Support Vector Machineを用いた医学用語の表記ゆれ解消

荒牧英治†　　今井健†　　美代賢吾†　　大江和彦†
†東京大学医学部附属病院

{aramaki, ken, kohe}@hcc.h.u-tokyo.ac.jp
miyo-sup@h.u-tokyo.ac.jp

1　はじめに

他言語からの借用が多い医学用語では「コリアー徴候」「コリエー徴候」といった表記ゆれが数多く存在し，多くのテキスト処理アプリケーションにとって大きな問題となっている．本研究では日本語医学用語の表記ゆれを解消することを目指す．

従来の表記ゆれ研究では，借用語のもととなる英語を推定する back-transliteration を行うものが主であった．この手法は暗に正しい transliteration が存在していることを仮定している．しかし，「アヴォガドロ」「アボガドロ」のように，現実的にはある語に対して，複数の transliteration 結果がともに広く使用される場合も存在し，この問題設定では正解を一意に定めれないことも多い．そこで，我々は，原言語を考慮せず，二つの借用語が表記ゆれ関係にある（同一概念を指す）か否かを判定する二値分類問題として表記ゆれを扱い，これを機械学習するアプローチをとる．

まず，提案手法は，複数の翻訳辞書を用いトレーニングデータを自動構築する．これは，近いスペルを持ち，同じ訳語をもつ語ペア（コリアー徴候（Collier sign）:コリエー徴候（Collier sign）など）を正例とみなし，逆に，近いスペルを持ち，同じ訳語をもたない語ペア（フックス徴候（Fuchs sign）: ヒックス徴候（Hicks sign）など）を負例とみなすことで行う．次に，正例と負例の語ペアのうち，異なっている文字とその周辺の文字と表記の類似度を素性として，SVM[14] にて機械学習を行う．

この方法は，従来の transliteration を生成するといった研究と比べて，現実的な問題設定であるだけでなく，transliteration 以外の原因による表記ゆれ（文字が省略／追加／置換される現象）も自然に扱うことができる．

実験の結果，未知の近いスペルをもった 2 語が表記ゆれ関係にあるか否かを高い精度で判別できた．

本研究により，表記ゆれ解消の可能性を示すことができたと考える．

2　医学辞書における表記ゆれの調査

まず，医学分野にて表記ゆれの問題がどの程度大きな問題であるかを調査した．これは医学分野で広く使われている 2 つの辞書 (DIC1[16] 見出し語数 69,604 語; DIC2[17] 見出語数 27,971 語) を用いて調べた．まず，2 つの辞書のうち，完全に一致した見出し語の数はわずか 10,577 語であった．これは DIC1 の 15.1%，DIC2 の 37.8%にすぎない．

次に，一致しない見出し語のうち，類似した表記（表 1 編集距離類似度 SIM_{ed} で 0.8 以上の値）をもつ語ペアを抽出し，それらが表記ゆれ関係にあるかどうかを人手で判定した．この結果，類似した表記を持つ 5,064 語ペアのうち 1,889 語ペア (37.3%) だけが表記ゆれであった．

この結果により，いくら表層上（文字上）で類似した表記であっても，かならずしも表記ゆれの関係にあるとは言えないことが分かる．例えば，「変異 B 型肝炎ウィルス」と「変異 C 型肝炎ウィルス」は一文字の違いしかなく，高い表記類似度を持つが，その意味が異なるため表記ゆれとは言えない．この問題に対応するためには，単純な表記類似だけでなく，どのような文字がどのように異なるかを考慮する必要があると言える．

3　提案手法

提案手法は，翻訳辞書を用いトレーニングデータを自動構築するモジュール (3.1 章) と得られたデータを用いて学習を行うモジュール (3.2 章) の 2 つからなる．

表 1: 編集距離による類似度 (SIM_{ed}).

