Available online at www.sciencedirect.com
ScienceDirect
Computer Speech and Language 25 (2011) 440–461
COMPUTER SPEECH AND LANGUAGE
www.elsevier.com/locate/csl

Topic tracking language model for speech recognition

Shinji Watanabe [a,*], Tomoharu Iwata [a], Takaaki Hori [a], Atsushi Sako [b,1], Yasuo Ariki [b]

[a] *NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation, 2–4, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan*
[b] *Graduate School of Engineering, Kobe University, 1-1, Rokkodai-cho, Nada-ku, Kobe 657-8501, Japan*

Received 28 December 2009; received in revised form 2 June 2010; accepted 8 July 2010
Available online 30 July 2010

Abstract

In a real environment, acoustic and language features often vary depending on the speakers, speaking styles and topic changes. To accommodate these changes, speech recognition approaches that include the incremental tracking of changing environments have attracted attention. This paper proposes a topic tracking language model that can adaptively track changes in topics based on current text information and previously estimated topic models in an on-line manner. The proposed model is applied to language model adaptation in speech recognition. We use the MIT OpenCourseWare corpus and Corpus of Spontaneous Japanese in speech recognition experiments, and show the effectiveness of the proposed method.

Language model; Latent topic model; Topic tracking; On-line algorithm; Speech recognition

1. Introduction

Speech recognition is a promising technique for automatically transcribing broadcast news, multimedia archives on the web, meetings, and lecture recordings for information retrieval (e.g., Makhoul et al. (2000) for broadcast news and Glass et al. (2007); Hori et al. (2009) for lectures). In these scenarios, speech includes temporal variations caused by changes of speakers, speaking styles, environmental noises, and topics. Thus, speech recognition models have to track temporal changes in both acoustic and language environments. This paper focuses on tracking temporal changes in language environments, as shown in Fig. 1. Fig. 1 depicts temporal change tracking by using a dynamic language model of a lecture.

The study of dynamic language models beyond *N*-gram deals with the temporal changes in language environments, which is a main theme of language model research (see Rosenfeld, 2000; Bellegarda, 2004 in detail). The dynamic language models are mainly classified into two types, i.e., those that estimate word (*N*-gram) probabilities *directly* and *indirectly*. The cache-based language model (Kuhn and De Mori, 1990) is representative of the direct estimation approaches. This model uses an *N*-gram probability obtained from a cache text (e.g., thousands of words in a text history), in addition to a normal (static) *N*-gram probability. The new *N*-gram probability is obtained by linearly interpolating the two probabilities. The other techniques employed in the direct estimation approaches are based on the maximum a posteriori (MAP) criterion. Then, the *N*-gram probability is obtained by the *N*-gram count, which is linearly

* Corresponding author.
E-mail address: watanabe@cslab.kecl.ntt.co.jp (S. Watanabe).
[1] Dr. Atsushi Sako is currently at Nintendo Co., Ltd.

0885-2308/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.csl.2010.07.006

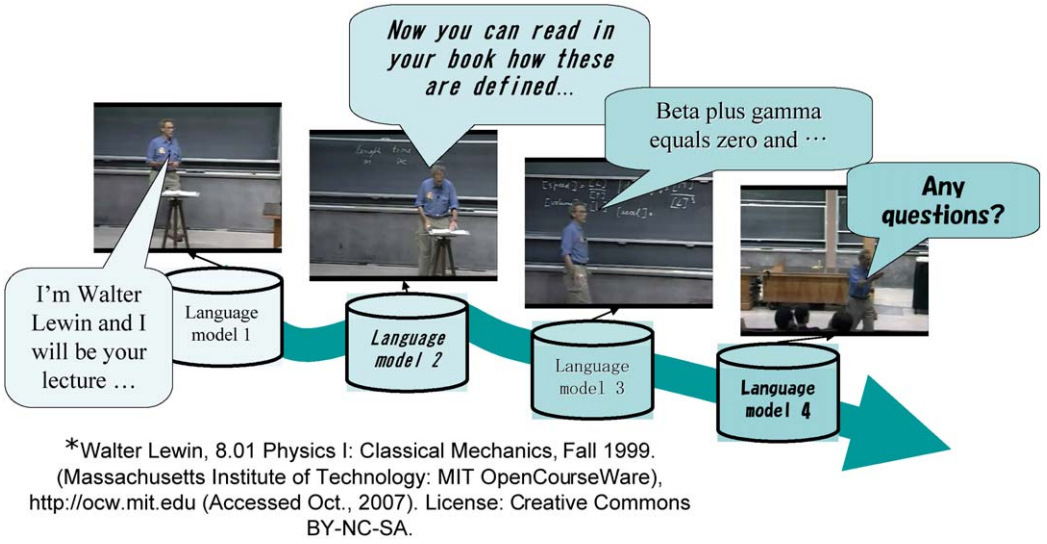

Fig. 1. Tracking temporal changes in language environments.

interpolating the two *N*-gram *counts*, unlike the *probability* based interpolation in the cache approach (Federico, 1996; Masataki et al., 1997).

The indirect estimation approaches mainly focus on mixture weight estimation where each mixture component is represented by a *topic* dependent word probability and each mixture weight corresponds to a topic proportion probability. The mixture models in the context of the language models are obtained by clustering articles (Iyer and Ostendorf, 1996) or by applying the well-known (Probabilistic) Latent Semantic Analysis (LSA, PLSA) to the language model (Bellegarda, 2000; Gildea and Hofmann, 1999). Then, the number of free parameters in the indirect estimation approaches corresponds to the number of topics, and is often fewer than that in the direct estimation approaches, where the number of free parameters corresponds to the vocabulary size. This property is effective especially for an on-line adaptation strategy for language models that mitigates over-training problems, and therefore this paper focuses on tracking temporal changes within the indirect estimation approaches.

LSA was originally formulated in the context of information retrieval (Deerwester et al., 1990) in the natural language processing field, and is extended to a probabilistic framework based on PLSA, and Latent Dirichlet Allocation (LDA) (Hofmann, 1999; Blei et al., 2003). PLSA and LDA assume that samples in a set of sentences (e.g., documents in text processing and chunks (sets of utterances) in speech processing.[1]) are exchangeable, and therefore they cannot deal with topic dynamics where samples are regarded as a time series, and have a time order. There are several approaches for extending PLSA and LDA to deal with topic dynamics in the context of information retrieval. Some of the approaches consider the sudden (discrete) topic changes caused by scene switching. These are modeled by the state transition from topic to topic (Griffiths et al., 2005; Gruber et al., 2007; Chen et al., 2009).[2] On the other hand, there is a different type of topic changes, where the topics are changed gradually by maintaining the topic continuity between several utterances. This dynamics can be modeled in terms of the time evolution of topics, which proceeds smoothly when the past and current topic models are interpolated, and again the original PLSA and LDA can not deal with the dynamics, either. We have considered such topic changes, and proposed the topic tracking model (TTM) (Iwata et al., 2009) for web data mining by extending the LDA to involve time-dependent hyper-parameters in the word and topic probabilities.

The TTM extends PLSA and LDA to track changes smoothly across chunks by establishing a dynamics between the previous and current topic model parameters. This establishment of the dynamics is often used in the Kalman filter based approach. The standard Kalman filter approach simply assumes a continuous value as a system output, which is modeled by a Gaussian distribution. This assumption is not suitable for language modeling since the language model

[1] This paper uses discrete values (e.g., document, chunk, and utterance) for time. For reference, dynamic topic models with continuous time have also been discussed in Wang et al. (2008) recently.

[2] These are applied to speech recognition (Hsu and Glass, 2006; Sako et al., 2008).

regards a discrete value (word count) as a system output. To solve the Kalman filter equation for a discrete value case, dynamic topic model (Blei and Lafferty, 2006) uses a softmax activation function to convert discrete values to continuous values and a model is estimated by using variational Bayes. On the other hand, the TTM and dynamic mixture model (Wei et al., 2007) consistently employ discrete values by using multinomial distributions and their conjugate distributions (Dirichlet distributions) in topic dynamics. Therefore, they can obtain a simple update equation of a topic probability in an on-line manner. In addition, the TTM can estimate the topic probability and the word (unigram) probability at the same time, unlike the dynamic mixture model. Furthermore, these probabilities depend on the long-range history by considering the past several terms in the history, unlike the dynamic mixture model and dynamic topic model, which consider only the previous term. Then, the TTM estimates the precision parameter of the topic probability for each past term, where the parameter corresponds to the degree of contribution of data in a certain past term of the long-range history. Therefore, the TTM can deal with long-range history by considering the importance of the data in each past term, which is powerful in practice. This estimation process is performed with a stochastic EM algorithm.

This paper considers that the topic changes discussed above are also important in speech recognition, and proposes a language model adaptation technique, the topic tracking language model (TTLM), that tracks the topic changes in speech by modifying the TTM. The TTLM is basically formulated as a TTM for on-line topic and word (unigram) probability extraction in Section 2. By integrating the model into an *N*-gram language model and further integrating it with the unlabeled (unsupervised) incremental adaptation of speech recognition in Section 3, the TTLM enables us to realize language model adaptation with topic tracking. We also discuss how to set the size of incremental adaptation step in speech recognition, since it is difficult to provide document or web page units for speech, unlike information retrieval. To show the effectiveness of the TTLM, we performed speech recognition experiments using lectures provided by the MIT OpenCourseWare corpus (MIT-OCW, Glass et al., 2007) and oral presentations provided by the Corpus of Spontaneous Japanese (CSJ, Furui et al., 2000). The experiments involved the unlabeled (unsupervised) incremental adaptation of language models for these talks.

2. Topic tracking language model

This section describes the topic tracking language model (TTLM) by considering the application of the topic tracking model (TTM) to speech recognition. First, Section 2.1 provides a general description of a latent topic model, and extends this model by establishing topic dynamics in Section 2.2. Then, Section 2.3 introduces the estimation of TTLM parameters and Section 2.4 considers the long-term dependences of TTLM. Section 2.5 discusses an interpolation of the latent topic model and topic-independent word probability in the TTLM framework. The mathematical notations used in this paper are summarized in Table 1.

2.1. Latent topic model

As we begin the formulation, we first define a chunk unit as a subsequence of a long word sequence concatenating all the word sequences in a speech transcription corpus. Spontaneous speech such as lecture and oral presentation does not have an explicit chunk size unit unlike a text corpus (e.g., an article unit in newspapers and a page unit on the web). In general, it is very difficult to set a chunk unit for talks, and this paper regards a set of adjacent utterances or sentences as a chunk, and uses chunk index t as a unit of time evolution in the following formulation and experiments.

