IEICE TRANS. INF. & SYST., VOL.E100–D, NO.9 SEPTEMBER 2017

PAPER

Modeling Content Structures of Domain-Specific Texts with RUP-HDP-HSMM and Its Applications

Youwei LU[†a)], ***Nonmember***, **Shogo OKADA**[††b)], ***and*** **Katsumi NITTA**[†††c)], ***Members***

SUMMARY We propose a novel method, built upon the hierarchical Dirichlet process hidden semi-Markov model, to reveal the content structures of unstructured domain-specific texts. The content structures of texts consisting of sequential local contexts are useful for tasks, such as text retrieval, classification, and text mining. The prominent feature of our model is the use of the recursive uniform partitioning, a stochastic process taking a view different from existing HSMMs in modeling state duration. We show that the recursive uniform partitioning plays an important role in avoiding the rapid switching between hidden states. Remarkably, our method greatly outperforms others in terms of ranking performance in our text retrieval experiments, and provides more accurate features for SVM to achieve higher F1 scores in our text classification experiments. These experiment results suggest that our method can yield improved representations of domain-specific texts. Furthermore, we present a method of automatically discovering the local contexts that serve to account for why a text is classified as a positive instance, in the supervised learning settings.

key words: *hidden semi-Markov models, content structure, local features, text mining, rapid switching*

1. Introduction

In this paper, we present a novel nonparametric Bayesian model called RUP-HDP-HSMM, aiming to uncover the content structures of unstructured texts of a specific domain. Besides, we show that content structures discovered in texts are useful for different kinds of application.

The content structure is defined in terms of the local contexts underlying a text. A local context here is referred to as a fragment of the text, exhibiting a salient word distribution, that is, a local feature or a topic. For instance, articles about earthquakes may comprise local contexts each containing information about quake strength, damage, government response, etc. In general, the existence of local contexts in an arbitrary collection of texts cannot be guaranteed; however, texts from the same domain are shown to have high similarity, and word recurrence patterns in them indeed exist [1], [2]. Domain-specific texts widely exist as digital resources both on the Internet (for example, news reports or discussions on a social issue) and in our daily life (such as meeting scripts). Revealing content structures of texts helps to extract insightful local information, and thus result in a more accurate representation of a text, than those algorithms that simply view each text as a "bag of words", e.g., topic models.

Inspired by [3], which, associating the topic of local contexts with a hidden state, employs a hidden Markov model (HMM) to capture the relations between topics and reveal drifts between different local contexts in texts, we propose a novel method, aiming at resolving two critical issues of modeling the content structure of real-world domain-specific texts with HMM/HSMM-based models—inference of the number of topics and avoidance of rapid switching.

To infer the number of topics, we adopt the nonparametric Bayesian approach, by which the number of topics can be estimated in the process of model inference. In specific, we put our method within the framework of the hierarchical Dirichlet process hidden semi-Markov model (HDP-HSMM) [4], an extension to the nonparametric Bayesian HMM (HDP-HMM) [5]. It is well known that if the duration times of hidden states are not properly modeled with HDP-HMM, rapid switching is very likely to occur, a situation where unrealistic states are created and rapidly switch between one another [6]. HDP-HSMM can succeed in preventing rapid switching in certain tasks [4]. However, for unstructured real-world natural language texts, suppressing rapid switching is even more challenging, because it is most probable that the duration times of a hidden state—lengths of local contexts associated with a specific topic—have such a complicated distribution that a conventional HDP-HSMM assuming simple duration distributions, like Poission or negative Binomial distributions, is still insufficient to avoid rapid switching. To address the rapid switching issue, we introduce a stochastic process called recursive uniform partitioning (RUP). Combining the RUP and the idea of HDP-HSMM, we propose the RUP-HDP-HSMM, which not only allows the number of hidden states to be inferred, but also succeeds in avoiding rapid switching for discovering content structures of domain-specific texts.

Modeling content structures can be used in different tasks, such as text retrieval, classification, and text mining. In particular, we propose a method of automatically finding texts' local contexts that contribute to explain why they are classified as positive instances.

Manuscript received January 30, 2017.
Manuscript revised May 9, 2017.
Manuscript publicized June 9, 2017.

[†]The author is with the Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, Yokohama-shi, 226–8503 Japan.
[††]The author is with the School of Information Science, Japan Advanced Institute of Science and Technology, Yokohama-shi, 226–8503 Japan.
[†††]The author is with the School of Computing, Tokyo Institute of Technology, Nomi-shi, 923–1292 Japan.
a) E-mail: royui@ntt.dis.titech.ac.jp
b) E-mail: okada-s@jaist.ac.jp
c) E-mail: nitta@dis.titech.ac.jp
DOI: 10.1587/transinf.2017EDP7043

In summary, the contributions of this paper are as follows. First, we propose the RUP-HDP-HSMM, a novel nonparametric Bayesian HSMM-based model for discovering the context structure of domain-specific texts, along with the corresponding model inference algorithms; in particular, we develop a stochastic process called recursive uniform partitioning, playing a critical role of preventing rapid switching. Second, we propose a text mining method of finding the informative local contexts that lead the text to be a positive instance. Finally, we empirically show that inferred content structures underlying texts can provide meaningful features for text retrieval and classification tasks.

2. Related Work

Existing unsupervised learning methods of modeling the content structures of texts generally include HMM/HSMM-based and topic model-based methods. We briefly review these methods here, and highlight their difference from our work.

2.1 HMM/HSMM-Based Methods

The content model proposed in [3] is the first attempt to reveal the content structures of domain-specific texts, which is however closely related to prior research on text segmentation [7], [8]. [3] employs a hidden Markov model, where a hidden state corresponds to a distinct topic, and a bigram language model is embedded in each topic, responsible for generating sentences that are assigned this topic. Because the topics are shared among all texts and the Markovian relations between topics are modeled with a transition matrix, both the inter-sentential relations within a text and inter-topic relations can be captured. A noticeable contribution of [3] is that it concentrates on the texts within a particular domain according to the findings of [2] and [1], differing from more recent topic model-based models that put little emphasis on the domain of texts.

However, the use of a simple HMM in [3] inherently has two disadvantages: (1) the number of involved hidden states is a fixed number that has to be set as a model parameter; (2) the duration times of a hidden state in HMMs implicitly have a geometric distribution, which is an unrealistic assumption for real-world texts, as the probability exponentially decays as local contexts increase in length. To overcome these weaknesses, the hierarchical Dirichlet process hidden semi-Markov model (HDP-HSMM) is proposed [4], which can explicitly specify the distributions of duration times of hidden states as traditional hidden semi-Markov models (HSMM) [9], and can infer the number of hidden states in the nonparametric settings as HDP-HMM [5]. However, as discussed earlier, simple duration distributions, usually employed in conventional HDP-HSMMs, cannot properly model the complicated duration times of local contexts in real-world texts. A further extension of the HDP-HSMM is therefore needed, leading to our method called RUP-HDP-HSMM, which is discussed in detail in Sect. 3.

2.2 Topic Model-Based Methods

Traditional topic models like LDA are built upon the "bag-of-words" assumption, which completely ignores local word patterns in texts. Hence, they are unsuitable for analyzing the content structure of domain-specific texts.