2 語 $(t_1, t2)$ i の編集距離による類似度 (SIM_{ed}) は以下の式によって定義する： $$SIM_{ed}(t_1, t_2) = 1 - \frac{\text{EditDistance}(t_1, t_2) \times 2}{len(t_1) + len(t_2)},$$ ここで，$len(t_1)$ は，語 t_1 の長さ（文字数）であり，$len(t_2)$ は語 t_2 の長さ（文字数）である，Edit Distance(t_1, t_2) の編集距離 [7] である．

3.1 トレーニングセットの自動構築

SVM で学習を行うためには正例（表記ゆれ関係にある語ペア）と負例（表記ゆれ関係にない語ペア）の両方が必要となる．

正例の作り方は先行研究でとられていた一般的な方法を用いた．その基本アイデアは，表記ゆれ関係にある二語は近いスペルを持ち，また，同じ英語訳を持つというものである．

この処理は以下の 2STEP からなる．

STEP1: まず，複数の翻訳辞書から，同じ英語訳を持つ語ペアを集める．

STEP2: 次に，2 章で述べた編集距離類似度が 0.8 以上のものを正例とする．

負例も正例と同様に翻訳辞書を用いた方法をとる．すなわち，編集距離類似度が 0.8 以上の異なる英語訳を持つ語ペアを集める．ただし，この方法では負例の数が正例の数をはるかに上回るため，負例を間引いて正例の数と等しくなるようにバランスした．

3.2 SVM を用いた学習

次の問題は，集めた正例・負例から，どのように学習を行うか（素性を作るか）という問題である．提案手法は文字列ベースと類似度ベースの 2 種類の素性を用いる．

文字列ベースの素性は語ペア間で異なっている文字（列）とその周辺の文字からの情報であり以下の 3 種類からなる．図 1 にそれぞれの例を示す．

1. **DIFF**: 二語の間で異なっている文字（列）ペアとその文字種（カタカナ・ひらがな・漢字・その他）．図 1 上では，それぞれ「ア:エ」「カタカナ:カタカナ」となる．
2. **PRE**: LEX の直前の文字と文字種．図 1 上では「リ」「カタカナ」となる．
3. **POST**:LEX の直後の文字と文字種．図 1 上では「ー」「カタカナ」となる．

図 1: 正例と正例から得られる文字ベースの素性．

もうひとつの素性は表記の類似度をそのまま素性とした類似度ベースの素性である．我々は次の 2 つの類似度を用いた．

1. **編集距離類似度** SIM_{ed}: 2 章と同じ編集距離による類似度 (表 1).
2. **翻字類似度** SIM_{tr}: 2 語が同じ翻訳元を持つ確率をスコア化した類似度で文献 [5] によるもの（表 2）．この類似度は入力された 2 語が翻字された場合（カタカナであった場合）のみ用いられる．この類似度を計算するために必要とされるアライメントは複数の辞書 (EIJIRO, EDICT, EDR) に含まれるカタカナ語エントリ 100,128 語から GIZA++*を用いて日本語：外来語の 1:n 対応を求めた．

4 実験

4.1 テストセット

提案手法の精度を調べるため 2 章で構築したコーパス (ALL-SET) を用いた．ただし，提案する翻字類似度は翻字以外の入力に対しては機能しない．そこで，翻字類似度の効果を正確に調べるため，翻字語ペアだけからなる，サブセット（TRANS-SET）を用意した．

1. **ALL-SET**: 5,064 語ペア（そのうち表記ゆれは 1,889 語ペア）．
2. **TRANS-SET**: ALL-SET のうち翻字された語ペア（カタカナ語）だけを集めたサブセット．1,111 語ペア，そのうち表記ゆれは 543 語ペア．

4.2 トレーニングセット

3 章で述べた手法でトレーニングセットを自動構築した（これにはテストセットの辞書は含まれていない）．結果，82,240 語ペアを得た (正例 41,120 ペア，負例 41,120 ペア)．

*http://www.fjoch.com/GIZA++.html

4.3 比較手法

次の手法を比較した．

1. **SIM-ED**: 編集距離類似度 (SIM_{ed}) が閾値 (TH) よりも大きいものを表記ゆれとみなす．
2. **SIM-TR**: 翻字類似度 (SIM_{tr} が閾値 (TH) よりも大きいものを表記ゆれとみなす (TRANS-SET のみ)．
3. **PROPOSED**: 提案手法．ただし，SIM_{tr} を素性として使わない．
4. **PROPOSED+TR**: SIM_{tr} を素性として使う提案手法 (TRANS-SET のみ)．

