# 企業名抽出への密度比推定の適用

中野 翔平† 吉田 光男† 梅村 恭司†

†豊橋技術科学大学 情報・知能工学系 〒441-8580 愛知県豊橋市天伯町雲雀ヶ丘 1-1

E-mail: †{s133355@edu.tut.ac.jp, yoshida@cs.tut.ac.jp, umemura@tut.jp}

**あらまし** スムージングされた尤度を用いて，ナイーブベイズに基づいた尤度比の計算を行っている先行研究において，名前の周辺と名前を構成する文字列を特徴量としている抽出方法がある．本研究では，尤度比の計算に確率密度比推定手法を基にした，尤度の推定を行わない方法を採用して，カーネルの計算に文字列のスムージングされた頻度を使用することで，先行研究の名前抽出手法に適用可能にする方法を提案する．先行研究と同様の条件で新聞記事から企業名の抽出を行う比較実験を行った結果，近似された適合率及び近似された再現率のそれぞれにおいて提案手法が尤度を推定して尤度比を計算する先行研究を上回り，有意水準 1%で提案手法と先行研究の有意差が認められた．

**キーワード** 情報抽出，企業名抽出，N-gram，スムージング，密度比推定

## 1. はじめに

企業名や氏名の名前のリストを持っているならば，その名前から情報の検索や分類を行うことができる．また，テキストマイニングで文書中の言及されている名前のリストを求めることができれば，その文書の分析を行うことができる．このように同じ種類の名前のリストを求めることができれば有用であるだろう．

こうした同じ種類の名前のリストを作成する方法として，既存の辞書から名前を取り出し利用する方法や手作業でリストに名前を追加していく方法，形態素解析又は構文解析で名前を取り出し利用する方法が挙げられる．しかし，既存の辞書から名前を取り出す方法は新たな語が含まれないという問題がある．手作業で追加する方法は一から作成した場合，コストが膨大となる，最初だけ既存の辞書を用いたとしても新たな語が出続けるたびに追加していくのは同様にコストが大きい，また人為的なミスも発生しやすいという問題がある．形態素解析又は構文解析を利用する方法は固有名詞が抽出できたとしても，そこからは特定の種類の名前だけを人手で選別しなければならない，また辞書に含まれない名前が出現した場合に漏れが生じるという問題もある．これらの問題を解決するために先行研究において，企業名を適用例とした特定の種類の名前の抽出法が提案されている[菅野 2014, 中野 2015, 中野 2016]．この名前抽出法では，ナイーブベイズ及び最尤推定に基づく手法を用いて，評価値である尤度比を計算している．また，ナイーブベイズ及び最尤推定に基づく手法では未知語が現れた場合，本来の確率が 0 でないにも関わらず全体の尤度の推定値が 0 になるという問題があるため，この問題を解決するために確率補正（スムージング）を用いている．

しかし，尤度比を用いた手法では，分母と分子の尤度それぞれにスムージングを行うため，それぞれのスムージング値が本来の確率と異なっていた場合，尤度比の値に影響を与え，手法全体の性能を低下させてしまう場合がある．この問題を回避する手法として，分母分子ごとに確率密度を推定するのでは無く，確率密度比自体を直接推定するという密度比推定[杉山 2014]という手法がある．これを尤度比の推定に適用することで手法全体の性能を向上できると考える．

本研究では，先行研究で提案された名前抽出（企業名抽出）における尤度比の推定方法に，新たに密度比推定に基づいた方法（尤度比推定）を適用することで尤度比直接推定を行う．この方法では尤度の推定を行わず，カーネルの計算に文字列のスムージングされた頻度を使用することで，名前抽出に適用可能にする．

さらに，先行研究で用いた尤度推定（Good-Turing）による尤度比の推定法と本稿で提案する尤度比推定の比較実験を行い，本稿で提案する尤度比の推定法の方が適合率及び再現率を有意に向上できることを示す．

## 2. 関連研究

未知語の抽出が可能な研究として，次のようなものがある．森ら[森 1998]は，N-gram 統計値を用いた単語の抽出と品詞の推定を同時に行う手法を提案している．この研究では形態素解析済みのコーパスに対し，名詞の前後の N-gram の分布を用いることで未知語を含む名詞の抽出を行っている．梅村[梅村 2000]は，出現頻度と出現集中を表す統計量を用いることで辞書を用いず文書中の特有の語を抽出する手法を提案している．この研究ではある文字列を含む文書の数を用いて文書中の特有の語を抽出している．以上の研究は未知語を抽出できるものであるが，特定の種類の名前の抽出は行っていない．

ナイーブベイズを基にした特定の種類の名前の抽出に関する研究として，名前の周辺と名前を構成する

文字列を特徴量としている次のような研究がある．菅野[菅野 2014]は， N-gram の統計値を用いて語の抽出を行う手法を提案している．この研究では，企業名を適用例として，企業名の直前直後の文字 N-gram を用いて抽出を行っている．また，企業名抽出においては企業名を構成する文字列自身の文字 N-gram の出現頻度も特徴量として用いることが有用であることを報告している．中野ら[中野 2015]は，菅野の手法を基にした新たな特徴量を提案している．この研究では企業名を構成する文字列をさらに前中語に分け，それぞれの文字 N-gram の出現頻度を特徴量として用いることが有用であることを報告している．また，中野ら[中野 2016]は，新たなスムージング法として信頼区間の下限値によるスムージングを提案している．これらのナイーブベイズを基にした先行研究で行われてきた手法は全て，スムージングによって尤度を推定して，その尤度から尤度比を計算する手法を行っている．一方，提案手法では尤度の推定を行わず，カーネルの計算に文字列のスムージングされた頻度を使用した尤度比推定を行うことで，尤度比を直接求める．

提案手法と中野ら[中野 2015]において，特徴とするもの，抽出の流れ，実験の行い方は同一であるが，尤度比の推定方法だけが異なる本研究では，尤度比の推定方法を取り換えることで適合率及び再現率が有意に向上することを示す．

