名詞句の内部構造を考慮したキーワードのスコア付け

村脇 有吾　　黒橋 禎夫
京都大学学術情報メディアセンター　京都大学大学院情報学研究科
murawaki@i.kyoto-u.ac.jp　　kuro@i.kyoto-u.ac.jp

1　はじめに

複雑な概念は、しばしば単語ではなく、単語の組み合わせによって表現される。しかし、情報分析分野では現在でも単語単位の処理が主流である。例えば、トピックモデルは、一部の例外 [1, 4] を除いて、bag-of-words を採用している。一方で、単語間の関係を捉える解析技術としては、構文解析が実用段階に入りつつあり、情報分析への応用の見込みがある。

そこで、本稿では、単語以上の意味的なまとまりを応用処理で利用するための準備として、まずそうした意味的なまとまりを自動認識するタスクに取り組む。このタスクは一般にキーワード抽出とよばれ、文書集合が与えられたとき、各文書を代表するキーワードをテキストから抽出する。この際、複数の単語からなる名詞句を認識する必要がある。タスク設定としては、正解キーワードを用意して学習を行う場合もあるが、本稿では教師なし設定に着目する。典型的な教師なし手法は、各キーワード候補に対して何らかのスコアを与え、スコア上位の候補を出力する。

教師なしキーワード抽出においては、tf-idf に基づく単純なスコア付け手法 (tf-idf 法) の精度が、ほとんど場合に、より洗練された他の手法を上回ることが報告されている [2]。tf-idf 法は単語同士の関係を考慮しない bag-of-words 的スコア付けであり、言語的手法による改善の余地があると考えられる。

そこで、本稿では、tf-idf 法を拡張し、言語解析を組み込むスコア付け手法を提案する。tf-idf 法が単語 tf-idf の総和をキーワード候補のスコアとするのに対して、名詞句の tf を直接算出する。そのために、キーワード候補の各名詞句に対して名詞句解析を行い、その結果を元に名詞句の部分単語列に対しても適当な頻度を与え、その効果を実験により検証する。

2　名詞句解析

2.1　名詞句解析モデル

単語列 $\boldsymbol{w} = w_1, \cdots, w_L$ からなる名詞句に対して内部構造を付与するタスクを名詞句解析とよぶ。具体的には図 1 のような内部構造を考える。ここで、各エッジは単語間の係り受けを表し、各スパンは意味的にまとまった部分単語列を表す。主辞後置性を仮定すると、内部構造のエッジによる表現とスパンによる表現は一対一に対応する。

名詞句解析にはエッジとスパンを用いた半教師あり手法 [5] を用いる。この手法では、図 1 の木に対して、以下のように再帰的にスコアを与える。

$$\begin{aligned}
\text{score}(0,2,3) &= \text{score}(0,1,2) + \text{score}(2,3,3) \\
&\quad + \text{edge-score}(2 \leftarrow 3) \\
&\quad + \text{span-score}(0, \cdots, 3) \\
\text{score}(0,1,2) &= \text{score}(0,1,1) + \text{score}(1,2,2) \\
&\quad + \text{edge-score}(1 \leftarrow 2) \\
&\quad + \text{span-score}(0, \cdots, 2) \\
\text{score}(0,1,1) &= \text{score}(1,2,2) = \text{score}(2,3,3) = 0
\end{aligned}$$

ここで score(i,j,k) は位置 i から k までを被覆するスコアであり、$i = 0$ かつ $k = L$ のとき名詞句全体のスコアを表す。edge-score$(j \leftarrow k)$ は単語 w_j と単語 w_k の間のエッジのスコアを、span-score$(i, \cdots, k)$ は、単語列 $w_{i+1}, \cdots, w_k$ からなるスパンのスコアを返す。edge-score と span-score は、いずれも特徴ベクトルと重みベクトルの内積である。重みベクトルは教師データから Passive-Aggressive アルゴリズムにより学習される。デコード時にはスコアを最大とする木を動的計画法により選択する。

図 2 に特徴量の一覧を示す。名詞句の解析では、(品詞レベルではなく) 語彙レベルの単語の同士の関係を認識させる。単語の組み合わせは膨大な数となるため、少量の正解データから抽出される特徴量 (**Base**, **Span**) だけでは十分な学習が行えないと予想される。そこで、これらを補完するために、大量のウェブデータから統計量を計算し、特徴量として用いる (**TWNC**, **LTW**, **Web span**)。

