バイグラムの尤度比の直接推定法における周辺分布を用いた定式化

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@edu.tut.ac.jp, ‡‡ m143313@edu.tut.ac.jp, ‡‡‡ yoshida@cs.tut.ac.jp, ‡‡‡‡ umemura@tut.jp

あらまし　尤度比の推定は，異常検知や確率的パターン認識，文脈の推定など多くのタスクで用いられている．単純な手法で尤度比を推定するには，尤度比の定義に従い，尤度比を構成する個々の確率分布を最尤推定し，その比を求めるという手法が考えられる．この手法では，まず確率分布の推定を行い，推定値の比を求めるという二段階の推定を行う事になり，推定誤差を大きくしてしまう可能性がある．そこで，尤度比を直接推定する手法が提案されている．我々は尤度比の直接推定法を用いてバイグラムの尤度比の定式化を行う．連続な標本空間における尤度比の直接推定には，ガウス関数のような基底関数が用いられてきた．ガウス関数を基底関数に用いることで，学習データに存在しない事象の推定を行う場合，サンプリングされた要素の近傍の値を用いることで，尤度比の値を推定することが出来る．そこで，バイグラムの確率分布に対する周辺分布と考えられる，ユニグラムの確率分布を用いてバイグラムの尤度比を定式化する．この定式化により，学習データに存在しないバイグラムに対しても尤度比の推定を行う．

キーワード　テキストマイニング, バイグラム, 尤度比, uLSIF

1 はじめに

尤度比の推定は，異常検知や確率的パターン認識，文脈の推定など多くのタスクで用いられている [1]．尤度比は尤度関数の比で表されるため，推定を行う必要がある．単純な手法で尤度比を推定するには，尤度比の定義に従い，尤度比を構成する個々の確率分布を最尤推定し，その比を求めるという手法が考えられる．このような尤度比を構成する個々の確率分布を推定し，その比を求める手法を間接推定と呼ぶ．しかし，この手法では，まず確率分布の推定を行い，推定値の比を求めるという二段階の推定を行う事になり，推定誤差を大きくしてしまう可能性がある．そこで，尤度比を構成する確率分布の推定を経由せずに，尤度比を直接推定する手法が提案されている [2]．尤度比の直接推定を用いると，確率分布の推定を介する単純な手法よりも，推定誤差が小さくなることが報告されている．

直接推定において，基底関数は尤度比の形状を表現するために使用されるため基底関数の選択が重要である．連続分布の尤度比に対しては，基底関数としてガウス関数よく用いられている．ガウス関数を基底関数として用いることにより，推定を行う点の近傍の情報を用いることが出来る．しかし，ガウス関数では，確率の標本空間が離散であることを考慮することは出来ない．そこで，離散分布のそれぞれの事象に独立な基底関数を用いる手法が提案されている [3]．

尤度比の直接推定はこれまでは多くの研究において連続分布を対象に研究されてきた．連続分布では，推定を行いたい点における標本を集めることが困難であるため，尤度比の直接推定が有効である [4]．離散分布であっても，例えば自然言語のバイグラムのような膨大な組み合わせが存在する場合や，サンプルサイズが小さい場合などにおいては十分にサンプルを集めることは困難であり，離散分布においても，尤度比の直接推定は有効であると考える．

本研究では，[3] にて提案された基底関数を用いて，バイグラムの尤度比を定式化する．また，バイグラムの確率分布の周辺分布と考えられるユニグラムの確率分布を用いて，バイグラムの尤度比の定式化を行う．ガウス関数を基底関数に用いることで，サンプリングされた要素の近傍の値を用いる効果があり，これにより学習データに存在しないデータに対しても推定を行うことが出来る．ユニグラムの確率分布を用いることで，関数を基底関数に用いることと類似の作用をえる事を期待している．

実験には，カタカナ語の直前に出現するバイグラムの推定という単純な実験を行う．学習に使用するデータの量を変化させ，定式化したバイグラムの尤度比の推定式により，学習データに存在しないバイグラムに対して推定が行えているか，ユニグラムを用いることにより性能が低下していないかを確認する．

2 関連研究

尤度比の直接推定法 [2] は，確率分布の推定を介する単純な手法よりも，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分布の推定精度を向上するのではなく，尤度比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．尤度比をパラメータと基底関数の線形和によって表現出来ると仮定する．このパラメータをコスト関数が最小となるように決定し，尤度比の推定を行う．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．これまでの研究では，ガウス関数が基底関数によく用いられてきた [4], [5]．しかし，ガウス関数では，確率の標本空間が離散であることを考慮することは出来ない．本論文では，バイグラムのような非連続な分布に着目し研究を行う．これまでの研究において，連続分布では推定を行いたい点における標本を集めることが困難

であり，離散分布の場合は容易であるとされている．しかし，バイグラムのような膨大な組み合わせが存在する場合や，サンプルサイズが少ない場合などにおいては十分な標本を集めることは困難である．よって，我々は離散分布に置いても尤度比の直接推定は有用であると考えている．

これまでに，バイグラムの尤度比を直接推定する手法が提案されている [3]．この手法では，バイグラムの尤度比に対して直接推定可能な基底関数を提案し，それを用いて推定式の定式化を行った．また，カタカナ語の直前に出現するバイグラを推定するという単純な実験を用により提案手法の有効性を検証した．この手法では，尤度比の推定時に尤度比の分母の確率の推定値にパラメータ λ を加算することで，分母の確率の値に応じて，尤度比の推定値に補正を行っている．しかし，この手法ではバイグラムの頻度のみを使用するため，学習データに存在しないバイグラムに対しては，尤度比を推定することができない．