## 3. 使用する概念

### 3.1. 概要

ここでは，本研究で使用している 6 つの概念，N-gram，分布仮説，評価文字列，尤度比，スムージング，学習と抽出について述べる．尤度比の推定方法を除く，これらの概念は中野ら[中野 2015]と同じものである．

### 3.2. N-gram

N-gram[長尾 1993]とは，文字，単語又は品詞などの連続した組み合わせである．単語を空白で区切る英語などの言語では単語単位で区切った N-gram（単語 N-gram）が使用される．しかし，日本語は空白で区切られていないため，直接単語 N-gram を用いることは出来ないという問題がある．この問題の解決として，文字単位で分割を行う方法[森 1998，浅原 2002]や形態素解析を利用して形態素単位で分割を行う方法がある．今回は文字単位で区切った N-gram（文字 N-gram）を用いる．また，菅野[菅野 2014]は企業名抽出に対して文字 N-gram の大きさ別の比較実験を行い，図 1 のような 2 文字区切りの N-gram（文字 Bigram）を用いた場合に最も適合率及び再現率が高かったことを報告している．中野ら[中野 2015]も文字 Bigram を用いている．このため，本研究でも同様に文字 N-gram の大きさとして文字 Bigram を使用して抽出を行う．


**図 1 文字 Bigram の例**

### 3.3. 分布仮説

Harris の分布仮説[Harris 1954]とは，「同じ文脈で使われる言葉は，類似する意味をもつ傾向がある」という仮説である．中野ら[中野 2015]と同様に，本研究でもこの分布仮説における文脈を企業名の直前及び直後の文字 Bigram と考える．

### 3.4. 評価文字列

菅野[菅野 2014]が企業名抽出に適用した分布仮説の考え方に加え，中野ら[中野 2015]は抽出したい企業名を構成する文字列自身の先頭及び末尾にも業種や地名などの特徴的な語が現れることに着目し，企業名を構成する文字列を前中後の 3 つに分割して，それらを加えた計 5 つに分類することが有用であると報告している．本研究でもこの 5 つの分類を用いる．これ以降，企業名の直前，企業名自身の前．企業名自身の中，企業名自身の後，企業名の直後という各部分を先行部，先頭部，中間部，末尾部，後続部と表し，これら全てをまとめた文字列を評価文字列と表す．各部の例を図 2 に示す．また，文書全体の評価文字列を各部ごとに集めた集合を評価文字列集合とする．評価文字列集合の例を図 3 に示す．

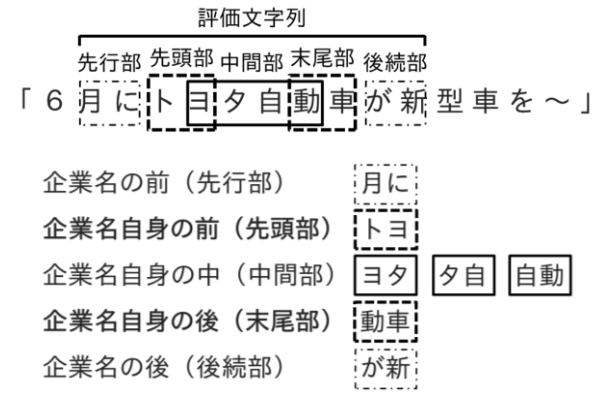


**図 2 評価文字列の各部の例**

| | 先行部 | 先頭部 | 中間部 | 末尾部 | 後続部 |
|---|---|---|---|---|---|
| 1 | 月に | トヨ | ヨタ,タ自,自動 | 動車 | が新 |
| 2 | たに | 日産 | 産自,自動 | 動車 | の社 |
| 3 | は、 | トヨ | ヨタ,タ自,自動 | 動車 | をは |
| 4 | 方、 | 日産 | 産自,自動 | 動車 | では |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| n | も、 | 三菱 | 菱自,自動 | 動車 | では |

**図 3 評価文字列集合の例**

### 3.5. 尤度比

文字列の企業名らしさを評価する値として尤度比を用いる．尤度比とは帰無仮説の尤度$L(H_0)$と対立仮説の尤度$L(H_1)$の比を取り，どちらが尤もらしいかを比較する指標である．

本研究では帰無仮説$H_0$を「与えられた文字 Bigram は文書中の任意の文字列から取り出したものである」(評価文字列集合の文字 Bigram ではない)，対立仮説$H_1$を「与えられた文字 Bigram が企業名を構成する文字列またはその直前直後の文字列から取り出したものである」(評価文字列集合の文字 Bigram である）とする．対立仮説$H_1$より帰無仮説$H_0$の方が尤もらしいとき（企業名の一部らしくない文字 Bigram のとき)には尤度比は小さくなり，帰無仮説$H_0$より対立仮説$H_1$の方が尤もらしいとき（企業名の一部らしい文字 Bigram のとき）には尤度比は大きくなる．どちらも同じぐらい尤もらしいとき（企業名の一部ともそうでないともいえない文字 Bigram のとき）には尤度比は 1 となる．

以上の尤度比を式として表すと式 1 のようになる．帰無仮説$H_0$に対する分母が「文書全体の文字 Bigram から求めた尤度」となり，対立仮説$H_1$に対する分子が「評価文字列が企業名であったときの文字 Bigram から求めた尤度」となる．評価値として必要なのは尤度比のみのため，今回提案する尤度比推定では，分母分子の尤度は計算せずに結果の文字 Bigram の尤度比のみを直接推定する．

$$\text{文字 Bigram の尤度比} = \frac{\text{評価文字列が企業名であったときの文字 Bigram から求めた尤度}}{\text{文書全体の文字 Bigram から求めた尤度}} \quad (1)$$

### 3.6. スムージング

ナイーブベイズ及び最尤推定に基づく手法では未知語が現れた場合，本来の確率が 0 でないにも関わらず全体の尤度の推定値が 0 になるという問題があるため，この問題を解決するためにスムージングを行う．スムージングは，本来の確率に近づけて抽出の適合率及び再現率を改善させるという点からも重要である．

