Introduction to Hierarchical Pitman-Yor Processes

Daichi Mochihashi

ATR SLC

`daichi.mochihashi@atr.jp`

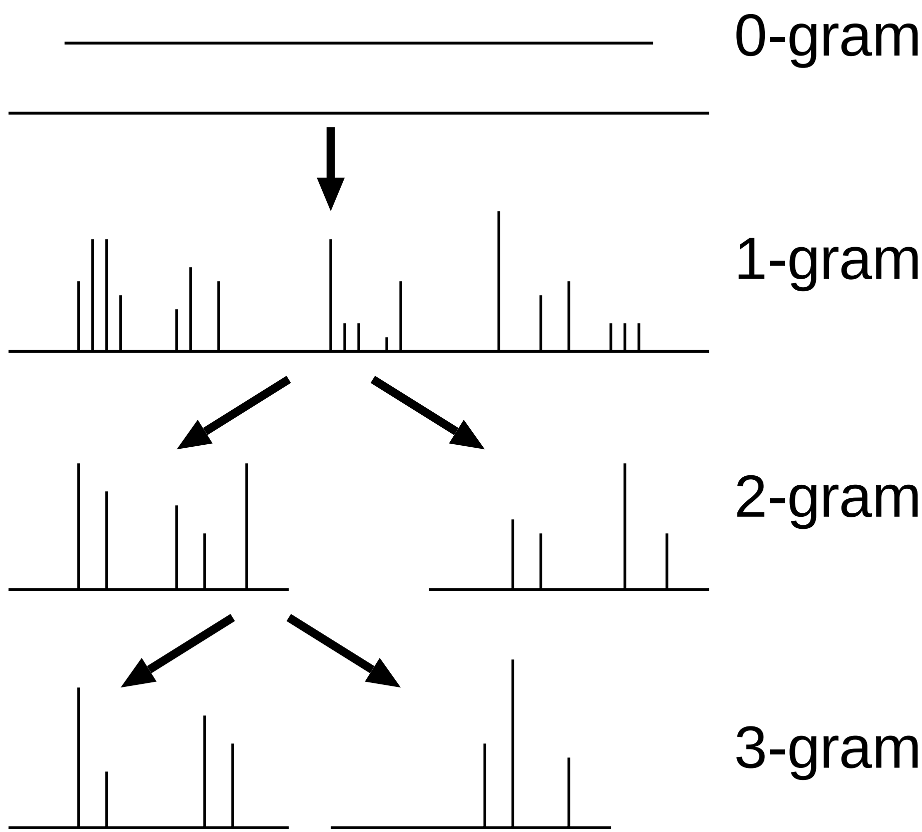
“Ultraconservative” SVM 2006

Aug 5, 2006

NAIST

Overview

- Pitman-Yor processes
  $=$ Two-parameter Poisson-Dirichlet process $\mathrm{PD}(d, \theta)$
  (Pitman and Yor 1997)
  - ディリクレ過程 (有限次元の場合はディリクレ分布) の拡張
- n-gram 分布の階層的な生成モデル



Prior: $(n-1)$ gram

$$\Big\downarrow \ \mathrm{PD}(d_n, \theta_n)$$

Posterior: (n) gram

Background

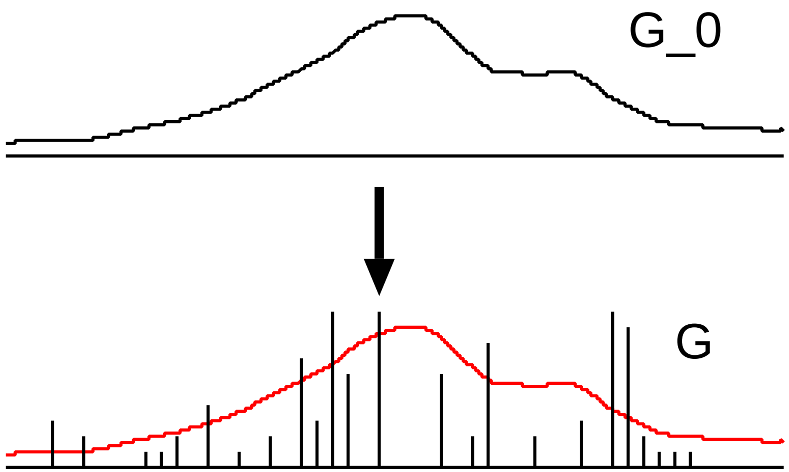
- 現状のトピックモデル (LDA,DM など) は unigram のみ
  - Trigram とのアドホックな混合が必要になる.
    - 計算量も大きい
  - 「トピックに特有な bigram, trigram, ...」が存在
    - “mixture of” $\rightarrow$ { Gaussians, cultures, flour }
    - “everyone” $\rightarrow$ { shall (法律), ’s (口語) }
    - 適切にできれば, NE recognition の代わりになる
- n-gram の, きちんとした確率モデルが必要
  - 他のモデルに部品として組み込める
    - 例. トピックモデル / 統計翻訳
  - これまでの n-gram は, カウントの smoothing ($\neq$ 確率モデル)
  - 無限語彙を自然に扱えるモデルが必要
    - 自然言語の語彙は本当は可算無限