A latent topic model considers the word sequence $\mathbf{W} = \{w_1, \ldots, w_m, \ldots\}$ and the corresponding latent topic sequence $\mathbf{Z} = \{z_1, \ldots, z_m, \ldots\}$ as follows:

$$\begin{aligned} \mathbf{W} &= \{\underbrace{w_1, \ldots w_{M_1}}_{\mathbf{W}_{t=1}}, \ldots, \underbrace{w_{M_{t-1}+1}, \ldots, w_{M_t}}_{\mathbf{W}_t}, \ldots\}, \\ \mathbf{Z} &= \{\underbrace{z_1, \ldots z_{M_1}}_{\mathbf{Z}_{t=1}}, \ldots, \underbrace{z_{M_{t-1}+1}, \ldots, z_{M_t}}_{\mathbf{Z}_t}, \ldots\}, \end{aligned} \tag{1}$$

where w_m and z_m indicate the mth word and latent topic, respectively, and they are hierarchically represented by subset sequences of $\mathbf{W}_t$ and $\mathbf{Z}_t$ at chunk t, which is a longer time unit than a word. M_t denotes the sequential serial number of the last word at chunk t. Then PLSA assumes that the unigram probability of w_m at chunk t is decomposed into topic

Table 1
Notation list.

w_m	:	mth word
z_m	:	mth corresponding latent topic
t	:	Chunk index
$\mathbf{W}_t$	:	Word sequence at chunk t
$\mathbf{Z}_t$	:	Latent topic sequence at chunk t
M_t	:	Sequential serial number of last word at chunk t
l	:	Word index
k	:	Topic index
L	:	Number of words
K	:	Number of latent topics
ϕ	:	Topic probability
θ	:	Word probability
$\boldsymbol{\Phi}$	:	Set of topic probability parameters
$\boldsymbol{\Theta}$	:	Set of word probability parameters
γ	:	Dirichlet hyper-parameters of topic probability in LDA
$\hat{\phi}$	:	Mean parameter of topic probability
$\hat{\theta}$	:	Mean parameter of word probability
α	:	Precision parameter of topic probability
β	:	Precision parameter of word probability

and word probabilities as follows:

$$\begin{aligned} \underbrace{P(w_m|t)}_{\text{Unigram probability}} &= \sum_{k=1}^{K} \underbrace{P(k|t)}_{\text{Topic probability}} \underbrace{P(w_m|k,t)}_{\text{Word probability}} \\ &\triangleq \sum_{k=1}^{K} \phi_{tk}\theta_{tkw_m}, \end{aligned} \tag{2}$$

where k is a topic index (i.e. $z_m = k$) and K is the number of topics. Topic probability ϕ_{tk} means a probability where topic k exists at chunk t with $\phi_{tk} \geq 0$ and $\sum_k \phi_{tk} = 1$. When the mth word has an index l, i.e., $w_m = l$, word probability θ_{tkl} means a probability where word l exists in topic k at chunk t with $\theta_{tkl} \geq 0$ and $\sum_l \theta_{tkl} = 1$. The joint distribution of data $\mathbf{W}$ and latent topics $\mathbf{Z}$ can be represented with a set of topic probability parameters $\boldsymbol{\Phi}$ and a set of word probability parameters $\boldsymbol{\Theta}$ as follows:

$$\begin{aligned} P(\mathbf{W}, \mathbf{Z}|\boldsymbol{\Phi}, \boldsymbol{\Theta}) &= \prod_t P(\mathbf{W}_t, \mathbf{Z}_t|\boldsymbol{\phi}_t, \boldsymbol{\Theta}_t) \\ &= \prod_t \prod_{m=M_{t-1}+1}^{M_t} \phi_{tz_m}\theta_{tz_m w_m}, \end{aligned} \tag{3}$$

where we assume that each joint distribution at a chunk is independent and identically distributed, conditional on parameters $\boldsymbol{\phi}_t = \{\phi_{tk}\}_{k=1}^K$ and $\boldsymbol{\Theta}_t = \{\{\theta_{tkl}\}_{l=1}^L\}_{k=1}^K$ at chunk t. L is vocabulary size. Based on this joint distribution, the topic and word probability parameters ($\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$) can be estimated by using the EM algorithm to maximize $\sum_{\mathbf{Z}} P(\mathbf{W}, \mathbf{Z}|\boldsymbol{\Phi}, \boldsymbol{\Theta})$ (Hofmann, 1999).

In addition to Eqs. (2) and (3), LDA considers the prior distribution of topic probability $\boldsymbol{\phi}_t$, which is represented by the Dirichlet distribution with hyper-parameter $\boldsymbol{\gamma} = \{\gamma_k\}_{k=1}^K$ as follows:

$$P(\boldsymbol{\phi}_t|\boldsymbol{\gamma}) \propto \prod_{k=1}^{K} \phi_{tk}^{\gamma_k - 1}. \tag{4}$$

The joint distribution ($P(\mathbf{W}_t, \mathbf{Z}_t|\boldsymbol{\Theta}_t, \gamma)$) is computed by marginalizing Eq. (3) and this prior distribution with respect to $\boldsymbol{\phi}_t$ (Blei et al., 2003). This equation shows that topic probability $\boldsymbol{\phi}_t$ is independent of other chunks, and does not exhibit any explicit dynamics.

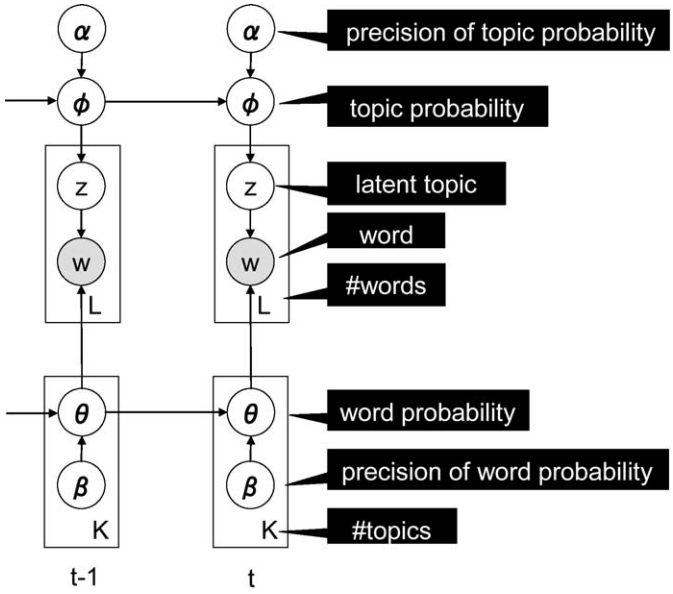

Fig. 2. Graphical representation of topic tracking language model (TTLM).

2.2. Topic tracking language model (TTLM)

To model the dynamics of topic probability $\boldsymbol{\phi}_t$, the TTLM assumes that the mean of the topic probabilities at the current chunk are the same as those at a previous chunk unless otherwise indicated by the newly observed data. In particular, we use the following Dirichlet distribution, in which the mean of the current topic probabilities is the same as the mean of the previous probabilities $\hat{\phi}_{t-1k}$ and the precision is α_t.

$$P(\boldsymbol{\phi}_t|\hat{\boldsymbol{\phi}}_{t-1}, \alpha_t) \propto \prod_{k=1}^{K} \phi_{tk}^{\alpha_t \hat{\phi}_{t-1k}-1}. \tag{5}$$

We use the Dirichlet distribution as a conjugate distribution, which simplifies its parameter estimation. Precision α_t provides a degree of unchangeability thus making it possible to follow the temporal changes of topics flexibly. Similar to the topic probability, the TTLM can also focus on the following prior distributions of word probabilities $\boldsymbol{\theta}_{tk} = \{\theta_{tkl}\}_{l=1}^{L}$ for word index l with vocabulary size L,

$$P(\boldsymbol{\theta}_{tk}|\hat{\boldsymbol{\theta}}_{t-1k}, \beta_{tk}) \propto \prod_{l=1}^{L} \theta_{tkl}^{\beta_{tk}\hat{\theta}_{t-1kl}-1}, \tag{6}$$

where β_{tk} is the precision of the word dynamics probability.

The generative process of the TTLM is as follows:

1. Draw $\boldsymbol{\phi}_t$ from Dirichlet($\alpha_t \hat{\boldsymbol{\phi}}_{t-1}$)
2. For each topic $k = 1, \ldots, K$:
   (a) Draw $\boldsymbol{\theta}_{tk}$ from Dirichlet($\beta_{tk} \hat{\boldsymbol{\theta}}_{t-1k}$)
3. For each word in chunk t ($m = M_{t-1} + 1, \ldots, M_t$):
   (a) Draw z_m from Multinomial($\boldsymbol{\phi}_t$)
   (b) Draw w_m from Multinomial($\boldsymbol{\theta}_{tz_m}$)

Fig. 2 is a graphical representation of the TTLM, where shaded and unshaded nodes indicate observed and latent variables, respectively.

2.3. Inference

We estimate the TTLM parameters based on a stochastic EM algorithm, which alternately iterates (1) the Gibbs sampling of latent topics and (2) the maximum joint likelihood estimation with respect to the precision parameters (α_t and $\boldsymbol{\beta}_t = \{\beta_{tk}\}_{k=1}^{K}$) (Wallach, 2006).

2.3.1. Gibbs sampling of latent topics

We infer latent topics based on collapsed Gibbs sampling (Griffiths and Steyvers, 2004), which requires the joint distribution of data and latent topics of the precision parameters ($P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t)$). This joint distribution is represented by marginalizing the joint distribution of data and latent topics conditional on topic and word probability parameters (Eq. (3)) and their prior distributions (Eqs. (5) and (6)) as follows:

$$P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = \iint P(\mathbf{W}_t, \mathbf{Z}_t|\boldsymbol{\phi}_t, \boldsymbol{\Theta}_t) \times P(\boldsymbol{\phi}_t|\hat{\boldsymbol{\phi}}_{t-1}, \alpha_t) P(\boldsymbol{\Theta}_t|\hat{\boldsymbol{\Theta}}_{t-1}, \boldsymbol{\beta}_t) d\boldsymbol{\phi}_t d\boldsymbol{\Theta}_t. \tag{7}$$

Here, we assume the following independence,

$$P(\boldsymbol{\phi}_t, \boldsymbol{\Theta}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = P(\boldsymbol{\phi}_t|\hat{\boldsymbol{\phi}}_{t-1}, \alpha_t) P(\boldsymbol{\Theta}_t|\hat{\boldsymbol{\Theta}}_{t-1}, \boldsymbol{\beta}_t). \tag{8}$$

This assumption means that the dynamics of the word and topic probabilities are independent of each other, as described in the previous section.

