Bigram Anchor Words Topic Model

Arseniy Ashuha[1] and Natalia Loukachevitch[2](✉)

[1] Moscow Institute of Physics and Technology, Dolgoprudny, Russia
ars.ashuha@phystech.edu
[2] Research Computing Center of Lomonosov Moscow State University,
Moscow, Russia
louk_nat@mail.ru

Abstract. A probabilistic topic model is a modern statistical tool for document collection analysis that allows extracting a number of topics in the collection and describes each document as a discrete probability distribution over topics. Classical approaches to statistical topic modeling can be quite effective in various tasks, but the generated topics may be too similar to each other or poorly interpretable. We supposed that it is possible to improve the interpretability and differentiation of topics by using linguistic information such as collocations while building the topic model. In this paper we offer an approach to accounting bigrams (two-word phrases) for the construction of Anchor Words Topic Model.

Keywords: Topic model · Anchor words · Bigram

1 Introduction

A probabilistic topic model is a modern statistical tool for document collection analysis that allows identifying a set of topics in the collection and describes each document as a discrete probability distribution over topics. The topic is meant a discrete probability distribution over words, considered as a thematically related set of words. Topic models are actively used for various applications such as text analysis [4,6,12], user analysis [8], information retrieval [10,14].

To recognize hidden topics, standard algorithms of topic modeling such as PLSA or LDA [3,7], take into account only the frequencies of words and do not consider the syntactic structure of sentences, the word order, or grammatical characteristics of words. Neglect of the linguistic information causes the low interpretability and the low degree of topic differentiation [14], which may hinder the use of topic models. If a topic has low interpretability then it may seem as a set of unrelated words or a mixture of several topics. It is difficult to differentiate topics when they are very similar to each other.

One of the approaches that improves the interpretability of the topics is proposed in [1,2] and is called Anchor Words. This approach is based on the assumption that in each topic there exists a unique word that describes the topic, but this approach is also built on word frequencies.

D.I. Ignatov et al. (Eds.): AIST 2016, CCIS 661, pp. 121–131, 2017.
DOI: 10.1007/978-3-319-52920-2_12

In this paper we put forward a modification of the Anchor algorithm, which allows us to take into account collocations when building a topic model. The experiments were conducted on various text collections (Banks Articles, 20 Newsgroups, NIPS) and confirmed that the proposed method improved the interpretability and the uniqueness of topics without downgrading other quality measures.

The paper is organized as follows. Section 2 reviews similar work. Section 3 describes the metrics used for evaluating the quality of topic models. In Sect. 4, we propose a method that allows us to take into account collocations in the Anchor Words topic model.

2 Related Work

2.1 Notation and Basic Assumptions

Many variants of topic modeling algorithms have been proposed so far. Researchers usually suppose that a topic is a set of words that describe a subject or an event; a document is a set of topics that have generated it. A **Topic** is a discrete probability distribution over words: topic $t = \{\mathrm{P}(w|t) : w \in W\}$ [3,7]. In this notation, each word in each topic has a certain probability, which may be equal to zero. Probabilities of words in topics are usually stored in the matrix $\Phi_{\mathrm{W}\times\mathrm{T}}$. A **Document** is a discrete probability distribution over topics $\mathrm{P}(t|d)$ [3,7]. These probabilities are represented as a matrix $\Theta_{\mathrm{T}\times\mathrm{D}}$.

In topic modeling, the following hypotheses are usually presupposed: a **Bag of words hypothesis** is the assumption that it is possible to determine which topics have generated the document without taking into account the order of words in the document; **Hypothesis of conditional independence** is the assumption that the topic does not depend on the document, the topic is represented by the same discrete distribution in each document that contains this topic. Formally, the probability of a word in the topic is not dependent on the document – $\mathrm{P}(w|d,t) = \mathrm{P}(w|t)$ [3,7,14]; **Hypothesis about the thematic structure of the document** usually assumes that the probability of a word in a document depends on the hidden topics that have generated the document, as, for example, in the simplest topic model:

$$p(w|d) = \sum_{t\in\mathrm{T}} \mathrm{P}(w|d,t)\mathrm{P}(t|d) = \sum_{t\in\mathrm{T}} \mathrm{P}(w|t)\mathrm{P}(t|d) \tag{1}$$

2.2 Specific Topic Models

In this section we consider several well-known approaches to topic modeling.