Pitman-Yor and DP

- HPY (Hierarchical Pitman-Yor process, Teh 2006)
  $\cdots$ HDP (Hierarchical Dirichlet process, Teh & Jordan 2004) の拡張
- Pitman-Yor process $\cdots$ Dirichlet Process (DP) の拡張
- 以下, この順番で:
  1. Dirichlet process とは?
  2. Pitman-Yor process とは?
  3. 階層 Pitman-Yor process とは?
  4. Gibbs Sampling によるパラメータ推定

Dirichlet process (DP) (Ferguson 1973) とは?

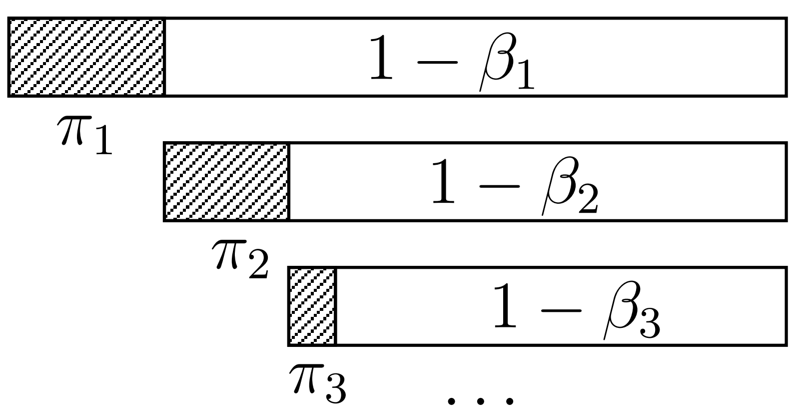
- 自然言語処理の文脈では (=1 次元の場合),
  <u>無限次元の離散分布を生成する確率モデル</u>
  のこと.
- $G \sim \mathrm{DP}(\alpha, G_0)$ とすると, 基になる連続分布 (測度) G_0 に対して



G はそれと少し異なる, 無限次元の離散分布.
  - concentration パラメータ α によって, G の G_0 との違いを調整 ($\alpha\!=\!0$ とすると完全に一致)
  - 無数の G がサンプルできるが, その期待値は $E[G] = G_0$.

Stick-breaking process

- どうやって具体的に $G \sim \mathrm{DP}(\alpha, G_0)$ をサンプルすればいい?
  $\rightarrow$ Stick-breaking process (Sethuraman 1994)
- 長さ 1 (確率の総和) の棒を, 左から切っていく
  1. まず, $\beta_1 \sim \mathrm{Be}(1, \alpha)$ の点で棒を分割
     - 左の切れ端 π_1 : 長さ β_1
  2. 残った方を, $\beta_2 \sim \mathrm{Be}(1, \alpha)$ でまた分割
     - 左の切れ端 π_2 : 長さ $\beta_2(1-\beta_1)$
  3. 残った方を, $\beta_3 \sim \mathrm{Be}(1, \alpha)$ でまた分割
     - 左の切れ端 π_3 : 長さ $\beta_3(1-\beta_2)(1-\beta_1)$
  4. $\cdots$



Stick-breaking process (2)

- 一般に, $G \sim \mathrm{DP}(\alpha, G_0)$ は

$$\pi_n = \beta_n \prod_{i=1}^{n-1}(1-\beta_i) \quad (n = 1, \ldots, \infty) \tag{1}$$

$$\beta_i \sim \mathrm{Be}(1, \alpha) \tag{2}$$

のように棒を分割して,

$$G = \sum_{k=1}^{\infty} \pi_k \delta(\theta_k) \tag{3}$$

$$\theta_k \sim G_0 \tag{4}$$

のような δ 関数の無限和で表せる.

