社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2008－ＦＩ－ 89 (14)
2008－ＮＬ－183 (14)
2008／1／22

Web情報を利用した確率モデルによる略語推定

村山 紀文† 奥村 学‡

概 要

原語－略語対を発見することは，検索などの言語処理に対して重要である．本研究では，与えられた語に対する略語を推定する確率モデルを提案する．提案する確率モデルは，正解データからの学習によって得られる原語－略語候補対の文字列的な正当性と，Web検索エンジンから得られる統計値から得られる意味的・社会的な正当性の双方を考慮することが出来る．提案モデルは，実験において旧来使われてきたテンプレートを用いた手法よりも良い結果を示すことが出来た．

Abbreviation Estimation using a Probabilistic Model with Web Information

Norifumi MURAYAMA† Manabu OKUMURA‡

Abstract

Detecting abbreviation-root pairs especially for proper nouns is an important technology in the field of natural language processing. To correctly detect the pairs, most previous studies have extracted pairs from a sentence or a document which contains both an abbreviation and its root.We present a new approach that detects the pairs by estimating abbreviations from a root. Our approach is based on the probabilistic model. We evaluated our approach using Japanese abbreviation-root pairs as a training/test set. The experiment demonstrated the effectiveness of our approach.

1 はじめに

同一のエンティティを指し示す様々な異なる表現を同定する，いわゆる名寄せの問題が検索などの分野で注目されている．例えば，“プレイステーション３”に関する情報を多く得るためには，「プレイステーション３」と書かれた文書だけではなく，同じエンティティを指す「プレステ３」と書かれた文書も獲得されることが望ましい．本研究では，このような名寄せ問題のうち，「プレイステーション３」と「プレステ３」のように略語関係にあるものに着目する．

本論文では，「略語関係」にある対を，「同義関係にある語の集合の中で，一方がもう一方から表層的に短縮されているような語の対」であると定義する[1]．また，短縮されている語を「略語」，短縮されていない語を「原語」と呼ぶ．

このような略語関係は，一般の人々によって生成され広められることも多いため，流動的であり捉えることが難しい．また，新聞などの文字数の制限がある文書や，ブログなどの一般の人々の書く文書では使用される頻度も多く，名寄せの問題の中でも特に重要な関係であると考える．

†東京工業大学大学院 総合理工学研究科
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology
murayama@lr.pi.titech.ac.jp
‡東京工業大学 精密工学研究所
Precision and Intelligence Laboratory, Tokyo Institute of Technology
oku@pi.titech.ac.jp

[1]「プレイステーション」と「PS」は直接の略語関係ではなく，同義語の略語であると考える．そのため，今回扱う略語関係にあるとは考えない．

これまでも略語関係対獲得の研究は行われてきたが，その多くは「Aを略してB」のようなテンプレートを用いた手法であった．しかし，Web文書のような文体が一致しないリソースに対しては，テンプレートを用いた手法では難しい．

本研究では，原語となりうる可能性のある語のリストが与えられているものとして，これらの名詞に対して適切な略語を推定することで，原語－略語対の獲得を目指す．そのために，文字列的な情報とWeb上の検索エンジンから得られる情報を総合して略語を推定する確率モデルを提案する．

2 関連研究

2.1 辞書自動生成

原語・略語対の自動獲得は，以前より辞書の自動生成の一環として行われてきた[1][2]．これらの手法は，「Aを略してB」「A（以下，Bと略す）」のような略語関係が記述されやすいテンプレートをあらかじめ用意し，それを利用して抽出を行う手法が多かった．これらの手法は，文体が一定の文章からの抽出にはよい結果を示しているが，様々な文体が入り混じるWeb文書などのコーパスから広く抽出することは難しい．

2.2 原語－略語対自動獲得

テンプレートを用いずに原語－略語対の自動獲得を行っている研究として，[3][4][5]が挙げられる．

[3][4]では，新聞コーパスからの原語－略語対の獲得

を行っている．酒井らは，まず原語－略語対候補から，表層上の制約ルールを用いて絞り込みを行い，次に両方の間接共起頻度により更なる絞り込みを行っている．[5] では，原語からの省略ルールをいくつか用意して略語候補を生成し，略語候補の web 上での出現数，辞書の情報などを用いて略語を推定している．

いずれの研究も，原語からの省略はルールベースで行われており，様々な略語関係に対して十分に柔軟であるとは言い難い．

2.3 英語の原語－略語対自動獲得

英語に対する原語・略語対の自動獲得の研究もいくつか行われている [6][7][8]．多くの場合，英語の自動獲得手法は文章中から略語を発見し，それに対応する原語を発見することで行われる．これは，日本語の略語と異なり，文章中から略語が発見しやすい[2]点をうまく利用したものであると言える．

