D-009

略語のフルネームのスケーラブルな推測
Scalable Inference of Full Names from Abbreviations

高 明敏
Mingmin Gao

肖　川
Chaun Xiao

石川 佳治
Yoshiharu Ishikawa

1　はじめに

略語は長い語の一部を省略して短くした語であり，文語でよく使われている．英語では長い語の単語が多く，ある概念を表す長い語を複数回参照する必要がある場合は，略語を使用する方が便利である．したがって，略語と長い語，すなわちそのフルネームの関係を把握することは，さまざまな分野で応用されると考えられる．例えば，文章検索ではキーワードの略語で検索する場合，略語とフルネーム両方が検索結果として得られる．長い語のキーワードの検索がたやすくなっただけではなく，ユーザーにとっても非常に効率的な機能だと考えられる．また，他には文章要約，検索エンジンや固有表現抽出などでも重要な役割を果たしている．

略語に関する研究の一つの難点として，明確的な省略ルールがないことが研究者たちの注目を集めている [1–5]．既存研究における略語とフルネームのマッチング問題には主にテンプレートや辞書を用いて，略語とフルネームのペアを獲得するものであった．これは既に存在する略語とフルネームのペアに対しては確かに実用的であるが，手動で注釈が付けられた略語にしか判別できない，つまり新しい略語に対応しにくいという問題点がある [5]．あるいは教師あり学習の方法を利用して，変換ルールに基づいて略語とそのフルネームのペアを生成する方法もある [1]．しかし，大規模なテキスト中の略語のフルネームを推測したい場合，この方法では略語とフルネームのマッチングの実行スピードが遅いという点で，まだ議論の必要性がある．

そこで本論文では，トライ木と混合ガウスモデルの組合せで略語のフルネームを推測するアプローチを検討する．まずトライ木でフルネームの索引を構築し，テキスト中の略語とフルネームをマッチングすることによって候補ペアを生成する．そして略語とそのフルネームの特徴を学習した混合ガウスモデルを用いて，生成した候補ペアをランク付けすることによって最も適切なペアを見つけ出す．

2 章では本稿の研究問題について述べる．3 章では提案した手法について述べる．4 章では本稿のまとめと今後について述べる．

2　問題定義

本章では，略語のフルネームのスケーラブルな推測問題について，いくつかの定義を行う．特に 2.1 節では略語の明確的な定義について，2.2 節では推測の定義について，2.3 節では略語とフルネームの推測問題の定式化についてそれぞれ述べる．

2.1　略語の定義

ここでは，略語はある単一の単語や複数の単語を組合わせた複合語に対して，その中のいくつかの文字を省略して得られた語である．ドメインによって，ある長い語を繰り返して使う場合は，長い語の略語の方がよく使用される．例えば，生物領域では Prion Forming Domain の代わりに PFD の方がよく使われる．なお，本研究では，規定された略語に限らず，ユーザー自ら作った略語も含めて研究対象とする．

2.2　略語のフルネームの推測の定義

略語とそのフルネームをマッチングする時，複数のフルネーム候補がある略語に対して，そのフルネームの候補をランクし，その中の top-k 候補を最も適切なフルネームとして出力する．

2.3　問題の定式化

テキスト中の略語を抽出して，その略語の最も適切なフルネームを推測するのは本研究の目的である．具体的な問題定義は以下となる．

テキスト集合 S とフルネームの辞書 D が与えられた場合，s_i はテキスト集合 S 中の i 番目のテキスト ($s_i \in S$)．x は s_i 中の文字列，y は辞書 D 中のフルネームの文字列．以下の条件を満たす (x，y) のすべてのペアを見つけ出す：

$$(x, y) = \{x \text{は} y \text{の略語, かつ} x \in s_i, y \in D\}$$

3　提案手法

本章では，2 章で述べた問題に対して提案したアプローチについて具体的に説明する．その手法のパイプラインの概要を図 1 に示す．特に，3.1 節では略語の抽出について，3.2 節ではトライ木索引の構築について，3.3 節では混合ガウスモデルについてそれぞれ述べる．

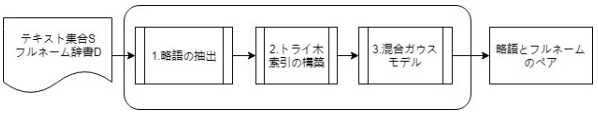

図 1　提案手法のパイプライン

3.1　略語の抽出

テキスト全体のトークンに対してマッチングするのは非効率的なので，まずテキスト中の略語を抽出する．しかし，略語を精確に抽出するのは難しいので，ここではテキストをフィルタ