ここでは，今回提案手法との比較対象として尤度推定に用いる，Good-Turing について説明する．

Good-Turing[Gale 1995]は頻度$r$の語の種類数$N_r$を用いて出現頻度に対して補正を行い，出現しなかった語の確率を推定する．また頻度が高い語の場合，$N_r$の値が不安定になるため，ジップの法則を用いることでさらに補正を行う．ジップの法則とは「頻度順位が n 位の単語は 1 位の単語の 1/n の確率で現れる」という法則である．この法則により，$\log(N_r)$及び$\log(r)$が線形の関係で表される．今回は式(2)に示す Gale ら[12]の方法に基づく，通常の Good-Turing と線形回帰を用いた Good-Turing を頻度が低いものと高いもので切り替える Good-Turing を用いて確率推定を行っている．

$$P_X(w_k^{k+2}) = \begin{cases} P_X'(w_k^{k+2}) & (\sigma \times 1.65 < |P_X'(w_k^{k+2}) - P_X''(w_k^{k+2})|) \\ P_X''(w_k^{k+2}) & (\sigma \times 1.65 \geqq |P_X'(w_k^{k+2}) - P_X''(w_k^{k+2})|) \\ \dfrac{N_1}{N_0 N} & (r = 0) \end{cases} \quad (2)$$

$$P_X'(w_k^{k+2}) = \frac{\{r(w_k^{k+2}) + 1\} \cdot \frac{N_{r+1}}{N_r}}{N}$$

$$P_X''(w_k^{k+2}) = \frac{r(w_k^{k+2}) \left\{1 + \frac{1}{r(w_k^{k+2})}\right\}^{b+1}}{N}$$

$$\sigma = \sqrt{\{r(w_k^{k+2}) + 1\}^2 \cdot \frac{N_{r+1}}{N_r}\left(1 + \frac{N_{r+1}}{N_r}\right)}$$

| | |
|---|---|
| $w_k^{k+2}$ | 評価文字列中の$k$番目の文字 Bigram |
| $X$ | 文字 Bigram 集合の一つ（先行部，先頭部，中間部，末尾部，後続部，文書全体の集合のどれか） |
| $P_X$ | 今回スムージング値として使用する，文字 Bigram 集合$X$内の Good-Turing の推定値 |
| $P_X'$ | 文字 Bigram 集合$X$内の通常の Good-Turing の推定値 |
| $P_X''$ | 文字 Bigram 集合$X$内の線形回帰を用いた Good-Turing の推定値 |
| $r$ | 文字 Bigram 集合$X$内の文字 Bigram $w_k^{k+2}$の頻度 |
| $N$ | 文字 Bigram 集合$X$内の文字 Bigram の総頻度 |
| $N_r$ | 文字 Bigram 集合$X$内の頻度$r$の文字 Bigram の種類数 |

### 3.7. 学習と抽出

企業名抽出は大きく分けて学習と抽出の 2 つのステップに別れる．この 2 つのステップの流れについて説明する．今回の提案手法である尤度比推定では章で説明する内容で尤度を計算せずに尤度比を求める点が，先行研究で用いられていた尤度から尤度比を計算する手法とは異なる．

まず，学習ステップの流れについて説明する．学習時には企業名の位置が分かっており直前直後の文字 Bigram を得ることのできる図 4 のような文書（学習用の文書）を用意する．学習用の文書から評価文字列の文字 Bigram を全て取り出して出現頻度の各部ごとに

集計する．尤度から尤度比を計算する手法では，出現頻度から尤度を計算し，図 5 のように尤度比の分子の値となる「評価文字列集合の文字 Bigram から求めた尤度」を求める．この尤度には 3.6 節のスムージングによってスムージングされた値を使用する．

次に抽出ステップの流れについて説明する．抽出時には，企業名を抽出したい文書（抽出用の文書）を用意する．抽出用の文書の企業名を抽出したい最小文字数から最大文字数の間の全部分文字列を企業名候補として，企業名候補に対する評価文字列の文字 Bigram を全て取り出して各部ごとに出現頻度を集計する．尤度から尤度比を間接的に計算する手法では，学習時と同様に出現頻度から尤度を計算し，尤度比の分母の値となる「文書全体の文字 Bigram から求めた尤度」を求める．この尤度も 3.6 節のスムージングによってスムージングされた値を使用する．

尤度から尤度比を計算する手法では，学習文書から求めた「評価文字列集合の文字 Bigram から求めた尤度」と抽出文書から求めた「文書全体の文字 Bigram から求めた尤度」より評価文字列に含まれる各部の文字 Bigram の尤度比が求められる．一方，今回の提案手法では，尤度比推定を用いて各部の文字 Bigram の尤度比を直接推定する．

評価値となる評価文字列の尤度比は，ナイーブベイズの考え方に基づき，条件付き独立性を仮定し，式 3 及び図 6 のように各部の文字 Bigram の尤度比の相乗平均をその値とする．平均を取っているのは文字列長による影響を正規化するためである．これにより，評価文字列の尤度比が大きな企業名候補は小さな企業名候補よりも企業名らしいと考えることが出来る．

$$LR(w_1^n) = \left( LR_{Pre} \times LR_{Head} \times \prod_{l=4}^{n-4} LR_{Mid} \times LR_{Tail} \times LR_{Post} \right)^{\frac{1}{n-3}} \quad (3)$$

