SENSEVAL2 日本語翻訳タスクに向けて作成した 語義判別規則学習システム Ibaraki

新納 浩幸†

†茨城大学工学部
〒316-8511 茨城県日立市中成沢町 4-12-1

E-mail: †shinnou@dse.ibaraki.ac.jp

あらまし SENSEVAL2 日本語翻訳タスクに向けて語義判別規則学習システム Ibaraki を作成した．当初，タスクの設定からラベルなし訓練データを用いた教師なし学習の手法が有効と判断し，その方向の戦略をたてたが，実現できなかった．結果的に単純な決定リストによる学習システムで終わってしまった．本報告では，教師なし学習手法を用いようとしたいきさつ，及び断念した理由を述べるとともに，作成したシステムとその評価，問題点などを述べる．コンテスト後， Ibaraki にラベルなし訓練データを用いた手法も組み入れた．この件についても簡単に簡単に報告する．

キーワード SENSEVAL2，翻訳タスク，教師なし学習，決定リスト

.

Ibaraki: learning system of WSD rules developed for SENSEVAL2 Japanese translation task

Hiroyuki SHINNOU†

†Faculty of Engineering, Ibaraki University
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8551 Japan

E-mail: †shinnou@dse.ibaraki.ac.jp

Abstract We made a learning system of word sense disambiguation rules, named Ibaraki, for SENSEVAL2 Japanese translation task. First, we estimated that an unsupervised learning method is efficient for this task, and planed to use it. However, we gave up using the method. As a result, our system was ended up a simple learning system using the decision list method. In this paper, we report why we estimate that an unsupervised learning method is efficient for this task, and why we gave up it. Moreover, we also evaluate our system made as a result. After the contest, we integrated an unsupervised learning technique into Ibaraki. We report about it briefly.

Key words SENSEVAL2, Translation Task, Unsupervised learning, Decision list

1. はじめに

SENSEVAL2 は語義判別のコンテスト形式の会議である．2001 年 7 月に ACL-01 に併設して行われた．SENSEVAL2 では日本語のタスクとして翻訳タスクと辞書タスクが設定された．筆者は翻訳タスクに Ibaraki と名付けたシステムで参加した．Ibaraki は語義判別規則の学習システムである．Ibaraki が目指したところは教師なし学習の手法である Co-training の手法を用いて，少量のラベル付きデータと大量のラベルなしデータから語義判別規則を学習することであった．しかし実現はされず，結果的に決定リストを用いた単純な学習システムに終わった．

本報告では，上記の狙いの背景と断念された理由を述べ，次に作成したシステムとその評価，その後の発展などを述べる．

2. 教師なし学習の利用

自然言語処理では個々の問題を分類問題に帰着させ，帰納学習の手法を利用してその問題を解決するというアプローチが大きな成功をおさめている．語義判別問題は典型的な分類問題であり，帰納学習の手法を用いることで，語義判別規則を自動構築できる．

SENSEVAL2 の日本語のタスクには翻訳タスクと辞書タスクが設定された．翻訳タスクも TM の例文番号を語義と考えれば，どちらも同じ語義判別問題として扱える．このため，どちらのタスクに対しても訓練データを準備し，帰納学習手法を用いるのが，最もオーソドックスかつ現実的なアプローチである．おそらくできるだけ質が高く，できるだけ大規模な訓練データを準備できたシステムが，コンテストでは最も良い成績をおさめるであろうことは予想できた．このようにコンテストが訓練データの作成競争になると，コンテスト自体の意味はない．また筆者のように個人で参加するチームには不利になる．

そこで教師なし学習を用いることを検討した．教師なし学習は上記の訓練データの作成コストが高いという問題の解決に向けて提案された手法である．基本的には，少量の正解の付与されたラベル付き訓練データから学習される分類器の精度を，正解の付与されていないラベルなし訓練データを用いて改善するアプローチを取る．教師なし学習の手法を用いることで，訓練データを作成する手間を大きく省くことができる．また日本語翻訳タスクは TM の各番号を語義と考えれば，ラベル付き訓練データの作成は非常に負荷の高い作業であることが予想された．そのため翻訳タスクが教師なし学習の手法を用いる格好のタスクと考えられた．