また英語の場合，略語は原語から単語の先頭一文字を取ることで生成される場合が大半であり，日本語の生成規則よりもはるかに単純であることも重要な点である．

2.4 中国語の原語－略語対自動獲得

Chang らの研究 [9][10] では，中国語における原語－略語対自動獲得の手法が提案されている．Chang らの研究は，本研究との共通点がいくつかある．

そもそも，中国語の略語と日本語の略語は非常に似通った背景を持っている．まず，日本語も中国語も文章に単語境界が存在していないことが挙げられる．このため，英語でおこなわれていたような文章中から略語と思われる語を発見する手法は共に取りづらい．また，略語自体の特徴も非常に似通っている．特に漢字から構成された略語では顕著である．さらに，Chang らの手法も，略語から原語を推定するというアプローチではあるものの，本研究と同じく確率モデルを用いた手法を取っている点も共通である．

しかし，Chang らのアプローチを本研究が対象とする問題に適用することは難しいと考える．まず，設定に大きな違いがあり，Chang らは略語から原語を推定するという本研究とは逆のアプローチを取っている．また，Chang らの提案している確率モデルでは，本研究で行っているような抽象化は行われておらず，確率モデルの変数として文字や文字列そのものが使われている．本研究が対象としている原語－略語対は漢字以外にも仮名，アルファベット，数字などが入り交じっており，かつ Web 上に現れるような若者言葉的な略語も含むため，Chang らが対象としていた原語－略語対よりも遙かに複雑であると考えられる．そのため本研究においては確率モデルの構築の際には適度に抽象化する必要がある．

[2]単語区切りが明確であり，大文字が含まれているなどの表層上の特徴が顕著であるため．

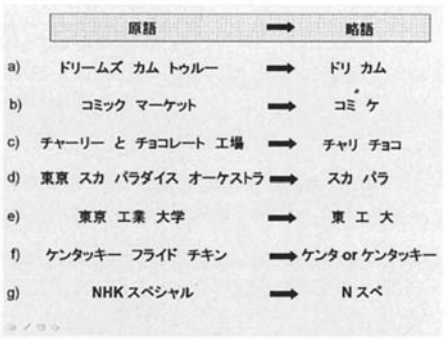

図 1: 本研究の対象となる原語－略語対の例

3 手法概要

3.1 提案モデル

本研究では，略語を取る可能性のある語のリストは事前に用意できるものとし，それらの原語候補が与えられた際に，ふさわしい略語を推定するという方法を採る．

このとき，ある原語候補 R に対して尤も可能性の高い略語 A を求める問題は以下のように定式化出来る．

$$A = \arg\max_{A^*} P(A^*|R) \quad (1)$$

$$= \arg\max_{A^*} \frac{P(A^*, R)}{P(R)} \quad (2)$$

ここで式 (2) において，原語 R と略語候補 A^* との関係性の深さを定義する必要があるが，このとき以下の 2 つの視点から考える必要がある．

1. 単語構成的な視点
2. 意味的・社会的な視点

1 は，人間が略語を考案する際に暗黙的に従っている傾向に，どれだけその原語－略語対が即しているかという視点である．例えば，略語の一部は原語の語基[3]の先頭 2 モーラ[4]の抜粋（図 1 の **a), c), d), e)**）であることが多いという傾向がある．また，略語は 3～5 モーラなどで構成されやすく，カタカナ語の略語の場合は 2 モーラと 2 モーラを組み合わせた合計 4 モーラで構成されることが多いという傾向もある[5]．本研究では，単語構成的な視点からの関係性を正解原語－略語対から傾向を学習した統計モデルで計算することで，この視点による関係性をモデルに組み込む．

一方 2 は，略語が原語と同じエンティティを示すものとして，どれだけ社会的に認知されているかという視点である．本研究では，この視点における関係性を

[3]複合語を構成する要素となっている語

[4]モーラとは，拍を表わす単位であり，日本語の場合は大抵仮名 1 文字が 1 モーラにあたるが，「ファ」などは 2 文字で 1 モーラに相当する．

[5]これらの特性を把握する手がかりとして，[11] を参考にした．

検索エンジンから得られるWeb上の情報を利用することで捉えることとする．

これら2つの視点は共に重要であり，最終的には両方とも考慮した上で略語の推定を行うべきである．これらの2視点から得られる情報が互いに独立であると仮定すると，式 (2) 中の $P(A,R)$ を以下のように展開することが出来る．

$$P(A,R) \simeq P_{static}(A,R) * P_{dynamic}(A,R) \quad (3)$$

ここで，$P_{static}(A,R)$ を視点1に基づく確率，$P_{dynamic}(A,R)$ は視点2に基づく確率をそれぞれ表現しているものとする．式 (2),(3) より，目的式は以下のような形に展開することが出来る．

$$A \simeq \arg\max_{A^*} P_{static}(A^*,R)\frac{P_{dynamic}(A^*,R)}{P(R)} \quad (4)$$

