Topic-dependent N-gram models based on Optimization of Context Lengths in LDA

Akira Nakamura¹, Satoru Hayamizu²

¹ ECO Technology Research Center, SANYO Electric Co., Ltd., Japan
² Department of Information Science, Gifu University, Japan

naka@eco.rd.sanyo.co.jp, hayamizu@gifu-u.ac.jp

Abstract

This paper describes a method that improves accuracy of N-gram language models which can be applied to on-line applications. The precision of a long-distance language model including LDA is influenced by a context length, or a length of the history used for prediction. In the proposed method, each of multiple LDA units estimates an optimum context length separately, then those predictions are integrated and N-gram probabilities are calculated. The method directly estimates the optimum context length suitable for prediction. Results show the method improves topic-dependent N-gram probabilities, particularly of a word related to specific topics, yielding higher and more stable performance comparing to an existing method.

Index Terms: language model, topic model, LDA

1. Introduction

Topic-based language models such as pLSI [1], LDA [2] and DM [3] incorporate long-term dependencies among words based on topics, which represent underlying semantic structures in document collections. These models adaptively estimate the unigram probability of each word. N-grams can also be adapted to topics by combining with an interpolation algorithm (e.g. unigram-rescaling [4]), and hence the topic models are applied to continuous speech recognition [4], machine translation [5] and predictive text entry [6].

When a topic model is applied to the on-line applications such as predictive text entry and speech recognition, the model needs to successively adapt to the transition of latent topics in unknown input text (in this paper, we refer to this as on-line adaptation). The accuracy of on-line adaptation is influenced by a context length, or a length of the word sequence used for prediction. In most of related works, however, the context length is simply fixed to a certain number of words [3,6], or the sequence of all the words in a current document is employed as the context [4].

Boundaries between documents are not necessarily given in real applications, and topics may shift even in a document. It is therefore desirable that an appropriate context length is dynamically selected according to input text. A pioneering work [7] shows a method that integrates predictions with various context lengths, each of which length is sampled by a Monte Carlo particle. Although the method combined with DM decreases perplexity by 6-9%, only a slight improvement is obtained when combined with LDA. On the other hand, nonstationary LDA [8] recently proposed incorporates Markov chain to detect a segment corresponding to a specific writing style within a document.

In this paper, we propose a method that estimates the context length optimum for word prediction in more straightforward way. In our method, each of multiple LDA models that are independently trained estimates the context length and calculates a topic-dependent unigram probability separately, and then those predictions are integrated. The context length is estimated by evaluating various lengths based on the relevancy to the observed part of a current sentence. Experimental results show our method improves prediction accuracy with less computational costs comparing to an existing method.

2. LDA (Latent Dirichlet Allocation)

2.1. Overview of LDA

LDA [2] is a probabilistic document model that assumes the distribution θ over C latent topics (z₁, z₂, ..., zC) is given by the Dirichlet distribution Dir(θα) for each document. The probability of a document d = (w₁, w₂, ..., wD) is expressed by:

\[ p(d | α, β) = \prod_{i=1}^{C} \frac{\alpha \prod_{j=1}^{V} \beta_{wj}}{\alpha + \beta_{wj}} \times p(z_1) \times \prod_{j=1}^{V} \alpha_j \times \frac{\beta_{w_1}}{\beta_{w_1} + \beta_{w_2}} \times \frac{\beta_{w_2}}{\beta_{w_2} + \beta_{w_3}} \times \frac{\gamma_{z_1}}{\gamma_{z_1} + \gamma_{z_2}} \]

where α and β are the model parameters of LDA and β₀ denotes p(w₀ | z₀); the unigram probability of w₀ in topic z₀ (1 ≤ j ≤ V; V: vocabulary size). In this work, the parameters α and β, which can be estimated by VB(variational Bayes) [2] or MCMC(Markov Chain Monte Carlo) [9], are obtained using VB inference.

2.2. On-line adaptation in LDA

When a context h = (w₁, w₂, ..., wᵢ) is observed, topic-dependent unigram probability can be computed by:

\[ p(wᵢ | h) = \frac{\sum_{zᵢ} \gamma_{zᵢ} \beta_{wᵢ}}{\sum_{zᵢ} \gamma_{zᵢ}} \]

where γᵢ (1 ≤ j ≤ C), which are the variational parameters introduced in VB inference, represent topic mixture proportions p(zᵢ | h).

