Wikipediaのエントリ構造と編集距離を用いた専門用語抽出の試み

Attempt to Extract Technical Terms Using Entry Structure of Wikipedia and Edit Distance

◯ [1] 中山　祐輝,　[1] 南保　英孝,　[1] 木村　春彦
◯ [1]Yuki Nakayama,　[1]Hidetaka Nambo,　[1]Haruhiko Kimura
[1] 金沢大学
[1]Kanazawa Univesity

Abstract: In this paper, we propose automated term-extraction considering category of terms and similarity of string. An automated term-extraction has been proposed. Especially, they are based on appearance frequency and pattern in monolingual corpus. However, whether terms has specific category domain or not isn't cosidered in conventional method. By using category information, we think that extracted terms is extracthighly-compatible in certain domain. As category data we use entry structure of Wikipedia. Also there is a possibility that a term which is similar to such terms in string is contained in same genre. We use edit distance as similarity of stirng. In experiments, we verified efficiency of our method on IT-words test collection. As the results, it found that our method is higher F-measure than FLR method.

1　はじめに

従来における専門用語の抽出は専門家の人手によって獲得していたため，大変な時間コストがかかってしまうという問題点があった．したがって，専門用語自動抽出は最新の辞書を構築するために極めて重要な技術となっている．また，近年専門用語の出現頻度を用いたシステムが提案されてきており，専門用語を自動抽出する必要性がさらに高まってきている．そこで，専門用語を自動抽出する様々な手法が提案されており，対象となる専門分野のコーパスの出現頻度や出現パターンを解析して抽出する手法が主流である．しかし，これらの手法はコーパスの頻度情報や出現パターンを解析するだけであり，その用語がどのようなカテゴリに属しているかということは一切考慮されていない．そこで，我々は語のカテゴリ情報をWikipediaのエントリ構造から求めることで適合度の高い用語を抽出できるのではないかと考えた．また，適合度の高い用語に文字列で類似した用語も専門用語であるという仮定を立て，適合度の高い用語に類似している用語も抽出する．本論文ではカテゴリ情報と文字列の類似度を考慮することで適合率・再現率の高い，専門用語抽出手法を目指しているが，その前段階として，従来手法と精度を比較し，改善につなげていくことを目的としている．

2　関連研究

あるコーパス内の専門用語を自動抽出するという研究は．大別して(1)単一コーパスのみを用いる手法(2)複数のコーパスを用いる手法[1](3)Wikipediaのエントリ構造を用いた手法[2]に分けられる．本論文では(1)単一コーパスのみを用いる手法に焦点を絞り，比較対象とする．その中でFLR[3]を紹介する．

2.1　単一コーパスのみを用いるFLR

[3]は専門用語の多くは複合名詞であるとし，連続して出現する単名詞$N_1, N_2, \cdots, N_L$の順で連接した複合名詞CNのスコア付けを考える．スコア付け方法として複合名詞CNを構成する単名詞N_iがバイグラムで連接する用語の出現頻度と種類，そしてCNの出現頻度を統計量としている．図1はその例を示している．単名詞「ネットワーク」,「アドレス」のそれぞれに対して左右の隣接頻度を求めている．例ではCNが「ネットワークアドレス」に対応する．CNのスコア$FLR(CN)$は次式で表される．

$$FLR(CN) = req(CN) \times LR(CN) \quad (1)$$

$$LR(CN) = (\prod_{i=1}^{L}(FL(N_i)+1)(FR(N_i)+1))^{\frac{1}{2L}} \quad (2)$$

この指標が大きいほど重要度の高い用語となる．

3　専門用語抽出手法

3.1　Wikipediaのエントリ構造による専門用語の抽出

本節ではWikipediaのエントリ構造を用いて適合度の高い用語を抽出する方法について述べる．まず，専門用語を抽出したい専門分野qを入力する．ただし，qに対する記事がWikipedia内に存在するものとする．

