類似言語及び小規模データに頑強な言語識別

行野 顕正[†]　田中 省作[††]　冨浦 洋一[†††]　松本 英樹[†]

[†] 九州大学大学院システム情報科学府

[††] 九州大学情報基盤センター

[†††] 九州大学大学院システム情報科学研究院

{yukino@lang.is, sho@srp, tom@is, hide@lang.is}.kyushu-u.ac.jp

1　はじめに

近年，コンピュータの発達に伴い，電子化文書の量は増加の一途を辿っている．そのため，言語・符号系の自動識別は，文書処理の基礎的なタスクとして，ますますその重要性が増している．ブラウザにおける文字コードの自動認識や，WWW からの少数言語データ抽出などは，言語識別の直接的な利用例である．

言語識別では，言語特徴の差を利用する．従来手法では，各言語に特有な表現 [1] や言語の統計情報 [2, 3, 4] を言語の識別に有効な特徴であると仮定してきた．特に統計情報を利用した手法は，特別な言語知識を必要としないなど，多くの利点を持っている．

しかし従来手法では，類似した言語間の識別や小規模文書を対象とした識別において充分な精度が得られていない．これは，従来手法で用いられてきた言語特徴が，こういった対象に適していないためである．

本稿では，従来手法において軽視されてきた，低頻度 byte 列に着目した識別手法を提案する．低頻度 byte 列には，類似言語においても互いに出現しない byte 列が多く存在する．したがって，こういった byte 列を言語特徴に取り入れることにより，類似言語間や小規模文書に対して頑強な識別方式を構築できる．また，本手法は学習データとして小規模な文書集合しか要求しない．そのため，大規模な辞書とのマッチングを行う手法と異なり，多くの言語・符号系に簡単に適用可能である．

2　従来研究

2.1　byte n-gram 統計量の利用

統計情報を用いた手法には，byte n-gram に基づくものが多い．byte を基本単位とすることで多くの言語・符号系に簡単に対応でき，事前の知識がほとんどいらないという利点がある．基本的な手法として，bi-gram マルコフモデルからの文書発生確率に基づく手法 [4] や，判別対象文書と各言語モデル間における，bi-gram 確率分布の KL-divergence に基づく手法 [3] などがある．本節では，代表的な 2 手法について概観し，これらの手法で用いられる言語特徴が類似言語・小規模文書の識別において有効に働かない原因を述べる．

2.2　bi-gram 出現頻度分布に基づく手法

[5] の手法では，bi-gram 出現頻度分布をユークリッド空間上のベクトルと捉え，言語特徴とする．識別は，言語ベクトルと識別対象文書ベクトルとの cosign 類似度により行う．[5] の手法の識別式は式 (2) で表される．x はある byte 列，$\hat{L}$ は識別結果，d は識別対象文書，$f_{L_j}(x)$ は言語 j の学習データにおける x の出現頻度，$f_d(x)$ は d における x の出現頻度を表す．

$$\hat{L} = \operatorname*{argmax}_j \frac{\sum_x f_{L_j}(x) f_d(x)}{\sqrt{\sum_x f_{L_j}(x)^2}\sqrt{\sum_x f_d(x)^2}} \quad (1)$$

$$= \operatorname*{argmax}_j \sum_x \frac{f_{L_j}(x)}{\sqrt{\sum_x f_{L_j}(x)^2}} f_d(x) \quad (2)$$

[5] の手法では byte 列の出現頻度が大きいほど，byte 列が識別結果に与える影響が大きくなる．これは，[2, 3] などの統計的手法でも同様である．しかし，これは言語識別において有効な特徴設定とは言えない．それは次の理由による．自然言語において，単語の出現頻度順位と出現確率の間には反比例の関係が成り立つことが経験的に知られている (Zipf 則)．byte 列においても同様の経験則はほぼ成り立つ [6]．すなわち，byte 列の出現確率は極一部の byte 列によりほとんどを占められる．そのためこれらの手法では，出現確率の高い極一部の byte 列のみにより識別結果が左右されてしまう．しかし，類似言語では言語差が小さい高頻度 byte 列が多いうえ，高頻度 byte 列の各文書での出現確率は大きく変動する．そのため，言語間の重なりは大きい．さらに，小規模

