百科事典を対象とした属性値抽出

鈴木琢也† 関根聡†† 増山繁†
†豊橋技術科学大学知識情報工学系
††ニューヨーク大学

1. はじめに

ある「対象物」について，その特徴や性質を現す関係である「属性」と，その関係が取りうる値である「属性値」を抽出する属性値抽出タスクは，質問応答のデータ構築などに有用であることが認識されはじめ，近年次第に研究されるようになってきた[1,2,3,4,5].

属性値を抽出する情報源として，新聞記事[1]やHTML文書[2,3,4,5,6]を用いたものが挙げられるが，本研究では情報源として百科事典を用いる．百科事典は，ある項目についての情報が比較的綺麗に，構造化しやすく書かれているため，属性とその属性値の対が発見しやすく，自動抽出を高い精度で実現できると考えたためである．

本研究では百科事典からの属性値の自動抽出を，チャンク同定問題を解くことにより行うこととした．これにともない，まず属性の枠組みを設定し，ある程度の量の百科事典テキストをタグ付けして学習データを作成する．これを用いて機械学習により属性値抽出器を作成し，残りの大量の百科事典テキストから属性値を抽出する．

以下，次章では属性の設定法について詳細に述べる．3章で百科事典テキストから，定義した属性の属性値を抽出する方法を述べ，4章でその改良案を述べる．5章で提案手法の評価実験を行った結果を報告し，6 章で関連研究を示し，7章でまとめと今後の課題について述べる．

2. 属性の設定

先行研究[1]等では，対象物に対する属性は文に出現するものからボトムアップに自動的に設定しているが，本手法では属性をトップダウンに，人手により設定する．しかし，対象としている百科事典は全部で約11万項目あるため，全ての項目の対象物に対して人手で属性を決定するのは現実的でない．幸いにも，我々は過去に行った百科事典を対象とした質問応答システムの開発[6,7]において，全項目を約150のカテゴリに分類している．なおその際，正確さの確保のため，百科事典の分類情報を利用した大まかな分類の後，全項目に対し人手により確認を行っている．ここで，各項目の属性は，この分類ごとに似通ったものになると考えられる．そこで我々は，分類ごとに最低10項目，最大50項目をランダムに取り出し，それらの説明文を読んで，可能な属性を抽出するという作業を行った．そして，基本的に2項目以上の例で共通の属性があったものを，それぞれのカテゴリの属性として設定した．そして，設定した属性を用いて，各カテゴリについて最大50項目の属性値表を作成した．表1に，カテゴリ「人物」に設定した属性とその例，その属性の属性値が，「人物」の全46項目のうち何項目に出現したかの頻度と割合の一例を示す．

表 1：属性と属性値の一例

属性（属性数：19）	属性値の例	頻度	%
職業	プロ野球選手, 経済学者, 詩人	46	100.0
国籍	アメリカ、ドイツ、ブルガリア	29	63.0
過去滞在地	イギリス，ニューヨーク，肥後（熊本県）南半	12	26.1
没年	１７０４年２月２３日，３５年（建武２），不詳	10	21.7
受賞歴	世界文化賞, 最優秀選手（ＭＶＰ），新潮社文芸賞	8	17.4
称号	ナイト, 野球殿堂入り, フンボルト大学から名誉博士号	6	13.0
父親	景帝, 参議橘恒平（たちばなのつねひら），政経（まさつね）	5	10.9
死因	交通事故, 斬首（ざんしゅ），長旅の疲労	5	10.9

表より，全ての項目の説明テキストに記載されている属性（例えば「職業」）もあれば，10.9%の項目にしか記載されていない属性（例えば「父親」）もあることが分かる．ここで，記載率の低い属性は，百科事典テキストの記述漏れである場合よりも，そもそもその属性の属性値が存在しないという場合の方が多い（存命の人物の説明に「没地」の属性値は記載されることはない）．また，ある項目について，一つの属性に唯一の属性値が決まるものと，複数の属性値が存在するものがある．例えばカテゴリ「人物」においては，前者は属性「誕生日」等が，後者には属性「代表作」等が該当する．

なお，現在も属性値表の作成は継続して行われており，先行研究[8]の段階ではカテゴリごとに最大 10 程度の項目しか無かったが，本研究の段階においてはその量が最大50程度に増加していることを付け加えておく．