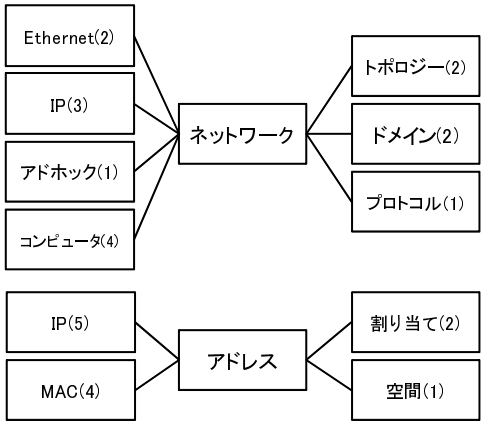


図 1: 複合名詞：「ネットワークアドレス」における左右連接単名詞とその頻度例

次に，q の Wikipedia 記事の属しているカテゴリ集合 $C_q = \{c_1, \cdots c_n\}$ を取得する．各カテゴリ c_i は q の上位語と見なすことができ，カテゴリ集合 C_q は q の属する分野であると考えられる [4]．そして，得られたカテゴリ集合 C_q の要素 c_i について，それぞれに含まれる記事を全て取得し，記事集合 A_q を作成する．最後に A_q のそれぞれの記事内に含まれているアンカーテキスト (リンクがある用語) を全て抽出し，アンカーテキスト集合 $W_q = \{t_1, t_2, \cdots t, m\}$ を作成する．W_q は q のカテゴリに内にある記事中のアンカーテキストを抽出しているので分野 q の専門用語が多く含まれていることになる．今，分野 q をコンピュータネットワークとしたときの専門用語候補集合 W_q を考える．図 3 で W_q 中には「2002 年」,「阪神淡路大震災」,「旅行代理店」などの専門用語ではないと考えられる用語も含まれている．このような明らかに専門用語ではない用語を本論文ではノイズと呼ぶ．このノイズを取り除くためには，ある用語がどんな分野のカテゴリに属しているかを判断すれば解決できると考えられる．例えば,「2002 年」は「カテゴリ：年代」というカテゴリに属していることが予想できる．また,「IP アドレス」はコンピュータネットワーク関連のカテゴリに属しているはずである．そこで，分野 q の属しているカテゴリ C_q をルートノードとするカテゴリグラフ G_q を構築する．G_q 内に存在するカテゴリに属していれば分野 q の用語であることが期待できる．逆に，G_q 内に存在しないカテゴリに属している用語であればノイズとみなされ排除される．アンカーテキストのリンク先のページがカテゴリグラフ内ののカテゴリに属する割合 c_{rate} を次式で示す．

$$c_{rate} = \frac{\text{ある記事がカテゴリグラフ内にあるカテゴリに属している数}}{\text{ある記事が属しているカテゴリ数}} \quad (3)$$

本論文では $c_{rate} \geq 0.5$ であるアンカーテキストの用語を専門用語候補集合 T_q に登録する．このような処理によって得られる用語集合 T_q は適合度の高い専門用語集合であることが期待できる．尚，$depth$ はカテゴリグラフの深さである．例の場合は $depth$=2 となっている．また，Wikipedia はリダイレクト機能を持っている．リダイレクト機能とはある記事が参照されたときに，別の語彙 (記事) に対して転送 (リダイレクト) するための機能である．例えば,「LAN」というアンカーテキストのリンク先は「Local Area Network」といった同義語のページに転送される．リダイレクト機能は，同義語や類義語など意味的に近い語同士に設定される場合が大半である．この機能を利用することにより，専門用語候補集合 T_q に「LAN」と「Local Area Network」の二語を登録することができ，表記の揺らぎに対応できると考えられる．そこで，T_q の中でリダイレクト関係によって抽出される用語を新たに T_q に登録する．

