10708 Probabilistic Graphical Model Project

Final report: Non-parametric Hierarchical Supervised Latent Dirichlet Allocation

December 3, 2010

Yandong Liu YANDONGL@CS.CMU.EDU

Yanbo Xu YANBOX@CS.CMU.EDU

1. Introduction

Topic Models such as PLSI[7] and LDA[1] have been widely used in text analysis communities as well as other fields such as computer vision. Since the original idea of LDA was proposed, there has been a great deal of extension work such as: *supervised Topic Models*[2], where a label node was introduced to the model; *Hierarchical Dirichlet Processes*[7], which solved the nuisance that the number of mixture components has to be determined beforehand; and online inference of topics[4] that can update estimates of the topics as new document comes in. All those seem to be useful building blocks that people can exploit for various purposes. In this project we addressed on finding topics by applying online inference of Gibbs samplers on sLDA with HDP so that we obtain a model that is able to perform online inference over growing collections of documents. HDP assumes nonparametric prior for the number of mixture components, so that the choice of number of topics is more principled. Being supervised, the model is also able to predict instead of returning descriptive statistics of topics, hence make the evaluation more effective. We believe this model will be of importance because sLDA has been proved to be able to effectively predict both continuous values and categorical values[2]. However, nobody seems to have implemented a Gibbs sampling version of it. We also hope that the addition of HDP can make the model further more accurate.

Final report: We finished implementing collapsed Gibbs sampling on sLDA that selects number of topics via cross validation. We introduced HDP to sLDA that is able to select number of topics automatically.

2. Supervised LDA

Supervised Latent Dirichlet Allocation (sLDA) which is built on LDA introduces a response variable that is associated with the *label* of each document, and it is able to make predictions including for unconstrained continuous values and constrained categorical values. This model has a wide range of applications since many datasets such as movie ratings from movie review community, image relevance score from image sharing community, revenue forecast from financial firms, etc. often have labels available. The labels can be from experts' ratings, or collected by massive crowds in its community. *Features* are usually rich and a special type of features, *text*, is particularly abundant with this type of data, thus making *sLDA* fit well in such settings.

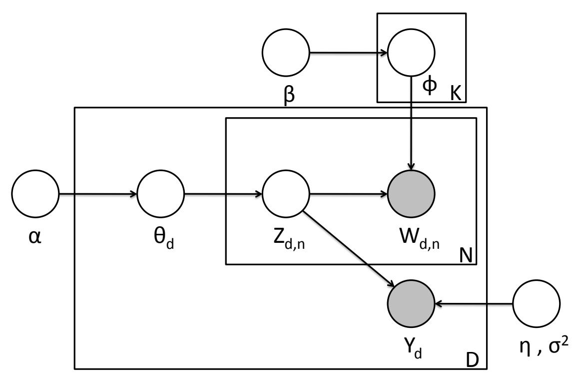

Figure 1. Plate notation of smoothed sLDA

sLDA is a generative model and under this model, each document and response variables arise from following process:

1. Draw topic proportions $\theta|\alpha \sim Dir(\alpha)$
2. For each word
   (a) Draw topic assignment $z_n|\theta \sim Multi(\theta)$
   (b) Draw topics $\phi_{1:K}|\beta \sim Dir(\beta)$
   (c) Draw word $w_n|z_n, \phi_{1:K} \sim Multi(\phi_{z_n})$
3. Draw response $y|z_{1:N}, \eta, \sigma^2 \sim N(\eta^T \bar{z}, \sigma^2)$

Here the notations defined are the same as in [2], except that we set a Dirichlet prior on $\beta_{1:K}$ to make a smoothed sLDA.

2.1. Inference

The original paper used variational expectation-maximization (EM) to approximate the maximum-likelihood of posterior distribution. Alternatively, we use collapsed Gibbs sampling, which is also used in HDP[3] to be consistent to our later work of adding HDP to sLDA.

