Suggesting Constraints to Interactive Topic Modeling

Yuening Hu YNHU@CS.UMD.EDU
Department of Computer Science, University of Maryland, College Park, MD

Jordan Boyd-Graber JBG@UMIACS.UMD.EDU
iSchool and Umiacs, University of Maryland, College Park, MD

1. Introduction

Understanding large amounts of unstructured text is a common information challenge. In contrast to the problem of retrieval, the problem of finding a specific fact or document, often the goal associated with large text collections is to *understand* and *explore* the data. Topic models, exemplified by latent Dirichlet allocation (LDA) (Blei et al., 2003), offer an unsupervised technique to discover latent themes automatically. However, these models are often a "take it or leave it" proposition. Mistakes cannot be corrected, and there are no accessible techniques for non-technical experts to adapt models to their specific use cases.

Recent work has attempts to address this issue by injecting domain knowledge into topic models (Andrzejewski et al., 2009) and allowing users to refine topics interactively through interactive topic modeling (ITM) (Hu et al., 2011). ITM starts with a set of initial topics by running LDA, shows these topics to users, and then incorporate feedback, given in the form of constraints, to improve the model. These constraints take two forms:

- positive constraints, which encourage a set of words to appear in the same topic.
- negative constraints, which push a set of words appear in the different topics.

Given these constraints, the underlying model is changed to better reflect the users' knowledge. The process of user input, adapting the model, and continuing inference repeats until users are satisfied with the topics.

However, ITM suffers from users having to search over a very large set of possible constraints. If the vocabulary size is V, there are about V^2 possible *pair* constraints, let alone higher order constraints.

Newman et al. suggest that pointwise mutual information (PMI) correlates with useful and coherent topics. While they focus on using PMI for evaluation, we attempt to use this insight for suggesting potential constraints. We evaluate the efficacy of such techniques based on human judgements of coherence and extrinsic evaluation. We also discuss future evaluation using information-seeking tasks.

2. Generating New Constraints

In this section, we discuss a simple and intuitive way using PMI to generate both positive and negative constraints.

Newman et al. argues that topics with a high PMI score (summed over all pairs of words) implies this topic has a better coherence. Following their lead, we could consider PMI as the ratio of the cooccurence probability of a word pair within a small *local* context window and the independent probability of the two words. We can use this to generate constraints by suggesting positive constraints for pairs with high PMI and negative constraints for pairs with low PMI.

However, it is not enough. PMI is usually used to measure the association between two words. High PMI only implies that the two words have a high probability to appear together. To constraints that generate meaningful constraints, we expect word pairs not only with high association but also describe what a document is about, as measured by tf-idf (Salton, 1968). We rank word pairs based on how well they would work as a positive constraint (PC) and as a negative constraint (NC).

$$PC(X,Y) = \prod_{W \in X,Y} \max_d (\text{tf-idf}(W,d)) \cdot \text{PMI}(X,Y) \quad (1)$$

$$NC(X,Y) = \prod_{W \in X,Y} \max_d (\text{tf-idf}(W,d)) / \text{PMI}(X,Y) \quad (2)$$

Because the number of word pairs is very large, we only consider the word pairs from different topics for PC and the word pairs from the same topic for NC.

3. Experiments

In this section, two sets of experiments are designed to test the automatically generated constraints. We use the 20 Newsgroups corpus[1]. The topic number is set to be 20, and 100 iterations are ran to get the initial topics. Four rounds of interaction are performed with 50 iterations each round.

[1] http://people.csail.mit.edu/jrennie/20Newsgroups/, it contains 18846 documents divided into 20 constituent newsgroups. Natural Language Toolkit (Loper & Bird, 2002) is used to perform tokenization and remove stopwords. After deleting short documents, we have 15160 documents, including 9131 training documents and 6029 test documents (default split). Topic modeling was performed using the most frequent 5000 lemmas as the vocabulary.

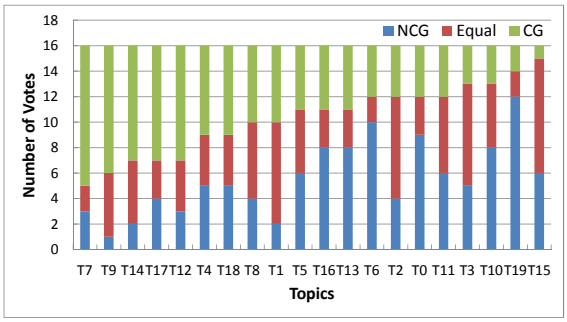

Figure 1. The vote for each topic by users on Mechanical Turk (16 votes for each topic). x-axis is in a decreasing order of the number of votes for CG.

