バイグラム分布の尤度比直接推定を可能とする基底関数の提案

川上　賢十† 　菊地　真人† 　吉田　光男† 　梅村　恭司†

† 豊橋技術科学大学　〒 441–8580 愛知県豊橋市天伯町雲雀ヶ丘 1–1

E-mail: †{k131820,m143313}@edu.tut.ac.jp, ††yoshida@cs.tut.ac.jp, †††umemura@tut.jp

あらまし　尤度比の推定は，異常検知，条件付き確率密度推定，確率的パターン認識などの多くのタスクに用いられている．尤度比の直接推定法では，モデルの選び方が重要であるが，バイグラム分布においては，適切な基底関数が知られていない．我々は，ユニグラムとバイグラムの関係に着目し，尤度比の直接推定に使用出来る基底関数を報告する．また，報告する基底関数を用いて，尤度比推定の定式化を行い，評価実験を行った．評価実験の結果，尤度比の直接推定を用いた推定が通常用いられる方法と比較し良い性能を示すことを確認した．

キーワード　テキストマイニング, 尤度比, 基底関数, 線形補間

1. はじめに

尤度比の推定は異常検知や特徴検出選択，条件付き確率密度推定，確率的パターン認識など，多くのタスクで使用される [1]．これらは，データの背後にある確率分布を推定し，尤度を求めることで，実現することが出来る．しかし，確率分布をそれぞれ推定し，尤度比を求めることは，推定誤差が大きくなる可能性がある．そこで，尤度比を構成する確率分布をそれぞれ推定するのではなく，尤度比を直接推定する手法が提案されている [2], [3]．しかし，既存の尤度比の直接推定手法は連続な分布を対象としている．尤度比の直接推定には，基底関数の選択が重要となるが，非連続な分布に対する基底関数は提案されていない．そこで，我々は，非連続な分布である，バイグラム分布の尤度比を直接推定する基底関数の提案する．また，提案した基底関数を元に，バイグラム分布の尤度比の直接推定を行った．そして，直接推定にて導出した推定式の性能を確認するために，カタカナの直前に出現するバイグラムの推定という評価用のタスクを用いて実験を行った．実験の結果，パラメータに依存するものの提案した基底関数を用いた推定が通常用いられる方法と比較し良い性能を示すことを確認した．

2. 関連研究

尤度比の直接推定法 [2] は，尤度比を構成する 2 つの確率の推定値を用いて推定を行うよりも，正確に推定を行うことが出来ると報告されている．この手法は，尤度比を構成する確率それぞれの精度を向上するのではなく，尤度比のそのものの推定誤差をコスト関数が最小になるように定式化し推定を行う．コスト関数は，観測データを用いて表現される．尤度比の推定が基底関数の線形結合によって表現出来ると仮定することによって，コスト最小化問題の解は各基底関数と尤度比の形状によって重みが加えられる．尤度比の直接推定法では，この基底関数の選択が重要となる．これまでに，ガウス関数のような，連続分布の場合に使用可能な基底関数が提案されている [3], [4]．本研究では，バイグラム分布のような非連続な分布に着目し研究を行う．

また，統計的なデータ解析の主なタスクは，データの背後にある確率分布を推定することであり，その確率分布を用いた，尤度比も同様に多くのタスクに用いられており，尤度比の推定は，異常検知や特徴選択，独立成分分析，条件付き確率密度推定などの様々なタスクに用いられている [1]．

バイグラムの尤度比を用いた応用研究としては，企業名の抽出が行われている [5]．この研究では，バイグラムの尤度比を用いて抽出を行っている．尤度比の推定性能向上が応用上の性能向上につながると期待できる．

3. 提案手法

我々は，非連続な分布であるバイグラム分布の尤度比の直接推定に使用可能な基底関数を提案する．また，連続分布の尤度比の直接推定に用いられる基底関数であるガウス関数は連続な確率空間の近傍を，尤度比の推定に用いる能力がある．バイグラム分布において近傍の確率空間を用いることは出来ないが，ユニグラムの尤度比を利用できるのではないかと考えた．そこで，線形補間を用いて，バイグラム分布の尤度比の直接推定にユニグラムの尤度比を用いる．提案する基底関数は，これを可能とする基底関数である．

本論文では，バイグラムの出現確率を $P(w_i \in R_c, w_j \in R_c)$ を用いて表す．これは “$w_i w_j$” という文字列が出現する確率である．よって，$P(w_i \in R_c, w_j \in R_c)$ と $P(w_j \in R_c, w_i \in R_c)$ は異なる確率を表す．また，$P(w_i \in R_c, *)$ は 1 文字目に w_i をとり，2 文字目を限定しない確率となり，2 文字目について周辺化を行った確率であり，$P(w_i, *) = \sum_{j=1}^{R_c} P(w_i, w_j)$ となる．また R_c データ中に存在する全ての文字空間である．

