Paper Discovery Via Topic Modelling

Usman Anwar
usman.anwar@itu.edu.pk
Information Technology University, Lahore, Pakistan.

June 2, 2021

Abstract

Scientific literature is growing at an alarming rate. Approximately 2.5 Million papers are published every year and the publication rate is still growing at a significant rate. The situation is particularly acute for popular fields like machine learning and computer vision where swathes of research manuscripts are released almost everyday on arxiv and popular venues in the field cumulatively accept tens of thousands of papers a year. However, importantly, for any given researcher, only a tiny amount of these papers are relevant to his/her research. Unfortunately, currently, there is no automated way to identify the relevant papers and a research may have to manually browse through the complete collection of papers to identify papers of his/her interests. In this project, we develop a topic model on a corpus of NIPS papers from 1987 to 2017 and show that our deep learning based model is able to learn coherent and diverse topics. Our model learns 10 distinct topics with mean topic coherence value of 0.195 and mean topic diversity value of is 0.87. We also demonstrate that a simple topic model such as lda2vec fails on this task and provide a theoretical explanation for that. We also outline utilization of our model for information retrieval task and also point out future directions such as learning hierarchical topic embeddings among other things.

1 Introduction

At the turn of 18th century, invention of commercial printing machines greatly improved the way humanity circulated and preserved information. While it was clearly a boon for medieval literature writers, an unexpected beneficiary was the scientific community. Up till that time, scientific works were typically passed from a generation to the next one by word of mouth and sometimes in the form of few handwritten manuscripts. Invention of printing press enabled scientific community to create scientific journals, periodicals that featured latest research work from scientists and researchers. This helped accelerate the dissemination of scientific research and ultimately enhanced both the pace and quality of research.

Scientific literature had been consistently growing for the past century and according to an estimate, more than 50 Million research papers have been published so far and approximately 2.5 Million new research papers are published every year. Further, there is high diversity in publication venues for these papers which makes it difficult for a given researcher to consume research publications relevant to his or her work. An interesting result of this high publication rate is that as fields have become segregated, many interesting results that were discovered in field A but are of interest to field B never reach a critical level of audience in field B and this ultimately adversely affect the quality and pace of research. Many other areas, such as movies or songs, have experienced similar level of explosive growth in content but availability of high quality

recommendation engines has prevented the consumers of such content from becoming overwhelmed with the wealth of content. Unfortunately, there is no such streamlined process for consumption of scientific manuscripts. This is, in part, due to the fact that developing a recommendation engine for scientific publications continues to be a challenging problem. Here, we try to outline some of these challenges.

Collaborative filtering based techniques have been hugely successful in areas like movie recommendation, song recommendation and news articles recommendation. These techniques are rooted in big data analytics of user behaviours. Scientific publications, however, have a considerable small audience, and do not provide excess of data required by big data methods. Further, a scientific publication is a complex document and may be 'interesting' to different people for different reasons. Lastly, not just high precision, but high recall is equally desirable in a scientific paper recommendation engine. Collaborative filtering methods, while generally very precise in their recommendations, do not guarantee full recall.

In light of above challenges, we argue that any practical recommendation engine for scientific publications must be rooted in understanding of publications themselves. To that end, we demonstrate the effectiveness of Embedding Topic Model [8] in capturing topics present in a paper in an unsupervised way on a collection of NIPS papers from 1987 to 2017.

2 Related Works

Finding new papers relevant to ones interest from a large collection of papers is a laboring task for a researcher. This labour can be considerably reduced if information about the papers is succinctly represented in some form which can help the researcher remove a large chunk of papers outside of his/her area of interests. [11] attempts to find scientific topics in a model using Latent Dirichlet Allocation (LDA). On similar lines, [3] attempts to develop a system for exploratory literature review. Another important work in this regard is [22] which combines collaborative filtering with topic modelling for paper recommendations.