Topic 7	car article cars insurance engine may water oil msg miles food medical disease doctor cancer health medicine body cause	
Round 1	SPLIT: (car, cancer)	
Shared	medical cancer may disease drug water food health treatment msg article cause cars doctor	
NCG/CG	engine anyone oil car ford used	medicine pain people dog aids effects

Table 1. The "evolution" of Topic 7 in the first round by adding one negative constraint, and we compared the topic words between CG and NCG. (Only show one round due to space limit.)

In the first experiment, we have two sets of topics: topics after automatically generating constraints (five positive and five negative constraints) to update the topics and control topics without constraints. We name the two groups as "CG" and "NCG" for short.

For evaluation, we showed the resulting topics to users on Mechanical Turk and asked whether they preferred the CG topics, the control NCG topics, or if they both looked equally coherent. Four users compared each pair. The positioning (i.e., left vs. right) and order of words within a topic were shuffled to avoid bias. Figure 1 shows that the results are inconclusive, although when users did prefer a topic they preferred the CG group.

Table 1 shows the change of Topic 7 when adding one negative constraint. While the initial topic seems to be a merge of "transportation" and "health", when a negative constraint between "car" and "cancer" was generated, "car" was pushed away from this topic. Though word "cars" was still there, the new related words like "medicine", "pain", "aids" and "effects" were popped up.

The second experiment is to use the wikipedia data to get the statistics to compute the correlations for word pairs, and we generate one negative constraint each time. For evaluation, we use two automatic evaluations: classification error rate (Hu et al., 2011) against *a priori* categories and PMI score.

Figure 2 shows the two evaluation results. With automatic

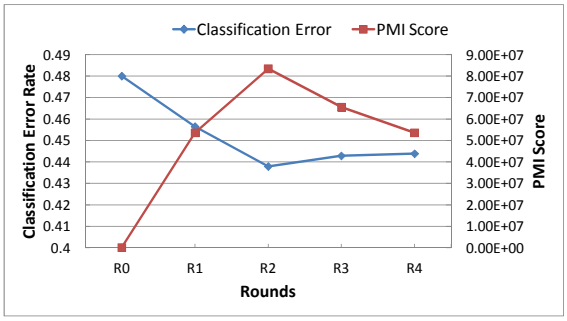

Figure 2. The results of two auto-evaluation: the results by classification error basically agrees with the results of PMI score.

Topic 7	car article medical water may food disease cars cancer doctor msg insurance engine health oil
Round 1	-
Words	car article medical water may food disease cars cancer doctor msg **good anyone drug engine**
Round 2	SPLIT: (engine, msg)
Words	car article medical water may food disease cars cancer doctor msg good anyone drug **cause**
Round 3	SPLIT: (doctor, msg)
Words	article medical water may food disease cancer doctor drug cause **treatment health study aids pain**
Round 4	-
Words	article medical water may food disease cancer doctor drug cause treatment health study aids **medicine**

Table 2. The "evolution" of Topic 7 by the constraints generated based on wikipedia data statistics. The bold words are those new words appear in this topic comparing with a previous topic.

constraints, the classification error is decreasing in the first two rounds. Table 2 shows the "evolution" of Topic 7. Since wikipedia is large enough to provide knowledge to generate constraint, we can see Topic 7 has better coherence after four rounds.

4. User Study and Future Work

We computed the correlation score based on the PMI and tf-idf to suggest constraints to ITM. While some topics were getting clearer and more focused, users had no clear preference between the topics with or without automatically generated constraints.

We intend to follow up this inconclusive result through a user study to examine whether ITM helps understand corpora better and the role of suggesting constraints used by a real user. Following the lead of Wacholder & Liu, we have created a study that will take a population and divide them into two groups. One group will have a static topic modeling visualization while the other will be able to add additional constraints. Both groups will then be asked to find answers to questions in the corpus (Wacholder & Liu, 2006). We hope to understand how users would use ITM and how it can help them find useful information in a corpus.

References

Andrzejewski, David, Zhu, Xiaojin, and Craven, Mark. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In *Proceedings of International Conference of Machine Learning*, 2009. ISBN 978-1-60558-516-1. doi: http://doi.acm.org/10.1145/1553374.1553378.

Blei, David M., Ng, Andrew, and Jordan, Michael. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Hu, Yuening, Boyd-Graber, Jordan, and Satinoff, Brianna. Interactive topic modeling. In *Association for Computational Linguistics*, 2011.

Loper, Edward and Bird, Steven. NLTK: the natural language toolkit. In *Tools and methodologies for teaching*, 2002. doi: 10.3115/1118108.1118117.

Newman, David, Lau, Jey Han, Grieser, Karl, and Baldwin, Timothy. Automatic evaluation of topic coherence. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

Salton, Gerard. *Automatic Information Organization and Retrieval.* McGraw Hill Text, 1968. ISBN 0070544859.

Wacholder, Nina and Liu, Lu. User preference: A measure of query-term quality. *JASIST*, 57(12):1566–1580, 2006.