  - 自然言語のような離散分布の場合は, G_0 は自然数上の (一様) 分布.

Polya urn scheme of DP

- $G \sim \mathrm{DP}(\alpha, G_0)$ からのサンプル $\mathbf{x} = x_1, x_2, \ldots, x_n$ が得られている時, 次の x は何になるか?

$$p(x|\mathbf{x}, \mathrm{DP}(\alpha, G_0)) = \int p(x|G) p(G|\mathbf{x}, \mathrm{DP}(\alpha, G_0)) dG \quad (5)$$

$$= \sum_{k=1}^{K} \frac{n_k}{n+\alpha} \delta(\theta_k) + \frac{\alpha}{n+\alpha} G_0. \quad (6)$$

  - θ_k ($k = 1, \ldots, K$) : $x_1, \ldots, x_n$ の中の異なり成分
  - n_k ($k = 1, \ldots, K$) : その頻度
- ベイズ的なスムージングが得られる
  - 頻度 0 のカテゴリの確率は, $\dfrac{\alpha}{n+\alpha} G_0$.

Polya urn scheme of DP (2)

- これは, 次の MacKay の Dirichlet smoothing とも本質的に同じ (SVM 2004)

$$p(w_i|w_j) = \frac{n_j}{n_j+\alpha}\hat{p}(i|j) + \frac{\alpha}{n_j+\alpha}\bar{\alpha}_i \tag{7}$$

$$= \frac{n_{i|j}+\alpha_i}{\sum_i (n_{i|j}+\alpha_i)} \tag{8}$$

- DM (山本 2003, Sjölander et al. 1996) とも同じ

$$p(v|\mathbf{h}) = \sum_m C_m \frac{n(v|\mathbf{h})+\alpha_{mv}}{\sum_v (n(v|\mathbf{h})+\alpha_{mv})} \tag{9}$$

$$C_m \propto \lambda_m \frac{\Gamma(\alpha_m)}{\Gamma(\alpha_m+h)} \prod_v \frac{\Gamma(n(v|\mathbf{h})+\alpha_{mv})}{\Gamma(\alpha_{mv})} \tag{10}$$

Polya urn scheme of DP (3)

- 何がいけないか?
  - 頻度 0 のとき,

$$p(w_i|w_j) = \frac{\alpha_i}{\sum_i (n_{i|j} + \alpha_i)} \tag{11}$$

  - 頻度 1 のとき,

$$p(w_i|w_j) = \frac{1 + \alpha_i}{\sum_i (n_{i|j} + \alpha_i)} \tag{12}$$

    - 通常, $\alpha \ll 1$ ($\alpha_i = 0.001$ くらい)
    - $3 + \alpha_i, 2 + \alpha_i, 1 + \alpha_i \Longleftrightarrow \alpha_i \quad \cdots$ ものすごい差.
- 頻度 $(1, 2, 3, \ldots)$ 自体をそのまま信用せず, ダンピングする必要があるのでは?