また，SVM 学習には TinySVM(多項式カーネル (d=2))†を用いた．

4.4 評価

評価は，適合率，再現率，F 値 ($\beta = 1$) で行った．それぞれの定義は文献 [1] と同じである．

4.5 結果

図 2 に結果を示す．ここで，類似度ベースの 2 つの手法（SIM-ED と SIM-TR）は閾値（TH）によってその精度が変化する．もっとも $F_{\beta=1}$ が高い閾値での値を表 3 に示す．

ALL-SET では PROPOSED は SIM-ED を大幅に上回った精度を出している．同様に，TRANS-SET においても，PROPOSED は類似度ベースの手法 (SIM-ED と SIM-TR) を上回っている．これらから，提案する二値分類アプローチの妥当性を示している．また，PROPOSED+TR は PROPOSED よりもよい精度を示した．この結果から翻字類似度が分類に貢献することが分かる．

最後に，本実験の精度と先行研究の精度のどちらが優れているかであるが，先行研究は異なるコーパスを用いており，さらに back-transliteration に焦点を当てているため，正確な比較は困難である．しかし，文献 [6] の 64%，文献 [4] の 87.7%と比較して遜色のない精度だと考えている．

4.6 誤り分析

提案手法（PROPOSED+TR）の誤りは大別すると次の 2 つに原因によることが大きい．

1. **文字種の差異**:「ガン」と「癌」といった文字種の差異を扱えない．今後，読みや音素を素性として組み込む等の処理が必要である．
2. **英語以外の外来語**: 翻字類似度は一般の辞書を用いているので主に原言語が英語となっている．一方，医療用語の原言語はドイツ語，ラテン語など様々な言語が存在しており，精度を下げる原因となっている．

5 関連研究

1 章にて述べたように，多くの関連研究は transliteration に焦点を当ててきた [4, 6]．この手法は，日本語に特化したものでなく，アラビア語 [12, 13]，中国語 [8, 9]，韓国語 [11] ペルシャ語 [5] などにも適応されている．本研究とこれらとの差異は，先行研究が transliteration される前の原言語を生成することを目標としているのに対し，本研究は対象言語の 2 語が表記ゆれかどうかを判別することを目標としている点である．

一方，Yoon 等 [15] は提案手法と同様に判別アプローチをとっているが，彼らは原言語と対象言語のペアが transliteration であるかどうかを識別する点で本研究のタスクと異なる．Bergsma 等 [2] や Aramaki 等 [1] は，本研究と同様に対象言語二語が表記ゆれかどうかを識別する手法を提案している．ただし，彼らの手法は transliteration probability を用いておらず，本研究よりも少ない情報しか利用していない．

Masuyama 等 [10] は表記ゆれ表現の自動収集法を提案したが，これは本研究でいう正例にあたる．本研究は負例をも自動収集する点で異なる．

もう一つの先行研究の分類軸は表記ゆれを扱う単位である．先行研究では文字単位 [9]，音素単位 [6]，両方の単位の混合 [3, 11] が提案されているが，我々は transliteration 以外の表記ゆれを扱うため，文字単位を採用している．

6 おわりに

本稿では翻字確率を考慮した SVM による表記ゆれを判別する手法を提案した．また，その学習のための正例と負例を自動構築する手法を提案した．実験結果は高い精度を示し，提案手法の妥当性を実証的に示すことができた．今後，多くの自然言語処理アプリケーションの基盤として，この手法が用いられることを期待している．

†http://chasen.org/ taku/software/TinySVM/

表 2: 翻字確率による類似度 (SIM_{tr}).

2語 $(t_1, t2)$ の翻字確率による類似度 (SIM_{tr}) は以下の式で定義した:

$$SIM_{tr}(t_1, t_2) = \sum_{s \in S} P(t_1|s)P(t_2|s),$$

ここで，S は t_1 and t_2 から back-transliteration される英語の集合である．
$P(t|s)$ は，t が s から翻字される確率であり，以下の式によって定義される．

$$P(t|s) = \prod_{k=1}^{|K|} P(t_k|s_k),$$

$$P(t_k|s_k) = \frac{\text{frequency of } s_k \rightarrow t_k}{\text{frequency of } s_k},$$

ここで，$|K|$ は語 t の文字数である，t_k は t の k 番目の文字である，s_k は s の k 番目の文字である，“frequency of $s_k \rightarrow t_k$” は s_k と t_k がアライメントされた頻度であり，“frequency of s_k” は s_k の出現頻度であるあ．

