自然言語処理 140－13
（2000．11．21）

固有名抽出のための短縮形知識の探索手法

鈴木 伸哉　劉 連文　桝井 文人　河合 敦夫　椎野 努
三重大学工学部
{suzuki, liu, masui, kawai, shiino}@shiino.info.mie-u.ac.jp

本論文では，固有名抽出における参照関係解析の補助手段として，英文から短縮形とその実体との対応関係を探索する手法について述べる．短縮形の代表的なものとして頭字語を取り上げ，括弧書き表記を伴う表現から頭字語と実体を抽出する手法を提案した．ある頭字語と実体とが同一であることを認識するためには，対応付けの処理が必要である．そのために頭字語化の規則について分析を行い，頭字語の文字数と実体の単語数とを対応付けする手法と，それを補完する手法を考案した．手法を実装して抽出実験を行った結果，Ｆ値で約８０の精度が得られており，本手法の有効性が確認できた．

キーワード：情報抽出，固有名抽出，短縮形，頭字語，参照関係解析

The Search Technique of the Contracted Form Knowledge for Proper Name Extraction

Shinya SUZUKI　Lian Wen LIU　Fumito MASUI　Atsuo KAWAI　Tsutomu SHIINO
Faculty of Engineering, Mie University
{suzuki, liu, masui, kawai, shiino}@shiino.info.mie-u.ac.jp

This paper describes a method that specifies the relation between the contracted forms and the entities in English texts to assists reference analysis in the proper name extraction. The method deals with acronyms as a typical form. We propose a method for extracting the entity from the notation, which includes the entity and the acronym in parentheses. It requires connecting process to recognition an acronym and an entity is same. We analyzed the rule of the contraction and devised an algorithm relating the number of acronym and the number of entity. Moreover we added several rules to the algorithm in order to complement it. The result of the evaluation was about 80 score of the F-measure. Therefore the validity of the technique was confirmed.

Keywords: information extraction, proper named extraction, contracted form, acronym, co-reference analysis

1 はじめに

今日，情報技術の進歩やインターネットの普及により，大量の文書が電子化されている．しかしユーザにとって必要な情報はそのうちの一部でしかなく，いかに効率よく取り出すかが課題となっている．そのような課題に対応する技術として情報抽出，情報検索，文書要約の技術がある．これらの技術のなかでも，情報抽出は情報検索や文書要約にも関連する重要な技術である．情報検索が文書からユーザにとって必要な情報を探し出す技術であるのに対し，情報抽出では文書中の主要な情報を取り出すことを目的としている．取り出した情報は，検索や要約のキーワードに利用することができる．

情報抽出に関する国際会議としては米国で行われたMessage Understanding Conference(MUC)や，日本で開かれたInformation Retrieval and Extraction Exercise(IREX)などがあり，与えられたタスクに対してさまざまなシステムが考えられている[1][2]．1998年に開催された第7回の会議MUC-7において，Named Entity(NE)，Co-reference(CO)，Templete Element(TE)，Templete Relation(TR)，Scenario Templete(ST)という5つのタスクが設定された．これらは内容理解のための段階的なタスクとなっている．NEでは固有名の抽出を行い，組織名や数量表現といった要素の認識をする．COでは固有名間の参照関係の認識を行う．TEでは属性的情報を記述または参照している名詞句や，場所を示す情報を抽出し，TRにおいてTEで認識された要素間の関係を抽出する．STではシナリオとして設定されたイベントを抽出する[3]．これらのタスクのうち，NEに関しては研究が進められているにも関わらず，短縮語の抽出に関してはほとんど研究が進んでいないのが現状であり，今後の課題として挙げられている[6]．

短縮語とは長い言葉を短縮した語である．短縮語は文書におけるキーワードである場合が多く，きわめて重要な位置を占めている．短縮語とその元になる語（実体）が同一文書内にあるとき，その参照関係を認識できれば短縮語が固有名として抽出できる．ある短縮語と実体とが同一であることを認識するためには，対応付けの処理が必要である．そのためには，短縮化の規則について分析を行わなければならない．そこで本研究では，実体が短縮される規則を見つけだし，固有名抽出へ応用することを目的として，短縮語の一種である頭字語と実体の対応付けについて提案する．文章中で頭字語が定義される際の表記には，“Input Method(IM)”のように括弧書きによって実体が併記されるという特徴がある．本研究ではこの点に注目し，頭字語と実体との対応付けによって抽出する手法を提案する．