According to the model, the joint distribution of the model is:

$$\begin{aligned} & P(\mathbf{Z}, \mathbf{W}, \mathbf{Y}, \theta, \phi; \alpha, \beta, \eta, \sigma^2) \\ = & \prod_{k=1}^{K} P(\phi_k|\beta) \prod_{d=1}^{D} P(\theta_d|\alpha) \prod_{n=1}^{N} P(Z_{d,n}|\theta_d) P(W_{d,n}|\phi_{Z_{d,n}}) \\ & \times P(Y_d|\bar{Z}_d, \eta, \sigma^2), \end{aligned} \tag{1}$$

where, θ and ϕ need to be integrated out. Notice that θ and ϕ are independent, and $P(\mathbf{Y}|\mathbf{Z}, \eta, \sigma^2)$ is irrelevant to θ and ϕ. So we have:

$$\begin{aligned} & P(\mathbf{Z}, \mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2) \\ = & \int_\theta \int_\phi P(\mathbf{Z}, \mathbf{W}, \mathbf{Y}, \theta, \phi; \alpha, \beta, \eta, \sigma^2) d\phi d\theta \\ = & \int_\theta \prod_{d=1}^{D} P(\theta_d|\alpha) \prod_{n=1}^{N} P(Z_{d,n}|\theta_d) d\theta \\ & \times \int_\phi \prod_{k=1}^{K} P(\phi_k|\beta) \prod_{d=1}^{D} \prod_{n=1}^{N} P(W_{d,n}|\phi_{Z_{d,n}}) d\phi \\ & \times \prod_{d=1}^{D} P(Y_d|Z_{d,n}, \eta, \sigma^2). \end{aligned} \tag{2}$$

The first two terms, integrating θ and ϕ, should be exactly the same as in the original LDA and the last term is given by

$$\begin{aligned} & \prod_{d=1}^{D} P(Y_d|Z_{d,n}, \eta, \sigma^2) \\ = & \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(Y_d - \eta^T \bar{Z}_d)^2}{2\sigma^2}), \end{aligned} \tag{3}$$

The goal of Gibbs sampling here is to approximate the posterior distribution of $P(\mathbf{Z}|\mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2)$, which is $\propto P(\mathbf{Z}, \mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2)$ since $P(\mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2)$ is invariable for any of Z. Thus, to sample the topic for the n^{th} word (which is the v^{th} word in the vocabulary) in the m^{th} document, $Z_{m,n}$, based on all the other topics $\mathbf{Z}_{-(\mathbf{m},\mathbf{n})}$, we will have

$$\begin{aligned} & P(Z_{(m,n)} = t|\mathbf{Z}_{-(\mathbf{m},\mathbf{n})}, \mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2) \\ \propto & P(Z_{(m,n)} = k, \mathbf{Z}_{-(\mathbf{d},\mathbf{n})}, \mathbf{W}, \mathbf{Y}; \alpha, \beta, \eta, \sigma^2) \\ \propto & \frac{n_{m,(.)}^{t,-(m,n)} + \alpha_t}{\sum_{k=1}^{K} n_{m,(.)}^{k,-(m,n)} + \alpha_k} \frac{n_{(.),v}^{t,-(m,n)} + \beta_v}{\sum_{i=1}^{V} n_{(.),i}^{t,-(m,n)} + \beta_i} \\ & \times \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(Y_m - \eta^T \bar{Z_m})^2}{2\sigma^2}), \end{aligned} \tag{4}$$

where $n_{d,i}^{k,-(m,n)}$ denotes the number of words in the d^{th} document with the same word symbol (the i^{th} word in the vocabulary) assigned to the k^{th} topic with the $Z_{(m,n)}$ excluded. We use parenthesized point (.) to denote unconstrained choice of variables.

Let $\mathbf{A}$ be the $(D \times K)$ matrix whose rows are the vectors $\bar{Z_d}^T$ and $\mathbf{Y}$ $D \times 1$ vector of document labels, then the η and σ can be predicted through MLE:

$$\eta_{\hat{MLE}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \tag{5}$$

$$\sigma^{2}_{\hat{MLE}} = \frac{1}{D}(\mathbf{Y}^T\mathbf{Y} - \mathbf{Y}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y}). \tag{6}$$

We update η and σ^2 by $\eta_{\hat{MLE}}$ and $\sigma^{2}_{\hat{MLE}}$ after sampling each document during each iteration.

