# 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選択 —三手法からの決定—

古宮　嘉那子（東京農工大学　工学研究院）†
奥村　学（東京工業大学　精密工学研究所）

# Determination of a Domain Adaptation Method for Word Sense Disambiguation Using Decision Tree Learning - Determination from Three Methods-

Kanako Komiya (Institution of Engineering, Tokyo University of Agriculture and Technology)
Manabu Okumura (Precision and Intelligence Laboratory, Tokyo Institution of Technology)

## １．はじめに

語義曖昧性解消（Word Sense Disambiguation，WSD）について領域適応を行った場合，最も効果的な領域適応手法は，ソースドメインのデータ（ソースデータ）とターゲットドメインのデータ（ターゲットデータ）の性質により異なる (Komiya and Okumura 2011)．WSD の対象単語タイプ，ソースデータ，ターゲットデータの三つ組を 1 ケースとして数えるとする．本稿では，このケースごとに，データの性質から，最も効果的な領域適応手法を，決定木学習を用いて自動的に選択する手法について述べるとともに，どのような性質が効果的な領域適応手法の決定に影響を与えたかについて考察する．また本稿では，3 つ以上の n 個の領域適応手法から選択するために, pairwise 方式で nC2 通りの二分決定木をつくり，最終的にそれらを統合することで，ひとつのケースにつきひとつの領域適応手法を決定する方法を，三手法からの選択を例にとって述べる．

## ２．関連研究

領域適応の研究は様々な分野で研究が行われているが, 本稿に最も近い研究は, (McClosky, Charniak, and Johnson 2010) である．この研究では, 多様なドメインからなる文書を構文解析する際，最も良いモデルは異なるという問題に注目している．彼らは様々な混合モデルによる構文解析の正解率を回帰分析で予測し，それぞれのターゲットデータに対して，最も高い正解率を出すと予測されたモデルを利用して構文解析を行っている．本研究との最も大きな違いは，対象のタスクが構文解析ではなく語彙曖昧性解消である点である．また，彼らは多様なドメインからなる文書があることを想定しているが，我々は想定していない．本研究では決定木学習を用いることで，どのような性質が最適な領域適応の決定に影響を与えるのかについて考察する．

また，二手法からの選択に関しては，(Komiya and Okumura 2011)にて既に述べたため，本稿では三手法以上からの選択にした場合にどのような工夫が必要かという部分を中心に述べる．

## ３．領域適応手法の自動選択

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．決定木学習を用いるこ

† kkomiya@cc.tuat.ac.jp

とで，どのような性質が最適な領域適応手法の決定に影響を与えるのかを明示的に示すことができる．

### ３．１　WSD のための領域適応手法

WSD のための領域適応手法として，本研究では以下に示す三つを用いる．したがって，pairwise 方式で三つ（Target Only と Random Sampling，Target Only とフィルタリングによる削除，Random Sampling とフィルタリングによる削除）の二分決定木をつくり，最終的にそれらを統合することで，ケースごとに領域適応手法を決定する．

- Target Only (TO)：ソースデータを用いず，ランダムに選んだ少量のターゲットデータにラベル付けしたものだけを訓練事例にする．
- Random Sampling (RS)：ランダムに選んだ少量のターゲットデータの単語トークンにラベル付けしたものとソースデータの両方を訓練事例にする．
- フィルタリングによる削除(FD)：ランダムに選んだ少量のターゲットデータの単語トークンにラベル付けしたものとソースデータの両方を訓練事例にする．このときソースデータは，フィルタリングによりターゲットデータにある一定の閾値以上似ているデータだけを用いる．

なお，追加するターゲットデータのトークン数は常に 10 件とした．また，WSD の分類器としてはマルチクラス対応の SVM（libsvm）の線形カーネルを使用した．また，WSD の学習の素性には，WSD の対象単語の前後二語までの形態素の表記，WSD の対象単語の前後二語までの品詞，WSD の対象単語の前後二語までの品詞の細分類（分類語彙表に(国立国語研究所 1964)よる），WSD の対象単語の前後二語までの分類コード，係り受けを用いた．

### ３．２　決定木学習のラベル

ケースごとに，最も WSD の正解率がよかった手法によって，領域適応の手法名のラベルか，Same ラベルをつけた．決定木は，ケースごとにソースデータとターゲットデータの性質から，二つのうちどちらの手法を使って領域適応するべきかを判定している．作成する三つの決定木のうちのひとつ，Target Only か Random Sampling を選択する決定木では以下のように付与する．