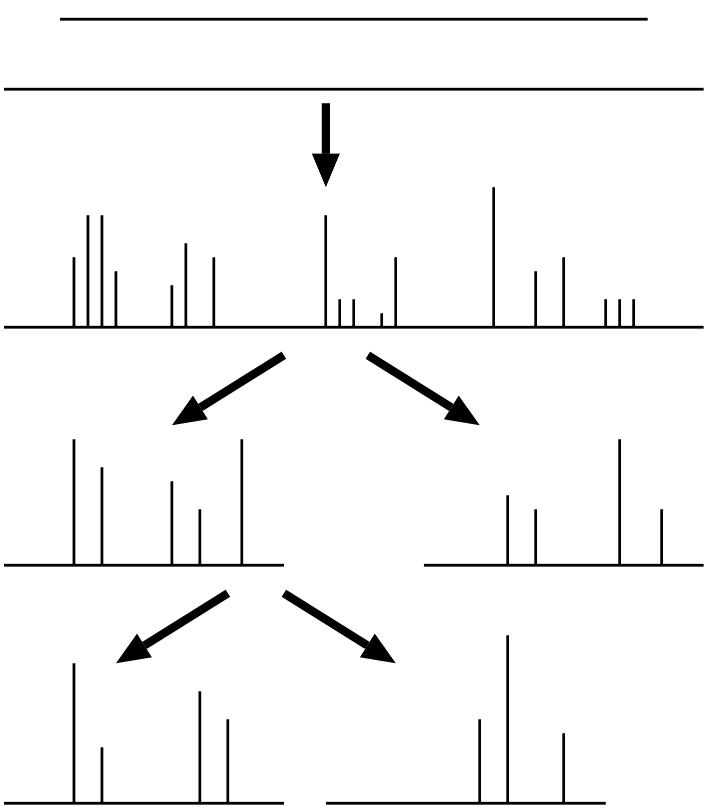
  $\rightarrow$
  - Kneser-Ney smoothing (Kneser and Ney 1995)
  - Pitman-Yor processes (Teh 2006, Pitman and Yor 1997).

Pitman-Yor process $\mathrm{PD}(d, \theta)$

- Stick-breaking process:
  - For $k = 1, 2, \ldots, \infty$,
$$\beta_k \sim \mathrm{Be}(1 - d, \theta + kd). \tag{13}$$
- k が増えるにつれ, 分割の平均位置が左に移動
  - より多くのコンポーネント (単語) に確率が割り振られる (heavy tail)
- Polya urn representation
$$p(x|\mathbf{x}, \mathrm{PD}(d, \theta)) = \sum_{k=1}^{t} \frac{n_k - d}{n + \theta} \delta(\theta_k) + \frac{\theta + dt}{n + \theta} G_0 \tag{14}$$
  - 前半で, カウントが常に d だけディスカウントされる
  - 後半のスムージングは DP と同じ

Hierarchical Pitman-Yor Processes

- Base measure G_0 が, また階層的に $\mathrm{PD}(d', \theta')$ に従う
  - 3-gram : $G_3 \sim \mathrm{PD}(d_3, \theta_3 | G_2)$
  - 2-gram : $G_2 \sim \mathrm{PD}(d_2, \theta_2 | G_1)$
  - 1-gram : $G_1 \sim \mathrm{PD}(d_1, \theta_1 | G_0)$
  - 0-gram : $G_0 \sim$ uniform. $\quad (p(v) = 1/V)$



0-gram

1-gram

2-gram

3-gram

Prior: $(n-1)$ gram

$$\Big\downarrow \mathrm{PD}(d_n, \theta_n)$$

Posterior: (n) gram

Polya Urn scheme of HPY

$$p(x|\mathbf{x},\mathrm{HPY}(d,\theta)) = \sum_{k=1}^{t} \frac{n_k - d}{c+\theta}\,\delta(\theta_k) + \frac{\theta + dt}{c+\theta}\,G_{n-1} \tag{15}$$

- n-gram 分布を, $(n-1)$-gram 分布から生成する
- G_{n-1} は離散 $\cdots$ 新しい θ_{k+1} は他の θ と同じになる可能性
  - 単語 w が G_{n-1} から引かれた異なり数を $t(w)$ とすると, 上式は

$$p(x\!=\!w|\mathbf{x},\mathrm{HPY}(d,\theta)) = \frac{n(w) - d\cdot t(w)}{c+\theta} + \frac{\theta + dt}{c+\theta}\,G_{n-1}(w) \tag{16}$$

- これは, 言語モデルで最高性能といわれる Kneser-Ney スムージング (Kneser and Ney 1995)

$$p(w|h) = \frac{n(w|h) - d(n(w|h))}{n(h)} + \gamma(h)p(w) \tag{17}$$

で, $t(w)$ を常に 1 とした場合に相当する.

Chinese Restaurant Process of HPY

- 客が, 無限個のテーブルがあるレストランに入って食事をする
  - 同じテーブルに座った人は, 同じ料理を食べる
- どのテーブルに座るか?
  - 各テーブル k の客数 $c_k - d$ に比例した確率で選ぶ [人気順]
  - テーブルの総数 t に対して, $\theta + dt$ に比例した確率で新しいテーブルを選ぶ
  - 新しいテーブルでは新しい料理を頼む
- レストラン $=$ n-gram 文脈, 料理 $=$ 単語, 客 $=$ 頻度
  - ある n-gram 文脈では, Pitman-Yor process に従って料理 (単語) の頻度が分布する
  - 新しいテーブルでの料理はどうやって選ぶ?