$$LR_{Pre} = \frac{L^*_{Pre}(w_1^2)}{L^*_{Doc}(w_1^2)}$$

$$LR_{Head} = \frac{L^*_{Head}(w_3^4)}{L^*_{Doc}(w_3^4)}$$

$$LR_{Mid} = \frac{L^*_{Mid}(w_l^{l+1})}{L^*_{Doc}(w_l^{l+1})}$$

$$LR_{Tail} = \frac{L^*_{Tail}(w_{n-3}^{n-2})}{L^*_{Doc}(w_{n-3}^{n-2})}$$

$$LR_{Post} = \frac{L^*_{Post}(w_{n-1}^{n})}{L^*_{Doc}(w_{n-1}^{n})}$$

| | |
|---|---|
| $n$ | 評価文字列の文字数 |
| $w_i^j$ | 評価文字列中の$i$文字目からj文字目までの部分文字列 |
| $X$ | 文字 Bigram 集合の一つ（先行部 Pre，先頭部 Head，中間部 Mid，末尾部 Tail，後続部 Post，文書全体 Doc の集合のどれか） |
| $LR$ | 評価文字列の尤度比（＝評価値） |
| $LR_X$ | 文字 Bigram 集合$X$内の尤度比 |
| $L_X^*(w_k^{k+2})$ | 文字 Bigram 集合$X$内の文字 Bigram $w_k^{k+2}$の尤度のスムージング値 |

実際の抽出処理では，企業名を抽出したい最小文字数から最大文字数までの企業名候補に対する評価文字列の尤度比を計算し，この値が高い評価文字列の順（企業名らしい順）から一定数の企業名候補を企業名として抽出する．表 1 の例では上位 1 件のみが企業名であるが，実際の文書では多くの企業名が含まれるため上位一定数を抽出する．

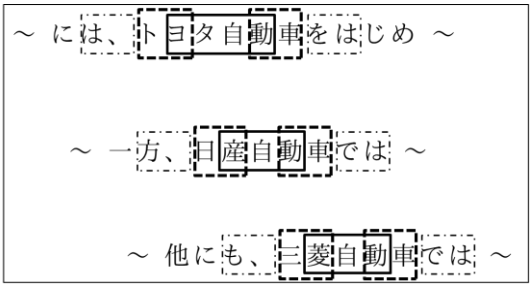


**図 4 複数の企業名を含む文書の例**

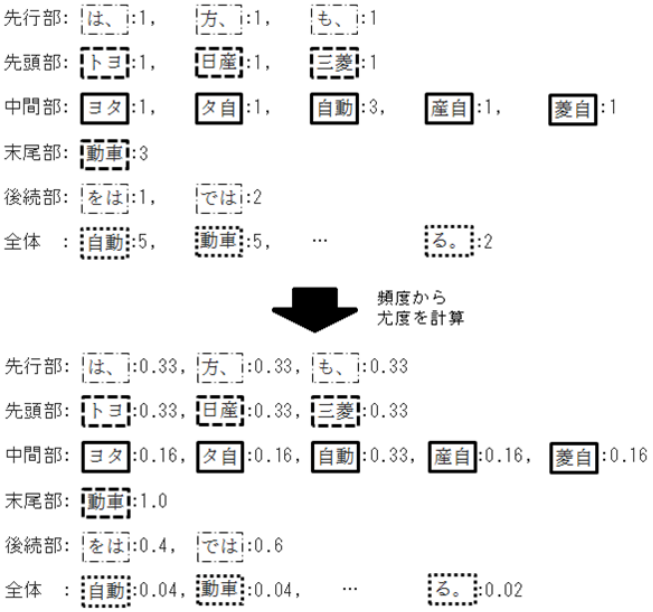


**図 5 各部の頻度の集計及び尤度の計算例**

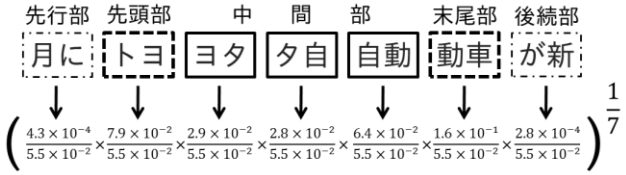


**図 6 評価値の計算例**

表 1 評価文字列と評価値の例

| 評価文字列 | 評価値(尤度比) |
|---|---|
| 月に　トヨタ自動車　が新 | 0.2447 |
| 月に　トヨタ自動車が　新型 | 0.0572 |
| にト　ヨタ自動車　が新 | 0.0510 |
| ６月　にトヨタ自動車　が新 | 0.0461 |
| 月に　トヨタ自動　車が | 0.0424 |
| にト　ヨタ自動車が　新型 | 0.0121 |
| にト　ヨタ自動車が新　型車 | 0.0082 |
| ６月　にトヨタ自動　車が | 0.0081 |
| トヨ　タ自動車が　新型 | 0.0065 |
| ・・・ | ・・・ |

## 4. 尤度比推定

### 4.1. 概要

従来の分母分子ごとに尤度を推定し，個々でスムージングを行う方法は，それぞれのスムージング値が本来の確率と異なっていた場合，尤度比の値に影響を与え，手法全体の性能を低下させてしまうという問題がある．問題の解決として確率密度推定を行わずに確率密度比を直接推定できる手法である密度比推定[杉山 2014]を尤度比の推定に適用する（これを尤度比推定と呼ぶ）．

前章までの方法は，スムージングされた尤度を用いて尤度比を推定していたが，尤度比推定ではカーネルの計算にスムージングされた頻度を使用することで，尤度を使用せずに尤度比を求めることができる．

ここでは，今回尤度比推定に適用している密度比推定の拘束無し最小二乗密度比推定法(uLSIF) [Kanamori 2009]及びその元となる手法である最小二乗密度比推定法(LSIF) [Kanamori 2009]，そしてuLSIFの企業名抽出への適用方法についての説明をする．

### 4.2. 最小二乗密度比推定法(LSIF)

LSIF[Kanamori 2009]は二乗損失を用いて確率密度比の最適化を行うことで直接推定を行う手法である．LSIFでは，確率密度比$\hat{\mathrm{r}}(\boldsymbol{x})$を式(4)の線形モデルでモデル化する．

$$\hat{\mathrm{r}}(\boldsymbol{x}) = \sum_{l=1}^{b} \alpha_l \varphi_l^{de}(\boldsymbol{x}) \tag{4}$$

| | |
|---|---|
| $\hat{\mathrm{r}}$ | 確率密度比 |
| $\alpha_l$ | データサンプルから学習されるパラメータ |
| $\varphi_l^{de}$ | 密度比の分母に対する非負の基底関数 |