3. 属性値の抽出

本章では提案する属性値の抽出方法について述べる．手法は以下の2ステップからなる．

(1) 属性値表の内容をトレーニングデータとし，教師あり機械学習を用いて属性ごとに抽出器を作成する

(2) 属性値表に存在しない百科事典の項目に対し，抽出器を用いて属性を抽出する

以下，各ステップについて述べる．

3.1 属性値抽出器の作成

本手法では属性値抽出を，形態素に分割された要素列に対し，IOB2 タグ[8]を用いたチャンク同定問題を解くことにより行う．形態素列を，属性値の先頭ないし属性値そのものを現すクラス（B タグ），属性値における先頭以外を現すクラス（I タグ），属性値以外を現すクラス（O タグ）の3値に分類する．なお，ここでのチャンク同定は，属性ごとにそれぞれ独立して行う．なぜなら，複数の属性値が同一の部分を指している場合も考えられるためである．

IOB2 タグの推定は，教師あり機械学習を用いて行う．

学習データには属性値表の内容を用い，形態素の要素列に正解としての IOB2 タグを割り当て，教師あり機械学習で正解を導くための規則を学習する. このようにして，属性値抽出器を，属性ごとに作成する.

3.2 属性値の抽出

作成した属性値抽出器を用いて，属性値表に存在しない項目から属性値を抽出する．本論文においては，この属性値抽出の性能を評価するため，属性値表に存在する項目を訓練データと評価データに分割し，訓練データを用いて学習した抽出器について，評価データを用いて性能を評価することとした.

4. 属性値の集合演算

前節で述べた手法の性能を高めるため，複数の学習条件を用いてチャンク同定を行うことを考える．複数の属性抽出器において，同一の訓練データを学習した場合，それらの学習条件が異なれば，評価時に出力する属性値の集合はそれぞれ異なるものとなる．その時，学習条件が近いほど，出力される属性値は同じものが多くなり，学習条件が離れているほど，出力される属性値に同じものが少なくなると考えられる．それが正しければ，複数の属性抽出器において学習条件が離れているとき，それらの抽出した属性値の和集合を求め，その集合を用いて評価を行えば，正解の属性値も増加し，再現率が上昇することが期待できる．同様に，積集合を求めると，隔たりの大きい学習条件の双方に支持される属性値のみが出力の対象となり，適合率の上昇が期待できる.

5. 評価実験

5.1 データ

百科事典のデータとして，小学館発行の「日本大百科全書」を用いる．各項目ごとの 5 行から 100 行程度のテキストによる解説を属性値抽出の対象として用いるが，解析誤りの原因となるため，前処理として，括弧の中に付された読み仮名や補足説明を削除している.

評価実験を行うコーパスには，「人物」「市区町村」「植物」の 3 つのカテゴリの属性値表を使用した. これらは，「人物」については 46 項目，「市区町村」は 50 項目，「植物」には 46 項目のデータがあったが，評価実験を行うには十分な量とはいえない．そこで，各カテゴリに 100 項目づつのデータを追加した．これにより，各属性において学習に使用できる項目の数が約 3 倍となった．これ以降，断りのない場合は，コーパスとはこの 100 項目づつを追加したものを指す.

評価は，コーパスにおける一つの属性の，属性値を持つ項目のうち 4 分の 3 を訓練データ，4 分の 1 を評価データとし，F 値(β=1)を算出することにより行う.

5.2 ベースライン

提案手法との比較のため，ベースライン手法として 2 種類の手法の評価値を測定した．一つめのベースライン手法は新里ら[1]の属性値抽出の研究で使用されている，SVM を用いた手法である．解析単位は形態素とし，形態素解析器には Juman[a]を用いている. SVM に与える素性は，単語自身，品詞，文字種である．ここで，文字種の定義[b]は若干異なっている．また，彼らが素性に用いている属性値の辞書は用いていない．それら以外の条件は同じで，カーネル関数には 2 次の多項式カーネル，チャンクタグには IOB2，多値分類に pairwise 法を用い，ウィンドウサイズは 2，解析の方向は「文末から文頭」に固定している．チャンカーには，yamcha[c]を用いている．表 2 に測定した評価値を示す．なおこの表に示す評価値は，全属性で算出した評価値をまとめたものである.

二つ目のベースライン手法は, SVM のベースラインと同様の素性, ウィンドウサイズ, チャンクタグを用いて，CRF で学習を行ったものである．なお，CRF では，学習条件として上記の素性の bigram 素性を追加している. チャンカーには CRF++[d]を用いている. 表 3 に測定した評価値を示す．表より，カテゴリ間で若干評価値が異なっているものの，総合した評価値では SVM が CRF の評価値を上回っていることがわかる．また，CRF は適合率が高く，SVM は再現率が高い傾向があることがわかる.