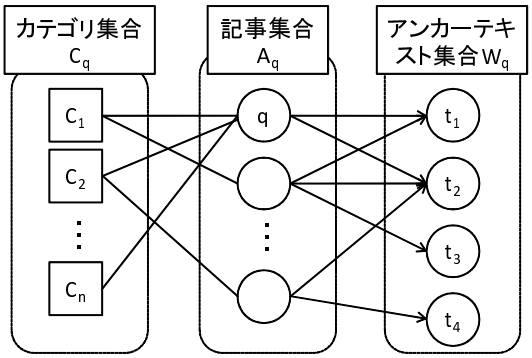


図 2: 専門分野 q からのアンカーテキストの取得方法

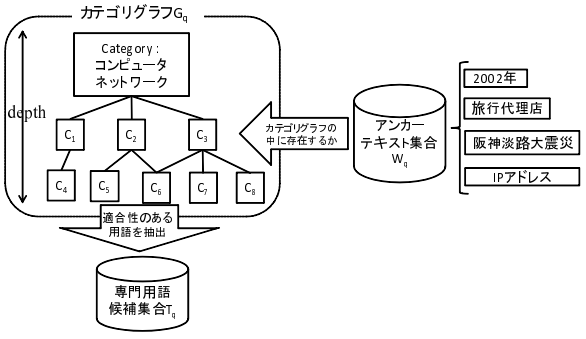


図 3: カテゴリグラフによるアンカーテキストの分野判別

3.2 編集距離による再現率向上のための用語抽出

3.1 節で抽出した用語集合 T_q は分野 q における適合性の高い専門用語が含まれている．つまり，専門用語を抽出したい分野 q のコーパスにも同一の用語含まれていると考えられる．しかしながら，図 4 のように抽出対象のコーパスには含まれているのに T_q には出現しない用語もある．Wikipedia は大規模なコーパスとは言われているが，必ずしも対象コーパス中に出現する用語と同じ用語が存在するとは限らない．これでは対象コーパス中に出現する専門用語を抽出するときには再現率が低下してしまう．そこで我々は対象コーパス中に出現する専門用語の中で T_q の中に含まれていない用語が出現したとしても T_q に登録されている用語との類似度を求めることで対象コーパス中の専門用語を抽出できると考えた．これは，用語を文字列と見て，文字列で類似する用語は同じ分野の専門用語であるという仮説に基づいている．そこで，対象コーパスの複合名詞と専門用語候補集合 T_q に含まれている用語との距離を求めることにする．対象コーパス中の複合名詞 $W_{1,i}$ の文字単位のリスト $W_{1,i} = w_1, w_2, \cdots, w_n$ と専門用語用語候補集合 $W_{2,j}$ の文字単位のリスト $W_{2,j} = w_1, w_2, \cdots, w_n$ の距離 $Dist(W_{1,i}, W_{2,j})$ を以下のように定義する．

$$Dist(W_{1,i}, W_{2,j}) = \begin{cases} 999 & \text{if } nprec(W_1,W_2)=0 \text{ または } nco(W_1,W_2)=0 \\ LD(W_1,W_2) \times nprec(W_1,W_2) \times nco(W_1,W_2) & \text{otherwise} \end{cases}$$

$LD(W_{1,i}, W_{2,j})$ は編集距離 (Levenshtein Distance) を表す．編集距離は二つの文字列がどの程度異なっているかを示す数値である．一つの文字列を別の文字列に変形するのに必要な手順の最小回数が距離となる．手順としては文字の挿入，削除，置換をそれぞれにコストを与える．しかし，この指標だけでは文字列長の短い単語を比較したときに編集距離が小さくなってしまう．そこで両方に共通して出現する文字数を考慮した非適合度 $nprec(W_{1,i}, W_{2,j})$ を定義する．