Since we use conjugate priors for parameters $\boldsymbol{\phi}_t$ and $\boldsymbol{\Theta}_t$, we can integrate out these parameters in the joint distribution by substituting Eqs. (3), (5), and (6) into Eq. (7) as follows (See Appendix A.1):

$$\begin{aligned} P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = {} & \frac{\Gamma(\alpha_t)}{\prod_k \Gamma(\alpha_t \hat{\phi}_{t-1k})} \frac{\prod_k \Gamma(n_{tk} + \alpha_t \hat{\phi}_{t-1k})}{\Gamma(n_t + \alpha_t)} \\ & \times \prod_k \frac{\Gamma(\beta_{tk})}{\prod_l \Gamma(\beta_{tk} \hat{\theta}_{t-1kl})} \frac{\prod_l \Gamma(n_{tkl} + \beta_{tk} \hat{\theta}_{t-1kl})}{\Gamma(n_{tk} + \beta_{tk})}, \end{aligned} \tag{9}$$

where $\Gamma(x)$ is the gamma function. n_t is a count of words at chunk t, n_{tk} is a count of words assigned to topic k at chunk t, and n_{tkl} is a count of word index l assigned to topic k at chunk t. Thus, the joint distribution is represented by the counts of words and hyper-parameters ($\hat{\boldsymbol{\phi}}_{t-1}$, $\hat{\boldsymbol{\Theta}}_{t-1}$, α_t, and $\boldsymbol{\beta}_t$).

From Eq. (9), Gibbs sampling assigns the mth word (w_m) in chunk t to a latent topic (k) by using the following assignment probability (See Appendix A.2):

$$P(z_m = k|\mathbf{W}_t, \mathbf{Z}_{t\backslash m}, \hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) \propto \frac{n_{tk\backslash m} + \alpha_t \hat{\phi}_{t-1k}}{n_{t\backslash m} + \alpha_t} \frac{n_{tkw_m\backslash m} + \beta_{tk} \hat{\theta}_{t-1kw_m}}{n_{tk\backslash m} + \beta_{tk}}. \tag{10}$$

$\mathbf{Z}_{t\backslash m}$ is a set of topics that does not include the mth word. $n_{\cdot\backslash m}$ means a count of words that does not include the mth word. Eq. (10) means that the assignment probability is proportional to the ratios of the word counts ($n_{tk\backslash m}, n_{tkw_m\backslash m}$) where these counts are linearly interpolated by the previously estimated probabilities ($\hat{\boldsymbol{\phi}}_{t-1}$ and $\hat{\boldsymbol{\Theta}}_{t-1}$) and the precision parameters (α_t and $\boldsymbol{\beta}_t$).

2.3.2. Maximum likelihood estimation of joint distribution

Then, precision parameters α_t and $\boldsymbol{\beta}_t$ can be obtained by the maximum likelihood estimation of the joint distribution (Eq. (9)), and precision of topic probability α_t can be estimated by the following update equation (Minka, 2000).

$$\alpha_t \leftarrow \alpha_t \frac{\sum_k \hat{\phi}_{t-1k} \left(\Psi(n_{tk} + \alpha_t \hat{\phi}_{t-1k}) - \Psi(\alpha_t \hat{\phi}_{t-1k}) \right)}{\Psi(n_t + \alpha_t) - \Psi(\alpha_t)}, \tag{11}$$

where Ψ is a digamma function. Similarly, the precision of word probability β_{tk} can also be estimated as follows:

$$\beta_{tk} \leftarrow \beta_{tk} \frac{\sum_k \hat{\theta}_{t-1kl} \left(\Psi(n_{tkl} + \beta_{tk}\hat{\theta}_{t-1kl}) - \Psi(\beta_{tk}\hat{\theta}_{t-1kl}) \right)}{\Psi(n_{tk} + \beta_{tk}) - \Psi(\beta_{tk})}. \tag{12}$$

After the iterative calculation of Eqs. (10)–(12), we can obtain $\mathbf{Z}_t$, α_{tk}, and $\boldsymbol{\beta}_t$, respectively.

From the obtained $\mathbf{Z}_t$, α_{tk}, and $\boldsymbol{\beta}_t$, the means of $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_{tk}$ are obtained as follows:

$$\begin{aligned} \hat{\phi}_{tk} &= \frac{n_{tk} + \alpha_t \hat{\phi}_{t-1k}}{n_t + \alpha_t}, \\ \hat{\theta}_{tkl} &= \frac{n_{tkl} + \beta_{tk}\hat{\theta}_{t-1kl}}{n_{tk} + \beta_{tk}}. \end{aligned} \tag{13}$$

Therefore, the unigram probability at chunk t can be computed by plugging $\hat{\phi}_{tk}$ and $\hat{\theta}_{tkl}$ into ϕ_{tk} and θ_{tkl}, respectively, in Eq. (2). Eq. (13) means that the word and topic probabilities are obtained by the ratios of the word counts (n_{tk} and n_{tkl}) where these counts are linearly interpolated by the previously estimated probabilities ($\hat{\boldsymbol{\phi}}_{t-1}$ and $\hat{\boldsymbol{\Theta}}_{t-1}$) and the precision parameters (α_t and $\boldsymbol{\beta}_t$). The $\hat{\theta}_{tkl}$ result is similar to those of the N-gram estimation approaches based on the maximum a posteriori (MAP) criterion (Federico, 1996; Masataki et al., 1997) since the TTLM and MAP-based approaches are within a Bayesian framework. Therefore, the TTLM can include the MAP-based N-gram estimation approaches, in addition to topic tracking via $\hat{\phi}_{tk}$.

Note that $\hat{\phi}_{tk}$ and $\hat{\theta}_{tkl}$ are used for the hyper-parameters of the prior distributions at the succeeding chunk $t+1$. This on-line algorithm only requires data at chunk t ($\mathbf{W}_t$) and the hyper-parameters at chunk $t-1$ ($\hat{\boldsymbol{\phi}}_{t-1}$ and $\{\hat{\boldsymbol{\theta}}_{t-1k}\}_{k=1}^{K}$) to estimate the parameters at chunk t, which can reduce the required computation time and memory size.

2.4. Topic tracking language model by capturing long term dependences

If we consider the long term dependences in topic dynamics (i.e., S chunks before t), we have to consider the time dependence from the $(t-S)$th chunk to the tth chunk in the TTLM.

Such a long-term TTLM can be modeled as follows, instead of using Eqs. (5) and (6),

$$\begin{aligned} P(\boldsymbol{\phi}_t | \{\hat{\boldsymbol{\phi}}_{t-s}, \alpha_{ts}\}_{s=1}^{S}) &\propto \prod_{k=1}^{K} \phi_{tk}^{(\alpha * \hat{\phi}_k)_t - 1}, \\ P(\boldsymbol{\theta}_{tk} | \{\hat{\boldsymbol{\theta}}_{t-sk}, \beta_{tks}\}_{s=1}^{S}) &\propto \prod_{l=1}^{L} \theta_{tkl}^{(\beta_k * \hat{\theta}_{kl})_t - 1}, \end{aligned} \tag{14}$$

where

$$(f * g)_t \triangleq \sum_{s=1}^{S} f_{ts} g_{t-s}. \tag{15}$$

Here, $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_{tk}$ depend on hyper-parameters $\{\hat{\boldsymbol{\phi}}_{t-s}, \alpha_{ts}\}_{s=1}^{S}$ and $\{\hat{\boldsymbol{\theta}}_{t-sk}, \beta_{tks}\}_{s=1}^{S}$, respectively. Then, the TTLM parameters are obtained in a similar way to that used in Section 2.3. In fact, equations for the Gibbs sampling of latent topics and the maximum likelihood estimation of the joint distribution of the TTLM with long-term dependences can be

obtained by using the following substitution for those in Section 2.3.

$$\begin{cases} \alpha_t \hat{\phi}_{t-1k} \rightarrow (\alpha * \hat{\phi}_k)_t \\ \beta_{tk} \hat{\theta}_{t-1kl} \rightarrow (\beta_k * \hat{\theta}_{kl})_t \\ \alpha_t \rightarrow \sum_{s=1}^{S} \alpha_{ts} \\ \beta_{tk} \rightarrow \sum_{s=1}^{S} \beta_{tks} \end{cases}. \tag{16}$$

Therefore, the remainder of this section only provides the analytical results of the equations of the TTLM with long-term dependences.

2.4.1. Gibbs sampling of latent topics

The assignment probability (Eq. (10)) is rewritten as follows:

$$P(z_m = k | \mathbf{W}_t, \mathbf{Z}_{t\backslash m}, \{\hat{\boldsymbol{\phi}}_{t-s}, \hat{\boldsymbol{\Theta}}_{t-s}, \alpha_{ts}, \boldsymbol{\beta}_{ts}\}_{s=1}^{S}) \propto \frac{n_{tk\backslash m} + (\alpha * \hat{\phi}_k)_t}{n_{t\backslash m} + \sum_s \alpha_{ts}} \frac{n_{tkw_m\backslash m} + (\beta_k * \hat{\theta}_{kw_m})_t}{n_{tk\backslash m} + \sum_s \beta_{tks}}. \tag{17}$$

2.4.2. Maximum likelihood estimation of joint distribution

The update equations of α_t (Eq. (11)) and β_{tk} (Eq. (12)) are rewritten as follows:

$$\begin{aligned} \alpha_{ts} &\leftarrow \alpha_{ts} \frac{\sum_k \hat{\phi}_{t-sk} \left(\Psi(n_{tk} + (\alpha * \hat{\phi}_k)_t) - \Psi((\alpha * \hat{\phi}_k)_t) \right)}{\Psi(n_t + \sum_{s'} \alpha_{ts'}) - \Psi(\sum_{s'} \alpha_{ts'})}, \\ \beta_{tks} &\leftarrow \beta_{tks} \frac{\sum_k \hat{\theta}_{t-skl} \left(\Psi(n_{tkl} + (\beta_k * \hat{\theta}_{kl})_t) - \Psi((\beta_k * \hat{\theta}_{kl})_t) \right)}{\Psi(n_{tk} + \sum_{s'} \beta_{tks'}) - \Psi(\sum_{s'} \beta_{tks'})}. \end{aligned} \tag{18}$$

The means of $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_{tk}$ in Eq. (13) are also rewritten as follows:

$$\begin{aligned} \hat{\phi}_{tk} &= \frac{n_{tk} + (\alpha * \hat{\phi}_k)_t}{n_t + \sum_{s'} \alpha_{ts'}}, \\ \hat{\theta}_{tkl} &= \frac{n_{tkl} + (\beta_k * \hat{\theta}_{kl})_t}{n_{tk} + \sum_{s'} \beta_{tks'}}. \end{aligned} \tag{19}$$

Thus, the long-term TTLM can also model the long-term dynamics in an on-line algorithm by using previous chunks. Similarly, the TTLM can also use current and Fth future chunks by considering the parameter set $\{\hat{\boldsymbol{\phi}}_{t-s}, \hat{\boldsymbol{\Theta}}_{t-s}, \alpha_{ts}, \boldsymbol{\beta}_{ts}\}_{s=-F}^{S}$ from the $(t-S)$th chunk to the $(t+F)$th chunk, and by changing the summation to $\sum_{s=-F}^{S}$ in Eqs. (17)–(19). Although it lacks causality, there are many effective off-line applications in speech recognition that do not require the causality. This extension is also an advantage of the TTLM, since it can be formulated as a Kalman filter for a discrete value, and can naturally use current and future data, as well as past data, which corresponds to the "Kalman smoother" from the analogy with the Kalman filter theory.