名古屋大学大学院情報学研究科　Graduate School of Informatics, Nagoya University

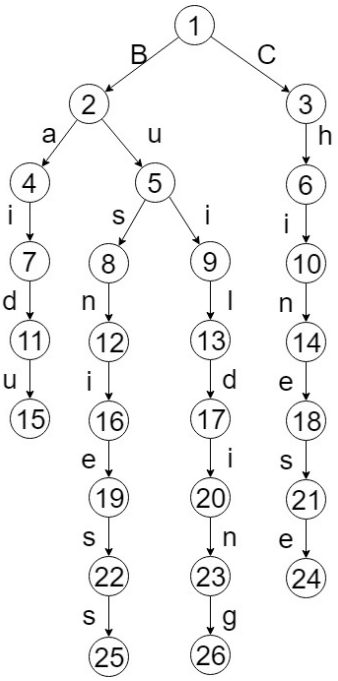

図 2　トライ木の索引

リングし，略語を一つの集合に含める方法を提案する．そのため，Pos-tag 方法 [6] を利用することによって，テキスト中のトークンに品詞のラベルを付与する．略語には名詞のラベルが付けられるので，名詞集合のみを出力すれば，略語が名詞集合に含まれ，処理範囲が縮まる．

3.2　トライ木索引の構築

略語とそのフルネームをマッチングするため，ここではフルネーム辞書を基づいてトライ木の索引を構築する．トライ木では共通接頭辞の検索が可能となり，検索スピードが速いというメリットがある．更に，ここでは索引を構築する時アルファベットの大文字と小文字を区別する．図 2 は四つの文字列のトライ木索引を示す．

そして 3.1 節で得られた名詞集合を構築した索引に入力し，略語とフルネームのマッチングすることを行う．検索時には，検索文字列中のすべての文字がトライ木中の一つの経路で見つかれる場合，検索文字列と検索経路の全文字を一つの候補文字列ペアとして生成する．例えば，検索文字列が bldg の場合，図 2 のトライ木索引では bldg が building の検索経路で見つかれるため，(bldg, building) を候補ペアとして出力する．

3.3　混合ガウスモデル (GMM)

略語を観察することにより，フルネームがいくつかのパターンに従って省略されると考えられる．例えば，単一の単語の場合は単語中の子音を取り，母音を捨てるなど．複合語の場合は各単語の頭文字を取るなどが考えられる．ここではいくつかの特徴で省略パータンを記述する．表 1 は省略パータンの特徴を持つベクトルを示す．

3.2 節で得られた候補ペアセット中の略語とフルネームが実際にマッチングするかどうかを判断するために，混合ガウスモデルを利用して候補ペアの略語とフルネーム省略パータン p の

表 1　省略パータン p の特徴

ID	Description
p_1	略語中の子音数
p_2	略語中の母音数
p_3	フルネーム中の子音数
p_4	フルネーム中の母音数
p_5	マッチングの子音数
p_6	マッチングの母音数
p_7	マッチングの大文字数
p_8	スキップした文字数

確率（密度関数）を評価する．

$$P(p) = \sum_{k=1}^{n} w_k N(p|\mu_k, \Sigma_k)$$

n は混合ガウス分布の数，w_k は k 番目のガウス分布の重み，$N(p|\mu_k, \Sigma_k)$ は平均 μ_k および共分散行列 Σ_k を有するガウス分布による p の確率密度関数である．n は調整でき，他のパラメータは EM アルゴリズムを利用して計算できる．

収集した実世界の略語とそのフルネームのデータセットを混合ガウスモデルに入力し，学習することによって，混合ガウスモデルの n と他のパラメータが得られる．そして，候補ペアセットを学習済みの混合ガウスモデルに入力し，得られた確率が定義した閾値より高い場合，マッチング成功だと判断できる．また，複数のペアがマッチングできた場合，それらのペアをランクし，一番確率が高いペアを出力する．

4　まとめと今後の課題

本論文では，略語のフルネームのスケーラブルな推測問題に対して，トライ木と混合ガウスモデルの組合せで略語のフルネームを推測するアプローチを提案した．今後は混合ガウスモデルを実装し，省略パターンの選択の有用性について検討する．また，大規模なデータセットに対して，実行スピードを最適化する方法を検討する．

謝辞

本研究の一部は科研費 (16H01722，19K11979) による．

参考文献

[1] A. Arvind, C. Surajit, and K. Raghav, “Transformation-based Framework for Record Matching,” in *Proc. ICDE*, pp. 40–49, 2008.
[2] A. Arvind, C. Surajit, and K. Raghav, “Learning String Transformations from Examples,” in *Proc. PVLDB*, vol. 2, pp. 514–525, 2009.
[3] L. Jin, C. Li, and S. Mehrotra, “Efficient Record Linkage in Large Data Set,” in *IEEE*, 2003.
[4] T. Wenbo, D. Dong, and S. Michael, “Approximate String Joins with Abbreviations,” in *Proc. PVLDB*, vol. 11, pp. 53–65, 2017.
[5] 岡田真 and 高橋幹浩, “漢字を中心とした複合語の略語の自動生成,” in *Proceedings of the Annual Meeting of the Association for Natural Language Processing*, vol. 14, pp. 787–789, 2008.
[6] P. Slav, D. Dipanjan, and M. Ryan, “A Universal Part-of-Speech Tagset,” in *Proc. LREC*, pp. 2089–2096, 2012.