確率を補正する手法としてはスムージングがよく用いられる．自然言語を対象にしたスムージングの一つとして Good–Turing スムージング [6] が存在する．Good–Turing スムージングは自然言語の出現頻度が Zipf の法則に従う事を利用して，ゼロ頻度の出現確率を推定し，その推定値を用いて全体の確率の補正を行う．また，n–gram モデルに対するスムージングに低次の (n-1)–gram を用いた手法としては，バックオフスムージング [7] や，その改良の Kneser–Ney スムージング [8] などが存在する．学習データ中に存在しない n–gram に対しては，(n-1)–gram の頻度や，種類数を用いて推定を行う．しかし，スムージング手法では，尤度比を構成する個々の確立に対する補正を行う事は出来るが，尤度比そのものに対しての補正として、効果がある方法とはかぎらない．

3 尤度比の直接推定

尤度比の直接推定手法には二乗誤差を用いた手法 [2]，ロジスティック回帰を用いた手法 [9]，カルバック・ライブラー距離を用いた手法 [10], エントロピーという条件下において最尤推定を用いた手法 [11] などが提案されている．その中で，本研究では二乗誤差を用いた手法である，unconstrained Least-Squares Importance Fitting（uLSIF）[2] を用いて定式化を行う．uLSIF は上記の手法のうち，解析的に尤度比の直接推定を行える唯一の手法である．本節では，uLSIF について説明を行う．

推定を行う尤度比を以下のように定義する．

$$r(\boldsymbol{x}) := \frac{p_{nu}(\boldsymbol{x})}{p_{de}(\boldsymbol{x})} \tag{1}$$

ここで $\boldsymbol{x}$ は入力で，標本空間からサンプリングされた要素である．学習により決定されるパラメータのベクトル $\boldsymbol{\alpha}$ と基底関数のベクトル $\boldsymbol{\phi}$ の線形和で尤度比が表現が出来ると仮定する．尤度比の直接推定において，この基底関数の選択が重要となる．

$$\hat{r}(\boldsymbol{x}) := \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}) \tag{2}$$

$$= \sum_{i=1}^{v} \alpha_i \phi_i(\boldsymbol{x}) \tag{3}$$

uLSIF では $r(\boldsymbol{x})$ と $\hat{r}(\boldsymbol{x})$ の二乗誤差を最小とするようにパラメータの決定を行う．コスト関数 $J_o(\boldsymbol{\alpha})$ を以下のように定義する．

$$J_0(\boldsymbol{\alpha}) := \frac{1}{2}\int (\hat{r}(\boldsymbol{x}) - r(\boldsymbol{x}))^2 p_{de}(\boldsymbol{x}) d\boldsymbol{x} \tag{4}$$

$$= \frac{1}{2}\int \hat{r}(\boldsymbol{x})^2 p_{de}(\boldsymbol{x}) d\boldsymbol{x} - \int \hat{r}(\boldsymbol{x}) p_{nu}(\boldsymbol{x}) d\boldsymbol{x} + \frac{1}{2}\int r(\boldsymbol{x})^2 p_{de}(\boldsymbol{x}) d\boldsymbol{x} \tag{5}$$

ここで，目的関数を最小化するパラメータ $\boldsymbol{\alpha}$ に関わらない項を取り除き， $J(\boldsymbol{\alpha})$ を定義する．

$$J(\boldsymbol{\alpha}) := J_0(\boldsymbol{\alpha}) - \frac{1}{2}\int r(\boldsymbol{x})^2 p_{de}(\boldsymbol{x}) d\boldsymbol{x} \tag{6}$$

そして，データから確率を推定し，コスト関数の推定式 $\hat{J}(\boldsymbol{\alpha})$ を得る．

$$\hat{J}(\boldsymbol{\alpha}) := \frac{1}{2}\sum_{i=1}^{n} \hat{r}(\boldsymbol{x}_i)^2 \hat{p}_{de}(\boldsymbol{x}_i) - \sum_{i=1}^{n} \hat{r}(\boldsymbol{x}_i)\hat{p}_{nu}(\boldsymbol{x}_i) \tag{7}$$

ここで， n はサンプルサイズである．$\hat{J}(\boldsymbol{\alpha})$ に安定化のため l_2 正則化項を付加し以下の最適化問題を得る．

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{\boldsymbol{v}}} \left[\hat{J}(\boldsymbol{\alpha}) + \frac{\lambda}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha}\right] \tag{8}$$

ここで， v はパラメータの次元数である．λ は正則化パラメータであり，データにより決定される．この最適化問題を解きパラメータ $\boldsymbol{\alpha}$ を決定する．この最適化問題は，制約がなく凸であるため，解析的に解くことが出来る．よって，この最適化問題にて，目的関数が最小となるパラメータ $\tilde{\boldsymbol{\alpha}}$ を決定する．そして，尤度比であるためパラメータが非負である必要があるため，uLSIF では以下のように補正を行う．

$$\hat{\boldsymbol{\alpha}} := \max(\mathbf{0}_{\boldsymbol{v}}, \tilde{\boldsymbol{\alpha}}) \tag{9}$$