Probabilistic Latent Semantic Analysis, PLSA was proposed by Thomas Hoffman in [7]. To build the model, he proposed to optimize the log-likelihood with the restrictions of normalization and non-negativeness:

$$log\ L(D, \Phi, \Theta) = log\ \prod_{d \in \mathrm{D}} \prod_{w \in d} p(w|\ d) \rightarrow \max_{\Phi, \Theta} \tag{2}$$

$$\phi_{wt} \geq 0;\ \sum_{w \in W} \phi_{wt} = 1;\ \theta_{td} \geq 0;\ \sum_{t \in T} \phi_{td} = 1 \tag{3}$$

To solve the optimization problem, the EM-algorithm was proposed, which is usually used to find the maximum likelihood estimate of probability model parameters when the model depends on hidden variables.

Latent Dirichlet Allocation, LDA was proposed by David Blei in [3]. This paper introduces the generative model that assumes that the vectors of topics and the vectors of documents are generated from the Dirichlet distribution. For training the model, it was proposed to optimize the following function:

$$log\ \left[L(D, \Phi, \Theta) \prod_{d} Dir(\theta_d|\beta) \prod_{t} Dir(\phi_t|\alpha)\right] \rightarrow \max_{\Phi, \Theta} \tag{4}$$

$$\phi_{wt} \geq 0;\ \sum_{w \in W} \phi_{wt} = 1;\ \theta_{td} \geq 0;\ \sum_{t \in T} \phi_{td} = 1 \tag{5}$$

To solve the optimization problem, the authors use the Bayesian inference, which leads to EM-algorithm similar to PLSA. Because of the factored-conditional conjugate prior distribution and the likelihood, the formula for the parameters update can be written explicitly.

Additive Regularization Topic Model was proposed by Konstantin Vorontsov in [14]. The "Additive Regularization Topic Model" generalizes LDA (the LDA approach can be expressed in terms of an additive regularization) and allows applying a combination of regularizers to topic modeling by optimizing the following functional:

$$log\ L(D, \Phi, \Theta) + \sum_{i=1}^{n} \tau_i R_i(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta} \tag{6}$$

$$\phi_{wt} \geq 0; \quad \sum_{w \in W} \phi_{wt} = 1; \quad \theta_{td} \geq 0; \quad \sum_{t \in T} \phi_{td} = 1 \tag{7}$$

where, τ_i – weight of regularizer, R_i – regularizer.

To introduce regularizers, the Bayesian inference is not used. On the one hand, it simplifies the process of entering regularizers because it does not require the technique of Bayesian reasoning. On the other hand, the introduction of a new regularizer is an art, which is hard to formalize.

The paper [14] shows that the use of the additive regularization allows simulating reasonable assumptions about the structure of topics, which helps to improve some properties of a topic model such as interpretability and sparseness.

Anchor Words Topic Model was proposed by Sanjeev Arora in [1,2]. The basic idea of this method is the assumption that for each topic t_i there is an anchor word that has a nonzero probability only in the topic t_i. If one has the *anchor words* one can recover a topic model without EM algorithm.

The Algorithm 1 consists of two steps: the search of anchor words and recovery of a topic model with anchor words. Both procedures use the matrix $Q_{W \times W}$ that contains joint probabilities of co-occurrence of word pairs $p(w_i, w_j)$, $\sum Q_{ij} = 1$. Let us denote row-normalized matrix Q as $\hat{Q}$, the matrix $\hat{Q}$ can be interpreted as $\hat{Q}_{i,j} = p(w_j|w_i)$. It should be note that Algorithm 1 does not need to keep Q matrix in memory, it can be possessed by blocks.

Algorithm 1. High Level Anchor Words

```
**Input**: collection D, number of topics |T|
**Output**: matrix $\Phi$;
$Q$ = Rows_normalized Word Co-occurences(D)
$\hat{Q}$ = $Random\_projection(\hat{Q})$
$S$ = FindAnchorWords($\hat{Q}$, |T|)
$\Phi$ = RecoverWordTopic($\hat{Q}$, $S$)
**return** $\Phi$
```

Let us denote indexes of anchor words $S = \{s_1, \ldots, s_T\}$. The rows indexed by elements of S are special in that every other row of $\hat{Q}$ lies in the convex hull of the rows indexed by the anchor words [1]. At the next step optimization problems are solved. It's done to recover the expansion coefficients of $C_{it} = p(t|w_i)$, and then using the Bayes rule we restore matrix $(p(w|t))_{W \times T}$. The search of anchor words is equal to the search for almost convex hull in the vectors of the matrix $\hat{Q}$ [1]. The combinatorial algorithm that solves the problem of finding the anchor words is given in Algorithm 2.

