Accounting ngrams and multi-word terms can improve topic models

Michael Nokel
Yandex,
Moscow, Russian Federation
mnokel@yandex-team.ru

Natalia Loukachevitch
Lomonosov Moscow State University,
Moscow, Russian Federation
louk_nat@mail.ru

Abstract

The paper presents an empirical study of integrating ngrams and multi-word terms into topic models, while maintaining similarities between them and words based on their component structure. First, we adapt the PLSA-SIM algorithm to the more widespread LDA model and ngrams. Then we propose a novel algorithm LDA-ITER that allows the incorporation of the most suitable ngrams into topic models. The experiments of integrating ngrams and multiword terms conducted on five text collections in different languages and domains demonstrate a significant improvement in all the metrics under consideration.

1 Introduction

Topic models, such as PLSA (Hofmann, 1999) and LDA (Blei et al., 2003), have shown great success in discovering latent topics in text collections. They have considerable applications in the information retrieval, text clustering and categorization (Zhou et al., 2009), word sense disambiguation (Boyd-Graber et al., 2007), etc.

However, these unsupervised models may not produce topics that conform to the user's existing knowledge (Mimno et al., 2011). One key reason is that the objective functions of topic models do not correlate well with human judgements (Chang et al., 2009). Therefore, it is often necessary to incorporate semantic knowledge into topic models to improve the model's performance. Recent work has shown that interactive human feedback (Hu et al., 2011) and information about words (Boyd-Graber et al., 2007) can improve the inferred topic quality.

Another key limitation of the original algorithms is that they rely on a "bag-of-words" assumption, which means that words are assumed to be uncorrelated and generated independently. While this assumption facilitates computational efficiency, it loses the rich correlations between words. There are several studies, in which the integration of collocations, ngrams and multi-word terms is investigated. However, they are often limited to bigrams (Wallach, 2006; Griffiths et al., 2007) and often result in a worsening of the model quality due to increasing the size of a vocabulary or to a complication of the model, which requires time-intensive computation (Wang et al., 2007).

The paper presents two novel methods that take into account ngrams and maintain relationships between them and the words in topic models (e.g, *weapon – nuclear weapon – weapon of mass destruction; discrimination – discrimination on basis of nationality – racial discrimination*). The proposed algorithms do not rely on any additional resources, human help or topic-independent rules. Moreover, they lead to a huge improvement of the quality of topic models.

All experiments were carried out using the LDA algorithm and its modifications on five corpora in different domains and languages.

2 Related work

The idea of using ngrams in topic models is not a novel one. Two kinds of methods are proposed to deal with this problem: the creation of a unified topic model and preliminary extraction of collocations for further integration into topic models.

Most studies belong to the first kind of methods and are limited to bigrams: i.e, the Bigram Topic Model (Wallach, 2006) and LDA Collocation Model (Griffiths et al., 2007). Besides, Wang et al. (2007) proposed the Topical N-Gram Model that allows the generation of ngrams based on the context. However, all these models are mostly

Proceedings of the 12th Workshop on Multiword Expressions, pages 44–49,
Berlin, Germany, August 7-12, 2016.

of theoretical interest since they are very complex and hard to compute on real datasets.

The second type of methods includes those proposed in (Lau et al., 2013; Nokel and Loukachevitch, 2015). These works are also limited to bigrams. Nokel and Loukachevitch (2015) extend the first work and propose the PLSA-SIM algorithm, which integrates top-ranked bigrams and maintains the relationships between bigrams sharing the same words. The authors achieve an improvement in topic model quality.

Our first method in the paper extends the PLSA-SIM algorithm (Nokel and Loukachevitch, 2015) by switching to ngrams and the more widespread LDA model. Also we propose a novel iterative LDA-ITER algorithm that allows the automatic choice of the most appropriate ngrams for further integration into topic models.

The idea of utilizing prior knowledge in topic models is not a novel one, but the current studies are limited to words. So, Andrzejewski et al. (2011) incorporated knowledge by Must-Link and Cannot-Link primitives represented by a Dirichlet Forest prior. These primitives were then used in (Petterson et al., 2010; Newman et al., 2011), where similar words are encouraged to have similar topic distributions. However, all such methods incorporate knowledge in a hard and topic-independent way, which is a simplification since two words that are similar in one topic are not necessarily of equal importance for another topic.

Also several works seek to utilize the domain-independent knowledge available in online dictionaries or thesauri (such as WordNet) (Xie et al., 2015). We argue that this knowledge may be insufficient in the particular text corpus.