Nevertheless, a number of variants of LDA have emerged that relax the "bag-of-words" assumption, by incorporating local context information. The model that is most closely related to our method is the hidden topic Markov model (HTMM) [10]. The HTMM posits a generative process, where the generation of words in a sentence depends on not only the topic assigned to the sentence but also the topic of the previous sentence; the probability of two adjacent sentences having the same topic is governed by a global Bernoulli distribution. In HTMM, the transitions between topics within a text are still modeled in a Markovian fashion. However, instead of a global transition matrix as assumed in [3], HTMM generates a transition matrix for every document, with all rows being the same as the topic distribution of the text in LDA. As an extension of LDA, the HTMM provides an efficient framework of incorporating the local information for topic inference. Yet it is still a parametric model like the LDA, and thus cannot make inference over the number of topics given the data. Besides, HTMM assumes that topics are independently generated given the topic distribution in a text, and thus the absence of the global transition matrix makes it unable to capture the topic relations in domain-specific texts.

[11] proposed another topic model-based model for explicit content modeling. In contrast to our focus on unstructured texts, [11] analyzes the underlying structures of structured texts (e.g., articles from the Wikipedia about famous cities in the world). Although the content model of [11] is particularly effective in modeling the structured texts, the use of the GMM prior is too limited for unstructured real-world texts, as the topic orders may be very different in different texts.

The Multi-grain LDA [12] makes topic inference based on the local context information. To this end, a sequence of sliding windows are used, each covering several adjacent sentences. Every sliding window is associated with a topic distribution over local topics, and a topic distribution over global topics is shared among all sliding windows in a text. The Multi-grain LDA is flexible and powerful in topic modeling. However, the loose connection between topic distributions in adjacent sentences makes Multi-grain LDA so sensitive to local information that the underlying relations between sentences in a longer range tend to be overlooked.

It is worth noting that topic segmentation based on topic models [13], [14] is different from modeling content structure. In these topic segmentation models, all segments are locally identified, which can have different topic distributions. Because similar segments are not directly grouped together, the content structure of domain-specific texts cannot be explicitly represented.

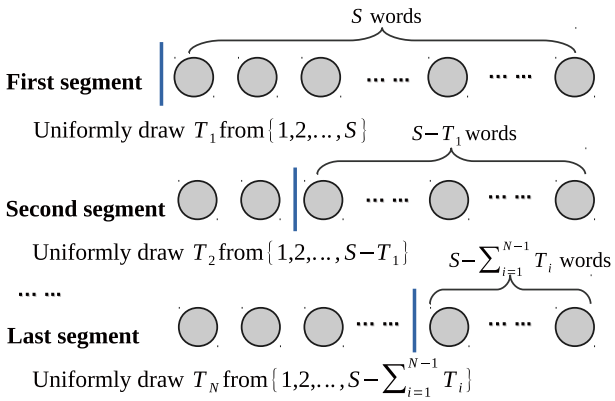


Fig. 1 An illustration of RUP generating a partition of S-word text. Each node here represents a word. After a few steps of uniformly drawing the ending position of a segment from the remaining nodes, a partition for the text is obtained.

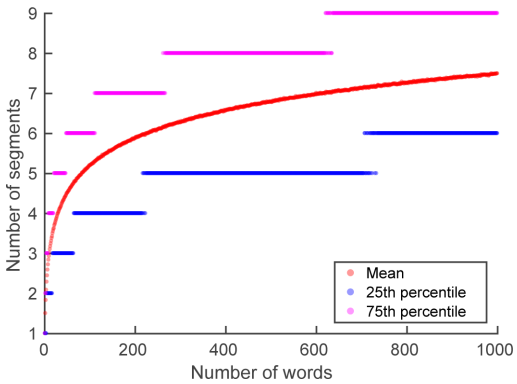


Fig. 2 Simulation of drawing partitions of documents of different lengths, i.e., the number of words. For each length, we simulate 10,000 draws according to RUP, and depict the mean, 25th and 75th percentiles of the number of segments.

3. The Model

In this section, we present our method called RUP-HDP-HSMM, which derives from combining the RUP and HDP-HSMM. Because we associate the underlying topics in domain-specific texts with the hidden states in the HDP-HSMM, topics and hidden states are interchangeably used here for describing our method.

3.1 Notation

Let $C = \{d_1, \cdots, d_M\}$ represent the corpus consisting of M texts, where d_i denotes the i-th text. Each text is composed of a sequence of words, with the length denoted by S. We use a superscript to denote the text number, and a subscript to index a word. Let w_j^i denote the j-th word in text d_i, and its hidden topic z_j^i. Let a_c^i represent the topic associated with the c-th local context in d_i. Let the length of the c-th local context in d_i be T_c^i. We use vectors to represent the sets of corresponding variables in all texts. For example, $\mathbf{w}$ and $\mathbf{z}$ represent the sets of all words and hidden topics, respectively. We use ϕ_k to represent the parameters of a multinomial distribution, associated with the k-th topic. All ϕ_k are drawn from a symmetric Dirichlet distribution with the parameter η. Rows of the transition matrix are denoted by $\{\pi_k\}$, which are generated from a hierarchical Dirichlet process (HDP) prior, where two concentration parameters γ and α_0 are involved.

3.2 Recursive Uniform Partitioning

Unlike HDP-HSMM generating duration times of a hidden state with a state-dependent duration distribution, RUP-HDP-HSMM utilizes the recursive uniform partitioning (RUP) to draw a partition for each text. Given a partition, the duration time of the i-th local context in a text is equal to the length of the i-th segment of the partition. The generating process of drawing a partition by RUP is illustrated in Fig. 1.

Let $P = \{T_1, \cdots, T_N\}$ represent a partition for an S-word text, consisting of N segments. $T_1, \cdots, T_N$ are random variables over positive integers, each representing the length of a segment and subject to the constraint $T_1 + T_2 + \cdots + T_N = S$. The procedure of drawing a partition for this text is as follows. First, we draw T_1 uniformly from $\{1, 2, \cdots, S\}$. If T_1 is equal to S, then all sentences constitute the one-segment partition $P = \{T_1\}$; otherwise, we proceed to draw the second segment. With the first segment fixed that consists of the first T_1 words, T_2 is drawn from the discrete uniform distribution over $\{1, 2, \cdots, S - T_1\}$. If $T_2 = S - T_1$, then the partition consisting of two segments $P = \{T_1, T_2\}$ is obtained. Otherwise, we recursively draw the length of each segment from the remaining words, until there is no words left. The probability of drawing $P = \{T_1, \cdots, T_N\}$ is shown as follows.

$$\mathscr{P}\{P|S\} = \frac{1}{S \times (S - T_1) \times \cdots \times (S - \sum_{i=1}^{N-1} T_i)} \tag{1}$$

Although we cannot precisely yield some statistical properties of RUP, we can simulate drawing samples according to RUP. The simulation results as shown in Fig. 2, reveal that RUP tends to use much less segments to make up a partition than the traditional HSMM as the document length increases, which explains why RUP is useful for avoiding rapid switching.