文書を対象とした場合，文書中での出現確率の変動はより大きくなる．したがって，これら高頻度 byte 列を重視した特徴設定では誤識別が発生しやすい．

2.3 byte 列出現頻度順位に基づく手法

もう一つの代表的な手法として byte 列出現頻度順位に基づく手法が挙げられる [6]．本手法は byte 1〜5-gram の高頻度 byte 列の出現頻度順位を言語特徴と捉え，各言語モデルと識別対象文書との byte 列出現頻度順位の類似度を比較して識別を行う．[6] のアルゴリズムは以下の通りである．

1. 各言語の学習データにおける byte 列の出現頻度順位 (降順) を求め，上位 k 位を言語モデルとする．
2. 識別対象文書の byte 列の出現頻度順位 (降順) を求め，上位 k 位を文書プロファイルとする．
3. 各 byte 列の文書プロファイル中での順位と，各言語モデルでの順位の差を取り，その総和を各言語のスコアとする．ただし，言語モデル中に出現しなかった byte 列については k+1 を足すこととする．
4. スコアの最も低い言語を識別結果として返す.

出現頻度順位は出現確率よりも byte 列間における差が小さいため，極高頻度 byte 列のみに出力結果が左右されることはない．しかし，順位は極めて揺れが大きく，言語識別のための言語特徴には向かない．例えば，1000byte の文書 100 個において，全文書中で k 位である byte 列が各文書でどのような順位に分布するか調べると，図 (1) のようになる．これは各文書で出現した順位の最大値と最小値を示したものである．この図から，比較的安定する上位 byte 列における言語間差異などは，下位 byte 列における偶然変動に簡単に飲み込まれてしまうことがわかる．したがって本手法もまた，類似言語や小規模文書に対して頑強な特徴設定とは言い難い．

3 提案手法

3.1 低頻度 byte 列の活用

従来手法では，高頻度 byte 列を重視した言語特徴を用いてきた．これは，言語の特徴は機能語・接辞に現れ，それは出現頻度上位の byte 列により代用できるという仮定に基づく．しかし，類似言語間では言語差の小さい高頻度 byte 列が多く，小規模文書においては言語差の大きい高頻度 byte 列が出現しないこともある．したがって，高頻度 byte 列のみで類似言語・小規模文書に対応することは困難である．

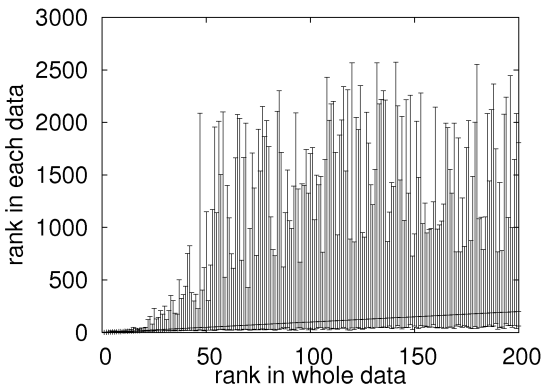

図 1: デンマーク語における出現頻度総合順位 k 位の byte 列の，各文書における順位の変動幅

また，調査の結果，比較的低頻度であっても言語固有の byte 列は多く含まれることがわかった．出現する byte 列としては，長めの機能語や接辞，出現頻度の高い一般動詞の語幹などが存在する．したがって，これらの byte 列を言語特徴に導入することにより，識別性能を向上させることが期待できる．

しかしながら，低頻度 byte 列は識別対象文書中に出現しない可能性が高く，またその種類も多い．そのため，識別に用いる手法は byte 列の未出現に対して頑強であり，識別に効果のある byte 列だけを効率的に活用できるものでなければならない．

3.2 出現 byte 列の積集合サイズによる言語識別

本稿では，低頻度 byte 列を扱うための最も単純な手法として，出現 byte 列の積集合サイズに基づいた言語識別手法を提案する．出現 byte 列の取得は図 (2) のように行う．本手法のアルゴリズムを以下に示す．言語 j の文書集合 $\boldsymbol{T}_j$ を j に対する学習データとする．また，各言語の学習データ中の文書数を $N_j(=|\boldsymbol{T}_j|)$ とする．