式 (4) 中の $P_{static}(A^*,R)$ を静的モデル，$P_{dynamic}(A^*,R)/P(R)$ を動的モデルと呼ぶこととする．静的モデルの詳細な説明を4章で，動的モデルの詳細な説明は5章で行う．

本研究ではこのような確率モデルに従っているが，これには大きな利点が2つある．一つには，略語候補が確率に基づいたスコアと共に得られる点である．手法の入力となる語は略語を持つことは保障されておらず，あるいは複数の略語に対応している可能性もある．つまり，解は一つとは定められていない．出力としてスコアと結び付けられた略語候補が得られた場合，適切な閾値をスコアに対して定めることが出来れば，語に対して適切な数の解を得られることが期待できる．また，前述のように2つの視点の情報を総合的にスコアに反映することが出来ることも大きな利点である．例えば，単語構成的な視点から見て通常の傾向から若干逸脱した略語（「ゲームボーイアドバンス」に対する「ゲボドバ」など）に関しても，社会的・意味的な視点からの情報を考慮することで高いスコアを得ることが出来る可能性がある．

3.2 処理の流れ

ここで，提案手法の実際の処理の流れについて説明を行う．

1. 入力の原語を語基に分割
2. 静的モデルのみで計算を行い，静的モデルのスコアのみで略語候補をランキングする
3. ランキングの上位N位の略語候補のみに対して，動的モデルの計算．静的モデル，動的モデルを統合したスコアを計算する

まず，静的モデルの計算の際に原語が語基に分割されている必要があるため，その処理を行う．この分割問題は厳密には通常の日本語の分かち書きとは異なる[6]が，

[6]"東京工業大学"は通常の分かち書き問題では一つの固有名詞と捉えられるべきである場合が多いが，語基分割としては"東京""工業""大学"という三つの語基に分割されるべきである

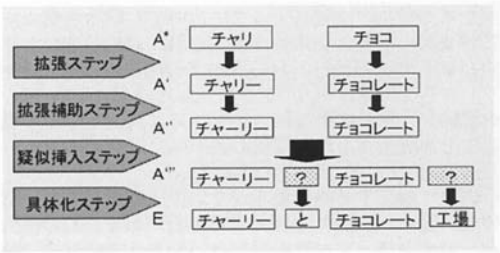

図 2: 変換モデルのステップ化

今回は代用として形態素解析器であるMecabの5-best解のうち，もっとも分割数が多いものを用いた．

動的モデルを計算する際には，検索エンジンでの検索が必要となる．しかし，考えられうる全ての略語候補（原語の全ての部分文字列数に近似）は多くの場合膨大な数になることが考えられるため，それら全てに対して動的モデルの計算を行うことは現実的ではない．そこで上に示したように，静的モデルのスコアを用いて解候補の絞り込みを行った後に動的モデルの計算を行うという，二段階の方法を取ることで検索回数を抑えることを行う．なお，6章では静的モデルの上位30位を動的モデルの計算対象として実験を行った．

4 静的モデル

静的モデルでは，原語－略語候補間の単語構成的な視点からの関係性を求める．まず，式 (4) の静的モデル部分を以下のように展開する．

$$P_{static}(A,R) = P_s(A)P_s(R|A) \quad (5)$$

展開後の式 (5) は，翻訳 [12] や要約 [13] などで使用される確率モデルの一つであるNoisy-channel modelの形を取っている．ここで，$P_s(R|A)$ の項はAからRへの変換モデル，$P_s(A)$ の項は言語モデルと呼ばれる．本研究においては，変換モデルは略語候補Aから原語Rの想定しやすさ，また言語モデルは略語候補Aの略語としてのありやすさを示している．

4.1 変換モデル

変換モデルにおいては，略語候補Aから原語Rへの変換を考えたときの変換確率を求める．ここでAからRの変換を以下の4ステップの変換過程として考える．図2に，「チャリ チョコ」から「チャーリー と チョコレート 工場」（映画のタイトル）の変換過程を例として挙げる．

1. 拡張ステップ
2. 拡張補助ステップ
3. 疑似挿入ステップ
4. 具体化ステップ

まず，最初の拡張ステップにおいて，略語を構成する各要素の前後に文字列が追加され，対応している原語要素への拡張が行われる（図2の例における「チャリ」から「チャリー」,「チョコ」から「チョコレート」の変換）．次の拡張補助ステップでは，仮名略語の場合にのみ発生する略語要素の途中への撥音文字（ッ），長音記号（ー）の挿入を行う．

次に，対応する略語要素がない原語要素（図2における「と」「工場」）を挿入する必要がある．この挿入を，要素を挿入すべき要素枠を用意するステップ（疑似挿入ステップ）と，用意された枠に実際の要素を挿入するステップ（具体化ステップ）にてモデル化する．