Note that RUP puts no explicit constraint on the number of segments in the resulting partition, allowing the number of segments to take a value from 1 to S. For unstructured texts with very different content structures, it is difficult and time-consuming to determine their numbers of segments in advance. And since RUP is capable of generating partitions of different lengths encouraging partitions containing a relatively small number of segments, RUP provides a convenient prior here.

3.3 RUP-HDP-HSMM

RUP-HDP-HSMM combines the RUP and HDP-HSMM,

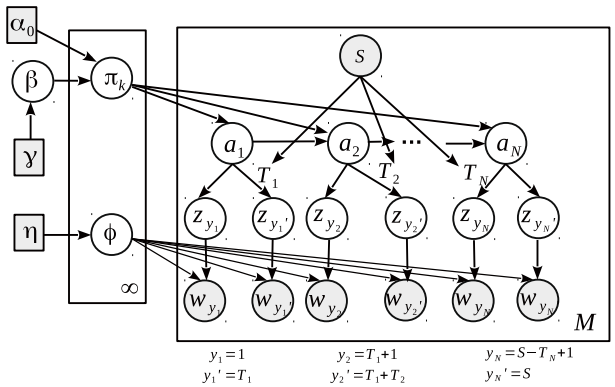


Fig. 3 The graphical representation of RUP-HDP-HSMM. The only one parameter that RUP takes for generating a partition of the text is the text length, denoted by S in the graph, which is observable and hence shaded. However, the partition generated by the RUP are unobserved.

with RUP responsible for generating the partition, and HDP-HSMM for generating global topics and topic allocation in each text. As with HDP-HSMM, RUP-HDP-HSMM also requires the diagonal elements of the transition matrix to be zeros, i.e., $\pi_{kk} = 0$ for all k. The graphical representation of the RUP-HDP-HSMM is shown in Fig. 3, and the corresponding generative process is given as follows:

1. Generate transition probabilities
   a. Draw $\beta \sim \text{GEM}(\gamma)$
   b. Draw $\pi_k \sim \text{DP}(\alpha_0, \beta),\ k = 1, 2, \cdots$
   c. Map π_k to $\bar{\pi}_k$ with the transform $\bar{\pi}_{ij} = \frac{\pi_{ij}}{1-\pi_{ii}}(1 - \delta_{ij}), j = 1, 2, \cdots$
2. Draw multinomial parameters $\phi_k \sim \text{Dirichlet}(\eta), k = 1, 2, \cdots$
3. For each document $d_i, i = 1, \cdots, M$
   a. Draw a partition $P_i = (T_1, T_2, \cdots, T_N) \sim \text{RUP}(S)$
   b. For each segment $s = 1, 2, \cdots, N$
      i. Draw a hidden state $a_s^i \sim \bar{\pi}_{a_{s-1}^i}$
      ii. Draw T_s words in the s-th segment independently according to Multinomial($\phi_{a_s^i}$)

where GEM denotes a stick breaking process [15] and DP(α_0, β) denotes a Dirichlet process with a base probability distribution β and a strength parameter α_0 [5]. According to this generative process, The hidden variables involved include $\{\beta, \pi, \phi, \mathbf{T}, \mathbf{a}\}$; hidden topics $\mathbf{z}$ in Fig. 3 will be automatically fixed, as long as $\mathbf{T}$ and $\mathbf{a}$ are drawn.

3.4 Model Inference

Gibbs sampling algorithms are frequently used for model inference in nonparametric Bayesian models. We give two Gibbs samplers for RUP-HDP-HSMM here—a direct Gibbs sampler (DG) and a forward-backward Gibbs sampler (FB)—both based on the weak-limited Gibbs sampler [4].

In a weak-limited Gibbs sampler, the HDP prior is approximated with two finite dimensional Dirichlet distributions having the same dimension; the dimension is called truncation level, denoted by L. In particular, we have

$$\beta \sim \text{Dirichlet}(\gamma/L, \cdots, \gamma/L), \tag{2}$$

and

$$\pi_k \sim \text{Dirichlet}(\alpha_0\beta_1/L, \cdots, \alpha_0\beta_L/L). \tag{3}$$

There is the theoretical guarantee that the resulting finite prior converges in distribution to the true HDP prior, as $L \to \infty$. As for the probabilities of initial hidden states, we can assume an unknown multinomial distribution generated from its prior β, denoted by π_0. Sampling β in both DG and FB is the same as that in HDP-HSMM, and hence is omitted here (see [4] for details).

3.4.1 Direct Gibbs Sampler

The direct Gibbs sampler (DG) samples each hidden state z_j^i at a time, from the conditional distribution $p(z_j^i = k|\mathbf{z}^{-(i,j)}, \beta)$, where $\mathbf{z}^{-(i,j)}$ denotes $\mathbf{z}$ with z_j^i excluded. When we compute $p(z_j^i = k|\mathbf{z}^{-(i,j)}, \beta)$, hidden variables π and ϕ can be marginalized out, because of closed-form posterior predictive distributions for multinomial parameters π_k and ϕ_k, both having Dirichlet priors.

$p(z_j^i = k|\mathbf{z}^{-(i,j)}, \beta)$ can be computed as follows

$$p(z_j^i = k|\mathbf{z}^{-(i,j)}, \beta) \propto \underbrace{\frac{\alpha\beta_k + n_{x_1,k}}{\alpha + n_{x_1,\cdot}}}_{\text{left-transition}} \cdot \underbrace{\frac{\alpha\beta_{x_2} + n_{k,x_2}}{\alpha + n_{k,\cdot}}}_{\text{right-transition}} \times \underbrace{p(P(\mathbf{z}_i^{-j}, z_j = k))}_{\text{partition}} \cdot \underbrace{p(w_j|z_j = k)}_{\text{emission}}, \tag{4}$$

where x_1 and x_2 denote the hidden states of the segments preceding and following the segment that contains z_j respectively.

The first and second terms of Eq. (4) correspond to the posterior predictive probabilities of left-transition from x_1 to k, and right-transition from k to x_2, where $n_{x,y}$ denotes the number of transitions from x to y, and $n_{x,\cdot}$ represents the total number of transitions from x, in all texts. The third term is the probability of the partition of d_i, resulted from letting $z_j^i = k$ yet keeping the other hidden states in d_i unchanged. We can compute $p(P(\mathbf{z}_i^{-j}, z_j = k))$ according to Eq. (1), by finding the corresponding partition represented by a sequence of hidden states, as illustrated in Fig. 4. The fourth term can either be directly computed by integrating out ϕ_k, or takes the value ϕ_{k,w_j}. Note that if $z_j^i = k$ leads w_j^i to be contained in the first segment of the text, the left-transition probability should be replaced with $\frac{\alpha\beta_k + m_k}{\alpha + M}$, where m_k represents the number of texts with starting segment of topic k, including the current text d_i.

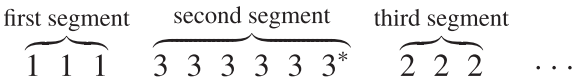


Fig. 4 An illustration of mapping a sequence of hidden states to the corresponding partition. Each number represents the hidden state of a word, with the one marked with an asterisk being the current state being considered, namely $z_j^i = 3$. z_j^i taking different values would lead to different partitions.

