ARTM vs. LDA: an SVD Extension Case Study

Sergey Nikolenko[1,2,3,4]

[1] Laboratory for Internet Studies, NRU Higher School of Economics
[2] Steklov Institute of Mathematics at St. Petersburg, Russia
[3] Kazan (Volga Region) Federal University, Kazan, Russia
[4] Deloitte Analytics Institute, Moscow, Russia
sergey@logic.pdmi.ras.ru

Abstract. In this work, we compare two extensions of two different topic models for the same problem of recommending full-text items: previously developed SVD-LDA and its counterpart SVD-ARTM based on additive regularization. We show that ARTM naturally leads to the inference algorithm that has to be painstakingly developed for LDA.

1 Introduction

Topic models are an important part of the natural language processing landscape, providing unsupervised ways to quickly evaluate what a whole corpus of texts is about and classify them into well-defined topics. LDA extensions provide ways to augment basic topic models with additional information and retool them to serve other purposes. In a previous work, we have combined the SVD and LDA decompositions into a single unified model that optimizes the joint likelihood function and thus infers topics that are especially useful for improving recommendations. We have provided an inference algorithm based on Gibbs sampling, developing an approximate sampling scheme based on a first order approximation to Gibbs sampling [1]. A recently developed ARTM approach [2–5] extends the basic pLSA model with regularizers and provides a unified way to add new additive regularizers; inference algorithms result with simple differentiation of the regularizers. In this work, we apply ARTM to the problem of adding SVD decompositions to a topic model; we show that one can automatically arrive at an inference algorithm very similar to our previous SVD-LDA approach.

2 LDA and SVD-LDA

The graphical model of LDA [6, 7] is shown on Figure 1a. We assume that a corpus of D documents contains T topics expressed by W different words. Each document $d \in D$ is modeled as a discrete distribution $\theta^{(d)}$ on the set of topics: $p(z_w = j) = \theta^{(d)}$, where z is a discrete variable that defines the topic of each word $w \in d$. Each topic, in turn, corresponds to a multinomial distribution on words: $p(w \mid z_w = k) = \phi_w^{(k)}$. The model also introduces prior Dirichlet distributions with parameters α for the topic vectors θ, $\theta \sim \mathrm{Dir}(\alpha)$, and β for

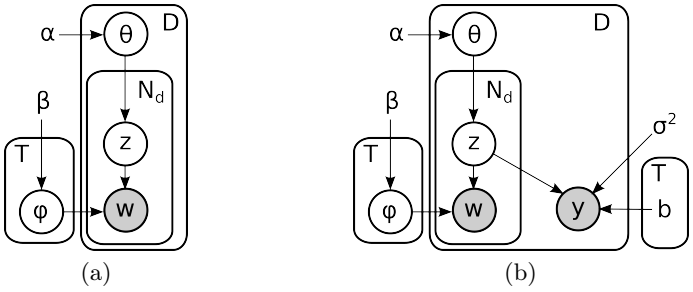

Fig. 1: The (a) LDA and (b) sLDA graphical models.

the word distributions ϕ, $\phi \sim \mathrm{Dir}(\beta)$. A document is generated word by word: for each word, we (1) sample the topic index k from distribution $\theta^{(d)}$; (2) sample the word w from distribution $\phi_w^{(k)}$. Inference in LDA is usually done via either variational approximations or Gibbs sampling; we use the latter since it is easy to generalize to further extensions. In the basic LDA model, Gibbs sampling reduces to the so-called *collapsed Gibbs sampling*, where θ and ϕ variables are integrated out, and z_w are iteratively resampled according to the following distribution: $p(z_w = t \mid \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \propto \frac{n^{(d)}_{-w,t}+\alpha}{\sum_{t' \in T}\left(n^{(d)}_{-w,t'}+\alpha\right)} \frac{n^{(w)}_{-w,t}+\beta}{\sum_{w' \in W}\left(n^{(w')}_{-w,t}+\beta\right)}$, where $n^{(d)}_{-w,t}$ is the number of words in document d chosen with topic t and $n^{(w)}_{-w,t}$ is the number of times word w has been generated from topic t apart from the current value z_w; both counters depend on the other variables $\boldsymbol{z}_{-w}$. Samples are then used to estimate model variables: $\theta_{d,t} = \frac{n^{(d)}_{-w,t}+\alpha}{\sum_{t' \in T}\left(n^{(d)}_{-w,t'}+\alpha\right)}$, $\phi_{w,t} = \frac{n^{(w)}_{-w,t}+\beta}{\sum_{w' \in W}\left(n^{(w')}_{-w,t}+\beta\right)}$, where $\phi_{w,t}$ denotes the probability to draw word w in topic t and $\theta_{d,t}$ is the probability to draw topic t for a word in document d.