ここで， $\mathbf{0}_{\boldsymbol{v}}$ は v 次元の要素が全て 0 のベクトルであり，この max は $\boldsymbol{\alpha}$ の全ての要素に個別に適用される．以上の手順により $r(\boldsymbol{x})$ の推定量 $\hat{r}(\boldsymbol{x})$ を求める．

4 提 案 手 法

ここでは，バイグラム “w_iw_j” の出現確率を $P(w_i \in R_c, w_j \in R_c)$ と表す．R_c は全ての文字空間を表す．また， w_i は文字空間内の i 番目の要素を表す．$P(w_i \in R_c, w_j \in R_c)$ はバイグラムの出現確率であるが，定式化の簡単化のために，引数を 2 つ用いる．$P(w_i \in R_c, w_j \in R_c)$ はバイグラム “w_iw_j” の出現確率を表すため $P(w_i \in R_c, w_j \in R_c)$ 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，つまり，2 文字目について周辺化を行った確率であり，$P(w_i, *) = \sum_{x\in R_c} P(w_i, x)$ となる．

この定式化で使用する関数を以下のように定義する．

$$p(w_i, w_j) := P(w_i \in R_c, w_j \in R_c) \tag{10}$$
$$p(w_i, w_j \mid \mathcal{D}) := P(w_i \in R_c, w_j \in R_c \mid \mathcal{D}) \tag{11}$$
$$p(w_i, *) := P(w_i \in R_c, *) \tag{12}$$
$$p(*, w_j) := P(*, w_j \in R_c) \tag{13}$$
$$p(w_i, * \mid \mathcal{D}) := P(w_i \in R_c, * \mid \mathcal{D}) \tag{14}$$
$$p(*, w_j \mid \mathcal{D}) := P(*, w_j \in R_c \mid \mathcal{D}) \tag{15}$$

ここで， $\mathcal{D}$ は推定を行う条件を表している．実験では，カタカナ後の直前という条件を使用している．

推定対象の尤度比を以下のように定義する．

$$r(w_i, w_j) := \frac{p(w_i, w_j \mid \mathcal{D})}{p(w_i, w_j)} \tag{16}$$

[3] の研究においては，バイグラムの尤度比に対して，バイグラムのみを用いた定式化が行われいる．この定式化では，学習データにおいて出現頻度が低いバイグラムに対して強く補正を行うことで，推定性能を向上させている．しかし，学習データに出現しないバイグラムに対しては，推定値がゼロとなってしまう問題がある．

提案手法では，学習データに存在しないバイグラムに対して推定を行うためにバイグラムの確率分布の周辺分布であるユニグラムの確率分布を用いることを考える．ユニグラムの独立を仮定すると，バイグラムの尤度比をその周辺分布であるユニグラムを用いて以下のように表現できる．

$$\frac{P(w_i, w_j \mid \mathcal{D})}{P(w_i, w_j)} = \frac{P(w_i, * \mid \mathcal{D})}{P(w_i, *)} \times \frac{P(*, w_j \mid \mathcal{D})}{P(*, w_j)}$$

このユニグラムの尤度比を用いて構成したバイグラムの尤度比を補正する形式で推定値を以下のように定めた．

$$\hat{r}(w_i, w_j) := \hat{e}(w_i, w_j) + \hat{r}_u(w_i, *)\hat{r}_u(*, w_j)$$

ここで， $\hat{r}_u$ はユニグラムの尤度比の推定値であり， $\hat{e}$ は補正関数である．まず，独立して計算可能な，ユニグラムの尤度比を定式化する．基底関数には，推定する事象に対して独立な基底関数を用いる．そして，推定したユニグラムの尤度比を用いて，バイグラムの尤度比全体の真値と推定値の二乗誤差が最小となるように，補正関数を定式化する．推定した補正関数と，ユニグラムの尤度比で構成したバイグラムの尤度比を用いて，バイグラムの尤度比の推定式とする．

4.1 定 式 化

バイグラムの尤度比の推定値を補正関数とユニグラムの尤度比を用いて以下のように定義する．

$$\hat{r}(w_i, w_j) := \hat{e}(w_i, w_j) + \hat{r}_u(w_i, *)\hat{r}_u(*, w_j) \tag{17}$$

これは，ユニグラムの独立を仮定することで，ユニグラムの尤度比を用いてバイグラムの尤度比を表現している．ここで， $\hat{r}_u$ はユニグラムの尤度比の推定量である．ユニグラムの尤度比を以下ように定義する．

$$r_u(w_i, *) := \frac{p(w_i, * \mid \mathcal{D})}{p(w_i, *)} \tag{18}$$
$$r_u(*, w_j) := \frac{p(*, w_j \mid \mathcal{D})}{p(*, w_j)} \tag{19}$$

また， $\hat{e}(w_i, w_j)$ はバイグラムとユニグラムの頻度を用いた補正関数である．定式化では，まずユニグラムの尤度比の推定を行う．そしてユニグラムの尤度比と補正関数を用いて，バイグラムの尤度比の真値との二乗誤差が最小となるように推定を行う．