2.2. Prediction

Given a new document $w_{1:N}$ and a fitted model including parameter η, we want to predict the response variable Y. The key here is to obtain the new $\bar{Z}$ for each document. For each word w_i in the new document, we can sample Z_{w_i} according to formula (4) but excluding the Y part. Let $\bar{Z} = \sum_{w_i} Z_{w_i}$, we have:

$$\hat{Y} = \eta^T \bar{Z} \tag{7}$$

as our prediction.

3. HDP+sLDA

A Dirichlet process *DP(α_0, G_0)* defines a distribution of a random probability measure G_j. In other words, a draw from DP will return a random distribution G_j with values drawn from G_0.

A hierarchical Dirichlet process is a distribution over a set of random probability measures. It returns a set of random probability measures G_j, one for each mixture, all from a global random probability measure G_0. The global measure G_0 is also distributed as a Dirichlet process with concentration parameters γ and base probability measure H:

$$G_0|\gamma, H \sim DP(\gamma, H)$$

and G_j obtained are conditionally independent given G_0, each distributed by a Dirichlet process with base probability measure G_0:

$$G_j|\alpha_0, G_0 \sim DP(\alpha_0, G_0)$$

More details about HDP, please refer to [3].

The characteristics of HDP, meaning during each draw HDP generates a different set of components while allowing some components to be shared among draws, is perfect for a mixture model such as LDA to automatically decide its number of components of the whole text collection. And in order to make it supervised, we can simply add a response node, yielding a graphical model as shown in figure 2. G_0 can be considered as the whole set of all possible topics. During each draw, we obtain a subset of topics G_j. $\theta_{i,j}$ is the same to the multinomial $\phi_{z_{d,n},j}$ in LDA and $x_{i,j}$ is the jth word observed in document i.

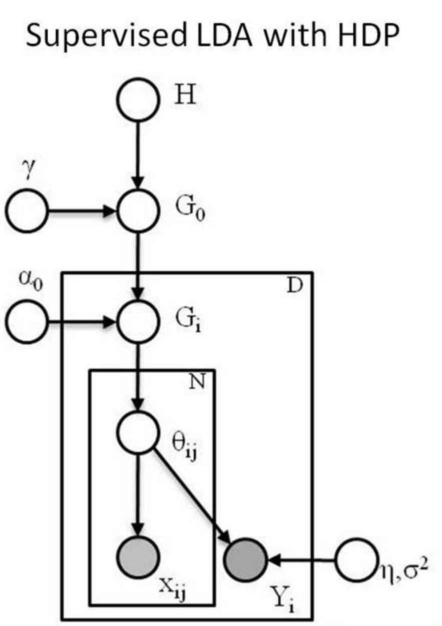

Figure 2. Plate notation of HDP+sLDA

We summarize all the distributions as follows:

- $G_0|\gamma, H \sim DP(\gamma, H)$
- $G_i|\alpha, G_0 \sim DP(\alpha, G_0)$
- $\theta_i j|G_i \sim G_i$
- $x_i j|\theta_i j \sim Multi(\theta_i j)$

