辞書を使わない日本語専門用語の自動分割

森脇 敏　河部 恒　辻井 潤一
東京大学理学部情報科学科
{moriwaki,kawabee,tsujii}@is.s.u-tokyo.ac.jp

1 はじめに

日本語専門用語には、名詞が合成してできた複合名詞となっているものが多く含まれている。そこである分野の専門用語の基本的な要素、すなわち要素語を抽出することは、新出・未知の用語を分析する際や、シソーラスを構築する際の足掛かりとなり、重要である。

従来の同様の研究では、要素語をエントリーとして含んだ辞書をあらかじめ用意して置くことが前提になっていた [1] が、辞書の更新が難しいことや、専門用語への対応が不十分であることが問題である。

そこで、我々は、辞書を全く用いずに分析することを提案する。これにより、辞書の性質によらず分析できるだけでなく、自動的に辞書を作成・更新することも可能になる。

本研究では、ある分野の専門用語の集合のみから得られる、頻度情報や対訳情報を用いることにより、辞書を全く用いずに基本語の抽出を行うことを主眼とした、2つのアルゴリズムを提案する。1つは、専門用語を1つの文字列と見て、部分文字列の要素語優先度を調べ、要素語候補集合を作成しながら切っていく下降型のアルゴリズムであり、もう1つは、1文字単位から専門用語を組み上げていく上昇型のアルゴリズムである。各々に対する実験・評価を行ない、その性能について考察する。

2 下降型語分割アルゴリズム [2]

2.1 漸進的下降型語分割アルゴリズム

ここでは、専門用語の単語リストのみを入力として、top down 的に要素語を切り出していく方法を紹介する。この方法は、コーパスに対する仮定を一切おかず、辞書や対訳情報も用いていないので、汎用的である。実際、入力が日本語である必要もないのでドイツ語の複合名詞の分割等にも応用できるだろう。

このアルゴリズムは二つの部分からなっている。一つは、長い専門用語からそこに含まれる短い要素語の候補を作り出す部分 (要素語の切りだし) で、もう一つは、その要素語の候補を使って、専門用語の適切な分割を決定する部分 (分割の決定) である。

専門用語は短い文字列が組み合わさって長くなっているという特徴を持つので、二つの長い専門用語を比較すると、その共通部分が要素語となっている場合が多い。要素語の切りだしでは、上記の性質を利用し、要素語の候補をつくり出す。いま、「二酸化炭素」と「炭素」が要素語の候補だとすると、それらの差分文字列「二酸化」も要素語の候補だとみなす。このように用語をお互いに比較していって、差分文字列を漸進的に抽出していく。(図1参照)

この時、ただ差分文字列を抽出していったのでは、最終的に漢字一文字にまでバラバラになってしまうので、それに対処するために、各要素語の候補に優先度 age をつける。age は、もともとあった候補ほど大きく、後から生まれた候補は、要素語である可能性は低いとみなして、小さくなるように付ける。上の例では、「二酸化」の age は、「炭素」と「二酸化炭素」より生まれたので、一つ小さくする。この値は分割の決定のところで使われ、うまくバラバラになるのを防いでいる。

以下に詳しいアルゴリズムの説明をする。まず表記法を以下のように定める。

アルファベットの小文字は、漢字の列である。

$|a|$ は漢字列 a の長さである。

"/" は、連結を表す。つまり、a と b が漢字列なら、a/b はそれらの連結を表す。

$a^{l,m}$ は、漢字列 a の l から m 番めまでの部分漢字列である。[1]。

$age(a)$ は、a の優先度を表す値である。[2]