定数項を無視した二乗誤差の期待値を標本平均で近似し，確率密度比が非負であることを考慮するとパラメータ$\{\alpha_l\}_{l=1}^{b}$は非負となるため，その制約を追加する．それに加えて，非負の正則化パラメータ$\lambda$を含む一次の正則化項$\lambda\sum_{l=1}^{b}\alpha_l$を導入すれば，パラメータ$\{\alpha_l\}_{l=1}^{b}$は式(5)の最適化問題で求められる．

$$\min_{\{\alpha_l\}_{l=1}^{b}} \left[\frac{1}{2}\sum_{l,l'=1}^{b}\alpha_l\alpha_{l'}\widehat{H}_{l,l'} - \sum_{l=1}^{b}\alpha_l\hat{h}_l + \lambda\sum_{l=1}^{b}\alpha_l\right] \tag{5}$$

$$subject\ to\ \alpha_1, \alpha_2, \ldots, \alpha_b \geq 0$$

$$\widehat{H}_{l,l'} = \frac{1}{n_{de}}\sum_{i=1}^{n_{de}}\varphi_l^{de}(x_i)\varphi_{l'}^{de}(x_i)$$

$$\hat{h}_l = \frac{1}{n_{nu}}\sum_{j=1}^{n_{nu}}\varphi_l^{nu}(x_j)$$

| | |
|---|---|
| $\alpha_l$ | データサンプルから学習されるパラメータ |
| $\varphi_l^{de}$ | 密度比の分母に対する非負の基底関数 |
| $\varphi_l^{nu}$ | 密度比の分子に対する非負の基底関数 |
| $\lambda$ | 非負の正則化パラメータ |
| $n_{de}$ | 密度比の分母に対するデータのサンプル数 |
| $n_{nu}$ | 密度比の分子に対するデータのサンプル数 |

### 4.3. 拘束無し最小二乗密度比推定法(uLSIF)

uLSIF[Kanamori 2009]は式(5)のLSIFよりも収束性や正則化パス追跡アルゴリズムの数値的な安定性を向上させた方法であり，解析的に解を求めることができる．この式はLSIFの最適化問題の式の非負拘束を無視し，正規化項を二次に変更した式(6)の拘束無し最適化問題となる．

$$\min_{\{\alpha_l\}_{l=1}^{b}} \left[\frac{1}{2}\sum_{l,l'=1}^{b}\alpha_l\alpha_{l'}\widehat{H}_{l,l'} - \sum_{l=1}^{b}\alpha_l\hat{h}_l + \lambda\sum_{l=1}^{b}\alpha_l^{\,2}\right] \tag{6}$$

また，先に紹介したように解は式(7)で解析的に求めることができる．

$$\tilde{\boldsymbol{\alpha}}(\lambda) = \left(\widehat{\boldsymbol{H}} + \lambda\boldsymbol{I}_b\right)^{-1}\hat{\boldsymbol{h}} \tag{7}$$

| | |
|---|---|
| $\boldsymbol{I}_b$ | $b$次元の単位行列 |
| $\widehat{\boldsymbol{H}}$ | $l$行$l'$列目の要素が$\widehat{H}_{l,l'}$となるような$b$行$b$列の行列 |
| $\hat{\mathbf{h}}$ | $l$番目の要素が$\hat{h}_l$となるような$b$次元のベクトル |

uLSIFでは，LSIFの非負拘束を無視したため，パラメータ$\{\alpha_l\}_{l=1}^{b}$が負になる可能性がある．このため負のパラメータは0にするという式(8)の事後補正を行う．

$$\hat{\boldsymbol{\alpha}}(\lambda) = \max\left(0, \tilde{\boldsymbol{\alpha}}(\lambda)\right) \tag{8}$$

uLSIFの正則化パラメータ$\lambda$は，正則化パス追跡アルゴリズムによっても求めることができるが，今回パラメータ$\lambda$は，パラメータ決定用のコーパス（テストコーパスよりも小さなコーパス）を用いて抽出を行い，交差検証で適合率及び再現率の良いパラメータλを各部ごとに決定して$\lambda_x$とした．

### 4.4. 企業名抽出への適用

今回の企業名抽出のタスクに適用するにあたって標本$\boldsymbol{x}$として，uLSIF の原著論文の様な連続的な値ではなく文字 Bigram から得られる離散的な値を用いる．そのため，基底関数に uLSIF の原著論文で使われているガウスカーネルを使用することは出来ない．

そのため今回は基底関数として，式(9.1)及び式(9.2)のような対角成分以外を 0 とする近似を用いる基底関数$\varphi_l^{Doc}$及び$\varphi_l^X$を使用する．

$$\varphi_l^{Doc}(x) = \begin{cases} 0 & (x \neq w_l) \\ \dfrac{K_{Doc}(x,x)}{L_{Doc}} & (x = w_l) \end{cases} \tag{9.1}$$

$$\varphi_l^X(x) = \begin{cases} 0 & (x \neq w_l) \\ \dfrac{K_X(x,x)}{L_X} & (x = w_l) \end{cases} \tag{10.2}$$

| | |
|---|---|
| $l$ | 文字 Bigram の種類の index |
| $w_l$ | 文字 Bigram の種類の集合の中の$l$番目の文字 Bigram |
| $L_{Doc}$ | 文書全体におけるカーネル値の総和 |
| $L_X$ | 文字 Bigram 集合$X$におけるカーネル値の総和 |