3.1. Inference

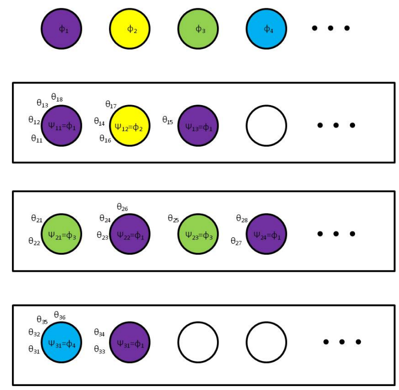

Figure 3. Chinese Restaurant Franchise

The Chinese restaurant franchise can yield a Gibbs sampling scheme for posterior sampling given observations x. Let the factor θ_{ji} be associated with the table t_{ji} in the restaurant representation; i.e. $\theta_{ji} = \psi_{ji}$. The random variable ψ_{jt} is an instance of mixture component k_{jt}; i.e. $\psi_{jt} = \phi_{k_{jt}}$. The prior over the parameters ϕ_k is H ($\sim Dir(\omega)$, conjugate to $F(\theta_{ji}) \sim Multi(\theta_{ji})$). We use the notation n_{jtk} to denote the number of customers in restaurant j at table t eating dish k, while m_{jk} denotes the number of tables in restaurant j serving dish k. Denote the conditional density of (x_{ji}, y_j) under mixture component k given all data items except x_{ji} as equation (8).

Similarly denote $f_k^{-x_{jt}}(x_{jt})$ as the conditional density of x_{jt} given all data items associated with mixture component k leaving out x_{jt}. Rather than dealing with the θ_{ji}'s and ψ_{jt}'s directly, we shall sample their index variables t_{ji} and k_{jt} instead. To compute the conditional distribution t_{ji} and k_{jt} given the remainder of the variables, we make use of exchangeability and treat t_{ji} and k_{jt} as the last variables being sampled in the last group.

The conditional distribution of t_{ji} is then in equation (9), where the likelihood for $t_{ji} = t^{new}$ can be calculated by integrating out the possible values of k_{jt}^{new} using equation (10).

If the sampled value of t_{ji} is t^{new}, we obtain a sample of k_{jt}^{new} by sampling from equation (11).

If the sampled value of t_{ji} is not a new table, then changing k_{jt} actually changes the component membership of all data items in table t, the likelihood obtained by setting $k_{jt} = k$ is given by $f_k^{-x_{jt}}(x_{jt})$, so that the conditional probability of k_{jt} is in equation (12).

4. Dataset

There are several datasets publicly available on the Web such as Movie data with ratings (http://www.ark.cs.cmu.edu/movie$-data/) which comes with textual movie review that was scraped from 7 different review websites such as *Austin Chronicle (www.austinchronicle.com), Boston Globe (www.boston.com)*, etc. For each movie, this dataset provides two response variables: gross revenue in its opening week, and number of screens on which the movie opened). As in paper[6], we use **Mean Absolute Error**(MAE) to evaluate our suggested approaches. We preprocessed the dataset in following ways:

- Combine reviews from all sources together as a single piece of text
- Convert reviews to lowercase. Remove non-alphabetical symbols and eliminate stop words

And we use its gross revenue in movie's opening weekend as the variable to predict in this project.

We randomly selected a subset of the whole dataset and split it to 2 separate parts:

- train_dev set (1000 documents): We need to regulate the number of topics for *sLDA* therefore we first applied a 10-fold cross validation to the randomized train_dev set. Then we used the whole train_dev set to learn other parameters of the model for both *sLDA* and *HDP*.
- test set (200 documents): The trained models(*sLDA* and *HDP*) were applied to this test set for inference as well as prediction.

5. External resources

We developed our implementation of sLDA based on the toolbox of [8][9] which is a Gibbs sampling version of original LDA. On top of it we introduced the response label for prediction functionality appropriately[2], as well as adapting Gibbs sampling accordingly so that η and σ can be updated as well during iterations.

For HDP+sLDA, we worked on the basis of Wang's code[1] which implemented a HDP-based unsupervised LDA with Gibbs sampling for inference.

6. Experiments

6.1. sLDA training with cross validation

As mentioned, a 10-fold **cross validation** was applied to the training_dev set. To get an idea how many topics there roughly are, we applied the off-the-shelf *HDP* to our 1000-document training set (500 iterations) and it returned 66 topics. Accordingly, we set K for *sLDA* to loop from 1 to 70 with 5 iterations each. We report "predictive R^2(pR^2)" in figure 4. As shown in [2], predictive R^2 is defined as:

$$pR^2 = 1 - (\Sigma(y - \hat{y})^2)/(\Sigma(y - \bar{y})^2) \qquad (13)$$

while $\hat{y}$ is the value we predicted and $\bar{y}$ is the mean of true labels.