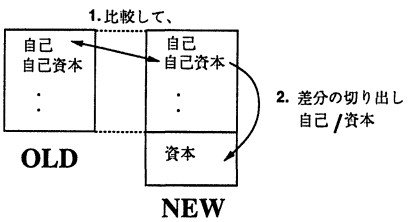

図 1: 要素語の切りだしの様子

[1] もし $l > m$ ならば、$a^{l,m}$ は空列とする。

OLD は、要素語の現在の候補、NEW は、要素語の新しい候補である。

要素語の切りだし

- **step 0**
  OLD_0, NEW_0 を入力の用語のリストとする。$\forall\, a \in OLD_0$, $age(a) = 1$. $cycle = 0$.
- **step 1**
  $\forall\, a \in OLD_{cycle}$,$\forall\, b \in NEW_{cycle}$,
  $NEW_{cycle} := \{k_1, k_2 |\ b = k_1/a/k_2$,
  $k_1 = b^{1,m}, k_2 = b^{m+|a|+1,|n|}\}$.
  $age(k_1) = age(k_2) = 0$.
  新しい k がみつからなくなったら、step 2 へ進む。そうでなければ step 1 を繰り返す。
- **step2**
  $OLD_{cycle+1} = NEW_{cycle}$.
  $\forall\, a \in OLD_{cycle+1}$, $age(a) := age(a) + 1$.
  NEW_{cycle} の数が増えなくなったら止まる。
  そうでなければ、$cycle$ に 1 を加えて、step 1 へいく。¶

要素語の候補の集合 $OLD_{cycle+1}$ がアルゴリズムの出力である。age の値をつけるために、切り出されたものは NEW のみに入り、OLD ($\subset NEW$) だけを使って切っていくことに注意。そして切り出されるものがなくなってはじめて、$OLD := NEW$ と更新し、同時に age の更新も行なっている。

分割の候補の決定　分割したい専門用語を W とすると、分割の候補

$$S = \{a_i |\ W = a_1/a_2/.../a_n,\ a_i \in OLD,\ n > 1\}$$

に対し、平均

$$Avr(S) = \frac{\sum_{i=1}^{n} age(a_i)}{n}$$

を計算し、MAX $Avr(S)$ をもつ S を W の分割として選択する。もし Avr の値が同じだったら分割数の少ない方を選択する。¶

例えば、「言語処理学会」 に対して、言語 (age = 5)/ 処理 (4)/ 学会 (5) と 言語 (5)/ 処 (2)/ 理 (1)/ 学会 (5) の二つの候補があった時、前者が選択される。

[2]この値は繰り返し処理がすすむにしたがって増えていくので、"age" と呼ばれる。

2.2 実験

実験には以下の 2 つの専門用語リストを用いた。

- 経済用語 4759 語、平均長 4.82
- 化学用語 10561 語、平均長 4.73

評価方法としては、4 名にあらかじめ 500 個の専門用語を見せて分割の正解をつくってもらい、その各々の結果とシステムの出力とを比較した。正解を C, システムの出力を A をすると、

$$precision = \frac{|A \cap C|}{|A|}$$

$$recall = \frac{|A \cap C|}{|C|}$$

として、ここでは、切り出された部分文字列が正例とどれだけ一致したかどうかを基準にして precision , recall 及び正解率 (部分文字列が全て正解した率) の 3 つを算出した。

つまり、「言語 / 処理 / 学会」を正解とする時、システムの出力が「言語 / 処 / 理 / 学会」だとすると、部分文字列「言語」、「学会」が正しいので、precision = 2/4, recall = 2/3 , 正解率 = 0 となる。
結果は以下のようになった。

	経済	化学
正解率	49%	48%
Precision	64%	56%
Recall	61%	55%

表 1: 漸進的下降型アルゴリズムの性能

2.3 考察

このアルゴリズムでは分割するかどうかの判定はできないので、「ヘリウム」などのような本来分割しない用語も「へ / リ / ウ / ム」のように分割してしまい、それが正解率を下げる原因になっている。また、経済用語中の「金」や化学用語中の「酸」のような頻度の非常に高い 1 文字漢字が、「金 / 融」、「酸 / 性」といった分割をしてしまうという問題点もある。

3 上昇型語分割アルゴリズム

3.1 上昇型語分割アルゴリズム

日本語の複合名詞は、意味的な分割を行なえば、究極には全て 1 文字ずつのレベルまで分割ができるはずである。

そこで、この1文字のレベルから、分割の逆の操作、即ち構築を行ない、もとの複合名詞まで再構成したとき、構築の途中のレベルで求める分割の解が出現しているはずだ、というのが基本的な発想である。