教師なし学習の代表的な手法としては Naive Bayes に EM アルゴリズムを組み合わせたもの[1], Transductive 法[2], Co-training[3] などがある．EM アルゴリズムの手法はデータの発生源に混合モデルを仮定しているので，語義判別問題に適用できるかどうかは明らかではない．また Transductive 法は SVM を用いて，ラベルなしデータをどのクラスに分類すれば，全体としての精度が上がるかを確認しながら，ラベル付きのデータを増やす手法であり，これは明らかに膨大な計算時間を必要とし，実用的な手法とは考えられない．そこで 3 番目の手法の Co-training を用いることにした．

Co-training は，ラベルなし訓練データを用いることで, 分類規則の精度向上がなされることを, PAC 学習の枠組みで理論的に示している点で注目されている．Co-training は独立な 2 つの素性集合を設定し，一方の素性集合のみを用いてラベル付き訓練データから分類器 1 を作成する．分類器 1 を用いてラベルなし訓練データの判別を行い，信頼性の高いものをラベル付き訓練データに加える．同様にして，もう一方の素性集合のみを用いることによって，ラベルなし訓練データの一部をラベル付き訓練データに加える．このようにラベル付き訓練データを増やすことで分類器の精度を向上させてゆく．Co-training は独立な 2 つの素性集合さえ設定できれば，実装は容易であるために，文書分類[3]，固有表現抽出[4]に応用されている．そして Ibaraki は Co-training の手法を語義判別に応用することを目指した．

3. 素性集合の設定

Co-training を利用するためには，ある程度独立と考えられる 2 組の素性集合が必要になる．この設定のために，単語 w が現れた文脈 b を左文脈 bl と右文脈 br に分割する． bl とは w の左に位置する単語列であり，br は w の右に位置する単語列である．bl から得られる素性集合を x1 とし，br から得られる素性集合を x2 とする．

例えば,「声」の語義は『意見』という語義と『喉から発声される音』という語義がある．「日本国民の

声を集めました」という文の「声」の左文脈は「日本国民の」であり，右文脈は「を集めました」である．

次に x1 に対して表 1 に示す L1, L2, L3 の 3 つの素性を設定する. 同様に x2 に対しても表 1 に示す R1, R2, R3 の 3 つの素性を設定する．

表 1： 設定した素性

素性	素性値
L1	1 つ前の単語
L2	2 つ前の単語-1 つ前の単語
L3	3 つ前の単語-2 つ前の単語-1 つ前の単語
R1	1 つ後の単語
R2	1 つ後の単語-2 つ後の単語
R3	1 つ後の単語-2 つ後の単語-3 つ後の単語

例えば,「日本国民の声を集めました」という文は以下のように形態素解析される．各行が分割された単語であり，第 1 列が表記，第 2 列が原型，第 3 列が品詞を表す.

日本	日本	名詞－固有名詞－地域－国
国民	国民	名詞－一般
の	の	助詞－連体化
声	声	名詞－一般
を	を	助詞－格助詞－一般
集め	集める	動詞－自立
まし	ます	助動詞
た	た	助動詞

この結果から以下の 6 つの素性ができる．

L1=の，L2=国民－の，L3=日本－国民－の，
R1=を，R2=を－集める，R3=を－集める－ます

このようにして設定した x1 と x2 は，ある程度独立な 2 組の素性集合になる．

ラベル付き訓練データを利用して，素性集合 x1 のみから，あるいは素性集合 x2 のみから語義判別の規則を学習する必要がある．これらの学習に，本論文では決定リスト[5]を利用した．

4. Co-training 適用の困難性

素性集合 x1 と x2 を合わせた素性を用いれば，少なくとも決定リストによる語義判別規則の学習は可能である．目指したのは，Co-training とラベルなしデータを用いて，この決定リストの精度を向上させることであった．

しかしながら Co-training の適用は断念された．その理由を述べる．主に 3 つの問題があった．

A) クラスの種類数の問題

分類問題が扱うクラスの種類数は一般に多値である．しかし判別すべきクラスがあまりに多い場合には，さまざまな雑多な問題が発生する．翻訳タスクの場合, 各単語の語義は 10 種類以上存在する．

もちろん，２値分類から多値分類を行う手法はいくつか提案されており，理論的に無理というわけではない．作業量の問題であった．一般的な多値（それも 10 以上の多値）に対応したシステムを作成するのは作業量的に避けたかった．今回のコンテストだけで実験的に試す場合，結局，各単語についてシステムを調整する必要が生じる．そのような作業を行うよりも単純にラベル付けした方が，作業としても容易でしかもその方が正解率は上がると予想された．この点でラベルなしデータを用いるという実質的なアドバンテージがなくなってしまった．