The basic LDA model has been used and extended in numerous applications; the relevant class of extensions for us now takes into account additional information that may be available together with the documents and may reveal additional insights into the topical structure. For instance, the *Topics over Time* model and *dynamic topic models* apply when documents have timestamps of their creation (e.g., news articles or blog posts) [8–10], *DiscLDA* assumes that each document is assigned with a categorical label and attempts to utilize LDA for mining topic classes related to classification [11], the *Author-Topic model* incorporates information about the authors of a document [12, 13], and so on.

The SVD-LDA model, presented in [1], can be regarded as an extension of the Supervised LDA (sLDA) model [14]. The sLDA graphical model is shown on Fig. 1b. In sLDA, each document is now augmented with a response variable y drawn from a normal distribution centered around a linear combination of the document's topical distribution ($\bar{\boldsymbol{z}}$, average z variables in this document) with some unknown parameters $\boldsymbol{b}$, a that are also to be trained: $y \sim \mathcal{N}(y \mid \boldsymbol{b}^\top \bar{\boldsymbol{z}} + a, \sigma^2)$.

The original work [14] presents an inference algorithm for sLDA based on variational approximations, but in this work we operate with Gibbs sampling which will be easier to extend to SVD-LDA later. Thus, we show an sLDA Gibbs sampling scheme. It differs from the original LDA in that the model likelihood gets another factor corresponding to the y variable: $p(y_d \mid \boldsymbol{z}, \boldsymbol{b}, \sigma^2) \propto \exp\left(-(y_d - \boldsymbol{b}^\top \bar{\boldsymbol{z}} - a)^2/2\right)$, and the total likelihood is now $p(\boldsymbol{z} \mid \boldsymbol{w}, \boldsymbol{y}, \boldsymbol{b}, \sigma^2) \propto \prod_d \frac{B(\boldsymbol{n}_d+\alpha)}{B(\alpha)} \prod_t \frac{B(\boldsymbol{n}_t+\beta)}{B(\beta)} \prod_d e^{-\left(y_d - \boldsymbol{b}^\top \bar{\boldsymbol{z}}_d - a\right)^2/2}$. On each iteration of the sampling algorithm, we now first sample $\boldsymbol{z}$ for fixed $\boldsymbol{b}$ and then train $\boldsymbol{b}$ for fixed (sampled) $\boldsymbol{z}$. The sampling distributions for each z variable, according to the equation above, are $p(z_w = t \mid \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \propto q(z_w, t, \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) e^{-\frac{1}{2}\left(y_d - \boldsymbol{b}^\top \bar{\boldsymbol{z}} - a\right)^2} = \frac{n^{(d)}_{-w,t}+\alpha}{\sum_{t'}\left(n^{(d)}_{-w,t'}+\alpha\right)} \frac{n^{(w)}_{-w,t}+\beta}{\sum_{w'}\left(n^{(w')}_{-w,t}+\beta\right)} e^{-\frac{1}{2}\left(y_d - \boldsymbol{b}^\top \bar{\boldsymbol{z}} - a\right)^2}$. The latter equation can be either used directly or further transformed by separating $\boldsymbol{z}_{-w}$ explicitly.

SVD-LDA considers a recommender system based on likes and dislikes, so it uses the logistic sigmoid $\sigma(x) = 1/\left(1 + \exp(-x)\right)$ of a linear function to model the probability of a "like": $p(\text{success}_{i,a}) = \sigma\left(\boldsymbol{b}^\top \bar{\boldsymbol{z}} + a\right)$. In this version of sLDA, the graphical model remains the same, only conditional probabilities change. The total likelihood is now $p(\boldsymbol{z} \mid \boldsymbol{w}, \boldsymbol{y}, \boldsymbol{b}, \alpha, \beta, \sigma^2) \propto$