Algorithm 2. The combinatorial algorithm FastAnchorWords

```
**Input**: dots $V = v_1, \ldots, v_n$, dim of convex hull $K$, parameter of error $\epsilon$;
**Output**: $\{v'_1, \ldots, v'_k\}$ – set of points which constitute the convex hull;
put $v_i$ into random subspace $V$, $dimV = 4logV/\epsilon^2$
S = $\{s_0\}$, $s_0$ – point that has the largest distance to origin.
**for all** i **do**=1 to K-1:
    denote point $\in$ V that has the largest distance to $span(S)$ as $s_i$
    $S = S \cup \{s_i\}$
**for all** i **do**=1 to K-1:
    denote point $\in$ V that has the largest distance to $span(S \setminus \{s_i\})$ as $s'_i$
    update $s_i$ on $s'_i$
**return** S
```

2.3 Integration of N-Grams into Topic Models

The above-discussed topic models are based on single words (unigrams). Sometimes collocations can more exactly define a topic than individual words, therefore various approaches have been proposed to take into account word combinations while building topic models.

Bigram Topic Model proposed by Hanna Wallach in [15]. This model involves the introduction of the concept a hierarchical language model Dirichlet [9]. It is assumed that the appearance of a word depends on the topic and the previous word, all word pairs are collocations.

LDA Collocation Model proposed by M. Steyvers in [13]. The model introduces a new type of hidden variables x ($x = 1$, if $w_{i-1}w_i$ is collocation 0 else). This model can take into account the bigrams and unigrams, unlike the bigram topic model, where each pair of words are collocations.

N-Gram Topic Model proposed by Xuerui Wang in [16]. This model adds the relation between topics and indicators of bigrams that allows us to understand the context depending on the value of the indicator [16].

PLSA-SIM proposed by Michail Nokel in [12]. The algorithm takes into account the relation between single words and bigrams (PLSA-SIM). Words and bigrams are considered as similar if they have the same component word. Before the start of the algorithm, sets of similar words and collocations are precalculated. The original algorithm PLSA is modified to increase the weight of similar words and phrases in case of their co-occurrence in the documents of the collection.

3 Methods to Estimate the Quality of Topic Models

To estimate the quality of topic models, several metrics were proposed.

Perplexity is a measure of inconsistency of a model towards the collection of documents. The perplexity is defined as:

$$P(D, \varPhi, \varTheta) = exp\left(-\frac{1}{len(\mathrm{D})}\ log\ L(D, \varPhi, \varTheta)\right) \tag{8}$$

Low perplexity means that the model correctly predicts the appearance of terms in the collection. The perplexity depends on the size of a vocabulary: usually perplexity grows with increase of the vocabulary.

Coherence is an automatic metric of interpretability proposed by David Newman in [11]. It was showed that the coherence measure has the high correlation with the expert estimates of topics interpretability.

$$PMI(w_i, w_j) = log \frac{p(w_1, w_2)}{p(w_1)p(w_2)} \tag{9}$$

The coherence of a topic is the median PMI of word pairs representing the topic, usually it is calculated for n most probable elements in the topic. The coherence of the model is the median of the topics coherence.

A Measure of the Kernels Uniqueness. Human-constructed topics usually have unique kernels, that is words having high probabilities in the topic. The measure of kernel uniqueness shows to what extent topics are different to each other.

$$U(\Phi) = \frac{| \cup_t kernel(\Phi_t)|}{\sum_{t \in T} |kernal(\Phi_t)|} \tag{10}$$

If the uniqueness of the topic kernels is closer to one then we can easily distinguish topics from each other. If it is closer to zero then many topics are similar to each other, contain the same words in their kernels. In this paper the kernel of a topic means the ten most probable words in the topic.