また Co-traininig には各クラスに関して判別関数の一貫性の仮定が必要となる．クラスの数が多い場合，この仮定が崩れることも悪影響を及ぼす恐れを感じさせた．

B) ラベルなしデータの量の問題

Co-training はラベルなしデータから少量のサンプルを取り，その中で高い信頼度でラベル付けできた事例をラベル付き訓練データに加える．ここで付け加える事例は全体の分布から外れないようにしなくてはならない．例えばクラス a , b , c の出現割合が 5∶3∶2 であれば，付け加える事例も 5∶3∶2 になるようにしなければならない．このためクラスの種類数が多い場合，必要とされるラベルなしデータは非常に大規模になる．

報告者が利用できるラベルなしデータは新聞記事 5 年分程度が限度であり，Co-training により精度向上が図れるかどうかは疑問であった．

この問題も避けられるかもしれなかったが，結局，先の問題と絡んだ作業量の問題でもあっ

た.

C) 設定した素性集合の問題

Co-training では独立した 2 組の素性集合を設定しなければならないという大きな条件がある．さらに一方の素性集合だけから学習できる分類器は正しい分類を行えなければならないという条件もある．ここでは左文脈と右文脈に素性集合を分割している．これらの素性集合はほぼ独立と考えて良いが，一方の素性集合だけで語義が判別できるかどうかは疑問がある．

特に動詞の場合，連体修飾の用法でない限り，右文脈だけからその語義を判断はできないことは明らかである．そのために動詞については，はじめから Co-training を断念していた.

Co-training は両刃の剣である．Co-training はラベルなしデータからある事例にラベルをつけて訓練データの事例を増やす．そのラベルが正しければ，精度は向上するが，ラベルが誤れば精度は逆に悪くなる. この点から Co-training を無理に適用することはできないと判断した．結果的に作成できたシステムは名詞に関しては，前章で述べた 6 つの素性を利用した決定リストによる学習システムとなった．動詞に関しては，注目する単語の前後３単語内の内容語を素性とした.

5. 評価結果と考察

5.1. システムの評価

学習のための訓練データとして，毎日新聞'95 年度版から該当単語を含む文を適当な数だけ取りだし，ラベルをつけた．TM のデータは基本的に全部使った．ただし，学習に不都合が起こりそうな部分は，省くなどした．表 2 と表 3 に，単語ごとに用意できた訓練データの事例数，作成できた決定リストの大きさ，コンテストでの正解率を示す.

名詞の場合，事例数から考えて，決定リストが比較的大きいのは頻度による間引きをしなかったからである．動詞に対しては Co-training の適用を断念していたので，訓練事例数は大目に用意した．名詞については Co-training の適用を考えていたので，少な目の 50 強の訓練事例を用意した.

名詞の正解率は 0.5893，動詞の正解率は 0.6533 であった.

表 2：システム評価（名詞）

単語	事例数	決定リストの大きさ	正解率
ippan	87	174	0.4667
ippou	63	101	0.5667
ima	67	135	0.2667
imi	69	181	0.7000
kaku_n	58	121	0.8000
kiroku	65	159	0.4667
kokunai	62	144	0.7333
kotoba	79	183	0.8000
shimin	64	157	0.7333
jigyou	66	186	0.4000
jidai	89	249	0.8000
sugata	77	206	0.3667
chikaku	64	165	0.6000
chushin	61	157	0.5000
hana	64	139	0.5333
hantai	73	176	0.7333
baai	73	194	0.7333
mae	62	161	0.7000
mune	79	179	0.5667
mondai	81	204	0.5000
平均	70	169	0.5893

表 3：システム評価（動詞）

単語	事例数	決定リストの大きさ	正解率
ataeru	300	230	0.5333
iu	279	210	0.8667
ukeru	296	241	0.2000
egaku	284	235	0.6000
kau	303	206	0.8000
kaku_v	274	176	0.9333
kiku	270	191	0.3000
koeru	299	208	0.7667
tsukau	307	238	0.9333
tsukuru	295	238	0.5667
tsutaeru	280	199	0.3667
deru	288	220	0.3000
noru	294	236	0.4667
hakaru	286	217	0.8667
matsu	282	223	0.6333
mamoru	279	214	0.6667
miseru	258	205	0.9000
mitomeru	276	204	0.7000
motsu	327	264	0.9000
motomeru	273	206	0.7667
平均	288	218	0.6533