Topic Modelling: Topic modelling is intrinsically an approach to understand the latent structure of the documents. The most prominent work in this regard is [5] by Blei et al. It posits topics as a distribution over words, and documents as distribution over topics and uses Dirichlet distribution as a prior for distribution of topics. Despite the popularity of LDA model, there does not seem to be a straightforwardly simple way to do inference using LDA model. While Gibbs sampling [11] can infer accurately, it is computationally very slow and while variational inference can be fast, it is not guaranteed to be accurate. Further, there has been some criticism on the interpretability of topics discovered by LDA and interpretation of these topics has been linked to reading tea leaves [7]. This criticism has resulted in focus on alternative approaches. Most notable among them is anchor word hypothesis which assumes that there is some key word, called anchor word, for every topic that occurs in a document only and only if that document belongs to that topic. An implication of this statement is that an anchor word has zero probability of occurring in a document that does not contain its corresponding topic. Anchor word approach is appealing due to the fast inference, however, its performance on real world data is non-satisfactory [2]. A related approach, demonstrated by [23], [24], [25] is to exploit the convex geometry of the topic simplex induced by the Dirichlet prior. Vertices of topic simplex correspond to the topics and points inside the simplex corresponds to the documents that may be generated. Finally, [10] based on a Bayesian non-parametric approach learns a hierarchical topic distributions. Despite the natural appeal of hierarchical topic modelling, [10] is computationally slow and does not

remain practical for large collection of documents.

Neural Embeddings: Classical way of encoding documents and any text in general is to use one-hot scheme: a vector of the size of the vocabulary is defined where each index in the vector corresponds to the term frequency of that particular word. This scheme, despite its simplicity, has high memory cost and is not optimal for large vocabularies. On the other hand, language, and consequently, words, have a natural structure. The idea behind neural embeddings is to exploit this natural structure to learn a compact and compressed representation of the words. First notable work on learning neural embeddings is [4] by Bengio et al., however, neural embeddings were really popularized by works of Tomas Mikolov [17], [15] which inspired numerous other notable works such as [21], [16], [9] and [6].

The idea of learning neural embeddings of words has since been extended to learning neural embeddings of sentences, paragraphs and documents [13]. An interesting new line of work is to learn word embeddings on special manifolds such as Poincare ball [19] or spherical ball [14].

The idea of learning neural embeddings has also been used for topic modelling [20] and is also considered in considerable detail in this manuscript.

3 Dataset And Methods

We train our model on a dataset of Neural Information Processing Systems (NeurIPS) papers from year 1987 to 2017 [12]. We split our complete dataset into train, test and validation datasets with train set consisting of 85% of the documents in the original dataset, validation set consisting of 5% of and test set consisting of 10% of original papers.

We also created a custom dataset by scrapping papers from online repositories of International Conference Of Machine Learning (ICML), International Conference on Learning Representations (ICLR) and Conference On Vision and Pattern Recognition (CVPR). However, due to scarcity of time, we were not able to include this dataset in our testing.

3.1 Preprocessing

For each paper, we remove author names and list of references from the paper text. Further, we remove any stop words, punctuation, emails, numbers and any tokens which contain any mathematical symbols.

Further, we remove any tokens with document frequency greater than 70% or which occur in less than 20 papers. Note that both of these are hyper-parameters, and were chosen heuristically.

3.2 Deep Learning Models

We employ two deep learning techniques for this task as it is known that LDA and other classical techniques fail to work for large corpus and large vocabularies and provide sub-optimal performance. [8] Details of these models are given below.

3.2.1 Lda2vec

In LDA and other traditional topic models, words, or tokens, are still represented as one hot vectors, This makes it difficult to utilize large vocabularies and results in sub optimal models. To enable models to utilize large vocabularies, [15] popularized the idea

of using word embeddings which are dense representation of words in a low dimensional space to represent words in a document.