まず，ユニグラムの尤度比 $r_u(w_i, *)$ の推定を行う．尤度比 $r_u(w_i, *)$ の推定量 $\hat{r}_u(w_i, *)$ をパラメータ $\boldsymbol{\beta}$ と，基底関数 $\boldsymbol{\phi}_u$ を用いて表す．

$$\hat{r}_u(w_i, *) := \boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{\phi}_u(w_i, *) \tag{20}$$
$$= \sum_{i'=1}^{v_u} \beta_{i'} \phi_{u_{i'}}(w_i, *) \tag{21}$$

ここで，$\boldsymbol{\beta}$ 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．現実のデータに対しては，i は文字コード， v_u は文字コードの種類数に対応する．

ユニグラムの尤度比の推定のために以下の事象に対して独立な基底関数を定義する．

$$\phi_{u_i}(x, *) := \begin{cases} 1 & (x = w_i) \\ 0 & otherwise \end{cases} \tag{22}$$

ここで，x は関数の引数で w_i のような R_c の要素であり，基底関数の定義の簡単化のために使用している．この基底関数の定義により $\hat{r}_u(w_i, *)$ は以下のように展開することが出来る．

$$\hat{r}_u(w_i, *) = \sum_{i'=1}^{v_u} \beta_{i'} \phi_{u_{i'}}(w_i, *) \tag{23}$$
$$= \beta_i \tag{24}$$

この $\hat{r}_u(w_i, *)$ を真の尤度比 $r_u(w_i, *)$ 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．

$$J_{u0}(\boldsymbol{\beta}) := \frac{1}{2} \sum_{i=0}^{v_u} (\hat{r}_u(w_i, *) - r_u(w_i, *))^2 p(w_i, *) \tag{25}$$

これを展開し，定数を除外した $J_u(\boldsymbol{\beta})$ を以下のように定義する．

$$J_u(\boldsymbol{\beta}) := \frac{1}{2} \sum_{i=1}^{v_u} \hat{r}_u(w_i, *)^2 p(w_i, *) - \sum_{i=1}^{v_u} \hat{r}_u(w_i, *) p(w_i, * \mid \mathcal{D}) \tag{26}$$

データから $J_u(\boldsymbol{\beta})$ の推定値 $\hat{J}_u(\boldsymbol{\beta})$ を得る．

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

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．

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^{v_u}}\left[\hat{J}_u(\boldsymbol{\beta})+\frac{\lambda_u}{2}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}\right] \tag{28}$$

ここで， λ_u は正則化パラメータであり，データから決定される．この最適化問題は $\boldsymbol{\beta}$ に関する二次関数であり，係数は全て非負であるため下に凸な関数である．よって最適化問題は解析的に解くことが出来る．$\boldsymbol{\beta}$ で微分し，$\mathbf{0}_{\boldsymbol{v}}$ と置き，最適解を得る．

$$\frac{\partial}{\partial\boldsymbol{\beta}}(\hat{J}_u(\boldsymbol{\beta})+\frac{\lambda_u}{2}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta})=\mathbf{0}_{\boldsymbol{v}} \tag{29}$$

微分した i 番目の要素に対して解くと以下のようになる．

$$\frac{\partial}{\partial\beta_i}(\hat{J}_u(\boldsymbol{\beta})+\frac{\lambda_u}{2}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta})=0 \tag{30}$$

$$\beta_i(\hat{p}(w_i,*)+\lambda_u)=\hat{p}(w_i,*\mid\mathcal{D}) \tag{31}$$

$$\beta_i=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_u} \tag{32}$$

また，この β_i 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．

$$\hat{r}_u(w_i,*)=\sum_{i'=1}^{v_u}\beta_{i'}\phi_{u_{i'}}(w_i,*) \tag{33}$$

$$=\beta_i \tag{34}$$

$$=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i*)+\lambda_u} \tag{35}$$

また，$\hat{r}_u(*,w_j)$ の場合も同様に定式化すると，その推定量は以下のようになる．

$$\hat{r}_u(*,w_j)=\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_u} \tag{36}$$

定式化したユニグラムの尤度比を使用して， $\hat{r}(w_i,w_j)$ を以下のように表す．

$$\hat{r}(w_i,w_j)=\hat{e}(w_i,w_j)+U(w_i,w_j) \tag{37}$$

$$U(w_i,w_j):=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_u}\times\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_u} \tag{38}$$

ここで，補正関数 $\hat{e}(w_i,w_j)$ をデータにより決定されるパラメータ $\boldsymbol{\alpha}$ と基底関数 $\boldsymbol{\phi}(w_i,w_j)$ を用いてて表し，尤度比の推定値を以下のように表す．

$$\hat{r}(w_i,w_j)=\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j)+U(w_i,w_j) \tag{39}$$

$$=\sum_{i',j'=1}\alpha_{i'j'}\phi_{i'j'}(w_i,w_j)+U(w_i,w_j) \tag{40}$$

ここで，$\boldsymbol{\phi}$ 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．v は変数 w_i と w_j が取り得る事象の数となる．実験では，データに出現した文字の種類数となっている．また，$\boldsymbol{\alpha}$ 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．

基底関数は，[3] にて提案されている，以下の基底関数を用いる．

$$\phi_{ij}(x,y):=\begin{cases}1 & (x=w_i,y=w_j)\\ 0 & otherwise\end{cases} \tag{41}$$

