

13 章 統計的言語モデルを作ろう

- 統計的言語モデルとは
 - $P(\text{単語列})$ を言語統計から計算
 - 正しい文には高い確率を与えたい
 - 誤っている文には低い確率を与えたい
 - 音響的に似ている単語との置換誤りを除外する
 - 例) × 駅の規格のホテルを探して
 △ 駅の死角のホテルを探して
 ○ 駅の近くのホテルを探して

13.1 文の出現確率の求め方

- 言語モデルの作り方

1. コーパスを準備する

- 大量の電子化された文章を集める

例) 新聞記事データ, web, Wikipedia, etc.

2. 単語に区切る

- 形態素解析処理

3. 単語列の出現確率を求める

- スパースネスの問題を解決することが必要

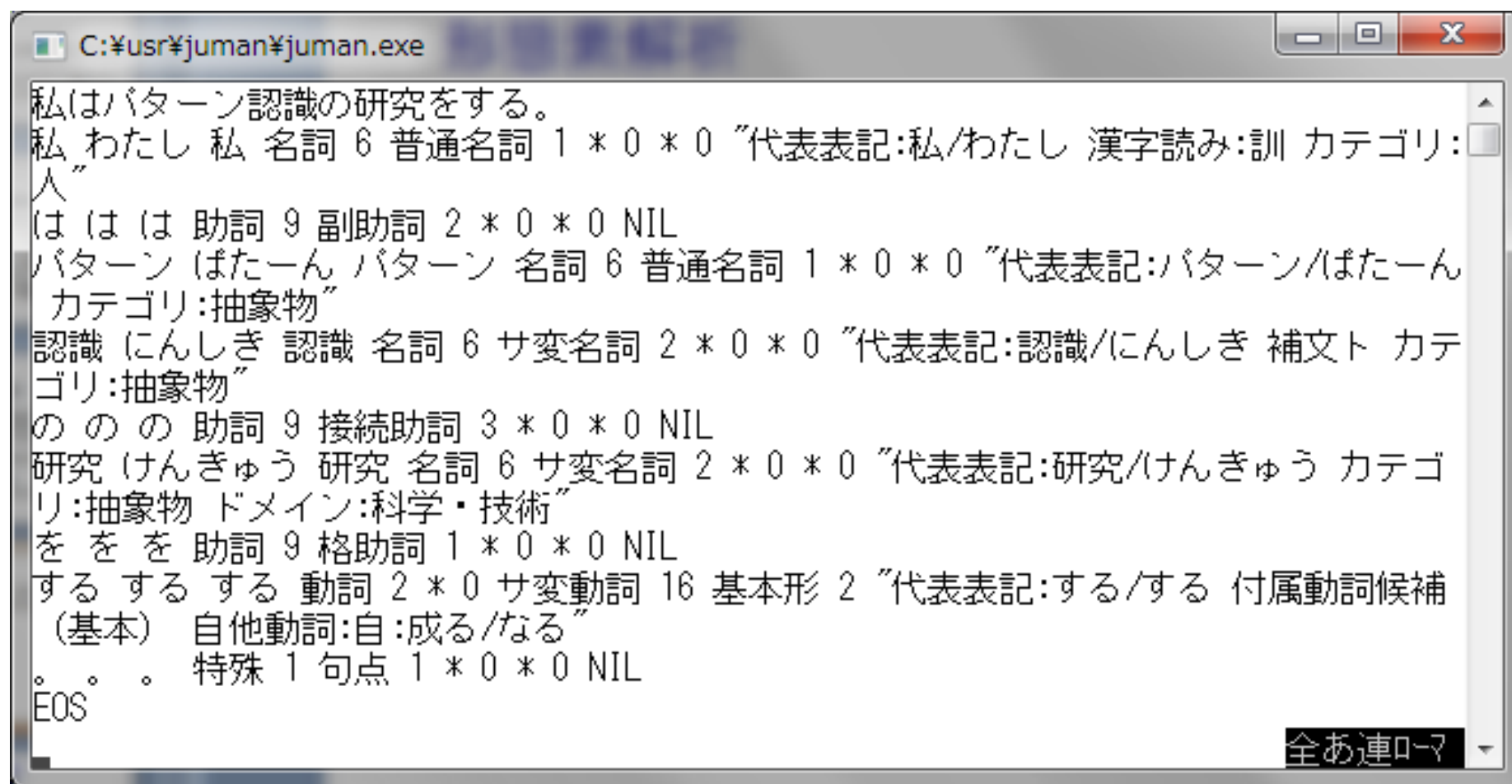
13.1 文の出現確率の求め方

- コーパスとは
 - 自然言語文の文例集
- コーパスの例
 - 新聞記事
 - 毎日新聞 各年度約 10 万記事
 - Wikipedia（日本語, 2019 年 12 月現在）
 - 約 118 万記事
 - 青空文庫（日本語, 2019 年 12 月現在）
 - 著作権のないものを中心に 約 16,000 作品

13.1 文の出現確率の求め方

- 形態素解析ソフト Juman

- <http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman.html>



```
C:\usr¥juman¥juman.exe
私はパターン認識の研究をする。
私 わたし 私 名詞 6 普通名詞 1 * 0 * 0 "代表表記:私/わたし 漢字読み:訓 カテゴリ:人"
は は は 助詞 9 副助詞 2 * 0 * 0 NIL
パターン ばたーん パターン 名詞 6 普通名詞 1 * 0 * 0 "代表表記:パターン/ばたーん カテゴリ:抽象物"
認識 にんしぎ 認識 名詞 6 サ変名詞 2 * 0 * 0 "代表表記:認識/にんしぎ 補文ト カテゴリ:抽象物"
の の の 助詞 9 接続助詞 3 * 0 * 0 NIL
研究 けんきゅう 研究 名詞 6 サ変名詞 2 * 0 * 0 "代表表記:研究/けんきゅう カテゴリ:抽象物 ドメイン:科学・技術"
を を を 助詞 9 格助詞 1 * 0 * 0 NIL
する する する 動詞 2 * 0 サ変動詞 16 基本形 2 "代表表記:する/する 付属動詞候補 (基本) 自他動詞:自:成る/なる"
。 。 。 特殊 1 句点 1 * 0 * 0 NIL
EOS
全あ連らゝ
```

13.1 文の出現確率の求め方

- 単語列 w の生成確率

