半教師ありクラスタリングによる動詞辞書への大規模用例付与*

上野 孝治　飯田 龍　乾 健太郎　松本 裕治
奈良先端科学技術大学院大学
{takaha-u,ryu-i,inui,matsu}@is.naist.jp

1 はじめに

述語項構造解析とは，文中の各述語について多義性を解消し，述語がとる項を同定するタスクである．述語項構造解析は，格の交替といった表現の多様性を吸収でき，言い換えや質問応答といった自然言語処理の応用技術の精度向上が期待できる．

述語項構造解析は，おおきく分けて教師なし手法と教師あり手法に分類することができる．教師なし手法では，学習コーパスが不要という利点を持っているが，教師あり手法に比べて解析精度が低いという欠点がある [5]．一方，教師あり手法では，教師なし手法に比べて高い精度で解析が可能であるが，学習するための訓練事例が必要になる．訓練事例を作成するためには，1つ1つの用例に対して，人手で正解を付与する必要があり，莫大な作業がかかる．このため，現実的なコストで訓練事例を作成する方法が必要になる．

そこで，本稿では，教師あり手法に必要な訓練事例を効率的に作成するために，用例を語義で分類するコストを削減する手法を提案する．まず，各述語について新聞記事等の生コーパスから大規模に用例を獲得し，獲得した用例集合に対して半教師ありクラスタリングを適用する．次に，得られた各クラスタに対して，その代表元に人手でラベル (例えば語義) を付与する．この手法により，各述語について十分な量のラベルつき用例を作成することが可能になる．

クラスタリングの手法については，K-Means といった教師なしクラスタリングから，事例間の制約を部分的に与えてクラスタリングを行う半教師ありクラスタリング，十分なラベルつき事例を用意してラベルなし事例を分類する教師ありクラスタリングなど，様々な手法が提案されている．提案手法では，用例間の類似度によるボトムアップクラスタリングと Basu ら [2, 3] が提案した MPCK-Means を用いる．また，実数素性だけではなくシンボル素性も扱えるように MPCK-Means を拡張する．

2 節では，関連研究として，用例間の類似度によるボトムアップクラスタリングと Basu らが提案した半教師ありクラスタリングである MPCK-Means について述べる．3 節では，2 節の問題点を考慮したクラスタリングの手法を提案する．4 節では，提案手法の実現可能性を調査することを目的とした予備的な評価実験を行い，その結果について考察する．最後に 5 節でまとめる．

*Large-scale Example-assignment to Argument Structures using Semi-supervised Clustering
Takaharu Ueno, Ryu Iida, Kentaro Inui, and Yuji Matsumoto
Nara Institute of Science and Technology

2 先行研究

2.1 用例間類似度によるボトムアップクラスタリング

平野ら [12] は，各述語について生コーパスから大規模に用例を獲得して，クラスタリングし，得られたクラスタにタグを付与することによって，タグ付与作業のコストを削減する手法を提案している．平野らのクラスタリングでは，河原ら [4] が提案した用例間の類似度を用いて，ボトムアップクラスタリングを行う．クラスタリングには，表層格フレーム辞書を自動構築する河原らの「動詞の用法を決定する重要な格要素は動詞の直前にくることが多く，動詞と直前の格要素をペアにして考えると動詞の用法はほとんど一意に決定される」という主張に基づき，動詞の直前格に着目したクラスタリングを行う．平野らは，この手法により，タグ付与誤りをほとんど発生させないクラスタリングを実現している．

しかし，平野らの手法では，経験的に定めた距離関数を用いているため，動詞毎に効果的な素性を考慮しながらクラスタリングできているとは言いがたい．また，平野らの手法では，たとえラベルつき用例が入手可能であったとしても，それをクラスタリングに利用することができない．以下に述べるように，これらの問題は半教師ありクラスタリングアルゴリズムを用いることで解決される．