まず，本論文で使用する関数を以下のように定義する．

$$p(w_i, w_j) := P(w_i \in R_c, w_j \in R_c) \tag{1}$$

$$p(w_i, w_j \mid \mathcal{D}) := P(w_i \in R_c, w_j \in R_c \mid \mathcal{D}) \tag{2}$$

$$r(w_i, w_j) := \frac{P(w_i \in R_c, w_j \in R_c \mid \mathcal{D})}{P(w_i \in R_c, w_j \in R_c)} \tag{3}$$

ここで $\mathcal{D}$ は推定を行いたい条件を表しており，推定の内容により変化する．

以降，バイグラムの尤度比である $r(w_i, w_j)$ を推定することを目的として定式化を行う．

まず，単純なバイグラムの尤度比の直接推定を行う．次に，バイグラムの線形補間 [6] を用いてユニグラムを使用する尤度比の直接推定を行う．

3.1 バイグラム分布の尤度比推定

単純なバイグラム分布の尤度比の直接推定を行う．提案の基底関数に対して，直接推定手法として uLSIF を選んだ．具体的な式は，基底関数によって変化するが，式の運用方法は [2] と同様である．まず，バイグラムの尤度比 $r(w_i, w_j)$ を学習によって決定されるパラメータ $\boldsymbol{\alpha}$ と基底関数 $\boldsymbol{\phi}$ を用いて以下のように定義する．

$$\hat{r}(x, y) := \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\phi}(x, y) \tag{4}$$

$$= \sum_{i,j=1}^{v} \alpha_{ij} \phi_{ij}(x, y) \tag{5}$$

ここで $\boldsymbol{\alpha}$ はベクトルであり，α_{ij} は $\boldsymbol{\alpha}$ の $(i \times v + j)$ 番目の要素を表す．α_{ij} はバイグラム分布を扱うために，式の簡単化のために，ベクトルの要素にもかかわらず添え字を 2 つ使用している．同様に $\boldsymbol{\phi}$ はベクトルであり，ϕ_{ij} は $\boldsymbol{\phi}$ の $(i \times v + j)$ 番目の要素を表す．また v はデータの種類数，今回は全ての文字の種数を表す．

ここで我々は以下のような基底関数を提案する．

$$\phi_{ij}(x, y) := \begin{cases} 1 & (x = w_i, y = w_j) \\ 0 & otherwise \end{cases} \tag{6}$$

この基底関数は，尤度比の推定を行うバイグラムのみを見ていることになる．

基底関数の定義により $\hat{r}(w_i, w_j)$ を以下のように展開することが出来る．

$$\hat{r}(w_i, w_j) = \sum_{i',j'=1}^{v} \alpha_{i'j'} \phi_{ij}(w_i, w_j) \tag{7}$$

$$= \alpha_{ij} \tag{8}$$

そして，コスト関数 J_0 を以下のように定義する．

$$J_0(\boldsymbol{\alpha}) := \frac{1}{2} \sum_{i,j=1}^{v} (\hat{r}(w_i, w_j) - r(w_i, w_j))^2 p(w_i, w_j) \tag{9}$$

$$= \frac{1}{2} \sum_{i,j=1}^{v} \hat{r}(w_i, w_j)^2 p(w_i, w_j) - \sum_{i,j=1}^{v} \hat{r}(w_i, w_j) p(w_i, w_j \mid \mathcal{D}) + C \tag{10}$$

$$C := \frac{1}{2} \sum_{i,j=1}^{v} r(w_i, w_j)^2 p(w_i, w_j) \tag{11}$$

このコスト関数が最小になった時に，二乗誤差が最小となる．よって，このコスト関数が最小となるようにパラメータ α の推定を行う．ここで，学習により決定するパラメータ $\boldsymbol{\alpha}$ を含まない項を削除し J を定義する．

$$J(\boldsymbol{\alpha}) := J_0(\boldsymbol{\alpha}) - C \tag{12}$$

$$= \frac{1}{2} \sum_{i,j=1}^{v} \hat{r}(w_i, w_j)^2 p(w_i, w_j) - \sum_{i,j=1}^{v} \hat{r}(w_i, w_j) p(w_i, w_j \mid \mathcal{D}) \tag{13}$$

また，確率 p を推定量 $\hat{p}$ にした，コスト関数の推定 $\hat{J}$ は以下のようになる．

$$\hat{J}(\boldsymbol{\alpha}) := \frac{1}{2} \sum_{i,j=1}^{v} \hat{r}(w_i, w_j)^2 \hat{p}(w_i, w_j) - \sum_{i,j=1}^{v} \hat{r}(w_i, w_j) \hat{p}(w_i, w_j \mid \mathcal{D}) \tag{14}$$