- Target Only : Random Sampling より Target Only を使用した方が WSD の正解率が良いケース
- Random Sampling : Target Only より Random Sampling を使用した方が WSD の正解率が良いケース
- Same : Target Only と Random Sampling のどちらを使っても WSD の正解率に差がないケース
-

### ３．３　決定木学習の素性

最適な領域適応手法はソースデータとターゲットデータの分布や距離などの性質によって異なると考えられるため，それぞれの決定木に 24 種類，合計 40 の素性を利用した．これらのうちには，すべての領域適応手法において共通して使用している，ランダムに選んだターゲットデータの 10 トークンを使用して Leave-one-out 法で求めた領域適応手法のシミュレーションの正解率や，その比率，ターゲットデータやソースデータの件数や，ランダムに選び人手でラベル付けしたターゲットデータの 10 トークン中の最も頻度の高い語義

に関する情報，また WSD に使用した素性の JS 距離などが含まれている．

## ４．実験データ

実験には，現代日本語書き言葉均衡コーパス（BCCWJ）(Maekawa 2008) の白書のデータと Yahoo! 知恵袋のデータ，また RWC コーパスの毎日新聞コーパス (Hashida, Isahara,Tokunaga, Hashimoto, Ogino, and Kashino 1998) の三つのデータを利用し，ソースデータとターゲットデータを変えることで，全部で 6 通りの領域適応を行った．これらのデータには岩波国語辞典(西尾，岩淵，水谷 1994) の語義が付与されている．これらのコーパス中の多義語のうち，ソースデータおよびターゲットデータ中にともに 50 トークン以上存在する単語を実験対象とした．対象単語は「場合」，「自分」，「事業」，「情報」，「地方」，「社会」，「思う」，「子供」，「分かる」，「考える」，「含む」，「使う」，「技術」，「関係」，「時間」，「一般」，「現在」，「作る」，「今」，「前」，「持つ」，「進む」，「見る」，「入る」，「言う」，「出す」，「手」，「出る」 である．

## ５．決定木学習におけるラベル付きデータの作成方法と学習方法

決定木学習におけるデータのラベル付けの際は，決定木で判定する領域適応手法二手法の WSD の正解率を比較してカイ二乗検定を行い，有意差がないものに Same をつけ，領域適応手法名のラベルを付与した．なお，カイ二乗検定の有意水準は 0.05 を利用した．

また，決定木学習において Same が付与されたケースを訓練事例から削除して決定木で判定する領域適応手法二手法の 2 値分類の決定木学習を行った．なお，テストには全ケースを利用した．

さらに，決定木学習の際は全てのケースに同等の重みがあるとして決定木学習を行った．

領域適応手法決定のための決定木作成アルゴリズムには C4.5 (Quinlan 1993) 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．決定木作成の枝刈りの閾値は訓練事例の 1/4 を開発用データとした予備実験により最適化した．

### ５.１　決定木の統合

決定木の統合は，以下のように行った．pairwise の性質上，三つの決定木が三つとも同じ方法がよいと答えることはなく，答えが 2:1 に分かれるか，三つ巴になるはずである．

このうち，2:1 に分かれるときは，かならず 2 つの決定木が出した答えが理論的に一番良くなるため，その答えを選択すればよい．手法 1> 手法 2 のとき手法 1 のほうがよい手法であるとすると，例えば， Target Only>Random Sampling かつ，フィルタリングによる削除>Random Sampling かつ Target Only> フィルタリングによる削除であれば，Target Only> フィルタリングによる削除>Random Sampling なので，Target Only を選択する．

次に，三つ巴のときには，事例が割りつけられた葉についている確率を比較し，一番高い確率のところに割り付けた．確率は，「学習時にその葉に割りつけられた最も多いケース数/学習時に，その葉に割りつけられた全ケース数」として計算した．たとえば，テストデータが，実行時に「学習時に，Target Only が 1 件，Random Sampling が 2 件割り当てられた葉」に割り当てられた場合，そのテストデータは 2/3 の確率で Random Sampling となる．三つ巴の場合には，この確率で比較し，最も高い確率の手法を割り当てた．