4 Bigram Anchor Words Topic Modeling

The bag of words text representation does not take into account the order of words in documents, but, in fact, many words are used in phrases, which can form completely different topics.

Usually adding collocations as unique elements of the vocabulary significantly impairs the perplexity by increasing the size of the vocabulary, but the topic model interpretability is increased. The question arises: if it is possible to consider collocations in the Anchor Words algorithm without adding them to the vocabulary.

4.1 Extracting Collocations (Bigrams)

To extract collocations, we used the method proposed in [5]. The authors propose the following algorithm. If several words in a text mean the same entity then in this text these words should appear beside each other more often than separately. It was assumed that if a pair of words co-occurs as immediate neighbors more than half of their appearances in the same text box, it indicates that this pair of words is a collocation. For further use in topic models, we will use 1000 most frequent bigrams extracted from the source text collection.

4.2 Representation of Collocations in Anchor Words Model

One of the known problems in statistical topic modeling is the high fraction of repeated words in different topics. If one wants to describe topics in a collection only with unigrams there are many degrees of freedom to determine the topics. Multiword expressions such as bigrams can facilitate more diverse description of extracted topics. Typically, the addition of bigrams as unique elements of a vocabulary increases the number of model parameters and degrades the perplexity. Further in the article, we put forward the modification of Anchor Words algorithm that can use the unigrams and bigrams as anchor words and improve the perplexity of the source Anchor topic model.

In the step 3 of the Algorithm 1, each word w_i is mapped to vector $\hat{Q}_i$. The problem of finding the anchor words is the allocation of the "almost convex hull" [1] in the vectors $\hat{Q}_i$. Each topic has a single anchor word with corresponding vector from the set of $\hat{Q}_i$.

The space, which contains the vector $\hat{Q}_i$, has a thematic semantics, therefore each word may become an anchor, and thus may correspond to a some topic. To search anchor words means to find vectors corresponding to the basic hidden topics, so that the remaining topics are linear combination of basic topics.

Our main assumption is that in the space of word candidates onto anchor words positions ($\hat{Q}$), bigrams $\boldsymbol{w_i w_j}$, are presented as a sum of vectors $\boldsymbol{w}_i + \boldsymbol{w}_j$. We prepare a set of bigrams and add vectors according to this bigrams in a set of anchor word candidates. It should be noted that, after that modification, bigrams can be anchors words but are not introduced as elements of topics.

The search of the anchor words happens directly using the distance of each word on the current convex hull (Algorithm 2). Bigrams that have on their composition two vectors close to the borders of current convex hull are given the priority in the process of selection of anchor words. It is caused by the increase of the norm of the resultant vector in the direction of convex hull expansion. Therefore, while searching anchor words, we take into account bigrams and increase the probability of choosing a bigram as an anchor word that can be interpreted as a regularization.

The expansion of the convex hull helps to describe more words through the fixed basis. It is important to note that unreasonable extension of the convex hull can break the good properties of the model, such as interpretability. An algorithm for constructing the bigram anchor words model is shown at Algorithm 3. It differs from the original algorithm only in lines 4 and 5.

4.3 Experiments

The experiments were carried out on three collections:

1. ***Banks Articles*** – a collection of Russian banking articles, 2500 documents (2000 for the train, 500 for the control), 18378 words.
2. ***20 Newsgroups*** – a collection of short news stories, 18846 documents (11314 for the train, 7532 for the control), 19570 words http://goo.gl/6js4G5.

Algorithm 3. High Level Bigram Anchor Words

```
Input: collection D, number of topics |T|, set of bigrams C
Output: matrix $\Phi$;
$Q$ = Word Co-occurences(D)
$\hat{Q} = Rows\_normalized(Q)$
$\hat{Q} = Random\_projection(\hat{Q})$
$\hat{B} = \hat{Q}_1, \dots, \hat{Q}_n, \hat{Q}_{C_{11}} + \hat{Q}_{C_{12}}, \dots, \hat{Q}_{C_{n1}} + \hat{Q}_{C_{n2}}$
$S$ = FindAnchorWords($\hat{B}$, |T|)
$\Phi$ = RecoverWordTopic($\hat{Q}$, $S$)
return $\Phi$
```

3. ***NIPS*** – a collection of abstracts from the Conference on Neural Information Processing Systems (NIPS), 1738 documents (1242 for the train, 496 for the control), 21358 words https://goo.gl/EaGmT0.