ここで，x と y は関数の引数で w_i のような R_c の要素であり，基底関数の定義の簡単化のために使用している．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．

$$\hat{r}(w_i,w_j)=\sum_{i',j'=1}\alpha_{i'j'}\phi_{i'j'}(w_i,w_j)+U(w_i,w_j) \tag{42}$$

$$=\alpha_{ij}+U(w_i,w_j) \tag{43}$$

バイグラムの尤度比の推定量 $\hat{r}(w_i,w_j)$ と真値 $r(w_i,w_j)$ の二乗誤差が最小となるようにパラメータ $\boldsymbol{\alpha}$ の推定を行う．以下のようにコスト関数 $J_0(\boldsymbol{\alpha})$ を定義する．

$$J_0(\boldsymbol{\alpha}):=\frac{1}{2}\sum_{i,j=1}^{v}(\hat{r}(w_i,w_j)-r(w_i,w_j))^2p(w_i,w_j) \tag{44}$$

$$=\frac{1}{2}\sum_{i,j=1}^{v}\hat{r}(w_i,w_j)^2p(w_i,w_j) - \sum_{i,j=1}^{v}\hat{r}(w_i,w_j)p(w_i,w_j\mid\mathcal{D})+C_1 \tag{45}$$

$$C_1:=\frac{1}{2}\sum_{i,j=1}^{v}r(w_i,w_j)^2p(w_i,w_j) \tag{46}$$

ここで， C_1 はパラメータである $\boldsymbol{\alpha}$ を含まないため最小化に関わらない項である．

続いて， $\hat{r}(w_i,w_j)$ を以下のように展開する．

$$J_0(\boldsymbol{\alpha})=\frac{1}{2}\sum_{i,j=1}^{v}(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j))^2p(w_i,w_j) + \sum_{i,j=1}^{v}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j)U(w_i,w_j)p(w_i,w_j) - \sum_{i,j=1}^{v}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\phi}(w_i,w_j)p(w_i,w_j\mid\mathcal{D})+C_2 \tag{47}$$

$$C_2:=\frac{1}{2}\sum_{i,j=1}^{v}U(w_i,w_j)^2p(w_i,w_j) - \sum_{i,j=1}^{v}U(w_i,w_j)p(w_i,w_j\mid\mathcal{D})+C_1 \tag{48}$$

最適化に関係のない項 C_2 を削除しコスト関数 $J(\boldsymbol{\alpha})$ を得る．

$$J(\boldsymbol{\alpha}):=J_0(\boldsymbol{\alpha})-C_2 \tag{49}$$

データから p を推定し J の推定値 $\hat{J}$ を得る．

$\hat{J}(\boldsymbol{\alpha})$ に安定化のために l_2 正則化項を追加し，以下のような最小化問題を得る．

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{v\times v}}\left[\hat{J}(\boldsymbol{\alpha})+\frac{\lambda_b}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha}\right] \tag{50}$$

ここで， λ_b は正則化パラメータであり，データから決定される．これは誤差関数を構成するパラメータ $\boldsymbol{\alpha}$ を決定する最小化問題である．このパラメータは補正関数を表現するためのパラメータであるため $\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha}$ の正則化項があるのは妥当である．そして，$\boldsymbol{\alpha}$ は負数になっても問題がないため，制約は必要ない．この最適化問題は， $\boldsymbol{\alpha}$ に関する二次関数であり，係数は全て非負であるため下に凸な関数である．よって最適化問題は，解

析的に解くことが出来る．$\boldsymbol{\alpha}$ で微分し $\mathbf{0}_{v\times v}$ と置き，最適解を得る．微分した，$(i\times v+j)$ 番目の要素に対して解くと以下のようになる．

$$\frac{\partial}{\partial\alpha_{ij}}(\hat{J}+\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha})=0 \tag{51}$$

$$\alpha_{ij}\hat{p}(w_i,w_j)+U(w_i,w_j)\hat{p}(w_i,w_j) - \hat{p}(w_i,w_j\mid\mathcal{D})+\alpha_{ij}\lambda_b=0 \tag{52}$$

$$\alpha_{ij}(\hat{p}(w_i,w_j)+\lambda_b)= \hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j) \tag{53}$$

$$\alpha_{ij}=\frac{\hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j)}{\hat{p}(w_i,w_j)+\lambda_b} \tag{54}$$

式 (54) を式 (43) に代入すると以下のようになる．

$$\tilde{r}(w_i,w_j)=\frac{\hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j)}{\hat{p}(w_i,w_j)+\lambda_b} +U(w_i,w_j) \tag{55}$$

$$U(w_i,w_j)=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_u}\times\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_u} \tag{56}$$

尤度比は非負である必要があるため，以下のように尤度比の推定値全体に補正を行い，バイグラムの尤度比の推定式を得る．

$$\hat{r}(w_i,w_j)=\max(\tilde{r}(w_i,w_j),0) \tag{57}$$

$$\tilde{r}(w_i,w_j)=\frac{\hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j)}{\hat{p}(w_i,w_j)+\lambda_b} +U(w_i,w_j) \tag{58}$$

$$U(w_i,w_j)=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_u}\times\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_u} \tag{59}$$