$$nprec = \frac{|W_{1,i} \cap W_{2,j}|}{|W_{2,j}|} \tag{4}$$

また，両方に共通して出現していても語順が違えば全く意味の違う用語かもしれない．語順不一致率 $nco(W_{1,i}, W_{2,j})$ を次式で定義する．

$$nco(W_{1,i}, W_{2,j}) = \frac{\sum_{a \in W_{1,i} \cap W_{2,j}} \sum_{b \in W_{1,i} \cap W_{2,j}} correct_{a,b}}{|W_{1,i} \cap W_{2,j}| C_2}$$

$$correct_{a,b} = \begin{cases} 1 & \text{if } a \succ_{W_{1,i}} b \text{ かつ } a \succ_{W_{2,j}} b \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$\succ_{W_{1,i}}$ は文字の順序関係を表す．$Dist(W_{1,i}, W_{2,j})$ が小さいほど両者の文字列間距離が小さく，類似した用語と言える．また $nprec(W_{1,i}, W_{2,j})$，$nco(W_{1,i}, W_{2,j})$ のいずれかの値が 0 になるということは二つの文字列間の距離は大きい方へ向かわないといけない．しかしながら，$Dist(W_{1,i}, W_{2,j}) = 0$ となり，同一の文字列と判断されてしまう．そこで，文字列間の距離が大きいことを示す値として 999 を設定した．式 5 は対象コーパスの 1 つの用語 $W_{1,i}$ につき j 通りの $Dist(W_{1,i}, W_{2,j})$ を求めることになる．そのうち距離が小さい top$N$ の平均を用語のスコアとする．専門用語であれば文字列で類似した用語が複数個含まれることが予想される．対して，一般的な用語は $Dist(W_{1,i}, W_{2,j})$ が 999 に近づく値が出現すると考えられる．表 1 は用語間の類似度算出例を示している．編集距離の置けるコストは全て 1 としている．

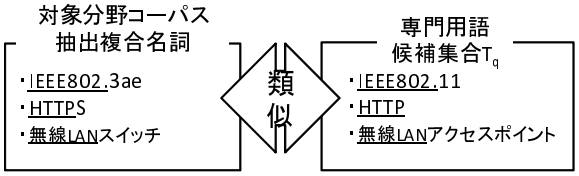


図 4: 用語の文字列類似度

表 1: 用語：IP 電話における各用語のそれぞれの指標例

用語	LD	$nprec$	nco
IPv6	2	2	1
IP-VPN	4	2	1.5

4 評価実験

本論文で示した手法が従来手法と比較してどれくらいの精度を持っているのかを検証し，改善のとする．従来の比較手法としては FLR を用いる．専門用語を抽出するテストコレクションは Web で公開されているコンピュータ用語辞典の中で IT 用語辞典「e-Words」とする．またジャンルは「ネットワーク」と「プログラミング」の二種類とした．用語の語義文を対象コーパスとし，用語の見出し語のうち語義文に出現する用語の集合を正解集合とした．Web 上の用語辞書のような電子辞書をテストコレクションにした理由として擬似的な正解集合を備えているからである [5]．語義文を

MeCab で形態素解析し，連続する名詞 (複合名詞) を抽出し，候補用語とした．候補用語の中に正解用語が部分文字列として入っている場合でも正解用語とした．正解用語数および総用語数を表 2 に示す．候補用語に対し本手法と FLR でスコア付けし，用語のスコアが高い順にソートし，語数を増やしていったときの再現率と適合率とのの調和平均をとった F 値を評価とする．尚、本手法 $depth = 3$，$N = 10$ と設定した．

表 2: テストコレクションの正解用語数と総用語数

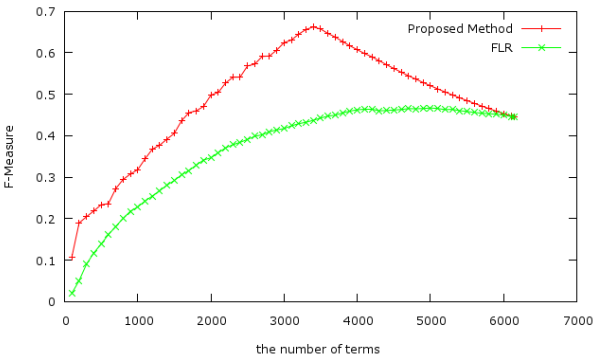
ジャン	正解用語数	総用語数
ネットワーク	1759	6150
プログラミング	380	6049