All collections have been preprocessed. The characters were brought to lowercase, characters which do not belong to Cyrillic and Latin alphabet were removed, words have been normalized (or stemmed for English collections), stop words and words with the length less than four letters were removed. Also words occurring less than 5 times were rejected. Collocations have been extracted with the algorithm described in Sect. 4.1. The preprocessed collections are available on the page, github.com/ars-ashuha/tmtk. In all experiments, the number of topics was fixed $|T| = 100$.

The metrics were calculated as follows:

- To calculate perplexity, the collection was divided into train and control parts. When calculating perplexity on test samples, each document was subdivided into two parts. In the first part, the vector of topics for the document was estimated, on the second part, perplexity was calculated.
- When calculating the coherence, the conditional probabilities are calculated with a window of 10 words.
- When calculating the unique kernel, ten most probable words in a topic were considered as its kernel.

The experiments were performed on the following models: PLSA (PL), Anchor Words (`AW`), Bigram Anchor Words (`BiAW`), Anchor Words and PLSA combination (`AW + PL`), Bigram Anchor Words and PLSA combination (`BiAW + PL`). The combination was constructed as follows: the topics obtained by the Anchor Word or Bigram Anchor Word algorithm, were used as an initial approximation for PLSA algorithm. In experiments, perplexity was measured on the control sample (P_{test}), coherence is denoted as (PMI), the uniqueness of the kernel is denoted as (U). The results are shown in Table 1.

As in the experiments of the authors of the Anchor Words model, the perplexity grows (in two collections out of three), which is a negative phenomenon, but the uniqueness and interpretability of the topics also grows. The combination of *Anchor Words* and *PLSA* models shows the results better than *Anchor Words* or *PLSA* separately.

Table 1. Results of Numerical experiments

Collection	*Banks articles*			*20 Newsgroups*			*NIPS*		
Metric	P_{test}	*PMI*	*U*	P_{test}	*PMI*	*U*	P_{test}	*PMI*	*U*
`PL`	2116	0.60	0.40	2155	0.31	0.40	1635	0.21	0.32
`AW`	2330	0.63	0.53	2268	0.38	0.41	1505	0.41	0.38
`BiAW`	2248	0.79	0.60	2183	0.68	0.54	1500	0.50	0.41
`AW+PL`	2052	0.78	0.58	2053	0.54	0.55	1434	0.52	0.46
`BiAW+PL`	**1848**	**0.87**	**0.63**	**2027**	**0.78**	**0.64**	**1413**	**0.58**	**0.49**

The Bigram Anchor Words model shows the results better than the original *Anchor Words*: has lower perplexity, greater interpretability and uniqueness of kernels, but is still inferior to the *PLSA* model in perplexity. The combination of *Bigram Anchor Words* and *PLSA* models shows better results than other models; this combination has higher interpretability and uniqueness of the kernels.

It can be concluded that the initial approximation, given by the *Bigram Anchor Words* model, is more optimal in terms of achieving final perplexity and other metrics of quality. This approximation improves the sensitivity of *PLSA* to the initial approximation, which, in turn, can be formed taking into account the linguistic knowledge. Tables 2 and 3 contain examples of topics for Bank and NIPS collections. Note that main achievement is that our approach allows us to use bigrams as anchors. Also we present the tables with topic examples to show that our model is not similar to others.

Table 2. Examples of topics for the Russian Bank collection

PLSA		**ANW**		**ANW+ PLSA + BI**	
Topic 1	***Topic 2***	***Topic 1***	***Topic 2***	***Topic 1***	***Topic 2***
рынок	кредит	акция	кредит	рынок	кредит
российский	кредитный	рынок	кредитован	инвестор	замщик
инвестор	замщик	размещение	потребитель	акция	ипотечный
фондовый	кредитован	акционер	ипотечный	фондовый	кредитован
облигация	ставка	инвестор	замщик	инструмент	залог
бумага	банка	капитал	банк	биржа	портфель
инструмент	процентный	фондовый	население	облигация	задолжен
фонд	срок	биржа	клиент	сегмент	потребитель

Anchor words for unigram anchor model: *москва* (moscow), *налоговый* (tax), *история* (history), *акция* (share), *сила* (power), *платеж* (payment)