以下2章では頭字語について説明し，文章中で頭字語が定義される際の表記について述べる．3章では頭字語とその実体の抽出手法について詳述する．4章では，提案した抽出手法の適用実験を行い，5章で考察を行う，

2 頭字語と表記パターン

頭字語とは，複数の単語からなる名詞を短縮したもので，通常は頭文字を取って作られる．また，頻繁に使われる用語を便宜上短縮したり，長い語を縮める役割がある．頭字語として短縮されるものには，規格名・組織名・人名などがある．例えば“WAI”は“Web Accessibility Initiative”の頭字語である．実際に文章中で頭字語が定義される場合，“Input Method(IM)”のように括弧書きを伴って実体が併記されるという特徴パターンがみられる．

本研究ではこの特徴を利用して抽出処理を行う．表記のパターンとしては，大きく分けて次の2つがある．

- パターン1：頭字語（実体）
  例：Web Accessibility Initiative (WAI)
  IM(Input Method)
  頭字語に続けて実体が括弧書きされる．

- パターン2：実体（頭字語）
  例：WAI(Web Accessibility Initiative)
  Input Method(IM)
  実体の後に頭字語が括弧書きされる．

この他，付加情報が表記される場合，例えば“ANSI X3.135-1992(American National Standards Institute, 1992)”などの特殊なパターンもあるが，出現頻度が低いため，本論文では対象としない．

3 抽出手法

本章では実体と頭字語の対応付けを行う手法について述べる．

3.1 実体との対応付け手法

頭字語の特定は，“大文字で構成される単語である”という特徴を利用する．一方，実体は複数の単語からなっているため，複数の単語列から範囲を特定する処理が必要となる．以下，実体を特定する手法について述べる．

2章で挙げたパターンのうち，パターン1では実体は括弧で区切られているため，複雑な処理を必要とせずに抽出が可能である．抽出は以下の様に行う．

- 頭字語の先頭1文字と，括弧内の最初の1文字が等しければ，括弧の直前の1語と括弧内の語すべてを抽出

例えば“RDML (relational database manipulation language)”という文字列があった場合，頭字語として“RDML”を抽出し，実体としては括弧の中の“relational database manipulation language”を抽出する．

パターン2では，パターン1のように範囲を限定する手がかり（括弧など）がないため，まず範囲を特定する処理が必要となる．ここで頭字語の構造を考えると，実体の各単語の頭文字から作られる場合が多いため，この場合頭字語の文字数はそのまま実体の単語数を表していることになる．これに当てはまるパターンは約半数を占めるので，まず基本的な手法として次のようなものを用いる．

1. 頭字語に含まれる大文字数をカウントし，その分だけ直前の数語を実体候補とする
2. 頭字語候補と実体候補それぞれの先頭1文字が等しければ，出力

例えば文字列として“... Internet service providers(ISPs)”があった場合，頭字語として“ISPs”を，実体としては“Internet service providers”を出力する．

しかしこの手法では約半数が正しく抽出できないため，補完する必要がある．以下で変則バリエーションを挙げ，補完の手法について述べる．

3.2 変則バリエーションに対する補完の手法

頭字語形成における変則バリエーションには以下のものがある．各バリエーションに対する補完の手法について説明する．

1. 記号による単語分割
2. 単語数を表す数字
3. 頭文字が小文字の単語
4. Exで始まる語
5. 頭字語と単語との組み合わせによるもの

1の場合では，スラッシュ'/'やハイフン'-'で単語を分割する処理を行う．例えば“human-machine interface(HMI)”の場合，ハイフンによって2つの単語に区切ることができる．また，区切る必要がない場合もあり，最初に区切った場合で対応を取り(例では'h', 'm', 'i')，取れなかったときに区切らず対応付けをとる（'h', 'i'）ようにした．

2の場合は，数字が単語数を表すものとそうでないものがある．単語数を表す例としては，“World Wide Web Consortium(W3C)”がある．この場合は数字'3'を単語数として展開することで計4単語を抽出することになり，“World”，“Wide”，“Web”“Consortium”の4語が正しく抽出できる．

3の場合は，“Center for Educational Computing(CEC)”のようなものである．これは“for”のように小文字で始まる単語を無視することとする．

4の場合は特殊なものとして，Exで始まる単語はXが頭字語に現れるようなものである．例えば“Extensible Stylesheet Language(XSL)”である．これはXを候補に入れることで解決できる．