2.2　テキストからの名詞句抽出

名詞句解析器に名詞句を入力するには、まずテキストから名詞句を抽出する必要がある。日本語の場合は

Base
$\langle d \rangle$
$\langle l_j, d \rangle$
$\langle l_k, d \rangle$
$\langle l_j, l_k \rangle$
$\langle l_j, l_k, d \rangle$

(a)

TWNC
: $log1p(c_{\mathrm{TWNC}}(l_j, l_k))$

LTW
: $log1p(c_{\mathrm{LTW}}(l_j, l_k))$

(b)

Span
$\langle l_{i+1}, \cdots, l_j \rangle$

Web span
$\langle s \rangle$: $log1p(c_{\mathrm{SPAN}}(l_{i+1}, \cdots, l_j))$

(c)

図 2: 名詞句解析で用いる特徴量。特徴量抽出には、単語列 $\boldsymbol{w} = w_1, \cdots, w_L$ に対して、各単語を小文字に正規化した列 $l_1, \cdots, l_L$ を用いる。score(i, j, k) について、単語 l_j と単語 l_k の間のエッジ、および単語列 $l_{i+1}, \cdots, l_k$ からなるスパンに対して特徴量を抽出する。$\langle x \rangle$ は複数の特徴量に展開されるテンプレートを表す。コロンの左辺は特徴量の名前であり、自明な場合は省略する。右辺は特徴量の値であり、省略された場合 2 値である。(a) 正解データから学習される Base 特徴量。ここで $d = k - j$ は l_j と l_k の間の距離 (1, 2, 3, 4 or ≥ 5)。(b) エッジに対する 2 種類のウェブ特徴量。ここで $log1p(x) = log(1+x)$。$c_{\mathrm{TWNC}}(l_j, l_k)$ は、l_j, l_k からなる 2 単語名詞句のウェブデータにおける出現頻度。$c_{\mathrm{LTW}}(l_j, l_k)$ は、末尾 2 単語が l_j, l_k からなる名詞句の出現頻度。(c) スパンに対する特徴量。$s = k - i$ はスパンの幅 (2, 3, 4, 5 or ≥ 6)。$c_{\mathrm{SPAN}}(l_{i+1}, \cdots, l_j)$ は $l_{i+1}, \cdots, l_j$ からなる名詞句の出現頻度。

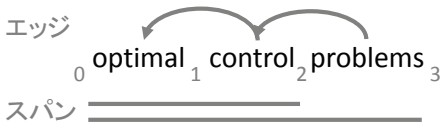

図 1: 英語名詞句の解析例

名詞句は文節に包含されるが、英語の場合は文節のようなチャンクは通常の解析では用いられない。そこで、本稿では係り受け木から規則によって名詞句を抽出する。すなわち、以下の条件をすべて満たす最長の部分単語列 $\boldsymbol{w} = w_p, \cdots, w_q$ を名詞句として抽出する。

- 主辞 w_q が名詞 (品詞タグについて正規表現 "^N*" がマッチするもの)
- w_i $(p \leq i < q)$ が名詞あるいは形容詞 ("^J*")
- エッジが $p, \cdots, q$ 内で閉じており、かつすべての親が右側
- 各エッジのラベルが NMOD, TITLE, NAME のいずれか

2.3 英語の実験データ

名詞句の正解データとして、Penn Treebank を用いた。Penn Treebank の Wallstreet Journal (WSJ) 部分に対して、まず名詞句アノテーション [6] を適用し、次に pennconverter[1] で句構造を係り受けに変換した。さらに、2.2 節で述べた手順で単語数 3 以上の名詞句を抽出した。セクション 2–21 を学習に、セクション 23 をテストに用いた。

統計量を計算するためのウェブデータとして、英語ウェブコーパス約 30 億文を用いた。まず各文に対して係り受けを自動付与した。ここで、品詞タグ付けに lapos[2]、係り受け解析に MSTParser[3] を用いた。これらのモデルの学習には WSJ 部分全部を用いた。自動解析結果から単語数 2 以上の名詞句を自動抽出し、名詞句解析のための統計量を計算した。

[1] http://nlp.cs.lth.se/software/treebank_converter/

[2] http://www.logos.ic.i.u-tokyo.ac.jp/~tsuruoka/lapos/

[3] http://www.ryanmcd.com/MSTParser/MSTParser.html

モデル	精度
すべて隣に係る	60.05
すべて最後に係る	82.12
ランダム	70.00
Base	94.40
Base + **TWNC**	96.02
Base + **LTW**	95.69
Base + **Span** + **Web span**	96.02
Base + **Span** + **Web span** + **TWNC**	**96.30**