Our current work proposes an approach to maintain the relationships between ngrams, sharing the same words. Our method does not require any complication of the original LDA model and just gives advice on whether ngrams and words can be in the same topics or not.

3 Proposed algorithms

First, we adapt the PLSA-SIM algorithm proposed in (Nokel and Loukachevitch, 2015). We argue that the more widespread model is LDA (Blei et al., 2003). So we transfer the idea of the PLSA-SIM algorithm to LDA and adapt it to multi-word expressions and terms of any length.

The main idea of the approach of including multi-word expressions into topic models is that similar ngrams sharing the same words (e.g, *hidden – hidden layer – hidden Markov model – number of hidden units*) often belong to the same topics, under one important condition that they often co-occur within the same texts.

To implement the approach, we introduce the sets of similar ngrams and words: $S = \{S_w\}$, where S_w is the set of ngrams similar to w, that is $S_w = \{w \bigcup_n (\bigcup_{w_1 \dots w_n : \exists i : w_i = w} w_1 \dots w_n)\}$, where w is the lemmatized word, and $w_1 \dots w_n$ is the lemmatized ngram. While adding ngrams to the vocabulary as single tokens, we decrease the frequencies of unigram components by the frequencies of encompassing ngrams in each document d. The resulted frequencies are denoted as n_{dw}.

The pseudocode of the resulting LDA-SIM algorithm is presented in Algorithm 1.

Algorithm 1: LDA-SIM algorithm

Input: collection D, vocabulary W, number of topics $|T|$, initial $\{p(w|t)\}$ and $\{p(t|d)\}$, sets of similar ngrams S, hyperparameters $\{\alpha_t\}$ and $\{\beta_w\}$

Output: distributions $\{p(w|t)\}$ and $\{p(t|d)\}$

while *not meet the stop criterion* **do**

  for $d \in D, w \in W, t \in T$ **do**

    $p(t|d,w) = \frac{p(w|t)p(t|d)}{\sum_{u \in T} p(w|u)p(u|d)}$

  for $d \in D, w \in W, t \in T$ **do**

    $n'_{dw} = n_{dw} + \sum_{s \in S_w} n_{ds}$

    $p(w|t) = \frac{\sum_{d \in D} n'_{dw} p(t|d,w) + \beta_w}{\sum_{d \in D} \sum_{w \in d} n'_{dw} p(t|d,w) + \sum_{w \in W} \beta_w}$

    $p(t|d) = \frac{\sum_{w \in d} n'_{dw} p(t|d,w) + \alpha_t}{\sum_{w \in W} \sum_{t \in T} n'_{dw} p(t|d,w) + \sum_{t \in T} \alpha_t}$

So, if similar ngrams co-occur within the same document, we sum up their frequencies during calculation of probabilities, trying to carry similar ngrams and words to the same topics. Otherwise we make no modification to the original algorithm.

Then we hypothesized that it is possible to automatically choose the most suitable ngrams to incorporate into topic models. For this purpose we can compose all possible ngrams from the top elements from each previously inferred topic and further incorporate them into a topic model (e.g., we can compose *"support vector machine"* from the top words *"machine"*, *"vector"*, *"support"*). To be precise, we can choose the most frequent ngram that can be composed from the given set of words.

To verify this hypothesis, we propose the novel **LDA-ITER** algorithm that utilizes the LDA and LDA-SIM algorithms (Algorithm 2). In fact, there is some similarity in extracting ngrams with the approach presented in (Blei and Lafferty, 2009), where the authors visualize topics with ngrams consisting of words mentioned in these topics. But in that approach the authors do not create a new topic model taking into account extracted ngrams.

Algorithm 2: LDA-ITER algorithm

Infer topics via the LDA algorithm using vocabulary W containing only words
while *not meet the stop criterion* **do**
  Form sets C_t from the top-10 elements from each topic t
  Form sets B_t containing all possible ngrams from the elements in each set C_t
  Create sets of similar ngrams and words $S = \bigcup_t (B_t \cup C_t)$
  Run LDA-SIM using set of similar ngrams and words S and vocabulary $W = W \bigcup \left(\bigcup_t B \right)$

In the proposed LDA-ITER algorithm we select top-10 elements from each topic at each iteration. We established experimentally that topic coherence does not depend highly on this parameter, while the best value for perplexity is achieved when selecting top-5 or top-7 elements. Nevertheless in all experiments we set this parameter to 10.

We should note that the number of parameters in the proposed algorithms equals to $|W||T|$ as in the original LDA, where $|W|$ is the size of vocabulary, and $|T|$ is the number of topics (cf. $|W|^N|T|$ parameters in the topical n-gram model (Wang et al., 2007), where N is the length of n-grams).