The main idea behind lda2vec [18] is to jointly learn word embeddings and topic embeddings by conditioning the word embeddings on the document embedding. To better understand lda2vec, remember that in negative sampling skip gram model of word2vec, our objective is to maximize the probability of predicting a surrounding word (called target word) w_j and minimize the probability of predicting non related random words w_l given a context word c_j.

Mathematically, this can be expressed in the form of following loss function

$$\mathcal{L}_{neg,w2v} = \log \sigma(\vec{c}_j \cdot \vec{w}_j) + \sum_{l=0}^{n} \log \sigma(-\vec{c}_j \cdot \vec{w}_l)$$

lda2vec simply modifies the context vector c_j to $c_j + d_m$ where d_m is the document embedding of document m containing the context vector c_j and target word w_j.

This gives us the following modified negative sampling loss for lda2vec

$$\mathcal{L}_{neg} = \log \sigma((\vec{c}_j + \vec{d}_m) \cdot \vec{w}_j) + \sum_{l=0}^{n} \log \sigma(-(\vec{c}_j + \vec{d}_m) \cdot \vec{w}_l)$$

Further, to extract topics, lda2vec proposes to project document embeddings into a K dimensional space, where K is the number of topics, as follows:

$$\vec{d}_m = p_{m1} \cdot \vec{t}_1 + p_{m2} \cdot \vec{t}_2 + ... + p_{mk} \cdot \vec{t}_k$$

where t_i is the topic embedding of the topic i in the same vector space as word embeddings and $p_m i$ is the corresponding proportion of the topic i present in the document m. To aid interpretability, a constraint is imposed that all topic proportions should sum to one i.e. $\sum_i p_{mi} = 1$ and to sparsify the word embeddings, we maximize the Dirichlet likelihood for topic proportions for all documents.

$$\mathcal{L}_{\mathcal{D}} = \lambda \sum_{mi} (\alpha - 1) \log p_{mi}$$

The final loss function is the sum of modified negative sampling loss and Dirichlet likelihood loss.

$$\mathcal{L} = \mathcal{L}_{neg} + \mathcal{L}_{\mathcal{D}}$$

3.2.2 Embedding Topic Model

Embedding Topic Model (ETM) contains two latent space: first L-dimensional latent space contains vocabulary tokens i.e. words and the topic embeddings. And the other K-dimensional latent space contains documents which are represented in the form of weighed linear combination of K topics.

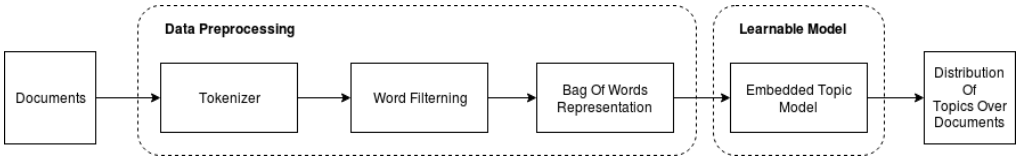


Figure 1: Data Science Pipeline For Embedding Topic Model

In LDA, distribution of topics over words is a full distribution over all the tokens in the vocabulary. However, in ETM, topic embedding is a vector $\alpha_k \in \mathbb{R}^L$ i.e. space of word embeddings, so, probabilistically, it is a distribution over the set of basis vectors of all the word embeddings rather than the word embedding themselves directly. This enables ETM to accommodate new words which it may have not seen during training but it has access to the embedding of at the test time.

The parameters of ETM are word embeddings ρ and topic embeddings α. Precisely, ETM defines the following generative process for a document:

- Draw topic proportions $\theta_d \sim \mathcal{LN}(0, I)$.
- For each word n in the document:
  - Draw topic assignment $z_{dn} \sim \text{Cat}(\theta_d)$
  - Draw the word $w_{dn} \sim \text{softmax}(\rho^T \alpha_{z_{dn}})$

where $\mathcal{LN}$ is the logistic normal distribution which transforms a standard normal variable to the simplex and Cat is the categorical distribution. In order to train a deep learning model, we typically try to maximize the log probability of data which is documents in our case. However, in ETM this requires us to compute an integral over all the topic proportions, which, is intractable. This intractable integral can be bypassed by using variation inference. To carry out variational inference, we posit Gaussian Distribution as a prior on variational distribution $q(\delta; \vec{w}_d, \nu)$ where δ is the untransformed topic proportion, $\vec{w}_d$ is the bag of words representation of a document and ν is the variational parameter. We then train a neural network to output the parameters of this distribution i.e. mean and variance. Using this family of distributions, we create a lower bound on log likelihood of data and optimize that lower bound.

$$\mathcal{L}(\alpha, \rho, \nu) = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\log p(w_{nd}|\delta, \rho, \alpha)] - \sum_{d=1}^{D} KL(q(\delta_d; \vec{w}_d, \nu)||p(\delta_d)$$

Intuitively, the first term in the loss function forces the model to place mass on topic proportions δ_d that explain the data and the second term forces the model to abide by the prior on variational distribution.

4 Results And Discussion

4.1 lda2vec

lda2vec performed very poorly and mode collapse of topic distribution occurred. This result is collaborated by [1] and various user comments on the original repository of lda2vec.

Topic Coherence	0.195
Topic Diversity	0.87

Table 1: Topic coherence evaluates how internally coherent a topic is and topic diveristy measures how different two topics are on the average.

4.2 Embedding Topic Model

ETM works very well and we are able to attain meaningful topics. Even though it attains low score on traditional topic modelling metrics i.e. topic coherence and topic diversity (shown in table 1), manual examination of topics and word embeddings shows that it captures different themes present in the collection well as shown in tables 2 and 3.

Figure 2 shows evolution of some parameters as training of the model converges. Note that validation perplexity does not correlate with topic coherence and topic diversity. Figure 4 shows topic distribution discovered by the model on some sample papers.

Topic Index	Human Label	Top 9 Words In Topic
1	Dimensionality Reduction	matrix, sparse, kernel, rank,norm, solution,component, column, dimensional
2	Optimization Algorithms	bound, theorem, bind, convex, loss, let, optimization, gradient, convergence
3	Bayesian Methods	sample, estimate, gaussian, prior, likelihood, process, log, posterior, bayesian
4	Brain Related	neuron, spike, cell, stimulus, response, input, activity, signal, system
5	Graph Theory	graph, node, variable, tree, cluster, edge, structure, network, inference
6	Image Classification	label, feature, classification, kernel, training, class, classifier, test, dataset
7	NLP Papers	word, topic, feature, user, document, task, human, sequence, learning
8	Neural Networks	network, input, unit, weight, output, training, system, layer, hide
9	Deep Neural Networks	image, feature, object, network, deep, layer, deep, train, training, representation
10	Reinforcement Learning	policy, action, reward, agent, optimal, game, control, regret, reinforcement

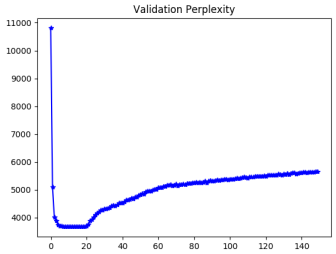
Table 2: Topics learned by the ETM model.

Why lda2vec does not work?

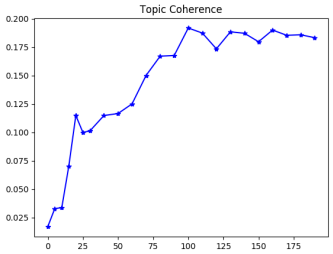
As reported in the section 4, lda2vec performs poorly and suffers from mode collapse of topic distribution i.e. model only generates only one or two unique topics. This is the consequence of the fact that during training lda2vec only loosely conditions the output of context word on document vector. Notice that forward pass for ordinary word2vec