これらの各ステップにおける変換が互いに独立であり，各変換の生起確率が直前の状態にしか依存しないと仮定すると，式 (4) の変換モデルは以下のように展開することが出来る．

$$P_s(R|A) \simeq P_s(R|A''')P_s(A'''|A'')\\ P_s(A''|A')P_s(A'|A) \quad (6)$$

各々のステップにおける変換確率は，原語－略語の言語構成的な特徴を踏まえた上で，抽象化した形で計算する．それぞれのステップにおける計算の詳細は，以降で説明を行う．

拡張ステップ

拡張ステップでは，各略語要素に対して前後に文字を加えることによって，対応する原語要素に近づける変換を行う．この変換確率の計算は，抽象化した以下の式で求める．

$$\begin{aligned} & P_s(A'|A) \\ &= \prod_{i=0}^{N} P_s(a'_i|a_i) \\ &\simeq \prod_{i=0}^{N} P(headnum, char|type, abbnum), \end{aligned} \quad (7)$$

式 (7) における各変数は，以下を示している．

- N: 略語要素の数
- a_i: A を構成する要素
- $headnum$: 略語の前方に加えられる文字数
- $char$: 略語の直後に加えられる文字
- $type$: 略語要素の文字タイプ[7]
- $abbnum$: 略語要素の文字数[8]

このとき，図2の変換の確率は以下の式で求められる．

$$\begin{aligned} & P_s(A'|A) \\ &= P(\text{チャリー} \mid \text{チャリ})P(\text{チョコレート} \mid \text{チョコ}) \\ &= P(0, \text{ー} \mid \text{仮名}, 2)P(0, \text{レ} \mid \text{仮名}, 2) \end{aligned}$$

[7]文字タイプは対象の文字種により，「仮名」「漢字」「アルファベット」「数字」「以上の組み合わせ」「記号のみ」の値を取る．

[8]本研究では，基本として文字で数えるが，仮名においてはモーラで数えている．

式 (7) で用いている変数は，変換の傾向を捉えやすい情報を厳選したものである．例えば，人間が略語を考案する際には，略語要素は原語要素の先頭あるいは先頭に近い位置から抜粋されて構成されやすいという傾向が見られる．また，抜粋の終端位置はその次にある文字に影響されて変化する．例えば，図1における「マーケット」から抜粋される略語要素が「マー」ではなく「マ」となっているケースなどがこれに当てはまる．また，これらの傾向は略語要素を構成している文字タイプや文字数によっても変化することが考えられたため，上記の式としている．

拡張補助ステップ

拡張補助ステップは，仮名略語における撥音，長音の省略に対応したステップであり，略語要素の途中に「ッ」「ー」が挿入される変換となる．この変換確率の計算は，以下の式で求める．

$$\begin{aligned} P_s(A''|A') &= \prod_{i=0}^{N} P_s(a''_i|a'_i) \\ &\simeq \prod_{i=0}^{N} P(wordch|type, abbch), \end{aligned} \quad (8)$$

式 (8) における各変数は，以下を示している．

- $wordch$: 原語要素の中に含まれる長音記号，撥音文字
- $type$: 略語要素の文字タイプ
- $abbch$: 略語要素の中に含まれる長音記号（「ー」），撥音文字（「っ」）

このとき，図2の変換確率は以下の式で求められる．

$$\begin{aligned} & P_s(A''|A') \\ &= P(\text{チャーリー} \mid \text{チャリー}) \\ &\quad *P(\text{チョコレート} \mid \text{チョコレート}) \\ &= P(\text{ー} \mid \text{仮名}, none)P(none \mid \text{仮名}, none) \end{aligned}$$

疑似挿入ステップ

疑似挿入ステップでは，対応する略語要素がない原語要素枠を，略語要素列に挿入する．この変換確率の計算は，以下の式で求める．

$$\begin{aligned} & P_s(A'''|A'') \\ &= P_s(begin, middle, end|type, abbfragnum) \end{aligned} \quad (9)$$

式 (9) における各変数は，以下を示している．

- $begin$: 略語要素列の前に挿入される原語要素数
- $middle$: 略語要素の間に挿入される原語要素数
- end: 略語要素列の後に挿入される原語要素数
- $type$: 略語要素列の文字タイプ
- $abbfragnum$: 略語要素数

このとき，図2の変換確率は以下の式で求められる．

$$P_s(A'''|A'') = P(0, 1, 1 \mid \text{仮名}, 2)$$

ここで捉えたいのは，人間が略語を考案する際には，前方にある連続した原語要素からの抜粋で構成されやすいという傾向である．そのため，*beggining* や *middle* の値が大きければ確率が小さく，*end* の値が小さければ確率が小さくなることが期待される．

