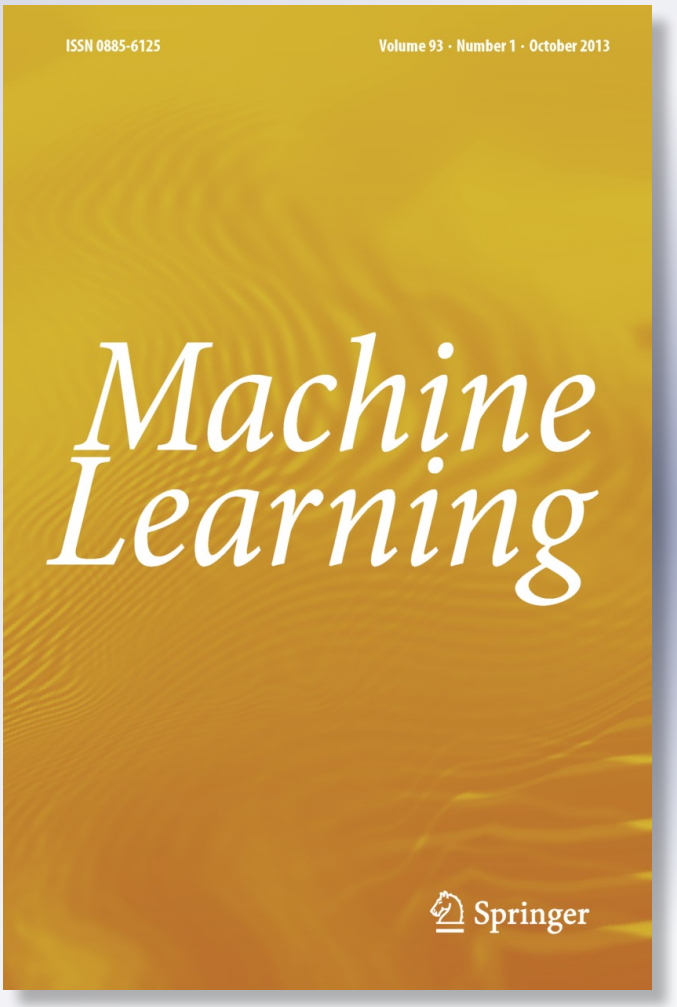
Probabilistic topic models for sequence data

Nicola Barbieri, Giuseppe Manco, Ettore Ritacco, Marco Carnuccio & Antonio Bevacqua

Machine Learning

ISSN 0885-6125
Volume 93
Number 1

Mach Learn (2013) 93:5-29
DOI 10.1007/s10994-013-5391-2



Springer

Mach Learn (2013) 93:5–29
DOI 10.1007/s10994-013-5391-2

Probabilistic topic models for sequence data

Nicola Barbieri · Giuseppe Manco · Ettore Ritacco · Marco Carnuccio · Antonio Bevacqua

Received: 9 September 2012 / Accepted: 4 June 2013 / Published online: 3 July 2013

Abstract Probabilistic topic models are widely used in different contexts to uncover the hidden structure in large text corpora. One of the main (and perhaps strong) assumption of these models is that generative process follows a bag-of-words assumption, i.e. each token is independent from the previous one. We extend the popular Latent Dirichlet Allocation model by exploiting three different conditional Markovian assumptions: (i) the token generation depends on the current topic and on the previous token; (ii) the topic associated with each observation depends on topic associated with the previous one; (iii) the token generation depends on the current and previous topic. For each of these modeling assumptions we present a Gibbs Sampling procedure for parameter estimation. Experimental evaluation over real-word data shows the performance advantages, in terms of recall and precision, of the sequence-modeling approaches.

Editors: Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný.

N. Barbieri (✉)
Yahoo Research, Av. Diagonal 177, Barcelona, Spain
e-mail: barbieri@yahoo-inc.com

G. Manco · E. Ritacco
Institute for High Performance Computing and Networks (ICAR), Italian National Research Council, via Bucci 41c, 87036 Rende, CS, Italy

G. Manco
e-mail: manco@icar.cnr.it

E. Ritacco
e-mail: ritacco@icar.cnr.it

M. Carnuccio · A. Bevacqua
Department of Electronics, Informatics and Systems, University of Calabria, via Bucci 41c, 87036 Rende, CS, Italy

M. Carnuccio
e-mail: mcarnuccio@deis.unical.it

A. Bevacqua
e-mail: abevacqua@deis.unical.it

Keywords Recommender systems · Collaborative filtering · Probabilistic topic models · Performance

Notations

M	# Traces
N	# Distinct tokens
K	# Topics
$\mathbf{W}$	Collection of traces, $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_M\}$
N_d	# tokens in trace d
$\mathbf{w}_d$	Token trace d, $\mathbf{w}_d = \{w_{d,1}.w_{d,2}.\cdots.w_{d,N_d-1}.w_{u,N_d}\}$
$w_{d,j}$	j-th token in trace d
$\mathbf{Z}$	Collection of topic traces, $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_M\}$
$\mathbf{z}_d$	Topics for trace d, $\mathbf{z}_d = \{z_{d,1}.z_{d,2}.\cdots.z_{d,N_d-1}.z_{d,N_d}\}$
$z_{d,j}$	j-th topic in trace d
$n^k_{d,s}$	Number of times token s has been associated with topic k for trace d
$\mathbf{n}_{d,(\cdot)}$	Vector $\mathbf{n}_{d,(\cdot)} = \{n^1_{d,(\cdot)}, \ldots, n^K_{d,(\cdot)}\}$
$n^k_{d,(\cdot)}$	Number of times topic k has been associated with trace d in the whole data
$\mathbf{n}^k_{(\cdot),r}$	Vector $\mathbf{n}^k_{(\cdot),r} = \{n^k_{(\cdot),r.1}, \ldots, n^k_{(\cdot),r.N}\}$
$n^k_{(\cdot),r.s}$	Number of times topic k has been associated with the token pair $r.s$ in the whole data
$\mathbf{n}^k_{(\cdot)}$	Vector $\mathbf{n}^k_{(\cdot)} = \{n^k_{(\cdot),1}, \ldots, n^k_{(\cdot),N}\}$
$n^k_{(\cdot),s}$	Number of times token s has been associated with topic k in the whole data
$\mathbf{n}^k_{d,(\cdot)}$	Vector $\mathbf{n}^k_{d,(\cdot)} = \{n^{k.1}_{d,(\cdot)}, \ldots, n^{k.K}_{d,(\cdot)}\}$
$n^{h.k}_{d,(\cdot)}$	Number of times that topic pair $h.k$ has been associated with the trace d
$n^{h.(\cdot)}_{d,(\cdot)}$	Number of times that a topic pair, that begins with topic h, has been associated with the trace d
$\mathbf{n}^{h.k}_{(\cdot)}$	Vector $\mathbf{n}^{h.k}_{(\cdot)} = \{n^{h.k}_{(\cdot),1}, \ldots, n^{h.k}_{(\cdot),N}\}$
$n^{h.k}_{(\cdot),s}$	Number of times that topic pair $h.k$ has been associated with the token s in the whole data
$\boldsymbol{\alpha}$	(LDA, TokenBigram and TokenBitopic Model) hyper parameters for topic Dirichlet distribution $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_K\}$ (Topic Bigram Model) set of hyper parameters for topic Dirichlet distribution $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}_0, \ldots, \boldsymbol{\alpha}_K\}$
$\boldsymbol{\alpha}_h$	Hyper parameters for topic Dirichlet distribution $\boldsymbol{\alpha}_h = \{\alpha_{h.1}, \ldots, \alpha_{h.K}\}$
$\boldsymbol{\beta}$	(LDA and TopicBigram Model) set of hyper parameters for token Dirichlet distribution $\boldsymbol{\beta} = \{\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K\}$ (TokenBigram Model) set of hyper parameters for token Dirichlet distribution $\boldsymbol{\beta} = \{\boldsymbol{\beta}_{1,1}, \ldots, \boldsymbol{\beta}_{K,1}, \ldots, \boldsymbol{\beta}_{1,2}, \ldots, \boldsymbol{\beta}_{K,2}, \ldots, \boldsymbol{\beta}_{K,N}\}$ (TokenBitopic Model) set of hyper parameters for token Dirichlet distribution $\boldsymbol{\beta} = \{\boldsymbol{\beta}_{1.1}, \ldots, \boldsymbol{\beta}_{K.1}, \ldots, \boldsymbol{\beta}_{1.2}, \ldots, \boldsymbol{\beta}_{K.2}, \ldots, \boldsymbol{\beta}_{K.K}\}$
$\boldsymbol{\beta}_k$	Hyper parameters for token Dirichlet distribution $\boldsymbol{\beta}_k = \{\beta_{k,1}, \ldots, \beta_{k,N}\}$
$\boldsymbol{\beta}_{k,s}$	Hyper parameters for token Dirichlet distribution $\boldsymbol{\beta}_{k,s} = \{\beta_{k,s.1}, \ldots, \beta_{k,s.N}\}$
$\boldsymbol{\beta}_{h.k}$	Hyper parameters for token Dirichlet distribution $\boldsymbol{\beta}_{h.k} = \{\beta_{h.k,1}, \ldots, \beta_{h.k,N}\}$
$\boldsymbol{\Theta}$	Matrix of parameters $\boldsymbol{\theta}_d$
$\boldsymbol{\theta}_d$	Mixing proportion of topics for trace d
$\vartheta_{d,k}$	Mixing coefficient of the topic k for trace d
$\vartheta_{d,h.k}$	Mixing coefficient of the topic sequence $h.k$ for the trace d
$\boldsymbol{\Phi}$	(LDA and TopicBigram Model) matrix of parameters $\boldsymbol{\varphi}_k = \{\varphi_{k,s}\}$ (TokenBigram Model) matrix of parameters $\boldsymbol{\varphi}_k = \{\varphi_{k,r.s}\}$ (TokenBitopic Model) matrix of parameters $\boldsymbol{\varphi}_{h.k} = \{\varphi_{h.k,s}\}$

$\varphi_{k,s}$ Mixing coefficient of the topic k for the token s
$\varphi_{k,r.s}$ Mixing coefficient of the topic k for the token sequence $r.s$
$\varphi_{h.k,s}$ Mixing coefficient of the topic sequence $h.k$ for the token s
$\boldsymbol{Z}_{-(d,j)}$ $\boldsymbol{Z} - \{z_{d,j}\}$
$\Delta(\boldsymbol{q})$ Dirichlet's Delta $\Delta(\boldsymbol{q}) = \frac{\prod_{p=1}^{P} \Gamma(q_p)}{\Gamma(\sum_{p=1}^{P} \Gamma(q_p))}$

1 Introduction and background

Probabilistic topic models, such as the popular *Latent Dirichlet Allocation (LDA)* (Blei et al. 2003), assume that each collection of documents exhibits an hidden thematic structure. The intuition is that each document may exhibit multiple topics, where each topic is characterized by a probability distribution over words of a fixed size dictionary. This representation of the data into the latent-topic space offers several advantages from a modeling perspective, and topic modeling techniques have been applied to different contexts. Example scenarios range from traditional problems (such as dimensionality reduction and classification) to novel areas (such as the generation of personalized recommendations).