$$\begin{aligned} P(w) &= P(w_1, \dots, w_n) \\ &= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1, \dots, w_{n-1}) \end{aligned}$$

- $P(w_i)$: 単語の出現確率
 - 単語の頻度なので容易に推定可能
- $P(w_i|w_{i-1})$: 単語の接続確率
 - ある程度の規模のコーパスがあれば推定可能
- 条件部が長い条件付き確率
 - 推定は困難

13.2 N- グラム言語モデル

13.2.1 N- グラムによる近似

- N- グラム言語モデルとは
 - 単語の生起を (N-1) 重マルコフ過程で近似したモデル
 - ある時点での単語の生起確率は直前の (N-1) 単語にのみ依存すると仮定
 - $P(w_i | w_1, \dots, w_{i-1})$ の近似
 - 1- グラム : $P(w_i)$
 - 2- グラム : $P(w_i | w_{i-1})$
 - 3- グラム : $P(w_i | w_{i-2}, w_{i-1})$

13.2.1 N- グラムによる近似

- 単語列 w の生成確率
 - 3- グラムによる近似

$$P(w_1, \dots, w_n) = P(w_1)P(w_2|w_1) \prod_{k=3}^n P(w_k|w_{k-2}, w_{k-1})$$

- 確率の最尤推定

$$P(w_i) = \frac{C(w_i)}{\sum_{w_i} C(w_i)}$$

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i)}{C(w_{i-1})}$$

$$P(w_i|w_{i-2}, w_{i-1}) = \frac{C(w_{i-2}, w_{i-1}, w_i)}{C(w_{i-2}, w_{i-1})}$$

13.2.2 言語モデルの評価

- よい言語モデルとは
 - タスク内の文に対しては高い確率、そうでない文に対しては低い確率を出力するもの
- 評価式（パープレキシティ）

$$PP = P(w_1, \dots, w_n)^{\frac{1}{n}}$$

- ある単語の後に出現し得る単語数の平均
- 学習データとは別の評価データで評価する

13.2.3 ゼロ頻度問題

- ゼロ頻度問題とは
 - 妥当な単語列であっても偶然コーパスに出現しなければ、最尤推定値が 0 になる
→ その単語列を含む文の出現確率も 0 になる
- 対処法
 - 観測された情報を使って、観測されていない情報の確率を推定する
- アプローチ
 - 頻度のスムージングによる方法（13.3 節）
 - 補間法による方法（13.4 節）