Anchor words for bigram anchor models: *компания* (company), ***миллионрубль*** (million rubles), ***страна ес*** (eu country), *управление* (control), ***юридическийлицо*** (company), ***российский федерация*** (russian federation)

Table 3. Examples of topics for the NIPS collection

PLSA		ANW		ANW+ PLSA + BI	
Topic 1	*Topic 2*	*Topic 1*	*Topic 2*	*Topic 1*	*Topic 2*
Neuron	Tree	Neuron	Tree	Synaps	Tree
Spike	Featur	Synaps	Decis	Synapt	Decis
Fire	Branch	Synapt	Branch	Neuron	Branch
Time	Thi	Input	Structur	Hebbian	Set
Synapt	Class	Pattern	Leaf	Postsynapt	Probabl
Synaps	Imag	Neural	Prune	Pattern	Prune
Rate	Object	Activ	Set	Function	Algorithm
Input	Decis	Connect	Probabl	Activ	Leaf

Anchor words for unigram anchor model: *face*, *charact*, *fire*, *loss*, *motion*, *cluster*, *tree*, *circuit*, *trajectori*, *word*, *extra*, *action*, *mixtur*

Anchor words for bigram anchor model: *likelihood*, *network*, *loss*, *face*, ***ocular domain***, ***reinforc learn***, ***optic flow***, ***boltzmann machin***, *markov*

5 Conclusion

We propose a modification of the Anchor Words topic modeling algorithm that takes into account collocations. The experiments have confirmed that this approach leads to the increase of the interpretability without deteriorating perplexity.

Accounting of collocations is only the first step to add linguistic information into a topic model. Further work will focus on the study of the possibilities of using the sentence structure of a text, as well as the morphological structure of words in the construction of topic models.

Acknowledgments. This work was supported by grant RFFI 14-07-00383A «Research of methods of integration of linguistic knowledge into statistical topic models».

References

1. Arora, S., Ge, R., Halpern, Y., Mimno, D., Moitra, A., Sontag, D., Wu, Y., Zhu, M.: A practical algorithm for topic modeling with provable guarantees. arXiv preprint arXiv:1212.4777 (2012)
2. Arora, S., Ge, R., Moitra, A.: Learning topic models - going beyond SVD. In: 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science (FOCS), pp. 1–10. IEEE (2012)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Machine Learn. Res. **3**, 993–1022 (2003)

4. Cheng, X., Yan, X., Lan, Y., Guo, J.: BTM: topic modeling over short texts. IEEE Trans. Knowl. Data Eng. **26**(12), 2928–2941 (2014)
5. Dobrov, B., Lukashevich, N., Siromytnikov, S.: Forming the base of terminological phrases in the texts of the subject area, pp. 201–210 (2003)
6. Gao, W., Li, P., Darwish, K.: Joint topic modeling for event summarization across news and social media streams. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1173–1182. ACM (2012)
7. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM (1999)
8. Krestel, R., Fankhauser, P., Nejdl, W.: Latent dirichlet allocation for tag recommendation. In: Proceedings of the third ACM Conference on Recommender Systems, pp. 61–68. ACM (2009)
9. MacKay, D.J., Peto, L.C.B.: A hierarchical dirichlet language model. Nat. Lang. Eng. **1**(03), 289–308 (1995)
10. Mei, Q., Cai, D., Zhang, D., Zhai, C.: Topic modeling with network regularization. In: Proceedings of the 17th International Conference on World Wide Web, pp. 101–110. ACM (2008)
11. Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 100–108. Association for Computational Linguistics (2010)
12. Nokel, M., Loukachevitch, N.: The method of accounting bigram structure in topical models. Computational Methods and Programming (2015)
13. Steyvers, M., Griffiths, T.: Matlab topic modeling toolbox 1.3 (2005)
14. Vorontsov, K.: Additive regularization for topic models of text collections. In: Doklady Mathematics, pp. 301–304. Pleiades Publishing (2014)
15. Wallach, H.M.: Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 977–984. ACM (2006)
16. Wang, X., McCallum, A., Wei, X.: Topical n-grams: phrase and topic discovery, with an application to information retrieval. In: Seventh IEEE International Conference on Data Mining, ICDM 2007, pp. 697–702. IEEE (2007)