具体化ステップ

次に，疑似挿入ステップで挿入されたそれぞれの原語要素枠に，具体的にどのような原語要素を挿入するかを決定する．この変換確率は以下の式で計算される．

$$\begin{aligned} & P_s(R|A''') \\ &= \prod_{r_i \in R^-} P_s(r_i|a_i''') \\ &\simeq \prod_{r_i \in R^-} P(wtype, wlength|type, location) \end{aligned} \quad (10)$$

式 (10) における各変数は，以下を示している．

- R^-: 疑似挿入ステップで挿入された原語要素枠集合
- *wtype*: 挿入する原語要素の文字タイプ
- *wlength*: 挿入する原語要素の文字数
- *type*: 略語要素列の文字タイプ
- *location*: 挿入場所（beggining,middle,end）

このとき，図 2 の変換確率は以下の式で求められる．

$$\begin{aligned} & P_s(R|A''') \\ &= P(\text{仮名}, 1|\ \text{仮名}, middle) P(\text{漢字}, 2|\ \text{仮名}, end) \end{aligned}$$

4.2 言語モデル

言語モデルは，略語らしさを示すモデルである．本研究では，この値を文字列長などから求めるものとした．具体的には以下の式を用いる．

$$\begin{aligned} & P_s(A) \\ &\simeq P(type, length, fragnum) \\ &\quad * \prod_{a_i} P(fraglength, fragtype) \end{aligned} \quad (11)$$

ここで，式 (11) における各変数は，以下のものを示している．

- *type*: A の文字タイプ
- *length*: A の文字列長
- *fragnum*: A の要素数
- a_i: A の i 番目の要素
- *fraglength*: 要素 a_i の文字列長
- *fragtype*: 要素 a_i の文字タイプ

例えば，図 2 の「チャリチョコ」の言語モデルは以下のように計算できる．

$$P_s(A) = P(\text{仮名}, 4, 2) P(\text{仮名}, 2) P(\text{仮名}, 2)$$

5 動的モデル

3 章で述べたように，動的モデルでは意味的・社会的な視点からの原語と略語候補の関係性，すなわち略語候補が原語と同じエンティティを指すものとして十分に認知されているか，という情報を捉えることを目指す．

本論文では，それらを捉える情報源として，Web 検索エンジンの検索結果の情報を利用した，2 種類の異なるコンセプトによる動的モデルを提案する．なお，今回の実験においては検索エンジンとして Yahoo web search APIs[14] を利用しており，検索スニペットを用いる場合は検索結果の上位 50 位までのスニペットを用いているものとする．

5.1 コンテキストモデル

1 つ目のモデルでは，原語と略語の出現するコンテキストの類似性を捉える．原語と略語が同じエンティティを指す場合，原語と略語は極めて類似したコンテキストを持っていることが期待される．

そこで，原語を検索した際に得られる検索スニペットと，略語を検索した際に得られる検索スニペットの比較を行う．

ここで，式 (4) における動的モデルの項を以下のように展開する．

$$\begin{aligned} \frac{P_{dynamic}(A,R)}{P(R)} &= \frac{P_{dynamic}(R|A)P(A)}{P(R)} \\ &= P_{dynamic}(R|A)\frac{P(A)}{P(R)} \end{aligned} \quad (12)$$

ここで，Web 全体の総文書数を $|D|$, 略語が出現する文書数と原語が出現する文書数をそれぞれ $|A|$,$|R|$ と表すとすると，以下のように展開できる．

$$\begin{aligned} \frac{P(A)}{P(R)} &= \frac{|A|/|D|}{|R|/|D|} \\ &= \frac{|A|}{|R|} \end{aligned} \quad (13)$$

$|A|$,$|R|$ は，それぞれ A と R の検索ヒット数と捉えることが出来る．式 (12),(13) 式より，動的モデルの式は以下のように展開できる．

$$\frac{P_{dynamic}(A,R)}{P(R)} = P_{dynamic}(R|A)\frac{|A|}{|R|} \quad (14)$$

ここで，$P_{dynamic}(R|A)$ は A に対する検索スニペット集合から，R の検索スニペット集合をどれだけ想定しやすいかという計算で求めることができる．これには [15] で提案された，文書から文書に対する言語モデルを用いた生成確率計算の方法を用いる．

$$\begin{aligned} & P(R|A) \\ &\simeq P(S_R|S_A) \\ &\simeq (\Pi_{w \in S_R} P_{gen}(w|S_A))^{\frac{1}{|S_R|}} \\ &\simeq (\Pi_{w \in S_R} \frac{tf(w,S_A)}{\sum_{w' \in S_A} tf(w',S_A))})^{\frac{1}{|S_R|}} \end{aligned} \quad (15)$$