表 3: 実験結果

	ALL-SET		
	適合率	再現率	$F_{\beta=1}$
SIM-ED	65.2%	64.6%	0.65
PROPOSED	78.2%	70.2%	0.73
	TRANS-SET		
SIM-ED	91.2%	36.3%	0.51
SIM-TR	**92.6%**	43.9%	0.59
PROPOSED	81.9%	75.6%	0.78
PROPOSED+TR	81.7%	**82.7%**	**0.82**

参考文献

[1] Eiji Aramaki, Takeshi Imai, Kengo Miyo, and Kazuhiko Ohe. Support vector machine based orthographic disambiguation. In *Proceedings of the Conference on Theoretical and Methodological Issues in Machine Translation (TMI2007)*, pp. 21–30, 2007.

[2] Shane Bergsma and Grzegorz Kondrak. Alignment-based discriminative string similarity. In *Proceedings of the Association for Computational Linguistics (ACL2007)*, pp. 656–663, 2007.

[3] Slaven Bilac and Hozumi Tanaka. A hybrid back-transliteration system for Japanese. In *Proceedings of The 20th International Conference on Computational Linguistics (COLING2004)*, pp. 597–603, 2004.

[4] Isao Goto, Naoto Kato, Terumasa Ehara, and Hideki Tanaka. Back transliteration from Japanese to English using target English context. In *Proceedings of The 20th International Conference on Computational Linguistics (COLING2004)*, pp. 827–833, 2004.

[5] Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. Collapsed consonant and vowel models: New approaches for English-Persian transliteration and back-transliteration. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL2007)*, pp. 648–655, 2007.

[6] Kevin Knight and Jonathan Graehl. Machine transliteration. *Computational Linguistics*, Vol. 24, No. 4, pp. 599–612, 1998.

[7] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR*, Vol. 163, No. 4, pp. 845–848, 1965.

[8] Haizhou Li, Khe Chai Sim, Jin-Shea Kuo, and Minghui Dong. Semantic transliteration of personal names. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL2007)*, pp. 120–127, 2007.

[9] Haizhou Li, Min Zhang, and Jian Su. A joint source-channel model for machine transliteration. In *Proceedings of the Meeting of the Association for Computational Linguistics (ACL2004)*, pp. 159–166, 2004.

[10] Takeshi Masuyama, Satoshi Sekine, and Hiroshi Nakagawa. Automatic construction of Japanese KATAKANA variant list from large corpus. In *Proceedings of The 20th International Conference on Computational Linguistics (COLING2004)*, pp. 1214–1219, 2004.

[11] Jong-Hoon Oh and Key-Sun Choi. An English-Korean transliteration model using pronunciation and contextual rules. In *Proceedings of The 19th International Conference on Computational Linguistics (COLING2002)*, pp. 758–764, 2002.

[12] Tarek Sherif and Grzegorz Kondrak. Bootstrapping a stochastic transducer for Arabic-English transliteration extraction. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL2007)*, pp. 864–871, 2007.

[13] Bonnie Glover Stalls and Kevin Knight. Translating names and technical terms in arabic text. In *Proceedings of The International Conference on Computational Linguistics and the 36th Annual Meeting of the Association of Computational Linguistics (COLING-ACL1998) Workshop on Computational Approaches to Semitic Languages*, 1998.

[14] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1999.

[15] Su-Youn Yoon, Kyoung-Young Kim, and Richard Sproat. Multilingual transliteration using feature based phonetic method. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL2007)*, pp. 112–119, 2007.

[16] 伊藤正男, 井村裕夫, 高久史麿. 医学大辞典. 医学書院, 2003.

[17] 日本医学会医学用語管理委員会. 日本医学会医学用語辞典. 南山堂, 2001.

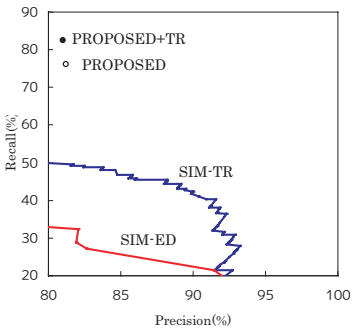

図 2: 精度.