Traditional LDA-based approaches propose a data generation process that is based on a "bag-of-words" assumption, i.e. such that the order of the items in a document can be neglected. This assumption fits textual data, where probabilistic topic models are able to detect recurrent co-occurrence patterns, which are used to define the topic space. However, there are several real-world applications where data can be "naturally" interpreted as sequences, such as biological data, web navigation logs, customer purchase history, etc. Ignoring the intrinsic sequentiality of the data, may result in poor modeling: according to the bag-of-word assumption, co-occurrences are modeled independently for each word, via a probability distribution over the dictionary in which some words exhibit a higher likelihood to appear than others. On the other hand, sequential data may express causality and dependency, and different topics can be used to characterize different dependency likelihoods. The focus here is the *context* where a current user acts and expresses preferences, i.e., the environment, characterized by side information, where the observations hold. Our claim is that the context can be enriched by the sequential information, and the latter allows a more refined modeling. In practice, a sequence expresses a context which provides valuable information for the modeling.

The above observation is particularly noteworthy when data express preferences made by users, and the ultimate objective is to model a user's behavior in order to provide accurate recommendations. The analysis of the sequential patterns has important applications in modern recommender systems (RSs), which are significantly focusing on an accurate balance between personalization and contextualization techniques. For example, in Internet based streaming services for music or video (such as Last.fm[1] and Videolectures.net[2]), the context of the user interaction with the system can easily be interpreted by analyzing the content previously requested. The assumption here is that the current item (and/or its genre) influences the next choice of the user. In particular, if a specific user is in the "mood" for classical music (as observed in the current choice), it is unlikely that the immediate subsequent choice will depart from the aforementioned mood, in favor of a song of different genre.

[1] http://last.fm.

[2] http://videolectures.net.

Being able to capture such properties and exploiting them in recommendation strategy can greatly improve the accuracy of the recommendation.

Recommender systems have greatly benefited from probabilistic modeling techniques based on LDA. Recent works in fact have empirically shown that probabilistic latent topics models represent the state-of-the-art in the generation of accurate personalized recommendations (Barbieri and Manco 2011; Barbieri et al. 2011, 2012). More generally, probabilistic techniques offer some renowned advantages: notably, they can be tuned to optimize a variety of loss functions; moreover optimizing the likelihood allows to model a distribution over rating values which can be used to determine the confidence of the model in providing a recommendation; finally, they allow the possibility to include prior knowledge into the generative process, thus allowing a more effective modeling of the underlying data distribution. Notably, when preferences are implicitly modeled through selection (that is, when no rating information is available), the simple LDA best models the probability that an item is actually selected by a user so far (Barbieri and Manco 2011).

Following the research direction outlined above, in this paper we study the effects of "contextual" information in probabilistic modeling of preference data. We focus on the case where the context can be inferred from the analysis of the sequence data, and we propose some topic models which explicitly make use of dependency information. As a matter of fact, the issue has been dealt with in similar papers (like, e.g. Wallach 2006). Here, we summarize and extend the approaches in the literature, by covering different ways of modeling dependency within preference data. Furthermore, we concentrate on the effects of such modeling on recommendation accuracy, as it explicitly reflects accurate modeling of user behavior.

In short, the contributions of the paper can be summarized as follows.

1. We propose a unified probabilistic framework to model dependency in preference data, and instantiate the framework in accordance to different assumptions on the sequentiality of the underlying generative process.
2. We study and experimentally compare the proposed models, and highlight relative advantages and weaknesses.
3. We study how to adapt the proposed frameworks to support a recommendation scenario. In particular, for each of the proposed model, we provide the relative ranking functions that can be used to generate personalized and context-aware recommendation lists.
4. We finally show that the proposed sequential modeling of preference data better models the underlying data, as it allows more accurate recommendations in terms of precision and recall.

The paper is structured as follows. In Sect. 2 we introduce sequential modeling according to different dependency assumptions, and specify in Sect. 3 the corresponding item ranking functions for supporting recommendations. The experimental evaluation of the proposed approaches is then presented in Sect. 4, in which we measure the performance of the approaches in a recommendation scenario. In Sect. 5 we qualitatively compare the models studied in this paper with the current literature. Section 6 concludes the paper with a summary of the findings and a discussion of possible extensions.

2 Modeling sequence data

In a general setting, we consider a set $\mathcal{I} = \{1, \ldots, N\}$ of tokens, representing the vocabulary of possible events that can be observed. Example events are words that can be observed in

a document, or items that can be purchased by a customer. A corpus $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_M\}$ is a collection of traces, where $\mathbf{w}_d = [w_{d,1}.w_{d,2}.\cdots.w_{d,N_d-1}.w_{d,N_d}]$ is the sequence of tokens for trace d, and $w_{d,j} \in \mathcal{I}$. The set $\mathcal{I}_d \subseteq \mathcal{I}$ denotes all the tokens in $\mathbf{w}_d$. We also assume that each token is characterized by a latent factor, called topic, triggering the underlying event. That is, a topic set $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_M\}$ is associated to the data, where, again $\mathbf{z}_d = [z_{d,1}.z_{d,2}.\cdots.z_{d,N_d-1}.z_{d,N_d}]$ is a latent topic sequence, and $z_{d,j} \in \{1, \ldots, K\}$ is the latent topic associated with token $w_{d,j}$. By assuming that $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ are the distribution functions governing the likelihood of $\mathbf{W}$ and $\mathbf{Z}$ (with respective priors $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$), we can express the complete likelihood as:

$$\begin{aligned} P(\mathbf{W}, \mathbf{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{\alpha}, \boldsymbol{\beta}) &= P(\mathbf{W}|\mathbf{Z}, \boldsymbol{\Phi})P(\boldsymbol{\Phi}|\boldsymbol{\beta})P(\mathbf{Z}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}|\boldsymbol{\alpha}) \\ P(\mathbf{W}|\mathbf{Z}, \boldsymbol{\Phi}) = \prod_{d=1}^{M} P(\mathbf{w}_d|\mathbf{z}_d, \boldsymbol{\Phi}), &\qquad P(\mathbf{Z}|\boldsymbol{\Theta}) = \prod_{d=1}^{M} P(\mathbf{z}_d|\boldsymbol{\theta}_d) \end{aligned} \tag{1}$$

where $P(\boldsymbol{\Phi}|\boldsymbol{\beta})$ and $P(\boldsymbol{\Theta}|\boldsymbol{\alpha})$ are specified according to the modeling assumptions. In particular, in the standard LDA setting where all tokens are independent and exchangeable, we have:

$$\begin{aligned} P(\mathbf{w}_d|\mathbf{z}_d, \boldsymbol{\Phi}) = \prod_{j=1}^{N_d} P(w_{d,j}|z_{d,j}, \boldsymbol{\Phi}), &\qquad P(w|k, \boldsymbol{\Phi}) = \prod_{s=1}^{N} \varphi_{k,s}^{\delta_{s,w}} \\ P(\mathbf{z}_d|\boldsymbol{\theta}_d) = \prod_{j=1}^{N_d} P(z_{d,j}|\boldsymbol{\theta}_d), &\qquad P(z|\boldsymbol{\theta}_d) = \prod_{k=1}^{K} \vartheta_{d,k}^{\delta_{k,z}} \\ P(\boldsymbol{\Theta}|\boldsymbol{\alpha}) = \prod_{d=1}^{M} P(\boldsymbol{\theta}_d|\boldsymbol{\alpha}), &\qquad P(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \vartheta_{d,k}^{\alpha_k - 1} \\ P(\boldsymbol{\Phi}|\boldsymbol{\beta}) = \prod_{k=1}^{K} P(\boldsymbol{\varphi}_k|\boldsymbol{\beta}_k), &\qquad P(\boldsymbol{\varphi}_k|\boldsymbol{\beta}_k) = \frac{\Gamma(\sum_{s=1}^{N} \beta_{k,s})}{\prod_{s=1}^{N} \Gamma(\beta_{k,s})} \prod_{s=1}^{N} \varphi_{k,s}^{\beta_{k,s} - 1} \end{aligned} \tag{2}$$

Here, $\delta_{a,b}$ represents the Kronecker delta function, returning 1 when $a = b$ and 0 otherwise. Figure 1(a) graphically describes the generative process. As usual, the joint topic-data probability can be obtained by marginalizing over the $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ components:

$$P(\mathbf{W}, \mathbf{Z}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \int_{\boldsymbol{\Phi}} \int_{\boldsymbol{\Theta}} P(\mathbf{W}|\mathbf{Z}, \boldsymbol{\Phi})P(\boldsymbol{\Phi}|\boldsymbol{\beta})P(\mathbf{Z}|\boldsymbol{\Theta})P(\boldsymbol{\Theta}|\boldsymbol{\alpha})d\boldsymbol{\Phi}d\boldsymbol{\Theta}$$

In the following, we model further assumptions on both $\mathbf{w}_d$ and $\mathbf{z}_d$, which explicitly reject the exchangeability assumption and instead rely on the idea of sequential dependency. We concentrate on three basic models, which in a sense subsume the core of sequential modeling. Here, a sequence can be modeled as a stationary first order Markov chain:

- A Markovian process naturally models the sequential nature of the data, where dependencies among past and future tokens reflect changes over time that are still governed by similar features;
- The chain is stationary, as a fixed number of tokens is likely to frequently appear in sequences;

- The order of the chain is 1 because the possibility that two subsequent tokens share some features is more likely than that of two tokens distant in time.[3]

We now analyze each model in turn.

Token-bigram model In this model, we assume that $\mathbf{w}_d$ represents a first-order Markov chain, where, each token $w_{d,j}$ depends on the most recent token $w_{d,j-1}$ observed by far. This is essentially the same model proposed in Wallach (2006), Cadez et al. (2000), and the probability of a trace has to be changed from Eq. (2) as

$$P(\mathbf{w}_d|\mathbf{z}_d, \boldsymbol{\Phi}) = \prod_{j=1}^{N_d} P(w_{d,j}|w_{d,j-1}, z_{d,j}, \boldsymbol{\Phi}) \tag{3}$$

In practice, a token $w_{d,j}$ is generated according to a multinomial distribution $\boldsymbol{\phi}_{z_{d,j},w_{d,j-1}}$ which depends on both the current topic $z_{d,j}$ and the previous token $w_{d,j-1}$. (Notice that when $j = 1$, the previous token is empty and the multinomial resolves to $\boldsymbol{\phi}_{z_{d,j}}$, representing the initial status of a Markov chain). The conjugate prior for $\boldsymbol{\phi}$ can be defined as:

$$P(\boldsymbol{\Phi}|\boldsymbol{\beta}) = \prod_{k=1}^{K}\prod_{r=0}^{N} P(\boldsymbol{\phi}_{k,r}|\boldsymbol{\beta}_{k,r}) = \prod_{k=1}^{K}\prod_{r=0}^{N} \frac{\Gamma(\sum_{s=1}^{N}\beta_{k,r,s})}{\prod_{s=1}^{N}\Gamma(\beta_{k,r,s})} \prod_{s=1}^{N} \varphi_{k,r,s}^{\beta_{k,r,s}-1}$$

Since the Markovian process does not affect the topic sampling, both $P(\mathbf{z}_d|\boldsymbol{\theta}_d)$ and $P(\boldsymbol{\Theta}|\boldsymbol{\alpha})$ are defined as in Eq. (2). The generative model, depicted in Fig. 1(b), can be described as follows:

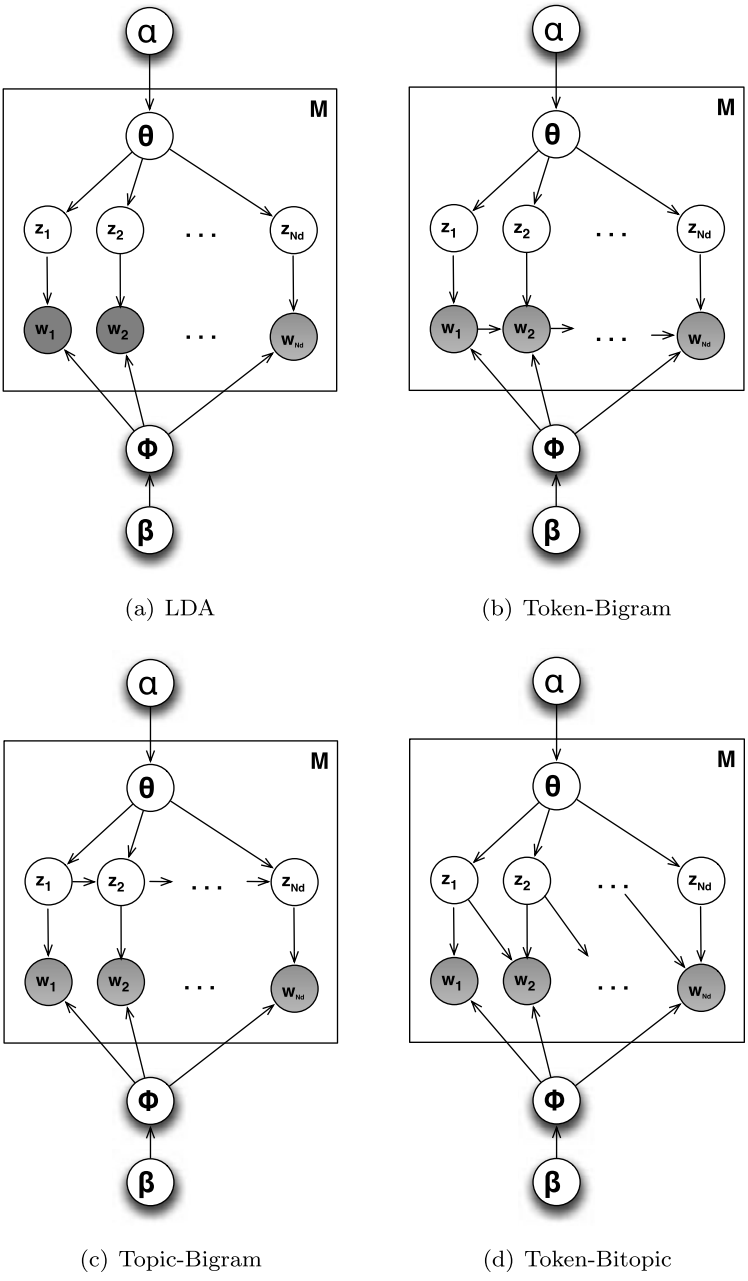
- For each trace $d \in \{1, \ldots, M\}$ sample the topic-mixture components $\boldsymbol{\theta}_d \sim \mathit{Dirichlet}(\boldsymbol{\alpha})$ and sequence length $n_d \sim \mathit{Poisson}(\xi)$
- For each topic $k \in 1, \ldots, K$ and token $r \in \{0, \ldots, N\}$
  - sample token selection components $\boldsymbol{\phi}_{k,r} \sim \mathit{Dirichlet}(\boldsymbol{\beta}_{k,r})$
- For each trace $d \in \{1, \ldots, M\}$ and $j \in \{1, \ldots, N_d\}$
  - sample a topic $z_{d,j} \sim \mathit{Discrete}(\boldsymbol{\theta}_d)$
  - sample a token $w_{d,j} \sim \mathit{Discrete}(\boldsymbol{\phi}_{z_{d,j},w_{d,j-1}})$

Notice that we explicitly assume the existence of a family $\{\boldsymbol{\beta}_{k,r}\}$ with $k = \{1, \ldots, K\}$ and $r = \{0, \ldots, N\}$ of Dirichlet coefficients, and of a special token $r = 0$ which represents the previous token of the first token of each trace. As shown in Wallach (2006), different modeling strategies (e.g., shared priors $\beta_{k,r,s} = \beta_s$) can affect the accuracy of the model.

By algebraic manipulations, the joint token-topic distribution can be simplified into:

$$P(\mathbf{W}, \mathbf{Z}|\alpha, \beta) = \left(\prod_{d=1}^{M} \frac{\Delta(\mathbf{n}_{d,(\cdot)} + \boldsymbol{\alpha})}{\Delta(\boldsymbol{\alpha})}\right)\left(\prod_{k=1}^{K}\prod_{r=0}^{N} \frac{\Delta(\mathbf{n}^{k}_{(\cdot),r} + \boldsymbol{\beta}_{k,r})}{\Delta(\boldsymbol{\beta}_{k,r})}\right) \tag{4}$$

[3]It is also worth noticing that higher order dependencies introduce an unpractical computational overhead, as the number of parameters grows exponentially with the order of the chain (Bishop 2006, Chap. 13).



(a) LDA (b) Token-Bigram

(c) Topic-Bigram (d) Token-Bitopic

Fig. 1 Graphical models

The latter is the basis for developing a stochastic EM strategy (Bishop 2006, Sect. 11.1.6), where the E step consists in a collapsed Gibbs sampling procedure (Heinrich 2008; Bishop 2006) for estimating $\mathbf{Z}$, and the M step estimates both the predictive distributions $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ and the hyper parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ given $\mathbf{Z}$. Within Gibbs sampling, topics are iteratively

sampled, according to the probability:

$$P(z_{d,j}=k|\mathbf{Z}_{-(d,j)},\mathbf{W}) \propto \left(n^k_{d,(\cdot)}+\alpha_k-1\right)\cdot\frac{n^k_{(\cdot),r,s}+\beta_{k,r,s}-1}{\sum_{s'=1}^{N} n^k_{(\cdot),r,s'}+\beta_{k,r,s'}-1} \tag{5}$$

relative to the topic to associate with the n-th token of the d-th trace, where $w_{d,j-1}=r$ and $w_{d,j}=s$.

Given $\mathbf{Z}$, the parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ can be estimated according to the following equations:

$$\vartheta_{d,k}=\frac{n^k_{d,(\cdot)}+\alpha_k}{\sum_{k'=1}^{K}(n^{k'}_{d,(\cdot)}+\alpha_{k'})},\qquad \varphi_{k,r,s}=\frac{n^k_{(\cdot),r,s}+\beta_{k,r,s}}{\sum_{s'=1}^{N}(n^k_{(\cdot),r,s'}+\beta_{k,r,s'})} \tag{6}$$

The estimation of the hyper parameters will be approached later in the paper.

Topic-bigram model A different approach can be taken by assuming that sequentiality regards topics, rather than tokens. That is, we can still consider tokens independent to each other and related to a latent topic. However, since topics represent the ultimate factors underlying a token appearance in the sequence, correlation between topics can better model an evolution of the underlying themes. Assuming a first-order Markovian dependency, the probability of a sequence of latent topics in Eq. (2) can be redefined as:

$$P(\mathbf{z}_d|\boldsymbol{\theta}_d)=\prod_{j=1}^{N_d} P(z_{d,j}|z_{d,j-1},\boldsymbol{\theta}_d) \tag{7}$$

The difference here is in the distribution generating $z_{d,j}$, which is a multinomial $\boldsymbol{\theta}_{d,z_{d,j-1}}$ parameterized by both a trace d and a previously sampled topic $z_{d,j-1}$. The conjugate Dirichlet distributions can be expressed as:

$$P(\boldsymbol{\Theta}|\boldsymbol{\alpha})=\prod_{d=1}^{M}\prod_{h=0}^{K}\frac{\Gamma(\sum_{k=1}^{K}\alpha_{h.k})}{\prod_{k=1}^{K}\Gamma(\alpha_{h.k})}\prod_{k=1}^{K}\vartheta_{d,h.k}^{\alpha_{h.k}-1} \tag{8}$$

$P(\mathbf{w}_d|\mathbf{z}_d,\boldsymbol{\Phi})$ and $P(\boldsymbol{\Phi}|\boldsymbol{\beta})$ are still defined as in Eq. (2). Again, the generative process is shown in Fig. 1(c) and described below.

- For each trace $d\in\{1,\ldots,M\}$ and topic $h\in\{0,\ldots,K\}$ sample topic-mixture components $\boldsymbol{\theta}_{d,h}\sim Dirichlet(\boldsymbol{\alpha}_h)$ and sequence length $N_d\sim Poisson(\xi)$
- For each topic $k=1,\ldots,K$
  - sample token selection components $\boldsymbol{\varphi}_k\sim Dirichlet(\boldsymbol{\beta}_k)$
- For each $d\in\{1,\ldots,M\}$ and $j\in\{1,\ldots,N_d\}$ sequentially:
  - sample a topic $z_{d,j}\sim Discrete(\boldsymbol{\theta}_{d,z_{d,j-1}})$
  - sample a token $w_{d,j}\sim Discrete(\boldsymbol{\phi}_{z_{d,j}})$

Here, $h=0$ is a special topic that precedes the first topic of each trace.

The joint token-topic distribution becomes:

$$P(\mathbf{W},\mathbf{Z}|\alpha,\beta)=\left(\prod_{d=1}^{M}\prod_{h=0}^{K}\frac{\Delta(\mathbf{n}^k_{d,(\cdot)}+\boldsymbol{\alpha}_h)}{\Delta(\boldsymbol{\alpha}_h)}\right)\left(\prod_{k=1}^{K}\frac{\Delta(\mathbf{n}^k_{(\cdot)}+\boldsymbol{\beta}_k)}{\Delta(\boldsymbol{\beta}_k)}\right) \tag{9}$$

and the corresponding collapsed Gibbs sampler works by iteratively sampling a topic k relative to token $w_{d,j} = s$ of trace d according to the following:

$$P(z_{d,j} = k|\mathbf{Z}_{-(d,j)}, \mathbf{W}) \propto \frac{n_{d,(\cdot)}^{z_{d,j-1}.k} + \alpha_{z_{d,j-1}.k} - 1}{\sum_{k'} n_{d,(\cdot)}^{z_{d,j-1}.k'} + \alpha_{z_{d,j-1}.k'} - 1} \cdot \frac{n_{d,(\cdot)}^{k.z_{d,j+1}} + \alpha_{k.z_{d,j+1}} - 1}{\sum_{k'} n_{d,(\cdot)}^{k'.z_{d,j+1}} + \alpha_{k'.z_{d,j+1}} - 1} \cdot \frac{n_{(\cdot),s}^{k} + \beta_{k,s} - 1}{\sum_{s'=1}^{N} n_{(\cdot),s'}^{k} + \beta_{k,s'} - 1} \tag{10}$$

Also, the multinomial parameters can be estimated according to the following equations:

$$\vartheta_{d,h.k} = \frac{n_{d,(\cdot)}^{h.k} + \alpha_{h.k}}{\sum_{k'=1}^{K} n_{d,(\cdot)}^{h.k'} + \alpha_{h.k'}}, \qquad \varphi_{k,s} = \frac{n_{(\cdot),s}^{k} + \beta_{k,s}}{\sum_{s'=1}^{N} n_{(\cdot),s'}^{k} + \beta_{k,s'}} \tag{11}$$

Token-bitopic model In the last model, we still relate tokens to past events. However, the events we are interested in are the recent latent topics which trigger the past tokens. The generative model is shown in Fig. 1(b). Again, topic selection probability is defined like in Eq. (2), whereas token selection probability can be defined in terms of the multinomial $\boldsymbol{\phi}_{z_{d,j},z_{d,j-1}}$ (and its related conjugate):

$$P(\mathbf{w}_d|\mathbf{z}_d, \boldsymbol{\Phi}) = \prod_{j=1}^{N_d} P(w_{d,j}|z_{d,j}, z_{d,j-1}, \boldsymbol{\Phi}) \tag{12}$$

$$P(\boldsymbol{\Phi}|\boldsymbol{\beta}) = \prod_{h=0}^{K} \prod_{k=1}^{K} \frac{\Gamma(\sum_{s=1}^{N} \beta_{h.k,s})}{\prod_{s=1}^{N} \Gamma(\beta_{h.k,s})} \prod_{s=1}^{N} \varphi_{h.k,s}^{\beta_{h.k,s}-1} \tag{13}$$

These assumptions are at the basis of the following generative process.

