共変量シフト下の学習による語義曖昧性解消の教師なし領域適応

新納　浩幸[†]・佐々木　稔[†]

本論文では語義曖昧性解消 (Word Sense Disambiguation，WSD) の教師なし領域適応の問題に対して，共変量シフト下の学習を試みる．共変量シフト下の学習では確率密度比 $w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ を重みとした重み付き学習を行うが，WSD の場合，推定される確率密度比の値が小さくなる傾向がある．ここでは $P_T(\boldsymbol{x})$ と $P_S(\boldsymbol{x})$ をそれぞれ求めて，その比を取ることで $w(\boldsymbol{x})$ を推定するが，$P_S(\boldsymbol{x})$ を求める際に，ターゲット領域のコーパスとソース領域のコーパスを合わせたコーパスを，新たにソース領域のコーパス S と見なすことで，先の問題に対処する．BCCWJ の 3 つの領域 OC （Yahoo! 知恵袋），PB（書籍）及び PN（新聞）を選び，SemEval-2 の日本語 WSD タスクのデータを利用して，多義語 16 種類を対象に，WSD の領域適応の実験を行った．$w(\boldsymbol{x})$ を推定する手法として，$P_T(\boldsymbol{x})$ と $P_S(\boldsymbol{x})$ を求めずに，$w(\boldsymbol{x})$ を直接推定する uLSIF も試みた．また確率密度比を上方修正するために「p 乗する」「相対確率密度比を取る」という手法も組み合わせて試みた．それらの実験の結果，提案手法の有効性が示された．

キーワード：語義曖昧性解消，領域適応，共変量シフト，uLSIF，負の転移

Unsupervised Domain Adaptations for Word Sense Disambiguation by Learning under Covariate Shift

Hiroyuki Shinnou[†] and Minoru Sasaki[†]

In this paper, we apply the learning under covariate shift to the problem of unsupervised domain adaptation for word sense disambiguation (WSD). This learning is a type of weighted learning method, in which the probability density ratio $w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ is used as the weight of an instance. However, $w(\boldsymbol{x})$ tends to be small in WSD tasks. In order to address this problem, we calculate $w(\boldsymbol{x})$ by estimating $P_T(\boldsymbol{x})$ and $P_S(\boldsymbol{x})$, where $P_S(\boldsymbol{x})$ is estimating by regarding the corpus combining the source domain corpus and target domain corpus as the source domain corpus. In the experiment, we use three domains -OC (Yahoo! Chiebukuro), PB (books) and PN (news papers)- in BCCWJ, and 16 target words provided by the Japanese WSD task in SemEval-2. For calculating $w(\boldsymbol{x})$, we also use uLSIF, which directly estimates $w(\boldsymbol{x})$ without estimating $P_T(\boldsymbol{x})$ or $P_S(\boldsymbol{x})$. Moreover, we use the "p power" method and the "relative probability density ratio" method to boost the obtained probability density ratio. These experiments prove our method to be effective.

[†] 茨城大学工学部情報工学科, Department of Computer and Information Sciences, Ibaraki University

Key Words: *word sense disambiguation, domain adaptation, covariate shift, uLSIF, negative transfer*

1 はじめに

本論文では，語義曖昧性解消 (Word Sense Disambiguation, WSD) の領域適応に対して，共変量シフト下の学習を試みる．共変量シフト下の学習では確率密度比を重みとした重み付き学習を行うが，WSD のタスクでは算出される確率密度比が小さくなる傾向がある．ここではソース領域のコーパスとターゲット領域のコーパスとを合わせたコーパスをソース領域のコーパスと見なすことで，この問題に対処する．なお本手法はターゲット領域のデータにラベル付けしないため，教師なし領域適応手法に分類される．

WSD は文中の多義語の語義を識別するタスクである．通常，あるコーパス S から対象単語の用例を取り出し，その用例中の対象単語の語義を付与した訓練データを作成し，そこから SVM 等の分類器を学習することで WSD を解決する．ここで学習した分類器を適用する用例がコーパス S とは異なるコーパス T 内のものである場合，学習した分類器の精度が悪い場合がある．これが領域適応の問題であり，自然言語処理では WSD 以外にも様々なタスクで問題となるため，近年，活発に研究されている (Sogaard 2013; 森 2012; 神嶌 2010).

今，対象単語 w の用例を $\boldsymbol{x}$，w の語義の集合を C とする．$\boldsymbol{x}$ 内の w の語義が $c \in C$ である確率を $P(c|\boldsymbol{x})$ とおくと，WSD は $\arg\max_{c \in C} P(c|\boldsymbol{x})$ を求めることで解決できる．領域適応では，コーパス S（ソース領域）から得られた訓練データを用いて，$P(c|\boldsymbol{x})$ を推定するので，得られるのは S 上の条件付き分布 $P_S(c|\boldsymbol{x})$ であるが，識別の対象はコーパス T（ターゲット領域）内のデータであるため必要とされるのは T 上の条件付き分布 $P_T(c|\boldsymbol{x})$ である．このため領域適応の問題は $P_S(c|\boldsymbol{x}) \neq P_T(c|\boldsymbol{x})$ から生じているように見えるが，用例 $\boldsymbol{x}$ がどのような領域で現れたとしても，その用例 $\boldsymbol{x}$ 内の対象単語 w の語義が変化するとは考えづらい．このため $P_S(c|\boldsymbol{x}) = P_T(c|\boldsymbol{x})$ と考えられる．$P_S(c|\boldsymbol{x}) = P_T(c|\boldsymbol{x})$ が成立しているなら，$P_T(c|\boldsymbol{x})$ の代わりに $P_S(c|\boldsymbol{x})$ を用いて識別すればよいと思われるが，この場合，識別の精度が悪いことが多い．これは $P_S(\boldsymbol{x}) \neq P_T(\boldsymbol{x})$ から生じている．$P_S(c|\boldsymbol{x}) = P_T(c|\boldsymbol{x})$ かつ $P_S(\boldsymbol{x}) \neq P_T(\boldsymbol{x})$ という仮定は共変量シフトと呼ばれる (Sugiyama and Kawanabe 2011)．自然言語処理の多くの領域適応のタスクは共変量シフトが成立していると考えられる (Sogaard 2013).

ソース領域のコーパス S から得られる訓練データを $D = \{(\boldsymbol{x}_i, c_i)\}_{i=1}^N$ とおく．一般に共変量シフト下の学習では確率密度比 $w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ を重みとした以下の重み付き対数尤度を最大にするパラメータ $\boldsymbol{\theta}$ を求めることで，$P_T(c|\boldsymbol{x})$ を構築する．

$$\sum_{i=1}^{N} w(\boldsymbol{x}_i) \log P_T(c_i|\boldsymbol{x}_i; \boldsymbol{\theta})$$

共変量シフト下の学習の要は確率密度比 $w(\boldsymbol{x})$ の算出であるが，その方法は大きく 2 つに分類できる．1 つは $P_T(\boldsymbol{x})$ と $P_S(\boldsymbol{x})$ をそれぞれ求め，その比を求めることで $w(\boldsymbol{x})$ を求める方法である．もう 1 つは $w(\boldsymbol{x})$ を直接モデル化する方法である (杉山 2010)．ただしどちらの方法をとっても，WSD の領域適応に対しては，求められる値が低くなる傾向がある．この問題に対しては，確率密度比を p 乗 $(0 < p < 1)$ したり (杉山 2006)，相対確率密度比 (Yamada, Suzuki, Kanamori, Hachiya, and Sugiyama 2011) を使うなど，求めた確率密度比を上方に修正する手法が存在する[1]．本論文では $P_T(\boldsymbol{x})$ と $P_S(\boldsymbol{x})$ をそれぞれ求める手法を用いる際に，ターゲット領域のコーパスとソース領域のコーパスを合わせたコーパスを，新たにソース領域のコーパス S と見なして確率密度比を求めることを提案する．提案手法は必ずしも確率密度比を上方に修正する訳ではないが，多くの場合，この処理により $P_S(\boldsymbol{x})$ の値が減少し，結果的に $w(\boldsymbol{x})$ の値が増加する．

なお，本論文で利用する手法は，ターゲット領域のラベル付きデータを利用しないために，教師なし領域適応手法に属する．当然，ターゲット領域のラベル付きデータを利用する教師付き領域適応手法を用いる方が，WSD の識別精度は高くなる．しかし本論文では教師なし領域適応手法を扱う．理由は 3 つある．1 つ目は，教師なし領域適応手法はラベル付けするコストがないという大きな長所があるからである．2 つ目は，共変量シフト下の学習はターゲット領域のラベル付きデータを利用しない設定になっているからである．3 つ目は， WSD の領域適応の場合，対象単語毎に領域間距離が異なり，コーパスの領域が異なっていても，領域適応の問題が生じていないケースも多いからである．領域適応の問題が生じている，いないの問題を考察していくには，ターゲット領域のラベル付きデータを利用しない教師なし領域適応手法の方が適している．