具体的には、次のようなアルゴリズムで実現される。

1. 初期状態において 構成要素は各々1文字。
2. 構成要素同士の結合する確率 (結合度) を求め、一番高いものを結合させる。
3. 構成要素が1つになるまで繰り返す。

実際の例を 図2に挙げる。

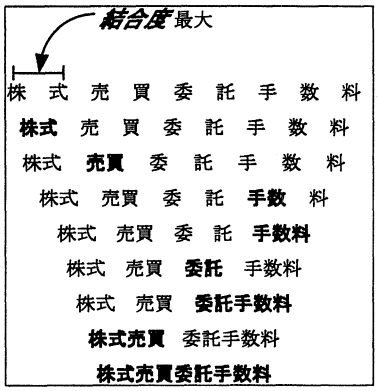

図 2: 上昇型語分割の例

3.1.1 上昇型語分割アルゴリズムの問題点

終了条件が不明 ある段階で構築を中止すれば解が得られている可能性が高いが、その段階の具体的な指標を決められない限り、解は得られない。

3.2 対訳対応上昇型語分割アルゴリズム

上記の上昇型語分割アルゴリズム に、日本語専門用語と対応した英語表現との両方を含んだ用語集を仮定し、そこから得られた対訳情報を採り入れたものが対訳対応上昇型語分割アルゴリズムである。

英語の要素語に対応させることにより「英語要素語と日本語の断片との対応が全てとれ次第終了する」という終了条件を明示的に規定できるというメリットがある。

対応制約 英語との対応関係をとるうえで、今回は次の制約を仮定した。

- 英単語一つは連続した日本語断片に対応する。

株	式	売	買	委	託	手	数	料
株式:stocks		売	買	委	託	手	数	料
株式:stocks		売	買	委	託	**手数:commission**		料
株式:stocks		売	買	委	託	**手数料:commission**		
株式:stocks		**売買:brokerage**		委	託	手数料:commission		
株式:stocks		売買:brokerage		委	**託手数料:commission**			
株式:stocks		売買:brokerage		**委託手数料:commission**				

図 3: 対訳対応分割 (例：株式売買委託手数料と"brokerage commission for stocks" の対応より)

- 英単語の冠詞、助詞、助動詞に対応する日本語断片はない。[3]
- その他の英単語について、1つの単語は、1つの要素語である。

これらの仮定により、単語の対応を考察していくことで、要素語となる日本語断片の抽出が可能になる。

アルゴリズム

結局、アルゴリズムは次の通りになる。

1. 初期状態で構成要素は各々1文字。また、この用語 J に対応する英語表現 E を入力として与える。
2. 隣あった構成要素 j_i 間の結合の仮説を立てる。
   $(j_1j_2, j_2j_3, \cdots, j_{m-1}j_m)$
3. 仮説に 対応制約を満たす E 中の英単語 e_i との対応関係の情報を加える。
   $((j_1j_2, e_1), (j_1j_2, e_2), \cdots, (j_{m-1}j_m, e_n))$
4. 対応関係の有意性、即ち対応度をもとに、もっとも対応関係が強いものを選び、仮説を実際の結合に反映させる。
5. 構成要素と英語要素語との対応が全てとれ次第終了する。

対応度 今回、対応関係の有意性を示す尺度として、次のものを採用した。

$$\begin{aligned} A_{je} &= p_1(j,e) * p_2(j) \qquad (1) \\ &= (alignment\text{仮説の出現確率}) * \\ &\quad (\text{日本語断片の出現確率}) \end{aligned}$$

それぞれの記号の具体的意味は次の通りである。

[3]実際には存在しないことは自明ではないので、判定を行なうべきである。

$p_1(j,e)$: 日本語断片 j と英単語 e とが同時に出現する確率 (頻度)。

$p_2(j)$: 日本語断片 j の出現確率。朝日新聞一年分の記事から抽出した、漢字列 約 9,250,000 個の断片の頻度を調べたものである。

A_{je} : 対応度。日本語断片 i と英単語 j の組合せが実際の対応に用いられる確率。

3.2.1 実験と評価

前述と同じく、要素語に分割した正例を 500 用意し、求められる要素語がどれだけ抽出できたかを正解率、precision、recall によって評価した。評価方法は次の 3 つである。(結果: 表 2)