式 (15) で，S_R と S_A はそれぞれ R と A に対する検索スニペットの集合を示し，w はその中に現れている単語を示している．この計算を行う際には，S_A から生成される言語モデルは非常にスパースであることが想定されるため，ラプラススムージングを行った．スムージングの際に加える値 δ には，0.00001 を用いた．

5.2 共起モデル

2つ目のモデルでは，原語と略語は同じ文章に出現しやすいという特徴を捉える．ただし，単純に共起を見るだけでは，原語と略語の同義性を計ることにはならない．そこで，本研究ではスニペット上での原語と略語の出現位置の近さが近ければ近いほど意味的な繋がりがさらに強くなると考え，共起に対する補正という形で出現位置の近さをモデルに組み込む．

式 (4) における動的モデルの項を以下のように展開する．

$$\frac{P_{dynamic}(A,R)}{P(R)} \simeq \frac{closeness(R,A)^{\frac{|R,A|}{|D|}}}{\frac{|R|}{|D|}} = \frac{closeness(R,A)^{|R,A|}}{|R|} \quad (16)$$

$closeness(R,A)$ は 0 から 1 までの値を取り，以下のようにして算出した．

1. 「"原語" "略語"」検索で得られた検索 snippet 上位 50 位を使用
2. 各 snippet を「…」を区切りとして分割
3. それぞれの分割された文字列上で，原語—略語間のバイト数を計る
4. 一つの分割された文字列内に原語と略語が同時に存在しない場合はバイト数＝ 100 とする
5. バイト数が 100 以上の時も 100 とする
6. snippet 内で一番小さいバイト数をその snippet における距離と定義
7. 50snippet 内での平均を取る
8. (100 - 平均距離) / 100 を closeness(R,A) とする

6 データ

6.1 必要となるデータ

手法の実験を行うにあたって，データを用意する．提案手法に対しては，手法の評価を行うためのデータセットと共に，静的モデルの確率を求めるための学習データとなる正解原語—略語対のデータが必要となる．そこで，正解となる原語—略語のデータを作成するために，旧来よく用いられたテンプレートを用いた原語—略語対獲得の手法を Web に適用するために拡張した手法（以降，テンプレート手法と呼ぶ）を適用した．次節でテンプレート手法の説明を簡単に行う．

表 1: 使用したテンプレート

テンプレート	略語の出現位置
原語の略	前後
原語略して	後
原語略称	前後
原語通称	前後
原語以下	後
以下原語は	後

6.2 テンプレート手法

提案手法と同様に，原語となる可能性のある語のリストが手元にあるものとして，語 R が与えられた際に対応する略語を出力するものとする．手法の手順を以下に示す．

1. 表 1 の 6 つのテンプレートの原語の位置に与えられた語 R を挿入．これを Web 検索に対する検索クエリとする．
2. それぞれの検索クエリでフレーズ検索を行う．今回は，提案手法と同様に Yahoo Web Search APIs を用いた．
3. 検索で得られた全てのエントリーの検索スニペットを獲得
4. スニペットより，クエリの前後あるいは後方（使用したテンプレートによって異なる．表 1 参照）15 文字を切り出し，与えられた語 R と DP マッチングを行う．2 文字以上マッチした場合，マッチした文字列を略語候補として抽出．
5. 全てのテンプレート，全てのスニペットを通じて，同じ略語候補が何回出現したかをカウント．その出現回数と共に略語候補を出力．

テンプレート手法で得られた全ての原語—略語対は，全て人手で評価を行い，正しい原語ー略語対のみを取り出して正解データとした．なお，評価の際には上記の出現回数と，抽出元のスニペットを参考にした．

このテンプレート手法は，学習データを作成するために行ったものであるが，そのまま提案手法に対する比較手法として捉えることも出来るため，7 章で示す実験結果では，テンプレート手法による結果と提案手法による結果を併せて示している．

6.3 使用したデータ

データの作成にあたって，まず Web 上の百科事典である Wikipedia[9]から，全ての項目の項目名[10]を抜粋し，原語候補のリストを作成した．

学習データとなる正解原語ー略語対データは，この原語候補リストを全てテンプレート手法にかけ，その出力を人手で評価することによって作成した．

テストデータは，原語候補リストからランダムサンプリングしたサブセットを用いた．このとき，評価のた

[9] http://ja.wikipedia.org/wiki/

[10] ただし，「〜の登場人物」「〜シリーズ」などの集合に関する項目は簡単なルールを作成して事前に除去した

めにこれらの原語候補に対する正解の略語を用意する必要がある．しかし，全ての原語に対する略語を網羅することは困難であるため，各原語候補をテンプレート手法と提案手法にかけた際の出力を全て人手で評価を行い，正解の略語とするという方法を採った．