三つ巴のときに，ふたつの決定木で割りつけられた葉の確率が同率一位である場合には，Random Sampling>Target Only かつフィルタリングによる削除>Random Sampling なら，フィルタリングによる削除>Random Sampling>Target Only なのでフィルタリングによる削除を選択，というように論理的に選択した．

また，三つ巴でどれも確率が等しい時など，上記のルールを利用してもどうしても領域適応手法が選べない時には，一括的に領域適応を行ったときに正解率が高い順，つまり，フィルタリングによる削除，Target Only，Random Sampling の順で割りつけた．

## ６．結果

表 1 に，WSD の平均正解率の比較を示す．なお，144 のケースには合計 232116 語義曖昧性解消の対象単語トークンが含まれており，それらのマイクロ平均である．また，人手による選択は，決定木学習を用いる代わりに，ラベルとなっているふたつの領域適応のうち，WSD の正解率の高い領域適応手法をケースごとに人手で選択して，WSD の平均正解率を求めた値であり，upper bound である．

決定木学習を用いて選択した手法を利用した際の WSD の平均正解率は 83.52%であり，個別の手法を用いた際の最高の正解率，フィルタリングによる削除の 82.27%よりも正解率が高いため，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が上がることが分かる．またこのとき，カイ二乗検定により十分な有意差が認められた．

表 1　WSD の平均正解率の比較

| 領域適応手法 | WSD の平均正解率 |
|---|---|
| Target Only | 81.23% |
| Random Sampling | 80.28% |
| フィルタリングによる削除 | 82.27% |
| 決定木により選択された領域適応手法 | 83.52% |
| 人手により選択された領域適応手法 | 85.87% |

## ７．考察

五分割交差検定の五回の検定のうち，最も高い正解率だった決定木を付録として示し，生成に特に貢献した素性と素性値について以下に述べる．まず，Target Only と Random Sampling の決定木のルートノードでは，「ふたつの正解率の比=0.70 以上」が no のとき Target Only が割り当てられた．これは「the Other のシミュレーションの正解率/ Target Only のシミュレーションの正解率」の割合が 0.70 以下であれば，Target Only が割り当てられたということである．つまり，10 件のターゲットデータにラベル付けし，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学習し，10 件のターゲットデータにラベル付けしたもので評価した正解率よりも高いときには Target Only が割り当てられたということに等しい．このことから，10 件のラベル付けしたターゲットデータによるシミュレーションの予測が，最適な領域適応の手法を予想する強力な手がかりになることが分かる．

また，Random Sampling とフィルタリングの削除の決定木のルートノードでは，「ソースデータ件数/ターゲットデータに一定以上似ているソースデータ件数=186.85 以上」のときフィルタリングの削除が割り当てられた．フィルタリングの削除は，ターゲットデータに閾値以上似たソースデータだけを訓練事例に利用する手法であるため，ターゲットデータに閾値以上似ていないソースデータ件数が多量にあるときには，ソースデータ全件を利用せず，ターゲットデータに似ているデータだけを利用すればよいことが分かる．このことから，ターゲットデータに十分似ていないデータを足しすぎると，誤った学習が行われてしまうことが推察できる．

また，Target Only とフィルタリングの削除の決定木のルートノードでは，「ターゲットデータ 10 件の MFS の，ターゲットデータに閾値以上似たソースデータ中のパーセンテージ=12.58 以下」である場合に，Target Only が割り当てられた．このことにより，ターゲットデータ 10 件中に最頻出する語義が，フィルタリングの削除の訓練事例として利用される，

「ターゲットデータに一定以上似ているソースデータ」に少ない時には，Target Only を用いた方がよいことが分かる．このことから，二つのデータのラベルが似ていないときは，ソースデータから訓練事例を一切足すことなく，ターゲットデータだけで学習した方がよいと考えられる．

## ８．まとめ

語義曖昧性解消（WSD; Word Sense Disambiguation）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．そのため本稿では，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，作成した決定木について考察した．

## 文　献


Hashida, K., Isahara, H., Tokunaga, T., Hashimoto, M., Ogino, S., and Kashino, W. (1998). The RWC text databases. In *Proceedings of The First International Conference on Language Resource and Evaluation*, pp. 457-461.

Komiya, K. Okumura, M. (2011). Automatic Determination of a Domain Adaptation Method for Word Sense Disambiguation Using Decision Tree Learning. In Proceedings of 5th International Joint Conference on Natural Language Processing, pp. 1107-1115.