3.4.2 Forward-Backward Gibbs Sampler

In addition to DG, we develop the forward-backward Gibbs sampler (FB) here. While DG updates each hidden state z_i^j at a time, FB draws a sequence of hidden states each time. After sampling the hidden states, we need to explicitly sample ϕ and π.

Given π and ϕ, we show the forward recursive algorithm for computing the probability $p(\mathbf{w}_i|\phi,\pi)$ in RUP-HDP-HSMM, based on which a sequence of hidden states in d_i can be drawn. Note that Eq. (1) is important, because the probability of all possible partitions of a text can be computed. Based on the results of the forward algorithm, we can sample the whole sequence of hidden states $\mathbf{z}_i$ in d_i at a time. Inspired by recursive algorithm given in [16], we give the modified version using RUP to generate duration times of hidden states for a text. Let

$$\begin{aligned}\alpha(m,k) &= p(w_{m+1},\ldots,w_S|z_m = k \neq z_{m+1}) \\ &= \sum_{z_{m+1},\ldots,z_S} p(z_{m+1},\ldots,z_S,w_{m+1},\ldots,w_S|z_m = k \neq z_{m+1}).\end{aligned} \tag{5}$$

We have, for $m \leq S-2$

$$\alpha(m,k) = \sum_{j\neq k}\Big\{\pi_{kj}\Big\{\sum_{n\in\{m+1,\ldots,S-1\}}\{\frac{1}{S-m}p(w_{m+1},\ldots,w_n|\phi_j)\alpha(n,j)\} + \frac{1}{S-m}p(w_{m+1},\ldots,w_N|\phi_j)\Big\}\Big\}, \tag{6}$$

and

$$\alpha(S-1,k) = \sum_{j\neq k}\pi_{kj}p(w_S|\phi_j). \tag{7}$$

In order to effectively compute $\alpha(m,k)$, we give an explicit implementation of the algorithm.

Step 1:
- a. Initialize $\gamma(k) \leftarrow p(w_S|\phi_k)$
- b. Compute $\alpha(S-1,k) \leftarrow \sum_{j\neq k}\pi_{kj}\gamma(j)$
- c. Initialize $\xi(S-1,k) \leftarrow \frac{1}{2}p(w_{S-1}|\phi_k)\alpha(S-1,k)$

Step 2: For $n \in \{S-1,\ldots,1\}$
- a. Update $\gamma(k) \leftarrow \gamma(k)p(w_n|\phi_k)\frac{S-n}{S-n+1}$
- b. For $m \in \{n+1,\ldots,S-1\}$ Update $\xi(m,j) \leftarrow \xi(m,j)p(w_n|\phi_j)\frac{m-n}{m-n+1}$
- c. If $n > 1$, compute $\alpha(n-1,k) \leftarrow \sum_{j\neq k}\Big\{p_{kj}\Big[\sum_{m=n,\ldots,S-1}\xi(m,j)+\gamma(j)\Big]\Big\}$
- d. If $n > 1$, initialize $\xi(n-1,j) \leftarrow h_j(n-1,n-1)p(w_{n-1}|\phi_j)\alpha(n-1,j)$

Step 3: Compute $p(w_1,\ldots,w_S) = \sum_k\Big\{\pi_{0,k}\Big[\sum_{m=1,\ldots,S-1}\xi(m,k)+\gamma(k)\Big]\Big\}$

Having run the forward algorithm, we can draw the fist segment and its hidden state. Note that, for every possible pair (m,k), the joint probability of $(T_1 = m, a_1 = k, w_1,\cdots,w_S)$ corresponds to a term in the summation computing $p(w_1,\cdots,w_S)$. With $(T_1 = m, a_2 = k)$ drawn from the last step, we can recursively draw the next segment by referring to $\alpha(m,k)$. This process can be repeated, until the last segment that contains the last word w_S is determined, completing the sampling of a sequence of hidden states. Updating ϕ_k and π_k is simple, and can be performed as follows

$$\phi_{k,l} \propto \#\{\mathbf{z} = k, \mathbf{w} = l\} + \eta,\ l \in \{1,\cdots,V\}, \tag{8}$$

$$\pi_{x,y} \propto \alpha\beta_y + n_{x,y},\ x,y \in \{1,\cdots,L\}, \tag{9}$$

and

$$\pi_{0,x} \propto \alpha\beta_x + m_x. \tag{10}$$

4. Applications and Empirical Results

Modeling content structures of domain-specific texts by RUP-HDP-HSMM leads to different applications. In this section, we show three distinct tasks—including text retrieval, classification, and text mining tasks—all depending on the content structures discovered in domain-specific texts with RUP-HDP-HSMM. In the text retrieval and classification tasks, we map the texts into the local feature space, with RUP-HDP-HSMM applied as a new feature extraction technique. In the text mining task, we propose a method of automatically finding the meaningful local contexts from the positive instances, which explain why they are classified as positive instances. This method only relies on the statistical information underlying the current data set, and therefore can be generally applied to other real-world data sets. As illustrated in our experiments, for text retrieval and classification tasks, RUP-HDP-HSMM, benefiting from the accurate modeling of content structures, provides more accurate text representations than other existing methods, like LDA and HDP-HSMM. Besides, the proposed text mining method also yields insightful results on our data set.

4.1 Text Retrieval Task

Representing texts in the local feature space based on their content structures is a direct application of RUP-HDP-HSMM. The goal of task retrieval experiments is to compare the performance of different methods representing texts in different feature spaces.

We first derive the representation of texts in the local

feature space based on their content structures found with RUP-HDP-HSMM, and then provide a measure of computing the "distance" between any two texts.

LDA† models texts using latent topics, and the explicit representation of a text is its topic distribution θ in the latent topic space. In order to estimate θ, for each text, the γ^* parameters in the variational distribution are inferred with a variational method [8]. The normalized γ^* can be viewed as an estimate to the topic distribution θ in LDA. Similarly, for HMM/HSMM-based algorithms—including RUP-HDP-HSMM and HDP-HSMM—can also represent texts in terms of underlying local features, which are composed of sequential word tokens. We count the number of words assigned different local features in all local contexts in each text, and record the result with a vector $\mathbf{v}$, with the k-th element being the total number of words generated by the k-th local feature. Then we update $\mathbf{v}$ by adding a small smoothing factor to it. The normalized $\mathbf{v}$ thus gives a representation of the text, defined over the space of local feature distributions. In specific, for the k-th local feature, we define

$$v_k = \alpha_0\beta_k + \#\{\text{words assigned the } k\text{-th topic}\}, \tag{11}$$

where $\alpha_0\beta_k$ serves as the smoothing factor. Normalizing v, we obtain a local feature distribution, which is also denoted by θ. The same use of notation θ as that in LDA should not cause confusion here.

We measure the "distance" between any two texts, which is the difference between their topic distributions, denoted by θ_1 and θ_2, with respect to the symmetric Kullback-Leibler divergence (KL divergence). In particular, the distance, denoted by $D(\theta_1\|\theta_2)$, is computed as follows

$$D(\theta_1\|\theta_2) = 0.5D_{KL}(\theta_1\|\theta_2) + 0.5D_{KL}(\theta_2\|\theta_1), \tag{12}$$

where $K_{KL}(\theta_1\|\theta_2)$ denotes the Kullback-Leibler divergence of θ_2 from θ_1, and

$$D_{KL}(\theta_1\|\theta_2) = \sum_k \theta_{1k} \log \frac{\theta_{1k}}{\theta_{2k}}. \tag{13}$$

The text retrieval tasks are performed using $D(\theta_1\|\theta_2)$, after mapping texts to the topic space in LDA and the local feature space in RUP-HDP-HSMM and HDP-HSMM.