我々は，2007 年 8 月 24 日時点での Wikipedia を利用し，167,429 の項目タイトルを原語候補リストとして獲得した．それらにテンプレート手法を適用した結果 167,429 組の原語－略語対が出力され，人手で評価した結果そのうちの 13,685 組を正解原語－略語対として獲得することが出来た．学習データには，このうち下記のテストデータと共通でない 13,359 組を用いた．テストデータとしては，原語候補リストから 1,000 件ランダムサンプリングしたものを用いた．これらに対して正解の略語を求めたところ，正解略語が存在する語が 248 語あり，正解原語－略語対としては 326 組あった．残りの 752 語に関しては，略語と思われる語は発見できなかった．

7 実験

7.1 評価指標

手法の評価として，まず正解原語－略語対をどれだけ推定出来るかを評価する．提案手法は出力した略語にスコアがついており，それによって略語を順位付けすることができる．よって，出来るだけ高い順位あるいはスコアで正解を出力できることが望ましい．なお，比較手法となるテンプレート手法においても出現回数をスコアと見なすことが出来るため，同様のことが言える．

これらを評価するために，情報検索などでよく用いられる MAP(Mean Average-Precision) と MRR(Mean Reciprocal Rank) を評価指標として用いる．MAP と MRR はそれぞれ以下の式で求められる．

$$MRR = \frac{1}{|R|}\sum_{R}\frac{1}{|\text{正解}|}\sum_{\text{正解}} 1/RANK$$

$$MAP = \frac{1}{|R|}\sum_{R}\frac{1}{|\text{正解}|}\sum_{\text{正解}} (\text{正解出現時の精度})$$

ここで，R は原語候補集合を示し，$RANK$ は正解略語の出力における順位を示している．

また，評価の際の参考のために，最終的なシステムの出力を上位 n 位までとしたとき，n によって precision と recall がどのように遷移するかを描いた precision-recall グラフも提示する．さらに，最終的な出力をスコア p 以上としたとき，p によって precision と recall がどのように遷移するかを描いたグラフも同時に示す．

もう一つの観点の評価として，与えられた語に対する略語の有無を判定できるかどうかの評価を行う．具体的には，一位の略語のスコアに着目し，そのスコアが p 以上であれば略語を持つと判定する判定器として考えた場合，閾値 p の変化によって precision と recall がどのように変化するかを見る．この観点では，略語を持たない語を含めた上で，出力しているスコアにどの程度信頼性があるかの評価を行う．

表 2: MAP と MRR

手法	MAP	MRR
context	0.433	0.424
co-oc	**0.525**	**0.509**
static	0.353	0.340
temp	0.214	0.259

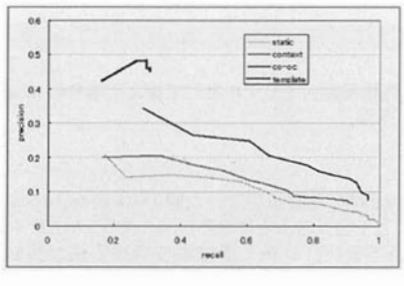

図 3: 順位に対する閾値を変化させた際の precision-recall 遷移

7.2 評価手法

今回は，以下の手法に関してそれぞれ評価を行った．斜体は次節の結果における表現を示す．

- *context* 提案手法 1 （動的モデルがコンテキストモデルに基づく）
- *co-oc* 提案手法 2 （動的モデルが共起モデルに基づく）
- *static* 静的モデルのみ．動的モデルは 1 として計算
- *temp* テンプレート手法

7.3 結果

まず，MAP と MRR の結果を表 2 に示す．表からわかるように，共起モデルが一番良い性能を示すことができた．コンテキストモデルがそれに続き，静的モデル，テンプレート手法と続いている．

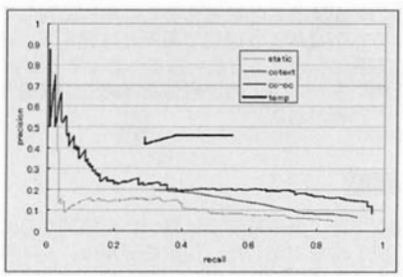

図 4: スコアに対する閾値を変化させた際の precision-recall 遷移

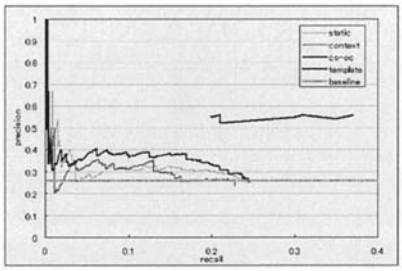

図 5: 略語有無判定モデルとして捉えた場合の precision-recall 遷移

順位に対する閾値を設けた際の precision-recall 遷移を描くと，図 3 のようになる．図 3 より，MAP や MRR においてテンプレート手法より提案手法が良いのは，recall において大きなアドバンテージを持っているからであると分かる．これはすなわち，提案手法がテンプレート手法よりより柔軟な略語現象に対応できていることを示している．当然，用いるテンプレートの種類を増やすことでテンプレート手法の recall は多少なりとも底上げできると考えられるが，その場合には多大な検索コストがかかることが想定される．