13.3 一度も出現しないものの確率は？

- 頻度のスムージングの問題設定
 - 学習コーパス中に 1 回も出現しない N- グラムは, 未知コーパスで平均何回出現すると期待されるか
- 加算法
 - すべての N- グラムに一定値を加算
- 削除推定法
 - 出現回数ごとの予測出現数を評価用データを用いて推定
- Good-Turing 法
 - 出現 0 回の確率の和が、出現 1 回の確率の和と等しくなるように全体を調整

13.3.1 一定値を加えることによるスムージング

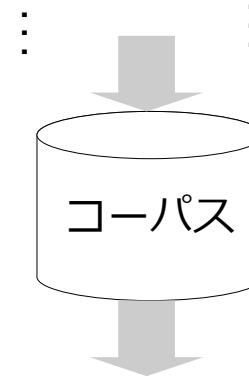
- 加算法

- すべての N- グラムの頻度計算の際に, あらかじめ一定の値 α を加えておく

$$P(w_i | w_{i-2}, w_{i-1}) = \frac{C(w_{i-2}, w_{i-1}, w_i) + \alpha}{C(w_{i-2}, w_{i-1}) + \alpha \cdot v}$$

v : 語彙数

単語列	出現回数	
明日 の 雨	1	
明日 天気 は	1	初期値 1 から開始
雨 の 明日	1	
天気 雨 晴れ	1	($\alpha=1$)

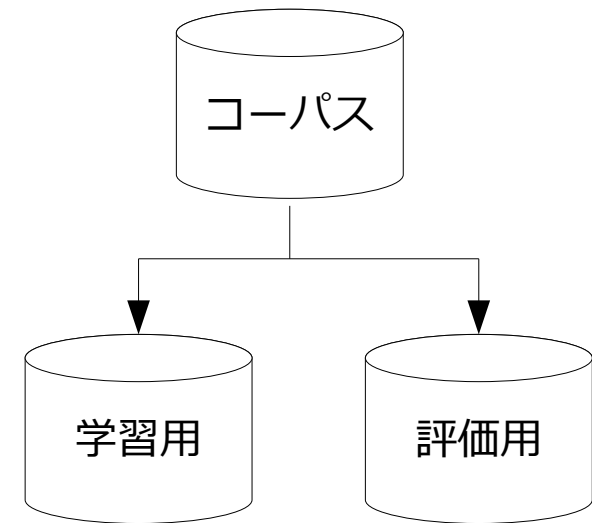


単語列	出現回数
明日 の 雨	3
明日 天気 は	5
雨 の 明日	1
天気 雨 晴れ	1

⋮ ⋮

13.3.2 削除推定法

- 削除推定法の考え方
 - 出現回数の少ないものが、未知データで平均的に何回出現することが期待できるか
 - 交差確認法の手順を使うことも可能



出現回数 k	種類数 R_k (学習用)	出現数 T_k (評価用)	推定 $r_k = \frac{T_k}{R_k}$
0	7,500	25	0.0033
1	1,500	900	0.6
2	300	450	1.5
3	100	270	2.7