Maekawa, K. (2008). Balanced Corpus of Contemporary Written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources (ALR)*, pp. 101-102.

McClosky, D., Charniak, E., and Johnson, M. (2010). Automatic Domain Adaptation for Parsing. In Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 28-36.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

国立国語研究所(1964). 分類語彙表．秀英出版．

西尾実，岩淵悦太郎，水谷静夫(1994). 岩波国語辞典第五版．岩波書店．


## 生成された決定木

上の枝が yes，下の枝が no に相当する．

```
        ┌──Target Only
    ┌──JS 距離（WSD の対象単語の一つ前の形態素）>=0.61
    │   └──Random Sampling
ふたつの正解率の比>=0.70
    └──Target Only
```

図 1　Target Only と Random Sampling の決定木

```
    ┌──フィルタリングによる削除
ソースデータ件数/ターゲットデータに一定以上似ているソースデータ件数>=186.85　|
    ┌──フィルタリングによる削除
    └──JS 距離（WSD の対象単語の二つ前の形態素）>=0.74
        └──Random Sampling
```

図 2　Random Sampling とフィルタリングの削除の決定木

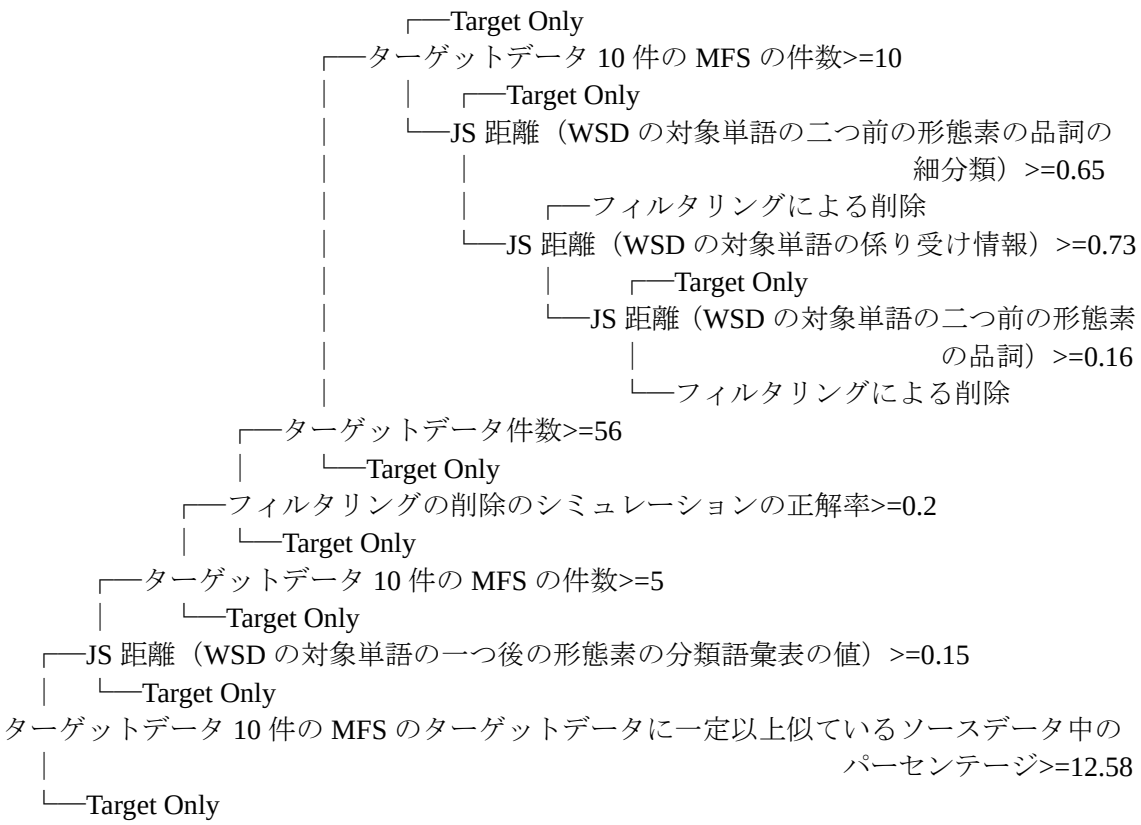


図 3　Target Only とフィルタリングの削除の決定木