Chinese Restaurant Process of HPY (2)

- 新しいテーブルの料理は何にする?
  - 親レストランに代理の客を行かせて, そこで同様に座って出た料理と同じにする
  - 親レストランでも新しいテーブルになったら, さらに親レストランに代理の客を行かせる
- 親レストラン =(n-1)-gram 文脈
  - ある文脈で, 同じ単語が複数のテーブルでサーブされている可能性がある
    - 親レストランに行ったとき, 同じ料理を出される可能性が高い [人気順]
    - $t(w)$ に相当する
- これ以上親レストランがない =0-gram 文脈
  - 仕方がないので, 厨房が新しく考える (Base measure)
  - 新しく作った単語の確率は, 文字 HMM などで計算

Inference of seating arrangements

$$p(\mathbf{s}) = \prod_w G_0(w)^{c_{0w\cdot}} \prod_u \frac{(\theta_{|u|})_{d_{|u|}}^{(t_{u\cdot})}}{(\theta_{|u|})^{(c_{u\cdot\cdot})}} \prod_w \prod_{k=1}^{t_{u\cdot}} (1 - d_{|u|})^{(c_{uwk}-1)} \tag{18}$$

- c_{uwk} : Frequency of word w at table k in the n-gram context u
- t_{uw} : Number of tables with word w in the n-gram context u
- Gibbs sampling:

  For $t = 1 \ldots T$,

  For $w \in randperm(\{$all counts in the model$\})$,

  1. Remove a customer w from the restaurant.
  2. Add w to the restaurant at a random table following the Pitman-Yor process.
     - When a new table is selected, choose a dish by sending a proxy customer to the parent restaurant.

Inference of hyperparameters

- Through Data augmentation (adding auxiliary variables x_u, y_{ui}, and z_{uwkj}),

$$p(\mathbf{s}) = \prod_w G_0(w)^{c_{0w\cdot}} \cdot \prod_u \frac{\int_0^1 x_u^{\theta_{|u|}} (1-x_u)^{c_{u\cdot\cdot}-2} dx}{\Gamma(c_{u\cdot\cdot}-1)} \prod_{i=1}^{t_{u\cdot}-1} \sum_{y_{ui}} \theta_{|u|}^{y_{ui}} (d_{|u|} i)^{1-y_{ui}}$$
$$\prod_w \prod_{k=1}^{t_{u\cdot}} \prod_{j=1}^{c_{uwk-1}} \prod_{z_{uwkj}} (j-1)^{z_{uwkj}} (1-d_{|u|})^{1-z_{uwkj}} \tag{19}$$

- Therefore, posterior for hyperparameters are

$$d_n \sim \mathrm{Be}\Big(1 + \sum_{|u|=n} \sum_{i=1}^{t_{u\cdot}-1} (1-y_{ui}),\ 1 + \sum_{|u|=n} \sum_w \sum_{k=1}^{t_{u\cdot}} \sum_{j=1}^{c_{uwk}-1} (1-z_{uwkj})\Big), \tag{20}$$
$$\theta_n \sim \mathrm{Ga}\Big(\alpha_n + \sum_{|u|=n} \sum_{i=1}^{t_{u\cdot}-1} y_{ui},\ \beta_n + \sum_{|u|=n} \log x_u\Big). \tag{21}$$

Implementation

```
~/work/ngram/src :pxs052
pxs:~/work/ngram/src% ./hpylm -h                                    [ 9:20AM]
usage: hpylm -n [order] -t [iteration] [-i interval] [-d] train.dat [test.dat]
$Id: hpylm.c,v 1.7 2006/07/14 13:49:48 dmochiha Exp $
pxs:~/work/ngram/src%                                               [ 9:20AM]
```

```
struct ngram {          /* c_uxx */
        ngram *parent;
        ngram **children;
        restaurant **words;
        int nchildren;
        int nwords;
        int ncounts;
        int ntables;
        int level;
        int id;
};
struct restaurant {     /* c_uwx */
        int id;
        int ntables;
        int maxtables;
        int ncounts;
        int *tables;
        ngram *context;
};
```