$$\prod_d \frac{B(\boldsymbol{n}_d+\alpha)}{B(\alpha)} \prod_t \frac{B(\boldsymbol{n}_t+\beta)}{B(\beta)} \prod_d \prod_{x \in X_d} \sigma\left(\boldsymbol{b}^\top \bar{\boldsymbol{z}}_d + a\right)^{y_x} \left(1 - \sigma\left(\boldsymbol{b}^\top \bar{\boldsymbol{z}}_d + a\right)\right)^{1-y_x},$$

where X_d is the set of experiments (ratings) for document d, and y_x is the binary result of one such experiment. The sampling procedure also remains the same, except that now we train logistic regression with respect to $\boldsymbol{b}$, a for fixed $\boldsymbol{z}$ instead of linear regression, and the sampling probabilities for each z variable are now $p(z_w = t \mid \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \propto$

$$\begin{aligned} &q(z_w, t, \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \prod_{x \in X_d} \left[\sigma\left(\boldsymbol{b}^\top \bar{\boldsymbol{z}}_d + a\right)\right]^{y_x} \left[1 - \sigma\left(\boldsymbol{b}^\top \bar{\boldsymbol{z}}_d + a\right)\right]^{1-y_x} \\ &\quad = \frac{n^{(d)}_{-w,t} + \alpha}{\sum_{t' \in T}\left(n^{(d)}_{-w,t'} + \alpha\right)} \frac{n^{(w)}_{-w,t} + \beta}{\sum_{w' \in W}\left(n^{(w')}_{-w,t} + \beta\right)} e^{s_d \log p_d + (|X_d| - s_d)\log(1-p_d)}, \end{aligned}$$

where s_d is the number of successful experiments among X_d, and $p_d = \frac{1}{1+e^{-\boldsymbol{b}^\top \bar{\boldsymbol{z}}_d - a}}$.

The SVD-LDA extension has been introduced in [1] as follows: for recommendations we use an SVD model with additional predictors corresponding to how much a certain user or group of user likes the topics trained in the LDA model; since our dataset is binary (like-dislike), we use a logistic version of the SVD model: $p(\text{success}_{i,a}) = \sigma\left(\hat{r}_{i,a}\right) = \sigma\left(\mu + b_i + b_a + q_a^\top p_i + \theta_a^\top l_i\right)$, where p_i may be absent in case of cold start, and l_i may be shared among groups (clusters) of users. The total likelihood of the dataset with ratings comprised of triples $D = \{(i, a, r)\}$ (user i rated item a as $r \in \{-1, 1\}$) is a product of the likelihood of each rating (assuming, as usual, that they are independent): $p(D \mid$

$\mu, b_i, b_a, p_i, q_a, l_i, \theta_a) = \prod_D \sigma\left(\hat{r}_{i,a}\right)^{[r=1]} \left(1 - \sigma\left(\hat{r}_{i,a}\right)\right)^{[r=-1]}$, and the logarithm is $\log p(D \mid \mu, b_i, b_a, p_i, q_a, l_i, \theta_a) = \sum_D \left([r=1] \log \sigma\left(\hat{r}_{i,a}\right) + [r=-1] \log\left(1 - \sigma\left(\hat{r}_{i,a}\right)\right)\right)$, where $[r=-1] = 1$ if $r = -1$ and $[r=-1] = 0$ otherwise, and θ_a is the vector of topics trained for document a in the LDA model, $\theta_a = \frac{1}{N_a} \sum_{w \in a} z_w$, where N_a is the length of document a. Sampling probabilities for each z variable now look like $p(z_w = t \mid \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \propto q(z_w, t, \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) p(D \mid \mu, b_i, b_a, p_i, q_a, l_i, \theta_a^{w \to t}) =$

$$= \frac{n_{-w,t}^{(d)} + \alpha}{\sum_{t' \in T} \left(n_{-w,t'}^{(d)} + \alpha\right)} \frac{n_{-w,t}^{(w)} + \beta}{\sum_{w' \in W} \left(n_{-w,t}^{(w')} + \beta\right)} e^{\sum_D \log\left([r=-1] - \sigma\left(\hat{r}_{i,a}^{\mathrm{SVD}} + l_i^\top \theta_a^{w \to t}\right)\right)},$$

where $\hat{r}_{i,a}^{\mathrm{SVD}} = \mu + b_i + b_a + q_a^\top p_i$, and $\theta_a^{w \to t}$ is the vector of topics for document a where topic t is substituted in place of z_w. We see that in the formula above, to compute the sampling distribution for a single z_w variable one has to take a sum over all ratings all users have provided for this document, and due to the presence of the sigmoid function one cannot cancel out terms and reduce the sum to updating counts. It is possible to store precomputed values of $\hat{r}_{i,a}^{\mathrm{SVD}}$ in memory, but it does not help because the z_w variables change during sampling, and when they do all values of $\sigma(\hat{r}_{i,a}^{\mathrm{SVD}} + l_i^\top \theta_a^{w \to t})$ also have to be recomputed for each rating from the database.