ここで $\hat{p}$ の推定方法は限定しない．本論文における実験では最尤推定量を用いた．

コスト関数に，安定化のために l_2 正則化項を追加し，以下の最適化問題を得ることが出来る．ここで，パラメータ λ は 0 以上の実数となる．

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^{v \times v}} \left[\hat{J}(\boldsymbol{\alpha}) + \frac{\lambda}{2} \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\alpha} \right] \tag{15}$$

この最適化問題は解析的に解くことが出来る．

すべての α_{ij} てそれぞれ微分し，イコール 0 とおく．

$$\frac{\partial}{\partial \alpha_{ij}} (\hat{J}(\boldsymbol{\alpha}) + \frac{\lambda}{2} \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\alpha}) = 0 \tag{16}$$

$$\alpha_{ij} (\hat{p}(w_i, w_j) + \lambda) - \hat{p}(w_i, w_j \mid \mathcal{D}) = 0 \tag{17}$$

$$\alpha_{ij} = \frac{\hat{p}(w_i, w_j \mid \mathcal{D})}{\hat{p}(w_i, w_j) + \lambda} \tag{18}$$

uLSIF では，α_{ij} を非負とするために，以下のように補正した推定量 $\hat{\alpha_{ij}}$ を得る．

$$\hat{\alpha_{ij}} := \max(0, \alpha_{ij}) \tag{19}$$

だが，今回導出した，α_{ij} は常に正となる．そのため，補正は必要なく，我々の基底関数の定義により，$\hat{r}(w_i, w_j)$ は以下のようになる．

$$\hat{r}(w_i, w_j) = \frac{\hat{p}(w_i, w_j \mid \mathcal{D})}{\hat{p}(w_i, w_j) + \lambda}$$

パラメータ λ はデータから決定する必要がある．また，$\lambda = 0$ の場合は，間接推定と同一の推定式になる．

3.2 線形補間を用いたバイグラム分布の尤度比推定

線形補間を用いて，バイグラム分布の尤度比を推定する．推定する尤度比 $\hat{r}(w_i, w_j)$ を次のモデルで表現する．

$$\hat{r}(w_i, w_j) := \mu \hat{r}_b(w_i, w_j) + (1 - \mu) \hat{r}_u(w_i, *) \hat{r}_u(*, w_j) \tag{20}$$

ここで，$r_b(x, y)$ はバイグラムに対する尤度比，$r_u(x, y)$ はユニグラムに対する尤度比とする．また，$\mu (\in [0, 1])$ は線形補間のパラメータであり，データから決定される．

まず，独立して計算可能な $r_u(x, *)$ の尤度比を推定する．$r_u(x, *)$ は以下のように定義する．

$$r_u(x, *) := \frac{p(x, * \mid \mathcal{D})}{p(x, *)}$$

$r_u(x,*)$ を学習により決定されるパラメータ $\boldsymbol{\beta}$ と，基底関数 $\boldsymbol{\phi}_u$ を用いて，以下のようにモデル化する．

$$\hat{r}_u(x,*) := \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\phi}_u(x,*) \tag{21}$$

$$= \sum_{i=1}^{v_u} \beta_i \phi_{u_i}(x,*) \tag{22}$$

$\boldsymbol{\beta}$ はベクトルであり，β_i は $\boldsymbol{\beta}$ の i 番目の要素である，同様に $\boldsymbol{\phi}_u$ もベクトルであり，ϕ_{u_i} は $\boldsymbol{\phi}_u$ の i 番目の要素である．ここで，v_u はデータに含まれるユニグラムの種類数であり，ベクトルの次元数でもある．

そして，我々は，ユニグラムの尤度比の推定のために以下の基底関数を提案する．

$$\phi_{u_i}(x,*) := \begin{cases} 1 & (x = w_i) \\ 0 & otherwise \end{cases} \tag{23}$$

この基底関数の定義により $\hat{r}_u(x,*)$ は以下のように展開することが出来る．

$$\hat{r}_u(x,*) = \sum_{i=1}^{v_u} \beta_i \phi_{u_i}(x,*) \tag{24}$$

$$= \beta_i \tag{25}$$

この $\hat{r}_u(x,y)$ を真の尤度比 $r_u(x,y)$ の二乗誤差を最小にするように決定していく．コスト関数を以下のように定義する．

$$J_{u0}(\boldsymbol{\beta}) := \frac{1}{2}\sum_{i=0}^{v_u} (\hat{r}_u(w_i,*) - r_u(w_i,*))^2 p(w_i,*) \tag{26}$$