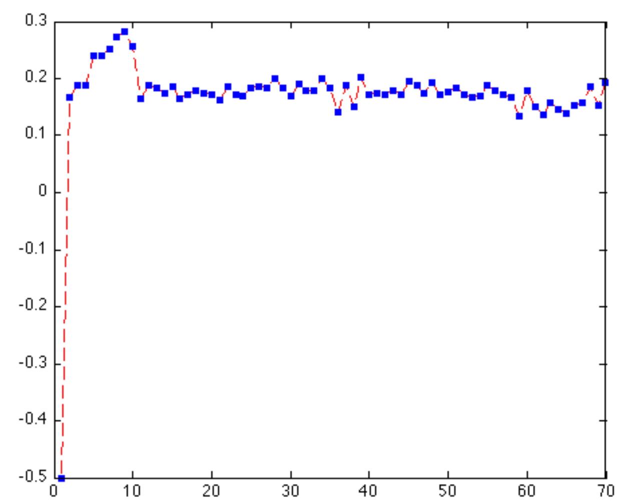

Figure 4. 10-fold cross validation for different numbers of topics. Note that pR^2 with *a single topic* is extremely bad (-219755685.990) and was manually changed to -0.5 for plotting purpose.

As we can see from figure 4 that *59 topics* gave us the closest point to 0 although through number *12-70* the pR^2 values are fairly close. We fixed $K = 59$ in our following experiments.

[1] http://www.cs.princeton.edu/ chongw/software/hdp.tar.gz

6.2. HDP+sLDA training

Similar to sLDA, we set the starting number of topics of HDP+sLDA to be 59. Priors of HDP (γ_a, γ_b, α_a, α_b), which control the probability of generating new *topics* and new *tables* during HDP's *Chinese Restaurant Process* seem to have somewhat significant impact to the result. Due to time constraint, we have a set of different γ_a to see how the performance might waggle.

6.3. Testing

On the test set which contains 200 documents, we report **predictive R^2, Label mean, Predicted mean**, as well as **Mean Absolute Error(MAE)**. All revenue numbers are in million US dollars. We started HDP+sLDA with 59 topics, and we also included the final number of topics (after 20 iterations).

pR^2	Label mean	Predicted mean	MAE
0.328	1.81	1.86	2.13

Table 1. sLDA prediction result on test

γ_a	#topics	pR^2	Label mean	Pred mean	MAE
0.5	49	0.13	1.81	0.84	1.97
1	30	0.18	1.81	1.49	2.18
2	49	0.18	1.81	0.97	1.94
3	39	0.12	1.81	0.68	1.88
4	60	0.13	1.81	0.84	1.95
5	45	0.12	1.81	0.66	1.91

Table 2. HDP+sLDA prediction result

[6] obtained the smallest **MAE of 5.738 million** with meta features and text. Their meta features include *whether file is of U.S. origin, running time in minutes*, etc. Text features consist of n-grams(unigram, bigrams and trigrams), Part-of-speech and dependency relations. [6] used linear regression to directly predict the opening weekend gross revenue and as a general rule, linear regression with a large number of terms are prone to overfit. Through topic models we essentially reduced the whole text space to only a few dimensions and is thus more robust to overfitting.

We report our sLDA prediction result in Table 1 and HDP+sLDA prediction result in Table 2. It is easy to see that both the results from sLDA and HDP+sLDA have a significantly better MAE than what [6] reported. We reached a **minimum MAE of 1.88M** from HDP+sLDA with $\gamma_a = 3$. Meanwhile we can observe that, although with some randomness during the sampling process, the number of topics HDP resolved is not very different from what cross-validation produced.