4.1.1 Experiment Settings

The data set we use in this experiment contains 500 editorials collected from famous Japanese newspapers††, among which 200 instances, forming a subset C_1, discuss the impact of closing the nuclear plants in Japan and/or make comments on whether the nuclear power plants in Japan should be restarted. The others, however, may contain nuclear-related content, but do not relate to restart of the nuclear power plants. This subset is denoted by C_2. All editorials involved result in a corpus of 500 texts containing 320,951 tokens in total, and a vocabulary having 11,079 unique terms. The sample mean of the numbers of words is 641.9, and the standard deviation is 198.7.

For the text retrieval task, given a text d in C_1 as a query, we want to collect other texts that are also in C_1. If the texts are properly modeled with some method, given d in C_1, the remaining texts in C_1 should be closer to d than those in C_2. We use the mean average precision (MAP) and mean reciprocal rank (MRR) as evaluation metrics [19] for comparisons of text retrieval performance based on different methods. The MAP value is the arithmetic mean of average precision values for each text in C_1. The MRR is the arithmetic mean of the reciprocal ranks of results for each query d in C_1; the reciprocal rank of d is the multiplicative inverse of the rank of the first relevant text in the ranked retrieval result, where however d itself is excluded. The MRR only focuses on the first relevant text, and thus especially useful for evaluation of responses where there is only one correct answer. In contrast, MAP, taking account of all relevant texts in the ranked retrieval result, has been shown to have good discrimination and stability.

Existing methods, including HDP-HSMM models with Poisson and negative Binomial duration distributions, "bag-of-words" model (BOW) and LDA are used for comparison in our experiment. In RUP-HDP-HSMM, we set η to 0.1, and put gamma priors for the concentration parameters γ and α_0; in particular, $\gamma \sim \text{Gamma}(1,1)$ and $\alpha_0 \sim \text{Gamma}(1,0.1)$, where $\text{Gamma}(g_a, g_b)$ denotes a Gamma distribution with the mean equal to g_a/g_b. The same settings are also used in HDP-HSMM with Poisson and negative Binomial duration distributions, denoted by HDP-HSMM (Poisson) and HDP-HSMM (NBin), respectively. For Poisson duration distributions in HDP-HSMM (Poission), we put a Gamma prior Gamma(1, 0.1). In HDP-HSMM (NBin), the negative Binomial duration distributions, denoted by $\text{NBin}(r,p)$, have a Beta prior $\text{Beta}(1,1)$ over p, and r is fixed to 10. The truncation level L in these HDP-HSMM-based algorithms is set to 200. Because LDA requires the number of topics K to be a fixed model parameter, we obtain different models, by letting K take the value from $\{20, 40, 60, 80, 100\}$.

4.1.2 Experiment Results

The experiment results of $\{G_d,\ d \in C_1\}$ based on different methods are summarized in Table 1. We view the number of topics in LDA as the number of features. The number of features in a HDP-HSMM-based algorithm is the number of local features, after the Gibbs sampling mixes.

As we see from Table 1, all methods can achieve rather high MRR values, while there are clear differences among different methods in terms of their MAP values. RUP-HDP-HSMM achieves the highest MAP value, hence the best retrieval performance of finding all relevant texts. In particular, we can see that HDP-HSMM-based methods can yield

†LDA software can be downloaded from http://www.cs.princeton.edu/~blei/lda-c/

††The data set can be found following the instruction at https://github.com/yuui-ro/JapaneseEditorialDataset/blob/master/dataset-1

Table 1 Performance of collecting relevant instances with different methods

Method	Number of Features	Mean average precision	Mean reciprocal rank
RUP-HDP-HSMM	32	0.85	0.95
HDP-HSMM (Poisson) [4]	53	0.75	0.95
HDP-HSMM (NBin) [4]	58	0.75	0.94
BOW [17]	11,079	0.68	0.97
LDA [18]	20	0.71	0.94
	40	0.61	0.90
	60	0.56	0.91
	80	0.64	0.88
	100	0.49	0.90

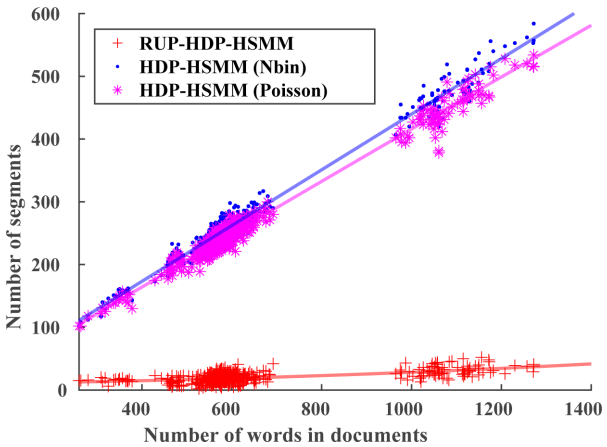


Fig. 5 Plot of number of segments in texts yielded by different methods, with regression on means at different numbers of words

higher MAP values than LDA and BOW. This is not a surprising result, considering that both LDA and BOW ignore the local contexts in the learning texts that are closely related to the topic of nuclear power plants in Japan. From the text-level perspective, the difference between texts may be small, because there are also nuclear-related texts in C_2, sharing some feature words of texts in C_1. In contrast, HDP-HSMM-based methods focus on local patterns in texts, which can exist because of the same domain of the texts. Consequently, it is easier for HDP-HSMM-based methods to retrieve the other texts in C_1 for a query.

Moreover, we can observe that RUP-HDP-HSMM gives less number of local features than HDP-HSMM (Poission) and HDP-HSMM (NBin). On the other hand, HDP-HSMM (Poission) and HDP-HSMM (NBin) yield similar numbers of local features. A possible explanation is that both HDP-HSMM (Poisson) and HDP-HSMM (NBin) cannot prevent rapid switching, thus creating extra hidden states. We plot the number of segments in texts found by different methods in Fig. 5. We also perform a quadratic regression on the means of the numbers of segments, as the number of words changes. Figure 5 shows that, on average, HDP-HSMM (Poisson) and HDP-HSMM (NBin) lead a text to contain much more segments than RUP-HDP-HSMM. And the number of segments produced by HDP-HSMM (Poisson) and HDP-HSMM (NBin) increases very fast, indicating that the resulting hidden states switch very fast between one another. By contrast, RUP-HDP-HSMM can successfully prevent the rapid switching, because RUP-HDP-HSMM uses much less segments to model local contexts in a text, while retaining the ability to find useful local features to represent the content structures in texts.

4.2 Text Classification Task

In this section, we consider text classification using SVM, where all texts are classified to two exclusive classes. The input for training SVM classifiers can be vectors based on "bag-of-words" model, or vectors of smoothed counts of words in the feature space obtained with RUP-HDP-HSMM and HDP-HSMM as defined in Eq. (11).