式(9.1)および式(9.2)の$K_{Doc}$及び$K_X$として，文字列カーネルで 2 つの語の頻度の乗算を用いることに基づき，式(10.1)及び式(10.2)のようなカーネルを使用する．また，カーネルに使用する頻度のスムージング方法としては，3.6 節でのべた Good-Turing を使用する．

$$K_{Doc}\left(w_i^{i+1}, w_j^{j+1}\right) = \mathrm{r}^*_{Doc}\left(w_i^{i+1}\right)\mathrm{r}^*_{Doc}\left(w_j^{j+1}\right) \tag{10.1}$$

$$K_X\left(w_i^{i+1}, w_j^{j+1}\right) = \mathrm{r}^*_X\left(w_i^{i+1}\right)\mathrm{r}^*_X\left(w_j^{j+2}\right) \tag{10.2}$$

| | |
|---|---|
| $w_i^{i+1},\ w_j^{j+1}$ | 評価文字列中の$i$番目及び$j$番目の文字 Bigram |
| $\mathrm{r}^*_{Doc}$ | 文書全体における文字 Bigram のスムージングされた頻度 |
| $\mathrm{r}^*_X$ | 文字 Bigram 集合$X$における文字 Bigram のスムージングされた頻度 |

企業名抽出に適用した場合には LSIF 及び uLSIF における確率密度比r̂の式(4)は，文字 Bigram 集合 X の確率密度比r̂$_X$の式(11)のようになる．文書全体に対する添字 Doc は式(5)の添字 de に，文字 Bigram 集合に対する添字$X$は式(5)の添字 nu にそれぞれ対応する．

$$\hat{\mathrm{r}}_X(\boldsymbol{x}) = \sum_{l=1}^{v} \alpha_l \varphi_l(\boldsymbol{x}) = LR_X(\boldsymbol{x}) \tag{11}$$

| | |
|---|---|
| $x$ | 任意の文字 Bigram |
| r̂$_X$ | 文字 Bigram 集合$X$の確率密度比<br>→ 文字 Bigram 集合$X$の尤度比$LR_X$ |
| $v$ | 文字 Bigram の全種類数 |
| $l$ | 文字 Bigram の種類の index |
| $\alpha_l$ | 文字 Bigram から学習されるパラメータ |
| $\varphi_l^{Doc}$ | 文書全体における基底関数 |

ここでは，評価文字列中の i 番目の文字 Bigram $w_i^{i+1}$に対する尤度比$LR_X\left(w_i^{i+1}\right)$を推定することを考える．このとき$w_i^{i+1}$は，文字 Bigram の種類の集合において k 番目に存在する文字 Bigram $w_k$と同じ文字 Bigram であるとする．文字 Bigram 集合$X$は，評価文字列集合の各部（先行部，先頭部，中間部，末尾部，後続部）に対応し，実際の抽出処理では各部ごとに尤度比を計算する．

式(5)の$\hat{H}_{l,l'}$は式(12.1)，$\hat{h}_l$は式(13.1)となる．また，分母分子の全文字 Bigram に対して尤度比を計算するため，$n_{de}$及び$n_{nu}$として文書全体と文字 Bigram 集合$X$を合わせた文字 Bigram 集合における文字 Bigram の種類数$t_{X+Doc}$を用いる．

$$\hat{H}_{l,l'} = \frac{1}{t_{X+Doc}} \sum_{m=1}^{t_{X+Doc}} \varphi_l^{Doc}(w_m) \cdot \varphi_{l'}^{Doc}(w_m) \tag{12.1}$$

$$\hat{h}_l = \frac{1}{t_{X+Doc}} \sum_{n=1}^{t_{X+Doc}} \varphi_l^X(w_n) \tag{13.1}$$

式(9.1)より$\varphi_l^{Doc}(w_m)$は$w_m = w_l$以外のときは 0 となり，$\varphi_{l'}^{Doc}(w_m)$は$w_m = w_{l'}$以外のときは 0 となる．$\varphi_l^{Doc}(w_m)$と$\varphi_{l'}^{Doc}(w_m)$のどちらも 0 にならないときは$w_m = w_l = w_{l'}$の場合のみであるため，総和の計算は$\mathrm{m} = l = l'$の部分のみが残り，式(12.2)となる．

$$\hat{H}_{l,l'} = \begin{cases} 0 & (l \neq l') \\ \dfrac{1}{t_{X+Doc}} \varphi_l^{Doc}(w_l)\varphi_{l'}^{Doc}(w_{l'}) & (l = l') \end{cases} \tag{12.2}$$

$\varphi_l^{nu}(w_n)$も$w_n = w_l$以外のときは 0 となるため，総和の計算は$n = l$の部分のみが残り，式(13.2)となる．

$$\hat{h}_l = \frac{1}{t_{X+Doc}} \varphi_l^X(w_l) \tag{13.2}$$

パラメータのベクトル$\tilde{\boldsymbol{\alpha}}(\lambda_X)$は以下の式(14)のように表される．

$$\tilde{\boldsymbol{\alpha}}(\lambda_X) = \left(\hat{\boldsymbol{H}} + \lambda_X \boldsymbol{I}_{t_{X+Doc}}\right)^{-1}\hat{\boldsymbol{h}} \tag{14}$$

式(12.2)より$\hat{\boldsymbol{H}}$の対角成分以外は 0 となる．また，$\lambda_X \boldsymbol{I}_{t_{X+Doc}}$も対角成分以外は 0 であることより，$\hat{\boldsymbol{H}} + \lambda_X \boldsymbol{I}_{t_{X+Doc}}$は対角行列となる．対角行列の逆行列は対角成分のみを逆数にした行列で計算できる．また，求めた逆行列と$\hat{\boldsymbol{h}}$を乗算した際も対角成分との乗算以外の項は 0 となる．そのため，パラメータのベクトル$\tilde{\boldsymbol{\alpha}}(\lambda_X)$の$l$番目の要素$\tilde{\alpha}_l(\lambda_X)$は以下の式(15)のように表される．

$$\tilde{\alpha}_l(\lambda_X) = \left(\hat{H}_{l,l} + \lambda_X\right)^{-1}\hat{h}_l \tag{15}$$

$$= \frac{\dfrac{1}{t_{X+Doc}}\varphi_l^X(w_l)}{\dfrac{1}{t_{X+Doc}}\{\varphi_l^{Doc}(w_l)\}^2 + \lambda_X}$$