3. Proposed method

3.1. Optimization of context lengths

Figure 1 illustrates the procedure for optimizing context lengths in the proposed method. Here, wᵢ is a target word at a current time t, wᵢ, indicates the first word of the current sentence that includes wᵢ, and wᵢ, indicates the first word of the previous sentence (1 ≤ j ≤ K), respectively (i.e., sᵢ<sᵢ<sᵢ<sᵢ<sᵢ<sᵢ<sᵢ). Assuming all the words wᵢ (t ≤ j ≤ 1) are already observed, let us consider selecting one from the candidates \{wᵢ, wᵢ, wᵢ, wᵢ, wᵢ, wᵢ, wᵢ\} as the most suitable beginning of the history h.

Suppose wᵢ, is selected as bᵢ (= the beginning of h) and each word in the known part of the current sentence \{wᵢ, wᵢ, ..., wᵢ\} are sequentially predicted, the probability of wᵢ is expressed as:

\[ p(wᵢ | h = wᵢ) = \prod_{j=0}^{t-1} p(w_j | h = wᵢ) \]
where $p_{L}(w|h)$ denotes the topic-dependent unigram probability computed by LDA with a history $h$. The larger the probability obtained by Eq.(3), the more accurately the topic-dependent unigram probability only of the last word $w_t$ can be estimated with the history $h$. Expecting the current target $w_t$ is also likely to be estimated most precisely in such a case, we select the candidate that gives the largest probability by Eq.(3) as the optimum $b_h$ at the current time $t$. Thus, estimate the optimum beginning of the history $\hat{b}_h(t)$ by:

$$\hat{b}_h(t) = \arg\max_{w_{s_t}} p(w_{s_t-1} | b_h = w_{s_t})$$  \hspace{1cm} (4)

then compute the target’s topic-dependent unigram probability $p_{s_t}(w_t | h = \hat{b}_h(t) \cdots w_{s_t-1})$ with $\hat{b}_h(t)$.

Since containing iterations of VB inference, calculating $p_{s_t}(w_t | h = w_{s_t-1}^{m})$ in Eq.(3) for all words in $w_{s_t-1}^{m}$ requires much computational cost. However, as long as both $w_t$ and $w_{s_t-1}$ belong to the same sentence (i.e., unless $w_t$ is the first word of a sentence), by holding the probability $p(w_{s_t-2} | b_h = w_{s_t})$ calculated at $(t-1)$, Eq.(3) can be computed efficiently. That is, the topic-dependent unigram probability only of the last word $w_{s_t-1}$ should be computed then update by:

$$p(w_{s_t-1} | \hat{b}_h = w_{s_t}) = p_{s_t}(w_{s_t-1} | h = w_{s_t}^{m}) \cdot p(w_{s_t-2} | \hat{b}_h = w_{s_t})$$  \hspace{1cm} (5)

While the candidates of $b_h$ are restricted only to the first word of each sentence in the explanation above, any word before $w_t$ could be included as the candidates in principle. However, this restriction significantly reduces computational time with only a slight decline in accuracy.

3.2. Integration of multiple hypotheses

It is shown in [6] that integrating predictions obtained by the multiple LDA models that are independently trained considerably reduces the perplexity in case of fixed context lengths. In this paper, each of multiple LDA’s estimates the optimum context length and computes the topic-dependent unigram probability, then those hypotheses are integrated so as to improve accuracy.

First, each of $M$ LDA models estimates the optimum context length $\hat{b}_h(t;m)$ along the procedure described in the previous section ($1 \leq m \leq M$). Then based on the probability $p^{m}(w_{s_t} | \hat{b}_h = \hat{b}_h(t;m))$ obtained from Eq.(3), the likelihood of each context length is calculated by:

$$L(t,m) = \log \frac{p^{m}(w_{s_t} | \hat{b}_h = \hat{b}_h(t;m)) \prod_{m=1}^{M} p^{m}(w_{s_t} | \hat{b}_h = \hat{b}_h(t;m))}{\prod_{m=1}^{M} p^{m}(w_{s_t} | \hat{b}_h = \hat{b}_h(t;m))}$$  \hspace{1cm} (6)

Each LDA then computes the topic-dependent unigram probability $p^{m}_{s_t}(w_{t} | h = \hat{b}_h(t;m) \cdots w_{s_t})$ with the context $h$ that begins at $\hat{b}_h(t;m)$. All the probabilities are averaged using the likelihoods as weights to generate final prediction by:

$$p_{L}(w_{t} | h) = \sum_{m=1}^{M} L(t,m) p_{s_t}^{m}(w_{t} | h = \hat{b}_h(t;m) \cdots w_{s_t}) / \sum_{m=1}^{M} L(t,m)$$  \hspace{1cm} (7)

3.3. Topic adaptation of N-gram models

Aside from LDA, topic-independent N-gram models are also trained from a corpus in advance. Topic-dependent N-gram probabilities ($N \geq 2$) are derived from $p_{L}(w|h)$ in Eq.(7) using unigram-rescaling [4] below:

$$p(w_{t} | w_{s_t-1}^{m+1}, h) \propto p_{s_t}(w_{t} | h) p(w_{t} | w_{s_t-1}^{m+1})$$  \hspace{1cm} (8)

$p_{s_t}(w_{t} | w_{s_t-1}^{m+1})$: topic-independent unigram probability

$p(w_{t} | w_{s_t-1}^{m+1})$: topic-independent N-gram probability

4. Experiments

4.1. Training and evaluation text

Training text and evaluation text are shown below.