実験では現代日本語書き言葉均衡コーパス (Balanced Corpus of Contemporary Written Japanese, BCCWJ (Maekawa 2007)) における 3 つの領域 OC（Yahoo! 知恵袋），PB（書籍）及び PN（新聞）を利用する．SemEval-2 の日本語 WSD タスク (Okumura, Shirai, Komiya, and Yokono 2010) ではこれらのコーパスの一部に語義タグを付けたデータを公開しており，そのデータを利用する．すべての領域である程度の頻度が存在する多義語 16 単語を対象にして，WSD の領域適応の実験を行う．領域適応としては OC → PB，PB → PN，PN → OC，OC → PN，PN → PB，PB → OC の計 6 通りが存在する．結果 $16 \times 6 = 96$ 通りの WSD の領域適応の問題に対して実験を行った．その結果，提案手法による重み付けの効果を確認できた．

また，従来手法はベースラインよりも低い値となったが，これは多くの WSD の教師なし領域適応では負の転移が生じていない，言い換えれば実際には領域適応の問題になっていないこ

[1] これらの手法は正確には確率密度比を 1 に近づける手法であるが，多くの場合，確率密度比は 1 以下の値であるため，ここではこれらの手法も確率密度比を上方に修正する手法と呼ぶことにする．

とから生じていると考えられる．考察では負の転移と重み付けとの関連，また負の転移と関連の深い Misleading データの存在と重み付けとの関連を中心に議論した．

2 関連研究

自然言語処理における領域適応は，帰納学習手法を利用する全てのタスクで生じる問題であるために，その研究は多岐にわたる．利用手法をおおまかに分類すると，ターゲット領域のラベル付きデータを利用するかしないかで分類できる．利用する場合を教師付き領域適応手法，利用しない場合を教師なし領域適応手法と呼ぶ．提案手法は教師なし領域適応手法の範疇に入るので，ここでは教師なし領域適応手法を中心に関連研究を述べる．

領域適応の問題は，一般の教師付き学習手法における訓練事例のスパース性の問題だと捉えることもできる．そのためターゲット領域のデータにラベルを付与しないという条件では，半教師付き学習 (Chapelle, Schölkopf, and Zien 2006) が教師なし領域適応手法として使えることは明らかである．ただし半教師付き学習では大量のラベルなしデータを必要とする．半教師付き学習を WSD に利用する場合，対象単語毎に用例を集める必要があり，しかもターゲット領域のコーパスは新規であることが多いため，対象単語毎の用例を大量に集めることは困難である．このため WSD の領域適応の場合，半教師付き学習を利用しようとすれば，Transductive 学習 (Joachims 1999) に近い形となるが，ソース領域とターゲット領域が異なる領域適応の形に Transductive 学習が利用できるかどうかは明らかではない．

WSD の領域適応をタスクとした教師なし領域適応の研究としては，論文 (新納，佐々木 2013) の研究がある．そこでの基本的なアイデアは WSD で使うシソーラスをターゲット領域のコーパスから構築することであるが，WSD で使うシソーラスが分野依存になっているかどうかは明らかではない (新納，國井，佐々木 2014) [2]．また Chan はターゲット領域上の語義分布を EM アルゴリズムで推定している (Chan and Ng 2005, 2006)．これも教師なし領域適応手法であるが，本論文で扱う領域適応では語義分布の違いは顕著ではなく，効果が期待できない．

本論文は，WSD の領域適応では共変量シフトの仮定が成立していると考え，共変量シフト下の学習を利用する．共変量シフト下の学習を領域適応に応用した研究としては Jiang の研究 (Jiang and Zhai 2007) と齋木の研究 (齋木，高村，奥村 2008) がある．Jiang は確率密度比を手動で調整し，モデルにはロジステック回帰を用いている．また齋木は $P_S(\boldsymbol{x})$ と $P_T(\boldsymbol{x})$ を unigram でモデル化することで確率密度比を推定し，モデルには最大エントロピー法を用いている．ただしどちらの研究もタスクは WSD ではない．しかもターゲット領域のラベル付きデータを利

[2] この論文 (新納，佐々木 2013) は本論文と同じタスクに対して，一部同じデータを用いた実験結果を示しているため，考察において提案手法との比較を行う．

用しているために，教師なし領域適応手法でもない．また新納は WSD の領域適応に共変量シフト下の学習を用いているが (新納，佐々木 2014)，そこでは Daumé が提案した素性空間拡張法 (Feature Augmentation)(Daumé 2007) を組み合わせて利用しているために，これも教師なし領域適応手法ではない．

一方，共変量シフト下の学習は，事例への重み付き学習の一種である．Jiang は識別精度を悪化させるようなデータを Misleading データとして訓練データから取り除いて学習することを試みた (Jiang and Zhai 2007)．これは Misleading データの重みを 0 にした学習と見なせるため，この手法も重み付き学習手法と見なせる．吉田はソース領域内の訓練データ $\boldsymbol{x}$ がターゲット領域から見て外れ値と見なせた場合，$\boldsymbol{x}$ を Misleading と判定し，それらを訓練データから取り除いて学習している (吉田，新納 2014)．これは WSD の教師なし領域適応手法であるが，Misleading データの検出は困難であり，精度の改善には至っていない．また WSD の領域適応をタスクとした古宮の手法 (古宮，小谷，奥村 2013) も重み付き学習と見なせる．そこでは複数のソース領域のコーパスを用意し，そこから訓練事例をランダムに選択し，選択された訓練データセットの中で，ターゲット領域のテストデータを識別するのに最も適した訓練データセットを選ぶ．これは全ソース領域のコーパスの訓練データから選択された訓練データの重みを 1，それ以外を重み 0 としていることを意味する．ただし複数のソース領域のコーパスから対象単語のラベル付き訓練データを集めるのは実際は困難である．また古宮は上記の研究以外にも WSD の領域適応の研究 (Komiya and Okumura 2011, 2012; 古宮，奥村 2012) を行っているが，これらは教師付き学習手法となっている．

3　期待損失最小化に基づく共変量シフト下の学習

対象単語 w の語義の集合を C，また w の用例 $\boldsymbol{x}$ 内の w の語義を c と識別したときの損失関数を $l(\boldsymbol{x}, c, d)$ で表す．d は w の語義を識別する分類器である．$P_T(\boldsymbol{x}, c)$ をターゲット領域上の分布とすれば，本タスクにおける期待損失 L_0 は以下で表せる．

$$L_0 = \sum_{\boldsymbol{x},c} l(\boldsymbol{x}, c, d) P_T(\boldsymbol{x}, c)$$

また $P_S(\boldsymbol{x}, c)$ をソース領域上の分布とすると以下が成立する．

$$L_0 = \sum_{\boldsymbol{x},c} l(\boldsymbol{x}, c, d) \frac{P_T(\boldsymbol{x}, c)}{P_S(\boldsymbol{x}, c)} P_S(\boldsymbol{x}, c)$$

ここで共変量シフトの仮定から

$$\frac{P_T(\boldsymbol{x}, c)}{P_S(\boldsymbol{x}, c)} = \frac{P_T(\boldsymbol{x}) P_T(c|\boldsymbol{x})}{P_S(\boldsymbol{x}) P_S(c|\boldsymbol{x})} = \frac{P_T(\boldsymbol{x})}{P_S(\boldsymbol{x})}$$

となり，$w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ とおくと以下が成立する．

$$L_0 = \sum_{\boldsymbol{x},c} w(\boldsymbol{x}) l(\boldsymbol{x}, c, d) P_S(\boldsymbol{x}, c)$$

訓練データを $D = \{(\boldsymbol{x}_i, c_i)\}_{i=1}^N$ とし，$P_S(\boldsymbol{x}, c)$ を経験分布で近似すれば，

$$L_0 \approx \frac{1}{N} \sum_{i=1}^N w(\boldsymbol{x}_i) l(\boldsymbol{x}_i, c_i, d)$$

となるので，期待損失最小化の観点から考えると，共変量シフトの問題は以下の式 L_1 を最小にする d を求めればよいことがわかる．

$$L_1 = \sum_{i=1}^N w(\boldsymbol{x}_i) l(\boldsymbol{x}_i, c_i, d) \tag{1}$$

分類器 d として以下の事後確率最大化推定に基づく識別を考える．

$$d(\boldsymbol{x}) = \arg\max_c P_T(c|\boldsymbol{x})$$

また損失関数として対数損失 $-\log P_T(c|\boldsymbol{x})$ を用いれば，式 (1) は以下となる．

$$L_1 = -\sum_{i=1}^N w(\boldsymbol{x}_i) \log P_T(c_i|\boldsymbol{x}_i)$$

つまり，分類問題の解決に $P_T(c|\boldsymbol{x}, \boldsymbol{\lambda})$ のモデルを導入するアプローチを取る場合，共変量シフト下での学習では，確率密度比を重みとした以下に示す重み付き対数尤度 $L(\boldsymbol{\lambda})$ を最大化するパラメータ $\boldsymbol{\lambda}$ を求める形となる．

$$L(\boldsymbol{\lambda}) = \sum_{i=1}^N w(\boldsymbol{x}_i) \log P_T(c_i|\boldsymbol{x}_i, \boldsymbol{\lambda}) \tag{2}$$