2.5. Interpolation of latent topic model and topic-independent word probability

In a practical situation, we sometimes face a problem, namely that unigram probabilities obtained via topic models cannot be appropriately estimated (e.g., due to data sparseness) and this degrades the performance compared with that

of conventional unigram models. To prevent this degradation, we extend a latent topic model from Eq. (2) as follows:

$$\begin{aligned} P(w_m|t) &= \sum_{k=1}^{K} \phi_{tk}\theta_{tkw_m} + \phi_{t0}\theta_{0w_m} \\ &= \sum_{k=0}^{K} \phi_{tk}\theta_{tkw_m}, \end{aligned} \tag{20}$$

where θ_{0w_m} is a topic-independent word probability. This extension means that a new unigram probability is represented by a linear interpolation of topic-based and universal unigram probabilities. Then, the dynamics of topic probability ϕ_t in Eqs. (5) and (15) are respectively extended as follows:

$$P(\boldsymbol{\phi}_t|\hat{\boldsymbol{\phi}}_{t-1}, \alpha_t) \propto \prod_{k=0}^{K} \phi_{tk}^{\alpha_t\hat{\phi}_{t-1k}-1}. \tag{21}$$

and

$$P(\boldsymbol{\phi}_t|\{\hat{\boldsymbol{\phi}}_{t-s}, \alpha_{ts}\}_{s=1}^{S}) \propto \prod_{k=0}^{K} \phi_{tk}^{(\alpha*\hat{\phi}_k)_t-1}. \tag{22}$$

Namely, an additional mixture weight ϕ_{t0} is appended to the original $\boldsymbol{\phi}_t$ (i.e., $\boldsymbol{\phi}_t = \{\phi_{t0}, \phi_{t1}, \ldots, \phi_{tk}, \ldots, \phi_{tK}\}$ in this extension). Note that this extension does not change the estimation process in the inference part of the TTLM framework in Sections 2.3 and 2.4. We can obtain α_t and $\hat{\phi}_{tk}$ by preparing a topic-independent word probability (θ_{0w_m}) and simply considering the $k=0$ component in the estimation process. This kind of interpolation technique is very familiar in language modeling where the interpolation coefficient parameters are estimated under the maximum likelihood EM algorithm on held-out data (e.g., Section 3.1 in Bellegarda (2004)). The key aspect of this interpolation in the proposed approach is that the estimation of the interpolation coefficient parameter ϕ_{t0} is involved in the stochastic EM algorithm in the TTLM. Therefore, we do not have to prepare the held-out data for the maximum likelihood EM algorithm, and the parameter estimation is performed within the TTLM framework, as well as the other topic probability parameters.

Thus, the TTLM is basically formulated as a TTM for on-line topic and word probability extraction in Section 2. The next section introduces the TTLM implementation for the unlabeled (unsupervised) incremental adaptation of language models.

3. Implementation for unlabeled (unsupervised) incremental adaptation of language models

This paper mainly focuses on temporal changes of topics. Therefore, we only consider the dynamics of the topic probabilities ϕ_{tk} while the word probability for each chunk is fixed (i.e., $\theta_{tkl} \approx \theta_{kl}$). In fact, since the word probability in our language model adaptation tasks has a large number of parameters, word probability estimation would cause an over-training problem. θ_{kl} is initially obtained via conventional LDA by using training data. Then, a TTLM for the unlabeled (unsupervised) incremental adaptation of language models is realized by employing the following steps:

(1) A word sequence in chunk t ($\mathbf{W}_t$) is obtained by using a speech recognizer (decoder) with a previously estimated N-gram model ($P(w_m|w_{m-1}^{m-N-1}, t-1)$).
(2) The TTLM updates a current unigram model ($P(w_m|t)$) by using the obtained word sequences ($\mathbf{W}_t$) and previously estimated TTLM parameters ($\{\hat{\boldsymbol{\phi}}_{t-s}\}_{s=1}^{S}$).
(3) To obtain a current N-gram model ($P(w_m|w_{m-1}^{m-N-1}, t)$), a rescaling technique based on the dynamic unigram marginal is used, as discussed in Section 3.1.
(4) A recognition result is obtained by using a decoder with the adapted N-gram model ($P(w_m|w_{m-1}^{m-N-1}, t)$).

These four steps are undertaken incrementally for each chunk, as shown in Fig. 3.

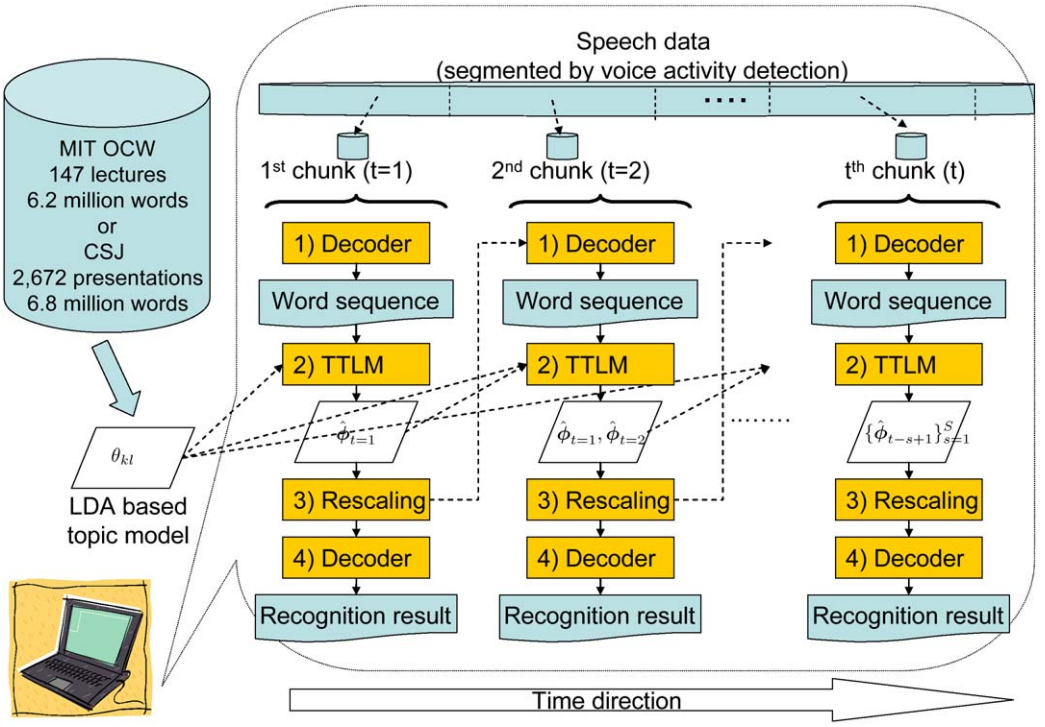

Fig. 3. Unlabeled (unsupervised) incremental adaptation of language models in speech recognition based on the TTLM.

Thus, we can realize the unlabeled (unsupervised) incremental adaptation of language models in an on-line manner within the TTLM framework.[3] The following subsections describe a unigram rescaling technique, which is used at Step 3) in the implementation and the discussion about the chunk size and long-term dependency.

3.1. Unigram rescaling

Once we obtain a unigram probability using the TTLM, we can compute the N-gram probability of the TTLM by rescaling the original N-gram probability with the ratio of the new unigram probability to the original one. This paper uses the dynamic unigram marginal (or known as minimum discrimination information (MDI) adaptation) as a unigram rescaling technique, which can consider back-off probabilities in an N-gram probability (Kneser et al., 1997; Niesler and Willett, 2002). First, we define the following unigram scaling factor for word l:

$$\pi(l,t) \triangleq \left(\frac{P(l|t)}{P(l)}\right)^{\rho}, \tag{23}$$

where $P(l)$ and $P(l\,|\,t)$ are the original unigram probability and the new unigram probability obtained with the TTLM, respectively. ρ is a tuning parameter. Then, the new N-gram probability with a history of word sequence h^{N-1} is represented as follows:

$$P(l|h^{N-1},t) = \begin{cases} \dfrac{\pi(l,t)}{C_0(h^{N-1},t)} P(l|h^{N-1}) & \text{if} \quad n(h^{N-1},l) > 0 \\ \dfrac{1}{C_1(h^{N-1},t)} P(l|h^{N-2},t) & \text{else} \end{cases}, \tag{24}$$

[3] To achieve totally "unsupervised" incremental adaptation, we have to consider how to obtain an appropriate chunk size, as well as label information. This paper uses voice activity detection (e.g., Fujimoto et al., 2007) to divide long speeches automatically into sets of utterances. This paper also examines experimentally how to set an appropriate chunk size from the set of utterances.

where

$$C_0(h^{N-1}, t) = \frac{\sum_{l:n(h^{N-1},l)>0} \pi(l, t) P(l|h^{N-1})}{\sum_{l:n(h^{N-1},l)>0} P(l|h^{N-1})}, \tag{25}$$

and

$$C_1(h^{N-1}, t) = \frac{1 - \sum_{l:n(h^{N-1},l)>0} P(l|h^{N-2}, t)}{1 - \sum_{l:n(h^{N-1},l)>0} P(l|h^{N-1})}. \tag{26}$$

Then, $P(l|h^{N-2}, t)$ is also iteratively calculated by $\pi(l, t)$, $P(l|h^{N-2})$, and $P(l|h^{N-3}, t)$. Thus, the unigram rescaled language model is obtained by modifying the back-off coefficients.

3.2. Chunk size and long term dependency

As regards the chunk size setting, this paper adopts an utterance-based unit, i.e., a chunk unit (t) is composed of several utterances. This is because automatic speech recognition often uses utterance units for decoding, which can be automatically extracted by using voice activity detection (VAD), and an on-line topic model based on utterance units is desirable. Therefore, this paper examines experimentally how to set an appropriate chunk size from the set of utterances in Section 4.1. Especially in the second experiments (Section 4.2), we used one utterance as the chunk size but we used the many terms (S) in the history of the long time dependency to model the topic dynamics across utterances. The reason of adopting one utterance per chunk is that one utterance is the smallest unit, which holds some topic information and can be efficiently integrated to the current automatic speech recognition system. Although the number of terms in history is fixed during one talk, the precision parameter (α_{ts}), which denotes the degree of contribution of the utterances of a certain term to the topic probability, is automatically estimated from data utterance by utterance, based on Section 2.4. Therefore, the TTLM can automatically disregard any useless terms in a long history based on this estimation process. Consequently, the TTLM prefers the long term setting of history, if the precision parameter estimation process works perfectly.

4. Experiments

We performed speech recognition experiments to show the effectiveness of the TTLM for the unlabeled (unsupervised) incremental adaptation of language models. We used two speech recognition tasks; the MIT OpenCourseWare corpus (MIT-OCW, Glass et al., 2007) and the Corpus of Spontaneous Japanese (CSJ, Furui et al., 2000). MIC-OCW is mainly composed of classroom lectures, while CSJ is mainly composed of conference presentations. These were given by one speaker on one subject (e.g., physics, computer science). In such cases, the topics change gradually and the topic continuity is maintained between utterances. Therefore, the TTLM would be suitable for modeling the topic changes in these talks, and we examined the effectiveness of tracking the topic changes using the TTLM.