5 実験結果・考察

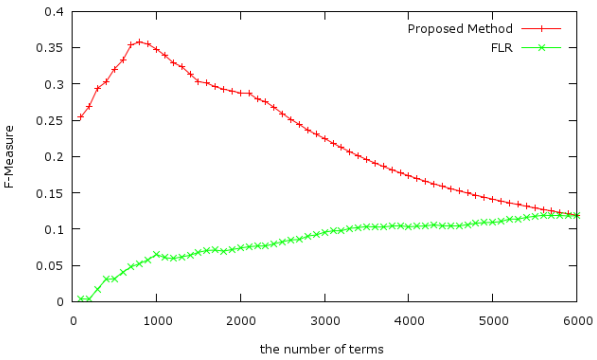
表 5 は F 値の推移を示している．二つのジャンルにおいて本手法は FLR よりも F 値が高いことが分かる．また，今回用語抽出のノイズとなるようなストップワードは考慮していない．このことから本手法はノイズの影響に強い手法なのではないかと考えている．FLR を評価した [3] では複合名詞スコア付けし，スコアの高い順に並び替えた時，上位 3,000，6,000 語よりも 12,000 語や 15,000 語で F 値が最も高くなることが知られている．すなわち，対象コーパスの規模が大きくなればなるほど FLR は精度のよい専門用語抽出法になってくると考えられる．今回の実験からもネットワークコーパスはプログラミングコーパスと比べ，総用語数と正解用語数が多く規模の大きいコーパスである．図 5(a) の方が F 値が高く，規模の大きいコーパスに対して精度がよくなることがわかる．今回は規模の小さいコーパスで FLR より良い精度が得られたが，コーパスを拡大していくと FLR の方が良い結果になることも考えられる．

6 おわりに

本論文では語のカテゴリ情報を文字列間の類似度を用いて専門用語抽出ストップワードを考慮することなく，従来手法の一つである FLR と比較し精度の良い結果が得られた．今回用いたテストコレクションは小規模なコーパスであるためコーパスの規模を変えて本手法を検証してみる必要がある．またジャンルを他分野に拡大することで本手法の汎用性も確認する必要がある．



(a) ジャンル：ネットワーク



(b) ジャンル：プログラミング

図 5: F 値の推移

参考文献

[1] 久保順子，辻慶太，杉本重雄：” 異なる専門分野のコーパスを利用した専門用語抽出手法の提案 ”，情報知識学会誌，vol.1，No.1，pp.15-31，2010

[2] 中谷誠，AdamJatowt，大島裕明，田中克己：” Wikipedia のリンク構造とカテゴリ構造を用いた検索語からの専門語の抽出 ”，情報処理学会研究報告，2008

[3] 中川裕志，森辰則，湯本紘章：” 出現頻度と連接頻度に基づく専門用語抽出 ”，自然言語処理，Vol.10，No.1，pp.27-45，2003

[4] 中谷誠，AdamJatowt，大島裕明，田中克己：” 理解容易度に基づく Web ページの検索とランキング ”，電子情報通信学会，第 1 回データ工学と情報マネジメントに関するフォーラム (DEIM Forum 2009)，A7-1，2009

[5] 辻川亨，吉田稔，中川裕志，：” 語彙空間の構造に基づく専門用語抽出 ”，情報処理学会 NL 研究会 159，pp.152-162，2004

[6] 森竜也，増田英孝，清田陽司，中川裕志： ” Wikipeida エントリ構造抽出ツール：Wik-IE ”，第 20 回セマンティックウェブとオントロジー研究会，2009