1. 各言語の学習データ $\boldsymbol{T}_j$ における各 byte 列 x の document frequency($DF(x)$) を求める．

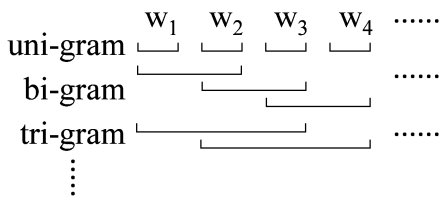

図 2: byte 列の計数方法

2. 各言語において，$DF(x)/N_j$ が一定比率 θ 以上の byte 列集合を求める．これを言語モデルと呼ぶ．
3. 識別対象文書に出現した全 byte 列の集合を求める．これを文書プロファイルと呼ぶ．
4. 各言語モデルと文書プロファイルとの積集合サイズを求め，最も大きい言語を識別結果として返す．

本手法では，文書プロファイル中に出現しなかった byte 列は識別に影響を与えない．そのため，大量の低頻度 byte 列を言語モデルに含んでも悪影響となりにくい．各 byte 列が識別に与える影響は 1 か 0 の 2 値であるため，小規模文書における出現頻度の揺らぎの影響も少ない。また，低頻度 byte 列でも高頻度 byte 列と同等の影響を識別に与えることができる．さらに，識別能力のない byte 列 (全言語モデルに出現する byte 列) をあらかじめ削除できるうえに計算が簡単であり，高速な処理が可能である．

4 評価実験

4.1 実験対象

本手法の有効性を確認するため，類似言語，小規模文書に対する識別実験を行った．対象言語としてノルド語族 (ノルウェー語・デンマーク語・スウェーデン語) を選択した．ノルド語族は，最も類似した言語族の一つとして知られている．

4.2 実験データ

実験データとして，Web 上のニュースサイト (ノルウェー語:Aften Posten[1], デンマーク語:Dagbladet Arbejderen[2], スウェーデン語:Dagens Nyheter[3]) から記事本文を収集した．サイト特有の形式の影響を排除するため，前処理として記事先頭及び末尾の一文の削除，改行の削除，HTML エンコードの復号化などを行った．データは各言語 2000〜3000 文書ずつ，計 19MB が収集された．記事の内容は，政治・経済・芸能など多岐にわたっている．

学習データとして，各言語，1 文書あたり 1kB ずつ 100 文書をランダムに抽出した (学習データ 1)．テストデータは，学習データと重複しないように各言語 2000 文書をランダム抽出した．小規模文書に対する識別性能を比較するため，各文書，ランダム位置から一定サイズを抽出し，識別対象データとする．

また，データ源固有の表現が識別精度に与える影響を評価するため，テストデータと異なるデータ源から収集した学習データセットを作成し，実験を行った (学習データ 2)．

学習データ 2 のデータ源として European Parliament Proceedings Parallel Corpus(EUROPARL)[7] を用いた．本コーパスは EU 議事録を各国語に翻訳したものであり，ニュースサイトとは全く異なる言葉遣いがなされている．デンマーク語・スウェーデン語の学習データを，学習データ 1 と同様の抽出法に従い，本コーパスより収集した．

ただし，ノルウェー語に関しては学習データ 1 と同じ学習データを用いた．これは，低頻度 byte 列の利用により，言語固有の特徴よりも各国ニュースに共通の表現 (地名・人名など) の影響が上回ることがないかどうかを確認するためである．

テストデータは前実験と同じデータを用いた．

4.3 実験結果

学習データ 1 において提案手法と従来手法を比較した結果，低頻度 byte 列の利用により識別性能の向上が見られた．また，識別に使用する byte 列の出現頻度を従来手法と同程度に抑えた実験においても，従来手法と同等の正解率が得られた．表 (1) に各手法による各文書サイズに対する正解率を示す．提示した手法はそれぞれ，TextCat が Cavnar らの手法 [6] を，Cosign が前田らの手法 [5] を，IS-DF が提案手法を表す．括弧内は識別に使用した byte 列の最低出現比率 (θ) である．$\theta = 0.6$ の時，出現頻度上位約