これを展開し，定数を除外すると以下のようになる．

$$J_u(\boldsymbol{\beta}) := \frac{1}{2}\sum_{i=1}^{v_u} \hat{r}_u(w_i,*)^2 p(w_i,*) - \sum_{i=1}^{v_u} \hat{r}_u(w_i,*) p(w_i,* \mid \mathcal{D}) \tag{27}$$

この二乗誤差を最小とするようにパラメータ $\boldsymbol{\beta}$ を決定する．

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^{v_u}} \left[\hat{J}_u(\boldsymbol{\beta}) + \frac{\lambda}{2}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta} \right]$$

この最適化問題は解析的に解くことが出来る．すべての β_{ij} でそれぞれ微分し，0 と置き，最適解を得る．

$$\frac{\partial}{\partial \beta_i}(\hat{J}_u(\boldsymbol{\beta}) + \frac{\lambda}{2}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}) = 0 \tag{28}$$

$$\beta_i(\hat{p}(w_i,*) + \lambda) = \hat{p}(w_i,* \mid \mathcal{D}) \tag{29}$$

$$\beta_i = \frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i,*) + \lambda} \tag{30}$$

また，この β_{ij} は常に正となり，補正は必要ない．よって，定義より $\hat{r}_u(w_i,*)$ は以下のようになる．

$$\hat{r}_u(w_i,w_j) = \sum_{i=1}^{v_u} \beta_i \phi_i(x,*) \tag{31}$$

$$= \beta_i \tag{32}$$

$$= \frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i*) + \lambda} \tag{33}$$

また，同様に処理を行い $\hat{r}_u(*,w_j)$ の推定は以下のようになる．

$$\hat{r}_u(*,w_j) = \frac{\hat{p}(*,w_j \mid \mathcal{D})}{\hat{p}(*,w_j) + \lambda} \tag{34}$$

上記の推定結果を用いて，$\hat{r}(w_i,w_j)$ の推定式は以下のようになる．

$$\hat{r}(w_i,w_j) = \mu\hat{r}_b(w_i,w_j) + (1-\mu)\frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i,*) + \lambda}\frac{\hat{p}(*,w_j \mid \mathcal{D})}{\hat{p}(*,w_j) + \lambda} \tag{35}$$

続いて，$r_b(w_i,w_j)$ の推定を行う．$r(x,y)$ は $p(x,y \mid \mathcal{D})$ と $p(x,y)$ の比であり，推定量は学習により決定するパラメータ $\boldsymbol{\alpha}$ と基底関数 $\boldsymbol{\phi}$ を用いて以下のようにモデル化する．

$$\hat{r}(w_i,w_j) := \mu\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(x,y) + (1-\mu)U(w_i,w_j) \tag{36}$$

$$U(w_i,w_j) := \frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i,*) + \lambda}\frac{\hat{p}(*,w_j \mid \mathcal{D})}{\hat{p}(*,w_j) + \lambda} \tag{37}$$

$U(x,y)$ はユニグラムを用いたバイグラムの推定式をまとめた関数である．基底関数は式 6 を用いる．ここで $\boldsymbol{\alpha}$ はベクトルであり，α_{ij} は $\boldsymbol{\alpha}$ の $(i \times v + j)$ 番目の要素を表す．α_{ij} はバイグラム分布を扱うために，式の簡単化のために，ベクトルの要素にもかかわらず添え字を 2 つ使用している．同様に $\boldsymbol{\phi}$ はベクトルであり，ϕ_{ij} は $\boldsymbol{\phi}$ の $(i \times v + j)$ 番目の要素を表す．また v はデータの種類数，今回は全ての文字の種類数を表す．

この基底関数の定義により，$\hat{r}(w_i,w_j)$ を以下のように変形することが出来る．

$$\hat{r}(w_i,w_j) = \mu\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(x,y) + (1-\mu)U(w_i,w_j) \tag{38}$$

$$= \mu\sum_{i,j=1}^{v} \alpha_{ij}\phi_{ij}(x,y) + (1-\mu)U(w_i,w_j) \tag{39}$$

$$= \mu\alpha_{ij} + (1-\mu)U(w_i,w_j) \tag{40}$$

$\hat{r}(w_i,w_j)$ と $r(w_i,w_j)$ の二乗誤差を最小とするようにパラメータ $\boldsymbol{\alpha}$ を決定する．

$$J_0 := \frac{1}{2}\sum_{i,j=1}^{v} (\hat{r}(w_i,w_j) - r(w_i,w_j))^2 p(w_i,w_j) \tag{41}$$

$$= \frac{1}{2}\sum_{i,j=1}^{v} \hat{r}(w_i,w_j)^2 p(w_i,w_j) - \sum_{i,j=1}^{v} \hat{r}(w_i,w_j) p(w_i,w_j \mid \mathcal{D}) + C_r \tag{42}$$

