情報処理学会研究報告
IPSJ SIG Technical Report

Vol.2010-NL-198 No.5
2010/9/17

語義曖昧性解消のための領域適応手法の自動選択

古宮　嘉那子[†1]　奥村　学[†2]

ソースドメインのデータによって分類器を作り，ターゲットドメインに適応することを領域適応といい，近年さまざまな手法が研究されている．本稿では，WSD（Word Sense Disambiguation，語義曖昧性解消）について領域適応を行った場合，ソースデータとターゲットデータの性質により，最も効果的な領域適応手法が異なることを示す．また，決定木学習を用いてそれらの性質から，最も効果的な領域適応手法を自動的に選択する手法について述べる．それぞれ自動的に選択された手法を用いて領域適応を行うことで，もともとの手法を一括的に使った時に比べ，WSD の平均の正解率が有意に向上した．

Automatic Selection of Domain Adaptation Method for WSD

Kanako Komiya[†1] and Manabu Okumura[†2]

Domain adaptation; to adapt the classifier developed from source data to target data has been studied intensively in recent years. In this paper the authors show that when domain adaptation for WSD (word sense disambiguation) was performed, the most effective domain adaptation method varies according to the properties of the source data and target data. This paper also describes the way to select the most effective method for domain adaptation depending on these properties using decision tree learning. The average accuracy of WSD showed significant improvement when the domain adaptation method which is selected automatically was used respectively, compared to when the original methods were used collectively.

†1 東京農工大学　工学研究院
Tokyo University of Agriculture and Technology, Institute of Engineering
†2 東京工業大学　精密工学研究所
Tokyo Institute of Technology, Precision and Intelligence Laboratory

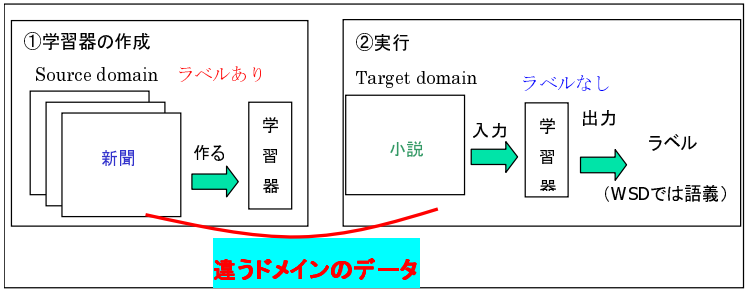

図 1　領域適応時の機械学習

1. はじめに

通常，機械学習とは，新聞で新聞用の学習器を作るなど，データ A を用いてデータ A 用の学習器を作るものであった．しかし一方，データ B についての答えを知りたいのに，データ A にしかラベルがついていないことがあり得る．このとき，データ A（Source domain）によって分類器を作り，データ B （Target domain）に適応することを考える．これが領域適応であり，さまざまな手法が研究されている．**図 1** は Source domain を新聞，Target domain を小説にした際の領域適応の様子を示している．

本稿では，WSD（Word Sense Disambiguation，語義曖昧性解消）について，領域適応を行った場合，ソースデータとターゲットデータの性質により，最も効果的な領域適応手法が異なることを示す．また，本稿では，決定木学習を用いてソースデータとターゲットデータの性質から，最も効果的な領域適応手法を自動的に選択する手法について述べる．対象の単語，ソースデータ，ターゲットデータの三つが一意に決まるときの領域適応を 1 ケースとして数えるとする．このケースごとに適切な領域適応手法を自動的に選択し，その手法を適宜用いて領域適応を行うことで，どれかひとつの手法を一括的に使った時に比べ，WSD の平均の正解率が有意に向上した．

2. 関連研究

領域適応は，学習に使用する情報により，supervised，semi-supervised，unsupervised の三種に分けられる．まず supervised の領域適応は，ソースドメインからの多量なラベル

つきデータに加え，ターゲットドメインからの少量のラベルつきデータを用いて学習を行うもので，ターゲットドメインだけを訓練事例とした場合よりも改良することを目指すものである．次に semi-supervised の領域適応は，ソースドメインからの多量なラベルつきデータに加え，ターゲットドメインからの多量なラベルなしデータを利用するものである．こうすることで，ソースドメインだけを訓練事例とした場合よりも改良することを目指している．また，最後の unsupervised の領域適応は，ソースドメインで学習後，ターゲットドメインで実行するものである．