表 2 : ベースラインの評価

	F値(β=1)	
	SVM	CRF
人物	65.3	63.6
市区町村	66.8	64.9
植物	61.2	62.5
総合	64.4	63.7
適合率(%)	80.3	84.6
再現率(%)	53.8	51.0

5.3 属性値の集合演算

4 節で述べた属性値の集合演算の効果を調べるため，学習条件をベースラインのものから変化させた抽出器を作成し，その評価時の出力に，ベースラインの出力と集合演算を行ったものを集合出力とし，評価を行った．以下では，変化させた学習条件ごとに結果を述べる.

5.3.1 素性

素性を変えた学習条件で作成した抽出器の出力の集合演算の効果を調べるため，ベースラインの学習で使用した文字種の素性を使わず，拡張固有表現[9]を素性として用いた抽出器を作成し，出力された属性値をベースラインのものと集合演算を行った．ここで拡張固有表現は，新納ら[10]の提案している SVM を用いたタガーにより付与している．素性以外の学習条件は，ベースラインのものと同様である．表 3 に SVM と CRF で実験を行った結果を示す.

表より，SVM，CRF 共に，回答数が積集合においては減少し，和集合においては増加していることがわかる．そして，積集合においては，単体の抽出器のどの抽出精度よりも適合率が上がり，再現率が下がっている.

---

a http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman.html

b アルファベット，アルファベット数字，漢数字，平仮名，片仮名，漢字，句点，句読点，中黒，これら以外の 10 種類の定義を用いている.

c http://chasen.org/~taku/software/yamcha/

d http://www.chasen.org/~taku/software/CRF++/

表 3：素性を変えた組の集合演算の評価

	F値(β=1)							
	SVM				CRF			
	文字種	NE	積集合	和集合	文字種	NE	積集合	和集合
人物	65.3	64.3	64.2	65.4	63.6	62.9	60.3	65.9
市区町村	66.8	69.5	65.0	71.0	64.9	68.1	63.6	69.2
植物	61.2	61.6	61.3	61.5	62.5	62.3	61.4	63.3
総合	64.4	65.2	63.5	66.0	63.7	64.5	61.8	66.2
適合率(%)	80.3	78.7	83.9	76.0	84.6	83.4	87.1	81.5
再現率(%)	53.8	55.6	51.0	58.3	51.0	52.6	47.9	55.7
回答数	812	856	738	930	732	765	667	830
正解数	652	674	619	707	619	638	581	676

しかし，再現率の下がり幅の方が適合率の上がり幅より大きいため，F 値は下がってしまっている．和集合においては，積集合とは逆に，適合率が下がり，再現率が上がっている．再現率の上がり幅のほうが適合率の下がり幅より大きく，F 値が上がっていることが確認できた．

5.3.2 チャンクタグ

チャンクタグを変えた学習条件で作成した抽出器出力の集合演算の効果を調べるため，ベースラインの IOB2 タグから IOE2 タグに学習条件を変更した抽出器を作成し，出力された属性値をベースラインのものと集合演算を行った．チャンクタグ以外の学習条件は，ベースラインのものと同様である．結果を表 4 に示す．

表 4：チャンクタグを変えた組の集合演算の評価

	F値(β=1)							
	SVM				CRF			
	IOB2	IOE2	積集合	和集合	IOB2	IOE2	積集合	和集合
人物	65.3	65.1	64.5	65.9	63.6	64.2	62.9	64.8
市区町村	66.8	67.3	66.2	67.9	64.9	65.3	64.1	66.1
植物	61.2	61.4	61.0	61.6	62.5	62.3	61.8	63.0
総合	64.4	64.6	63.9	65.1	63.7	63.9	62.9	64.6
適合率(%)	80.3	79.1	83.0	76.9	84.6	84.7	86.2	83.1
再現率(%)	53.8	54.6	51.9	56.4	51.0	51.4	49.6	52.8
回答数	812	837	759	890	732	736	697	771
正解数	652	662	630	684	619	623	601	641

表より，前節と同様の結果となっているものの，適合率や再現率の変化が少なくなっていることがわかる．

5.3.3 ウィンドウサイズ

ウィンドウサイズを変えた学習条件で作成した抽出器出力の集合演算の効果を調べるため，ベースラインのウィンドウサイズの 2 を 3 に変更した抽出器を作成し，出力された属性値をベースラインのものと集合演算を行った．ウィンドウサイズ以外の学習条件は，ベースラインのものと同様である．結果を表 5 に示す．