$$= F - S + C_r \tag{43}$$

$$C_r := \sum_{i,j=0}^{v} r(w_i,w_j)^2 p(w_i,w_j) \tag{44}$$

v はバイグラムの種類数であり，パラメータの次元数となっている．また，展開式の第一項を F，第二項を S としている．

F を定義より展開すると以下のようになる，

$$F = \frac{1}{2}\sum_{i,j=1}^{v} \hat{r}(w_i,w_j)^2 \hat{p}(w_i,w_j) \tag{45}$$

$$= \frac{1}{2}\sum_{i,j=1}^{v} (\mu\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(x,y) + (1-\mu)U(w_i,w_j))^2 \hat{p}(w_i,w_j) \tag{46}$$

$$= \frac{1}{2}\sum_{i,j=1}^{v} \{\mu^2(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(x,y))^2 + 2\mu(1-\mu)\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(x,y)U(w_i,w_j) + \{(1-\mu)U(w_i,w_j)\}^2\}\hat{p}(w_i,w_j) \tag{47}$$

$$= \frac{1}{2}\sum_{i,j=1}^{v} \{(\mu^2(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_i))^2 + 2(\mu-\mu^2)\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_i)U(w_i,w_j))\}\hat{p}(w_i,w_j) + C_F \tag{48}$$

ここで C_F は $\boldsymbol{\alpha}$ に関わらない定数をまとめた項である．

S を定義より展開すると以下のようになる．

$$S = \sum_{i,j=1}^{v} \mu\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j)p(w_i,w_j \mid \mathcal{D}) + C_S \tag{49}$$

ここで C_S は $\boldsymbol{\alpha}$ に関わらない定数をまとめた項である．

これらにより最小化する目的関数 J は J_0 から定数を除外し以下のようになる．

$$J := J_0 - C_r - C_F - C_S \tag{50}$$

J に安定化のために l_2 正則化項を追加し以下の最適化問題を得ることが出来る．

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{v\times v}} \left[\frac{1}{2}\sum_{i,j=1}^{v}\{(\mu^2(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_i))^2 + 2(\mu-\mu^2)\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_i)U(w_i,w_j))\}\hat{p}(w_i,w_j) - \sum_{i,j=1}^{v}\mu\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j)p(w_i,w_j \mid \mathcal{D}) + \frac{\lambda}{2}\boldsymbol{\alpha}^{T}\boldsymbol{\alpha}\right]$$

この最適化問題は解析的に解くことが出来る．

すべての α_{ij} でそれぞれ微分しイコール 0 とおく．

$$\mu^2\alpha_{ij} + 2(\mu-\mu^2)U(w_i,w_j)p(w_i,w_j) - \mu p(w_i,w_j \mid \mathcal{D}) + \lambda\alpha_{ij} = 0 \tag{51}$$

$$\alpha_{ij}(\mu^2 p(w_i,w_j) + \lambda) = \mu p(w_i,w_j \mid \mathcal{D}) - 2(\mu-\mu^2)Up(w_i,w_j) \tag{52}$$

$$\alpha_{ij} = \frac{\mu p(w_i,w_j \mid \mathcal{D}) - 2(\mu-\mu^2)U(w_i,w_j)p(w_i,w_j)}{\mu^2 p(w_i,w_j) + \lambda} \tag{53}$$

α_{ij} は負の値を取る可能性があるため，uLSIF の考え方に従って，以下のように補正を行う．

$$\hat{\alpha_{ij}} := \max(0, \frac{\mu p(w_i,w_j \mid \mathcal{D}) - 2(\mu-\mu^2)U(w_i,w_j)p(w_i,w_j)}{\mu^2 p(w_i,w_j) + \lambda}) \tag{54}$$

導出した $\hat{\alpha_{ij}}$ を $\hat{r}(w_i,w_j)$ に代入することで以下の推定式を得る．

$$\begin{aligned}\hat{r}(w_i,w_j) &:= \mu\hat{\alpha_{ij}} + (1-\mu)U(w_i,w_j)\\ U(w_i,w_j) &:= \frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i,*)+\lambda} \times \frac{\hat{p}(*,w_j \mid \mathcal{D})}{\hat{p}(*,w_j)+\lambda}\\ \hat{\alpha_{ij}} &:= \max(0, \frac{\hat{p}(w_i,w_j \mid \mathcal{D}) - 2(1-\mu)U(w_i,w_j)\hat{p}(w_i,w_j)}{\mu\hat{p}(w_i,w_j) + \frac{\lambda}{\mu}})\end{aligned}$$