この式は 4.3.節の uLSIF ではパラメータ$\tilde{\alpha}_l$は負になる可能性があるため，式(16)で事後補正を行う必要があるが，今回の場合は逆行列の対角部分しか使用しておらず，その他の部分も正値のため，この事後補正がない場合も等価である．

$$\hat{\alpha}_l(\lambda_X) = max\left(0, \tilde{\alpha}_l(\lambda_X)\right) \tag{16}$$

式(4)より，尤度比$LR_X$（確率密度比$\hat{r}_X$）は以下の式(17.1)で求められる．

$$LR_X(w_k) = \hat{r}_X(w_k) = \sum_{l=1}^{t_{X+Doc}} \hat{\alpha}_l(\lambda_X)\varphi_l^{Doc}(w_k) \tag{17.1}$$

$\varphi_l^{Doc}(w_k)$は$w_k = w_l$以外のときは 0 となることより，総和の計算は$l = k$の部分のみが残り，以下の式(17.2)となる．

$$LR_X(w_k) = \hat{r}_X(w_k) = \hat{\alpha}_k(\lambda_X)\varphi_k^{Doc}(w_k) \tag{17.2}$$

$w_k$と$w_i^{i+1}$が同じ文字 Bigram であることより，以下の式(17.3)のように表される．この尤度比を評価値の計算に用いる．

$$LR_X\left(w_i^{i+1}\right) = LR_X(w_k) \tag{17.3}$$
$$= \hat{r}_X\left(w_i^{i+1}\right) = \hat{\alpha}_l(\lambda_X)\varphi_l^{Doc}\left(w_i^{i+1}\right)$$

## 5. 比較実験

### 5.1. 概要

ここでは，尤度比の推定方法を変更した影響を確認するため，企業名抽出を対象として，スムージングされた尤度から尤度比を計算する方法（ベースライン）と，尤度比推定（提案手法）の比較実験を行う．

### 5.2. 実験条件

実験の各条件は表 2 に示す．パラメータ決定用文書，スムージング法以外は，中野ら[中野 2015]と同様の条件である．ベースラインにおいて最も適合率及び再現率の高かった条件を使用する．文書は毎日新聞コーパス 91-97 年[毎日新聞社 1991-1997]の年始から 2 万記事を 1 万記事ごとに分割したものを 1 つの文書として計 14 文書を作成する．また，K-分割交差検証で 14 文書中の 13 文書を学習用，残りの 1 文書を抽出用とする．パラメータ決定用文書は，学習用文書中の各文書の先頭から 1 千記事（年始から 1-1000 記事，10001-11000 記事）を使用する．既知の企業名は抽出用文書から形態素解析で組織名を抽出後，パターンマッチにより企業名以外を除去したものを用いる．5 文字から 30 文字までの企業名を対象に評価値の計算を行い，評価値の高い順に上位 2000 件を企業名として抽出する． スムージング法は，ベースラインにおける尤度のスムージング，提案手法における頻度のスムージングのどちらも Good-Turing 法を用いる．

**表 2 実験条件**

| | |
|---|---|
| **使用文書** | 毎日新聞コーパス 91-97 年の年始から 2 万記事（1 万記事ごとに分割）の計 14 文書 |
| **抽出用文書** | 使用文書中の 1 文書 |
| **学習用文書** | 使用文書中から抽出用の 1 文書を除いた 13 文書 |
| **パラメータ決定用文書** | 学習用文書中の各文書の先頭から 1 千記事 |
| **既知の企業名リストの作成方法** | 形態素解析で組織名を抽出後，パターンマッチにより企業名以外を除去 |
| ***N*-gram** | 文字 Bigram |
| **企業名抽出の文字数の範囲** | 5 - 30 [文字] |
| **抽出件数** | 評価値の上位 2000 [件] |
| **スムージング法** | Good-Turing（尤度，頻度） |

### 5.3. 評価方法

正解の評価方法は中野ら[中野 2015]と同じく部分適合率及び部分再現率を使用している．

人が企業名だと認識できる全ての文字列の集合を全体正解集合とする．この以外に企業名の正解は無いものとする．抽出の正誤の判定には本来ならば全体正解集合を用いるべきであるが，全体正解集合は実際には得られない，または得るために大きなコストがかかる．そのため，既知の企業名の集合を全体正解集合に含まれる部分正解集合として，この部分正解集合のみを用いて正誤の判定を行う．

ここで部分正解集合から得られる適合率及び再現率を全体正解集合から得られるものと区別して，部分適合率及び部分正解率と表現する．本研究では評価値として部分適合率及び部分正解率の両方を用いる．それぞれ，式(18.1)及び式(18.2)となる．

また，中野ら[中野 2015]は「正解データに一度でも現れた企業名と名前が同じ企業名候補」を正解としていたが，今回はより正確な「正解データと同じ位置で現れた企業名と名前が同じ企業名候補」を正解とする．これにより，部分適合率及び部分正解率が変化している．

$$\text{部分適合率} = \frac{\text{部分正解に含まれる抽出文字列の数}}{\text{抽出文字列の数}} \tag{18.1}$$

$$\text{部分再現率} = \frac{\text{部分正解に含まれる抽出文字列の数}}{\text{文書に存在する部分正解に含まれる企業名の数}} \tag{18.2}$$

### 5.4. 実験結果

表 3 に部分適合率及び部分再現率を示す．また，各項目で上回っている手法を下線で示す．部分適合率と部分再現率のいずれも'95(1)以外の全ての対象文書においてベースラインが提案手法を上回っており，符号検定を行った結果，有意水準 1%で提案手法とベースラインとの有意差が認められた．