5の場合は頭字語と単語との組み合わせによって新たな頭字語が作られる場合で，例えば“XSL Transformations(XSLT)”などがある．この場合は実体の大文字数を数え，頭字語の大文字数と一致したときに抽出する．

4 実験と評価

3章で述べた抽出手法を検証するために，以下の実験を行った．基本アルゴリズムと1～5の補完ア

ルゴリズムを実装し，対象文書に対して抽出処理を行った．

4.1 実験対象

対象文書は英文とし，WWW上で公開されている情報分野に関するニュース記事や論文などの，HTML文書やテキスト文書を用いた．情報分野を選んだのは，進化の速い分野であり，新たに作られる頭字語が多いという傾向が強いためである．まず，プログラムの精度向上のためトレーニングデータとして２０文書を用意し，評価用のテストデータとして１４文書を用意した．このうち，人手によって抽出した括弧を伴う頭字語の定義部分を正解データとした．

- トレーニングデータ (20文書)
  平均単語数：2988.1単語 (59762単語/20文書)
  正解数：143個 (7.15個/文書)，出現率：0.2%

- テストデータ (14文書)
  平均単語数：5100.1単語 (71402単語/14文書)
  正解数：108個 (7.7個/文書)，出現率：0.15%

パターン１とパターン２の内訳は表1のようになっている．

表 1: パターンの比率（%）

	A(*1)	B(*2)	A+B
Pattern1	5.6	4.6	4.0
Pattern2	90.8	88.9	90.0
Etc.	3.5	7.4	5.2

*1: トレーニングデータ，*2: テストデータ

表においてAはトレーニングデータを指しており，Bはテストデータである．A+Bは両データを総合したものである．パターン１の正解数は総合で１０個であり，4.0%であった．パターン２は総合で２２５個であり，90%であった．その他（パターン１と２以外）は総合で１３個であり，5.2%であった．

4.2 評価方法

プログラムの出力結果と，人手による正解データの比較によって評価を行った．評価では，頭字語とその実体がともに完全に抽出されているものを正解とした．

評価尺度には再現率（Recall）と適合率(Precision)を用いた．両者を求める計算式を以下に示す．

$$Recall = \frac{\text{抽出できた正解数}}{\text{人手による正解数}}$$

$$Precision = \frac{\text{抽出できた正解数}}{\text{全体の抽出数}}$$

また，再現率と適合率を総合して評価するために，F-measure（F値）も用いた．F-measureはMUCやIREXでも使用された評価尺度である．

$$F = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

4.3 評価結果

評価結果を表2に示す．

表 2: 評価結果

	A(*1)	B(*2)	A+B
再現率	80.3%	63.9%	72.8%
適合率	98.3%	79.3%	90.1%
F	88.4	70.8	80.5

*1: トレーニングデータ，*2: テストデータ

実験の結果，トレーニングデータでは114個が正しく抽出された．再現率は80.3%（114/142），適合率98.3%（114/116）である．テストデータでは69個が正しく抽出され，再現率63.9%（69/108），適合率79.3%（69/87）となった．F値はそれぞれ88.4, 70.8となっている．両データを総合すると，再現率72.8%（182/250）適合率90.1%（182/202）であり，F値は80.5となっている．なお，抽出成功例と失敗例をそれぞれ表3と表4に示す．

5 考察

総合的にはF値が80を越えており，概ね良好な結果であった．本論文で示した手法に関して，正し

く抽出できていることが確かめられた．再現率と適合率をみると，適合率に関してはトレーニングデータで98%，テストデータで79%と非常に高い値が出ている．一方再現率は80%と64%であるが，適合率と比較すると15%以上も低くなっている．これは精度の高さに比べてプログラムによる抽出数が少ないことを示している．実際には不正解も含めた抽出数は，トレーニングデータで１１６個，テストデータで８７個であり，人手による正解数を下回っている．

正しく抽出できなかったのは，本手法では対応できないような複雑なルールを必要とするケースである．例えば“Datatypes for DTDs(DT4DTD)”では，“Datatypes”という１つの単語から“DT”という２語が頭字語に出現している．そのため既存のルールでは正しく抽出することができない．解決法としては，“data”と“types”の２つに単語を切り分ける手法が考えられる．

また，“Universal Coded Character Set (UCS)”では，“Coded”もしくは“Character”が頭字語に現れていないため頭字語の文字数と実体の単語数が等しくない．従って本手法では適用できず，正しく抽出されない．この問題の解決には，品詞情報を利用して頭字語には現れないような単語を認識し，対応付けを行う手法が考えられる．