- 正解率。正例と完全に一致した確率。
- 切り出した部分文字列が正例とどれだけ一致したかを基準とした、要素語の一致についての precision、recall。
- 切れ目の一致についての precision、recall。日本語として意味のない単位は切り出さないというシステムの信頼度の高さを示す。

	経済	化学
正解率	0.65	0.64
要素語 (precision/recall)	0.80/0.76	0.73/0.71
切れ目 (precision/recall)	0.95/0.81	0.85/0.75

表 2: 対訳対応上昇語分割アルゴリズムの性能

正解率が 60 数%でとどまっているのに対し、切れ目の precision が高い (94 %) ことが特徴的である。これは、切る位置は正しいが、切り方が足りないことを示している。

3.2.2 問題点

- 英単語 1 つが日本語の phrase に対応する意味を持つ時、それ以上分割できない。
- 化学で成績が落ちているのは、カタカナ語・ひらがな語の処理に失敗しているためである。かな文字の意味的な独立性が小さいため、致し方ない。
- 同様に、日本語 1 文字 が 独立した 要素語である場合、これを抽出できない。日本語 1 文字について英単語との対応について述べるには、情報が少なすぎるためである。[4]

[4]例えば、apartment と「団地」の関係はある程度重視されるが、apartment と「地」の関係は全く自明でなく、「地」自体の情報を利用することができない。

4 比較

対訳対応を採用するかどうかを考察すると、対訳情報を用いたアルゴリズムでは、結果を対訳辞書に転用できる利点がある。しかし、精度を上げるためにはある専門分野の対訳コーパスが大量に必要となり、限界が生じる。その点、日本語のみで行なう場合、電子化されている文書があれば自動的に専門用語を抽出する手法などにより精度を上げることは容易になる。

5 おわりに

本研究では、辞書を用いずに専門用語を分割する方法として、漸進的型下降型アルゴリズムと対訳対応上昇型アルゴリズムを提案した。精度は前者が約 50%、後者が約 60% と後者が良い結果が出たが、前者では日本語のみのコーパスを用いているので汎用性は高い。
今後の課題としては、以下のものがある。下降型アルゴリズムでは、対訳情報を用いて精度を上げることが考えられる。上昇型アルゴリズムでは、(1) スタートを 1 文字とする前提をなくすことでひらがな、カタカナ、1 文字のみからなる要素語に対応する、(2) 漸進的なアプローチにより精度を上げる、(3) 対訳情報を使わずに精度を上げる手法を検討するなどが考えられる。全体として、上昇型下降型の両方の利点を組合せた手法を開発する予定である。

参考文献

[1] 小林義行, 徳永健伸, 田中穂積. 名詞間の意味的共起情報を用いた複合名詞の解析. 自然言語処理, vol.3(1), January 1996.

[2] KAWABE Koh. Automatic extraction of the basic element from terminological corpora. Senior's thesis, the University of Tokyo, 1996.

[3] 小山照夫. 用語集からの要素語推定の試み. Technical Report A-5, 情報知識学会, 1995.

[4] Jingjuan Lai, Xiaojing Wang, and Yuzuru Fujiwara. 専門用語における階層関係及び関連関係抽出法. Technical Report A-2, 情報知識学会, 1994.

[5] 小川泰嗣, 望主雅子, 別所礼子. 複合語キーワードの自動抽出法. 自然言語処理, vol.97(15):103–110, 1993.

[6] 宇津呂武仁 , 松本裕治. コーパスを用いた言語知識の獲得. 人工知能学会誌, 10(2):197–204, March 1995.

[7] Kyo Kageura. Terminological semantics. An Examination of 'Concept' and 'Meaning' in the Study of Terms, 1995.