4.1. Experiments for MIT OpenCourseWare

We designed our first experiments based on MIT-OCW to examine the effectiveness of the TTLM in a simple application of on-line topic models without using the interpolation technique, as described in Section 2.5, and without adjusting the tuning parameter in the unigram rescaling technique, as described in Section 3.1. Generally, on-line topic models consider documents or web pages, which include more than hundred words, as time units (Wei et al., 2007; Iwata et al., 2009). However, it is difficult to provide such a long unit for speech. Actually, most speech recognition tasks do not consider such long speeches and their corpuses do not provide us with information about document-like units. Therefore, in our first experiments, as an initial attempt, we examined the relationship between the speech recognition performance and the length of unit (chunk size) by using the TTLM and we also examined the effectiveness of the

Table 2
Experimental setup for MIT-OCW.

Sampling rate	16 kHz
Feature type	MFCC + energy + Δ + ΔΔ (39 dim.)
Frame length	25 ms
Frame shift	10 ms
Window type	Hamming
# of categories	51 (42 phonemes + 9 noises)
Context-dependent	2193 HMM states (3-state left to right)
HMM topology	32 GMM components
Language model	3-gram (Good-Turing smoothing)
Vocabulary size	70,397
Perplexity	194.1
OOV rate	1.4 %

long-term dependence of the TTLM. We adopted MIT-OCW for this attempt because MIT lectures are classroom lectures, each of which is more than 1 h long and contains an average of more than 10,000 words. Therefore we can use document-like units for the TTLM (e.g., if we use 64 utterances (approximately 500 words) as one chunk, we can monitor 20-epoch dynamics.). We also provide examples of topic dynamics in a lecture about physics.

4.1.1. Speech recognition setup

The training data consisted of 147 lectures from MIT-OCW (128 h of speech data and corresponding 6.2M word transcriptions). The evaluation data consisted of 4 lectures (4.5 h, 43,002 words). Table 2 shows the acoustic and language model information. We used a standard acoustic model, which is a context-dependent model with a continuous density HMM. The HMM parameters were estimated by employing the MIT-OCW training data based on the conventional maximum likelihood approach. Lexical and language models were also obtained by employing the MIT-OCW training data. We used a 3-gram model with a Good-Turing smoothing technique. The Out Of Vocabulary (OOV) rate was 1.4 % and the testset perplexity was 194.1. For decoding, we used a WFST based decoder. The acoustic model construction and LVCSR decoding procedures were performed by using the NTT speech recognition platform SOLON (Hori, 2004). During the adaptation process, we fixed the number of topics at 50 (i.e., $K = 50$). We also fixed the scaling factor $\rho = 1$ in Eq. (23). Namely we use the dynamic unigram marginal Kneser et al. (1997) as a unigram rescaling technique in the implementation and experiments for a simple evaluation of the effect of the topic models.

This work used a set of utterances as a chunk (t). The utterances were obtained by segmenting speech using voice activity detection (Fujimoto et al., 2007). We then prepared several sizes of chunk unit consisting of 16, 32, 64, 128, and 256 utterances for use in the experiments. The lectures in the evaluation set contained an average of 1422 utterances, and the average numbers of chunks in one lecture were 89, 45, 23, 12, and 6.

4.1.2. Experimental results

As a preliminary investigation, we first examined the effectiveness of the TTLM with/without using on-line word probability $\hat{\boldsymbol{\Theta}}$ estimation for 64 utterances per chunk and $S = 10$ long-term dependence. We found that the approach improved the recognition performance (38.5 %) from the baseline performance (38.8 %). However, by comparison with the performance obtained when only using the topic probabilities $\hat{\boldsymbol{\Phi}}$, there was a 0.3 % degradation that was probably due to the sparse training data problem. In addition, the TTLM with the word probabilities are sensitive to the word recognition errors in unlabelled (unsupervised) speech recognition where the word recognition errors directly affect the word (unigram) probabilities. This sensitivity would also be the reason of the degradation. Therefore, we decided to use only the topic probabilities in the language model adaptation in the following experiments.

Then, we examined the TTLM performance as regards the length (S in Section 2.4) of the long-term dependence in the TTLM for a fixed chunk size (64) as shown in Fig. 4. The baseline recognition results were obtained by using the initial 3-gram language model.

When $S = 0$, the approach becomes semi-batch LDA, which estimates the unigram probability using only the data in a chunk. It does not consider the time dependence between different chunks. From Fig. 4, we found that the TTLM was up to 0.4 % better than semi-batch LDA. This result means that considering the topic dynamics across chunks in

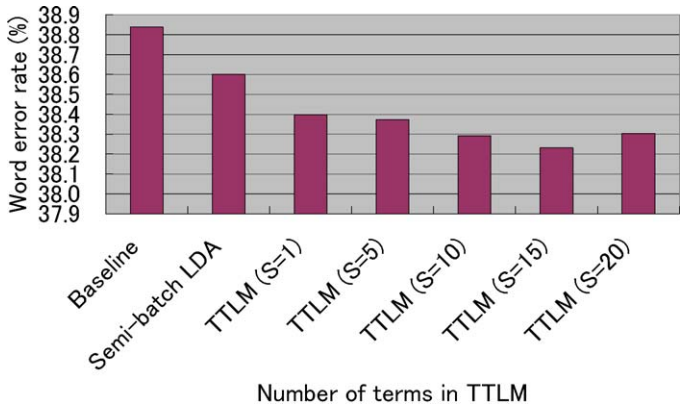

Fig. 4. Dependence of long-term effect of TTLM on length.

the TTLM was effective as regards the lectures. Although the TTLM prefers a long term setting of history (S) if the estimation process of the precision parameter works perfectly, as discussed in Section 3.2, the performance degraded slightly at $S = 20$. Therefore, the result indicates that the TTLM depends on the S setting practically even with the process for estimating the precision parameters. However, we also found that the degradation was not very large, and the S setting was not very sensitive when we chose an S value around 15. Therefore, we used a fixed S ($S = 15$) for the following experiments.

Then, we examined the TTLM performance with respect to chunk size by changing the number of utterances in a chunk from 16 to 256, as shown in Fig. 5. We also examined the batch LDA performance using all the adaptation data for each lecture. With a small chunk size (16 utterances), the TTLM performance was not greatly improved owing to the problem of the sparseness of the data available for estimating the TTLM parameters. As the chunk size increased, the recognition performance improved from the baseline performance by up to 0.6 % at 64 utterances per chunk. However, the 128 and 256 utterance results were again not greatly improved, and the performance was the same as that of batch LDA. This outcome was reasonable since TTLM theoretically converges with batch LDA if we increase the chunk size to include all the adaptation utterances. Therefore, these results indicate that the TTLM could track the topic dynamics if we choose an appropriate chunk size.

Finally, we summarized the TTLM results (64 utterances per chunk) with the baseline results and those for batch adaptation based on LDA and semi-batch adaptation based on LDA (64 utterances per chunk), as shown in Table 3. We compared performance by using the testset perplexity for 3-gram language models, and the word error rate. We also added the testset perplexity for 1-gram language models since this score is a direct measure of TTLM and LDA performance that focuses on the 1-gram probabilities. From Table 3, batch LDA and semi-batch LDA achieved improved performance in terms of the word error rate. However, semi-batch LDA in particular degraded the perplexities owing

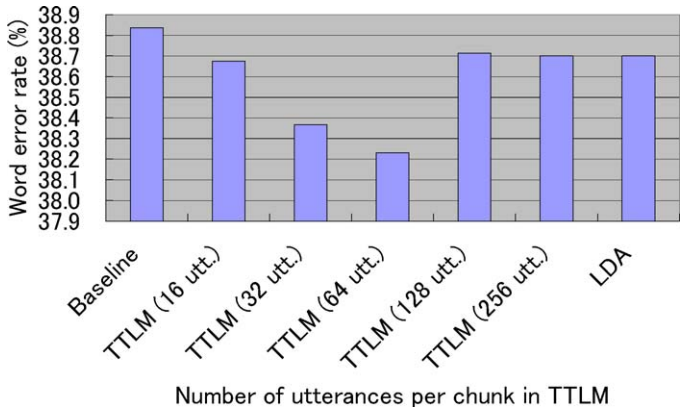

Fig. 5. WER of TTLM for each number of utterances per chunk.

Table 3
Perplexity and word error rate for baseline, batch adaptation based on LDA, semi-batch adaptation based on LDA, and on-line adaptation based on TTLM.

	Baseline	LDA (batch)	LDA (semi-batch)	TTLM (on-line)
Perplexity (1-gram)	599.7	521.8	641.1	499.3
Perplexity (3-gram)	194.1	175.0	218.7	170.4
Word error rate (%)	38.8	38.7	38.6	38.2
Error reduction rate (%)	–	0.3	0.5	1.5

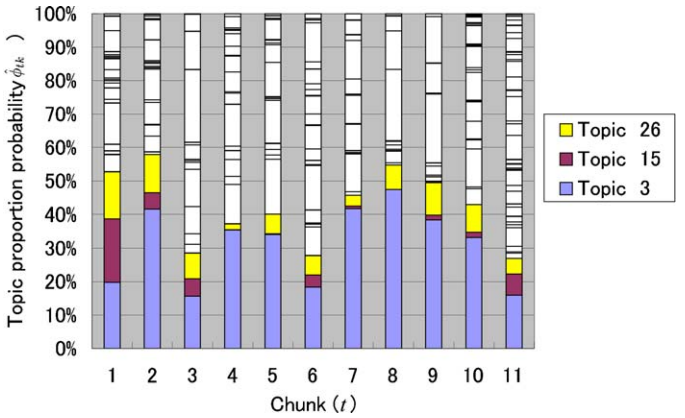

Fig. 6. Time evolution of topic proportion probabilities.

to the over-training problem. On the other hand, although the word error rate gain was not very large since the language model contributed little to the LVCSR performance, the performance of the TTLM steadily improved in terms of the perplexities and word error rate. Therefore, the TTLM performance improved sufficiently, and thus could successfully realize topic tracking in lectures.