4 Datasets and evaluation

In our experiments we used English and Russian text collections in different domains (Table 1).

Text collection	Number of texts	Number of words
Russian banking texts	10422	$\approx$ 32 mln
English part of Europarl corpus[1]	9672	$\approx$ 56 mln
English part of JRC-Acquiz corpus[2]	23545	$\approx$ 53 mln
ACL Anthology Reference corpus[3]	10921	$\approx$ 48 mln
NIPS Conference Papers (2000–2012)[4]	17400	$\approx$ 5 mln

Table 1: Text collections for experiments

As the sources of multi-word terms, we took two real information-retrieval thesauri in the following domains: socio-political (EuroVoc thesaurus comprising 15161 terms) and banking (Russian Banking Thesaurus comprising 15628 terms). We used the Eurovoc thesaurus in the processing of the Europarl and JRC-Acquiz corpora. The Russian Banking Thesaurus was employed for the processing of Russian banking texts.

At the preprocessing step, documents were processed by morphological analyzers. We do not consider function and low frequency words as elements of vocabulary since they do not play a significant role in forming topics. Also we extracted all collocations in the form of the regular expression *((Adj|Noun)$^+$|(Adj|Noun)*(Noun Prep)$^?$(Adj|Noun)*)*Noun* (similar to the one proposed in (Frantzi and Ananiadou, 1999)). We take into account only such ngrams since topics are mainly identified by noun groups. Also we emphasize that the proposed sets of similar ngrams cannot be formed by prepositions.

As for the quality of the topic models, we consider three intrinsic measures. The first one is **Perplexity**, which is the standard criterion of topic quality (Daud et al., 2010):

$$Perplexity(D) = e^{-\frac{1}{n} \sum_{d \in D} \sum_{w \in d} n_{dw} \ln p(w|d)}, \quad (1)$$

where n is the number of all considered words in the corpus, D is the set of documents in the corpus, n_{dw} is the number of occurrences of the word w in the document d, $p(w|d)$ is the probability of appearing the word w in the document d.

Another method of evaluating topic models is topic coherence (**TC-PMI**) proposed by Newman

[1]http://www.stamt.org/europarl
[2]http://ipsc.jrc.ec.europa.eu/index.php?id=198
[3]http://acl-arc.comp.nus.edu.sg/
[4]http://www.cs.nyu.edu/~rowels/data.html

et al. (2010), which measures the interpretability of topics based on human judgment:

$$TC\text{-}PMI = \frac{1}{|T|}\sum_{j=2}^{10}\sum_{i=1}^{j-1}\log\frac{P(w_j, w_i)}{P(w_j)P(w_i)}, \quad (2)$$

where $(w_1, w_2, \ldots, w_{10})$ are the top-10 elements in a topic, $P(w_i)$, $P(w_j)$ and $P(w_j, w_i)$ are probabilities of w_i, w_j and ngram (w_j, w_i) respectively.

Following the idea of Nokel and Loukachevitch (2015), we also used the variation of this measure – **TC-PMI-nSIM**, which considers top-10 terms, no two of which are from the same set of similar ngrams. To avoid the effect of considering very long ngrams, we took the most frequent item in each found set of similar ngrams.

5 Experiments

To compare the proposed algorithms with the original one, we extracted all the ngrams in each text corpus. For ranking ngrams we used *Term Frequency* (*TF*) and one of the eight context measures: *C-Value* (Frantzi and Ananiadou, 1999), two versions of *NC-Value* (Frantzi and Ananiadou, 1997; Frantzi and Ananiadou, 1999), *Token-FLR, Token-LR, Type-FLR, Type-LR* (Nakagawa and Mori, 2003), and *Modified Gravity Count* (Nokel and Loukachevitch, 2013). We should note that context measures are the most well-known method for extracting ngrams and multi-word terms.

According to the results of (Lau et al., 2013) we decided to integrate the top-1000 ngrams and multi-word terms into all the topic models under consideration. We should note that in all experiments we fixed the number of topics $|T| = 100$ and the hyperparameters $\alpha_t = \frac{50}{|T|}$ and $\beta_w = 0.01$.

We conducted experiments with all **nine** aforementioned measures on all the text collections to compare the quality of the LDA, the LDA with top-1000 ngrams or multi-word terms added as "black boxes" (similar to (Lau et al., 2013)), and the LDA-SIM with the same top-1000 elements.

In Table 2 we present the results of integrating the top-1000 ngrams and multi-word terms ranked by *NC-Value* (Frantzi and Ananiadou, 1999) for all five text collections. Other measures under consideration demonstrate similar results.