With different feature representations, we train different SVMs and compare their classification performances in terms of F1 scores. We aim to show that local features discovered with RUP-HDP-HSMM can provide useful features for the text classification task.

4.2.1 Experiment Settings

The data set we use here is a subset of the data used in the text retrieval experiment, consisting of 100 editorials[†], all related to nuclear power plants in Japan. This data set results in 4,354 terms and 61,220 word tokens in total.

We conduct four classification experiments, each involving a binary classification task. In particular, we extracted four different opinions that are often referred to in the data set. Thus, for each opinion, according to its presence in a text, the text is classified as a positive or negative training instance; all class labels are assigned manually by three college students under majority rule. Each opinion has 17, 29, 27 and 22 positive training instances, respectively. The four classification experiments are performed independently.

As discussed earlier, HDP-HSMM-based methods and LDA can represent a text with real-valued vectors $\mathbf{v}$ and γ^* respectively in the latent spaces.

For each classification experiment, indexed by the involved opinion number, we randomly split the data set into training data and test data, each containing both positive and negative instances. This process is repeatedly performed, in

[†]The data set can be found following the instruction at https://github.com/yuui-ro/JapaneseEditorialDataset/blob/master/dataset-2

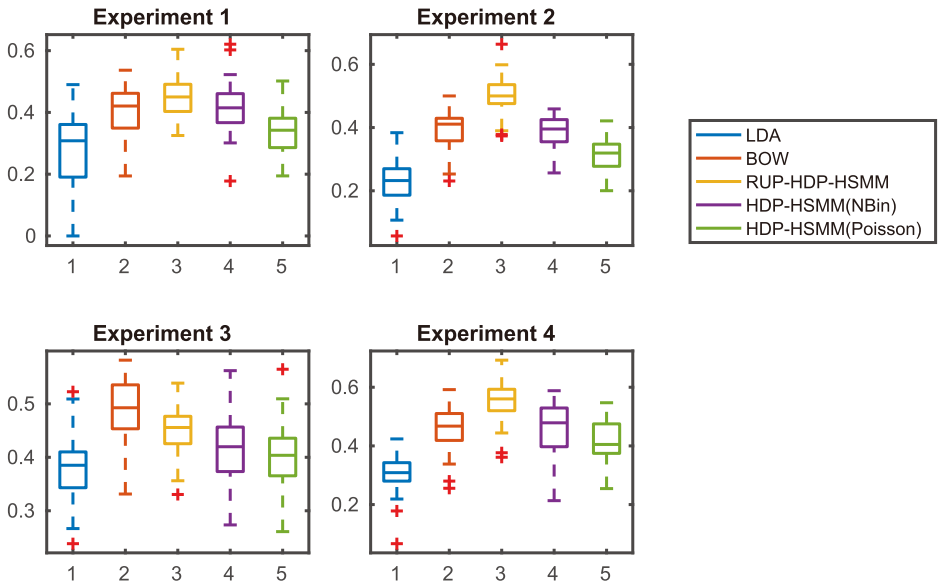


Fig. 6 Box plots of F1 scores in four binary classification tasks based on different feature selection methods. On each box, the central mark is the median, and the edges of the box are the 25th and 75th percentiles. The plus marks denotes the data points considered as the outliers, if they are larger than Q3+1.5*(Q3-Q1) or smaller than Q1-1.5*(Q3-Q1), where Q1 and Q3 are the 25th and 75th percentiles, respectively.

order to give an estimate of the performance of resulting SVM classifiers. The LIBSVM software[†] is used for training a classifier using the training data. The linear kernel is utilized, and the cost parameter C and weight parameter wi are tuned, representing how much we want to avoid misclassifying the training data, and the weight for a positive class, respectively. In particular, the C parameter is selected from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{2}, 10^{3}, 10^{4}\}$, and wi can take a positive value in $\{1, 2, \cdots, 10\}$. The performance of classification is evaluated using the average F1 score, computed by performing five-fold cross validation on our data set. Five subsamples used in the five-fold cross validation are obtained by randomly partitioning the original data set; however, it is guaranteed that there are both positive and negative instances in each subsample. The tuning parameters are chosen by performing five-fold cross validation only on the training data set, and then applied to prediction on the testing data set.

The configuration of the parameters involved in RUP-HDP-HSMM and HDP-HSMM remains the same as in the text retrieval experiment. The topic number of LDA is set to 20.

4.2.2 Experiment Results

We perform five-fold cross validation 100 times, each time the partitions of training and testing data sets are randomly selected. The average F1 score is computed after each cross validation, as an indicator of classification performance. All results based on different feature extraction methods are summarized in Fig. 6, from which we can see that except Experiment 3, the SVM classifiers yielded by RUP-HDP-HSMM give the highest medians, and show relatively stable performance. This result tells us that the extracted local features in texts can serve as another form of features that can be useful for text classification, in addition to the "bag-of-words" representation.

Besides, BOW shows relatively good performances here too, indicating that, for each experiment, there may exist some very important feature words. If the SVM is trained by making good use of the presence information of these feature words, the resulting classifier can also be competitive. By contrast, LDA cannot lead to powerful classifiers here, because both the local features and feature words are not properly captured.

Furthermore, it is interesting to see that although HDP-HSMM (Poisson) and HDP-HSMM (NBin) perform better than LDA and BOW in the text retrieval task, they perform worse than BOW here. As discussed earlier, rapid switching between hidden states still occurs with HDP-HSMM (Poisson) and HDP-HSMM (NBin), and can prevent them from discovering true local patterns underlying the texts. Therefore, the local features that are closely related to an opinion may be missed by HDP-HSMM (Poisson) and HDP-HSMM (NBin), though, to some extent, they are capable of grouping similar texts together by using the temporal information underlying the texts.

4.3 Text Mining Task

In our text classification experiments, each binary classification corresponds to the prediction of a particular opinion's presence. We can use a label to represent an opinion, only

[†]https://www.csie.ntu.edu.tw/˜cjlin/libsvm/

Table 2 Weight vectors of local features obtained from SVM.

Label No.	Ordered local feature with weight [*Feature No.* (*Weight*)]				
	1	2	3	4	5
1	8 (0.0195)	32 (0.0089)	2 (0.0050)	18 (0.0045)	14 (0.0031)
2	8 (0.0170)	2 (0.0094)	18 (0.0081)	16 (0.0064)	11 (0.0022)
3	2 (0.0295)	6 (0.0157)	14 (0.0082)	19 (0.0058)	27 (0.0044)
4	6 (0.0158)	28 (0.0089)	7 (0.0073)	21 (0.0071)	3 (0.0064)