ここで $\hat{p}$ の推定方法は限定しない．本論文における実験では最尤推定量を用いた．

4. 実　　験

提案した基底関数を用いた，バイグラム分布の尤度比の推定性能を評価することが実験の目的である．直接推定の性能を確認するために最尤推定と，線形補間を用いたバイグラム分布の直接推定の性能を確認するためにバイグラムの線形補間を行った尤度比とそれぞれ比較を行う．

比較対象の式はそれぞれ以下のようになる．

- 直接推定

$$\hat{r}_{de}(w_i,w_j) = \frac{\hat{p}(w_i,w_j \mid \mathcal{D})}{\hat{p}(w_i,w_j)+\lambda}$$

- 最尤推定

$$\hat{r}_{hl}(w_i,w_j) = \frac{\hat{p}(w_i,w_j \mid \mathcal{D})}{\hat{p}(w_i,w_j)}$$

- 直接推定 (線形補間)

$$\begin{aligned}\hat{r}(w_i,w_j) &:= \mu\hat{\alpha_{ij}} + (1-\mu)U(w_i,w_j)\\ U(w_i,w_j) &:= \frac{\hat{p}(w_i,* \mid \mathcal{D})}{\hat{p}(w_i,*)+\lambda} \times \frac{\hat{p}(*,w_j \mid \mathcal{D})}{\hat{p}(*,w_j)+\lambda}\\ \hat{\alpha_{ij}} &:= \max(0, \frac{\hat{p}(w_i,w_j \mid \mathcal{D}) - 2(1-\mu)U(w_i,w_j)\hat{p}(w_i,w_j)}{\mu\hat{p}(w_i,w_j) + \frac{\lambda}{\mu}})\end{aligned}$$

- 線形補間

$$\hat{r}_{li}(w_i,w_j) = \frac{\mu p(w_i,w_j \mid \mathcal{D}) + (1-\mu)p(w_i,* \mid \mathcal{D})p(*,w_j \mid \mathcal{D})}{\mu p(w_i,w_j) + (1-\mu)p(w_i,*)p(*,w_j)}$$

また，実験では，$\hat{p}$ は最尤推定によって求められるバイグラムやユニグラムの出現頻度であり，スムージングなどは行わない．この 4 手法のうち，直接推定を用いている推定式が提案手法となる．

評価用のタスクには，尤度比そのものの推定性能を評価するという目的のために，評価のための正解が明快なタスクを選択した．具体的には新聞の記事データを用いてカタカナの直前に出現するバイグラムの推定を行った．次の尤度比を推定対象とする．

$$\frac{p(w_i, w_j \mid \mathcal{D})}{p(w_i, w_j)}$$

分子はカタカナの直前のバイグラムの頻度，分母はデータ全体のバイグラムの頻度になっている．これは，バイグラムがカタカナの直前に出現する確率に，そのバイグラムがデータ全体で出現する確率を用いて補正を行うという式になる．つまり，この尤度比の値が大きければ大きいほど，カタカナの直前に出現しやすいバイグラムということになる．

実験ではまず，新聞の記事データから学習データとテストデータをランダムにサンプリングする．学習データとテストデータをそれぞれ，全体のバイグラム，カタカナの直前に出現するバイグラムに分ける．学習データの全体のバイグラムとカタカナの直前に出現するバイグラムを用いて，事前にバイグラムの出現頻度を計算する．そして，テストデータの全体のバイグラムから，そのバイグラム 1 つ 1 つの尤度比の推定値を計算していく．この推定値が大きければ大きいほど，カタカナの直前のバイグラムである可能性が高いため，バイグラムの推定値を元に，降順でソートする．ソートされたバイグラムを上位から順に，テストデータのカタカナの直前に出現するバイグラムの中に存在するかを確認していく．カタカナの直前に出現するバイグラム中にバイグラムが存在すれば正解，出現しなければ不正解とする．バイグラムがカタカナの直前とそれ以外の両方に出現する場合は正解とする．そうして，テストデータのバイグラムの上位 n 件での再現率を 10,000 件までの再現率を評価する．再現率は以下の式で表される．

$$\text{Recall}(i) = \frac{\text{上位 } i \text{ 件の正解バイグラム数}}{\text{カタカナの直前のバイグラム数}}$$

評価には，上位 1 件から 10,000 件までの再現率を合計し，それぞれの手法での再現率の面積として使用する．

$$\text{Area} = \sum_{i=1}^{10000} \text{Recall}(i)$$

この評価では，面積が大きければ大きいほど，精度良くバイグラムの尤度比の推定が行えていることになる．

4.1 実験設定

今回比較を行う手法にはいくつかのパラメータが存在する．今回は以下の 2 組の組み合わせのパラメータを使用した．

（1） $\lambda = 0.1$, $\mu = 0.9$

（2） $\lambda = 0.01$, $\mu = 0.99$

パラメータ λ は正則化項に由来し分母に対してどの程度のペナルティを与えるかを表している．パラメータ μ は線形補間において，バイグラムとユニグラムのどちらの値を重視するかを表している．