- For each trace $d \in \{1, \ldots, M\}$ sample topic-mixture components $\boldsymbol{\theta}_d \sim Dirichlet(\boldsymbol{\alpha})$ and sequence length $N_d \sim Poisson(\xi)$
- For each topic pair $h.k$, where $h \in \{0, \ldots, K\}$ and $k \in \{1, \ldots, K\}$
  - sample token selection components $\boldsymbol{\varphi}_{h.k} \sim Dirichlet(\beta_{h.k})$
- For each $d \in \{1, \ldots, M\}$ and $j \in \{1, \ldots, N_d\}$ in sequence:
  - sample a topic $z_{d,j} \sim Discrete(\boldsymbol{\theta}_d)$
  - sample a token $w_{d,j} \sim Discrete(\boldsymbol{\phi}_{z_{d,j},z_{d,j-1}})$

Once again $h = 0$ is the special topic which precedes all the first topics of the traces. As usual, by algebraic manipulations, the joint token-topic distribution can be expressed as

$$P(\mathbf{W}, \mathbf{Z}|\alpha, \beta) = \left(\prod_{d=1}^{M} \frac{\Delta(\mathbf{n}_{d,(\cdot)} + \boldsymbol{\alpha})}{\Delta(\boldsymbol{\alpha})} \right) \left(\prod_{h=0}^{K} \prod_{k=1}^{K} \frac{\Delta(\mathbf{n}_{(\cdot)}^{h.k} + \boldsymbol{\beta}_{h.k})}{\Delta(\boldsymbol{\beta}_{h.k})} \right) \tag{14}$$

which induce the following inference steps:

E step: for the token $w_{d,j} = s$ at position j in trace d, sample a topic k according to the following probability:

$$P(z_{d,j} = k|\mathbf{Z}_{-(d,j)}, \mathbf{W}) \propto \left(n^k_{d,(\cdot)} + \alpha_k - 1\right) \cdot \frac{n^{z_{d,j-1}.k}_{(\cdot),s} + \beta_{z_{d,j-1}.k,s} - 1}{\sum_{s'=1}^{N} n^{z_{d,j-1}.k}_{(\cdot),s'} + \beta_{z_{d,j-1}.k,s'} - 1} \cdot \frac{n^{k.z_{d,j+1}}_{(\cdot),s} + \beta_{k.z_{d,j+1},s} - 1}{\sum_{s'}^{N} n^{k.z_{d,j+1}}_{(\cdot),s'} + \beta_{k.z_{d,j+1},s'} - 1} \tag{15}$$

M Step: estimate multinomial probabilities according to the following equations:

$$\vartheta_{d,k} = \frac{n^k_{d,(\cdot)} + \alpha_k}{\sum_{k'=1}^{K} n^{k'}_{d,(\cdot)} + \alpha_{k'}}, \qquad \varphi_{h.k,s} = \frac{n^{h.k}_{(\cdot),s} + \beta_{h.k,s}}{\sum_{s'}^{N} n^{h.k}_{(\cdot),s'} + \beta_{h.k.s'}} \tag{16}$$

2.1 Log-likelihoods

A crucial component in the inference and estimation steps is the computation of the data likelihood. In general, the likelihood function is defined as:

$$P(\mathbf{W}) = \prod_{d=1}^{M} P(\mathbf{w}_d) = \prod_{d=1}^{M} P(w_{d,1}.\cdots.w_{d,N_d}) = \prod_{d=1}^{M}\sum_{k=1}^{K} P(w_{d,1}.\cdots.w_{d,N_d}, z_{d,N_d} = k)$$

Now, each model differs in the way the $P(w_{d,1}.\cdots.w_{d,N_d}, z_{d,N_d})$ component is defined.

Token-bigram Bayes rule and the first order Markov assumption over tokens simplifies the above probability into:

$$\log P(\mathbf{W}) = \sum_{d=1}^{M} \log\left(\prod_{j=1}^{N_d}\sum_{k} \vartheta_{d,k}\varphi_{k,w_{d,j-1}.w_{d,j}}\right) \tag{17}$$

Topic-bigram By algebraic manipulations (see Bishop 2006, Sect. 13.2 for details), we obtain

$$\begin{aligned} &P(w_{d,1}.\cdots.w_{d,N_d}, z_{d,N_d} = k) \\ &\quad = P(w_{d,1}.\cdots.w_{d,N_d}|z_{d,N_d} = k)P(z_{d,N_d} = k) \\ &\quad = \varphi_{k,w_{d,N_d}} \sum_{h} P(w_{d,1}.\cdots.w_{d,N_d-1}, z_{d,N_d-1} = h)\vartheta_{d,h.k} \end{aligned}$$

The result is a recursive equation which can be simplified into the following γ function:

$$\gamma_k(\mathbf{w}_d; 1) = \varphi_{k,w_{d,1}}; \qquad \gamma_k(\mathbf{w}_d; j) = \varphi_{k,w_{d,j}} \sum_{h} \gamma_h(\mathbf{w}_d; j-1)\vartheta_{d,h.k}$$

Substituting into the likelihood, yields:

$$\log P(\mathbf{W}) = \sum_{d=1}^{M} \log\left(\sum_{k} \gamma_k(\mathbf{w}_d; N_d)\right) \tag{18}$$

Token-bitopic The term $P(w_{d,1}.\cdots.w_{d,N_d}|z_{d,N_d}=k)$ can be decomposed according to the assumption of independence among topics:

$$\begin{aligned} &P(w_{d,1},\ldots,w_{d,N_d}|z_{d,N_d}=k) \\ &\quad=\sum_{h=1}^{K}\vartheta_{d,h}P(w_{d,1}.\cdots.w_{d,N_d}|z_{d,N_d-1}=h,z_{d,N_d}=k) \\ &\quad=\sum_{h=1}^{K}\vartheta_{d,h}\varphi_{h.k,s}P(w_{d,1}.\cdots.w_{d,N_d-1}|z_{d,N_d-1}=h) \end{aligned}$$

where $w_{d,N_d}=s$. Again, the latter yields the following recursive equations

$$\gamma_k(\mathbf{w}_d,1)=\varphi_{w_{d,1},\epsilon.k};\qquad \gamma_k(\mathbf{w}_d,j)=\sum_{h}\gamma_h(\mathbf{w}_d,j-1)\vartheta_{d,h}\varphi_{w_{d,j},h.k}$$

where ϵ is a special topic, referring to the begin of the trace. The likelihood can hence be expressed as:

$$\log P(\mathbf{W}) = \sum_{d=1}^{M} \log\left(\sum_{k} \gamma_k(\mathbf{w}_d; N_d)\vartheta_{d,k}\right) \tag{19}$$

2.2 Estimating the Hyper parameters

We consider asymmetric Dirichlet priors over the trace topic distributions and a symmetric prior over the topic distributions. This modeling strategy has been reported to achieve important advantages over the symmetric version (Wallach et al. 2009a). For the token-bigram and token-bitopic models, we adopted the procedure for updating the prior $\boldsymbol{\alpha}$ as described in Heinrich (2008), Minka (2000). The topic-bigram model requires a difference formulation of the latter. Given a state of the Markov chain $\mathbf{Z}$, the optimal α-hyper parameters can be computed by maximizing the likelihood of the observed pseudo-counts $n^{h.k}_{d,(\cdot)}$ via the fixed-point iteration method:

$$\alpha^{new}_{h.k} = \alpha_{h.k}\frac{\sum_{d=1}^{M}\Psi(n^{h.k}_{d,(\cdot)}+\alpha_{h.k})-M\Psi(\alpha_{h.k})}{\sum_{d=1}^{M}\Psi(n^{h.(\cdot)}_{d,(\cdot)}+\sum_{k'=1}^{K}\alpha_{h.k'})-M\Psi(\sum_{k'=1}^{K}\alpha_{h.k'})} \tag{20}$$

where $\Psi(\cdot)$ indicates the digamma function.

3 Application to Recommender Systems

The general framework introduced above has a natural interpretation when dealing with users' preference data: the set of users defines the corpus, each user is considered as a trace, the items purchased are considered as tokens and, finally, the topics correspond, intuitively,

to the reason why the users purchased particular products. In the following, we assume that a user can be denoted by a unique index d, and a previous history is given by $\mathbf{w}_d$ of size N_d. We are interested in providing a ranking for s, the $(N_d + 1)$-th choice w_{d,N_d+1}.

LDA Following (Barbieri and Manco 2011) we adopt the following ranking function:

$$rank(s, d) = \sum_{k=1}^{K} P(s|z_{d,N_d+1} = k) P(z_{d,N_d+1} = k|\boldsymbol{\theta}_d) = \sum_{k=1}^{K} \varphi_{k,s} \cdot \vartheta_{d,k}$$

It has been shown (Barbieri and Manco 2011) that LDA, equipped with the above ranking function, significantly outperforms the most significant approaches to modeling user preferences. Hence, it is a natural baseline function upon which to measure the performance of the other approaches proposed in this paper.

Token-bigram model The dependency of the current selection from the previous history can be made explicit, thus yielding the following upgrade to the LDA ranking function:

$$rank(s, d) = \sum_{k=1}^{K} P(s|z_{u,N_d+1} = k, \mathbf{w}_d) P(z_{d,N_d+1} = k|\boldsymbol{\theta}_d) = \sum_{k=1}^{K} \varphi_{k,r.s} \cdot \vartheta_{d,k}$$

where $r = w_{u,N_d}$ is the last item selected by user d in her currently history.