Feature 2	Feature 6	Feature 8	Feature 19	Feature 21	Feature 28	Feature 32
committee 委員	Abe 安倍	thermal power 火力	electricity power 電力	judgment 判断	Fukushima 福島	Japanese Yen 円
meeting 会	*	power generation 発電	power-saving 節電	evaluate 評価	accident 事故	%
regulation 規制	pro and con 賛否	fuel 燃料	electricity 電気	fault 断層	No. 第	around 約
nuclear power 原子力	Shinzō 晋三	gas ガス	needs 需要	active 活	one 一	one hundred million 億
more 長	*	cost 費	supply 供給	prime minister 首相	evacuation 避難	a billion 兆
member 委	ground 地盤	import 輸入	electricity rate 料金	*	tsunami 津波	year 年
nuclear power plant 原発	ex- 元	station 所	summer 夏	opinion 意見	nuclear power plant 原発	not ない
*	lead 率いる	natural 天然	family 家庭	investigation 調査	operation 稼働	*
Tanaka 田中	leader 首脳	wind energy 風力	supply and demand 需給	political 政治	re- 再	nuclear power plant 原発
*	talk 会談	regenerative 再生	company 会社	public opinion 世論	electricity 電力	*

Fig. 7 Representative terms in some typical local features. The terms of a local feature are sorted according to Eq. (14) in descending order. We show both the original Japanese terms and their translations. An asterisk (*) represents a function word in Japanese, and is not explicitly shown here, due to difficult interpretation to English.

assigned to the positive instances. Besides prediction, we may also wish to discover those text fragments that are in close relation to a label. The discovered fragments not only describe this label, but also account for why a text is classified as a positive instance. In this section, we present a text mining method of automatically finding such text fragments, which are modeled using local contexts with RUP-HDP-HSMM.

4.3.1 Measuring Local Contexts

Two kinds of metric are utilized in our method, one for evaluating the importance of a local feature to a label and one for measuring to what extent a particular term distinguishes a local feature from the others.

The first kind of metric evaluates the relation between a particular label and local features. The weight vector derived from a linear SVM or the logistic regression can serve the purpose here. The former is used here, and we denote the resulting vector by $\mathbf{u} = (u_1, \cdots, u_K)$, where K is the number of local features. The input of the linear SVM is the local feature vectors $\mathbf{v}$, as obtained in the text classification task, with each element being a smoothed version of number of words assigned a particular local feature. A positive u_k means that the k-th local feature positively contributes to the presence of the label, and vice versa. The absolute value of u_k stands for the degree of this contribution.

The second kind of metric models the relation between terms and local features. Note that we cannot directly use ϕ_k, because it cannot reflect the importance of terms when we compare the k-th local feature against the others. Rather, we compute the mutual information $MI(A_w, B_t)$ of two Binomial random variables A_w and B_t, associated with term w and local feature t, respectively. $MI(A_w, B_t)$ measures the dependence between local feature t and term w, thus representing how important term w is for local feature t to be distinguished from other local features. For any local context discovered with RUP-HDP-HSMM, we let $A_w = 1$ if term w is present in this local context, and $A_w = 0$ otherwise. If this local context is assigned local feature t, we let $B_t = 1$, and otherwise $B_t = 0$.

According to the definition of mutual information and A_w and B_t, we have

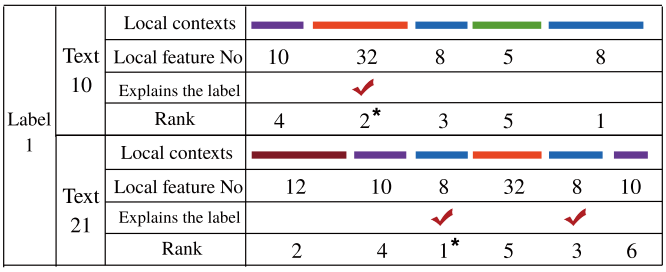


Fig. 8 An illustration of finding the rank of the first relevant local context in positive instances of Text 10 and 21 with respect to Label 1. We use line segments with different colors to represent local contexts assigned different local features in a text. Here the ranks of local contexts in Text 10 and 21 are obtained by computing their scores with respect to Label 1. Local contexts attached check marks are considered to explain the label. The rank of the first relevant local context is marked with an asterisk.

Table 3 Summary of the ranks of the first relevant local contexts in positive instances

Label No.	Mean	Median	15th percentile	85th percentile
1	1.76	1	1	2
2	5.84	5	2	10
3	3.08	2	1	6
4	3.26	1	1	8

Table 4 Examples of the first relevant local contexts for each label

Label	Label description	Example of the first relevant local contexts
1	The rising cost of power generation from fossil fuels exerts a negative impact on Japan's economy	Import of natural gas increases rapidly, and enormous consumption of fossil fuels causes an obvious harmful effect to Japan's economy 全国で膨大な火力発電燃料を消費している弊害も大きい 。液化天然ガスなど燃料の輸入が急増 する
2	Increasing electricity price causes inconvenience to individuals and corporations	Delayed restart of nuclear power plants will push electricity power companies to raise electricity rate again to face their difficult financial situations. 再稼働が遅くなるほど、電力会社の経営が 苦しくなり電気料金 の再 値上げにつながるのも事実だ
3	Nuclear power plants are needed for stable supply of electricity	In areas in short supply of electricity, electricity power companies must cooperate to prepare for unexpected emergencies. 電力不足 に陥った地域に、余裕のある電力会社が 電力を融通する体制を強化するなど、業界をあげて不測の事態に備える必要がある
4	Safety of nuclear power plants cannot be ensured	Even experts express their concerns about the earth fault. It should not be allowed to start nuclear power plants, unless more evidence is provided. 関電や政府は問題ないと判断する再稼働を決定した専門家会合でも活断層に否定的な見方があった。しかし今回示された資料だけでは全体像がわからないという明確な判断がつかないまま原発を稼働している現状は容認できない

$$MI(A_w, B_t) = \sum_{x\in\{0,1\}} \sum_{y\in\{0,1\}} p(A_w = x, B_t = y) \times \log \frac{p(A_w = x, B_t = y)}{p(A_w = x)p(B_t = y)}. \quad (14)$$

After obtaining the content structures of texts with RUP-HDP-HSMM, all the terms involved in Eq. (14) can be estimated. For example, $p(A_w = 0, B_t = 1)$ can be approximated by the proportion of the local contexts that are assigned local feature t and does not contain term w. Note that $p(A_w = x) = p(A_w = x, B_t = 0) + p(A_w = x, B_t = 1)$ and $p(B_y = y) = p(A_w = 0, B_t = y) + p(A_w = 1, B_t = y)$.

Combining the two kinds of metric, for the j-th local context that is assigned local feature t and consist of words $\{w_{T_j}, \cdots, w_{T_{j+1}-1}\}$, we compute its score in terms of its contribution to the label, as follows

$$\text{score}(t) = u_t \times \sum_{n=T_t,\cdots,T_{t+1}-1} MI(A_{w_n}, B_t). \quad (15)$$

4.3.2 Experiment Results

We use the same data set as in the text classification task, and wish to find the important local contexts from the positive instances in each binary classification.

Using the local feature vectors as the input, we train a linear SVM classifier, obtaining the weight vector of local features. We show the top five local features according to their weights in each classification task in Table 2. The cost parameter C and weight parameter wi are again determined by five-fold CV with all instances randomly split into five subsamples. The ranges of C and wi are the same as in the text classification task.

The representative terms of seven local features are shown in Fig. 7. Due to space limitations, we cannot present all local features. The ones in Fig. 7 are selected, because they achieve relatively large weights in the weight vectors,

with rather straightforward interpretations.