また，実験データとして毎日新聞の記事データを 1991 年から 1997 年まで用いた [7]．実験データからそれぞれ学習用データセットとして 10,000 件，テスト用データセットから 1,000 件の記事を重複がないようにサンプリングを行った．サンプリングを行った記事を文字バイグラム単位に分割し，そのバイグラムにカタカナの直前かどうかのラベル付けを行った．頻度の計算にはこのバイグラムを用いる， $p(w_i, w_j)$ においてはバイグラム全体を， $p(w_i, w_j \mid \mathcal{D})$ においてはラベル付けされたバイグラムを用いる．

実験データ中に含まれる，全体のバイグラムの数，種数，カタカナの直前のバイグラムの数，種数を表 1 に示す．

4.2 実験結果

それぞれのパラメータによる Area の実験結果を表 2,3 に示す．データに対して，それぞれ最も良い手法に対して，下線が引かれている．パラメータ (1) に関しては，いずれのデータに対しても，直接推定が最も良い性能を示しており，パラメータ (2) に関しては，線形補間を用いた直接推定手法が良い性能を示している．また，パラメータ (1) と比較し，パラメータ (2) において，線形補間を用いた直接推定手法がよい性能を示しているが，線形補間を用いない直接推定手法においては性能に大きな変化は見られなかった．

それぞれのパラメータにおける 1991 年における再現率の変化をグラフに描画したものを図 1,2 に示す．上位 5,000 件では大きな差はないが，以降，直接推定手法を用いた提案式がそれぞれよい性能を示していることが分かる．このことから，提案した基底関数が合理的であると解釈できる．

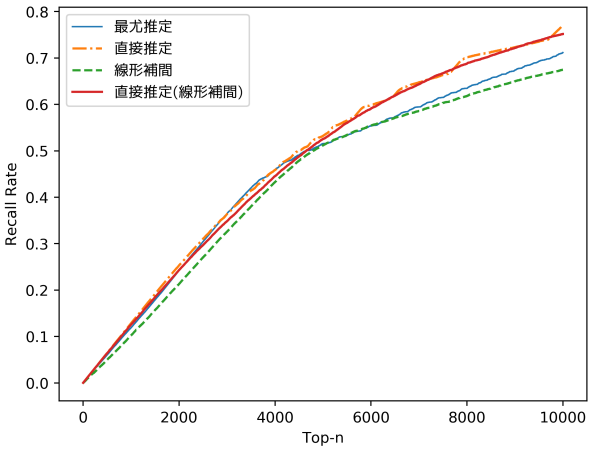

図 1　実験結果: $\lambda = 0.1$, $\mu = 0.9$ 1991 年

4.3 考　　察

実験結果の図 1 を見ると，上位 5,000 件以降において，最尤推定と線形補間を用いた尤度比の推定式の性能が低下していることが分かる．これは，上位 5,000 件までに，推定を行いやすいバイグラム，例えばバイグラムの両方がカタカナである場合などが全て出現したためではないかと考える．直接推定手法では，尤度比の分母にパラメータにより補正を付加している．この補正により，直接推定手法では尤度比を低く見積もる効果がある．この効果はサンプル数が少ない場合に強く影響する．サンプル数が少ない点は，推定が困難な点であり，今回の実験において推定が困難な点の尤度比を低く見積もることで上位において，性能が向上したのではないかと考える．

今回の実験において，パラメータの組み合わせによって線形補間を用いた直接推定の性能が変化した．これは，パラメータ

表 1　実験データ概要

	学習データ				テストデータ			
	全体		カタカナの直前		全体		カタカナの直前	
年度	バイグラム数	種数	バイグラム数	種数	バイグラム数	種数	バイグラム数	種数
1991 年	4,335,064	213,587	320,398	25,523	463,217	73,415	34,810	7,761
1992 年	4,133,346	215,184	303,392	25,524	397,745	68,003	29,239	7,266
1993 年	4,288,344	217,645	320,778	26,438	419,212	70,597	31,613	7,616
1994 年	4,616,557	228,645	353,105	28,462	465,018	74,970	36,037	8,219
1995 年	4,244,038	224,320	306,623	27,018	404,239	69,996	30,537	7,695
1996 年	4,756,396	234,527	362,732	28,354	479,139	78,377	39,644	8,668
1997 年	4,983,617	235,641	376,324	28,813	484,853	75,715	36,640	8,270