2.2 半教師ありクラスタリング: MPCK-Means

MPCK-Means は，事例間の制約つき K-Means クラスタリング [1] に素性の重み学習を組み合わせたクラスタリングアルゴリズムである [2, 3]．$\mathcal{X}$ を用例集合，x_i を用例，l_i を用例 x_i が属すクラスタ，μ_{l_i} をクラスタ l_i のセントロイド，A を重み行列，$\mathcal{M}$ を must-link の集合，$\mathcal{C}$ を cannot-link の集合とすると，MPCK-Means の目的関数は，以下の式で定義される．

$$\mathcal{J}_{mpckm} = \sum_{x_i \in \mathcal{X}} \left(\|x_i - \mu_{l_i}\|_A^2 - \log(\det(A)) \right) + \sum_{(x_i,x_j) \in \mathcal{M}} w_{ij} f_M(x_i, x_j) \mathbf{1}[l_i \neq l_j] + \sum_{(x_i,x_j) \in \mathcal{C}} \bar{w}_{ij} f_C(x_i, x_j) \mathbf{1}[l_i = l_j] \quad (1)$$

$$f_M(x_i, x_j) = \|x_i - x_j\|_A^2 \quad (2)$$

$$f_C(x_i, x_j) = \|x' - x''\|_A^2 - \|x_i - x_j\|_A^2 \quad (3)$$

ここで, x' と x'' は，最も距離の遠い 2 つの元である．

用例間の制約には，must-link と cannot-link の 2 種類の制約があり，クラスタリングの前にあらかじめユーザが指定する．must-link は 2 つの用例が同じクラスタに属さなければならないという制約，cannot-link は 2 つの用例が同じクラスタに属すことができないという制約である．指定する制約は，クラスタリングにおいて，違反した際のペナルティーとして用いられる．また，MPCK-Means は，素性の重みを表す行列 A を用意し，目的関数 $\mathcal{J}_{mpckm}$ を最小化するように最適な重み行列を学習することができる．

用例集合をクラスタリングする際，動詞ごと (あるいは動詞の語義ごと) に効果的な素性を選ぶことができれば，より高い精度でクラスタリングが可能になる．クラスタリングの途中で素性の重みを学習する MPCK-Means は，この点で有利である．ただし，オリジナルの MPCK-Means は，n 次元の実数空間における素性ベクトルのクラスタリングを対象としており，用例の格要素 (名詞) のようなシンボルを素性に加えるには何らかの工夫が必要である． 3 節で述べるように，我々は素性ベクトル間の距離を再定義することによってシンボル素性を柔軟に扱えるように拡張する．

3 提案手法

クラスタリングには，平野らの用例間類似度によるボトムアップクラスタリングと MPCK-Means を併用する．

3.1 制約

2 節で，用例間の制約として，must-link と cannot-link があることを述べた．提案手法では，must-link は用いずに，cannot-link だけを制約として用いる．具体的には，語義が異なると事前にわかっている用例間に cannot-link の制約を加える．must-link を用いないのは，語義が同じ用例に must-link を与えてしまうと，制約に用いた用例が大量に含まれるクラスタが構成されてしまう可能性があり，場合によっては，制約に用いた用例だけでクラスタを構成してしまう可能性があるためである．

3.2 クラスタの初期値

クラスタの初期値を，平野ら [12] のボトムアップクラスタリングによって求めるようにした．こうすることによって，平野らの手法の長所と MPCK-Means の長所の両方を踏まえたクラスタリングが可能になる．

3.3 MPCK-Means からの拡張

名詞や格をその有無を表現する 0-1 の実数素性として定義した場合，非常に高い次元数で，かつ疎なベクトルによりクラスタリングすることになってしまう．そこで，名詞間の類似度や格の出現パタンといった非スカラーなシンボル素性を扱えるように MPCK-Means を拡張する．

3.3.1 素性ベクトル間の距離の再定義

MPCK-Means では用例 x_i と x_j の距離関数は次式で定義される．

$$D_A(x_i, x_j) = \|x_i - x_j\|_A^2 = \sum_k w_k \cdot d(x_{ik}, x_{jk})^2 \quad (4)$$

ここで，重み w_k は，次式で定義される．

$$A_{ij} = \begin{cases} w_i & (i = j) \\ 0 & (otherwise) \end{cases}$$

用例 x_i と x_j の第 k 次元が実数素性の場合，差は以下の式で定義される．

$$d(x_{ik}, x_{jk}) = d_{ik} - d_{jk} \quad (5)$$

提案手法では，第 k 次元がシンボル素性の場合，このような定義は使えないので，個別に距離関数を定義する．例えば，名詞を素性とする場合，S_{n_i}，S_{n_j} を名詞 n_i と n_j の意味クラスの集合，x，y を意味クラス，l_x，l_y を x，y のシソーラスからの根の深さ，L を x，y の意味クラスで一致している階層の深さとするとき，次式のような距離関数 [4] で名詞 n_i と n_j の距離を与えることができる．

$$d(n_i, n_j) = 1 - max_{x \in S_{n_i}, y \in S_{n_j}} \frac{2L}{l_x + l_y} \quad (6)$$