4.1.3. Example of topic tracking

The advantage of the TTLM is that the time evolution of topic proportion probabilities can be observed by monitoring the topic probability for each chunk ($\boldsymbol{\phi}_t$). Fig. 6 shows an example of the time evolution of a lecture about physics where the topic proportion probabilities were obtained from the $\boldsymbol{\phi}_t$ value of the TTLM (64 utterances per chunk). To check the contents of each topic (k), Table 4 shows the 10 highest probability nouns from the word probabilities of Topics 3 ($\boldsymbol{\theta}_{k=3}$), 15 ($\boldsymbol{\theta}_{k=15}$), and 26 ($\boldsymbol{\theta}_{k=26}$). Table 4 shows that Topic 3 represents *classical mechanics*, Topic 15 represents *astronomy*, and Topic 26 represents *(time) unit*. The lecture theme was physics and the fact that Topic 3 was always a dominant topic for all chunks in Fig. 6 constitutes a reasonable result. At the beginning of the lecture (1st chunk),

Table 4
Top 10 high probability nouns in word probabilities of Topics 3 ($\boldsymbol{\theta}_{k=3}$), 15 ($\boldsymbol{\theta}_{k=15}$), and 26 ($\boldsymbol{\theta}_{k=26}$).

Topic 3 (∼ classical mechanics)	Topic 15 (∼astronomy)	Topic 26 (∼ (time) unit)
m	Light	Percent
Energy	Degrees	Time
Force	Angle	Dollars
Mass	Frequency	Times
Point	Energy	Minutes
Velocity	Direction	Day
Direction	Waves	Bit
v	Sun	Year
Times	Star	Hour
Speed	Speed	Half

Table 5
An example of the recognition result (HYP) and its correct reference (REF) at 1st chunk, where *astronomy* and *(time) unit* topics scored high probabilities. S and I mean the substitution and insertion errors, respectively.

```
REF:  astronomers ** even use the astronomical unit  which is the mean
HYP:  astronomers it can  use the astronomical units which is the mean
Eval:             I  S                        S

REF:  distance between the earth and the sun and they use light  years
HYP:  distance between the earth and the sun and the  use lights years
Eval:                                          S        S
```

Topics 15 and 26 scored high probabilities, and this indicates that this lecture began by using astronomy as an example with time unit. Tables 5 and 6 show examples of the recognition results and their correct references, which also support the above discussion.

Thus, the TTLM could monitor topic movements by using the topic proportion probabilities. This result also indicates the effectiveness of the TTLM in appropriately tracking the topic dynamics in the lectures.

Finally, we discuss the problem in this experiment. Although the MIT-OCW experimental results reveal the effectiveness of the TTLM, we found that if we reduced the chunk size, the performance did not improve greatly due to the shortage of words in a chunk. On the other hand, most speech recognition tasks deal with several-minute sentences (a few hundred words) uttered by many speakers, and they cannot prepare document-like units for the TTLM. Another problem we face is how to prepare chunk units. Since automatic speech recognition is often used as a real-time interface, on-line adaptation with a large chunk size causes a serious latency problem. Generally, automatic speech recognition uses utterance units (a few seconds) for decoding, which can be automatically extracted by using voice activity detection (VAD), and therefore, an on-line topic model based on utterance units is desirable. Therefore, our second experiments applied the TTLM to the more challenging task of one-utterance-unit topic tracking (one utterance as a chunk unit).

4.2. Experiments for Corpus of Spontaneous Japanese

This section describes large vocabulary continuous speech recognition experiments with on-line topic model adaptation performed using the CSJ. CSJ is mainly composed of conference presentations, and the average length of each presentation in the testset is 20 minutes, and is shorter than MIT-OCW. Since CSJ can not provide so long chunk unit as MIT-OCW, we adopt one utterance as a chunk unit (one utterance unit), which is useful in practical applications. Although the shortage of words in an utterance-based unit might pose a serious problem, we used an interpolation technique, as described in Section 2.5, and a dynamic unigram marginal technique, as described in Section 3.1, to mitigate the problem. Compared with the MIT-OCW task, the CSJ task officially provides three kinds of testset and we can use them for development and evaluation sets. Therefore, we can use various language model adaptation techniques whose parameters can be tuned by using the development set. This experiment mainly compares the TTLM with other adaptation techniques.

Table 6
An example of the recognition result (HYP) and its correct reference (REF) at 11th chunk, where a *classical mechanics* topic scored high probability. S and I mean the substitution and insertion errors, respectively

```
REF:  they try   to demonstrate that the time for an apple to fall does
HYP:  it   tried to demonstrate that the time for an apple to fall does
Eval: S

REF:  depend on it's mass even though it probably is only * very small
HYP:  depend on its  mass even though it probably is only a very small
Eval:          S                                          I
```

Table 7
Experimental setup for CSJ.

Sampling rate	16 kHz
Feature type	MFCC + energy + Δ + ΔΔ (39 dim.)
Frame length	25 ms
Frame shift	10 ms
Window type	Hamming
# of categories	43 phonemes
Context-dependent	5000 HMM states (3-state left to right)
HMM topology	32 GMM components
Training method	Discriminative training (MCE)
Language model	3-gram (Good-Turing smoothing)
Vocabulary size	100,808
Perplexity	82.4 (Dev.) 81.5 (Eval.)
OOV rate	2.3 % (Dev.) 2.9 % (Eval.)

4.2.1. Speech recognition setup

The training data for acoustic model construction consisted of 961 talks from the CSJ conference presentations (234 h of speech data), and the training data for language model construction consisted of 2672 talks from all the CSJ speech data (6.8M word transcriptions). We used a development set consisting of "CSJ testset 2" to tune some parameters (e.g., language model weight, number of topics, and the length of the long-term dependence) and an evaluation set consisting of "CSJ testset 1" with the tuned parameters. The development set consisted of 10 talks (2.4 h, 26,798 words), and the evaluation set consisted of 10 talks (2.3 h, 26,329 words). Table 7 shows acoustic and language model information (Nakamura et al., 2006). We used a state-of-the-art acoustic model, which is a context-dependent model with a continuous density HMM. The HMM parameters were estimated by employing the conference presentations in the CSJ based on a discriminative training (Minimum Classification Error: MCE) approach. Lexical and language models were also obtained by employing all the CSJ speech data. We used a 3-gram model with a Good-Turing smoothing technique. The OOV rates were 2.3 % (Dev.) and 2.9 % (Eval.) and the testset perplexities were 82.4 (Dev.) and 81.5 (Eval.). The acoustic model construction and LVCSR decoding procedures were also performed with the NTT speech recognition platform SOLON (Hori, 2004).

This experiment used one utterance as one chunk (t) unlike the MIT-OCW experiments in Section 4.1, since the CSJ testsets consisted of shorter talks than the MIT-OCW testset. Namely the CSJ testsets consisted of oral presentations of academic conferences ($\sim$ 15 minutes for each talk), while the MIT-OCW testset consisted of coursework lectures (more than 1 h for each lecture). Utterances were obtained by segmenting speech by using voice activity detection.

We used Cache adaptation (Kuhn and De Mori, 1990), the Dynamic Mixture Model (DMM) (Wei et al., 2007), and Batch LDA (Blei et al., 2003) for comparison. The language model based on Cache adaptation uses the following unigram model from the cache unigram model $P_{cache}(w_m|t)$ and the original unigram model $P(w_m)$ (Kneser et al., 1997):

$$P(w_m|t) = \lambda P_{cache}(w_m|t) + (1-\lambda)P(w_m), \tag{27}$$

where λ is an interpolation parameter. The length of the long-term dependence in the Cache adaptation and TTLM was fixed at 10 epochs (utterances) i.e., $S = 10$. The interpolation ratio of the Cache adaptation and the numbers of latent topics in DMM, batch LDA, and TTLM were also tuned by using the development set. All of the language model adaptation techniques were applied to unigram probability, and a 3-gram language model was obtained by using the unigram rescaling technique described in Section 3.1. Then, scaling factor was set at 0.5 ($\rho = 0.5$) in Eq. (23) by referring to speech recognition results for a dynamic unigram marginal (Kneser et al., 1997).

4.2.2. Experimental results

Table 8 compares the word error rates obtained with Baseline 3-gram, Cache, DMM, Batch LDA, and TTLM. In addition, we also list error reduction rates and the numbers of improved speakers from the baseline results.

Table 8
Word error rate (WER) and error reduction rate (ERR) for the development and evaluation sets of the baseline 3-gram (non-adapted), Cache adaptation, dynamic mixture model (DMM), batch LDA, and TTLM approaches.

	Baseline	Cache	DMM	Batch-LDA	TTLM
Dev. WER (ERR)	17.9	16.7 (6.7)	16.9 (5.6)	16.7 (6.7)	16.4 (8.4)
Eval. WER (ERR)	21.0	20.0 (4.8)	20.2 (3.8)	19.9 (5.2)	19.7 (6.2)
Dev. # improved speakers	–	10/10	10/10	10/10	10/10
Eval. # improved speakers	–	9/10	10/10	9/10	10/10

Table 9
Word Error Rate (WER) and Error Reduction Rate (ERR) for the development and evaluation sets of TTLM and its variants.

	TTLM 1st pass result	TTLM w/o interpolation	TTLM
Dev. WER (ERR)	16.4 (8.4)	16.5 (7.9)	16.4 (8.4)
Eval. WER (ERR)	19.9 (5.2)	19.8 (5.7)	19.7 (6.2)
Dev. # improved speakers	10/10	10/10	10/10
Eval. # improved speakers	10/10	10/10	10/10

Table 8 shows that the TTLM provided the best performance, thus confirming its effectiveness. First, we discuss the TTLM and Batch LDA results. The main difference between them relates to the consideration of the topic dynamics in a talk. Therefore, as in Section 4.1, we can confirm that the TTLM can properly model topic dynamics. Second, the main difference between the DMM and the TTLM relates to the consideration of the long-term dependence.[4] In this experiment, the incremental adaptation unit was an utterance, which is prone to suffer from the over-training problem. A TTLM with long-term dependence could properly mitigate the over-training problem by combining estimated parameters in the current adaptation step with those in the past ($S = 10$) adaptation steps. Finally, the main difference between the Cache and the TTLM is whether the unigram probabilities are estimated directly or indirectly via topic proportion probabilities, as discussed in the Introduction. Therefore, this superiority of the TTLM also reveals its effectiveness by further mitigating the over-training problem.

Finally, Table 9 (TTLM and TTLM w/o interpolation) compares the TTLM with/without using the interpolation of the topic-independent word probability, as discussed in Section 2.5. This interpolation slightly improved the word error rates by 0.1 %, which shows the effectiveness of this interpolation technique based on the TTLM by correctly estimating the interpolation weights of the topic-independent word probability as well as those of the topic-dependent word probabilities.

Thus, the two tasks in the speech recognition experiments (MIT-OCW and CSJ) show the effectiveness of the TTLM. Therefore, we can conclude that the TTLM properly tracked temporal changes in language environments.