表 3 部分適合率及び部分再現率

| | 部分適合率 | | 部分再現率 | |
|---|---|---|---|---|
| | 提案手法 | ベースライン | 提案手法 | ベースライン |
| '91(1) | **0.5710** | 0.5700 | **0.5818** | 0.5807 |
| '91(2) | **0.5825** | 0.5760 | **0.5700** | 0.5636 |
| '92(1) | **0.6260** | 0.6200 | **0.5968** | 0.5910 |
| '92(2) | **0.5765** | 0.5760 | **0.5462** | 0.5457 |
| '93(1) | **0.7400** | 0.7330 | **0.6379** | 0.6319 |
| '93(2) | **0.8375** | 0.8275 | **0.6323** | 0.6248 |
| '94(1) | **0.7760** | 0.7695 | **0.6358** | 0.6305 |
| '94(2) | **0.7960** | 0.7830 | **0.5994** | 0.5896 |
| '95(1) | 0.7930 | **0.7940** | 0.6261 | **0.6269** |
| '95(2) | **0.7305** | 0.7270 | **0.6347** | 0.6316 |
| '96(1) | **0.7890** | 0.7825 | **0.5648** | 0.5601 |
| '96(2) | **0.8105** | 0.7980 | **0.5793** | 0.5704 |
| '97(1) | **0.8050** | 0.7950 | **0.5365** | 0.5298 |
| '97(2) | **0.8200** | 0.8185 | **0.5361** | 0.5351 |
| 平均 | 0.7324 | 0.7264 | 0.5913 | 0.5866 |
| 分散 | 0.0091 | 0.0087 | 0.0013 | 0.0013 |

## 5.5. 抽出語の傾向の分析及び考察

抽出結果から，抽出語の傾向の分析を行った．ベースラインで抽出できなかったが提案手法で抽出できた例，及びベースラインで抽出できたが提案手法で抽出できなかった例を表 4 に示す．

提案手法のみに現れた例ではベースラインのみに現れた例に比べて，企業名に「**"ワシントン・ポスト"**」や「**"セゾングループ"**」などの片仮名を含む企業名が多く含まれている．これは，Good-Turing が用いているジップの法則が片仮名や平仮名といった異なる単語同士でも同じ文字 Bigram が現れやすい語に上手く適用できなかったためと考える．文字 Bigram に Good-Turing を適用する場合は，単語の頻度が文字 Bigram の頻度とおおよそ等しくなることを仮定しているのに対して,片仮名や平仮名の文字 Bigram は異なる単語同士でも同じ文字 Bigram が現れることが多いためジップの法則に従わない分布になると考えられる．このため，Good-Turing によるスムージングだけでなく尤度比推定も用いて尤度の補正を行った提案手法の方が有効に働いたと考える．

表 4 片方のみに現れた企業名（重複なし上位 10 件）

| ベースラインで抽出できなかったが，提案手法で抽出できた企業名 | ベースラインで抽出できたが，提案手法で抽出できなかった企業名 |
|---|---|
| ワシントン・ポスト | 卯辰山文庫 |
| 日本航空電子工業 | ほるぷ出版 |
| セゾングループ | 雅叙園観光 |
| インタファクス | ヱスビー食品 |
| ニューズウィーク | タイガー魔法瓶 |
| 電気化学工業 | 実業之日本社 |
| ノースウエスト航空 | 阪急交通社 |
| ポニーキャニオン | 音楽之友社 |
| アカデミー出版 | 森精機製作所 |
| 東京テレメッセージ | マツモト電器 |
| ⋮ | ⋮ |

## 6. おわりに

本稿では，スムージングによって尤度を推定してから尤度比を計算する方法の代わりに，尤度を推定せずに文字列カーネルを用いることで尤度比を求める尤度比推定の名前抽出（企業名抽出）への適用方法の提案を行った．そして，新聞記事を対象とした提案手法とベースラインの比較実験を行い，部分適合率及び部分再現率が向上できることを明らかにした．


## 謝辞

本研究は，住友電工情報システム株式会社との共同研究の成果です．ここに感謝の意を表します．


## 参 考 文 献


[森 1998] 森 信介，長尾 眞．n グラム統計によるコーパスからの未知語抽出．情報処理学会論文誌．1998, 39(7), p. 2093-2100.

[梅村 2000] 梅村 恭司．未踏テキスト情報中のキーワードの抽出システム開発．未踏ソフトウェア創造事業, 2000.

[菅野 2014] 菅野 弘太．n-gram の統計値による企業名の抽出．豊橋技術科学大学，2014，43p．修士論文.

[中野 2015] 中野 翔平ほか．企業名抽出のための特徴量の検討，第7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 2015), E8-5, 2015.

[中野 2016] 中野 翔平ほか．信頼区間の下限値による確率推定を用いた企業名抽出，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 2016), E8-1, 2016.

[長尾 1993] 長尾 眞，森 信介．大規模日本語テキストのnグラム統計の作り方と語句の自動抽出．情報処理学会研究報告. 1993, 93(61), p. 1-8.

[浅原 2002] 浅原 正幸,松本 裕治．日本語固有表現抽出におけるわかち書き問題の解決．情報処理学会論文誌. 2002, 45(5), p.1442-1450.

[Harris 1954] Zellig S. Harris. Distributional structure. Word. 1954, 10(23), p. 146-162.

[北 1999] 北 研二．確率的言語モデル．東京大学出版会, 1999, 239p.

[Gale 1995] W. A. Gale, G. Sampson. Good-Turing Frequency Estimation without Tears. Journal of Quantitative Linguistics. 1995, 2(3), p.217-237.

[杉山 2014] 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. 2014, 97(5), p.353-358.

[Kanamori 2009] Takafumi Kanamori, Shohei Hido, Masashi Sugiyama. A Least-squares Approach to Direct Importance Estimation. The Journal of Machine Learning Research. 2009, 10, p.1391-1445.

[毎日新聞社 1991-1997] 毎日新聞社．CD-毎日新聞データ集'91-97 年版．日外アソシエーツ，1991-1997. (CD-ROM).