Topic-bigram model This situation resembles the forward-backward algorithm for the hidden Markov models (Bishop 2006, Sect. 13.2.2). In practice, we need to build a recursive chain of probabilities, representing a hypothetical random walk among the hidden topics. As above, we can define the following rank:

$$\begin{aligned} rank(s, d) &= \sum_{k=1}^{K} P(w_{d,N_d+1} = s, z_{u,N_d+1=k}|\mathbf{w}_d) \\ &= \sum_{k=1}^{K} \frac{P(w_{d,1}.\cdots.w_{d,N_d+1}, z_{d,N_d+1})}{P(\mathbf{w}_d)} \end{aligned}$$

which requires solving $P(w_{d,1}.\cdots.w_{d,N_d+1}, z_{d,N_d+1})$. As shown in the previous section, the latter can be computed recursively by exploiting the γ function. Hence, the ranking function can be formulated as:

$$rank(s, d) \propto \sum_{k=1}^{K} \gamma_k(\mathbf{w}_d.s, N_d + 1)$$

Token-bitopic model Since in this case item selection depends on the previous topics, by exploiting the γ function, we can define the following:

$$\begin{aligned} rank(s, d) &= P(w_{d,N_d+1} = s|\mathbf{w}_d) \\ &\propto \sum_{k=1}^{K} P(w_{d,1}.\cdots.w_{d,N_d+1}, z_{d,N_d+1}) \\ &= \sum_{k=1}^{K} \gamma_k(\mathbf{w}_d.s, N_d + 1)\vartheta_{d,k} \end{aligned}$$

4 Experimental evaluation

In this section we study the behavior of the proposed models, compared to some baseline models. In particular, we study two main aspects.

- On a general setting, we study how the proposed method perform in terms of quality. We measure the quality as a function of the likelihood, as explained in the next section.
- On a more specific setting, we compare the models in the envisaged recommendation scenario. Here, the quality of a model is measured indirectly, in terms of the accuracy of the recommendations it boosts. This is explained in Sect. 4.2.

4.1 Perplexity

Topic models are typically evaluated by either measuring performance on some secondary task, such as document classification or information retrieval, or by estimating the probability of unseen held-out traces given some training traces. Notably, a better model will give rise to a higher probability of held-out traces, on average.

Since log likelihoods are usually large negative numbers, perplexity is used instead (Heinrich 2008; Blei et al. 2003), the latter being defined as the reciprocal geometrical mean of the token likelihoods in the test corpus given the data used to train the model:

$$Perp(\mathbf{W}_{Test}|\mathbf{W}_{Train}) = \exp\left\{-\frac{\sum_{d=1}^{N_{Test}} \log P(\mathbf{w}_d|\mathbf{W}_{Train})}{\sum_{d=1}^{N_{Test}} n_d}\right\}$$

Evaluating $P(\mathbf{w}_d|\mathbf{W}_{Train})$ is a little tricky, as exact inference would require integrating over all possible model parameters. In Wallach et al. (2009b) authors discuss some methods for an accurate inference using a point estimate. In our experiments we adopted the evaluation methods based on document completion. This method offers the advantage of providing unbiased estimates, as it infers the missing parameters on a separate part of the document, and then to evaluate the perplexity on the remaining part. In short, the evaluation methodology can be summarized as follows:

- For each $\mathbf{w}_d \in \mathbf{w}^{Test}$
  1. Let $\mathbf{w}_d^{(1)}$ and $\mathbf{w}_d^{(2)}$ be an arbitrary split of $\mathbf{w}_d$.
  2. For $s = 1, \ldots, S$
     (a) sample $\mathbf{z}^{(1,s)} \sim P(\mathbf{z}^{(1,s)}|\mathbf{w}_d^{(1)}, \mathbf{W}_{train}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Phi})$ using the Gibbs Sampling equations;
     (b) estimate $\boldsymbol{\theta}_d^{(s)}$ from $\mathbf{z}^{(1,s)}$;
  3. Approximate $P(\mathbf{w}_d|\mathbf{W}_{Train})$ with $\frac{1}{S}\sum_s P(\mathbf{w}_d^{(2)}|\boldsymbol{\theta}_d^{(s)}, \boldsymbol{\Phi})$, where the latter is computed by exploiting the formulas in Sect. 2.1.

Following (Wallach 2006), in the experiments we use a dataset composed by drawing 150 Psychological Review abstracts from the data made available by Griffith and Steyvers.[4] The drawing was made among those documents containing at least 54 tokens. Also, we preprocessed the data as specified in Wallach (2006), by remapping all numbers with the special token `NUMBER`, and all items with frequency 1 in the training set or appearing as tokens in the test set but not in the training set as `UNSEEN`. The result of the cleaning process

[4] http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm.

is a vocabulary of 860 item. Starting with the cleaned dataset, we did several random splits of the dataset, by choosing 100 documents as training data and keeping the remaining ones as test data. The splits roughly maintained the proportion 67–33 % on the tokens.

In the following we report the results obtained by the three proposed models. The results are compared with LDA. We also compare the models with the DCMLDA model (Doyle and Elkan 2009). The latter is a modification of LDA to account for the tendency of tokens to appear in bursts, that is if a token appears once in a trace, it is more likely to be appear again. DCMLDA does not model sequentiality, however burstiness can also be interpreted as non-independence between tokens. In this respect, it is interesting how the proposed models compare to it. It is worth noticing, however, that burstiness is not necessarily alternative to sequentiality, as the approaches proposed in this paper can easily be adapted to model a combination of burstiness and sequentiality.

Figure 2(a) reports the average perplexity on the test data. The values plot the error bars related to the perplexity values. Figure 2(b) also analyzes the pairwise comparisons: each of the three methods proposed here is compared with the baselines, and the difference in perplexity (in average and standard error) is plotted.

DCMLDA exhibits the best perplexity, as a result of the customized fitting of token probabilities to a specific document. As a matter of fact, the documents we are investigating here seem to naturally comply with the burstiness assumption.

Also, TokenBitopic seems to worsen the performance as the number of topics increase. This behavior is worth further explanation. The model conditions the probability of appearance of a token to a pair of latent factors. In a sense, this makes the model comparable to a "fresh" LDA model, where the number of latent factors is quadratic in K: in practice, a TokenBitopic model with $K = 4$ can be deemed similar to an LDA model with $K = 16$ topics, and each pair of latent factors is associated to a specific latent factor in the quadratic LDA model. In Fig. 2(c) we compare the two models: the models show the same tendency.

For the rest, models clearly outperform LDA. However, the TokenBigram model requires further explanation. Both the sampling process and the item selection probabilities rely on the frequencies of bigrams. Zero-frequency bigrams appearing in the test set compromise the evaluation just like zero-frequency items. We chose to treat them by associating them with a default frequency. Figure 2(d) shows how this affects the evaluation: here, `NoP` corresponds to keeping the original frequency, whereas `P3` associates a frequency which implicitly corresponds to flattening all the zero-frequency bigrams to a default `UNSEEN` bigram. The latter is the one reported in Fig. 2(a). The approaches `P1` and `P2` correspond to intermediate solutions, where the default frequency of the (implicit) `UNSEEN` bigram is lowered.[5]

Finally, Fig. 2(e) denotes the running times of the training algorithms on the training data. Although the TopicBigram model requires less parameters than the TokenBitopic approach, the learning time of the first one is considerably larger. This is mainly due to the larger number of hyper parameters ($K \times K$ vs. K) and to the complexity of the M step for the update of the hyper parameters $\boldsymbol{\alpha}$.

4.2 Recommendation accuracy

In this section we present an empirical evaluation of the proposed models which focuses on the recommendation problem. Given the past observed preferences of a users, the goal

[5]Clearly this is where non-parametric methods should be used to provide a gradual step into the TokenBigram model. The integration of non-parametric techniques in the TokenBigram would better handle cases in which there is less data and it would automatically solve the treatment of the zero-frequency items.

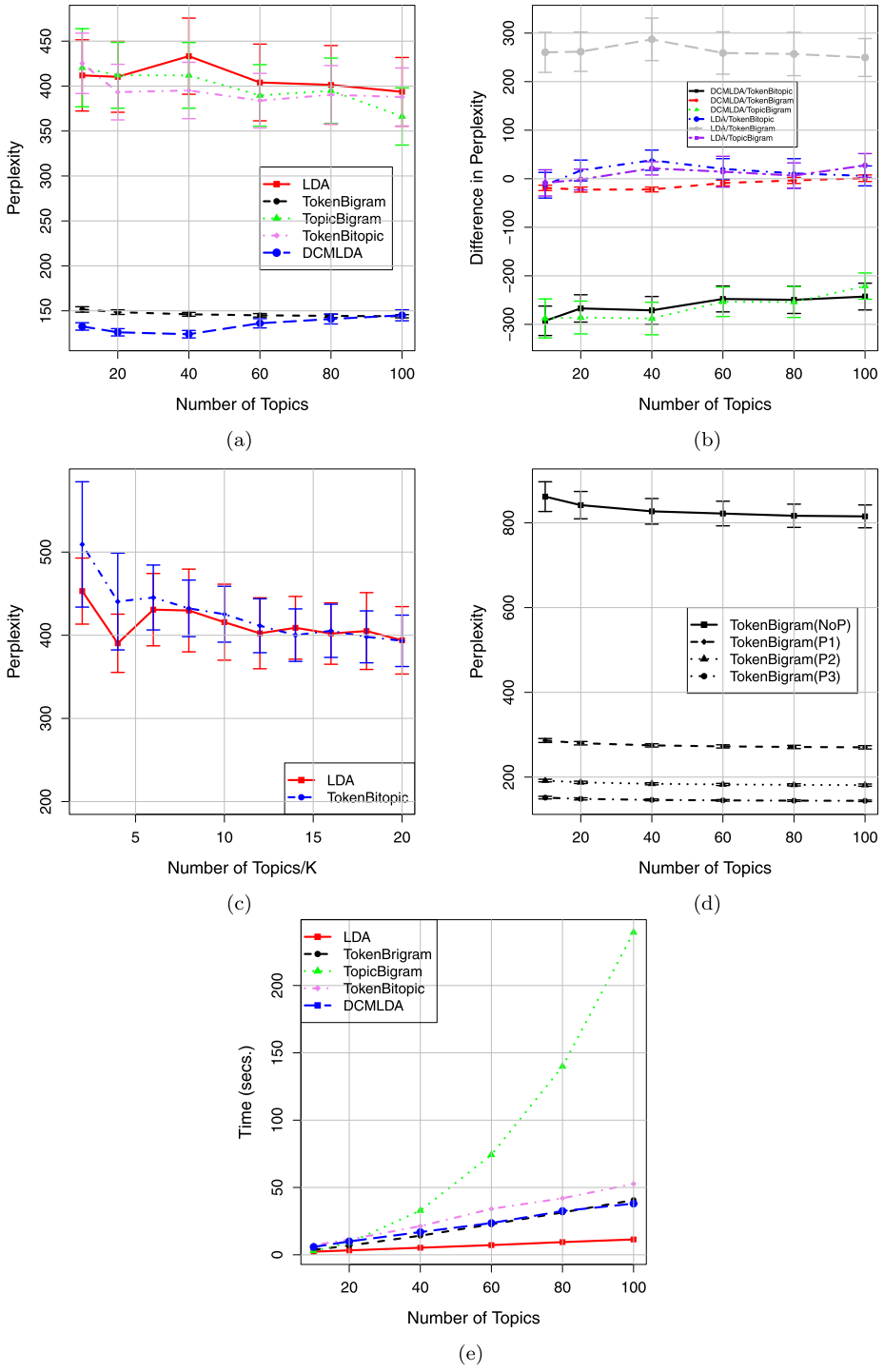


Fig. 2 Performance on Psychreview data

Table 1 Summary statistics on real-life recommendation datasets

	IPTV1		IPTV2	
	Training	Test	Training	Test
Users	16,237	16,153	64,334	63,878
Items	759	731	2802	2777
Evaluations	314,042	78,557	1,224,790	306,271
Avg # evals (user)	19	5	19	5
Avg # evals (item)	414	107	437	110
Min # evals (user)	4	1	4	1
Min # evals (item)	5	1	5	1
Max # evals (user)	252	15	497	17
Max # evals (item)	2284	1527	9606	3167
	Avg time between two evals			
per user	13 days		6 days	
per item	9 hours		23 hours	

of a recommender systems is to provide her with personalized (and contextualized) recommendations about previously non-purchased items that meet her interest. We evaluate the proposed techniques by measuring their predictive abilities on two datasets, namely IPTV1 and IPTV2. These data were collected by analyzing the pay-per-view movies purchased by the users of two European IPTV providers over a period of several months (Cremonesi and Turrin 2009; Bambini et al. 2011). The original data have been preprocessed by removing users with less than 10 purchases. We perform a chronological split of the data by selecting the final 20 % purchases of each user as test data, and using the remaining data for training purposes. The main features of the datasets are summarized in Table 1.