Seed Word	Nearest Neighbours
brain	activity, device, sound, neuron, coding, stimulus
manifold	laplacian, subspace, pca, spherical, geodesic
reinforcement	reward, planning, agent, policy, arm, help
theorem	proof, proposition, guarantee, lemma, satisfiability
transformation	transform, normalize, dimension, multi, invariant
probabalistic	hierarchical, joint, dependency, count, treat, intelligence

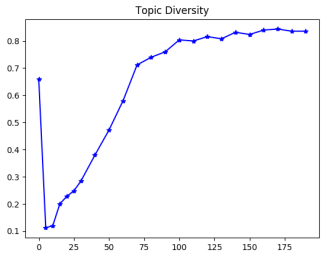
Table 3: Nearest neighbours of seed words in embedding space



(a) Validation Perplexity

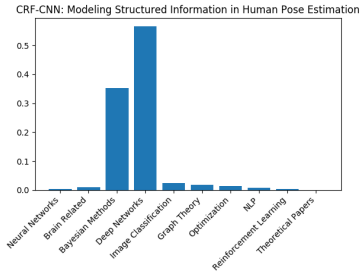


(b) Topic Coherence

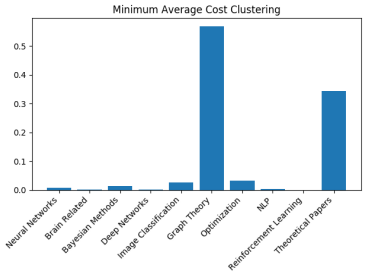


(c) Topic Diversity

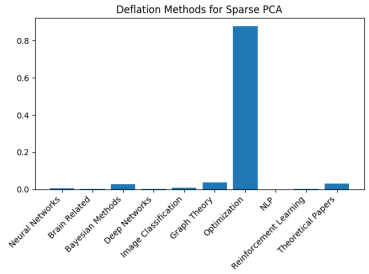
Figure 2: Evolution of some monitoring parameters as training is continued for the model. Note that while validation perplexity quickly converges, the quality of topics at that point is very poor. This points to the fact that perplexity is a poor measure for judging the quality of a topic model.



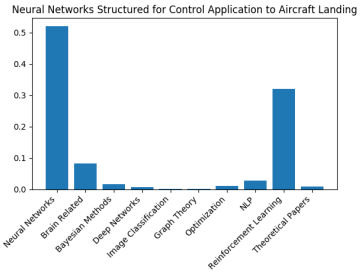
(a) Model has correctly identified that this paper is about deep networks as well as Bayesian Methods.



(b) This paper contains both a theoretical component and an application of graph theory.

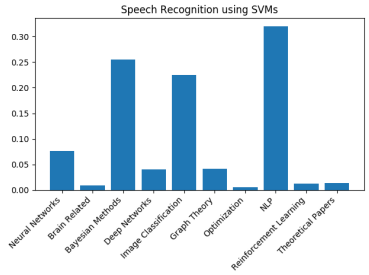


(c) This paper develops an optimization process for PCA

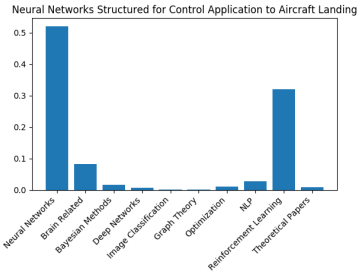


(d) This paper combines neural networks with reinforcement learning.

Figure 3: Topic distribution produced by the model on some papers present in the collection.



(a) Due to lack of papers related to speech in NIPS, model fails to capture the 'speech' part of paper at all and tries to adjust that in some other categories.



(b) Note that model has assigned non-trivial contribution to the 'brain related' even though paper is not about the brain at all. This is due to the frequent occurrence of control related words in the brain related papers as well in the context of 'control groups' for neurological and physiological studies..