同様にして，用例間の格の一致度や，共通格に含まれる格用例間の類似度といったシンボル素性について，それぞれ距離関数を定義する．シンボル素性を導入することによって，ベクトルが疎になる問題点を解消することができる．

3.3.2 代表元の定義

シンボル素性は，用例間に定義される素性であるため，実数素性とは異なり，四則演算が定義できない．そこで，代表元 (medoid) を定義し，用例と代表元との距離が計算できるように変更する．クラスタ C_i の代表元 m_i を，以下のステップで求める．

1. C_i に含まれる用例をマージしたものを μ_i とする．
   - μ_i の出現格は C_i の用例の出現格の和集合とする．
   - μ_i の各出現格に対応する格要素は，C_i の用例の当該格の格要素の集合とする．
2. C_i の元の中で，μ_i の要素との距離の和が最も小さいものを m_i とする．

4 評価実験

本節では，提案手法によるクラスタリングの精度を評価し，その結果を報告する．ベースラインは，用例間の類似度によるボトムアップクラスタリングとした．

述語	語義番号	語義	用例
まとめる	1	対立・混乱しているものをうまく一つに整える。	社員が 規範を まとめる。
	2	一つのものとして完成させる。	確認事項を 文書に まとめる。
	3	髪をきちんと整える。	髪を 後頭部で まとめる。
	4	ある一つの形をとって，何かを成立させる。	情報を 英文で まとめる。
	5	散らばっていた物を一か所に集めて、きちんと整える。	荷物を 一つに まとめる。

表 1: 動詞「まとめる」の動詞辞書の項目

4.1 素性

クラスタリングでは以下の 4 種類を素性として用いた.

- 統語情報 (動詞の直前格，用例の格パタン，格と格の bi-gram をその有無によって 0-1 の実数で表現)
- シンボル素性
  - 用例間の格の一致度
  - 共通格に含まれる格用例間の類似度
  - 用例間の類似度

ここで，共通格に含まれる格用例間の類似度を求める際，名詞間の類似度が必要になる．名詞間の類似度には，分類語彙表 [6] を用いた.

4.2 制約の選択

制約に用いる用例の選択方法には様々なものが考えられるが，今回の実験ではタグ付与された語義の分布を反映して無作為に選択した.

4.3 評価実験データ

実験データは，次の 4 つの動詞 (「まとめる」「加える」「認める」「向ける」) を選択して評価した．実験データは，各動詞について，新聞記事から自動的に獲得し，その中から頻出する 1000 用例に人手で語義を付与して作成した．さらに，人手で語義を付与した 1000 用例のうち，慣用表現や係り受け不備の用例を取り除き，語義が一意に定まっている用例だけを評価実験データとして用いた．また，制約に用いた用例は，評価の対象には含めなかった.

各動詞の語義数は「まとめる：5」「加える：5」「認める：5」「向ける：6」である．例として，表 1 に動詞「まとめる」の語義を示す．この動詞辞書は評価実験のために作成したものであり，IPAL 動詞辞書 [7] の中で比較的語義数の少ないものを選択して作成した．生コーパスからの用例の収集には，工藤らが提案した係り受け解析器 CaboCha[8] を用いた.

4.4 評価方法

クラスタリングによって得られるクラスタを，以下の 3 つのタイプに分類してクラスタリングの精度を評価する.

A) 制約に用いた用例の語義が 1 つだけ含まれているクラスタ

B) 制約に用いた用例の語義が複数含まれているクラスタ

C) 制約に用いた用例が含まれていないクラスタ

各タイプによって，語義の付与方法は異なり，タイプ A のクラスタについては，制約に用いた用例の語義を自動的に付与する．また，クラスタリングする前にあらかじめ cannot-link が加えられているにもかかわらず，制約を違反してクラスタを構成してしまったクラスタは，一意に語義を求めることができないタイプ B のクラスタとして分類した．また，制約に用いた用例が含まれていないクラスタをタイプ C に分類した．タイプ C のクラスタには，そのクラスタを代表する用例を人が見て，語義を付与する.