Small Experiments

- "Austen" language modeling in FSNLP
  - `http://nlp.stanford.edu/fsnlp/`
  - 6. "Statistical inference: n-gram models over sparse data" p.191–

- Modified Kneser-Ney smoothing, SRI LM Toolkit

```
% ngram -order 3 -lm austen3.arpa -ppl austen.test.dat
0 zeroprobs, logprob= -186076 ppl= 135.906 ppl1= 167.651
```

- HPY Language model, Gibbs

```
% hpylm -h
hpylm, Hierarchical Pitman-Yor language model.
$Id: hpylm.c,v 1.17 2007/03/19 09:11:05 dmochiha Exp $
usage: hpylm -n [order] -N [iteration] train model
% hpylm -n 3 -N 50 austen.dat austen.n3
Gibbs 50 : iteration 725242 / 725242.. ETA: 0:00:00 (6 sec/sweep)
done.
% hpyppl -N 50 austen.n3 austen.test.dat
loading language model from austen.n3 ..
Gibbs 50 : iteration 725242 / 725242.. PPL = 130.936
perplexity = 130.936
```

Seatings integrated out

- c_{uwk} 人の客 (単語) が t_{uw} 個のテーブルに座る場合の数の総和を考えると,

$$p(\mathbf{s}) = \prod_w G_0(w)^{c_{0w\cdot}} \cdot \prod_u \frac{(\theta_{|u|})_{d_{|u|}}^{(t_{u\cdot})}}{(\theta_{|u|})^{(c_{u\cdot\cdot})}} \prod_w s_{d_{|u|}}(c_{uw\cdot}, t_{uw}) \tag{22}$$

  - テーブルごとの c_{uwk} が必要なく, その総和 c_{uw} だけですむ
  - $s_d(c,t)$ は type$(-1,-d,0)$ の一般化 Stirling 数で
    1. $s_d(0,0) = s_d(1,1) = 1$
    2. $s_d(c,0) = s_d(0,t) = 0$
    3. $s_d(c,t) = s_d(c-1,t-1) + (c-1-dt)s_d(c-1,t)$ $(0 < t \le c)$

    をみたす.

- Polya 分布との類似に注意

$$p(\mathbf{x}|\boldsymbol{\alpha}) = \int p(\mathbf{x}|\mathbf{p})p(\mathbf{p}|\boldsymbol{\alpha})d\mathbf{p} = \frac{\Gamma(\sum_k \alpha_k)}{\Gamma(\sum_k \alpha_k + n_k)} \prod_k \frac{\Gamma(n_k + \alpha_k)}{\Gamma(\alpha_k)} \tag{23}$$

Prediction and Inference with seatings integrated out

- モデル全体がハイパーパラメータおよびカウント c_{uw}, t_{uw} の推定値で記述できる時,

$$p(w|u) = \sum_{c_{uw}} \sum_{t_{uw}} p(w, c_{uw}, t_{uw}|u) \tag{24}$$

$$= \sum_{c_{uw}} \sum_{t_{uw}} p(w|u, c_{uw}, t_{uw}) p(t_{uw}|u, c_{uw}) p(c_{uw}|u) \tag{25}$$

$$= \sum_{c_{uw}} p(c_{uw}|u) \sum_{t_{uw}} p(w|u, c_{uw}, t_{uw}) p(t_{uw}|u, c_{uw}). \tag{26}$$

- $p(c_{uw}|u)$ 及び $p(t_{uw}|u, c_{uw})$ を推定 $\rightarrow$ LBP で行けるらしい
  - 実際には, $p(c_{uw}, t_{uw}|u)$ の同時分布が必要 (行列)
  - カウント c_{uw} が大きいとき, パラメータ量膨大
    - Gibbs では, (c_{uwk}, t_{uw}) の確率の高い組だけを確定的にサンプリングして取り出している.

Future Works

- Topic modeling with HPY n-gram.
  - How to combine HPY n-gram with EM? (HMM)
  - 計算量の多い $s_d(c,t)$ の近似
- Mittag-Leffler distribution, Gamma process, Lévy measure, .. など, 知らないことが沢山
- Dependent Dirichlet process (DDP) (Griffin & Steel 2004)
  - レストランで「客がしばらくすると席を立つ」ことを許す.