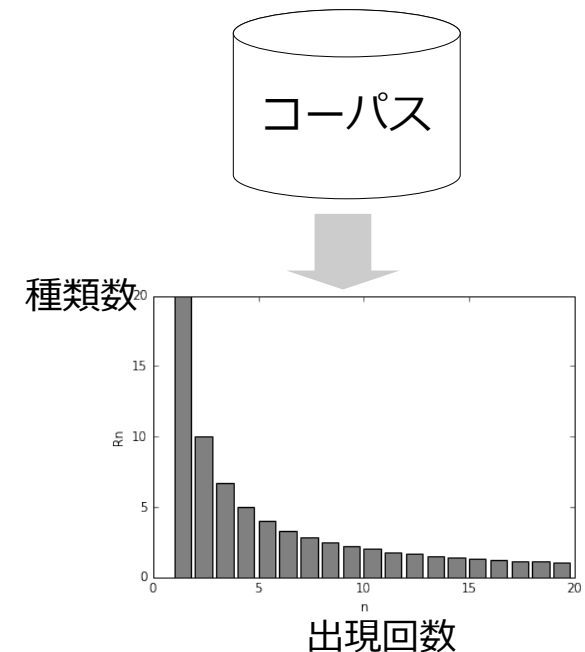
⋮

13.3.3 Good-Turing 法

- グッド・チューリングの推定
 - 出現回数 0 回の事象の確率と出現回数 1 回の事象の確率とを等しくする
 - 出現回数の推定法

$$r_n = (n + 1) \frac{R_{n+1}}{R_n}$$

- ただし R_n は、 n 回出現する 3-
グラムの種類数



$$r_0 = 1 \frac{R_1}{R_0}$$

$$r_1 = 2 \frac{R_2}{R_1}$$

$$r_2 = 3 \frac{R_3}{R_2}$$

⋮

確率の和が
1 回出現の
最尤推定値
と同じ

総和が 1 になる
ように調整

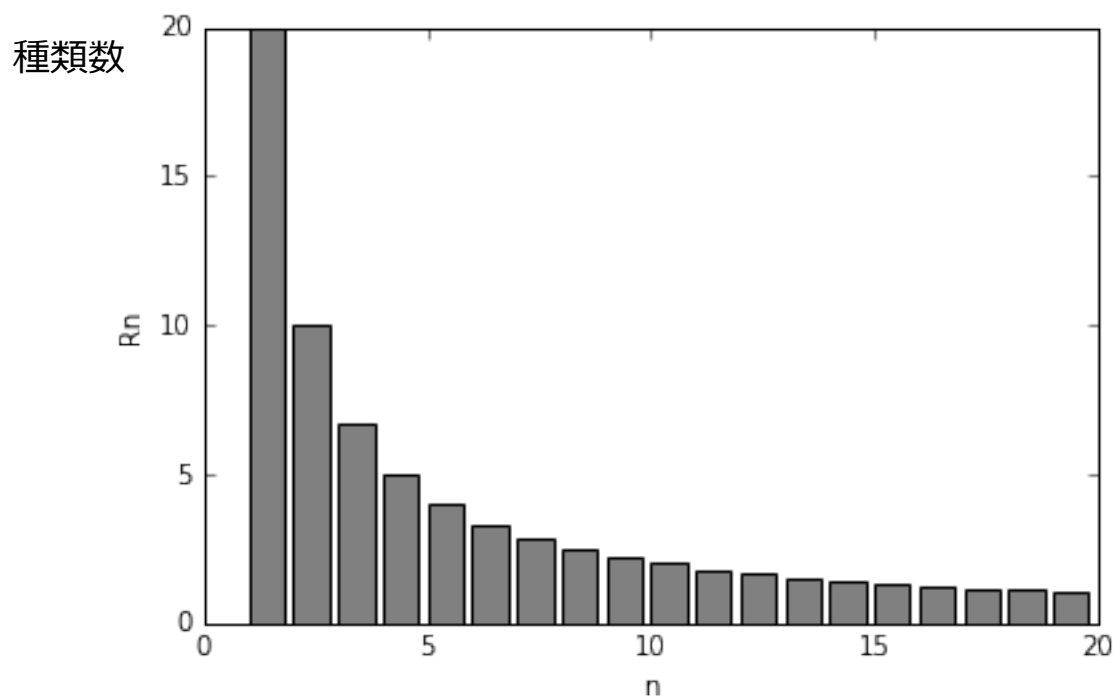
13.3.3 Good-Turing 法

- r_n の性質

$$r_0 = 1 \frac{R_1}{R_0}, \quad r_1 = 2 \frac{R_2}{R_1}, \quad r_2 = 3 \frac{R_3}{R_2}, \quad r_3 = 4 \frac{R_4}{R_3}, \dots$$