今回作成したシステムは単純な決定リストの学習システムであり，手法自体に新規性はない．

正解率は各単語の持つ語義判別の難しさに依存しているために，細かく調査することもここでは行わない．また正解率は「システムの判別力」,「各単語の持つ語義判別の難しさ」の他に「訓練データの質」も大きく関与しているはずである．今回のコンテストを通じて，大量の単語に語義のタグをつけたが，TM で設定している語義は細かすぎて，非常に高負荷な作業であった. 当然, 一貫性に自信などないし，多くの場合，語義決定の判断に迷わされた．正解率について考察するには，作成した「訓練データの質」も調査する必要がある．

もちろん，高負荷なラベル付け作業が予想できたからこそ，ラベルなしデータからの学習戦略が有効と判断したのだが，結局断念してしまった．この点は非常に反省している．正解率を下げても狙いのシステムを作成し，システムの評価を行うべきであった．

5.2. TM のみからの学習とクラスタリングの効果

作成したシステムに評価できる点はない．しかし訓練データを作成したことは無駄ではなかったはずである．これを確認するために，TM のみから決定リストを作成し，その正解率を調べた．この値は翻訳タスクの学習システムのベースラインとも考えられる．

またコンテストの解答を見ると，TM の番号を語義と考え，翻訳タスクを語義判別問題として扱ったのは単純すぎたことがわかる．なぜなら，コンテストの正解を眺めると，１つの解答には非常に多くの TM の番号が割り振られている．つまり判別のクラスを必要以上に細かく分ける必要はなかった．むしろ，TM の例文を吟味し，ある程度 TM の例文をその訳語からクラスタリングして，判別のクラス数を減らしてから，語義判別問題として解くべきであった．このクラスタリングの優劣が正解率に大きく影響しているはずである．この点を確認するために，コンテストの正解データから例文番号間の dice 係数を求め，その値が，0.67 以上のものを同じグループにして，TM の例文にクラスタリングを施した．その後に先と同様に決定リストを作成し，その正解率を調べた．

実験の結果を表 4，表 5 及び表 6 に示す．

表 4：TM のみの学習（名詞）

単語	TM のみ	TM＋クラスタリング	本システム
ippan	0.2333	0.3333	**0.4667**
ippou	0.4000	0.5000	**0.5667**
ima	0.3000	**0.3667**	0.2667
imi	0.7333	**0.7667**	0.7000
kaku_n	0.2333	0.3000	**0.8000**
kiroku	0.3667	**0.8333**	0.4667
kokunai	**1.0000**	**1.0000**	0.7333
kotoba	0.6333	**0.9667**	0.8000
shimin	**0.9000**	0.5667	0.7333
jigyou	0.4333	**0.6667**	0.4000
jidai	0.6333	0.5667	**0.8000**
sugata	0.1667	**0.3667**	**0.3667**
chikaku	0.4333	0.3667	**0.6000**
chushin	**0.5333**	0.2000	0.5000
hana	0.3667	**0.8000**	0.5333
hantai	0.5667	**0.9000**	0.7333
baai	0.5667	0.3667	**0.7333**
mae	0.3000	0.2667	**0.7000**
mune	0.1333	0.2667	**0.5667**
mondai	0.5000	**0.5333**	0.5000
平均	0.4717	0.5467	**0.5983**

表 5：TM のみの学習（動詞）

単語	TM のみ	TM＋クラスタリング	本システム
ataeru	0.3333	0.3333	**0.5333**
iu	0.3000	0.7333	**0.8667**
ukeru	0.1333	**0.3333**	0.2000
egaku	0.5667	0.5667	**0.6000**
kau	0.2333	0.7333	**0.8000**
kaku_v	**0.9667**	**0.9667**	0.9333
kiku	0.2333	**0.4667**	0.3000
koeru	0.6333	0.6333	**0.7667**
tsukau	0.0333	0.5333	**0.9333**
tsukuru	0.0333	0.0333	**0.5667**
tsutaeru	0.2333	**0.3667**	**0.3667**
deru	**0.5000**	0.3667	0.3000
noru	**0.4667**	0.3000	**0.4667**
hakaru	0.7333	0.7333	**0.8667**
matsu	0.2667	**0.8667**	0.6333
mamoru	0.1333	0.3667	**0.6667**
miseru	**0.9333**	**0.9333**	0.9000
mitomeru	0.2000	0.2000	**0.7000**
motsu	0.2333	**0.9000**	**0.9000**
motomeru	0.0667	0.6333	**0.7667**
平均	0.3617	0.5500	**0.6533**