Figure 4: Some failure cases for the model.

and lda2vec only differ slightly:

$$p_{word2vec} = \mathcal{C}(E(c))$$
$$p_{lda2vec} = \mathcal{C}(E(c) + d)$$

where E is the embedding matrix and $\mathcal{C}$ is the context matrix, c is the context word and p is the pivot word and d is the document embedding.

Conditioning by modifying the context embedding to be a linear combination of word embedding of context word and document embedding is quite weak, so, model quickly learns to represent d as a constant embedding for all documents and hence avoids learning any topics at all whatsoever.

5 Conclusion

We provide a model that can capture the topics present in the collection of scientific papers. Once a model has been learnt, it can be used to recommend papers based on topic interests of researchers and can be improved by adding meta deta of papers as well.

Current work, despite its promise, has several limitations. First it only outputs topics as a list of words and it is up to the human to interpret this word list. While this is generally not bothersome for a small number of topics, it becomes cumbersome for a large number of topics. Ideally, the model itself should be able to identify a key word which may serve as a label for the topic. Further, ETM learns embeddings in an Euclidean space, however, [19] and [14] have shown that Euclidean space, due to its flat curvature, is sub optimal for learning embeddings.

References

[1] Dan Antoshchenko. lda2vec-pytorch. `https://github.com/TropComplique/lda2vec-pytorch`, 2017.

[2] Sanjeev Arora, Rong Ge, Frederic Koehler, Tengyu Ma, and Ankur Moitra. Provable algorithms for inference in topic models. In *International Conference on Machine Learning*, pages 2859–2867, 2016.

[3] Claus Boye Asmussen and Charles Møller. Smart literature review: a practical topic modelling approach to exploratory literature review. *Journal of Big Data*, 6(1):93, 2019.

[4] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[5] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[7] Jonathan Chang, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber, and David M Blei. Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*, pages 288–296, 2009.

[8] Adji B Dieng, Francisco JR Ruiz, and David M Blei. Topic modeling in embedding spaces. *arXiv preprint arXiv:1907.04907*, 2019.

[9] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. *arXiv preprint arXiv:1802.06893*, 2018.

[10] Thomas L Griffiths, Michael I Jordan, Joshua B Tenenbaum, and David M Blei. Hierarchical topic models and the nested chinese restaurant process. In *Advances in neural information processing systems*, pages 17–24, 2004.

[11] Thomas L Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl 1):5228–5235, 2004.

[12] Ben Hamner. Nips papers: Titles, authors, abstracts, and extracted text for all nips papers (1987-2017). 2017. `https://www.kaggle.com/benhamner/nips-papers`. Online; accessed June 2, 2021.

[13] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1188–1196, Bejing, China, 22–24 Jun 2014. PMLR.

[14] Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance Kaplan, and Jiawei Han. Spherical text embedding. In *Advances in Neural Information Processing Systems*, pages 8206–8215, 2019.

[15] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[16] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*, 2017.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013.

[18] Christopher E Moody. Mixing dirichlet topic models and word embeddings to make lda2vec. *arXiv preprint arXiv:1605.02019*, 2016.

[19] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in neural information processing systems*, pages 6338–6347, 2017.

[20] Liqiang Niu and Xin-Yu Dai. Topic2vec: Learning distributed representations of topics. *CoRR*, abs/1506.08422, 2015.

[21] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

[22] Chong Wang and David M Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456. ACM, 2011.

[23] Mikhail Yurochkin, Zhiwei Fan, Aritra Guha, Paraschos Koutris, and XuanLong Nguyen. Scalable inference of topic evolution via models for latent geometric struc-

tures. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5949–5959. Curran Associates, Inc., 2019.

[24] Mikhail Yurochkin, Aritra Guha, and XuanLong Nguyen. Conic scan-and-cover algorithms for nonparametric topic modeling. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3878–3887. Curran Associates, Inc., 2017.

[25] Mikhail Yurochkin and XuanLong Nguyen. Geometric dirichlet means algorithm for topic inference. In *Advances in Neural Information Processing Systems*, pages 2505–2513, 2016.