[Training text]
All articles in CD-Mainichi Newspapers 2005 [10] (95,881 articles, approximately 29 million words)

[Evaluation text]
1000 randomly selected articles consisting of at least 200 characters from CD-Mainichi Newspapers 2006 [10]

From the evaluation text, three datasets with different rates of topic transition are generated as described later.

4.2. Training conditions

The parameters of LDA models were estimated using the training text shown above. The number of latent topics is set to 100 and the size of lexicon is limited to 77,794 words with a frequency $\geq 5$. VB inference was used for training and $\alpha$ were estimated by fixed-point iteration [11]. The training steps were iterated until the perplexity for the training text decreased less than 0.1% from the previous step. Eight LDA models were trained by seeding different initial values separately (i.e., the number of LDA models $M = 8$).

Topic-independent N-gram models ($N = 2, 3$) were trained with the same training text as well. Bi-gram and trigram probabilities were estimated by Kneser-Ney smoothing [12].
4.3. Evaluation

The proposed method should be evaluated on the text of which topics change with various rates. Following the related work [7], we hence generated three evaluation sets “raw”, “slow” and “fast” with different transition rates of topics by sampling sentences from the original evaluation text using the procedure below:

(1) Separate the original 1000 articles into 16 subsets based on the page-codes*

(2) Set the first sentence of each subset as an initial position.

(3) Select one of the subsets randomly and sample $X$ contiguous sentences.

(4) Skip $Y$ sentences in the current subset.

(5) Repeat (3) and (4) until $Q$ sentences are obtained.

$x$ and $y$ are the uniform random numbers shown in Table 1, which control transition rates of topics. The number of sentences $Q$ of each evaluation set was 2000.

<table>
<thead>
<tr>
<th>name</th>
<th>parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw</td>
<td>$X=100, Y=0$</td>
</tr>
<tr>
<td>slow</td>
<td>$1 \leq X \leq 10, 1 \leq Y \leq 10$</td>
</tr>
<tr>
<td>fast</td>
<td>$1 \leq X \leq 53, 1 \leq Y \leq 10$</td>
</tr>
</tbody>
</table>

### Table 1: Parameters of evaluation sets

<table>
<thead>
<tr>
<th>method</th>
<th>unigram</th>
<th>bigram</th>
<th>trigram</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed $h$ (baseline)</td>
<td>1222.7</td>
<td>1251.8</td>
<td>1407.1</td>
</tr>
<tr>
<td>proposed method</td>
<td>1104.5</td>
<td>1162.1</td>
<td>1335.3</td>
</tr>
<tr>
<td>MSM-LDA</td>
<td>1186.4</td>
<td>1226.7</td>
<td>1388.5</td>
</tr>
<tr>
<td>boundary-given</td>
<td>1128.5</td>
<td>1184.7</td>
<td>1357.4</td>
</tr>
</tbody>
</table>

### Table 2: Perplexities for evaluation sets

5. Experimental results

Table 2 shows the experimental results performed on the three evaluation sets “raw”, “slow” and “fast”. On our proposed method, $M$ (=the number of LDA models) is set to 8, and the parameters $(l_{min}, l_{max})$ that restrict the search range of $h_b$ (see in Figure 1) are configured as (5, 100) based on a preliminary experiment (i.e., a context length is limited between 5-100 words). The method “fixed $h$”, which we treat as a baseline, also adopts the first word of a context as the beginning of a context $h$, simulating the situation where boundaries between articles are given. On both “fixed $h$” and “boundary-given”, eight LDA models are trained as the proposed method are employed. In these two methods, topic-dependent unigram probabilities are estimated along their own methodology in each model, and then those probabilities are averaged without weights.

“MSM-LDA” [7] is one of the existing methods that also optimizes context lengths. This method integrates predictions with various context lengths, each of which length is sampled by a Monte Carlo particle [13]. In this experiment, this method uses one of the LDA models which has typical accuracy among those in the proposed method (i.e., training text and the number of latent topics are identical to those of the proposed method), and the number of particles is set to 50.\(^*1\) In order to compare fairly with the proposed method, the context length is restricted not more than 100 words. Topic-dependent bigram and trigram probabilities in the all methods are computed by unigram-rescaling given by Eq.(8).