7. Discussion

Although HDP can automatically provide us the number of topics in the text collection, we still need to tune its hyperparameters α and γ. As in the Chinese restaurant franchise schema, the probability of sampling a new table is proportion to α while the probability of sampling a new topic is proportion to γ. The sampling process can become extremely slow if the number of tables or topics becomes absurdly large. Therefore these two parameters can affect both the convergence rate and the number of topics it generates. In our experiments, we select α as 1 and γ of 0.5 to 5 to obtain an acceptable running time. Meanwhile, it is also possible to tune these two parameters in the way of getting closer to the number of topics according to our prior knowledge. Thus, we may be able to have a cross validation to obtain a better pair of HDP hyperparmeters regarding time and resource consuming.

8. Conclusion

We have developed the Gibbs sampling version of *supervised Latent Dirichlet Allocation*. Experiments showed that it worked well for a linear regression task. It is a nontrivial task to reveal the number of components for such mixture models and although cross-validation is feasible in this project it takes a long time to finish. In addition, cross-validation becomes impractical when inference is time-confusing or the dataset becomes considerably large. *Hierarchical Dirichlet Process*, which provides a nonparametric prior for the number of mixture components within each group while allowing the components to be shared across groups, elegantly solves this problem and our experiments showed that the prediction capability of our supervised model based on HDP is similar to what its cross-validation peer returns.

References

[1] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. JMLR, 3:9931022, January 2003.

[2] D. Blei, J. McAuliffe. Supervised topic models. NIPS, 2007.

[3] Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical Dirichlet processes. Journal of the American Statistical Association, 2006

[4] K. R. Canini, L. Shi, T. L. Griffiths. Online Inference of Topics with Latent Dirichlet Allocation. In Proceedings of AI Stats (2009)

[5] Banerjee, A. and Basu, S. Topic models over text

streams: a study of batch and online unsupervised learn- ing. In Proc. 7th SIAM Int'l. Conf. on Data Mining

[6] M. Joshi, D. Das, K. Gimpel, and N. A. Smith. Movie Reviews and Revenues: An Experiment in Text Regression. In Proceedings of the ACL-HLT 2010.

[7] T. Hofmann, Probabilistic Latent Semantic Indexing, Proceedings SIGIR-99, 1999

[8] X. Phan, L. Nguyen, and S. Horiguchi. Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-scale Data Collections. In Proc. of The 17th International World Wide Web Conference (WWW 2008), pp.91-100, April 2008, Beijing, China.

[9] I. Biro, J. Szabo, and A. Benczur. Latent Dirichlet Allocation in Web Spam Filtering. In Proc. of The Fourth International Workshop on Adversarial Information Retrieval on the Web, WWW 2008, April 2008, Beijing, China.

$$
\begin{aligned}
& f_k^{-x_{ji}}(x_{ji}, y_j) \\
= \; & \frac{\int_{\phi_k} f_{x_{ji}|\theta_{ji}}(x_{ji}|\theta_{ji} = \phi_k) f_{y_j|\bar{\theta_j}}(y_j|\bar{\theta_j}) \prod_{j'i' \neq ji, k=k_{jt_{ji}}} f_{x_{ji}|\theta_{ji}}(x_{j'i'}|\theta_{j'i'} = \phi_k) h(\phi_k) \prod_{j' \neq j} f_{y_j|\bar{\theta_j}}(y_j|\bar{\theta_j}) d\phi_k}{\int_{\phi_k} \prod_{j'i' \neq ji, k=k_{jt_{ji}}} f_{x_{ji}|\theta_{ji}}(x_{j'i'}|\theta_{j'i'} = \phi_k) h(\phi_k) \prod_{j' \neq j} f_{y_j|\bar{\theta_j}}(y_j|\bar{\theta_j}) d\phi_k} \\
\propto \; & \frac{n_{(.),v}^{k,-(j,i)} + \alpha_0}{\sum_{v=1}^{V} n_{(.),v}^{k,-(j,i)} + \alpha_0} \times \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(Y_j - \eta^T \bar{\theta}_j)^2}{2\sigma^2}).
\end{aligned}
\tag{8}
$$

$$
p(t_{ji} = t|\mathbf{t}^{ji}, \mathbf{k}) \propto \begin{cases} n_{jt.}^{-ji} f_{k_{jt}^{-x_{ji}}}(x_{ji}, y_j) & \text{if t previously used,} \\ \alpha_0 p(x_{ji}|\mathbf{t}^{-ji}, t_{ji} = t^{new}, \mathbf{k}) & \text{if } t = t^{new}. \end{cases} \tag{9}
$$