表 5：ウィンドウサイズを変えた組の集合演算の評価

	F値(β=1)							
	SVM				CRF			
	2	3	積集合	和集合	2	3	積集合	和集合
人物	65.3	62.2	61.6	65.7	63.6	64.5	63.0	65.0
市区町村	66.8	68.5	66.4	68.8	64.9	66.5	63.4	67.8
植物	61.2	60.3	57.5	63.6	62.5	63.1	60.5	64.9
総合	64.4	63.8	61.9	66.0	63.7	64.7	62.3	65.9
適合率(%)	80.3	80.2	85.1	76.5	84.6	86.1	87.5	83.5
再現率(%)	53.8	52.9	48.6	58.0	51.0	51.9	48.4	54.5
回答数	812	801	693	920	732	731	671	792
正解数	674	642	590	704	619	629	587	661

SVM による出力において単体の抽出器の評価値を見ると，ウィンドウサイズ 3 の単体の評価値はウィンドウサイズ 2 のものと比較して再現率が落ちただけのように見えるが，和集合をとると再現率がかなり高くなっていることから，出力される属性値の異なりが大きいことがわかる．CRF による出力においてはウィンドウサイズ 3 の方が 2 のものよりも評価値が良いものとなっていたが，和集合を取ったものはさらに評価値が良くなった．

5.3.4 学習アルゴリズム

学習アルゴリズムを変えた学習条件で作成した抽出器出力の集合演算の効果を調べるため，SVM と CRF のベースラインの出力を集合演算したものを評価した．結果を表 6 に示す．

表 6：学習アルゴリズムを変えた組の集合演算の評価

	F値(β=1)			
	SVM	CRF	積集合	和集合
人物	65.3	63.6	60.5	68.0
市区町村	66.8	64.9	61.9	69.4
植物	61.2	62.5	56.2	66.4
総合	64.4	63.7	59.6	67.9
適合率(%)	80.3	84.6	87.7	78.7
再現率(%)	53.8	51.0	45.9	59.7
回答数	732	812	624	920
正解数	619	652	547	724

和集合をとったものの F 値が単体のものと比較して 3.5 ポイント程度高くなっており，最も良い評価値を示した．

5.4 最大 50 項目の学習データにおける評価

本論文では評価実験のため，学習データの属性値表に，3 つのカテゴリにのみそれぞれ 100 項目づつを追加した．その結果，学習データがそれらのカテゴリだけ約 3 倍となったのだが，他のカテゴリもその学習量にすることはコストの面で難しい．そこで，100 項目を追加する前の学習データにおいても，本手法が有効であるかどうかを検証した．実験は，学習データとして 46 つのカテゴリの属性値表を使用し，前節の実験において最も良い評価値を示した学習アルゴリズムによる集合演算について，その評価値を測定した．それ以外の実験条件は，前節までのものと同様である．結果を表 7 に示す．

表 7：最大 50 項目の学習データにおいて学習アルゴリズムを変えた組の集合演算の評価

	F値(β=1)			
	SVM	CRF	積集合	和集合
人物	42.2	51.8	42.2	51.8
市区町村	56.1	59.3	55.1	60.1
植物	51.8	58.1	51.5	58.3
学問	28.1	26.5	28.6	36.4
両生類	59.8	60.2	57.6	62.0
神社仏閣	55.7	56.8	55.7	61.5
総合	52.1	49.8	46.3	54.9
適合率(%)	76.5	82.0	87.0	74.1
再現率(%)	39.5	35.7	31.6	43.6
回答数	1948	2309	1624	2633
正解数	1598	1766	1412	1952

表より，単体の抽出器を評価すると，特に再現率の減少幅が大きく，そのために F 値が大きく減少しているこ

とがわかる．しかしそれらの抽出器の出力の和集合をとった結果，再現率がある程度改善され，F 値が単体のものと比較して 2.8 ポイント程度高くなることが確認できた．

5.5 考察

和集合の演算においてF値に対する改善の効果は，学習アルゴリズム間の集合演算が最も大きく，チャンクタグ間が最も小さかった．両者を比較すると，単体の性能ではチャンクタグ間のものの方が F 値が高い．しかし，IOB2 を使った抽出器の出力と IOE2 タグのそれの違いは，語長がある程度長いものに限定され，差異が少ない．それは，積集合をとったときの回答数の減少量からも確認できる. このことから, 集合演算に用いる抽出器の組は，F 値の高いものどうしより，より異なる出力を行う組どうしの方が望ましいという考えが裏付けられたと考える.