ここではモデルとして以下の式で示される最大エントロピー法を用いる．

$$P_T(c|\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{x}, \boldsymbol{\lambda})} \exp\left(\sum_{j=1}^M \lambda_j f_j(\boldsymbol{x}, c)\right) \tag{3}$$

$\boldsymbol{x} = (x_1, x_2, \cdots, x_M)$ が入力，c がクラスである．関数 $f_j(\boldsymbol{x}, c)$ は素性関数であり，実質 $\boldsymbol{x}$ の真のクラスが c のときに x_j を返し，そうでないとき 0 を返す関数に設定される．$Z(\boldsymbol{x}, \boldsymbol{\lambda})$ は正規化項であり，以下で表せる．

$$Z(\boldsymbol{x}, \boldsymbol{\lambda}) = \sum_{c \in C} \exp\left(\sum_{j=1}^M \lambda_j f_j(\boldsymbol{x}, c)\right) \tag{4}$$

そして $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_M)$ が素性に対応する重みパラメータとなる．

4 確率密度比の算出

確率密度比 $w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ の算出法は大きく 2 つに分類できる．1 つは $P_S(\boldsymbol{x})$ と $P_T(\boldsymbol{x})$ を各々推定し，その比を取る手法であり，もう 1 つは $w(\boldsymbol{x})$ を直接モデル化する手法である．ここでは前者の方法として論文 (新納，佐々木 2014) において提案された手法を利用する．簡単化のために本論文ではこの手法を NB 法と名付ける．また後者の方法としては論文 (Kanamori, Hido, and Sugiyama 2009) において提案された拘束無し最小二乗重要度適合法 (unconstrained Least-Squares Importance Fitting, uLSIF) を利用する．

4.1 NB 法

対象単語 w の用例 $\boldsymbol{x}$ の素性リストを $\{f_1, f_2, \cdots, f_n\}$ とする．求めるのは領域 $R \in \{S, T\}$ 上の $\boldsymbol{x}$ の分布 $P_R(\boldsymbol{x})$ である．ここで Naive Bayes で使われるモデルを用いる．Naive Bayes のモデルでは以下を仮定する．

$$P_R(\boldsymbol{x}) = \prod_{i=1}^{n} P_R(f_i)$$

領域 R のコーパス内の w の全ての用例について素性リストを作成しておく．ここで用例の数を $N(R)$ とおく．また $N(R)$ 個の用例の中で，素性 f が現れた用例数を $n(R, f)$ とおく．MAP 推定でスムージングを行い，$P_R(f)$ を以下で定義する (高村 2010).

$$P_R(f) = \frac{n(R, f) + 1}{N(R) + 2}$$

以上より，ソース領域 S の用例 $\boldsymbol{x}$ に対して，確率密度比 $w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ が計算できる．

$$w(\boldsymbol{x}) = \frac{P_T(\boldsymbol{x})}{P_S(\boldsymbol{x})} = \prod_{i=1}^{n} \left(\frac{n(T, f_i) + 1}{N(T) + 2} \cdot \frac{N(S) + 2}{n(S, f_i) + 1} \right)$$

4.2 uLSIF

ソース領域内のデータを $\{\boldsymbol{x}_i^s\}_{i=1}^{N_s}$，ターゲット領域内のデータを $\{\boldsymbol{x}_i^t\}_{i=1}^{N_t}$ とする uLSIF では確率密度比 $w(\boldsymbol{x})$ を以下の式でモデル化する．

$$\begin{aligned} w(\boldsymbol{x}) &= \sum_{l=1}^{b} \alpha_l \psi_l(\boldsymbol{x}) \\ &= \boldsymbol{\alpha} \cdot \boldsymbol{\psi}(\boldsymbol{x}) \end{aligned}$$

ただしここで，$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_b)$，$\boldsymbol{\psi}(\boldsymbol{x}) = (\psi_1(\boldsymbol{x}), \psi_2(\boldsymbol{x}), \cdots, \psi_b(\boldsymbol{x}))$ である．また α_l は正の実数であり，$\psi_l(\boldsymbol{x})$ は基底関数と呼ばれるソース領域のデータ $\boldsymbol{x}$ から正の実数値への関数である．uLSIF では，概略，自然数 b と基底関数 $\boldsymbol{\psi}(\boldsymbol{x})$ を定めた後に，パラメータ $\boldsymbol{\alpha}$ を推定する手順をとる．

説明の都合上，b と $\boldsymbol{\psi}(\boldsymbol{x})$ が定まった後の $\boldsymbol{\alpha}$ の推定を先に説明する．$w(\boldsymbol{x})$ のモデルを $\hat{w}(\boldsymbol{x})$ とおくと，パラメータ α_l を推定するには，$w(\boldsymbol{x})$ と $\hat{w}(\boldsymbol{x})$ の平均 2 乗誤差 $J_0(\boldsymbol{\alpha})$ を最小にするような $\boldsymbol{\alpha}$ を求めれば良い．$w(\boldsymbol{x}) = P_T(\boldsymbol{x})/P_S(\boldsymbol{x})$ に注意すると，$J_0(\boldsymbol{\alpha})$ は以下のように変形できる．

$$\begin{aligned} J_0(\boldsymbol{\alpha}) &= \frac{1}{2}\int (\hat{w}(\boldsymbol{x}) - w(\boldsymbol{x}))^2 P_S(\boldsymbol{x}) d\boldsymbol{x} \\ &= \frac{1}{2}\int \hat{w}(\boldsymbol{x})^2 P_S(\boldsymbol{x}) d\boldsymbol{x} - \int \hat{w}(\boldsymbol{x}) w(\boldsymbol{x}) P_S(\boldsymbol{x}) d\boldsymbol{x} + \frac{1}{2}\int w(\boldsymbol{x})^2 P_S(\boldsymbol{x}) d\boldsymbol{x} \\ &= \frac{1}{2}\int \hat{w}(\boldsymbol{x})^2 P_S(\boldsymbol{x}) d\boldsymbol{x} - \int \hat{w}(\boldsymbol{x}) P_T(\boldsymbol{x}) d\boldsymbol{x} + \frac{1}{2}\int w(\boldsymbol{x})^2 P_S(\boldsymbol{x}) d\boldsymbol{x} \end{aligned}$$

3 項目の式は定数なので，$J_0(\boldsymbol{\alpha})$ を最小にするには，以下の $J(\boldsymbol{\alpha})$ を最小にすればよい．

$$J(\boldsymbol{\alpha}) = \frac{1}{2}\int \hat{w}(\boldsymbol{x})^2 P_S(\boldsymbol{x}) d\boldsymbol{x} - \int \hat{w}(\boldsymbol{x}) P_T(\boldsymbol{x}) d\boldsymbol{x}$$

$J(\boldsymbol{\alpha})$ を経験分布で近似した $\widehat{J}(\boldsymbol{\alpha})$ は以下となる．

$$\begin{aligned} \widehat{J}(\boldsymbol{\alpha}) &= \frac{1}{2N_s}\sum_{i=1}^{N_s} \widehat{w}(\boldsymbol{x}_i^s)^2 - \frac{1}{N_t}\sum_{j=1}^{N_t} \widehat{w}(\boldsymbol{x}_j^t) \\ &= \frac{1}{2}\sum_{l,l'=1}^{b} \alpha_l \alpha_{l'} \left(\frac{1}{N_s}\sum_{i=1}^{N_s} \psi_l(\boldsymbol{x}_i^s)\psi_{l'}(\boldsymbol{x}_i^s)\right) - \sum_{l=1}^{b} \alpha_l \left(\frac{1}{N_t}\sum_{j=1}^{N_t} \psi_l(\boldsymbol{x}_j^t)\right) \\ &= \frac{1}{2}\boldsymbol{\alpha}^T \widehat{H}\boldsymbol{\alpha} - \widehat{h}^T\boldsymbol{\alpha} \end{aligned} \tag{5}$$

ここで $\widehat{H}$ は $b \times b$ の行列であり，その l 行 l' 列の要素 $\widehat{H}_{l,l'}$ は以下である．

$$\widehat{H}_{l,l'} = \frac{1}{N_s}\sum_{i=1}^{N_s} \psi_l(\boldsymbol{x}_i^s)\psi_{l'}(\boldsymbol{x}_i^s)$$

また $\widehat{h}$ は b 次元のベクトルであり，その l 次元目の要素 $\widehat{h}_l$ は以下である．

$$\widehat{h}_l = \frac{1}{N_t}\sum_{j=1}^{N_t} \psi_l(\boldsymbol{x}_j^t)$$

$\widehat{J}(\boldsymbol{\alpha})$ の最小値を求める際に正則化を行う．このとき付加する正則化項を L2 ノルムに設定し，

$\boldsymbol{\alpha} > 0$ の条件を外して，以下の最小化問題を解く．ここでパラメータ λ が導入されることに注意する．λ は基底関数を設定する際に決められる．

$$\min_{\boldsymbol{\alpha}} \left[\frac{1}{2} \boldsymbol{\alpha}^T \widehat{H} \boldsymbol{\alpha} - \widehat{h}^T \boldsymbol{\alpha} + \frac{\lambda}{2} \boldsymbol{\alpha}^T \boldsymbol{\alpha} \right]$$