領域適応の研究は自然言語処理の分野の内外においてさまざまな手法でなされており，ここではその一部を紹介する．まず，(Chan and Ng (2006))[1] は，EM アルゴリズムによる Prior（意味の割合）推定により WSD の領域適応を行っている．また，筆者らは（Chan and Ng (2007))[2] でも，EM アルゴリズムによる Prior 推定を行っているが，ここでは Active-learning により用例をターゲットドメインから足す supervised の領域適応を行っている．Count-merging により重要文に重みをつけてから，推定を行う．

また，(Daumé III(2007))[3] はシーケンスラベリングを例に supervised の領域適応を行っている．具体的には素性空間を「ソースのみ」「ターゲットのみ」「両方」の三倍にして実験を行うというもので，さまざまな supervised の領域適応に併用できる手法である．利点としては，上記の併用可能性に加え，実装が簡単で処理が早いこと，マルチドメインに拡張が簡単（ドメイン数+1 倍にすればよい）であることが挙げられる．

さらに，(Daumé III et al. (2010))[4] は文献 3) を semi-supervised のために拡張した．この手法はなぜ有効であるかまだ解き明かされていないが，拡張前の利点を引き継いだ上，ラベルなしのターゲットデータを利用することでよりよい性能が得られる手法である．

（Agirre and Lacalle(2008))[5] は，semi-supervised の WSD の領域適応を行った．具体的には，ソースドメインのデータに，ターゲットドメインのラベルなしデータを足して行列を作り，特異値分解（SVD）により素性圧縮をして分類器を学習するものである．また筆者らは（Agirre and Lacalle(2009))[6] において同様の手法で supervised の領域適応を行っている．文献 5) との違いは，SVD をかける行列に，大量のラベルなしデータではなく，少量のラベルつきデータを使用することである．

さらに（Jiang and Zhai (2007))[7] は領域適応を行う際，用例の重み付けにより性能が向上することを示した．これは supervised および semi-supervised な手法の両方に使用できる手法である．また，ミスリーディングなソースドメインのデータを削除することも試したが，これらの手法は個々では有効であるものの，組み合わせると全てのソースドメインのデータを利用した方がよいと結論づけている．

本稿では，新しい領域適応手法を提案し，他手法と比較するのに用いる．この提案手法が active-learning とも深い関わりがあるため，ここで active-learning とも深い関わりがあるため，ここで active-learning の手法についても紹介する．active-learning とは，まずごく少量の訓練事例により学習器を作成して，ラベルなしデータを使用し，その結果を用いて用例を選んでラベル付けし，新たに訓練事例に加えることを繰り返す手法で，より少量のデータで効率的に性能の良い学習器を作ることを目指すものである．作成した学習器にラベルなしデータを予測する際にもっとも不確かなものから足していく手法が一般的である．

これに対し（Sassano (2002))[8] は，大量のデータに対して active-learning を行う際，データをランダムサンプリングしておいて，小さなプールを作り，そのプールから足すデータを選び，サポートベクターの増加が収束してきたころに新たなもっと大きなプールを用意するという手法が有効であることを示した．また（Zhu et al.(2008))[9] は，不確かさの指標に密度の指標を併用して active-learning に使うデータを選ぶと，外れ値を選ばなくなるため性能が向上することを示した．

また，本稿では，ソースデータとターゲットデータの性質をもとに領域適応に用いる手法を選択する手法を述べる．これに関連した研究として (張本ら (2010))[10] や (Asch and Daelemans (2010))[11] がある．文献 10) は，構文解析のタスクにおいて，分野間距離をはかり，より適切なコーパスを利用して領域適応を行えるようにするものである．また，文献 11) は，タグ付けのタスクにおいて，アノテートされていないコーパスに自動タグ付けしたデータを用いて，ソースドメインとターゲットドメインの類似度から性能を予測できることを示したものである．これらの研究では領域間の距離から，ソースドメインとして利用できるコーパスを利用するという立場をとっているが，本研究では領域間の距離などの性質から，手法を選択するという立場をとる．

3. WSD のための領域適応手法

WSD のための領域適応として，以下に示す四通りを行った．

- `Target Only`：ソースデータを用いずにアノテーションした少量のターゲットデータだけで学習する．
- `Random sampling`：ターゲットデータからランダムに選んだ用例をアノテーションし，ソースデータに追加して学習を行う．