式 (58) の第一項がバイグラムとユニグラムの頻度を用いた補正関数，第二項がユニグラムのみを用いたバイグラムの尤度比である．補正関数の分子は $\hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j)$ となっている．この補正関数の分子の第一項はバイグラムを用いた尤度比の分子の確率の推定値である．補正関数の分子の第二項はユニグラムを用いて表現したバイグラムの尤度比とバイグラムの尤度比の分母の確率の推定値の積であり，つまりバイグラムとユニグラムを用いて表現した尤度比の分子の確率の推定値である．これは，バイグラムのみで表現した尤度比の分子の確率の推定値と，バイグラムとユニグラムを用いて表現した尤度比の推定値の差を取ることで補正を行っていることになる．

5 実　　験

実験では，カタカナ語の直前に出現するバイグラムの推定という単純な実験を行う．例えば，“平方メートル” という文字列があったとすると，“メートル” がカタカナ語となり，“平方” がカタカナ語の直前に出現するバイグラムとなる．正解が明確であり，尤度比の推定性能を定量的に評価出来る実験である．

実験では，以下の尤度比を推定する．

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

$P(w_i,w_j)$ はバイグラム “w_iw_j” が出現する確率である．また，$\mathcal{D}$ はカタカナ語の直前に出現するという条件である．つまり，この尤度比の値が高ければ高いほど，カタカナ語の直前に出現しやすいバイグラムであるということになる．

比較手法は以下の 5 種類を用いる．

（1） 提案手法

$$\hat{r}_{Proposed}(w_i,w_j)=\max(\tilde{r}(w_i,w_j),0)$$

$$\tilde{r}(w_i,w_j)=\frac{\hat{p}(w_i,w_j\mid\mathcal{D})-U(w_i,w_j)\hat{p}(w_i,w_j)}{\hat{p}(w_i,w_j)+\lambda_b} +U(w_i,w_j)$$

$$U(w_i,w_j)=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_u}\times\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_u}$$

（2） ユニグラムのみを用いた推定

$$\hat{r}_{Unigram}(w_i,w_j)=\frac{\hat{p}(w_i,*\mid\mathcal{D})}{\hat{p}(w_i,*)+\lambda_{uu}}\times\frac{\hat{p}(*,w_j\mid\mathcal{D})}{\hat{p}(*,w_j)+\lambda_{uu}}$$

（3） バイグラムのみを用いた推定

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

（4） 最尤推定を用いた間接推定

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

（5） Simple Good-Turing を用いた間接推定

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

ここで，$\hat{p}$ は最尤推定を用いた確率の推定値であり，$\hat{p}^*$ は Simple Good–Turing を用いた確率の推定値である．

提案手法は，ユニグラムとバイグラムの両方を用いた直接推定法である．それに対して，ユニグラムのみを用いた推定は，提案手法のユニグラム部分のみを取り出した手法であり，独立を仮定したユニグラムのみを用いた時の推定性能を確認する．バイグラムのみを用いた推定は，[3] にて提案された手法である．この手法は分母に加算された λ により補正を行い，頻度が低い場合に推定値を低く見積もる手法である．最尤推定を用いた間接推定は，最も単純な尤度比の推定方法であり，本実験においてはベースラインに相当する．Simple Good–Turing を用いた間接推定は，尤度比を構成する個々の確率分布の推定を補正する手法である．

5.1 実 験 手 順

実験では，1991–1997 年の毎日新聞記事コーパス [1] を用いて，各年度に対してそれぞれ実験を行う．各年度のデータから，学習データ，バリデーションデータ，テストデータをそれぞれ重複がないように抽出する．学習データについては，10,000 記事，2,500 記事の大小 2 種類用意する．バリデーションデータとテストデータにはそれぞれ 1,000 記事を用いる．

実験は以下の手順で行う．まず，学習データとバリデーショ

1 : http://www.nichigai.co.jp/sales/mainichi/mainichi-data.html (accessed 2018–12–28)

ンデータを用いて，それぞれの手法のパラメータを決定する．詳細については後述する．学習データと決定したパラメータを用いて，テストデータに出現する全てのバイグラムに対して，尤度比の値を推定する．推定した尤度比の値を用いて，バイグラムを降順にソートし，上位からランクを付ける．上位から順に正誤判定を行い，ランク毎に再現率を算出する．正誤判定には，判定を行うバイグラムが，テストデータ中のカタカナ後の直前に出現していれば正解，出現していなければ不正解とする．このランクと再現率を用いて，上位 10,000 件までのランク–再現率曲線をプロットする．ランク–再現率曲線は横軸にランク，縦軸にそのランクにおける再現率をプロットしたグラフである．

評価にはランク–再現率曲線を用いる．再現率は以下の式で表される．

$$\text{Recall} = \frac{\text{Number of true positive}}{\text{Number of true bigrams}}$$

ランク–再現率曲線の下の面積（AUC）の値が大きければ大きいほど，推定の性能が良いことになる．ランク–再現率曲線において，原点からあるランクの曲線上に位置する点を結ぶ直線の傾きが，そのランクでの適合率の大きさに比例する．適合率は以下の式で表される．

$$\text{Precision} = \frac{\text{Number of true positive}}{\text{Number of total output}}$$

5.2 パラメータの決定

パラメータの決定は，各手法，各年度ごとに行う．このパラメータはバリデーションデータを用いて決定する．

パラメータは上位 10,000 件までのランク–再現率曲線の面積を用いて決定する．ランク–再現率曲線の下の面積の値が大きければ大きいほど，推定の性能が良いことになる，よってこの面積が最大になるようにパラメータを決定する．