図 4 はスコアに対する閾値を設けた際の precision-recall 遷移を示している．図 3 と比較した場合，結果は全体的に低下していることが分かる．また，略語の有無判定器としての precision-recall 遷移を図 5 に示す．図中のベースラインは全て略語を持つと判定したときの値である．このグラフからは，有無判定器としては提案手法が出力しているスコアはあまり有用でないことが分かる．これらのグラフは，同じ語に関する出力内でのスコアの上下関係ではスコアは有用であるものの，スコアそのものの正当性はまだ十分な段階に至っていないことを示している．

この問題の原因として，提案モデルでは複雑な略語現象に対応するために様々な抽象化や独立性の仮定を行っており，それが大きく影響していることが考えられる．しかし，この抽象化や独立性仮定などは，現在のデータ量や現象の複雑さを考えた場合に，効率的なモデル構築のためには欠かせないものである．だが，ある語に対する略語は複数存在しえたり逆に略語が存在しないことがあることを考えると，順位に対して閾値を設ける方法はあまり適切であるとは言えず，スコアに対して閾値を設ける方法を採るべきであると言える．そのためにも，スコアの正当性を高める工夫を今後考えていく必要がある．

8 結論

本論文では，確率モデルに基づいた原語からの略語自動生成手法を提案した．提案モデルは，文字列情報に基づく単語構成的な情報と，Web 検索エンジンから得られる意味的・社会的な情報を総合的に扱った上で，略語の推定を行うことが出来る．

実験において提案手法は，テンプレートを用いた手法よりもよい性能を示すことが出来た．特に Web 文書における共起情報を利用した場合に最も良い性能を示すことができた．提案手法は，様々な略語に対応できる柔軟性に優れており，多少順位の低い点まで考慮すれば，かなりの正解略語をカバーすることが可能である．

今後の課題としては，スコアの正当性の向上が第一に挙げられる．また，動的モデルに関して，よりよい検索方式や定式化がないか検討を行っていきたい．

参考文献

[1] 桜井裕, 佐藤理史. ワールドワイドウェブを利用した用語説明の自動生成. 情報処理学会論文誌, Vol. 43, No.5, pp. 1470–1480, 2002.

[2] 土田正明, 松井藤五郎, 大和田勇人. World wide web を用いた辞典システムの構築. 第 18 回 人工知能学会全国大会, 1A3-04, 2004.

[3] 酒井浩之, 増山繁. 名詞とその略語の対応関係のコーパスからの自動獲得. 電子情報通信学会論文誌 D-II, vol.J85-D-2, no.10, pp. 1624–1628, 2002.

[4] 酒井浩之, 増山繁. 略語とその原型語との対応関係のコーパスからの自動獲得手法の改良. 自然言語処理, Vol.12, No.4, pp. 207–231, 2005.

[5] 桝井文人, 松田良一, 野呂康洋, 河合敦夫, 井須尚紀. World wide web を知識源としたカタカナ語省略形の自動生成. 2004 年度電子情報通信学会ソサイエティ大会講演論文集, A-13-1, 2004.

[6] James Pustejovsky, Jose Castano, Brent Cochran, Maciej Kotecki, Michael Morrell, and Anna Rumshisky. Linguistic knowledge extraction from medline: Automatic construction of an acronym database. In *10th World Congress on Health and Medical Informatics (Medinfo 2001)*, 2001.

[7] Manuel Zahariev. An efficient methodology for acronym-expansion matching. In *Proceedings of the International Conference on Information and Knowledge Engineering, IKE 03, volume 1*, 2003.

[8] Youngja Park and Roy J. Byrd. Hybrid text mining for finding terms and their abbreviations. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2001.

[9] Jing-Shin Chang and Yu-Tso Lai. A preliminary study on probabilistic models for chinese abbreviations. In *Proceedings of the Third SIGHAN Workshop on Chinese Language Learning, ACL-2004*, pp. 9–16, 2004.

[10] Jing-Shin Chang and Wei-Lun Teng. Mining atomic chinese abbreviation pairs: A probabilistic model for single character word recovery. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, COLING-ACL 2006*, pp. 17–24, 2006.

[11] 窪薗晴夫. 新語はこうして作られる もっと知りたい!日本語. 岩波書店, 2002.

[12] Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. In *Computational Linguistics, 16(2):7985*, 1990.

[13] Hal Daume III and Daniel Marcu. A noisy-channel model for document compression. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, 2002.

[14] Yahoo Japan Corporation. Yahoo! developer network. `http://developer.yahoo.co.jp/`, 2007.

[15] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR*, pp. 275–281, 1998.