As shown in Table 2, the proposed method reduces perplexity for all the cases. The reduction in perplexity from the baseline is 5-10% for unigrams, 4-8% for bigrams, 3-5% for trigrams, respectively. Comparing with “boundary-given”, accuracy of the proposed method is slightly better (for unigrams) or almost same (for bigrams and trigrams). This result demonstrates that our method can predict a word as better as when boundaries between articles are given even though the boundaries are unknown in practice. In contrast, MSM-LDA only yields slight decrease for unigrams, and shows little or no improvement for bigrams and trigrams.

Illustrated in Figure 2 is a comparison between the two methods on their topic-dependent bigram probabilities for the evaluation set “slow”. The vertical axis represents the ratio of the topic-dependent bigram probability of the proposed method against MSM-LDA, which thus means if the ratio > 1 the proposed method performs better than MSM-LDA. The horizontal axis represents the topic-dependent unigram probability when a context length is fixed at 20 words. It is therefore shown that the proposed method more accurately predicts a word with lower frequency, or a word which is difficult to be predicted precisely with a fixed context length.

Computational time of the two methods relative to the case of “fixed $h$” was 1.133 in the proposed method and 3.378 in MSM-LDA, respectively, indicating the proposed method yields higher performance with less computational costs.

6. Discussion

As described in the previous section, the proposed method significantly reduces perplexity comparing to both MSM-LDA and “fixed $h$” in most cases. Since our method improves perplexity for bigrams and trigrams whereas MSM-LDA does little, we consider that the proposed method can alleviate the difficulty due to the sparseness problem.

The proposed method and MSM-LDA are similar in the way that they both integrate a number of hypotheses with various context lengths, while their methods for estimating a context length are completely different. Here, the number of hypotheses corresponds to the number of models in the proposed method, and the number of particles in MSM-LDA, respectively. The experimental results show the proposed method efficiently improves performance with much fewer hypotheses to integrate.

Shown in Figure 3 are the context lengths estimated by the two methods during the first 1000 words of the set “slow”. The vertical axes represent an average length of contexts in two methods during the first 1000 words of the set “slow”. The vertical axis represents the ratio of methods on their topic-dependent bigram probabilities for the evaluation set “slow”. The method “boundary-given” always adopts the first word of a current article as the beginning of a context, simulating the situation where boundaries between articles are given.

\(^*1\) Each article has an attribute indicating which page it appeared, such as international, sports, culture, etc.

\(^*2\) Accuracy of this method was found to be saturated around 50 particles through preliminary experiments.
the horizontal axes. The context length of the proposed method tends to have a locally minimum value right after a boundary between articles and then get longer. On the other hand, it seems that MSM-LDA tends to detect a shorter context even when it might not be necessarily appropriate, while quite a longer context is frequently adopted as well. In figure 3(b), for instance, the length plunges around \( t = 340, 550, 940 \) in spite of during an article and it affects subsequent estimation for a certain while. This behavior suggests that MSM-LDA may not be very robust to an appearance of an “unexpected” word. The proposed method, by contrast, rarely detects topic transition during an article, or even if such a case could occur it has little effect on subsequent estimation, so that the length will immediately recover, as illustrated in Figure 3(a).

As already shown in Figure 2, the difference between two methods is observed for a word with lower frequency, or a word which is difficult to be predicted with a fixed context length. Such a word is likely to be a topic word (i.e., a word which is related to a specific topic). Therefore presumably our method improves accuracy to predict a word which comes from a specific topic, due to stably adapting to topic transition.

7. Conclusion

We have proposed a method that improves accuracy of N-gram models which can be applied to on-line applications including continuous speech recognition and predictive text entry. In our method, each of multiple LDA models that are independently trained estimates an optimum context length separately, then those predictions are integrated and N-gram probabilities are calculated. The method directly estimates the optimum context length suitable for word prediction, by evaluating various lengths based on accuracy to predict the known part of a current sentence.

Experimental results performed on the datasets with different rates of topic transition have shown that the proposed method reduces perplexity of N-grams \((N \leq 3)\) from a baseline for all the cases. Having found to improve perplexity for bigrams and trigrams, our method is considered to alleviate the difficulty due to the sparseness problem. A comparison with one of the existing method has indicated that our method can adapt to topic transition more stably, thereby improving accuracy to predict a word particularly with lower frequency.

We consider such a word can be related to a specific topic. It has also been found that our method yields higher performance with less computational costs comparing to the existing method.

Future work involves further analysis on the experimental results shown here, in particular, exploring words and context where the proposed method successfully performs. We are also going to apply the method to real on-line applications such as predictive text entry and speech recognition, as well as topic segmentation of dynamic text and/or video streams.

8. References