4.3. Computational consideration of TTLM

In this work, we were not greatly concerned with computation time, and our aim was to evaluate the proposed approaches without search errors occurring during the beam search. Therefore, we used a sufficiently large beam width during decoding, which took about 1.0 RTF for the MIT task and 0.8 RTFs for the CSJ task. The TTLM and unigram-rescaling processes did not take much time (less than 0.1 RTF in total). The 2nd decoding step, as shown in Fig. 3, required the same computation time as the 1st decoding step. Then, although decoding the entire system

[4] The other difference between the TTLM and the original DMM is that the former can estimate hyper-parameters (precision parameters) in Dirichlet distributions by using a stochastic EM algorithm at each chunk, while the DMM uses fixed precision parameters for all chunks. Therefore, the precision parameters in the TTLM can be dynamically changed depending on the data in a chunk. However, we used the same precision parameter estimation for the TTLM and the DMM in this experiment, to clarify the effectiveness of the long-term dependences. The effectiveness of the hyper-parameter optimization in topic models is discussed in Asuncion et al. (2009).

took 2.0–1.6 RTFs, it would be effective for off-line speech recognition applications. Furthermore, we experimentally found that the difference of the recognition performance between the results of the 1st and 2nd pass decoding was not so large, as shown in Table 9 (TTLM and TTLM 1st pass result). Therefore, although we used the 2nd pass decoding in this paper, we may use only the 1st pass decoding if we require fast and low latency systems in some applications. In addition, we can also reduce the computation time required for decoding by using a narrower beam width. The 2nd pass decoding can also be quickly performed by searching for hypotheses in lattices, which were outputted by the 1st decoding step.

Thus, the TTLM can be efficiently realized in speech recognition without increasing total computational cost so heavily.

5. Summary

This paper proposed a topic tracking language model (TTLM) and applied it to the unlabeled (unsupervised) incremental adaptation of language models. Experiments showed the effectiveness of the TTLM by achieving improved performance in lecture and conference presentation adaptation tasks. However, the recognition performance depends on the chunk or history size, and our future work will focus on extending the TTLM by jointly optimizing chunk/history sizes and latent topic models. In fact, one of the authors has extended the topic tracking model to improve robustness against chunk size setting by considering the multiscale dynamics of latent topic models in parallel (Iwata et al., 2010). In the future, we will consider the chunk/history size determination problem by considering the multi-scale dynamics or well-known Dirichlet process mixture approaches (e.g., Rasmussen, 2000; Teh et al., 2006).

Our goal for this work is to model speech communication by taking various kinds of temporal changes in speech into consideration (e.g., speakers (Akita and Kawahara, 2004) and roles (Huang and Renals, 2008) in addition to topics). In this case, we have to deal with topic changes caused by scene switching. These are modeled by the state transition from topic to topic (Griffiths et al., 2005; Hsu and Glass, 2006; Gruber et al., 2007; Sako et al., 2008; Chen et al., 2009), and we will extend this work so that we can represent both the smooth and rapidly switching temporal changes in speech that occur in acoustic and language environments.

We also consider that the word probability θ plays an interesting role in that the probabilities of the important words in a certain latent topic change gradually over time. Although the experiments in this paper only used the dynamics of the topic probabilities while the word probability for each chunk was fixed to avoid the over-training problem, we want to exploit this phenomenon obtained by the on-line word probability estimation in speech recognition in the future.

Acknowledgments

We thank the MIT Spoken Language Systems Group and Dr. Atsunori Ogawa at NTT Communication Science Laboratories for helping us to perform speech recognition experiments based on MIT-OCW. We would also like to thank the anonymous reviewers for their valuable comments. This work was undertaken by NTT and Kobe University as part of an internship program.

Appendix A. Derivation of inference

A.1. Marginalization of joint distribution

This section provides the derivation of the following marginalized joint distribution of data and latent topics in Eq. (7).

$$P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = \iint P(\mathbf{W}_t, \mathbf{Z}_t|\boldsymbol{\phi}_t, \boldsymbol{\Theta}_t) P(\boldsymbol{\phi}_t|\hat{\boldsymbol{\phi}}_{t-1}, \alpha_t) P(\boldsymbol{\Theta}_t|\hat{\boldsymbol{\Theta}}_{t-1}, \boldsymbol{\beta}_t) d\boldsymbol{\phi}_t d\boldsymbol{\Theta}_t. \tag{A.1}$$

By substituting the concrete forms of the distributions (Eqs. (3), (5), and (6)) with normalization constants of the Dirichlet distribution ($C_{\mathcal{D}}(\cdot)$) into (A.1), we obtain the following equation

$$\begin{aligned}
&P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) \\
&= \frac{\int\int \prod_m \phi_{tz_m} \theta_{tz_m w_m} \prod_k \phi_{tk}^{\alpha_t \hat{\phi}_{t-1k}-1} \prod_l \theta_{tkl}^{\beta_{tk}\hat{\theta}_{t-1kl}-1} d\boldsymbol{\phi}_t d\boldsymbol{\Theta}_t}{C_{\mathcal{D}}(\{\alpha_t \hat{\phi}_{t-1k}\}_k) \prod_k C_{\mathcal{D}}(\{\beta_{tk}\hat{\theta}_{t-1kl}\}_l)} \\
&= \frac{\int\int \prod_k \prod_l \phi_{tk}^{n_{tk}} \theta_{tkl}^{n_{tkl}} \phi_{tk}^{\alpha_t \hat{\phi}_{t-1k}-1} \theta_{tkl}^{\beta_{tk}\hat{\theta}_{t-1kl}-1} d\boldsymbol{\phi}_t d\boldsymbol{\Theta}_t}{C_{\mathcal{D}}(\{\alpha_t \hat{\phi}_{t-1k}\}_k) \prod_k C_{\mathcal{D}}(\{\beta_{tk}\hat{\theta}_{t-1kl}\}_l)} \\
&= \frac{\int\int \prod_k \phi_{tk}^{n_{tk}+\alpha_t \hat{\phi}_{t-1k}-1} \prod_l \theta_{tkl}^{n_{tkl}+\beta_{tk}\hat{\theta}_{t-1kl}-1} d\boldsymbol{\phi}_t d\boldsymbol{\Theta}_t}{C_{\mathcal{D}}(\{\alpha_t \hat{\phi}_{t-1k}\}_k) \prod_k C_{\mathcal{D}}(\{\beta_{tk}\hat{\theta}_{t-1kl}\}_l)},
\end{aligned} \tag{A.2}$$

where the integrals in (A.2) are analytically solved as normalization constants of the Dirichlet distribution:

$$\int \prod_k \phi_{tk}^{n_{tk}+\alpha_t \hat{\phi}_{t-1k}-1} d\boldsymbol{\phi}_t = C_{\mathcal{D}}(\{n_{tk} + \alpha_t \hat{\phi}_{t-1k}\}_k), \tag{A.3}$$

and

$$\int \prod_k \prod_l \theta_{tkl}^{n_{tkl}+\beta_{tk}\hat{\theta}_{t-1kl}-1} d\boldsymbol{\Theta}_t = \prod_k C_{\mathcal{D}}(\{n_{tkl} + \beta_{tk}\hat{\theta}_{t-1kl}\}_l). \tag{A.4}$$

Therefore, by substituting Eqs. (A.3) and (A.4) into (A.2), $P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t)$ is rewritten as follows:

$$P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = \frac{C_{\mathcal{D}}(\{n_{tk} + \alpha_t \hat{\phi}_{t-1k}\}_k) \prod_k C_{\mathcal{D}}(\{n_{tkl} + \beta_{tk}\hat{\theta}_{t-1kl}\}_l)}{C_{\mathcal{D}}(\{\alpha_t \hat{\phi}_{t-1k}\}_k) \prod_k C_{\mathcal{D}}(\{\beta_{tk}\hat{\theta}_{t-1kl}\}_l)}. \tag{A.5}$$

The concrete form of the normalization constant $C_{\mathcal{D}}$ is defined as follows:

$$C_{\mathcal{D}}(\{x_i\}_i) = \frac{\Gamma(\sum_i x_i)}{\prod_i \Gamma(x_i)}. \tag{A.6}$$

Therefore, by substituting Eqs. (A.6) into (A.5), $P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t)$ is rewritten as follows:

$$\begin{aligned}
P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t) = &\frac{\Gamma(\alpha_t)}{\prod_k \Gamma(\alpha_t \hat{\phi}_{t-1k})} \frac{\prod_k \Gamma(n_{tk} + \alpha_t \hat{\phi}_{t-1k})}{\Gamma(n_t + \alpha_t)} \\
&\times \prod_k \frac{\Gamma(\beta_{tk})}{\prod_l \Gamma(\beta_{tk}\hat{\theta}_{t-1kl})} \frac{\prod_l \Gamma(n_{tkl} + \beta_{tk}\hat{\theta}_{t-1kl})}{\Gamma(n_{tk} + \beta_{tk})}.
\end{aligned} \tag{A.7}$$

Thus, $P(\mathbf{W}_t, \mathbf{Z}_t|\hat{\boldsymbol{\phi}}_{t-1}, \hat{\boldsymbol{\Theta}}_{t-1}, \alpha_t, \boldsymbol{\beta}_t)$ is obtained as a Polya (Dirichlet-Multinomial) distribution:

A.2. Gibbs sampling

This section provides the derivation of the conditional probability (Eq. (10)) for the Gibbs sampling. For simplicity, we omit $\hat{\boldsymbol{\phi}}_{t-1}$, $\hat{\boldsymbol{\Theta}}_{t-1}$, α_t, and $\boldsymbol{\beta}_t$ from the conditions of the probabilities in this derivation. From the product (Bayes) rule, we can derive the following equation:

$$P(z_m = k|\mathbf{W}_t, \mathbf{Z}_{t\backslash m}) = \frac{P(\mathbf{W}_t, \mathbf{Z}_t)}{P(\mathbf{Z}_{t\backslash m})P(\mathbf{W}_t|\mathbf{Z}_{t\backslash m})}. \tag{A.8}$$

Then, we focus on $P(\mathbf{W}_t \,|\, \mathbf{Z}_{t\backslash m})$. From the summation and production rules and the i. i. d. assumption of the latent topic model (i.e., $P(\mathbf{W}_t|\mathbf{Z}_t) = P(\mathbf{W}_{t\backslash m}|\mathbf{Z}_{t\backslash m})P(w_m|z_m = k)$), we can derive the following equation:

$$P(\mathbf{W}_t|\mathbf{Z}_{t\backslash m}) = \sum_{k=1}^{K} P(\mathbf{W}_{t\backslash m}, w_m, z_m = k|\mathbf{Z}_{t\backslash m}) = P(\mathbf{W}_{t\backslash m}|\mathbf{Z}_{t\backslash m}) \underbrace{\sum_{k=1}^{K} P(w_m|z_m = k)P(z_m = k)}_{(*)}. \tag{A.9}$$

Factor $(*)$ does not depend on $z_m = k$, which is the argument of the focused probability $P(z_m = k \,|\, \mathbf{W}_t, \mathbf{Z}_{t\backslash m})$ in Eq. (A.8). Therefore, we obtain the following proportional relation:

$$P(\mathbf{W}_t|\mathbf{Z}_{t\backslash m}) \propto P(\mathbf{W}_{t\backslash m}|\mathbf{Z}_{t\backslash m}). \tag{A.10}$$