全ての実験において，積集合をとることにより適合率を高くすることができ，和集合をとることにより再現率を高くすることができることを確認することができた．もともと本研究は百科事典を対象とした質問応答システム構築のサブタスクであり，質問応答において間違った回答を提示することを極力防ぐため，属性値抽出では適合率を重視せねばならない．このため，適合率の上がり幅より再現率の下がり幅が大きく，F 値の下がってしまう積集合による演算でも，適合率と再現率の調節という観点で有用であると考える．

6. 関連研究

本論文では，複数の学習器を用いて属性値を抽出し，それらの集合演算を取る手法を提案した．複数の学習器を使った既存の手法には，複数の学習器からの投票を行う Voting や，学習器の組み合わせを学習により制御する Stacking[11]等が挙げられる．

宇津呂ら[12]，岩倉[13]は日本語固有表現抽出タスクにおいて，Stacking により単独の学習器を使うよりも高い抽出精度が得られたことを報告している．しかし，この手法は訓練データとして，抽出器作成のためのものと，Stacking 学習用のためのものの２種類を必要とする．日本語固有表現抽出タスクのように分類すべきクラス数が少量の場合は，各クラスの訓練データを豊富に確保できるが，本タスクのようなクラス数が膨大である場合は訓練データの確保が困難であり，２段の学習を用いて抽出精度を高めることは難しいと考える．

7. まとめと今後の課題

本研究では，百科事典の項目を約 150 のカテゴリに分類し，それぞれについて説明文の中から人手で属性を設定した．その属性の属性値を，教師あり機械学習を用いて百科事典テキストから抽出を行うタスクを設定し，複数の手段で学習を行い複数の抽出器を作成し，それらが抽出した属性値に和集合や積集合などの集合演算を行うという手法を提案し，単一の学習方法のみを用いた場合に対してのF値の 3.5 ポイント程度の向上と，集合の演算方法により適合率と再現率の調節ができることを示した．

残念ながら現時点での属性値抽出の評価値は必ずしも高いとは言えないため，評価値の向上が当面の課題となる．また，抽出器の集合演算の方法は，分類すべきクラス数が膨大である拡張固有表現のタギングなど，自然言語処理の他のタスクにも応用可能であると考えられる．それらのタスクにもこの手法を適用し，その有効性を検証していきたいと考えている．

謝辞 百科事典を提供していただいている株式会社ネットアドバンスの方々に感謝いたします．また，データ作成を担当していただいている竹内康介さんにも感謝いたします.

参考文献

1) 高橋哲朗，乾健太郎，松本裕治. :テキストから属性関係を抽出する，情報処理学会研究報告，2004-NL-164 (2004).
2) 新里圭司，関根聡，吉永直樹，鳥澤健太郎 : 固有表現抽出手法を用いたレストラン属性情報の自動認識，言語処理学会第 12 回年次大会 (2006).
3) 徳永耕亮，風間淳一，鳥澤健太郎：ＨＴＭＬ文書からの属性語の自動抽出，言語処理学会第 11 回年次大会 (2005).
4) 阿辺川武，奥村学 : 形容詞を用いた対象・属性名詞対の収集および分析，言語処理学会第 12 回年次大会(2006).
5) 飯田龍，小林のぞみ，乾健太郎，松本裕治，立石健二，福島俊一 : 意見抽出を目的とした機械学習による属性-評価値対同定, 情報処理学会研究報告, NL165-4 (2005).
6) 関根聡 : 百科事典を対象とした質問応答システムの開発，言語処理学会第 9 回年次大会 (2002).
7) 関根聡，須藤清，安藤まや : 属性値の自動抽出と質問文パターンを使った百科事典質問応答システム，言語処理学会第 11 回年次大会 (2005).
8) Tjong Kim Sang, E. F. and Veenstra, J. : Representing text chunks, Proc. EACL'99, pp. 173-179(1999).
9) Sekine, S., Nobata, C. : Definition, dictionaries and tagger for extended namedentity hierarchy. In: Proc. LREC '04 (2004).
10) 新納浩幸，関根聡 : 拡張固有表現タガーの作成とその問題点の考察，言語処理学会第 12 回年次大会 (2006).
11) Wolpert, D. :Stacked Generalization, Neural Network 5(2), pp.241-260 (1992).
12) 宇津呂武仁，颯々野学，内元清貴 : 正誤判別規則学習を用いた複数の日本語固有表現抽出システム出力の混合，自然言語処理，Vol.9,No.1, pp. 65-100,2002
13) 岩倉友哉 : Stacking の効率的な学習方法と日本語固有表現抽出での評価，情報処理学会研究報告，2005-NL-167 (2005).