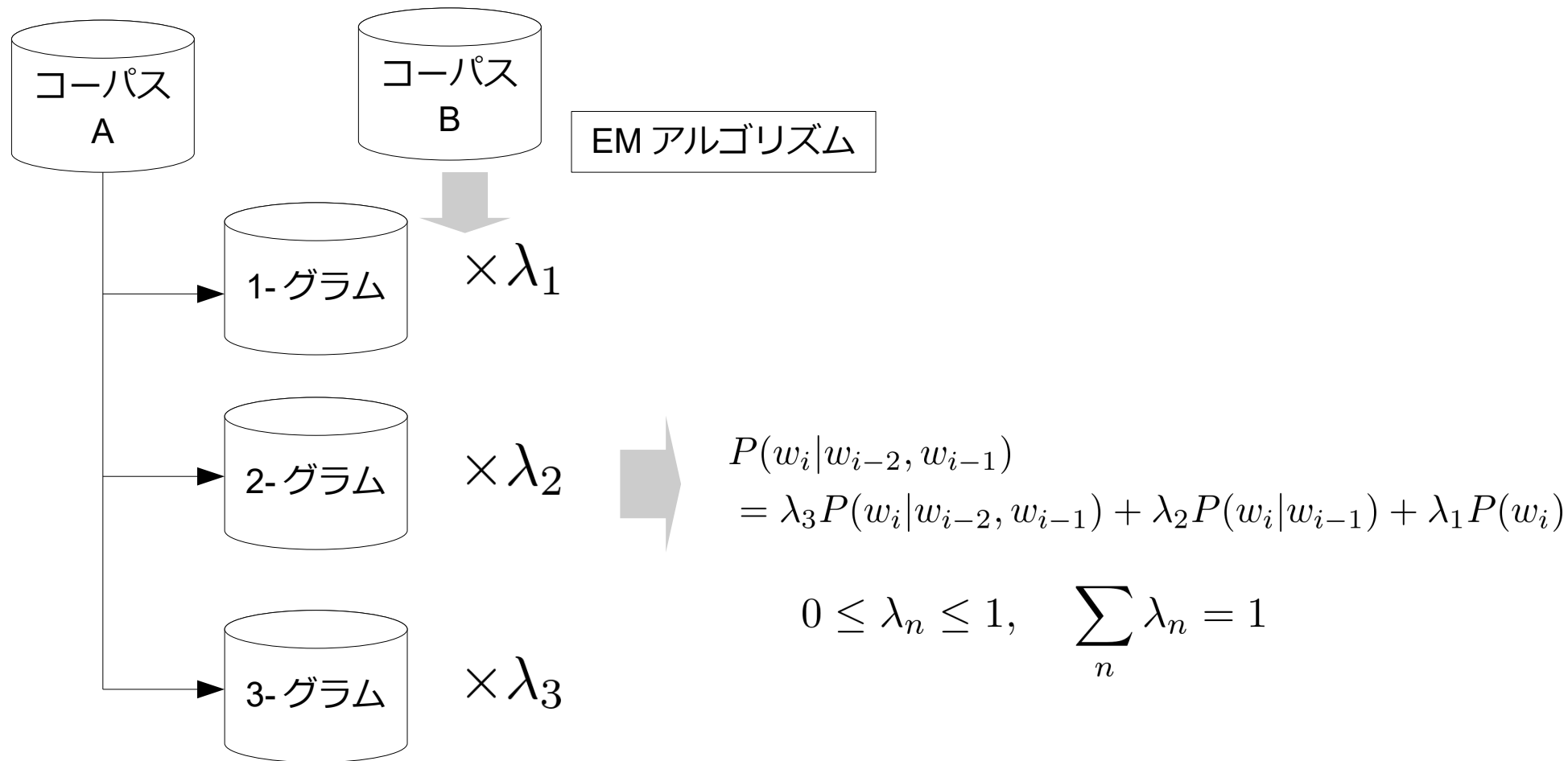
比較的小さな値

徐々に 1 に近づく



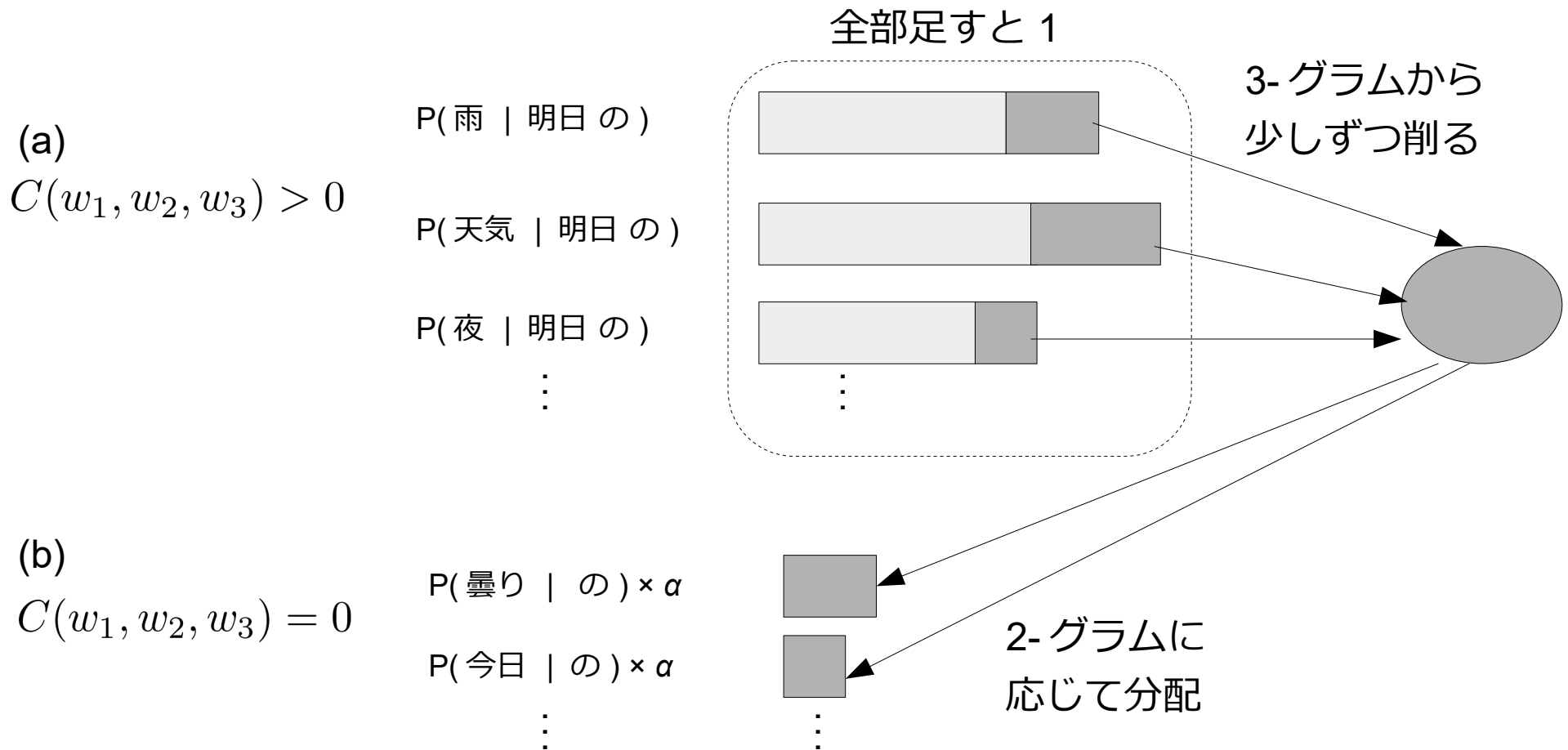
13.4 信頼できるモデルの力を借りる

- 線形補間法
 - 複数の確率を重み付きで足し合わせて、観測されないデータの確率を補間



13.4 信頼できるモデルの力を借りる

• バックオフ・スプーキングの考え方



13.4 信頼できるモデルの力を借りる