Vol.2010-NL-198 No.5
2010/9/17

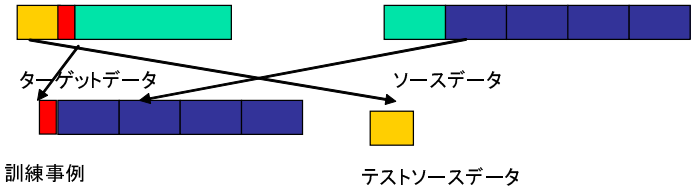

図 2 領域適応の五分割交差検定

- `Active Learning`：Active Learning のように，ソースデータによる分類器にターゲットデータを実行した際，より不確かな用例を選ぶ．これをアノテーションしてソースデータに追加し，学習を行う．
- 追加時のフィルタリング：提案手法（後述）．フィルタリングによりソースデータに似ていないターゲットデータを選び，さらにその集合からランダムに選択する．これをアノテーションしてソースデータに追加し，学習を行う．

なお，追加するターゲットデータは常に 10 件とした．さらに，学習器としてはマルチクラス対応の SVM（libsvm）を使用した．カーネルは予備実験の結果，線形カーネルが最も高い正解率を示したため，これを採用した．また，学習の素性には，問題の単語の前後二単語について形態素，品詞（＋細分類），係り受け，分類語彙表の値を使用した．

また，実験は五分割交差検定を用いた．この際，ソースデータの 4/5（ソースデータの青い部分）に加え，ターゲットデータの 4/5（ターゲットデータの緑の部分と赤い部分）から l 件（= 10 件，赤い部分）を訓練事例とする．テストデータは，ターゲットデータの残りの 1/5（黄色い部分）である．この様子を**図 2** に示す．

また参考として，ラベルつきターゲットデータが手に入ったと仮定して，supervised の学習を行った self と，ソースデータだけで学習を行った theOther についても実験を行った．以下**表 1** にソースデータとして Yahoo！知恵袋を，ターゲットデータとして白書を利用したときの領域適応の実験結果を，以下**表 2** にソースデータとして白書を，ターゲットデータとして Yahoo！知恵袋を利用したときの領域適応の実験結果を示す．

これらの結果は全ケースの平均であり，正解率の順に並べ替えられている（下へいくほど正解率が高い）．これらの表から使用するコーパスによって効果の出る手法が異なっていることが分かる．

表 1 ソースデータとして Yahoo！知恵袋を，ターゲットデータとして白書を利用したときの領域適応の実験結果

領域適応手法	正解率
theOther	79.65%
Active Learning	83.75%
追加時のフィルタリング	83.90%
Random sampling	85.40%
Target Only	88.20%
self	95.97%

表 2 ソースデータとして白書を，ターゲットデータとして Yahoo！知恵袋を利用したときの領域適応の実験結果

領域適応手法	正解率
Target Only	77.74%
Random sampling	83.86%
theOther	83.35%
Active Learning	84.20%
追加時のフィルタリング	84.59%
self	91.65%

さらに調べてみると，対象の単語やソース/ターゲットデータが異なれば，効果の出る手法が異なっていることが分かった．詳しく見てみると，この傾向はおおむね Target Only とそれ以外のソースデータにターゲットデータをなんとかして足す手法（以下，Adding と呼ぶ）に分けられ，ソースデータとターゲットデータの語義分布が似ていない場合には Target Only が有効であり，似ている場合には Adding が有効であることが分析できた．

4. 用例の類似度によるフィルタリングを利用した領域適応

本章では WSD の領域適応のための提案手法，用例の類似度によるフィルタリングを利用した手法について述べる．本手法は，学習器作成の際，ソースデータ（ソースドメインの用例）に加えて少量のターゲットデータ（ターゲットドメインの用例）をラベル付けして訓練事例とする supervised の領域適応である．ソースデータとターゲットデータの間の類似度を用いてフィルタリングを行うことで，訓練事例として利用するデータを選別する．

追加時のフィルタリングでは以下の手順を取る．

(1) ターゲットデータ $\forall t_i \in T$ について，全ソースデータ $\forall s_i \in S$ とのコサイン類似度

$sim_{i,j}$ を計算する．

(2) ターゲットデータ $\forall t_i \in T$ について，それぞれ最も自身と近いソースデータ $s_{i,nearest}$ を特定する．

(3) ターゲットデータ $\forall t_i \in T$ について $s_{i,nearest}$ との類似度 $sim_{i,nearest}$ をもとに，訓練事例として選ぶデータの候補かどうかを判定する．