この最小化問題は制約のない凸 2 次計画問題であるために，唯一の大域解が得られる．その解は以下である．

$$\tilde{\boldsymbol{\alpha}} = (\widehat{H} + \lambda I_b)^{-1} \widehat{h}^T \tag{6}$$

最後に $\boldsymbol{\alpha} > 0$ の条件に合わせるように，以下の調整を行う．

$$\begin{aligned} \widehat{\boldsymbol{\alpha}} &= ((\max(0, \tilde{\alpha_1}), \max(0, \tilde{\alpha_2}), \cdots, \max(0, \tilde{\alpha_b})) \\ &= \max(0_b, \tilde{\boldsymbol{\alpha}}) \end{aligned} \tag{7}$$

パラメータ b と基底関数の設定であるが，まず，b については以下で設定する[3]．

$$b = \min(100, N_t)$$

次にターゲット領域のデータから重複を許さずに b 個の点をランダムに取り出す．それらの点を $\{\boldsymbol{x}_j^t\}_{j=1}^b$ とおく．そして基底関数 $\psi_l(\boldsymbol{x})$ を以下のガウシアンカーネルで定義する．

$$\psi_l(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{x}_l^t) = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{x}_l^t||^2}{\sigma^2} \right)$$

以上より，確率密度比を求めるために残されているパラメータは正則化項の係数 λ とガウシアンカーネルの幅 σ の 2 つである．これらのパラメータはグリッドサーチの交差検定で求める．まずソース領域のデータとターゲット領域のデータをそれぞれ交わりのない R 個の部分集合に分割する．それらの部分集合の中で r 番目の部分集合を除き，残りを結合した集合を作る．それらを新たなソース領域のデータとターゲット領域のデータと見なす．そして λ と σ をある値に設定し，式 (6) と式 (7) より $\boldsymbol{\alpha}$ を求め，式 (5) より $\widehat{J}(\boldsymbol{\alpha})^{(r)}$ の値を求める．r を 1 から R まで変化させることで，R 個の $\widehat{J}(\boldsymbol{\alpha})^{(r)}$ の値が求まり，それらを平均した値を λ と σ に対する $\widehat{J}(\boldsymbol{\alpha})$ の値とする．次に λ と σ を変化させ，上記手順で得られる $\widehat{J}(\boldsymbol{\alpha})$ の値が最小となる $\hat{\lambda}$ と $\hat{\sigma}$ を求め，これを λ と σ の推定値とする．

[3] 本実験では b の値は最大 100 となるが，この 100 という数値はオリジナルの論文 (Kanamori et al. 2009) で使われた値であり，本論文でのなんらかの予備実験から得た値ではない．uLSIF の実験結果はこの値を調整することで多少の向上があったかもしれない．

4.3 $P_S(\boldsymbol{x})$ の補正による確率密度比の算出

WSD のタスクでは NB 法あるいは uLSIF で算出される確率密度比は小さい値を取る傾向があり，実際の学習で用いる際には，少し上方に修正した値を取る方が最終の識別結果が改善されることが多い．これは以下の 2 点から生じていると考えられる．

- T に $\boldsymbol{x}$ が入っているかは確率的であるが，S には必ず $\boldsymbol{x}$ が入っている．
- $P_S(\boldsymbol{x})$ を推定するために $\boldsymbol{x} \in S$ を用いるため，訓練データである $\boldsymbol{x}$ に過学習した結果 $P_S(\boldsymbol{x})$ は $P_T(\boldsymbol{x})$ に比べて高く見積もられてしまう．

このため，求まった確率密度比を上方に修正する手法が存在する．論文 (杉山 2006) では確率密度比 $w(\boldsymbol{x})$ を p 乗 $(0 < p < 1)$ することを提案している．また論文 (Yamada et al. 2011) では以下で示される相対確率密度比 $w'(\boldsymbol{x})$ を確率密度比として利用することを提案している．

$$w'(\boldsymbol{x}) = \frac{P_T(\boldsymbol{x})}{\alpha P_S(\boldsymbol{x}) + (1-\alpha) P_T(\boldsymbol{x})}$$

ここで $0 < \alpha < 1$ である．

確率密度比 $w(\boldsymbol{x})$ が 1 以下である場合，$w(\boldsymbol{x})$ を p 乗すると上方に修正できることは，それらの比の対数を取れば，$\log w(\boldsymbol{x}) < 0$であることから明らかである．

$$\log \frac{w(\boldsymbol{x})^p}{w(\boldsymbol{x})} = (p-1) \log w(\boldsymbol{x}) > 0$$

また相対確率密度比 $w'(\boldsymbol{x})$ は以下の変形から $w(\boldsymbol{x})$ を上方に修正していると見なせる．

$$\begin{aligned} w'(\boldsymbol{x}) &= \frac{P_T(\boldsymbol{x})}{\alpha P_S(\boldsymbol{x}) + (1-\alpha) P_T(\boldsymbol{x})} \\ &= \frac{1}{\alpha + (1-\alpha) w(\boldsymbol{x})} w(\boldsymbol{x}) \\ &> \frac{1}{\alpha + (1-\alpha)} w(\boldsymbol{x}) \\ &= w(\boldsymbol{x}) \end{aligned}$$

確率密度比が 1 以上である場合，これらの手法は確率密度比を下方に修正するので，正確には確率密度比を 1 に近づける手法である．しかし，ほとんどの訓練データの確率密度比は 1 以下であるために，ここではこれらの手法を上方修正する手法と呼び，提案手法と対比させる．

本論文では確率密度比を上方に修正するために，ソース領域のデータとターゲット領域のデータを合わせたデータを新たにソース領域のデータとみなし，NB 法を用いて $P_S(\boldsymbol{x})$ を補正することを提案する．これは S のスパース性を緩和させることを狙ったものである．確率密度比が真の値よりも低く見積もられる原因の 1 つは，$P_S(\boldsymbol{x})$ が真の値よりも高く見積もられるからだと考える．さらにその原因が S のスパース性なので，スパース性を緩和するために S にデータを追加するというアイデアである．ただし追加するデータは S と類似の領域のデータであるこ

とが望ましい．WSD の領域適応の場合，S と T は完全に異なることはなく，比較的似ているために，追加するデータとして T のデータが利用できると考えた．

提案手法の新たなソース領域を $S+T$ で表せば，$P_S(\boldsymbol{x}) > P_{S+T}(\boldsymbol{x})$ が成立していると考えるのは自然であり，この不等式が成立していれば，提案手法により確率密度比は上方に修正される．ただし，ここで提案手法は必ずしも NB 法の確率密度比を上方に修正できるとは限らないことに注意する．また提案手法は NB 法の確率密度比が 1 以下かどうかには無関係であることにも注意する．NB 法の確率密度比が 1 以上であっても，上方に修正する可能性がある．また $P_{S+T}(\boldsymbol{x})$ は以下の式を利用して求められる．

$$\begin{aligned} P_{S+T}(f) &= \frac{n(S+T,f)+1}{N(S+T)+2} \\ &= \frac{n(S,f)+n(T,f)+1}{N(S)+N(T)+2} \end{aligned}$$

5　実験

BCCWJ の PB（書籍），OC（Yahoo! 知恵袋）及び PN（新聞）を異なった領域として実験を行う．SemEval-2 の日本語 WSD タスク (Okumura et al. 2010) ではこれら領域のコーパスの一部に語義タグを付けたデータを公開しており，そのデータを利用する．この 3 つの領域からある程度頻度のある多義語 16 単語を WSD の対象単語とする．これら単語と辞書上での語義数及び各コーパスでの頻度と語義数を表 1 に示す[4]．領域適応の方向としては OC → PB，PB → PN，PN → OC，OC → PN，PN → PB，PB → OC の計 6 通りの方向が存在する．

本稿で利用した素性は以下の 8 種類である．(e0) w の表記，(e1) w の品詞，(e2) w_{-1} の表記，(e3) w_{-1} の品詞，(e4) w_1 の表記，(e5) w_1 の品詞，(e6) w の前後 3 単語までの自立語の表記，(e7) e6 の分類語彙表の番号の 4 桁と 5 桁．なお対象単語の直前の単語を w_{-1}，直後の単語を w_1 としている．

対象単語 w についてソース領域 S からターゲット領域 T への領域適応の実験について説明する．ソース領域 S の訓練データのみを用いて，手法 A により分類器を学習し w に対する正解率を求める．16 種類の各対象単語 $(w_1, w_2, \cdots, w_{16})$ に対する正解率の平均，つまりマクロ平均をソース領域 S からターゲット領域 T に対する手法 A の正解率とする．結果，手法 A について 6 種類の各領域適応に対しての正解率が得られる．それらの平均を手法 A の平均正解率とする．

上記の手法 A としては，以下の 8 種類を試す．(1) 重みを考慮しない（重みを 1 で固定する）手法 (Base)，(2) NB 法による重みをつけた手法 (NB)，(3) NB 法の重みを p 乗した値を重みに

[4] 語義は岩波国語辞書がもとになっている．そこでの中分類までを対象にした．また「入る」は辞書上の語義が 3 つだが，OC や PB では 4 つの語義がある．これは SemEval-2 の日本語 WSD タスクでは新語義のタグも許しているからである．