表 1: 名詞句解析の係り受け精度

2.4 英語名詞句解析の精度

先行研究 [5] では日本語に対する実験結果のみが報告されているため、本稿では英語に対する実験結果を簡単に報告する。表 1 に、様々な特徴量を組み合わせた場合の係り受けの精度を示す。いずれの特徴量を追加した場合も **Base** と比べて精度が向上した。日本語の場合と同様に、ウェブのエッジ特徴量としては、**TWNC** が **LTW** よりも有効であった。また、スパンとエッジを組み合わせることにより、より精度が向上した。以降では名詞句解析モデルとして **Base** + **Span** + **Web span** + **TWNC** を用いる。

3 データセット

キーワード抽出の先行研究 [2] は 4 種類の英語のデータセットを用いているが、本稿ではそのうちの一つ、Inspec データセット [3] を対象とする。このデータセットには、テキストが短い (文書あたり平均 134 語) という特徴がある。

Inspec データセットは 2,000 本のジャーナル論文からなるが、このうち先行研究でテストセットとされた 500 論文を用いる。各論文には表題と要旨に加え、いく

つかのキーワードが人手により付与されている。キーワードはシソーラスによる統制語と非統制語の 2 種類からなる。このうち、非統制語を正解キーワードとみなす。各論文について表題と要旨に対してキーワード抽出を行い、抽出結果と正解キーワードの比較により評価する。なお、キーワードは小文字に正規化する。

キーワード候補となる名詞句の抽出は、先行研究 [2] は品詞列に対する規則を使って行なっている。本稿では、名詞句解析にあわせるため、各論文に対して係り受け解析までを行ったうえで、2.2 節で述べた手順で単語数 1 以上の名詞句を抽出した。この名詞句抽出手法は、正解係り受け木に対しては期待通りの結果を出力するが、自動解析結果に適用した場合問題が多い。Inspec データセットについても、先行研究 [2] と比較して、同じ tf-idf 法で数%の精度低下をまねいている。今後の検討課題としたい。

4 tf-idf 法

教師なしキーワード抽出タスクにおいて、tf-idf 法は、単純でありながら、他手法を上回る精度が報告されている [2] ため、これをベースライン手法とする。tf-idf 法では、キーワード候補 $\boldsymbol{w} = w_1, \cdots, w_L$ の文書 doc におけるスコアは以下で与えられる。

$$\text{tf-idf}_{doc}(\boldsymbol{w}) = \text{unit}_{doc}(\boldsymbol{w}) \times \text{term}_{doc}(\boldsymbol{w}),$$
$$\text{unit}_{doc}(\boldsymbol{w}) = 1 \text{ if } \boldsymbol{w} \in doc, 0 \text{ otherwise},$$
$$\text{term}_{doc}(\boldsymbol{w}) = \sum_{i=1}^{L} \text{tf}_{doc}(w_i) \times log(D/D_{w_i})$$

ここで、$\boldsymbol{w} \in doc$ は、キーワード候補 $\boldsymbol{w}$ が文書 doc において名詞句として出現すること、$\text{tf}_{doc}(w)$ は doc における単語 w の頻度、D は文書総数、D_{w_i} は w_i を含む文書数を表す。

tf-idf 法は、キーワード候補の重要性を構成単語の tf-idf スコアの総和によって測る。一方、キーワード候補の言語的一体性の判定は、名詞句として出現するか否かに依存する。これにより、スコア 0 以上のすべてのキーワード候補が意味のある単語列であることが保証されるが、抽出したいキーワードが別の名詞句の部分列としてのみ出現する場合は取りこぼす。この制限は、Inspec データセットのように対象テキストが短い場合に特に問題となるのではないかと予想される。比較のために $\text{tf-idf-all}_{doc}(\boldsymbol{w}) = \text{term}_{doc}(\boldsymbol{w})$ も考える。

5 名詞句解析に基づくスコア付け

5.1 tf の一般化

tf-idf 法を基礎として、名詞句解析結果を組み込んだスコア付け手法を提案する。まずは tf の算出単位を単語から名詞句に一般化する。文書中にある名詞句が出現したとき、その名詞句の出現頻度に 1 を加える。それと同時に、名詞句の部分単語列に対しても適当な頻度を分配する。準備として、スパンのスコア総和 scoreS をボトムアップに求める。

$$\text{scoreS}(i,k) = \begin{cases} 0 & \text{if } i+1 = k \\ \sum_{j=i+1}^{k} \text{scoreE}(i,j,k) & \text{otherwise} \end{cases}$$

$$\begin{aligned} \text{scoreE}(i,j,k) = {} & \text{scoreS}(i,j) + \text{scoreS}(j,k) \\ & + \text{edge-score}(j \leftarrow k) \\ & + \text{span-score}(i, \cdots, k) \end{aligned}$$