By substituting (A.10) into (A.8) and using the production rule, we obtain

$$P(z_m = k|\mathbf{W}_t, \mathbf{Z}_{t\backslash m}) \propto \frac{P(\mathbf{W}_t, \mathbf{Z}_t)}{P(\mathbf{W}_{t\backslash m}, \mathbf{Z}_{t\backslash m})}. \tag{A.11}$$

The concrete form of the numerator ($P(\mathbf{W}_t, \mathbf{Z}_t)$) in Eq. (A.11) is given in Eq. (7) (or (A.7)). The concrete form of the denominator ($P(\mathbf{W}_{t\backslash m}, \mathbf{Z}_{t\backslash m})$) is obtained by using Eq. (A.7) as follows:

$$\begin{aligned} P(\mathbf{W}_{t\backslash m}, \mathbf{Z}_{t\backslash m}) = {} & \frac{\Gamma(\alpha_t)}{\prod_k \Gamma(\alpha_t \hat{\phi}_{t-1k})} \frac{\Gamma(n_{tz_m} - 1 + \alpha_t \hat{\phi}_{t-1z_m}) \prod_{k \neq z_m} \Gamma(n_{tk} + \alpha_t \hat{\phi}_{t-1k})}{\Gamma(n_t - 1 + \alpha_t)} \\ & \times \left(\prod_k \frac{\Gamma(\beta_{tk})}{\prod_l \Gamma(\beta_{tk} \hat{\theta}_{t-1kl})} \right) \left(\prod_{k \neq z_m} \frac{\prod_l \Gamma(n_{tkl} + \beta_{tk} \hat{\theta}_{t-1kl})}{\Gamma(n_{tk} + \beta_{tk})} \right) \\ & \times \frac{\Gamma(n_{tz_m w_m} - 1 + \beta_{tz_m} \hat{\theta}_{t-1z_m w_m}) \prod_{l \neq w_m} \Gamma(n_{tz_m l} - 1 + \beta_{tz_m} \hat{\theta}_{t-1z_m l})}{\Gamma(n_{tz_m} - 1 + \beta_{tz_m})}. \end{aligned} \tag{A.12}$$

Therefore, by substituting Eqs. (A.7) and (A.12) into Eq. (A.11), we obtain

$$\begin{aligned} P(z_m = k|\mathbf{W}_t, \mathbf{Z}_{t\backslash m}) \propto {} & \frac{\Gamma(n_{tk} + \alpha_t \hat{\phi}_{t-1k})}{\Gamma(n_t + \alpha_t)} \frac{\Gamma(n_t - 1 + \alpha_t)}{\Gamma(n_{tk} - 1 + \alpha_t \hat{\phi}_{t-1k})} \\ & \times \frac{\Gamma(n_{tkw_m} + \beta_{tk} \hat{\theta}_{t-1kw_m})}{\Gamma(n_{tk} + \beta_{tk})} \frac{\Gamma(n_{tk} - 1 + \beta_{tk})}{\Gamma(n_{tkw_m} - 1 + \beta_{tk} \hat{\theta}_{t-1kw_m})}. \end{aligned} \tag{A.13}$$

By using $\Gamma(x+1) = x\Gamma(x)$, Eq. (A.13) is represented as follows:

$$P(z_m = k|\mathbf{W}_t, \mathbf{Z}_{t\backslash m}) \propto \frac{(n_{tk} - 1 + \alpha_t\hat{\phi}_{t-1k})}{(n_t - 1 + \alpha_t)} \frac{\Gamma(n_{tk} - 1 + \alpha_t\hat{\phi}_{t-1k})\Gamma(n_t - 1 + \alpha_t)}{\Gamma(n_t - 1 + \alpha_t)\Gamma(n_{tk} - 1 + \alpha_t\hat{\phi}_{t-1k})}$$
$$\times \frac{(n_{tkw_m} - 1 + \beta_{tk}\hat{\theta}_{t-1kw_m})}{(n_{tk} - 1 + \beta_{tk})} \frac{\Gamma(n_{tkw_m} - 1 + \beta_{tk}\hat{\theta}_{t-1kw_m})\Gamma(n_{tk} - 1 + \beta_{tk})}{\Gamma(n_{tk} - 1 + \beta_{tk})\Gamma(n_{tkw_m} - 1 + \beta_{tk}\hat{\theta}_{t-1kw_m})}$$
$$= \frac{n_{tk\backslash m} + \alpha_t\hat{\phi}_{t-1k}}{n_{t\backslash m} + \alpha_t} \frac{n_{tkw_m\backslash m} + \beta_{tk}\hat{\theta}_{t-1kw_m}}{n_{tk\backslash m} + \beta_{tk}}. \tag{A.14}$$

Thus, we derive the concrete form of Eq. (10), which is proportional to conditional probability $P(z_m = k \,|\, \mathbf{W}_t, \mathbf{Z}_{t\backslash m})$.

References

Akita, Y., Kawahara, T., 2004. Language model adaptation based on PLSA of topics and speakers. In: Proc. ICSLP'04, pp. 1045–1048.

Asuncion, A., Welling, M., Smyth, P., Teh, Y., 2009. On smoothing and inference for topic models. In: Proc. UAI'09.

Bellegarda, J., 2000. Exploiting latent semantic information in statistical language modeling. Proceedings of IEEE 88 (8), 1279–1296.

Bellegarda, J., 2004. Statistical language model adaptation: review and perspectives. Speech Communication 42 (1), 93–108.

Blei, D.M., Lafferty, J.D., 2006. Dynamic topic models. In: Proc. ICML'06, pp. 113–120.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent Dirichlet allocation. Journal of Machine Learning Research 3, 993–1022.

Chen, H., Branavan, S., Barzilay, R., Karger, D., 2009. Global models of document structure using latent permutations. In: Proc. NAACL/HLT'09, pp. 371–379.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R., 1990. Indexing by latent semantic analysis. Journal of the American society for information science 41 (6), 391–407.

Federico, M., 1996. Bayesian estimation methods of *n*-gram language model adaptation. In: Proc. ICSLP'96, pp. 240–243.

Fujimoto, M., Ishizuka, K., Kato, H., 2007. Noise robust voice activity detection based on statistical model and parallel non-linear Kalman filtering. In: Proc. ICASSP'07. Vol. 4, pp. 797–800.

Furui, S., Maekawa, K.H., Isahara, M., 2000. A Japanese national project on spontaneous speech corpus and processing technology. In: Proc. ASR'00, pp. 244–248.

Gildea, D., Hofmann, T., 1999. Topic-based language models using EM. In: Proc. Eurospeech'99, pp. 2167–2170.

Glass, J., Hazen, T.J., Cyphers, S., Malioutov, I., Huynh, D., Barzilay, R., 2007. Recent progress in the MIT spoken lecture processing project. In: Proc. Interspeech'07, pp. 2553–2556.

Griffiths, T., Steyvers, M., 2004. Finding scientific topics. In: Proc. of the National Academy of Sciences 101 (Suppl.1), 5228–5235.

Griffiths, T., Steyvers, M., Blei, D., Tenenbaum, J., 2005. Integrating topics and syntax. Advances in Neural Information Processing Systems 17, 537–544.

Gruber, A., Rosen-Zvi, M., Weiss, Y., 2007. Hidden topic Markov models. In: Proc. AISTATS'07, vol. 2, pp. 163–170.

Hofmann, T., 1999. Probabilistic latent semantic analysis. In: Proc. UAI'99, pp. 289–296.

Hori, T., 2004. NTT Speech recognizer with OutLook On the Next generation: SOLON. In: Proceedings of the NTT Workshop on Communication Scene Analysis, vol. 1, SP-6.

Hori, T., Sudoh, K., Tsukada, H., Nakamura, A., 2009. World-wide media browser – multilingual audio–visual content retrieval and browsing system. NTT Technical Review 7 (2).

Hsu, B.J., Glass, J., 2006. Style & topic language model adaptation using HMMLDA. In: Proc. EMNLP 2006, pp. 373–381.

Huang, S., Renals, S., 2008. Unsupervised language model adaptation based on topic and role information in multiparty meeting. In: Proc. Interspeech'08, pp. 833–836.

Iwata, T., Watanabe, S., Yamada, T., Ueda, N., 2009. Topic tracking model for analyzing consumer purchase behavior. In: Proc. IJCAI'09, pp. 1427–1432.

Iwata, T., Yamada, T., Sakurai, Y., Ueda, N., 2010. Online multiscale dynamic topic models. In: Proc. of 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Minig (KDD2010), 663–672.

Iyer, R., Ostendorf, M., 1996. Modeling long distance dependence in language: topic mixtures vs. dynamic cache models. In: Proc. ICSLP'96, vol. 1, pp. 236–239.

Kneser, R., Peters, J., Klakow, D., 1997. Language model adaptation using dynamic marginals. In: Proc. Eurospeech'97, pp. 1971–1974.

Kuhn, R., De Mori, R., 1990. A cache-based natural language model for speech recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (6), 570–583.

Makhoul, J., Kubala, F., Leek, T., Liu, D., Nguyen, L., Schwartz, R., Srivastava, A., 2000. Speech and language technologies for audio indexing and retrieval. Proceedings of IEEE 88 (8), 1338–1353.

Masataki, H., Sagisaka, Y., Hisaki, K., Kawahara, T., 1997. Task adaptation using map estimation in *n*-gram language modeling. In: Proc. ICASSP'97, vol. 2, pp. 783–786.

Minka, T., 2000. Estimating a Dirichlet distribution. Tech. rep., MIT.

Nakamura, A., Oba, T., Watanabe, S., Ishizuka, K., Fujimoto, M., Hori, T., Mc-Dermott, E., Minami, Y., 2006. Evaluation of the SOLON speech recognition system: 2006 benchmark using the Corpus of Spontaneous Japanese. IPSJ SIG Notes 2006 (136), 251–256 (in Japanese).

Niesler, T., Willett, D., 2002. Unsupervised language model adaptation for lecture speech transcription. In: Proc. Interspeech'02, pp. 1413–1416.

Rasmussen, C., 2000. The infinite Gaussian mixture model. Advances in Neural Information Processing Systems 12, 554–560.
Rosenfeld, R., 2000. Two decades of statistical language modeling: where do we go from here. Proceedings of IEEE 88 (8), 1270–1278.
Sako, A., Takiguchi, T., Ariki, Y., 2008. Language modeling using PLSA-based topic HMM. IEICE Transactions on Information and Systems E91-D, 522–528.
Teh, Y., Jordan, M., Beal, M., Blei, D., 2006. Hierarchical dirichlet processes. Journal of the American Statistical Association 101 (476), 1566–1581.
Wallach, H.M., 2006. Topic modeling: beyond bag-of-words. In: Proc. ICML'06, pp. 977–984.
Wang, C., Blei, D., Heckerman, D., 2008. Continuous time dynamic topic models. In: Proc. UAI'08, pp. 579–586.
Wei, X., Sun, J., Wang, X., 2007. Dynamic mixture models for multiple timeseries. In: Proc. IJCAI'07, pp. 2909–2914.