The two datasets exhibit a substantial difference in the frequencies of bigrams, as shown in Fig. 3: in particular, IPTV2 exhibits frequencies which differ of an order of magnitude. Hence, by comparing the results of the proposed algorithms, we can characterize the effects of sparsity on the performances of the proposed methods.

Testing protocol Let $\mathbf{W}_{Train}$ and $\mathbf{W}_{Test}$ denote respectively training and test data. To evaluate the capabilities of the considered approaches in generating accurate recommendations, we check whether an actual token can be included into an hypothetical recommendation list containing H items, generated according to the model. More specifically the following protocol is adopted, which is justified and detailed in Barbieri and Manco (2011):

- For each user u, let $\mathbf{w}'_u$ be the trace associated to u in $\mathbf{W}_{Train}$, and $\mathbf{w}_u$ the trace in $\mathbf{W}_{Test}$ (with $n_u = |\mathbf{w}_u|$). For each token $w_{u,n} \in \mathbf{w}_u$:
  - generate the candidate list $\mathcal{C}_u$ by randomly drawing c items $i \neq w_{u,n}$ such that $i \notin \mathcal{I}_{\mathbf{w}'_u}$;
  - add $w_{u,n}$ to $\mathcal{C}_u$ and sort the list according to the scoring function provided by the RS;
  - record the position of the $w_{u,n}$ in the ordered list: if it belongs to the top-H items, we have a *hit* otherwise, we have a *miss*.

Recall and precision relative to u can hence be defined based on the number of hits. Recall can be defined as the number of hits, relative to the expected number of relevant items (which are all the items in $\mathbf{w}_u$). Also, precision represents the probability that the top-ranked items are actually a hit (and hence it represents the likelihood of a hit weighted by the size H) of

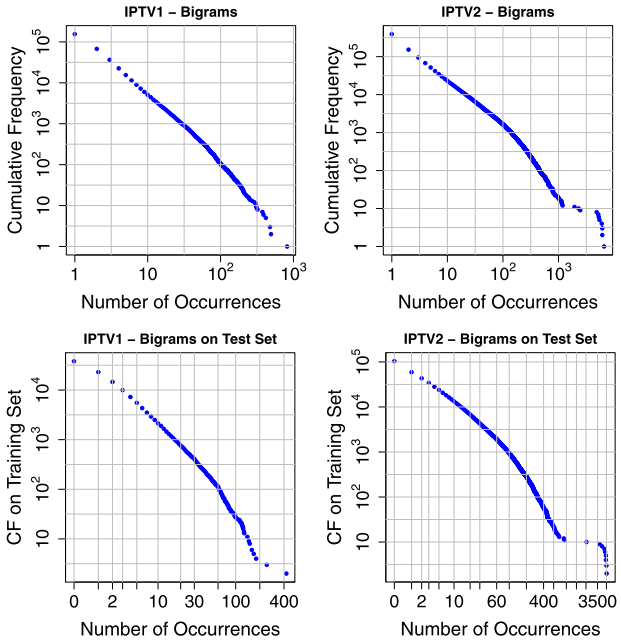


Fig. 3 Distributions of bigrams on real-life datasets

the recommendation list. In formulas:

$$Recall(u, H) = \frac{\#hits}{n_u}, \qquad Precision(u, H) = \frac{\#hits}{H \times n_u} = \frac{recall(u, H)}{H} \tag{21}$$

The final precision and recall values are obtained averaging on all users. All the considered models were run varying the number of topics. We perform 5000 Gibbs Sampling iterations, discarding the first 1000 (burn in period), and with a sample lag of 30. The length of the candidate random list is set to 250 for IPTV1 and 1000 for IPTV2.

In the evaluation, we compare the bigram models with some baseline methods from the current literature. These include the aforementioned DCMLDA model, and a version of the LDA where, for each user, the tokens represent (unordered) bigrams rather than single item occurrences. This is in practice a preprocessing of the data, which produces a different representation of the dataset upon which the standard LDA model is trained. Clearly, the ranking function has to be tuned accordingly.

We also provide two further baselines. The first one is a simple bigram model where the probability of occurrence of an item is modeled as $P(w_n) = \lambda f_{w_n} + (1 - \lambda) f_{w_n|w_{n-1}}$. Here, f_i is the relative frequency of i in the training set, whereas $f_{i|j}$ represents the same frequency conditioned to a preceding occurrence of j in the sequence. The λ parameter weighs the importance of the two components, and is tuned in a way proportional to the frequency of i, as typically low-frequency items do not provide a reliable estimates of the sequential part.

Finally, we also compare the proposed models to a baseline rooted on matrix factorization (Koren et al. 2009; Menon and Elkan 2011). The basic idea here is to exploit the matrix factorization for ranking, e.g., by providing an estimate of the probability of the item appearance (Menon and Elkan 2010). There are some issues to consider when applying matrix factorization to the case at hand. In our context, matrix factorization is

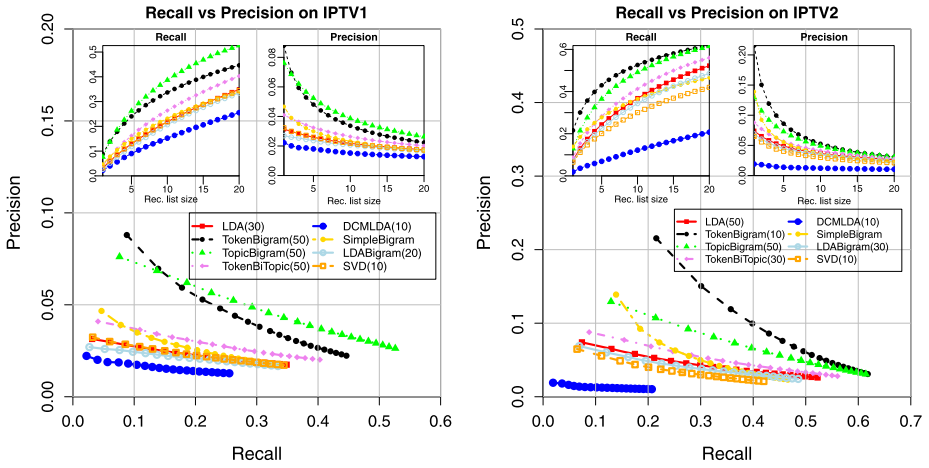


Fig. 4 Recommendation accuracy

aimed at modeling item occurrence rather then an explicit rating. In this respect, non-occurrence of an item has a bivalent interpretation, either as unknown (the user has not considered the item yet), or negative (she does not prefer it at all). Thus, the traditional approaches based on explicit preference (such as Salakhutdinov and Mnih 2007) cannot be applied. We experimented with several specific techniques, including (Hu et al. 2008; Sindhwani et al. 2010) and the standard SVD model. In the following, we report the results of the SVD[6], that still outperforms all the other methods, as a confirmation of the findings in Cremonesi et al. (2010), Barbieri and Manco (2011).

Results Figure 4 summarizes the results in recommendation accuracy achieved over the two considered datasets. For each model, the optimal number of topics is given in brackets.

On both datasets, the proposed models improve the baselines. Concerning IPTV1, both TopicBigram and TokenBigram achieve a significant margin with respect to the other competitors. On IPTV2, TokenBigram outperforms TopicBigram, which is still the runner-up performer.

In summary, the results suggest that:

- The underlying assumption within TokenBiTopic does not involve a remarkable increase of the predictive capabilities of the model. In practice, the topic structure of the TokenBiTopic model can be "simulated" by an LDA model with a quadratic number of topics. As a result, the model seems more prone to overfitting.
- Contextual information, with particular reference to sequence modeling, provides a substantial contribution to recommendation accuracy. This is proven not only by the models proposed in this paper: even the SimpleBigram baseline model achieves remarkable accuracy. In particular, when the recommendation list is relative small, the latter achieves an accuracy comparable to TokenBitopic. As a matter of fact, all the sequential approaches seem to provide a better estimate of the selection probability for the user's next choice.

[6]Based on the SVDLIBC implementation, http://tedlab.mit.edu/~dr/SVDLIBC/. The other matrix factorization methods were obtained from the Graphlab Library, http://graphlab.org/.

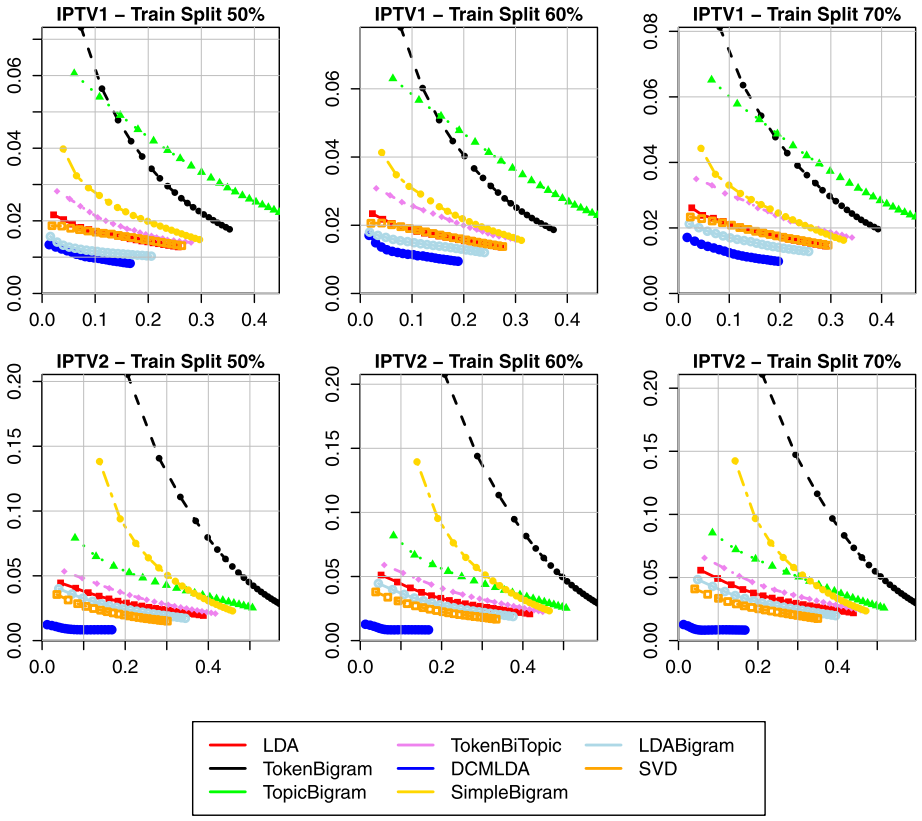


Fig. 5 Precision and Recall for different training splits

- There is a strict correlation between the frequencies exhibited by bigrams and the performance of the TokenBigram model. IPTV2 exhibits more frequent bigrams, and hence it is more likely to boost the performances of the TokenBigram model. By the converse, the TopicBigram exhibits a better capability in generalizing the dependency between the previous hidden context and the next choice. Geometrically, while the TokenBigram model focuses exclusively a restricted area of the topic space, induced by considering only the previous item, the TopicBigram model is actually able to identify larger homogeneous region within the topic space and to estimate the connections (transition probabilities) between them.
- Among the competitors, DCMLDA is rather weak. This is somehow surprising, considering that DCMLDA exhibits the best perplexity in the previous sets of experiments. A viable explanation of this dichotomy can be found in the nature of the sequential data explored here, which does not necessarily support burstiness: notably in a movie rental scenario, once a movie is rented by a user, it is unlikely that it is rented again in the future.
- LDABigram does not provide a substantial improvement either. Again, this is unexpected, in some sense, as bigrams can be considered contextual information as well. It seems that, when bigrams are introduced without an ordering relationship, the resulting ranking function is weakened.