次に、トップダウンに頻度を分配する。いま、スパン $w_{i+1}, \cdots, w_k$ が頻度 $f_{i,k}$ $(0 < f_{i,k} \leq 1)$ を持つとする $(f_{0,L} = 1)$。まず、頻度 $f_{i,k}$ を $\text{scoreE}(i,j,k)$ に応じて $g_{i,j,k}$ に分配する。

$$g_{i,j,k} = f_{i,k} \times d \times \frac{exp(\text{scoreE}(i,j,k))}{\sum_j exp(\text{scoreE}(i,j,k))}$$

ここで d は割引係数 $(0 \leq d \leq 1)$。次に、$g_{i,j,k}$ をより小さなスパンの頻度に加算する。

$$f_{i,j} \mathrel{+}= g_{i,j,k},\ f_{j,k} \mathrel{+}= g_{i,j,k}$$

これにより、名詞句の部分単語列にも 1 以下の頻度が付与される。頻度分配には softmax 関数を使っており、モデルがより自然と考える部分単語列に頻度が傾斜配分される。割引係数 d が小さいほど部分単語列に与えられる頻度は小さくなり、$d = 0$ のときは (最長の) 名詞句の頻度だけが考慮される。こうして算出される頻度の文書あたりの総和を $\text{tf-term}_{doc}(\boldsymbol{w})$ とする。

5.2 idf 相当の尺度

idf 相当の尺度を 2 通り考え、そのそれぞれと tf-term の積をキーワード候補のスコアとする。$\text{idf-sum}_{doc}(\boldsymbol{w})$ はキーワード候補の構成単語の idf の総和とする。

$$\text{idf-sum}_{doc}(\boldsymbol{w}) = \sum_{i=1}^{L} log(D/D_{w_i})$$

idf-term は $\boldsymbol{w}$ に対して直接 idf を算出する。

$$\text{idf-term}_{doc}(\boldsymbol{w}) = log(D/D_{\boldsymbol{w}})$$

ここで、$D_{\boldsymbol{w}}$ は $\text{tf-term}_{doc}(\boldsymbol{w}) > 0$ となる文書の総数。

名詞句に対して直接 idf を算出した場合、500 論文のテストセットでは信頼できる値が得られないと見込まれる。そこで idf の算出にウェブコーパスを利用した場合 (web-idf-term) も試す。比較のために、tf-web-idf、tf-web-idf-all、および web-idf-sum も考える。