提案手法には，パラメータ λ_u, λ_b が存在する．それぞれのパラメータを 0.1 から 10^{-20} までの間を 0.1 倍ずつ変化させて，その中で，AUC の面積が最大になったパラメータを用いた．また，性能が同じになるパラメータについては，パラメータの値が小さい方を採用する．

また，比較手法であるユニグラムのみを用いた手法とバイグラムのみを用いた手法にはそれぞれそれぞれパラメータ λ_{uu}, λ が存在する．こちらは，0 から 1 までの間を 0.00001 刻みで探索を行い，その中で，AUC の面積が最大となったパラメータを用いた．また，性能が同じになるパラメータについては，パラメータの値が小さい方を採用する．

決定したパラメータをそれぞれ表 1,2 に示す．提案手法のパラメータは何れのデータセットにおいても $\lambda_u = 0.1, \lambda_b = 0.001$ となっている．

また，実験に使用した NLTK バージョン 3.2.5[2]における Simple Good-Turing のパラメータには，1 文字目のバイグラムの種数と 2 文字目のバイグラムの種数の積を用いた．

2 : https://www.nltk.org/ (accessed 2019–01–25)

表 1　学習データ 10,000 件の時のチューニングしたパラメータ

year	λ_u	λ_b	λ_{uu}	λ
1991	0.1	0.001	0.00526	0.00495
1992	0.1	0.001	0.00493	0.00476
1993	0.1	0.001	0.00895	0.05398
1994	0.1	0.001	0.01125	0.00470
1995	0.1	0.001	0.00956	0.00420
1996	0.1	0.001	0.99997	0.07287
1997	0.1	0.001	0.96414	0.03774

表 2　学習データ 2,500 件の時のチューニングしたパラメータ

year	λ_u	λ_b	λ_{uu}	λ
1991	0.1	0.001	0.00544	0.01925
1992	0.1	0.001	0.00374	0.00093
1993	0.1	0.001	0.00850	0.00475
1994	0.1	0.001	0.01308	0.01540
1995	0.1	0.001	0.00931	0.00337
1996	0.1	0.001	0.98831	0.04789
1997	0.1	0.001	0.99979	0.00835

表 3　学習データ 10,000 件: AUC TOP-10000

year	Proposed	Unigram	Bigram	MLE	SGT
1991	5236.17	3520.98	5232.97	3897.38	906.53
1992	5171.60	3591.00	5149.53	3831.91	781.21
1993	5111.82	3391.24	5109.50	3729.15	826.95
1994	5058.16	3376.45	5064.55	3573.85	762.20
1995	5203.72	3387.37	5193.74	3766.72	834.79
1996	4825.31	3239.57	4840.10	3396.49	815.50
1997	4997.26	3251.62	5001.48	3449.03	835.26

5.3 実験結果

学習データを 10,000 件用いた場合の実験結果を表 3 に示す．各年度において，最も性能の良い手法に対して下線が引いてある．提案手法と，バイグラムのみを用いた手法の性能が高く，あまり性能の差が無いことが分かる．学習データを 10,000 件用いた場合の 1991 年のランク–再現率曲線を図 1 に示す．提案手法と，バイグラムのみをもちいた手法のグラフがほぼ重なっていることが分かる．ユニグラムのみを用いた推定でも，上位においては最尤推定を用いた間接推定より良い性能を示している．最尤推定を用いた間接推定は，ほぼ直線のグラフになっており，全てのランクにおいて一定の正解率となっている．

学習データを 2,500 件用いた場合の実験結果を表 4 に示す．提案手法がいずれのデータにおいても，最も良い性能であることが分かる．学習データを 2,500 件用いた場合の 1991 年のランク–再現率曲線を図 2 に示す．どのランクにおいても，提案手法がもっとも良い性能であることが分かる．また．ランク 3,000 件から 6,000 件までのバイグラムのみを用いた手法の性能が低下している区間でも提案手法は安定した性能を示している．上位 6,000 件以降は，学習データ中に存在しないバイグラムが出現し，バイグラムのみを用いた手法，最尤推定を用いた間接推定の 2 手法では，尤度比の値がゼロとなり，推定出来てないことが分かる．そのような状況下においても，提案手法はユニグラムの尤度比を推定出来ている．

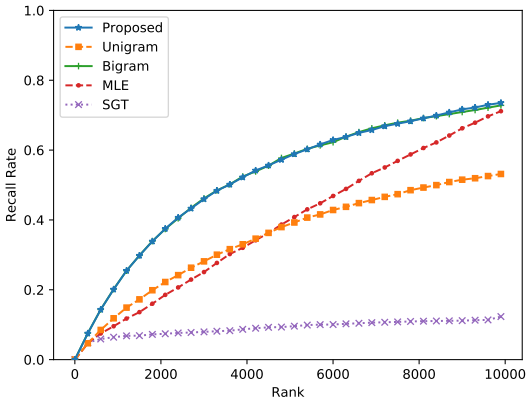

図 1　学習データ 10,000 件: 上位 10,000 件におけるランク–再現率曲線 (1991 年)

表 4　学習データ 2,500 件: AUC TOP-10000

year	Proposed	Unigram	Bigram	MLE	SGT
1991	4721.27	3471.35	4539.07	4013.22	731.79
1992	4489.98	3421.78	4316.80	3818.86	761.07
1993	4650.66	3388.82	4503.14	3937.51	644.32
1994	4530.18	3371.54	4351.56	3774.82	612.02
1995	4736.46	3380.14	4591.47	4008.43	753.08
1996	4398.24	3224.73	4241.56	3645.88	740.06
1997	4484.46	3206.43	4345.84	3778.61	683.06