表 1 対象単語

単語	辞書上の語義数	OC での頻度	OC での語義数	PB での頻度	PB での語義数	PN での頻度	PN での語義数
言う	3	666	2	1114	2	363	2
入れる	3	73	2	56	3	32	2
書く	2	99	2	62	2	27	2
聞く	3	124	2	123	2	52	2
子供	2	77	2	93	2	29	2
時間	4	53	2	74	2	59	2
自分	2	128	2	308	2	71	2
出る	3	131	3	152	3	89	3
取る	8	61	7	81	7	43	7
場合	2	126	2	137	2	73	2
入る	3	68	4	118	4	65	3
前	3	105	3	160	2	106	4
見る	6	262	5	273	6	87	3
持つ	4	62	4	153	3	59	3
やる	5	117	3	156	4	27	2
ゆく	2	219	2	133	2	27	2
平均	3.44	148.19	2.94	199.56	3.00	75.56	2.69

する手法 (P-NB)，(4) NB 法の重みを相対確率密度比により上方修正した値を重みにする手法 (A-NB)，(5) uLSIF による重みをつけた手法 (uLISF)，(6) uLSIF の重みを p 乗した値を重みにする手法 (P-uLSIF)，(7) uLSIF の重みを相対確率密度比により上方修正した値を重みにする手法 (A-uLSIF)，(8) 提案手法，またすべての手法において学習アルゴリズムとしては最大エントロピー法を用いた．またその実行にはツールの Classias を用いた (Okazaki 2009)．S から T への領域適応における各手法の正解率を表 2 に示す．ただし P-NB，A-NB，P-uLSIF，A-uLSIF については p と α のパラメータが存在する．これらの値については，その値を 0.01 から 0.09 まで 0.01 刻み，及び 0.1 から 0.9 まで 0.1 刻みで変化させ，平均正解率が最もよい値を示した値を採用した．結果，P-NB については $p = 0.2$，A-NB については $\alpha = 0.01$，P-uLSIF については $p = 0.04$，A-uLSIF については $\alpha = 0.01$ の値を採用した．

表 2 が示すように，領域適応のタイプ毎に最適な手法は異なるが，平均正解率としては提案手法が最も高い値を示した．また P-NB と A-NB の平均正解率は NB の平均正解率よりも高く，P-uLSIF と A-uLSIF の平均正解率は uLSIF の平均正解率よりも高い．つまり確率密度比を上方に修正する手法が有効であったことがわかる．

また有意差を検定するために以下の実験を行った．まず対象単語毎に OC のデータからランダムに 9 割のデータ取り出し，それらのデータセットを OC-1 とする．これを 20 回行い，

OC-1，OC-2，…，OC-20 を作成する．同様に PB のデータから PB-1，PB-2，…，PB-20 を作成する．また同様に PN のデータから PN-1，PN-2，…，PN-20 を作成する．そしてデータセットの組 (OC-i, PB-i, PN-i) を用いて，前述した実験と同様の実験を行い，20 個の平均正解率を算出し t-検定（両側検定の有意水準 5%）を行った．結果を表 3 に示す．表 3 における評価値は以下の式により計算されたものである．

$$\frac{\bar{X_1} - \bar{X_2}}{\sqrt{\left(\frac{1}{n_2} + \frac{1}{n_2}\right) \frac{n_1 S_1^2 + n_2 S_2^2}{n_1 + n_2 - 2}}}$$

ここで $\bar{X_1}$ と S_1^2 が提案手法の 20 個の平均正解率の平均と分散であり，$\bar{X_2}$ と S_2^2 が比較対象の手法の 20 個の平均正解率の平均と分散である．n_1 と n_2 は共にサンプル数 20 である．この評価値が自由度 38 の t 分布の 0.975 の分位点 2.0244 よりも大きい場合に，提案手法が対応する手法に対して有意であると判定される．

表 3 が示すように P-NB 以外の全ての手法に対して，提案手法が有意に優れていた．

表 2　各手法の平均正解率 (%)

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	平均正解率
Base	71.63	77.00	69.20	67.78	74.74	69.91	71.71
NB	72.25	76.44	63.83	68.96	62.62	68.62	68.79
P-NB	71.19	77.24	**71.25**	68.56	73.35	69.84	71.90
A-NB	70.80	77.25	70.55	67.86	72.58	69.95	71.50
uLSIF	70.08	72.89	68.54	68.40	71.10	67.60	69.77
P-uLSIF	71.60	**77.36**	69.27	67.63	**74.87**	70.06	71.80
A-uLSIF	71.11	76.59	69.52	67.19	74.61	70.06	71.51
提案手法	**71.66**	77.00	70.32	**69.40**	74.61	**70.29**	**72.21**

表 3　有意差の検定結果

	20 個の平均正解率の平均 $\bar{X}$	20 個の平均正解率の分散 $\bar{S^2}$ ($\times 10^{-4}$)	評価値	検定結果 提案手法に対して
Base	0.71031	0.11358	2.75626	**有意差あり**
NB	0.67347	1.10743	14.16433	**有意差あり**
P-NB	0.71412	0.18829	0.70677	有意差なし
A-NB	0.70878	0.31027	3.13972	**有意差あり**
uLSIF	0.65627	1.94611	16.29127	**有意差あり**
P-uLSIF	0.67076	2.00794	12.15918	**有意差あり**
A-uLSIF	0.66959	1.98588	12.53072	**有意差あり**
提案手法	0.71553	0.56790	—	—

6 考察

6.1 確率密度比を上方修正しないケース

「p 乗する」あるいは「相対確率密度比を取る」という手法は，元の確率密度比が 1 以下である全てのデータに対してその値を上方に修正するが，提案手法は一部のデータに対しては NB 法の確率密度比が 1 以下であっても，それらを上方に修正できない．提案手法により確率密度比の値が大きくならず，逆に小さくなったデータの個数を表 4 に示す．

ほとんどのデータに対して，その確率密度比を上方に修正しているが，修正できていないデータが極端に多いケースも存在する．例えば，PB → PN に関しては「言う」「自分」「見る」「やる」「ゆく」，OC → PN に関しては「書く」「見る」「やる」「ゆく」である．これらに関してのみ Base と NB と提案手法の正解率の比較を表 5 に示す．

表 5 からわかるように，上方修正ができないデータが多くなると，提案手法は NB 法よりも正解率が下がっている．ただし，下方に修正した場合には必ず正解率が下がるとも言えないことに注意したい．例えば，確率密度比の値を下げないようにするには提案手法を修正し，「NB 法の値を上方に修正できなければ，NB 法の値をそのまま使う」という形にすれば良い．この修正案の手法も試した結果を表 6 に示す．修正案の手法の平均正解率は，提案手法よりも若干悪かった．

上記の実験は NB 法による確率密度比が 1 以下かどうかは考慮していない．「p 乗する」や

表 4 上方修正できなかったデータの個数

単語	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC
言う	4 (666)	509 (1114)	7 (363)	48 (666)	3 (363)	142 (1114)
入れる	0 (73)	1 (56)	0 (32)	4 (73)	0 (32)	0 (56)
書く	3 (99)	9 (62)	0 (27)	39 (99)	0 (27)	0 (62)
聞く	0 (124)	16 (123)	1 (52)	20 (124)	0 (52)	4 (123)
子供	0 (77)	17 (93)	0 (29)	15 (77)	0 (29)	0 (93)
時間	0 (53)	2 (74)	1 (59)	0 (53)	0 (59)	0 (74)
自分	1 (128)	209 (308)	0 (71)	9 (128)	0 (71)	42 (308)
出る	0 (131)	7 (152)	0 (89)	6 (131)	1 (89)	3 (152)
取る	0 (61)	2 (81)	0 (43)	0 (61)	0 (43)	1 (81)
場合	1 (126)	5 (137)	0 (73)	8 (126)	0 (73)	1 (137)
入る	1 (68)	4 (118)	1 (65)	1 (68)	0 (65)	6 (118)
前	2 (105)	3 (160)	0 (106)	5 (105)	0 (106)	0 (160)
見る	12 (262)	99 (273)	0 (87)	92 (262)	0 (87)	1 (273)
持つ	0 (62)	8 (153)	2 (59)	0 (62)	0 (59)	12 (153)
やる	1 (117)	134 (156)	0 (27)	77 (117)	0 (27)	1 (156)
ゆく	10 (219)	105 (133)	0 (27)	216 (219)	0 (27)	0 (133)

表 5　上方修正できなかったデータの正解率 (%)

	Base	NB	提案手法
PB → PN 「言う」	93.39	91.46	92.01
PB → PN 「自分」	98.59	98.59	98.59
PB → PN 「見る」	70.11	77.01	77.01
PB → PN 「やる」	96.30	96.30	96.30
PB → PN 「ゆく」	85.19	81.48	70.37
OC → PN 「書く」	69.70	73.74	73.74
OC → PN 「見る」	56.92	56.54	56.54
OC → PN 「やる」	95.65	95.65	95.65
OC → PN 「ゆく」	68.49	68.49	68.49
平均	81.59	82.14	80.97

表 6　修正版提案手法の平均正解率 (%)