この判定方法として，$sim_{i,nearest}$ がより小さな用例から上位 k 用例を候補とした．またここで，k=100 とした．

(4) 候補の中から l 件のターゲットデータをランダムに選び，ソースデータと併せて訓練事例として分類器を作成する．

ここで，l=10 とした．また，ここでランダムサンプリングを行うのは，追加データの冗長性を減らすためである．

5. 領域適応手法の自動選択

対象の単語，ソースデータ，ターゲットデータの三つが一意に決まるときの領域適応を 1 ケースとして数えるとする．このケースごとに適切な領域適応手法を自動的に選択し，その手法を適宜用いて領域適応を行えば，どれかひとつの手法を用いるよりも，WSD の性能が向上することが予想される．このため，決定木学習を用いて，領域適応手法の自動選択を行う．決定木作成アルゴリズムには C4.5 を利用し，二分決定木を作成した．また，五分割交差検定を行った．

決定木はケースごとに，カイ二乗検定を行い Target Only と Adding の三種法（Random sampling，Active Learning，追加時のフィルタリング）のうちもっとも性能が良かった手法のふたつを比較して，その差が有意であるかを見た．有意であるものだけを訓練事例として，Target Only，Adding の二種類に分類する．これは，カイ二乗検定によって有意でないものは「偶然差が出たもの」ということになり，Target Only でも Adding でもどちらでもよいのであって，決定木作成においてミスリーディングな用例になり得るため，訓練事例に入れないという考えに基づいている．また，有意差のあるケースは有意差がないケースよりも WSD の正解率に差があるため，有意差のあるケースを正しく分類する決定木を学習することで，全体的な WSD の正解率をあげることができる．

また，決定木には以下の 24 種類の合計 40 の素性を利用した．

(1) the Other Target Only の正解率：ソースデータで分類器を作り，10 件のターゲットデータをアノテーションしたもので評価した正解率

(2) Target Only の正解率：10 件のターゲットデータをアノテーションし，Leave One Out 法で評価を行った際の正解率

(3) 前ふたつの正解率の比 ：the Other Target Only の正解率/ Target Only の正解率

(4) ソースデータ件数

(5) ターゲットデータ件数

(6) ソースデータ件数/ターゲットデータ件数

(7) 10 件のターゲットデータの 語義数

(8) ソースデータの語義数

(9) 辞書中の語義数

(10) ソースデータの MFS（Most Frequent Sense:データ中最も頻出する語義）の件数

(11) ソースデータの MFS のパーセンテージ

(12) ターゲット 10 件の MFS のソース中でのパーセンテージ

(13) MFS の語義番号がソースとターゲットで同じか

(14) ターゲット 10 件の MFS の件数

(15) ターゲット 10 件の MFS のパーセンテージ

(16) ソースの MFS のターゲット 10 件中のパーセンテージ

(17) 全件の 4/5 のソースデータと 10 件のターゲットデータの語義ラベルの JS 距離

(18) ソースデータ全件とターゲットデータ全件の素性ごとの JS 距離を足したもの

(19) ソースデータ全件とターゲットデータ全件の WSD の素性ごとの JS 距離：17 種類（WSD に利用した素性数による）

(20) ソースデータ全件とターゲットデータ全件の WSD ラベル以外の素性をひとつの単位としたときの JS 距離

(21) 新語義の数：10 件のターゲットデータになくて，ソースデータ全件にある語義の数

(22) ソースデータ全件と 10 件のターゲットデータで共通する語義数

(23) ソースデータ全件と 10 件のターゲットデータで共通する語義数の，10 件のターゲットデータ中のパーセンテージ

(24) ソースデータ全件と 10 件のターゲットデータで共通する語義数の，ソースデータ全件中のパーセンテージ

表 3 144 のケースのジャンルごとのデータ件数

コーパスの種類	最小	最多	平均
BCCWJ 白書	58	7610	2074.50
Yahoo！知恵袋	82	13976	2300.43
RWS　新聞	50	374	164.46