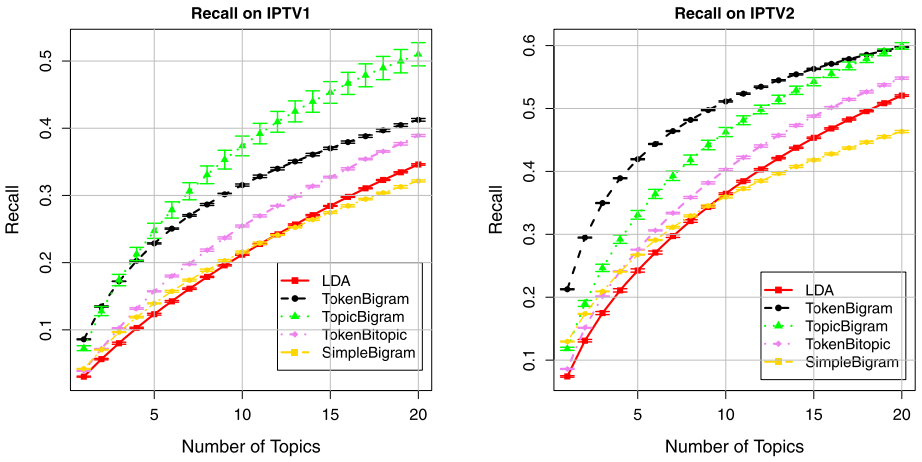


Fig. 6 Recall on random selections of users

In order to analyze the stability of the results, we perform some further experiments. First, we analyze the robustness of the previous experiment with regards to different training/test splits. Figure 5 shows the precision/recall results on three further batches where each user sequence is splitted respectively to 50 %, 60 % and 70 % of the size. In these plots, both TokenBigram and SimpleBigram tend to provide stable results, especially on IPTV2. All other methods seem to suffer the shrinking of the training partition.

In a second batch of experiments, we are interested in analyzing the robustness of the results with regards to random variations of the datasets. To this purpose, we repeat the above experiment on several random samples of the original dataset, where each sample includes 50 % of the whole user set. Training and test sets for each sample are obtained by splitting each sequence with the standard 80–20 percentages. Figure 6 shows average recall, as well as the intervals of variability. It is worth noticing that the TopicBigram model exhibits the highest variations (especially on IPTV1). Notwithstanding, the performances of Fig. 4 are confirmed, thus witnessing a viable robustness of the proposed methods.

Finally, we confront in Fig. 7 the performance with regards to the number of latent factors, with a recommendation list fixed at size 20. TokenBiTopic expresses a wide range of variability in IPTV1, and tends to improve with an increasing number of topics. The other models are stable, and in general do not show a large variance. On IPTV2, TopicBigram shows a progressive increase. However, the slope is progressively decreasing and hence we can expect a maximum on 50 topics. As for the competitors, SVD degrades as long as the number of latent factors is increased: a clear sign of overfitting (as also well-known from the literature). It is worth noticing that, albeit stabler, other matrix factorization approaches based on regularization (not reported here) are still weaker than SVD.

The results presented above experimentally show the effectiveness of sequential topic models in predicting future user choices. However those models increase significantly the number of parameters to be learned and this implies an increase in the learning time. In Fig. 8 we plot the learning time (5000 Gibbs Sampling iterations) for different numbers of topics. Again, TopicBigram exhibits a quadratic behavior, due to the Markovian dependency among topic.

The last two plots in Fig. 8 highlight the contribution of asymmetric priors in the learning process. As expected, asymmetric priors significantly improve the accuracy. However, the

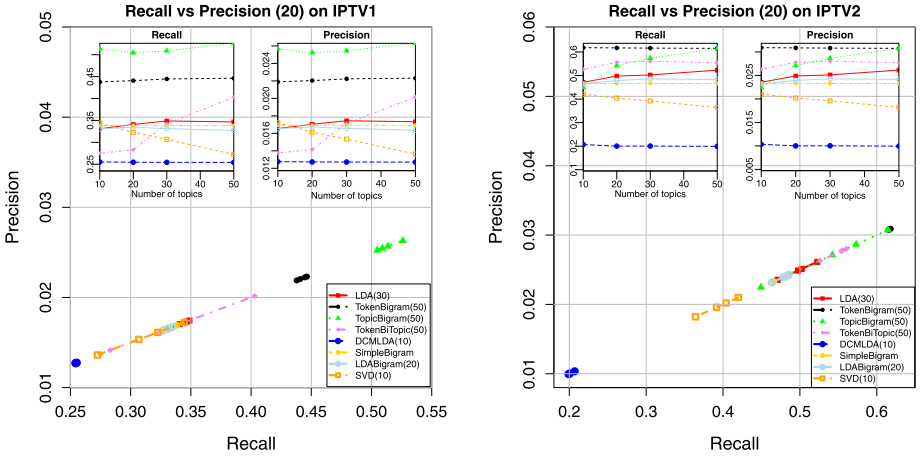


Fig. 7 Recall(20) and Precision(20) of the considered approaches varying the number of topics

learning time is greatly affected, as learning these parameters requires a further iterative fix point procedure to embed in the main algorithm, as explained in Sect. 2.2.

5 Related work

The generative process, which is common to many extensions of the Latent Dirichlet Allocation (Blei 2011), is strongly based on a "bag-of-words" assumption. Even if this assumption may sound unrealistic, this modeling works really good in practice. Latent Dirichlet Allocation and similar models combine the structure-discovery power of dimensionality reduction approaches, such as the *latent semantic indexing* (Deerwester 1988), with informative priors modeling, which are estimated by Bayesian inference techniques. The definition of the topic space and of the projection of each document into this space, provide an effective tool to infer the *semantic concept* of each document, or generally entity. In particular, these approaches support 3 main tasks (Griffiths et al. 2007): topic extraction, word sense disambiguation and prediction.

Among all the different contexts in which these approaches have achieved significant results, in this paper we consider the application of probabilistic topic models to the recommendation problem (Hofmann 2004). As mentioned above, this choice is motivated by some interesting recent findings (Barbieri and Manco 2011) which can be summarized as follows: (i) the item-selection probability computed for each user is a key component for generating accurate item-ranking functions; (ii) among all competitors, LDA provides the best results measured in precision and recall of the recommendation list. These promising results motivate us in exploring extensions of topic models which may provide better representation of the inherent sequential correlation between items, and thus provide better performances in predictions. In the following, we are going to briefly review state-of-the-art probabilistic approaches to sequence data modeling, mainly focusing on topic approaches.

A simple approach to model sequential data within a probabilistic framework has been proposed in Cadez et al. (2000). In their work, the authors present a framework based on mixtures of Markov models for clustering and modeling of web site navigation logs, which

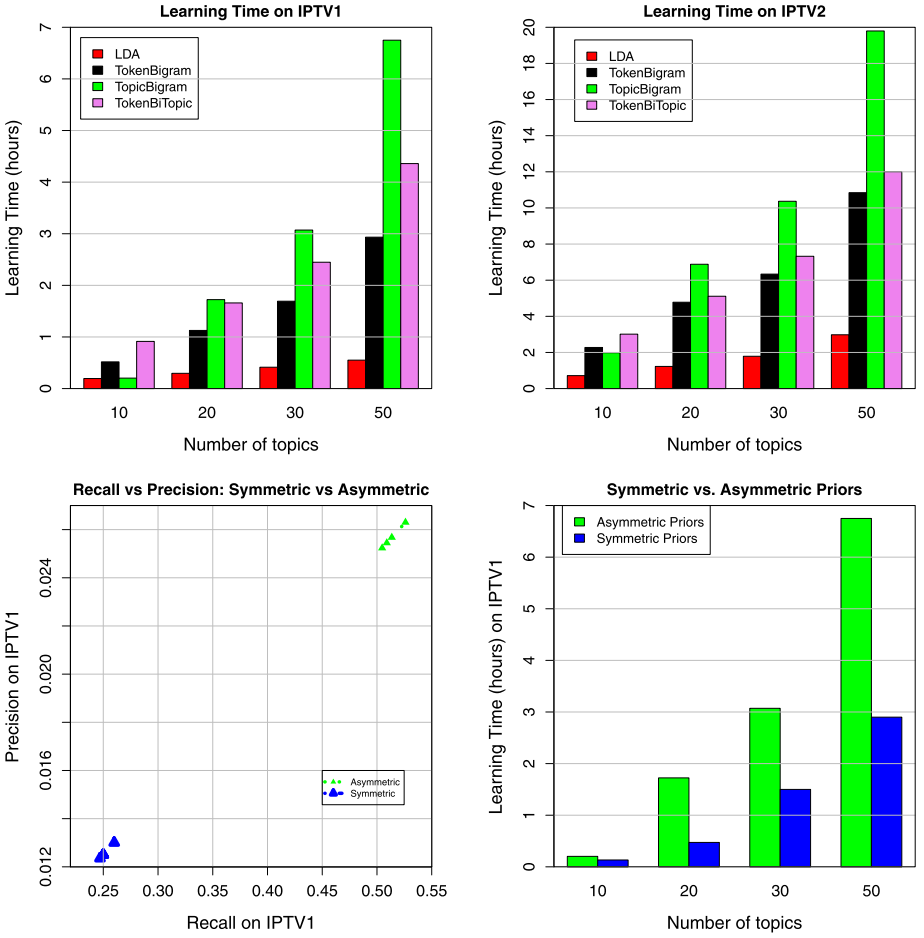
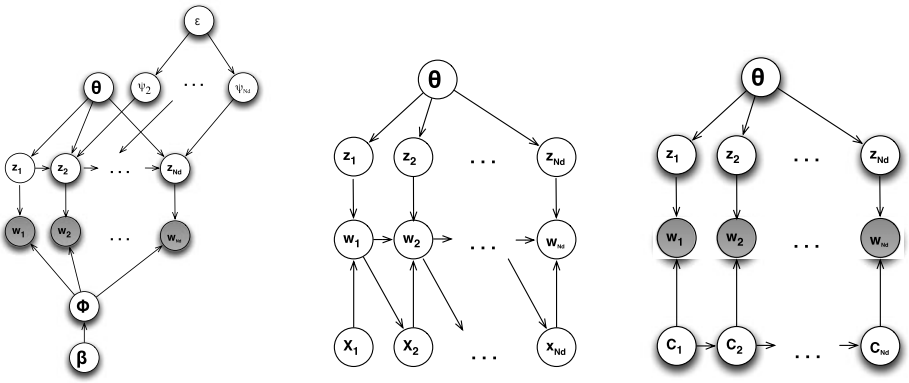


Fig. 8 Learning time of the models on IPTV1 and IPTV2 (*first row*); influence of the hyper parameters (*second row*)

is applied for clustering and visualizing user behavior on a web site. Albeit simple, the proposed model suffers from the limitation that a single latent topic underlies all the observation in a single sequence. This approach has been overtaken by other methods based on latent semantic indexing and LDA. In Wallach (2006), Wang and Wei (2007), for example, the authors propose extension of the LDA model which assume a first-order Markov chain for the word generation process. In the resulting *Token-Bigram Model* (see Sect. 2) and *Topical n-grams*, the current word depends on the current topic and the previous word observed in the sequence.