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	平均正解率
Base	71.63	77.00	69.20	67.78	74.74	69.91	71.71
NB	72.25	76.44	63.83	68.96	62.62	68.62	68.79
提案手法	71.66	77.00	70.32	69.40	74.61	70.29	72.21
修正版提案手法	71.45	77.43	70.67	68.00	74.98	69.83	72.07

「相対確率密度比を取る」手法では，確率密度比が 1 以上の場合に，その値を逆に小さくしている．確率密度比が 1 以上の場合に，上方修正する方がよいのか下方修正する方がよいのかは未解決である．参考として上記の修正案の手法を更に修正し，「 NB 法の値が 1 以上の場合，あるいは NB 法の値を上方に修正できな場合には NB 法の値をそのまま使う」という形の実験も行った．結果，平均正解率は 72.14 と若干改善はされたが，提案手法よりも若干悪いことに変化はなかった．

データの確率密度比（重み）はその値の大きさが重要ではなく，他データとの重みとの関係が本質的である．例えば全てのデータの重みを 10 倍して，値自体を増やしても，推定できるパラメータが変化しないのは，重み付き対数尤度（式 2）の最大化する部分が変化しないことから明らかである．

データの重みはタスクの背景知識から，その重要度を設定していくか，そのデータを数値化した後に確率密度比という観点から設定していくしか方法はないと考える．提案手法は後者であり，コーパスのスパース性への対処から NB 法を改良した手法と考えている．上方修正することに，どのような意味があるかを調べることは今後の課題である．

6.2　提案手法の重みの上方修正

提案手法は，確率密度比を上方修正する手法と組み合わせて利用することで更なる精度改善も可能である．提案手法の確率密度比を p 乗した場合の平均正解率の変化を図 1 に示す．$p = 0.6$ のとき最大値 72.54% をとった．また提案手法の確率密度比に対してパラメータ α の相対確率密度比をとった場合の平均正解率の変化を図 2 に示す．$\alpha = 0.6$ のとき最大値 72.30% をとった．ともに確率密度比を上方修正することで平均正解率は改善されている．

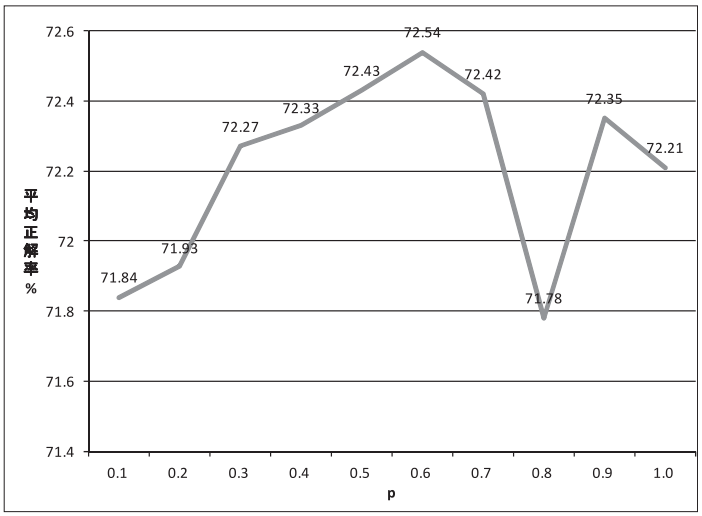

図 1　p 乗による提案手法値の上方修正

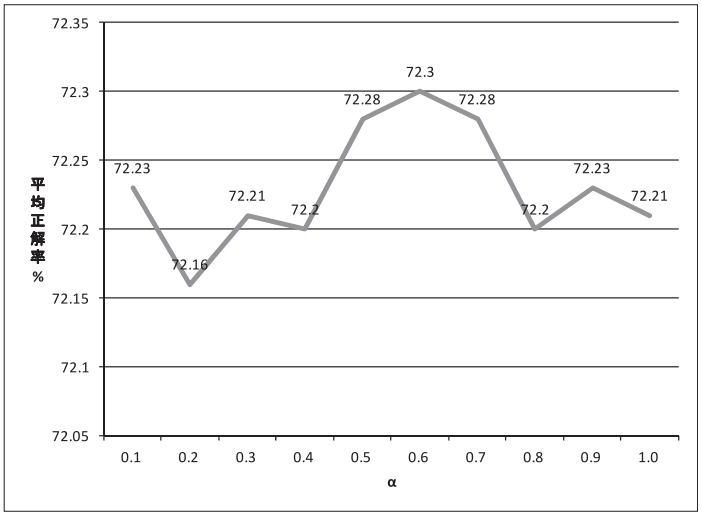

図 2　相対確率密度比による提案手法値の上方修正

本論文の以降の記述において，提案手法の重みを p 乗した値を重みにする手法を「P-提案手法」，提案手法の重みを相対確率密度比により上方修正した値を重みにする手法を「A-提案手法」と名付ける．ここで $p = 0.6$，$\alpha = 0.6$ である．

また前節で行った有意差の検定を「P-提案手法」と「A-提案手法」に対しても行った．結果，「P-提案手法」は P-NB や提案手法を含む全ての手法に対して有意に優れていた．ただし「A-提案手法」は P-NB や提案手法とに有意な差はなかった．

6.3 Misleading データからの評価

本論文で提案した確率密度比（重み）は NB 法や uLSIF による確率密度比よりも，有効に機能していた．ただし真の確率密度比の値は未知であるために，真の値に近いかどうかという観点での評価は不可能である．また重みの設定だけで，どの程度まで平均正解率が向上できるのかも未知である．一方，Misleading データを削除してから学習を行うことでかなりの精度向上が可能であることが論文 (吉田，新納 2014) により示されている．Misleading データを削除してから学習することは，Misleading データの重みを 0，それ以外のデータの重みを 1 とした重み付き学習と見なせる．この重み付けが真の確率密度比と類似しているかどうかは不明だが，Misleading データに対してはできるだけ小さな重みを与える手法が優れているとみなせる．そこでここでは各手法において Misleading データに付与された重みを調べることで手法を評価する．

まず論文 (吉田，新納 2014) で行ったように，しらみつぶしに Misleading を見つけ出す．領域 S から領域 T の領域適応において，対象単語 w の S 上のラベル付きデータ D が存在する．まず D で学習した識別器の T に対する正解率 p_0 を測る．次に D から 1 つデータ x を取り除き，$D - \{x\}$ から学習した識別器の T に対する正解率 p_1 を測る．$p_1 > p_0$ となった場合，データ x を Misleading データと見なす．これを D 内のすべてのデータに対して行い，S から T の領域適応における対象単語 w の Misleading データを見つける．この処理によって見つけ出された Misleading データの個数を表 7 示す．括弧内の数値は全データ数である．

また Misleading による重みを用いた学習の識別結果を表 8 に示す．表中の Mislead がそれにあたる．本論文の実験で得られている平均正解率よりもかなり高い．つまり重みの設定のみでも Base の平均正解率 71.71% を少なくとも 75.42% まで改善可能である．

次に各手法が Misleading データに付与した重みにより手法を評価する．領域 S から領域 T の領域適応において，対象単語 w の S 上のラベル付きデータを $D = \{x_i\}_{i=1}^{N_w}$ とする．まず D 内のデータの重みの平均値 m_w を調べる．

$$m_w = \frac{1}{N_w} \sum_{i=1}^{N_w} w(x_i)$$

表 7 Misleading データの個数

単語	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC
言う	159 (666)	75 (1114)	82 (363)	158 (666)	35 (363)	127 (1114)
入れる	6 (73)	15 (56)	3 (32)	28 (73)	1 (32)	19 (56)
書く	21 (99)	2 (62)	12 (27)	39 (99)	15 (27)	0 (62)
聞く	26 (124)	0 (123)	4 (52)	21 (124)	27 (52)	26 (123)
子供	5 (77)	1 (93)	12 (29)	0 (77)	13 (29)	12 (93)
時間	1 (53)	0 (74)	0 (59)	8 (53)	5 (59)	0 (74)
自分	13 (128)	0 (308)	0 (71)	25 (128)	1 (71)	0 (308)
出る	14 (131)	32 (152)	22 (89)	10 (131)	10 (89)	39 (152)
取る	6 (61)	18 (81)	12 (43)	5 (61)	22 (43)	10 (81)
場合	0 (126)	13 (137)	14 (73)	0 (126)	9 (73)	7 (137)
入る	36 (68)	27 (118)	27 (65)	11 (68)	42 (65)	38 (118)
前	8 (105)	1 (160)	15 (106)	5 (105)	2 (106)	10 (160)
見る	10 (262)	12 (273)	8 (87)	3 (262)	28 (87)	3 (273)
持つ	8 (62)	11 (153)	1 (59)	0 (62)	1 (59)	2 (153)
やる	0 (117)	0 (156)	0 (27)	0 (117)	0 (27)	0 (156)
ゆく	17 (219)	1 (133)	3 (27)	0 (219)	3 (27)	15 (133)

表 8 Misleading による重みを用いた学習の平均正解率 (%)