表 6：TM のみの学習（全体平均）

TM のみ	TM＋クラスタリング	本システム
0.4167	0.5483	**0.6258**

表からわかるように，ほとんどの単語で訓練データを作成した効果がわかる．しかし，値的には小さいが，一部では逆効果になっている．特に名詞にそのような例が多い．名詞については，増やした事例の数が少ないのが原因だと思われる．結局は，訓練データの多い方が正解率はよいという当然の結果である．また TM をクラスタリングした場合，劇的に正解率が向上する単語が存在する．トータルでは正解率が 0.1 以上も上がる．この点は注意すべきである．つまり，TM のクラスタリングの優劣が正解率に大きな影響を及ぼしている．今後，TM のデータを用いて，語義判別の学習システムを評価する際には，このクラスタリングの記述が必要であろう．筆者としては，オーソライズされた組織が基本となるクラスタリングを示してくれることを望む．

5.3. Co-training の適用実験

本章では Ibaraki が目指した Co-training の効果を参考までに示す．

名詞の場合，クラスの種類数が小さければ，Ibaraki の設定で Co-training が可能になる．そこで名詞 chikaku（近く）を例にとり，実験を行った．名詞 chikaku （近く）は主に 3 つの語義をもつ．英訳で言えば，near と soon と almost である．

ラベル付き訓練データは本試験で用いたものである．ラベルなし訓練データは毎日新聞 '93，'95 ～ '97 の４年分の記事から名詞 chikaku （近く）を含む文を取り出すことで収集した．17,142 文収集できた．これを 50 文ごとに分割した．Co-training の各繰り返しではその 50 文から各語義に関する事例を 1 個選び，訓練データに追加した．

結果を図 1 に示す．縦軸はコンテストで用いられたテスト文に対する正解率を示している．また横軸は Co-training の繰り返しの回数を示している．このグラフから Co-training の効果が確認できる．

ここで示した素性集合では Co-training が適用できる単語はおそらく粗い分類で正解が可能な単語に限定されている．適用できる単語をどのようにして増やしてゆくかが今後の課題である．

最後に参考として，本手法の設定で Co-training を用いた語義判別規則の学習を行った研究を，参考文献にあげておく[6].

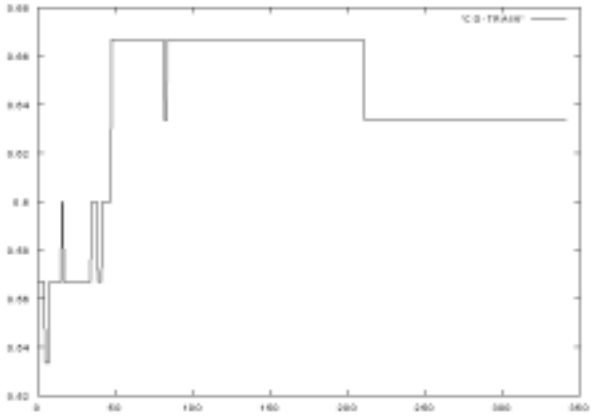

図 1：chikaku に対する Co-training

6. おわりに

SENSEVAL2 の日本語翻訳タスクに参加したシステム Ibaraki を紹介した．教師なし学習の適用を狙ったが，結果的には単純な決定リストのシステムで終わってしまった．オーソドックスな手法であり，手法自体に新規性はない．しかし，逆にオーソドックスであったが故か，似たようなシステムは参加していなかった．ひとつの参考値を与えられたと思う．

文　献

[1] Kamal Nigram, Andrew McCallum, Setastian Thrun, and Tom Mitchell, “Text classification from labeled and unlabeled documents using EM,” Machine Learning, Vol.39, pp.103-134, 2000.

[2] Thorsten Joachims, “Transductive inference for text classification using support vector machines,” 16th International Conference on Machine Learning (ICML-99),pp.200-209, 1999.

[3] Avrim Blum and Tom Mitchell, “Combining Labeled and Unlabeled Data with Co-Training,” 11th Annual Conference on Computational Learning Theory (COLT-98),pp.92-100, 1998.

[4] Michael Collins and Yoram Singer, “Unsupervised Models for Named Entity Classification,” 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, pp.100-110, 1999.

[5] David Yarowsky, “Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French,” 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94),pp.88-95, 1994.

[6] 新納浩幸, “ 素性間の共起性を検査する Co-training による語義判別規則の学習, ” 情報処理学会自然言語処理研究会,NL-145-5,pp.29-36, 2001.