As we can see, there is a huge improvement in topic coherence using the proposed algorithm in all five text collections. This means that the inferred topics become more interpretable. As for perplexity, there is also a significant improvement compared to LDA with ngrams as "black boxes". Moreover, sometimes the perplexity is even better than in the original LDA, although the proposed algorithm works on the larger vocabularies, which usually leads to the increase of perplexity.

Corpus	Model	Perplexity	TC-PMI	TC-PMI-nSIM
Banking	*LDA*	1654	81.3	81.3
	LDA + ngrams	2497.1	90.1	90.1
	LDA-SIM + ngrams	**1472.8**	**120.6**	**114.9**
	LDA-SIM + terms	**1621.4**	**133**	**118**
Europarl	*LDA*	1466.1	54	54
	LDA + ngrams	2084.9	53.6	53.6
	LDA-SIM + ngrams	**1343.4**	**122.1**	**121.2**
	LDA-SIM + terms	1594.7	**105.4**	**98.3**
JRC	*LDA*	807.7	64.1	64.1
	LDA + ngrams	1140.6	65.6	65.6
	LDA-SIM + ngrams	**795.8**	**85.4**	**80.4**
	LDA-SIM + terms	885.4	**76.6**	**73.9**
ACL	*LDA*	1779.8	73.4	73.4
	LDA + ngrams	2277.5	69.6	69.6
	LDA-SIM + ngrams	2059.3	**95.2**	**90.1**
NIPS	*LDA*	1284.4	72.2	72.2
	LDA + ngrams	1968.5	69.3	69.3
	LDA-SIM + ngrams	1526.7	**127.9**	**116.3**

Table 2: Results of integrating top-1000 ngrams and terms ranked by *NC-Value* into topic models

We should note that the results of the ACL and NIPS corpora are a little different. This is because the ACL corpus contains a lot of word segments hyphenated at ends of lines, while the NIPS corpus is relatively small.

At the last stage of the experiments, we compare the iterative and original algorithms. In Table 3 we present the results of the first iteration of the LDA-ITER algorithm (with the numbers of the added ngrams and terms) alongside the LDA.

As we can see, there is also an improvement in the topics, despite the fact that the LDA-ITER algorithm selects much more ngrams than in the experiments with the LDA-SIM. As for the multi-word terms, selecting just a few hundreds of them results in the similar or even better topic quality

Corpus	Model	Perplexity	TC-PMI	TC-PMI-nSIM
Banking	*LDA*	1654	81.3	81.3
	LDA-ITER + 2514 ngrams	**1448.8**	**106**	**108.7**
	LDA-ITER + 371 terms	**1384**	**101.6**	**99.7**
Europarl	*LDA*	1466.1	54	54
	LDA-ITER + 1848 ngrams	**1455.5**	**56.4**	**66.1**
	LDA-ITER + 210 terms	**1278.9**	**88.3**	**79.4**
JRC	*LDA*	807.7	64.1	64.1
	LDA-ITER + 2497 ngrams	**806.5**	**68.4**	**65.7**
	LDA-ITER + 225 terms	**741.5**	**73.8**	**70.2**
ACL	*LDA*	1779.8	73.4	73.4
	LDA-ITER + 2311 ngrams	1972.5	**95.9**	**79.7**
NIPS	*LDA*	1284.4	72.2	72.2
	LDA-ITER + 1161 ngrams	1434.2	**108**	**94.3**

Table 3: Results of integrating ngrams and multi-word terms into the LDA-ITER algorithm

than selecting regular ngrams. Thus, it seems very important that in the case of the LDA-ITER algorithm there is no need to select the desired number of integrating ngrams (cf. the LDA-SIM algorithm). We should also note that on the next iterations the results start to hover around the same values of the measures.

In Table 4 we present working time of the LDA-SIM and the first iteration of the LDA-ITER alongside the original LDA. All the algorithms conducted on a notebook with 2.1 GHz Intel Core i7-4600U and 8 GB RAM, running Lubuntu 16.04.

Corpus	LDA	LDA-SIM	LDA-ITER
Banking	11 min	13 min	11 min
ACL	13 min	15 min	16 min
Europarl	10 min	14 min	14 min
JRC	10 min	14 min	15 min
NIPS	1.75 min	2 min	1.75 min

Table 4: Working time of the algorithms

At the end, as an example of the inferred topics, we present in Table 5 the top-10 elements from the two random topics inferred by the LDA-SIM with 1000 most frequent ngrams and the first iteration of the LDA-ITER on the ACL corpus.