c \ k		10	50	100
baseline		73.4% (608/828)	80.3% (665/828)	82.2% (681/828)
0	(a)	-	-	-
	(b)	-	-	-
	(c)	76.9% (637/828)	80.1% (663/828)	81.8% (677/828)
	(a+b+c)	76.9% (637/828)	80.1% (663/828)	81.8% (677/828)
10	(a)	74.2% (395/532)	68.9% (151/219)	54.9% (62/113)
	(b)	-	-	-
	(c)	83.6% (239/286)	83.8% (502/599)	86.5% (610/705)
	(a+b+c)	77.5% (634/818)	79.8% (653/818)	82.2% (672/818)
50	(a)	74.8% (350/468)	73.5% (183/249)	66.7% (96/144)
	(b)	-	-	-
	(c)	83.2% (258/310)	85.1% (450/529)	86.1% (546/634)
	(a+b+c)	78.1% (608/778)	81.4% (633/778)	82.5% (642/778)
100	(a)	79.8% (446/559)	74.1% (163/220)	68.7% (79/115)
	(b)	-	-	-
	(c)	76.9% (130/169)	84.8% (431/508)	86.6% (531/613)
	(a+b+c)	79.1% (576/728)	81.6% (594/728)	83.8% (610/728)

表 2: 動詞「まとめる」の評価結果

今回の実験では，提案手法の実現可能性を調査するために，クラスタリングの精度の上界を求めた．すなわち，各タイプのクラスタの正解語義は，そのクラスタに最も多く含まれている語義になる．評価式を以下に示す.

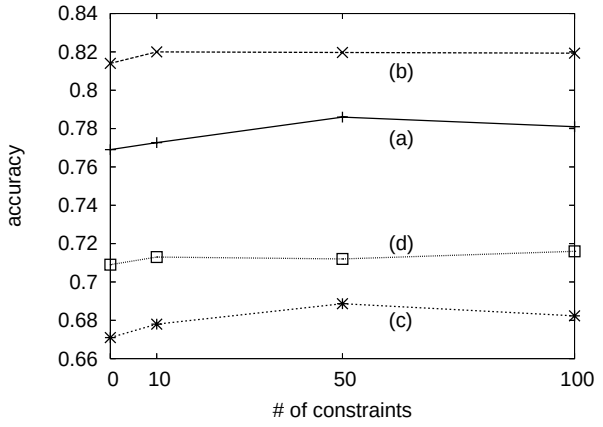
$$\text{各タイプの精度} = \frac{\text{正解語義と一致した用例の数}}{\text{各タイプの用例の総数}}$$

$$\text{総合的な精度} = \frac{\text{正解語義と一致した用例の数}}{\text{用例の総数}}$$

4.5 実験結果と考察

動詞「まとめる」についての結果を表 2 に示す．列はクラスタ数を表し，行は制約に用いた用例数を表している．表 2 より，全ての場合について，タイプ B のクラスタは生成されておらず，制約を違反せずにクラスタリングできていることがわかる．また，総合的な精度においては，常にベースラインの精度よりも提案手法の方が高い値を示している．この結果により，用例間の制約と重み学習がともに有効に働いていることがわかる.

次に，4 つの動詞に関して制約に用いた用例数と精度の関係を調査した．ただし，実験では制約集合を 3 回変更してクラスタリングを行い，得られた結果の平均を最終的な精度とした．クラスタ数を 10 に設定した



(a):まとめる，(b):加える，(c):向ける，(d):認める．
図 1: 制約に用いた用例数と精度の関係（クラスタ数を 10 に設定）

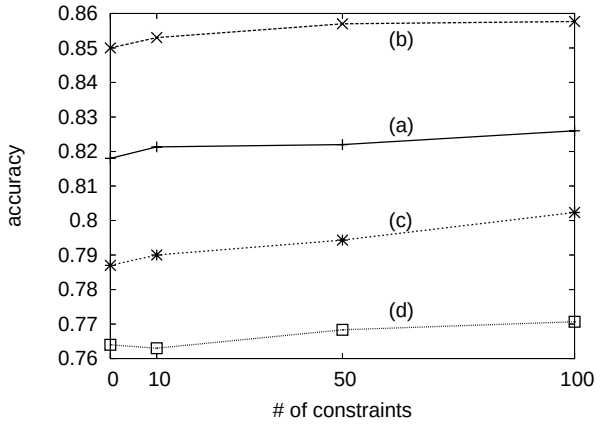


図 2: 制約に用いた用例数と精度の関係（クラスタ数を 100 に設定）

ときの結果を図 1 に，また，100 に設定したときの結果を図 2 に示す．表 2 に示したクラスタリングの結果と同様に，図 1 の全ての動詞に関しても，制約無しの結果に比べ，制約を利用した場合の精度が向上している．この結果より，全ての動詞に関して用例間の制約が有効に働いていることがわかる．