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	平均正解率
Base	71.63	77.00	69.20	67.78	74.74	69.91	71.71
NB	72.25	76.44	63.83	68.96	62.62	68.62	68.79
uLSIF	70.08	72.89	68.54	68.40	71.10	67.60	69.77
提案手法	71.66	77.00	70.32	69.40	74.61	70.29	72.21
Mislead	74.59	79.27	74.50	72.13	78.69	73.34	**75.42**

次に D 内の Misleading データを $\{x'_j\}_{j=1}^{M_w}$ とする．各 x'_j の重み $w(x'_j)$ が m_w と比較して小さな値であればよいので，対象単語 w に関する Misleading データを用いた評価値 d_w を以下で測る．

$$d_w = \frac{1}{M_w} \sum_{j=1}^{M_w} \frac{w(x'_j)}{m_w}$$

d_w は対象単語 w の訓練データの重みの平均値 m_w に対して，Misleading データ x'_j の重み $w(x'_j)$ の比を取り，その比の平均を取ったものである．このため d_w の値が小さいほど，適切に重み付けできていると考えられる．そして d_w の各単語に関して平均を取った値を，その手法における S から T の Misleading データを用いた評価値（小さいほど良い）とする．これをまとめたものが表 9 である．表 9 が示すように，Misleading データを用いた評価では，NB 法，uLSIF 及び提案手法の 3 つの中で uLSIF が最も優れている．ただし提案手法は NB 法よりも優れてい

表 9　Misleading データからの評価値

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	評価値
Base	1.000	1.000	1.000	1.000	1.000	1.000	1.000
NB	2.005	1.338	3.239	0.612	1.850	1.531	1.762
P-NB	1.210	0.727	1.216	0.772	1.065	0.874	0.977
A-NB	1.403	0.728	1.144	0.778	1.192	0.982	1.038
uLSIF	0.905	0.764	0.927	0.713	0.942	0.831	0.847
P-uLSIF	0.875	0.750	0.813	0.688	0.938	0.751	0.802
A-uLSIF	0.889	0.758	0.838	0.700	0.932	0.794	0.818
提案手法	1.257	0.763	1.138	0.760	1.156	0.858	0.989
P-提案手法	1.116	0.626	1.080	0.792	1.076	0.812	0.917
A-提案手法	1.159	0.708	1.141	0.738	1.086	0.856	0.948

た．更に全ての手法において「p 乗する」，あるいは「相対確率密度比を取る」ことで評価値は改善されており，重みを上方修正する効果があることがわかる．また「p 乗する」と「相対確率密度比を取る」を比較すると，「p 乗する」方が効果があることもわかる．

6.4　負の転移の有無

NB 法や uLSIF は Base よりも平均正解率が低い．これは確率密度比からの重み付き学習が効果がなかったことを示している．この原因として，WSD の領域適応では，領域の変化はあるが，実際には領域適応の問題が生じていない，つまり負の転移 (Rosenstein, Marx, Kaelbling, and Dietterich 2005) が生じていない対象単語がかなり存在するからだと考える．負の転移が生じていなければ，訓練データを全て利用して学習する方が有利であることは明らかであり，重みをつけると逆効果になると考えられる．

この点を確認するために，負の転移が生じているものと生じていないものに分けて，各手法の平均正解率を測ってみる．まず負の転移が生じている単語の判定であるが，これは表 7 で示した Misleading データの個数から行う．ここでは Misleading データが全データの 1 割以下の場合，負の転移が生じないと判定した．結果を表 10 に示す．チェックが付いているものが「負の転移が生じない」と判定したものである．

表 10 でチェックがついていない対象単語に限定して，各手法の平均正解率を測った結果が表 11 である．また逆に表 10 でチェックがついている対象単語に限定して，各手法の平均正解率を測った結果が表 12 である．

表 11 と表 12 からわかるように，NB 法や uLSIF は負の転移が生じる，生じないに関わらず，Base よりも平均正解率が低く，本実験においては有効ではなかった．一方，提案手法は負の転移が生じる場合でも，生じない場合でも Base よりも平均正解率が高く，どちらの場合でも有効であることがわかる．

表 10 負の転移が生じない単語

単語	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC
言う		✓			✓	
入れる	✓		✓		✓	
書く		✓				✓
聞く		✓	✓			
子供	✓	✓		✓		
時間	✓	✓	✓		✓	✓
自分		✓	✓		✓	✓
出る				✓		
取る	✓			✓		
場合	✓	✓		✓		✓
入る						
前	✓	✓		✓	✓	✓
見る	✓	✓	✓	✓		✓
持つ		✓	✓	✓	✓	✓
やる	✓	✓	✓	✓	✓	✓
ゆく	✓	✓		✓		

表 11 負の転移が生じる単語に限定した平均正解率 (%)

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	平均正解率
Base	74.47	63.15	66.28	62.69	67.53	57.58	65.28
NB	70.02	61.01	64.76	63.80	49.78	54.74	60.69
P-NB	72.42	64.45	69.95	64.19	63.23	56.90	65.19
A-NB	72.74	64.22	71.87	62.47	63.34	57.45	65.35
uLSIF	74.05	53.70	63.86	64.33	66.57	55.13	62.94
P-uLSIF	74.59	64.01	66.32	62.33	67.70	57.78	65.46
A-uLSIF	74.19	61.98	66.83	61.75	67.16	57.69	64.93
提案手法	73.06	65.89	69.36	63.86	67.44	57.40	66.17
P-提案手法	73.95	67.33	70.60	63.49	67.72	57.56	66.77
A-提案手法	73.51	65.12	70.11	63.94	67.51	57.30	66.25

また負の転移が生じる場合，提案手法の平均正解率は NB 法の平均正解率の 1.09 倍であり，uLSIF の平均正解率 1.05 倍である．一方，負の転移が生じない場合，提案手法の平均正解率は NB 法の平均正解率の 1.02 倍であり，uLSIF の平均正解率 1.03 倍である．つまり負の転移が生じるケースで提案手法と既存手法（NB 法，uLSIF）との差が大きくなる．

更に確率密度比を上方修正する効果をみてみる．負の転移が生じる場合，NB 法は平均正解率 60.69% が p 乗することで 65.19%，相対確率密度比を取ることで 65.35% まで向上している

表 12　負の転移が生じない単語に限定した平均正解率 (%)

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	平均正解率
Base	68.79	81.62	72.95	71.74	84.01	82.23	76.89
NB	74.48	81.58	62.62	72.98	79.13	82.51	75.55
P-NB	69.54	82.05	70.63	72.47	83.56	83.19	76.91
A-NB	68.86	81.60	68.86	72.05	84.46	82.44	76.38
uLSIF	66.10	79.29	74.55	71.57	76.91	80.06	74.75
P-uLSIF	68.61	81.80	73.07	71.74	84.08	82.33	76.94
A-uLSIF	68.03	81.46	72.98	71.42	84.19	82.43	76.75
提案手法	70.25	80.70	71.56	73.71	83.84	83.18	77.21
P-提案手法	69.36	81.95	72.01	73.10	84.13	83.06	77.27
A-提案手法	69.67	81.86	71.89	72.66	83.93	82.97	77.16

ので，平均的には 7.5% 平均正解率が向上している[5]．同様に計算して uLSIF の平均正解率は 3.6%，提案手法の平均正解率は 0.5% 向上している．負の転移が生じない場合，NB 法は 1.4%，uLSIF は 2.8% 平均正解率が向上している．また提案手法では平均正解率はほとんど変化しない．つまり確率密度比を上方修正する効果は負の転移が生じるケースで顕著になっている．

今後の課題としては Misleading データの検出方法を考案することである．Misleading データを検出し，そのデータに重みを 0 にすることはかなりの精度向上が期待できる．また Misleading データの割合から負の転移の有無を判定し，負の転移が生じる問題にだけ，重み付け学習手法を適用するアプローチも効果があると考えられる．

6.5　トピックモデルの利用

論文 (新納，佐々木 2013) は本論文と同じタスクに対して一部同じデータを用いた実験結果を示している．ここではそこでの実験結果の値と本論文の実験結果の値を比較し，手法間の違いを考察する．

論文 (新納，佐々木 2013) の核となるアイデアは，ターゲット領域 T のトピックモデルを作成し，ターゲット領域に特有のシソーラスを構築することである．このシソーラスの情報を素性として組み込むことで，識別精度を上げることを狙っている．実験は OC → PB と PB → OC の 2 方向である． また対象単語は本論文の 16 単語の他「来る」が含まれている[6]．

OC → PB と PB → OC の領域適応における，本論文の対象単語 16 単語についての識別精度の比較を表 13 と表 14 に示す．なお表中の SVM-TM-kNN は論文 (新納，佐々木 2013) の手法を意味する．

[5] $((65.19 + 65.35)/2)/60.69 \approx 1.075$ から算出した．
[6] 本論文では「来る」は PN の領域において曖昧性がないため対象単語から外した．

表 13 正解率 (%) の比較 (OC → PB)