6 Conclusion

The paper presents experiments on integrating ngrams and multi-word terms along with similarities between them and words into topic models. First, we adapted the existing PLSA-SIM algorithm to the LDA model and ngrams. Then we propose the LDA-ITER algorithm, which allows us to incorporate the most suitable ngrams and multi-word terms. The experiments conducted on five text collections in different domains and languages demonstrate a huge improvement in all the metrics of quality using the proposed algorithms.

LDA-SIM	
translation model	*speech*
statistical machine translation	*speech recognition*
machine translation	*speech communication*
statistical translation	*spontaneous speech*
translation	*speech processing*
language model	*speech recognizer*
translation probability	*spoken language processing*
reference translation	*speech synthesis*
translation quality	*automatic speech*
translation system	*automatic speech recognition*
LDA-ITER	
translation model	*speech recognition system*
statistical translation model	*speech recognition*
source word	*speech*
machine translation	*recognition system*
translation	*speech system*
language model	*recognition*
statistical translation	*system*
target word	*speaker*
translation system	*speech recognizer*
model	*speak*

Table 5: Topics inferred by the LDA-SIM and LDA-ITER on the ACL corpus

Acknowledgments

This work is partially supported by RFBR grant N14-07-00383.

References

David Andrzejewski, Xiaojin Zhu, and Mark Craven. 2011. Incorporating domain knowledge into topic modeling via dirichlet forest priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 25–32.

David M. Blei and John D. Lafferty. 2009. Visualizing topics with multi-word expressions. `https://arxiv.org/pdf/0907.1013.pdf`.

David M. Blei, Andrew Y. Ng, and Michael Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(3):993–1002.

Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. 2007. A topic model for word sense disambiguation. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 1024–1033.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrich, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Proceedings of the 24th Annual Conference on Neural Information Processing Systems*, pages 288–296.

Ali Daud, Juanzi Li, Lizhu Zhou, and Faqir Muhammad. 2010. Knowledge discovery through directed probabilistic topic models: a survey. *Frontiers of Computer Science in China*, 2(2):280–301.

Katerina Frantzi and Sophia Ananiadou. 1997. Automatic term recognition using contextual cues. In *Proceedings of the IJCAI Workshop on Multilinguality in Software Industry: the AI Contribution*, pages 73–80.

Katerina Frantzi and Sophia Ananiadou. 1999. The c-value/nc-value domain-independent method for multi-word term extraction. *Journal of Natural Language Processing*, 6(3):145–179.

Thomas L. Griffiths, Mark Steyvers, and Joshua B. Tenenbaum. 2007. Topics in semantic representation. *Psychological Review*, 114(2):211–244.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57.

Yuening Hu, Jordan Boyd-Graber, and Brianna Satinoff. 2011. Interactive topic modeling. In *Proceedings of the Association for Computational Linguistics*, pages 248–257.

Jey Han Lau, Timothy Baldwin, and David Newman. 2013. On collocations and topic models. *ACM Transactions on Speech and Language Processing*, 10(3):1–14.

David Mimno, Hanna M. Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *Proceedings of EMNLP'11*, pages 262–272.

Hiroshi Nakagawa and Tatsunori Mori. 2003. Automatic term recognition based on statistics of compound nouns and their components. *Terminology*, 9(2):201–219.

David Newman, Jey Han Lau, Kari Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 100–108.

David Newman, Edwin V Bonilla, and Wray Buntine. 2011. Improving topic coherence with regularized topic models. In *Advances in Neural Information Processing Systems*, pages 496–504.

Michael Nokel and Natalia Loukachevitch. 2013. An experimental study of term extraction for real information-retrieval thesauri. In *Proceedings of the 10th International Conference on Terminology and Artificial Intelligence*, pages 69–76.

Michael Nokel and Natalia Loukachevitch. 2015. A method of accounting bigrams in topic models. In *Proceedings of North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, pages 1–9.

James Petterson, Wray Buntine, Shravan M Narayanamurthy, Tiberio S Caetano, and Alex J Smola. 2010. Word features for latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, pages 1921–1929.

Hanna M. Wallach. 2006. Topic modeling: Beyond bag-of-words. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 977–984.

Xuerui Wang, Andrew McCallum, and Xing Wei. 2007. Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 697–702.

Pengtao Xie, Diyi Yang, and Eric P. Xing. 2015. Incorporating word correlation knowledge into topic modeling. In *Proceedings of North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, pages 725–734.

Shibin Zhou, Kan Li, and Yushu Liu. 2009. Text categorization based on topic model. *International Journal of Computational Intelligence Systems*, 2(4):398–409.