It is worth noting that showing the top terms w that have the largest conditional probabilities $\phi_{k,w}$ is far from as insightful as Figure 7, mainly because of the high-frequency words that commonly appear in all local contexts. Such high-frequency words include function words and some domain-specific feature words, and would lead to high conditional probabilities in all ϕ_k. Consequently, it would be difficult for us to interpret these local features, if we only look at a few terms having the highest conditional probabilities in ϕ_k.

For each positive instance, all its local contexts are sorted by their scores with respect to a label in descending order, which are computed according to Eq. (15). If our method works well, the relevant local contexts would have high ranks in the ranked list. To evaluate the performance of the proposed method, we examine the rank of the first relevant local context in each positive instance (an illustration is given in Fig. 8). The results are summarized in Table 3. The 15th percentiles in all cases are either 1 or 2, indicating that the relevant local contexts we desire can be well identified in some positive instances. Besides, both the means and medians for Label 1, 3 and 4 are small, meaning that on average we can target a relative local context in a text from the top 3 local contexts in the ranked list.

In Table 4, we show an example of relevant local contexts actually discovered for each label, along with the label description. Combining Table 2 and Table 4, we can see that Label 1 and Label 2, despite being separate labels, address the impact of shutting down nuclear power plants from different aspects, thus having a strong positive correlation. This explains why our method performs much better for Label 1 than for Label 2.

5. Conclusion

In this paper, we proposed the RUP-HDP-HSMM, which is a nonparametric Bayesian model built upon the HDP-HSMM, for modeling the content structure of domain-specific texts. RUP-HDP-HSMM incorporates RUP to tackle the rapid switching.

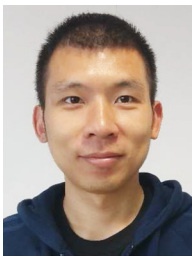
In our text retrieval experiments, we showed that RUP-HDP-HSMM could properly model the content structures of domain-specific texts, in the light of the experiment results on our data set. In particular, RUP-HDP-HSMM successfully avoided the rapid switching between hidden states, while typical HDP-HSMMs did not. Besides, RUP-HDP-HSMM greatly outperformed the others in terms of ranking performance, while using less local features than HDP-HSMM.

We also demonstrated that modeling content structures of texts could provide useful features for text classification tasks. In our experiments, the content structures discovered with RUP-HDP-HSMM could lead SVM to achieve higher F1 scores than other methods.

Furthermore, depending on RUP-HDP-HSMM, we presented a method of extracting the relevant local contexts, accounting for why a text is classified as a positive instance. We deem this method efficient at finding the relevant local contexts, only relying on the statistical information underlying the current data set. We also consider it a promising tool for discourse analysis, automatic summarization, etc., in NLP. A specific application of our method remains as future work.

References

[1] Z. Harris, Sublanguage: Studies of Language in Restricted Semantic Domains, Walter de Gruyter, Berlin; New York, 1982.
[2] A. Wray, Formulaic language and the lexicon, Cambridge University Press, 2002.
[3] R. Barzilay and L. Lee, "Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization," HLT-NAACL 2004: Proceedings of the Main Conference, pp.113–120, 2004.
[4] M.J. Johnson and A.S. Willsky, "Bayesian nonparametric hidden semi-markov models," The Journal of Machine Learning Research, vol.14, no.1, pp.673–701, 2013.
[5] Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei, "Hierarchical dirichlet processes," Journal of the American Statistical Association, vol.101, no.476, pp.1566–1581, 2006.
[6] E.B. Fox, E.B. Sudderth, M.I. Jordan, and A.S. Willsky, "A sticky HDP-HMM with application to speaker diarization," The Annals of Applied Statistics, vol.5, no.2A, pp.1020–1056, 2011.
[7] P. Van Mulbregt, I. Carp, L. Gillick, S. Lowe, and J. Yamron, "Text segmentation and topic tracking on broadcast news via a hidden markov model approach," 1998.
[8] D.M. Blei and P.J. Moreno, "Topic segmentation with an aspect hidden markov model," Proc. 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp.343–348, ACM, 2001.
[9] S.-Z. Yu, "Hidden semi-Markov models," Artificial Intelligence, vol.174, no.2, pp.215–243, 2010.
[10] A. Gruber, Y. Weiss, and M. Rosen-Zvi, "Hidden topic Markov models," International conference on artificial intelligence and statistics, pp.163–170, 2007.
[11] H. Chen, S. Branavan, R. Barzilay, and D.R. Karger, "Content modeling using latent permutations," Journal of Artificial Intelligence Research, vol.36, no.1, pp.129–163, 2009.
[12] I. Titov and R. McDonald, "Modeling online reviews with multi-grain topic models," Proc. 17th international conference on World Wide Web, pp.111–120, ACM, 2008.
[13] M. Purver, T.L. Griffiths, K.P. Körding, and J.B. Tenenbaum, "Unsupervised topic modelling for multi-party spoken discourse," Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pp.17–24, Association for Computational Linguistics, 2006.
[14] L. Du, W.L. Buntine, and M. Johnson, "Topic segmentation with a structured topic model.," HLT-NAACL, pp.190–200, 2013.
[15] J. Sethuraman, "A constructive definition of dirichlet priors," Statistica sinica, pp.639–650, 1994.
[16] T. Economou, T.C. Bailey, and Z. Kapelan, "MCMC implementation for bayesian hidden semi-Markov models with illustrative applications," Statistics and Computing, vol.24, no.5, pp.739–752, 2014.
[17] Z.S. Harris, "Distributional structure," Word, vol.10, no.2-3, pp.146–162, 2015.
[18] D.M. Blei, A.Y. Ng, and M.I. Jordan, "Latent dirichlet allocation," the Journal of machine Learning research, vol.3, pp.993–1022, 2003.
[19] C.D. Manning, P. Raghavan, and H. Schütze, Introduction to information retrieval, Cambridge university press Cambridge, 2008.



Youwei Lu received B.E. and M.E. degrees in Software Engineering from Dalian University of Technology in 2010, and in Computational Intelligence and Systems Science from Tokyo Institute of Technology in 2013. He is currently a doctoral student in Tokyo Institute of Technology. His research interests include probabilistic topic models, approximate Bayesian inference and natural language processing.

Shogo Okada Shogo Okada received Ph.D. degree in Computer Science from the Tokyo Institute of Technology 2008. He worked in Kyoto University to 2011 as a project assistant professor. He visited IDIAP research institute, Switzerland as a visiting faculty in 2014. He worked in Tokyo Institute of Technology to 2016 as an assistant professor. Currently, he is an associate professor of School of Information Science, Japan Advanced Institute of Science and Technology.

Katsumi Nitta received his B.E., M.E. and Dr.Eng. from Tokyo Institute of Technology in 1975, 1977 and 1980, respectively. In 1980, he joined Electrotechnical Laboratory as a computer scientist, and from 1989 to 1994, he worked for Institute for New Generation Computer Technology. From 1996 to 2015, as a professor, he worked for the Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology. Currently, he is a professor of School of Computing, Tokyo Institute of Technology. His research interest includes the legal informatics and man-machine interaction. He is a member of the Information Processing Society of Japan (IPSJ), the Japanese Society for Artificial Intelligence (JSAI) and Association for Computing Machinery (ACM).