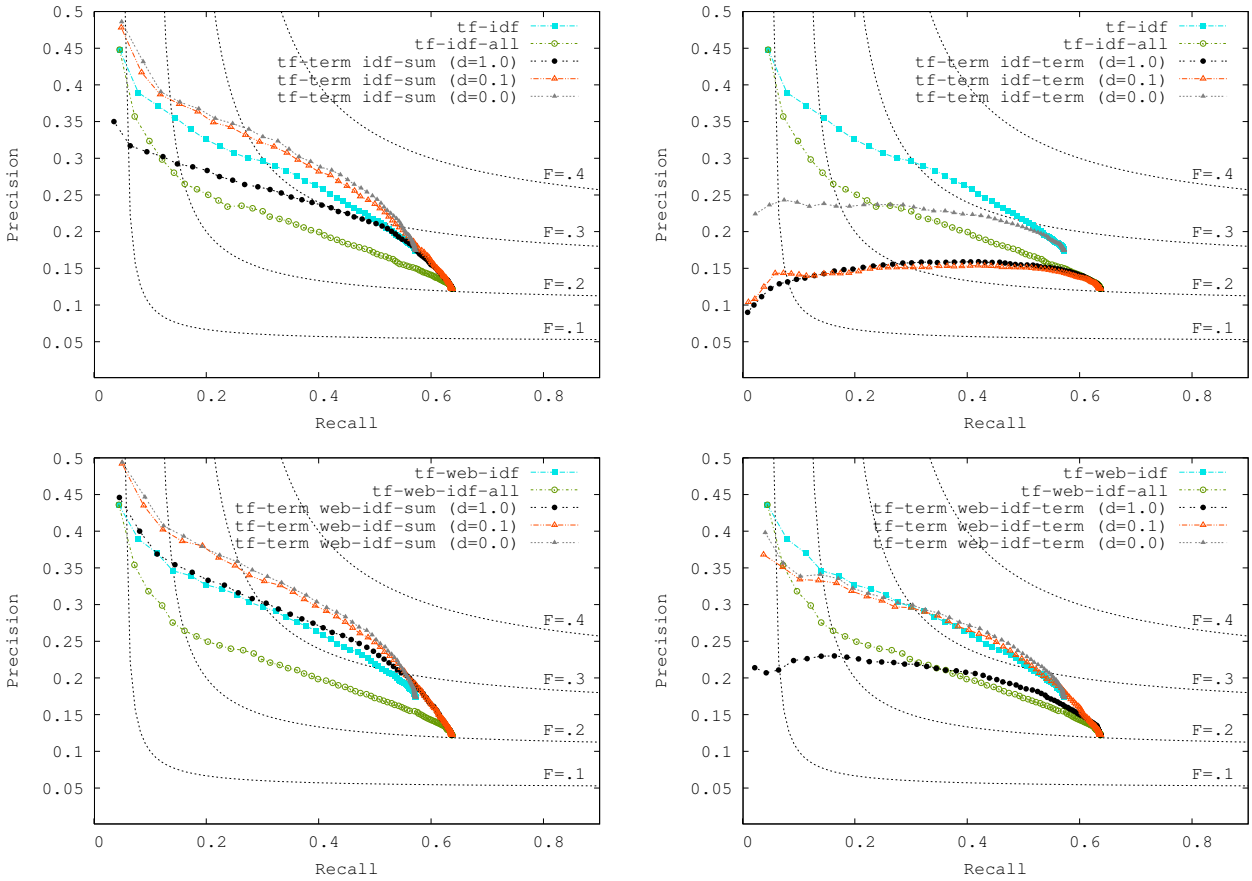

図 3: キーワード抽出の再現率–適合率曲線。文書ごとにスコア上位 N 語を出力し、その N を変化させる。

6 結果と議論

再現率–適合率曲線を図 3 に示す。idf-term は idf-sum をいずれの場合も下回っており、名詞句の idf が信用できないことを示唆する。idf 算出元の文書集合をテストセットからウェブコーパスに切り替えると、tf-idf には大きな変化はないが、名詞句単位の手法については精度が顕著に向上した。tf-term idf-sum はほとんどの設定で tf-idf-all を上回っており、名詞句解析による自然な単語列への重み付けの効果がうかがえる。しかし、割引係数 $d = 1.0$ のとき、tf-idf を下回る結果となった。

tf-term は割引係数 d が小さいほど高い性能が得られた。特に $d = 0$ の場合は部分単語列を考慮しない。この結果には次の理由が考えられる。キーワード候補 $\boldsymbol{w_a}$ とその部分単語列 $\boldsymbol{w_b}$ は意味内容的に競合関係にあり、たとえ両者とも単体ではキーワードらしくても、一覧としてはいずれかを採用すれば十分なことが多い。こうした場合、Inspec データセットには長いキーワードを優先する傾向が見られる。しかし、tf-term は、$\boldsymbol{w_a}$ が出現したとき、より短い $\boldsymbol{w_b}$ のスコアも引き上げる。部分単語列を候補に加えることによって再現率の上限が 6%ほど向上するものの、それ以上に競合関係にある短いキーワード候補の悪影響が大きい。キーワード候補に対する個別のスコア付けだけではなく、キーワード一覧の全体最適化が必要かもしれない。

7 おわりに

本稿では、tf-idf に基づくキーワードのスコア付けを名詞句の内部構造を考慮して拡張する手法を提案した。今後も、単語のみならず、より意味的に自然なまとまりを扱う取り組みを続けたい。

参考文献

[1] Thomas L. Griffiths, Mark Steyvers, David M. Blei, and Joshua B. Tenenbaum. Integrating topics and syntax. In *NIPS 17*, pp. 537–544, 2005.

[2] Kazi Saidul Hasan and Vincent Ng. Conundrums in unsupervised keyphrase extraction: Making sense of the state-of-the-art. In *Proc. of COLING*, pp. 365–373, 2010.

[3] Anette Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proc. of EMNLP*, pp. 216–223, 2003.

[4] Mark Johnson. PCFGs, topic models, adaptor grammars and learning topical collocations and the structure of proper names. In *Proc. of ACL*, pp. 1148–1157, 2010.

[5] Yugo Murawaki and Sadao Kurohashi. Semi-supervised noun compound analysis with edge and span features. In *Proc. of COLING*, pp. 1915–1932, 2012.

[6] David Vadas and James Curran. Adding noun phrase structure to the Penn Treebank. In *Proc. of ACL*, pp. 240–247, 2007.