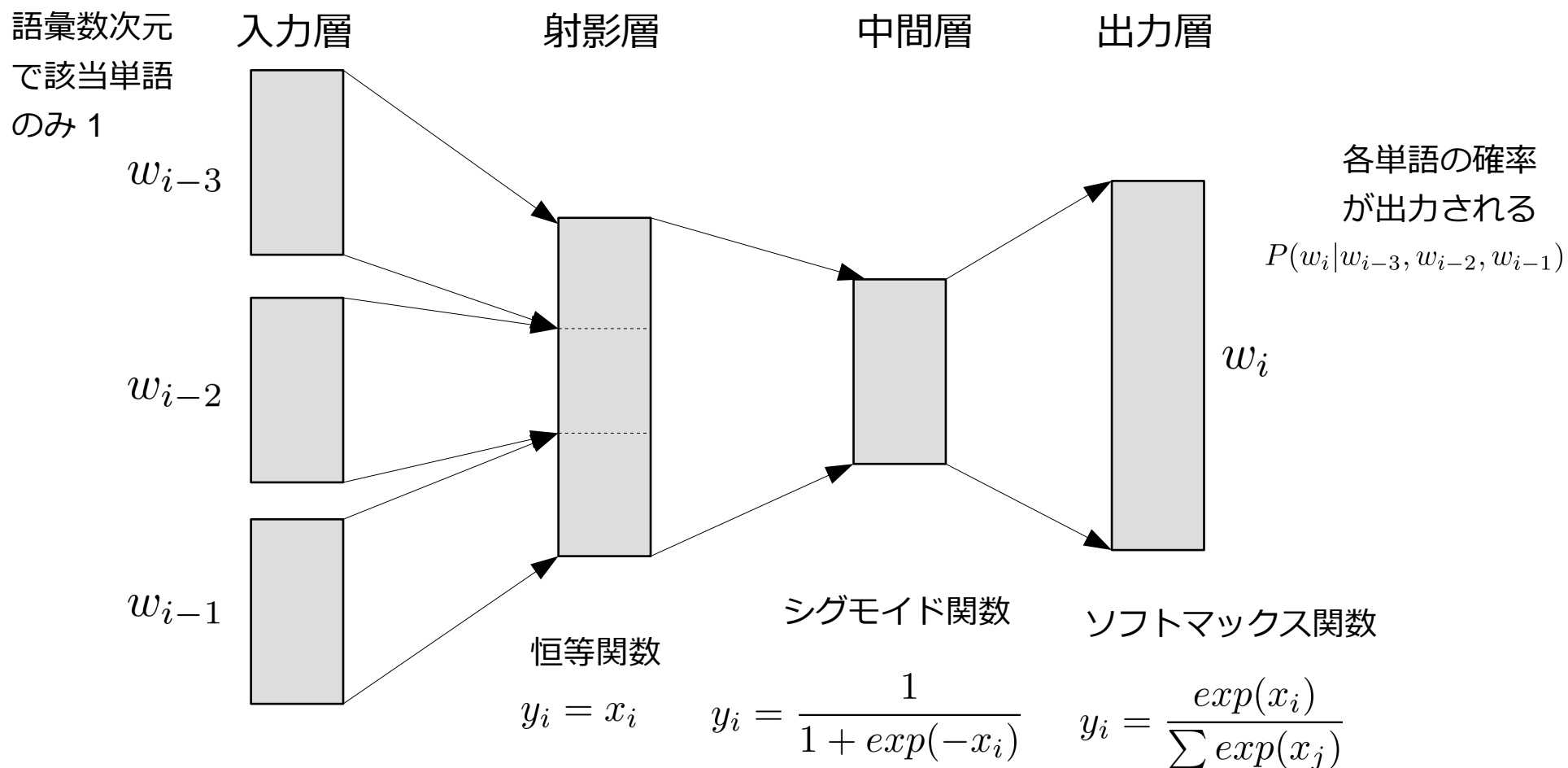
- バックオフ・スムージング
 - 学習データ中に出現しない N グラムの値を (N-1) グラムの値から推定する

$$P_3^{BO}(w_i|w_{i-2}, w_{i-1}) = \begin{cases} d(w_{i-2}, w_{i-1})P_3(w_i|w_{i-2}, w_{i-1}) & (C(w_{i-2}, w_{i-1}, w_i) > 0) \\ \alpha(w_{i-2}, w_{i-1})P_2^{BO}(w_i|w_{i-1}) & (C(w_{i-2}, w_{i-1}, w_i) = 0) \end{cases}$$

- バックオフ係数 d の求め方
 - Witten-Bell, Kneser-Ney, Modified Kneser-Ney

13.5 ニューラルネットワークを用いた言語モデル

- フィードフォワード型
 - 過去 N 単語から次単語の確率分布を求める



13.5 ニューラルネットワークを用いた言語モデル

- リカレント型
 - フィードバックで仮想的にすべての履歴を表現