To make the model feasible, a simplified SVD-LDA training algorithm was developed in [1] that could run reasonably fast on large datasets. It used a first order approximation to the log likelihood based on its Taylor series at zero:

$$\frac{\partial \log p(D \mid l_i, \theta_a, \ldots)}{\partial \theta_a} = \sum_D \left([r=1]\left(1 - \sigma(\hat{r}_{i,a}^{\mathrm{SVD}} + \theta_a^\top l_i)\right) l_i \right.$$
$$\left. -[r=-1]\sigma(\hat{r}_{i,a}^{\mathrm{SVD}} + \theta_a^\top l_i) l_i\right) = \sum_D \left([r=1] - \sigma(\hat{r}_{i,a}^{\mathrm{SVD}} + \theta_a^\top l_i)\right) l_i. \quad (1)$$

We denote $s_a = \sum_D \left([r=1] - \sigma\left(\hat{r}_{i,a}^{\mathrm{SVD}} + \theta_a^\top l_i\right)\right) l_i$. We can now precompute s_a (a vector over topics) for each document right after SVD training (with additional memory of the same size as the θ matrix) and use it in LDA sampling:

$$p(z_w = t \mid \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) \propto q(z_w, t, \boldsymbol{z}_{-w}, \boldsymbol{w}, \alpha, \beta) p(D \mid \mu, b_i, b_a, p_i, q_a, l_i, \theta_a^{w \to t})$$
$$\approx \frac{n_{-w,t}^{(d)} + \alpha}{\sum_{t' \in T} \left(n_{-w,t'}^{(d)} + \alpha\right)} \frac{n_{-w,t}^{(w)} + \beta}{\sum_{w' \in W} \left(n_{-w,t}^{(w')} + \beta\right)} e^{s_a \theta_a^{w \to t}},$$

and the latter is proportional to simply $\frac{n_{-w,t}^{(d)} + \alpha}{\sum_{t' \in T} \left(n_{-w,t'}^{(d)} + \alpha\right)} \frac{n_{-w,t}^{(w)} + \beta}{\sum_{w' \in W} \left(n_{-w,t}^{(w')} + \beta\right)} e^{s_t}$ because $s_a \theta_a^{w \to t} = s_a \theta_a - s_w z_w + s_t z_t$, and the first two terms do not depend on t which is being sampled. Thus, the first order approximation yields a simple modification of LDA sampling that incurs relatively small computational overhead as compared to the sampling itself.

Model	Topics	Features	λ	NDCG	AUC	MAP	WTA	Top3	Top5
SVD		5	0.1	0.9814	0.8794	0.9406	0.9440	0.9434	0.9424
SVD		10	0.15	0.9815	0.8801	0.9405	0.9448	0.9434	0.9425
SVD		15	0.2	0.9815	0.8802	0.9405	0.9453	0.9435	0.9426
SVD		20	0.2	0.9816	0.8803	0.9406	0.9453	0.9437	0.9427
SVD-LDA	50	5	0.025	0.9829	0.8893	0.9418	0.9499	0.9466	0.9445
SVD-LDA	100	10	0.025	0.9829	0.8893	0.9418	0.9500	0.9465	0.9445
SVD-LDA	200	15	0.01	0.9830	0.8895	0.9417	0.9524	0.9470	0.9446
SVD-LDA-DEM	50	10	0.01	**0.9840**	0.8901	**0.9428**	**0.9531**	**0.9481**	**0.9456**
SVD-LDA-DEM	100	5	0.01	**0.9840**	**0.8904**	**0.9428**	0.9528	0.9480	**0.9456**
SVD-LDA-DEM	200	10	0.01	**0.9840**	0.8898	**0.9428**	0.9524	**0.9481**	**0.9456**

Table 1: Ranking metrics on the test set. Only the best results w.r.t. λ and the number of features are shown [1].