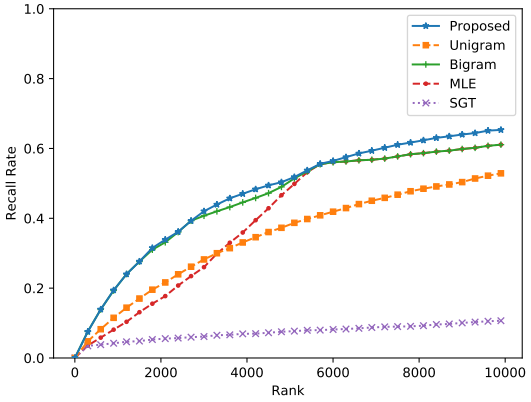

図 2　学習データ 2,500 件: 上位 10,000 件におけるランク–再現率曲線 (1991 年)

5.4　考　　察

図 1 を見ると，提案手法はデータが大きい場合は，バイグラムのみを用いた手法とほぼ同様の性能となり，バイグラムのみを用いた手法を除く他の手法よりも良い性能を示す．これは，推定値と真値の二乗誤差を最小とするという直接推定のフレームワークが，自然言語のような離散分布のデータに対しても有効であると考えている．

また，ユニグラムのみを用いた手法でも，最尤推定を用いた間接推定よりも，上位では良い性能を示している．これは，尤度比の分母の値に応じて補正を行うパラメータ λ_{uu} により，学習データ中に現れやすいバイグラムを上位に得ることが出来たためではないかと考えている．

Simple Good–Turing を用いた間接推定では上位では最尤推定を用いた間接推定を超える性能を示しているが，上位 200 件を超えると，性能が悪化している．これは，Zipf の法則を用いた，ゼロ頻度の補正を尤度比の分子と分母で個別に行っていることが問題になっているのではないかと考えている．分子の確率と分母の確率の補正を別々に行っているので，例えば学習データ 10,000 件の 1991 年のデータの場合は一度も出現しないバイグラムの尤度比の値は 18.987 という非常に高い値になる．これが原因で，学習データ中に存在しないバイグラムが上位に出現し，性能が悪化していると考えている．

図 2 を見ると，提案手法がもっとも良い性能であることが分かる．学習データが 2,500 件の場合には，上位 6,000 件以降において，バイグラムのみを用いた手法と最尤推定を用いた間接推定の 2 手法では分子の確率の推定値がゼロとなり，以降の尤度比の推定値はゼロとなる．提案手法は，そのような場合においても，ユニグラムの確率分布を用いて推定出来ている．また，学習データにバイグラムが存在する場合，例えば上位 3,000 件から 5,000 件の間でもバイグラムのみを用いた手法と比べて，性能が向上していることが分かる．これは，バイグラムの尤度比に対してユニグラムを用いた定式化を行ったことにより，ユニグラムの頻度を用いた尤度比の補正が適切に行えていることを示せていると考えている．

6　ま　と　め

バイグラムの尤度比に対して，バイグラムの確率分布の周辺分布であるユニグラムの確率分布を用いた定式化を行った．連続分布に対する直接推定では，ガウス関数を基底関数に用いることで，推定を行う点の近傍を推定に利用している．バイグラムでは，その周辺分布であるユニグラムを用いることで，頻度の小さいケースの推定に効果があるのではないかと考えた．

定式化したバイグラムの尤度比の推定式を用いて，バイグラムのみを用いた手法，ユニグラムのみを用いた手法，最尤推定を用いた間接推定，Simple Good–Turing を用いた間接推定などと比較を行った．実験には，カタカナ語の直前に出現するバイグラムの推定という単純な実験を行った．実験の結果，学習データが小さい場合に，提案手法はバイグラムの確率分布の周辺分布であるユニグラムの確率分布を用いて推定を行い，最も良い性能となることを確認した．また，学習データが大きい場合においても，バイグラムのみを用いた手法と同等の性能を示していることを確認した．

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] K. Kawakami, M. Kikuchi, M. Yoshida, and K. Umemura.

Direct Estimation of Likelihood Ratio for the Analysis of Context. In *Proceedings of the 5th International Conference on Advanced Informatics: Concept Theory and Applications*, pp. 1–6, 2018.

[4] 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.

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

[6] William A Gale. Good-Turing Smoothing Without Tears. *Journal of QuantitativeLinguistics*, Vol. 2, pp. 1–24, 1995.

[7] Slava M. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Trans. Acoustics, Speech, and Signal Processing*, Vol. 35, No. 3, pp. 400–401.

[8] R. Kneser and H. Ney. Improved backing-off for M-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, pp. 181–184 vol.1, 1995.

[9] Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative Learning for Differing Training and Test Distributions. In *Proceedings of the 24th International Conference on Machine Learning*, pp. 81–88, 2007.

[10] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, pp. 1433–1440, 2007.

[11] Taiji Suzuki, Masashi Sugiyama, Jun Sese, and Takafumi Kanamori. Approximating Mutual Information by Maximum Likelihood Density Ratio Estimation. In *Proceedings of the Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery at ECML/PKDD 2008*, pp. 5–20, 2008.