ただし，図 1 と図 2 の結果を見る限り，制約の個数を増やしてもほとんど精度に影響がない．これは人手でクラスタに語義付与を行うという観点から見ると，少量の語義付与事例を作成することで効果的にクラスタリングの精度改善に貢献できることを示している．

また，今回の実験では 4.2 で述べた単純な制約の選択方法を採用しているため，これを工夫することで，少量の語義付与事例を用意するだけでも，さらに精度が向上する可能性もあり，これについては今後の課題としたい．

5 おわりに

本稿では，述語項構造解析の精度の向上を目的として，生コーパスから自動的に獲得した大規模な用例をクラスタリングし，得られたクラスタに対して語義を付与することで，用例を分類するコストの削減手法を提案した．具体的には，用例間の類似度によるボトムアップクラスタリングとシンボル素性を組み込めるように拡張した半教師ありクラスタリングを用いた．予備的な実験の結果，用例間の類似度によるボトムアップクラスタリングと比較を行い，提案手法が高い精度でクラスタリングできることを確認した．また，制約を用いずにクラスタリングするのに比べ，制約を増やすにつれ若干の精度の向上が見られた．この現象がどの程度一般的であるかは今後調査を行う予定である．

今回の実験では，クラスタリングの精度を評価するために，4 つの動詞に限定して，小規模な動詞辞書を作成して評価を行った．しかし，実際に述語項構造解析を実現するためには，より大規模な動詞辞書が必要になってくる．現在，竹内ら [9, 10, 11] は，語彙概念構造 (LCS) を用いた動詞辞書の構築を進めており，この動詞辞書には，各動詞に対して語義が定義されている．今後は，この辞書を利用し，様々な動詞について実験を行い，結果の分析を進めていくと共に，制約に用いる用例の選び方やクラスタ数を指定しないクラスタリングアルゴリズムについても検討したい．

参考文献

[1] Sugato Basu, Arindam Banerjee, and RaymondJ.Mooney. Semi-supervised clustering by seeding. In *Proceedings of the 19th International Conference on Machine Learning(ICML-2002)*, pp. 19–26, 2002.

[2] Sugato Basu, Mikhail Bilenko, and RaymondJ.Mooney. Comparing and unifying search-based and similarity-based approaches to semi-supervised clustering. In *Proceedings of the ICML-2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, pp. 42–49, 2003.

[3] Mikhail Bilenko, Sugato Basu, and RaymondJ.Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the 21st International Conference on Machinew Learning(ICML-2004)*, 2004.

[4] 河原大輔, 黒橋禎夫. 用言と直前の格要素の組を単位とする格フレームの自動構築. Vol. 9, No. 1, pp. 3–19, 2002.

[5] M. Lapata and C. Brew. Using subcategorization to resolve verb class ambiguity. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 266–274, 1999.

[6] 国立国語研究所. 分類語彙表, 国立国語研究所資料集 6. 秀英出版, 1993.

[7] 情報処理振興事業協会技術センター. 計算機用日本語基本動詞辞書 IPAL(Basic Verbs) 辞書編. 1987.

[8] 工藤拓, 松本裕治. Support vector machine を用いた chunk 同定. 自然言語処理, Vol. 9, No. 5, pp. 3–21, 2002.

[9] 竹内孔一. 語彙概念構造による動詞辞書の作成. 第 10 回言語処理学会年次大会, pp. 576–579, 2004.

[10] 竹内孔一. 言語処理を意識した語彙概念構造の構築. 東京大学 21 世紀 COE 「心とことば」シンポジウム「語彙概念構造辞書の構築と応用」, 2005.

[11] 竹内孔一, 乾健太郎, 藤田篤, 竹内奈央, 阿部修也. 分類の根拠を明示した動詞語彙概念構造の構築. 自然言語処理研究会 2005-NL-169.

[12] 平野徹, 飯田龍, 藤田篤, 乾健太郎, 松本裕治. 動詞項構造辞書への大規模用例付与. 言語処理学会 11 回年次大会発表論文集, 2005.