表 2　実験結果 (Area): $\lambda = 0.1$, $\mu = 0.9$

年度	直接推定	最尤推定	直接推定 (線形補間)	線形補間
1991 年	4786.20	4521.10	4716.24	4337.76
1992 年	5010.16	4727.23	4926.19	4557.71
1993 年	4832.99	4547.18	4765.73	4387.20
1994 年	4649.72	4388.30	4559.78	4265.30
1995 年	4820.05	4541.39	4762.51	4401.67
1996 年	4496.23	4246.16	4418.16	4120.42
1997 年	4650.10	4397.73	4569.48	4250.72

表 3　実験結果 (Area): $\lambda = 0.01$, $\mu = 0.99$

年度	直接推定	最尤推定	直接推定 (線形補間)	線形補間
1991 年	4786.20	4521.10	4819.70	4263.70
1992 年	5010.15	4727.23	5028.27	4484.95
1993 年	4833.01	4547.18	4870.81	4319.07
1994 年	4649.73	4388.30	4664.22	4206.89
1995 年	4820.08	4541.39	4857.98	4331.06
1996 年	4496.21	4246.16	4520.15	4066.76
1997 年	4650.10	4397.73	4672.71	4193.91

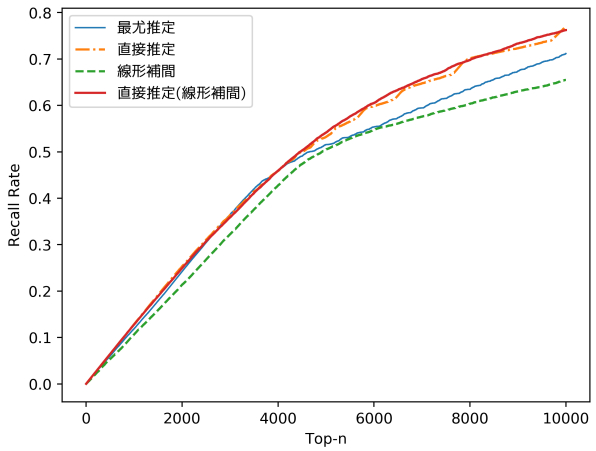

図 2　実験結果:$\lambda = 0.01$, $\mu = 0.99$ 1991 年

の決定方法の重要性を示唆している．今回は，推定式の性能を確認するため，比較手法のパラメータを固定し，比較を行った．線形補間のパラメータである μ の決定に関しては，EM アルゴリズムなどを用いて決定されることがあり，これらのアルゴリズムの使用を検討する必要がある [8]．また，正則化項によって導入されるパラメータである λ に関しては，決定方法などが提案されておらず，今後，決定方法を検討していく必要がある．

今回の実験において，確率の推定量 $\hat{p}$ として最尤推定量を用いている．これは，今回の実験は尤度比の推定精度を比較することを目的としていたためである．確率の推定値 $\hat{p}$ の補正には多くのスムージング手法が用いられる [6]．これらのスムージング手法と，提案手法は併用することが出来る．提案手法を実際のタスクに用いる場合には適切なスムージングを選択することでよりよい性能を得ることが期待できる．

5. ま と め

バイグラム分布の尤度比の直接推定のための基底関数の提案を行った．提案した基底関数を用いて，バイグラム分布の尤度比の直接推定と，線形補間を行ったバイグラム分布の直接推定を行った．実験の結果．パラメータに依存するものの，線形補間を用いた尤度比の直接推定手法がよい性能を示すことを確認した．また，線形補間を用いた式においては，パラメータの組み合わせにより，性能が大きく変化することを確認した．

今後は，線形補間のパラメータである μ，正則化項により導入されるパラメータ λ の決定方法について考える必要がある．

文　献

[1] 杉山将. 密度比推定によるビッグデータ解析. 電子情報通信学会誌, Vol. 97, No. 5, pp. 353–358, 2014.

[2] Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, Vol. 10 (Jul.), pp. 1391–1445, 2009.

[3] Masashi Sugiyama, Ichiro Takeuchi, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Daisuke Okanohara. Least-squares conditional density estimation. *IEICE Transactions on Information and Systems*, Vol. E93-D, No. 3, pp. 583–594, 2010.

[4] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, Cambridge, 2012.

[5] 中野翔平, 吉田光男, 梅村恭司. 企業名抽出への密度比推定の適用. 第 8 回データ工学 と情報マネジメントに関するフォーラム (DEIM 2017), Vol. F8-1, , 2017.

[6] Christopher D. Manning and Hinrich Schüetze. *Foundations of statistical natural language processing*. The MIT Press, Cambridge, MA, 1999.

[7] 毎日新聞社. CD-毎日新聞データ集. 日外アソシエーツ, 1991–1997.

[8] 北研二. 言語と計算 (4) 確率的言語モデル. 東京大学出版会, 1999.