実験にあたり，Yahoo! 知恵袋のデータと，現代日本語書き言葉均衡コーパス（BCCWJ コーパス）の白書のデータ，また RWC コーパスの毎日新聞コーパスを利用した．また，領域適応としては両方向について実験を行ったため，全部で 6 方向の領域適応を行った．コーパス中に 50 用例に満たなかった単語は実験対象外としたため，最終的なケースの数は，白書と Yahoo! 知恵袋 : 24　白書と新聞 : 22　Yahoo! 知恵袋と新聞 : 26 であり，それぞれが両方向あるため，合計 144 のケース，28 単語となった．このうち，カイ二乗検定で Target Only と Adding の三種法のうちもっとも性能が良かった手法のふたつを比較して，その差が有意であると判定され，訓練事例に利用したケースは，白書から Yahoo! 知恵袋への領域適応 : 17　白書から新聞への領域適応 : 10　Yahoo! 知恵袋から新聞への領域適応 : 13　Yahoo! 知恵袋から白書への領域適応 : 14　新聞から白書への領域適応 : 15　新聞から Yahoo! 知恵袋への領域適応 : 17 であり，合計 86 ケース，25 単語となった．なお，テストには全 144 ケースを利用し，カイ二乗検定などにより選択は行っていない．

語義数ごとの内訳は，2 語義 : 場合，自分，3 語義 : 事業，情報，地方，社会，思う，4 語義 : 子供，分かる，5 語義 : 含む，使う，考える，6 語義 : 関係，時間，一般，技術，現在，作る，7 語義 : 今，8 語義 : 前，10 語義 : 持つ，11 語義 : 進む，12 語義 : 見る，14 語義 : 入る，17 語義 : 言う，21 語義 : 出す，22 語義 : 手，出る である．また，144 のケースのそれぞれのジャンルにおける最小，最大，平均データ数は**表 3** のようになっている．

コーパスごとの特徴として，白書はデータ数が少なめで語義数最小であり，Yahoo!知恵袋はデータ数と語義数最多，新聞はデータ数が最小で語義数多めとなっている．

6. 結　　果

表 4 に，もともとの手法を一括的に用いた際の WSD の平均正解率を示す．なお，144 のケースは合計 232116 用例となり，平均正解率はこれら全件を等しい重みとしたときの平均である．

表 5 に，選択した手法を利用した際の WSD の平均正解率を示す．なお，選択の仕方は，人手による選択，決定木学習による選択の二通りである．また，Target Only と共に Adding として用いる領域適応手法は，Random sampling，Active Learning，追加時のフィルタリングである．

表 4 個別の手法を用いた際の WSD の平均正解率

領域適応手法	WSD の平均正解率
Target Only	81.23 %
Random sampling	80.28 %
Active Learning	79.27 %
追加時のフィルタリング	79.42 %

表 5 選択した手法を利用した際の WSD の平均正解率

Adding のときの領域適応手法	人手	決定木学習
Random sampling	85.21 %	84.10 %
Active Learning	85.45 %	84.34 %
追加時のフィルタリング	85.29 %	84.10 %

表 4 のどの平均正解率よりも，表 5 の決定木学習を用いた場合の平均正解率の方が高い値を示すことから，決定木を利用して適切な領域適応手法を利用した方が，個々の領域適応手法を使った時よりも正解率は上がったことが分かる．なお，カイ二乗検定により，十分な有意差が認められた．また，表 5 により決定木学習を用いて使用する領域適応手法を選択する場合には，Active Learning が最も良いことが分かる．

7. 考　　察

決定木の正解率は現在 69.64%である．(Target Only でも Adding でも WSD の正解率が同じ時には Adding を正解とした場合）そのときの木の生成に特に貢献した素性と素性値のうち，説明できるものを以下に述べる．

まず,「the Other Target Only の正解率/ Target Only」の正解率が低い値であれば，Target Only が割り当てられた．10 件のターゲットデータをアノテーションし，Leave One Out 法で評価を行った際の正解率のほうが，ソースデータで分類器を作り，10 件のターゲットデータをアノテーションしたもので評価した正解率よりもずっと高いということなので，その予測があたったことになる．

また,「係り受けの JS 距離」が大きいときや,「前の単語の形態素に相当する JS 距離」,「二つ後の単語の形態素に相当する JS 距離」が大きいときも，Target Only が割り当てられた．これらの JS 距離が大きいのは，素性の分布が異なっていることを示すため，分類の鍵となる素性の分布が遠く，ソースデータが十分に似ていないため，利用せずに Target Only を利用した方がよいためであると思われる．またこのことから，係り受けや，ひとつ前の単語の形態素，二つ後の単語の形態素が語義を予測するのに重要視されていることが分かる．

