaces*, pages 133–142, 2014.

[McCallum, 2002] Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet, 2002.

[Ramage *et al.*, 2009] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP*, pages 248–256, 2009.

[Song *et al.*, 2009] Yangqiu Song, Shimei Pan, Shixia Liu, Michelle X. Zhou, and Weihong Qian. Topic and keyword re-ranking for lda-based topic modeling. In *Conference on Information and Knowledge Management*, pages 1757–1760, 2009.

[Turney, 1995] Peter D. Turney. Technical note: Bias and the quantification of stability. *Machine Learning*, 20(1-2):23–33, 1995.

[Xie *et al.*, 2015] Pengtao Xie, Diyi Yang, and Eric P Xing. Incorporating word correlation knowledge into topic modeling. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2015.

[Yang *et al.*, 2014] Yi Yang, Shimei Pan, Doug Downey, and Kunpeng Zhang. Active learning with constrained topic model. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 30–33. Association for Computational Linguistics, 2014.

[Yang *et al.*, 2015] Yi Yang, Doug Downey, and Jordan Boyd-Graber. Efficient methods for incorporating knowledge into topic models. In *Conference on Empirical Methods on Natural Language Processing*, pages 308–317, 2015.