	SVM-TM-kNN	提案手法	P-提案手法	A-提案手法
言う	79.53	82.23	82.41	82.41
入れる	73.21	76.36	76.36	76.36
書く	85.48	79.03	79.03	79.03
聞く	69.10	65.85	65.85	64.23
子供	41.93	31.18	32.26	32.26
時間	89.18	90.54	90.54	90.54
自分	96.10	94.48	94.48	94.48
出る	61.18	62.50	61.84	61.84
取る	27.16	27.16	27.16	27.16
場合	84.67	84.67	84.67	84.67
入る	52.54	42.37	44.92	44.07
前	83.12	74.38	74.38	74.38
見る	84.24	84.19	84.19	84.19
持つ	77.77	69.28	72.55	69.93
やる	92.94	93.55	93.55	93.55
ゆく	90.97	88.72	88.72	88.72
平均	74.32	71.66	72.06	71.74

表 14 正解率 (%) の比較 (PB → OC)

	SVM-TM-kNN	提案手法	P-提案手法	A-提案手法
言う	80.33	83.03	83.18	83.78
入れる	76.71	73.97	73.97	75.34
書く	73.73	73.74	73.74	73.74
聞く	66.93	58.06	61.29	63.71
子供	22.07	15.58	15.58	15.58
時間	83.01	83.02	83.02	83.02
自分	87.50	87.50	87.50	87.50
出る	70.99	66.41	67.18	64.89
取る	29.50	32.79	31.15	29.51
場合	91.27	90.48	90.48	90.48
入る	57.35	61.76	61.76	58.82
前	89.52	90.38	90.38	90.38
見る	59.54	56.11	56.11	56.11
持つ	77.41	90.16	90.16	86.89
やる	94.01	94.02	94.02	94.02
ゆく	68.03	67.58	68.04	68.49
平均	70.49	70.29	70.47	70.14

対象単語に応じて最も高い正解率の手法は異なるが，平均的には SVM-TM-kNN が最も高い正解率を示している．ただし SVM-TM-kNN はトピックモデルを構築するために，ターゲット領域のコーパスを利用していることに注意したい．本論文の提案手法はターゲット領域の対象単語の用例を用いているが，コーパスは利用していない．つまり利用しているリソースが異なるために，単純に SVM-TM-kNN が提案手法よりも優れているとは結論できない．

また SVM-TM-kNN におけるトピックモデルは素性構築の際に利用されているだけであり，提案手法と競合するものではない．つまり SVM-TM-kNN の手法を利用して，WSD での素性を構築し，それに対して本論文の提案手法を適用することも可能である．今後はこの方向での改良も試みたい．

7　おわりに

本論文では，WSD の領域適応に対して，共変量シフト下の学習を試みた．共変量シフト下の学習では確率密度比を重みとした重み付き学習を行うが，WSD のタスクでは算出される確率密度比が小さくなる傾向があるため，ソース領域のコーパスとターゲット領域のコーパスとを合わせたコーパスをソース領域のコーパスと見なして NB 法を用いる手法を提案した．

BCCWJ の 3 つの領域 OC（Yahoo! 知恵袋），PB（書籍）及び PN（新聞）に共通して出現する多義語 16 単語を対象にして，WSD の領域適応の実験を行った．NB 法，uLSIF 及び提案手法を比較すると，提案手法が最も高い平均正解率を出した．また「p 乗する」や「相対確率密度比を取る」といった確率密度比を上方修正する手法も試し，提案手法のように確率密度比を上方修正する効果を確認した．

また Misleading データをしらみつぶし的に取り出し，Misleading データを用いた手法の評価も行った．Misleading データを利用した評価では uLSIF が優れていたが，提案手法は NB 法の改良になっていることを確認できた．WSD の領域適応の場合，Misleading データの検出あるいは負の転移の有無を判定することが，精度改善に大きく寄与できる．今後はこの点の研究を進めたい．またトピックモデルの利用も検討したい．

参考文献

Chan, Y. S. and Ng, H. T. (2005). “Word Sense Disambiguation with Distribution Estimation.” In *Proceedings of IJCAI-2005*, pp. 1010–1015.

Chan, Y. S. and Ng, H. T. (2006). “Estimating Class Priors in Domain Adaptation for Word Sense Disambiguation.” In *Proceedings of COLING-ACL-2006*, pp. 89–96.

Chapelle, O., Schölkopf, B., and Zien, A. (2006). *Semi-Supervised Learning*, Vol. 2. MIT press Cambridge.

Daumé, H. I. (2007). "Frustratingly Easy Domain Adaptation." In *Proceedings of ACL-2007*, pp. 256–263.

Jiang, J. and Zhai, C. (2007). "Instance Weighting for Domain Adaptation in NLP." In *Proceedings of ACL-2007*, pp. 264–271.

Joachims, T. (1999). "Transductive Inference for Text Classification using Support Vector Machines." In *ICML*, Vol. 99, pp. 200–209.

神嶌敏弘 (2010). 転移学習. 人工知能学会誌, **25** (4), pp. 572–580.

Kanamori, T., Hido, S., and Sugiyama, M. (2009). "A Least-Squares Approach to Direct Importance Estimation." *The Journal of Machine Learning Research*, **10**, pp. 1391–1445.

古宮嘉那子, 奥村学 (2012). 語義曖昧性解消のための領域適応手法の決定木学習による自動選択. 自然言語処理, **19** (3), pp. 143–166.

古宮嘉那子, 小谷善行, 奥村学 (2013). 語義曖昧性解消の領域適応のための訓練事例集合の選択. 言語処理学会第 19 回年次大会, pp. C6–2.

Komiya, K. and Okumura, M. (2011). "Automatic Determination of a Domain Adaptation Method for Word Sense Disambiguation using Decision Tree Learning." In *Proceedings of IJCNLP-2011*, pp. 1107–1115.

Komiya, K. and Okumura, M. (2012). "Automatic Domain Adaptation for Word Sense Disambiguation Based on Comparison of Multiple Classifiers." In *Proceedings of PACLIC-2012*, pp. 75–85.

Maekawa, K. (2007). "Design of a Balanced Corpus of Contemporary Written Japanese." In *Symposium on Large-Scale Knowledge Resources (LKR2007)*, pp. 55–58.

森信介 (2012). 自然言語処理における分野適応. 人工知能学会誌, **27** (4), pp. 365–372.

Okazaki, N. (2009). "Classias: A Collection of Machine-Learning Algorithms for Classification.".

Okumura, M., Shirai, K., Komiya, K., and Yokono, H. (2010). "SemEval-2010 Task: Japanese WSD." In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pp. 69–74.

Rosenstein, M. T., Marx, Z., Kaelbling, L. P., and Dietterich, T. G. (2005). "To Transfer or Not to Transfer." In *Proceedings of the NIPS 2005 Workshop on Inductive Transfer: 10 Years Later*.

齋木陽介, 高村大也, 奥村学 (2008). 文の感情極性判定における事例重み付けによるドメイン適応. 情報処理学会第 184 回自然言語処理研究会, NL-184-10.

新納浩幸, 佐々木稔 (2013). k 近傍法とトピックモデルを利用した語義曖昧性解消の領域適応. 自然言語処理, **20** (5), pp. 707–726.

新納浩幸，佐々木稔 (2014). 共変量シフトの問題としての語義曖昧性解消の領域適応. 自然言語処理, **21** (1), pp. 61–79.

新納浩幸，國井慎也，佐々木稔 (2014). 語義曖昧性解消を対象とした領域固有のシソーラスの構築. 第 5 回コーパス日本語学ワークショップ, pp. 199–206.

Sogaard, A. (2013). *Semi-Supervised Learning and Domain Adaptation in Natural Language Processing*. Morgan & Claypool.

杉山将 (2006). 共変量シフト下での教師付き学習. 日本神経回路学会誌, **13** (3), pp. 111–118.

杉山将 (2010). 密度比に基づく機械学習の新たなアプローチ. 統計数理, **58** (2), pp. 141–155.

Sugiyama, M. and Kawanabe, M. (2011). *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press.

高村大也 (2010). 言語処理のための機械学習入門. コロナ社.

Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., and Sugiyama, M. (2011). "Relative Density-ratio Estimation for Robust Distribution Comparison." *Neural Computation*, **25** (5), pp. 1370–1370.

吉田拓夢，新納浩幸 (2014). 外れ値検出手法を利用した Misleading データの検出. 第 5 回コーパス日本語学ワークショップ, pp. 49–56.

略歴

新納　浩幸：1985 年東京工業大学理学部情報科学科卒業．1987 年同大学大学院理工学研究科情報科学専攻修士課程修了．同年富士ゼロックス，翌年松下電器を経て，1993 年 4 月茨城大学工学部システム工学科助手．1997 年 10 月同学科講師，2001 年 4 月同学科助教授，現在，茨城大学工学部情報工学科准教授．博士（工学）．機械学習や統計的手法による自然言語処理の研究に従事．言語処理学会，情報処理学会，人工知能学会 各会員．

佐々木　稔：1996 年徳島大学工学部知能情報工学科卒業．2001 年同大学大学院博士後期課程修了．博士（工学）．2001 年 12 月茨城大学工学部情報工学科助手．現在，茨城大学工学部情報工学科講師．機械学習や統計的手法による情報検索，自然言語処理等に関する研究に従事．言語処理学会，情報処理学会 各会員．

（2014 年 1 月 22 日　受付）
（2014 年 4 月 1 日　再受付）
（2014 年 5 月 13 日　再々受付）
（2014 年 6 月 30 日　採録）