6 おわりに

本論文では，固有名抽出における参照関係解析の補助手段として，括弧書き表記を伴う英文頭字語表現を抽出する手法を提案した．頭字語の文字数と実体の単語数とを対応付けする手法と，それを補完する手法を考案した．本手法を実装し，抽出実験を行った．評価結果から，Ｆ値で約８０の精度が得られており，本手法の有効性が確認できた．

今後は対象文書を拡大し，さらに様々な視点から本手法の検証を進める．抽出精度が十分に向上した後は，固有名抽出への有効性を検証する予定である．その後，日本語テキストに対しても適用していきたいと考えている．

参考文献

[1] 桝井文人，関根聡: TIPSTER Text Program Phase III 24-Month Work Shop 参加報告，信学技報 vol.99，pp.23-30, 1999

[2] 福本淳一，関根聡，江里口善生: MUC-7, Tipster 参加報告，1998

[3] 福本淳一：情報検索について，人文学と情報処理第２１号，1999

[4] 関根聡，江里口善生：IREX-NEの結果と分析，言語処理学会第６回年次大会ワークショップ論文集，pp.25-32, 2000

[5] 森辰則：情報抽出に期待すること，言語処理学会第６回年次大会ワークショップ論文集，p.64, 2000

[6] Andrew Borthwick:A Japanese Named Entity Recognizer Constructed by a Non-Speaker of Japanese，IREXワークショップ予稿集，1999

[7] 鈴木伸哉，劉連文，桝井文人，河合敦夫，椎野努：固有名抽出における頭字語の解析，電気関係学会東海支部連合大会講演論文集，2000

表 3: 抽出成功例（一部）

	入力文のうち頭字語を含む部分	抽出結果（頭字語，実体）
Pattern1	IETF(Internet Engineering Task Force)	IETF,Internet Engineering Task Force
	WSD(Writing System Declaration)	WSD,Writing System Declaration
	HNE(Heure Normale de l'Est)	HNE,Heure Normale de l'Est
Pattern2	Internet service providers(ISPs)	ISPs,Internet service providers
	Frequently Asked Questions(FAQ)	FAQ,Frequently Asked Questions
	Draft International Standard(DIS)	DIS,Draft International Standard
	XML Inclusions(XInclude)	XInclude,XML Inclusions
	NaVigation Markup Language(NVML)	NVML,NaVigation Markup Language
	human-machine interface(HMI)	HMI,human-machine interface
	Left-to-right mark(LRM)	LRM,Left-to-right mark
	run-time libraries(RTLs)	RTLs,run-time libraries
	user/vendor-defined area(UDA)	UDA,user/vendor-defined area
	queued I/O(QIO)	QIO,queued I/O
	World Wide Web Consortium(W3C)	W3C,World Wide Web Consortium
	Center for Educational Computing(CEC)	CEC,Center for Educational Computing
	Extensible Markup Language(XML)	XML,Extensible Markup Language
	Extensible Stylesheet Language(XSL)	XSL,Extensible Stylesheet Language
	XSL Transformations(XSLT)	XSLT,XSL Transformations
	Language ID(LangID)	LangID,Language ID
	Extended UNIX Code(EUC)	EUC,Extended UNIX Code

表 4: 抽出失敗例（一部）

	入力文のうち頭字語を含む部分
Pattern1	Rdb (relational database application)
	ABBR(abbreviation)
Pattern2	Richard Stallman(RMS)
	Broadcast Music (BMI)
	East Japan Railway Co.(JR East)
	Issue 2 of its X/Open Portability Guide (XPG2)
	Timed Interactive Multimedia Extensions for HTML(HTML+TIME)
	Central and Communications TRON (CTRON)
	Document Content Description for XML(DCD)
	XML Path Language(XPath)
	Universal Multiple-Octet Coded Character Set(UCS)
	Universal Coded Character Set (UCS)
	Wireless Application Protocols Forum(WAP Forum)
	Datatypes for DTDs(DT4DTD)
	third-generation language (3GL)
	character manager (CMGR)
	input method library (IMLIB)
	screen management (SMG)
etc.	Cascading Style Sheets (CSS1) Level 1
	OSF/1 Release 1.1 (Cambridge MA: Open Software Foundation Inc. 1992)
	(SGML: An Author's Guide to the Standard Generalized Markup Language 92-93)
	KODA (the kernel of the data access, the lowest layer of the physical data access)