$$
p(x_{ji}|\mathbf{t}^{-ji}, t_{ji} = t^{new}, \mathbf{k}) = \sum_{k=1}^{K} \frac{m_{.k}}{m_{..} + \gamma} f_k^{-x_{ji}}(x_{ji}, y_j) + \frac{\gamma}{m_{..} + \gamma} f_{k^{new}}^{-x_{ji}}(x_{ji}, y_j) \tag{10}
$$

$$
p(k_{jt^{new}} = k|\mathbf{t}, \mathbf{k}^{-jt^{new}}) \propto \begin{cases} m_{.k} f_k^{-x_{ji}}(x_{ji}, y_j) & \text{if k previously used,} \\ \gamma f_{k^{new}}^{-x_{ji}}(x_{ji}, y_j) & \text{if } k = k^{new}, \end{cases} \tag{11}
$$

$$
p(k_{jt} = k|\mathbf{t}, \mathbf{k}^{-jt}) \propto \begin{cases} m_{.k}^{-jt} f_{k^{-x_{jt}}}(x_{jt}, y_j) & \text{if k previously used,} \\ \gamma f_{k^{new}}^{-x_{jt}}(x_{jt, y_j}) & \text{if } k = k^{new}. \end{cases} \tag{12}
$$

Appendix: top 5 words from each topic

	1	2	3	4	5	6	7	8	8	10
1	tristan	amir	wong	hairspray	pocahontas	war	walle	gilliam	djay	brier
2	spiderman	florentino	kong	nomi	caden	soldiers	wallace	thompsons	rae	bertino
3	isolde	fermina	2046	schnabel	malick	iraq	valiant	grimm	brewer	speedman
4	ritchie	kite	geisha	travolta	malicks	president	gromit	gonzo	hustle	firefly
5	poppy	runner	balloon	reno	synecdoche	the	silverman	yale	lazarus	masuoka

Table 3. top 5 words from topic 1 - 10

	11	12	13	14	15	16	17	18	19	20
1	clooney	chatterley	amin	clouseau	vineyard	diggers	the	bollywood	yuma	constantine
2	murrow	tonya	israeli	guantanamo	nossiter	breda	movie	henderson	virgil	bruges
3	bettie	untraceable	nativity	panther	socrates	lehane	one	jamal	appaloosa	solomon
4	juno	fateless	moshe	dreyfus	mondovino	mullan	like	millionaire	alain	hounsou
5	leatherheads	gyuri	roos	kumar	provence	wahwah	film	lalita	kailey	mcdonagh

Table 4. top 5 words from topic 11 - 20

	21	22	23	24	25	26	27	28	29	30
1	hitler	giselle	gardener	the	marie	rod	georgia	polanski	pegg	deuce
2	hitlers	crewe	justin	one	v	puchi	herbie	johnston	marcos	poseidon
3	downfall	semipro	meirelles	film	penguin	santiago	pettigrew	jerome	fuzz	eggleston
4	vitus	yu	le	movie	sweeney	samberg	lohan	clowes	reygadas	gigolo
5	perfume	aquamarine	justins	like	antoinette	hector	lassie	polanskis	urea	lazarescu

Table 5. top 5 words from topic 21 - 30

	31	32	33	34	35	36	37	38	39
1	caspian	clerks	norbit	rory	phrasavath	geldzahler	evp	melinda	zandt
2	sebastien	dante	sancho	wilberforce	kuras	radcliffe	sax	hobie	zandts
3	aslan	randal	rowena	augusten	vinicio	maps	mcneice	melindas	townes
4	miraz	germs	rasputia	finch	manolo	kellyanne	keatons	sevigny	pancho
5	stphanie	moobys	ariel	deirdre	crnicas	pobby	chandra	laurel	categorized

Table 6. top 5 words from topic 31 - 39