[1] http://www.aftenposten.no/

[2] http://www.arbejderen.dk/

[3] http://www.dn.se/

表 1: 類似言語・小規模データに対する識別性能

	サイズ (byte) 毎の正解率 (%)				
識別手法	50	100	200	300	400
TextCat	82.5	90.9	93.9	95.1	97.6
Cosign	73.7	85.8	92.8	95.2	97.0
IS-DF(0.6)	81.4	90.5	95.3	97.4	98.5
IS-DF(0.35)	86.5	94.3	98.2	99.2	99.7
IS-DF(0.1)	92.0	97.5	99.4	99.8	99.9

表 2: 異なる分野への適応実験結果

	サイズ (byte) 毎の正解率 (%)				
識別手法	50	100	200	300	400
IS-DF(0.6)	78.6	86.4	90.7	92.2	91.8
IS-DF(0.35)	83.6	91.9	96.5	97.9	98.7
IS-DF(0.1)	88.2	95.5	98.4	99.4	99.6

400 個を，$\theta = 0.35$ の時，約 1000 個を，$\theta = 0.1$ の時，約 5000 個を使用した場合とほぼ等しい (ただし，全ての言語に登場した byte 列を削除するため，実際に識別に用いる byte 列数はこの半分程度である). TextCat は上位 400 位までを特徴として使用した．

次に，学習データ 2 における提案手法の識別性能を調べた．結果として，同一の学習源を元にした場合より若干の性能低下が見られた。しかし，依然として低頻度 byte 列の利用により精度が向上しており，提案手法の有効性が示された．識別に使用する byte 列の最低出現比率を変化させた際の，識別性能の変化を表 (2) に示す.

5 おわりに

本稿では低頻度 byte 列の持つ言語識別能力に着目し，低頻度 byte 列を言語特徴として識別に導入した．また，低頻度 byte 列を有効に活用するための最も単純な手法として，出現 byte 列集合の積集合サイズに基づいた識別手法を提案した．

提案手法の有効性を検証するためにノルド語族に対して実験を行い，低頻度 byte 列の利用により識別性能が大幅に向上することを示した．

本手法は，従来手法の弱点であった類似言語や小規模データに対する識別性能を向上させつつ，多言語への応用が簡単であるという byte n-gram を利用する手法の利点も残している．これにより，同一文章中に複数の言語が含まれる多言語混在文書における言語識別など，小規模データの識別を必要とする分野への応用が期待できる．

今後の研究課題としては，まず，本手法の他言語への応用が考えられる．より多数の言語を同時に識別する際にも，高い識別性能を維持できるかなどを調査する必要がある．また，現在利用していない，言語における DF の差などを活用することで，より高い識別精度が得られると予想される．そのための手法の改良も課題である．

参考文献

[1] Emmanuel Giguet. Multilingual sentence categorization according to language. In *European Chapter of the Association for Computational Linguistics SIGDAT Workshop "From Text to Tags: Issues in Multilingual Language Analysis"*, pages 73–76, Dublin Ireland, 3 1995.

[2] Penelope Sibun and Jeffrey C. Reynar. Language identification: Examining the issues. In *5th Symposium on Document Analysis and Information Retrieval*, pages 125–135, Las Vegas, Nevada, U.S.A., 1996.

[3] 北 研二. 確率的言語モデルに基づく多言語コーパスからの言語系統樹の再構築. **自然言語処理**, 4(3):71–82, 7 1997.

[4] T. Dunning. Statistical identification of language. Technical report CRL MCCS-94-273, Computing Research Laboratory, New Mexico State University, 3 1994.

[5] 前田 亮, 関 慶妍, 吉川 正俊, and 植村 俊亮. Web 文章の符号系及び使用言語の自動識別. **電子情報通信学会論文誌**, J84-D-II(1):150–158, 1 2001.

[6] William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *Symposium On Document Analysis and Information Retrieval*, pages 161–176, University of Nevada, Las Vegas., 1994.

[7] Philipp Koehn. A multilingual corpus for evaluation of machine translation philipp koehn. *Unpublished*, 2002.