さらに，ソースデータ全件と 10 件のターゲットデータで共通する語義数の，10 件のターゲットデータ中のパーセンテージが低い時に Target Only が選ばれた．これも二つのドメインが遠いときである．

決定木はまず Target Only をより分けて，残ったものに Adding を割り当てる木となっていた．そのため，上記の特徴に当てはまらないケースにはおもに Adding が割り当てられた．

8. ま と め

本稿では，WSD（Word Sense Disambiguation，語義曖昧性解消）について，領域適応を行った場合，ソースデータとターゲットデータのデータの性質により，最も効果的な領域適応手法が異なることを示した．また，決定木学習を用いてソースデータとターゲットデータの性質から，最も効果的な領域適応手法を自動的に選択する手法について述べた．対象の単語，ソースデータ，ターゲットデータの三つが一意に決まるときの領域適応を 1 ケースとして数えるとする．決定木学習ではこのケースごとに，ターゲットデータ 10 件にアノテーションをして訓練事例とする Target Only と，それにソースデータを足したものを訓練事例とする Adding の二種類に分類した．この際，カイ二乗検定を用いて有意差が認められるケースのみを決定木学習の訓練事例とした．ケースごとに自動的に選択された手法を用いて領域適応を行うことで，もともとの手法を一括的に使った時に比べ，WSD の正解率が有意に向上した．

なお，Adding の手法としては，ターゲットデータからランダムに選んだ用例をアノテーションし，ソースデータに追加して学習を行う Random sampling，Active Learning のように，ソースデータによる分類器にターゲットデータを実行した際，より不確かな用例を選び，これをアノテーションしてソースデータに追加し，学習を行う Active Learning，フィルタリングによりソースデータに似ていないターゲットデータを選び，さらにその集合からランダムに選択し，これをアノテーションしてソースデータに追加し，学習を行うという提案手法の追加時のフィルタリングの三種類を試したが，このうちもっとも性能が良かったのは Active Learning であった．

参 考 文 献

1) Chan, Y. S. and Ng, H. T.: Estimating Class Priors in Domain Adaptation for Word Sense Disambiguation, *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp89-96 (2006).
2) Chan, Y. S. and Ng, H. T.: Domain Adaptation with Active Learning for Word Sense Disambiguation, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp49-56 (2007).
3) Daumé III, H.: Frustratingly Easy Domain Adaptation, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp 256-263 (2007).
4) Daumé III, H., Kumar, A. and Saha, A.: Frustratingly Easy Semi-Supervised Domain Adaptation, *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, ACL 2010*, pp 23-59 (2010).
5) Agirre, E. and de Lacalle, O. L.: On Robustness and Domain Adaptation using SVD for Word Sense Disambiguation, , *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 17-24 (2008).
6) Agirre, E. and de Lacalle, O. L.: Supervised Domain Adaption for WSD, *Proceedings of the 12th Conference of the European Chapter of the Association of Computational Linguistics*, pp42-50 (2009).
7) Jiang, J. and Zhai, C.: Instance Weighting for Domain Adaptation in NLP, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp 264-271 (2007).
8) Sassano, M.: An Empirical Study of Active Learning with Support Vector Machines for Japanese Word Segmentation, *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics*, pp. 505-512 (2002).
9) Zhu, J., Wang, H., Yao, T. and Tsou, B. K.: Active Learning with Sampling by Uncertainty and Density for Word Sense Disambiguation and Text Classification, *Proceedings of the 22nd International Conference on Computational Linguistics*, pp 1137-1144 (2008).
10) 張本佳子, 宮尾祐介, 辻井潤一: 構文解析の分野適応における精度低下要因の分析及び分野間距離の測定手法, 言語処理学会　第 16 回年次大会発表論文集, pp 27-30 (2010).
11) Asch, V. V. and Daelemans, W.: Using Domain Similarity for Performance Estimation, *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, ACL 2010*, pp 31-36 (2010).
12) 古宮嘉那子，奥村学: 用例の類似度によるフィルタリングを利用した領域適応，特定領域研究「日本語コーパス」平成 21 年度公開ワークショップ予稿集，pp235-242 (2010).