We have outlined a general approximate sampling scheme; several different variations are possible depending on which predictors are shared in the basic SVD model, $p(\text{success}_{i,a}) = \sigma\left(\hat{r}_{i,a}\right)$. In general, a separate set of l_i features for every user would lead to heavy overfitting, so we used two variations: either share $l_i = l$ among all users or share $l_i = l_c$ among certain clusters of users, preferably inferred from some external information, e.g., demographic features. Both variations can be used for cold start with respect to users. Table 1 summarizes the results of experiments that show that SVD-LDA does indeed improve upon the basic LDA model [1].

3 SVD-ARTM

In recent works [2–4], K. Vorontsov and coauthors demonstrated that if one adds regularizers in the objective function on the training stage of the basic probabilistic Latent Semantic Analysis (pLSA) model, which actually predates LDA [15], one can impose a very wide variety of constraints on the resulting topic model. This approach has been called *Additive Regularization of Topic Models* (ARTM). In particular, the authors showed that one can formulate a regularizer that imposes constraints on the smoothness of topic-document and word-topic distributions that will correspond to the Bayesian approach expressed in LDA (i.e., it will smooth out the distributions).

Formally speaking, for a set of regularizers $R_i(\Phi, \Theta)$, $i = 1..r$, and regularization weights ρ_i, $i = 1..r$, we can extend the objective function to maximize $L(\Phi, \Theta) + R(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in W} n_{dw} \log p(w \mid d) + \sum_{i=1}^{r} \rho_i R_i(\Phi, \Theta)$. By Karush–Kuhn–Tucker conditions, any solution of the resulting problem satisfies the following system of equations:

$$p_{tdw} = \text{norm}^{+}_{t \in T}\left(\phi_{wt}\theta_{td}\right), \quad n_{wt} = \sum_{d \in D} n_{dw} p_{tdw}, \quad n_{td} = \sum_{w \in d} n_{dw} p_{tdw},$$

$$\phi_{wt} = \text{norm}^{+}_{w \in W}\left(n_{wt} + \phi_{wt}\frac{\partial R}{\partial \phi_{wt}}\right), \quad \theta_{td} = \text{norm}^{+}_{t \in T}\left(n_{td} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right),$$

where norm^{+} denotes non-negative normalization: $\text{norm}^{+}_{a \in A} x_a = \frac{\max\{x_a, 0\}}{\sum_{b \in A} \max\{x_b, 0\}}$. This system of equations yields a natural iterative algorithm (Newton's method)

for finding the parameters ϕ_{wt} and θ_{td}, equivalent to EM inference in pLSA; see [3] for a full derivation and a more detailed treatment. Thus, we have a model which is very easy to extend and which is computationally cheaper to train than the LDA model, especially LDA extensions that rely on Gibbs sampling.

To extend ARTM with an SVD-based regularizer, we begin with a regularizer in the same form as in Section 2: the total likelihood of the dataset with ratings comprised of triples $D = \{(i, a, r)\}$ (user i rated item a as $r \in \{-1, 1\}$) is a product of the likelihood of each rating, so its logarithm is

$$R(\Phi, \Theta) = \log p(D \mid \mu, b_i, b_a, p_i, q_a, l_i, \theta_a) = \\ = \sum_D \left([r = 1] \log \sigma \left(\hat{r}_{i,a} \right) + [r = -1] \log \left(1 - \sigma \left(\hat{r}_{i,a} \right) \right) \right),$$

where $[r = -1] = 1$ if $r = -1$ and $[r = -1] = 0$ otherwise, and θ_a is the vector of topics trained for document a in the LDA model, $\theta_a = \frac{1}{N_a} \sum_{w \in a} z_w$, where N_a is the length of document a, and

$$\hat{r}_{i,a} = \hat{r}^{\text{SVD}}_{i,a} + \theta_a^\top l_i = \mu + b_i + b_a + q_a^\top p_i + \theta_a^\top l_i.$$