The N-gram modeling can be extended by considering different kind of dependencies between the hidden states of the model. These kind of dependencies are formalized by exploiting *Hidden Markov models* (HMM) (Bishop 2006, Chap. 13), which are a general reference framework both for modeling sequence data and for natural language processing (Manning and Schütze 1999). HMMs assume that sequential data are generated using a Markov chain of latent variables, with each observation conditioned on the state of the corresponding la-



(a) HTTM (b) Collocation (c) Composite

Fig. 9 HTMM, Collocation and Composite Graphical Model for the generation of a document

tent variable. The resulting likelihood can be interpreted as an extension of a mixture model in which the choice of mixture components for each observation is not selected independently but depends on the choice of components for the previous observation. In Gruber et al. (2007), authors explore this direction, and propose an *Hidden Topic Markov Model (HTMM)* for text documents. HTMM defines a Markov chain over latent topics of the document. The corresponding generative process, depicted in Fig. 9(a), assumes that all words in the same sentence share the same topic, while successive sentences can either rely on the previous topic, or introduce a new one. The topics in a document form a Markov chain with a transition probability that depends on a binary topic transition variable ψ. When $\psi = 1$, a new topic is drawn for the n-th sentence, otherwise the same previous topic is used.

The *LDA Collocation Model* (Griffiths et al. 2007) introduces a new set of random variables (for bigram status) which denotes whether a bigram can be formed with the previous word token. More specifically, as represented in Fig. 9(b), the generative process specifies for each word both a topic and a collocation status. The collocation status adds a more flexible modeling than Token Bigram model which always generates bigrams and, according to this formulation, the distribution on bigram does not depend on the topic. The introduction of the collocation status enrich the generative semantic of the model and this idea can be applied to all the approaches proposed in Sect. 2.

All the previously discussed models approach the problem of sequence modeling by inferring the underlying latent topic and then generate a sequence of words according to this distribution. This perspective does not take into account the fact that words in a text document may exhibit both syntactical and semantic correlations. A *Composite Model*, which captures both semantic and syntactic roles, has been proposed in Griffiths et al. (2005). The graphical model for the generation of a document, given in Fig. 9(c), clarify this concept. The semantic/syntactic dependencies among words are modeled by employing two different latent variables, namely Z and C; while the semantic layer follows a simple LDA model, the syntactic one is instantiated by modeling transitions between the set of classes C through a hidden Markov model. One of these classes corresponds to the semantic class and, when is observed, enables the generation of the word according the current topic. Other classes capture word co-occurrences that are due to syntactic aspects of the modeled language.

Textual documents exhibit a natural sequential structure: people develop documents by building upon a main semantic concept, and by interleaving several segments/subsections,

which express related topics, in a *coherent logical flow*. As described above, *HTMM* models topic cohesion at the level of phrases (words within the same sentence share the same latent topic), but does not model directly a smooth evolution between topics in different segments that frame a document. *Sequential LDA* (Du et al. 2010) is a variant of LDA which models a sequential dependency between sub-topics: the topic of the current segment is closely related to the topic of its antecedent and subsequent segments. This smooth evolution of the topic flow is modeled by using a Poisson-Dirichlet process.

The sequential structure is not limited exclusively to words, but it can affect also sentiments. *Dependency-Sentiment-LDA* (Li et al. 2010) builds on the assumption that sentiments are expressed in a coherent way. Conjunctive words, such as "and" or "but", can be used to detect sentiment transitions, and the sentiment of a word is dependent on the sentiment of its previous one.

6 Conclusion and future work

In this paper we studied three extensions of the LDA model which relax the bag-of-word assumption by hypothesizing that the current observation depends on previous information. For each of the proposed model we provided a Gibbs Sampling parameter estimation procedure and an experimental evaluation was accomplished by studying the models both from a model fitting and an applicative perspective. In particular, the proposed models provide a better framework for modeling contextual information in a recommendation scenario, when the data exhibit intrinsic temporal dependency.

We believe that the models and results presented in this paper open two interesting research directions. On the one side, it would be interesting to generalize the notion of "contextual information": in this paper, a context was represented by temporal dependency. However, there are other observable features that can contribute in the likelihood of observing an item in a user's trace, such as geographical location, tags etc.

Even further, the interaction of a user in a social network is having an increasing impact in her behavior. Analyzing the influence of the neighbors in a network (Barbieri et al. 2013) can help to better evaluate both the temporal dependencies and the likelihood of an item to be selected.

Acknowledgements We would like to thank Charles Elkan for kindly providing the Matlab code for the DCMLDA model.

References

Bambini, R., Cremonesi, P., & Turrin, R. (2011). A recommender system for an IPTV service provider: a real large-scale production environment. In F. Ricci, L. Rokach, B. Shapira, & P. Kantor (Eds.), *Recommender systems handbook* (pp. 299–331). Berlin: Springer.

Barbieri, N., Bonchi, F., & Manco, G. (2013). Cascade-based community detection. In *Sixth ACM international conference on web search and data mining (WSDM'2013)* (pp. 33–42).

Barbieri, N., Costa, G., Manco, G., & Ortale, R. (2011). Modeling item selection and relevance for accurate recommendations: a Bayesian approach. In *Proceedings of the 5th ACM conference on recommender systems (RecSys'11)* (pp. 21–28).

Barbieri, N., & Manco, G. (2011). An analysis of probabilistic methods for top-n recommendation in collaborative filtering. In *Proceedings of the European conference on machine learning and knowledge discovery in databases (ECML-PKDD'11)* (pp. 172–187).

Barbieri, N., Manco, G., Ortale, R., & Ritacco, E. (2012). Balancing prediction and recommendation accuracy: hierarchical latent factors for preference data. In *Proceedings of the 12th SIAM international conference on data mining (SDM'12)*.

Bishop, C. (2006). *Pattern recognition and machine learning*. New York: Springer.
Blei, D. M. (2011). Introduction to probabilistic topic models. *Communications of the ACM*.
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.
Cadez, I., Heckerman, D., Meek, C., Smyth, P., & White, S. (2000). Visualization of navigation patterns on a web site using model-based clustering. In *Proceedings of the 6th ACM SIGKDD international conference on knowledge discovery and data mining (KDD'00)* (pp. 280–284).
Cremonesi, P., Koren, Y., & Turrin, R. (2010). Performance of recommender algorithms on top-*n* recommendation tasks. In *Proceedings of the 4th ACM conference on recommender systems (RecSys'10)* (pp. 39–46).
Cremonesi, P., & Turrin, R. (2009). Analysis of cold-start recommendations in IPTV systems. In *Proceedings of the 3rd ACM conference on recommender systems (RecSys'09)* (pp. 233–236).
Deerwester, S. (1988). Improving information retrieval with latent semantic indexing. In C. L. Borgman & E. Y. H. Pai (Eds.), *Proceedings of the 51st ASIS annual meeting (ASIS '88)* (Vol. 25).
Doyle, G., & Elkan, C. (2009). Accounting for burstiness in topic models. In *Proceedings of the 26th international conference on machine learning (ICML'09)* (p. 36).
Du, L., Buntine, W. L., & Jin, H. (2010). Sequential latent Dirichlet allocation: discover underlying topic structures within a document. In *Proceedings of the 10th IEEE international conference on data mining (ICDM'10)* (pp. 148–157).
Griffiths, T., Steyvers, M., Blei, D., & Tenenbaum, J. (2005). Integrating topics and syntax. In *Advances in neural information processing systems (NIPS'05)*.
Griffiths, T. L., Steyvers, M., & Tenenbaum, J. B. (2007). Topics in semantic representation. *Psychological Review*, *114*.
Gruber, A., Weiss, Y., & Rosen-Zvi, M. (2007). Hidden topic Markov models. *Journal of Machine Learning Research*, *2*, 162–170.
Heinrich, G. (2008). Parameter estimation for text analysis (Tech. rep.). University of Leipzig.
Hofmann, T. (2004). Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, *22*(1), 89–115.
Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In *Proceedings of the 8th IEEE international conference on data mining (ICDM'08)* (pp. 263–272).
Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer*, *42*(8), 30–37.
Li, F., Huang, M., & Zhu, X. (2010). Sentiment analysis with global topics and local dependency. In *Proceedings of the 24th AAAI conference on artificial intelligence (AAAI'10)*.
Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge: MIT Press.
Menon, A., & Elkan, C. (2010). Predicting labels for dyadic data. *Data Mining and Knowledge Discovery*, *21*(2), 327–343.
Menon, A., & Elkan, C. (2011). Link prediction via matrix factorization. In *Proceedings of the European conference on machine learning and knowledge discovery in databases (ECML-PKDD'11)* (pp. 437–452).
Minka, T. P. (2000). Estimating a Dirichlet distribution (Tech. rep.). Microsoft Research. http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/minka-dirichlet.pdf.
Salakhutdinov, R., & Mnih, A. (2007). Probabilistic matrix factorization. In *Proceedings of the 21st annual conference on neural information processing systems (NIPS'07)*.
Sindhwani, V., Bucak, S., Hu, J., & Mojsilovic, A. (2010). One-class matrix completion with low-density factorizations. In *Proceedings of the 10th IEEE international conference on data mining (ICDM'10)* (pp. 1055–1060).
Wallach, H., Mimno, D., & McCallum, A. (2009a). Rethinking lda: why priors matter. In *Advances in neural information processing systems (NIPS'09)* (pp. 1973–1981).
Wallach, H., Murray, I., Salakhutdinov, R., & Mimno, D. (2009b). Evaluation methods for topic models. In *Proceedings of the 26th international conference on machine learning (ICML'09)*.
Wallach, H. M. (2006). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on machine learning (ICML'06)* (pp. 977–984).
Wang, X. A. M., & Wei, X. (2007). Topical n-grams: phrase and topic discovery, with an application to information retrieval. In *Proceedings of the 7th IEEE international conference on data mining (ICDM'07)* (pp. 697–702).