To add this regularizer to the pLSA model, we have to compute its partial derivatives with respect to the parameters:

$$\frac{\partial R(\Phi, \Theta)}{\partial \phi_{wt}} = 0, \qquad \frac{\partial R(\Phi, \Theta)}{\partial \theta_{ta}} = \sum_{(i,a,r) \in D} \left[[r = 1] - \sigma(\hat{r}^{\text{SVD}}_{i,a} + \theta_a^\top l_i) \right] l_i;$$

note that the latter equality is exactly the same as (1) (hence we omit the derivation), only now it is a direct part of the algorithm rather than a first order approximation to the sampling. The final algorithm is, thus, to iterate the following:

$$p_{taw} = \text{norm}^+_{t \in T} \left(\phi_{wt} \theta_{ta} \right), \quad n_{wt} = \sum_{a \in D} n_{aw} p_{taw}, \quad n_{ta} = \sum_{w \in a} n_{aw} p_{taw},$$

$$\phi_{wt} = \text{norm}^+_{w \in W} n_{wt},$$

$$\theta_{ta} = \text{norm}^+_{t \in T} \left(n_{ta} + \rho \theta_{ta} \sum_{(i,a,r) \in D} \left([r = 1] - \sigma(\hat{r}^{\text{SVD}}_{i,a} + \theta_a^\top l_i) \right) l_i \right).$$

Similar to SVD-LDA, we can precompute $s_a = \sum_D \left([r = 1] - \sigma \left(\hat{r}^{\text{SVD}}_{i,a} + \theta_a^\top l_i \right) \right) l_i$ (it is a vector over topics) for each document after SVD is trained and use it throughout a pLSA iteration.

4 Conclusion

In this work, we have developed an ARTM regularizer that adds an SVD-based matrix decomposition model on top of ARTM. We have shown that the resulting

inference algorithms closely match the inference algorithms developed in the SVD-LDA modification of LDA with a first-order approximation to the Gibbs sampling. In further work, we plan to implement this regularizer and incorporate it into the BigARTM library [2, 3].

Acknowledgements. This work was supported by the Russian Science Foundation grant no. 15-11-10019.

References

1. Nikolenko, S.I.: SVD-LDA: Topic modeling for full-text recommender systems. In: Proc. 14th Mexican International Conference on Artificial Intelligence. LNAI vol. 9414, Springer (2015) 67–79
2. Vorontsov, K.: Additive regularization for topic models of text collections. Doklady Mathematics **89** (2014) 301–304
3. Potapenko, A., Vorontsov, K.: Robust pLSA performs better than LDA. In: Proc. 35th European Conf. on IR Research. LNCS vol. 7814, Springer (2013) 784–787
4. Vorontsov, K., Frei, O., Apishev, M., Romov, P., Suvorova, M., Yanina, A.: Non-bayesian additive regularization for multimodal topic modeling of large collections. In: Proc. of the 2015 Workshop on Topic Models: Post-Processing and Applications. TM '15, New York, NY, USA, ACM (2015) 29–37
5. Sokolov, E., Bogolubsky, L.: Topic models regularization and initialization for regression problems. In: Proc. of the 2015 Workshop on Topic Models: Post-Processing and Applications. TM '15, New York, NY, USA, ACM (2015) 21–27
6. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. Journal of Machine Learning Research **3** (2003) 993–1022
7. Griffiths, T., Steyvers, M.: Finding scientific topics. Proceedings of the National Academy of Sciences **101 (Suppl. 1)** (2004) 5228–5335
8. Wang, X., McCallum, A.: Topics over time: a non-Markov continuous-time model of topical trends. In: Proc. of the 12$^{\text{th}}$ ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, ACM (2006) 424–433
9. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proc. of the 23$^{\text{rd}}$ International Conference on Machine Learning, New York, NY, USA, ACM (2006) 113–120
10. Wang, C., Blei, D.M., Heckerman, D.: Continuous time dynamic topic models. In: Proceedings of the 24$^{\text{th}}$ Conference on Uncertainty in Artificial Intelligence. (2008)
11. Lacoste-Julien, S., Sha, F., Jordan, M.I.: DiscLDA: Discriminative learning for dimensionality reduction and classification. Advances in Neural Information Processing Systems **20** (2008)
12. Rosen-Zvi, M., Griffiths, T., Steyvers, M., Smyth, P.: The author-topic model for authors and documents. In: Proceedings of the 20$^{\text{th}}$ Conference on Uncertainty in Artificial Intelligence, Arlington, Virginia, United States, AUAI Press (2004) 487–494
13. Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P., Steyvers, M.: Learning author-topic models from text corpora. ACM Trans. Inf. Syst. **28** (2010) 1–38
14. Blei, D.M., McAuliffe, J.D.: Supervised topic models. Advances in Neural Information Processing Systems **22** (2007)
15. Hoffmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Machine Learning **42** (2001) 177–196