D-014

対訳コーパスに基づく最適なローマ字化システムの構築
Discovering Optimal Systems for Romanization from Bilingual Corpora

田口 恵子† フィンチ アンドリュー‡ 山本 誠一† 隅田 英一郎‡
Keiko Taguchi Andrew Finch Seiichi Yamamoto Eiichiro Sumita

概要

私たちは対訳コーパスから音節単位でのローマ字化システムを開発する方法を提案する．提案手法では，まず，①ノンパラメトリックなベイジアンのアラインメントを用いて，（日本語の場合、カタカナとその英訳からなる）対訳コーパスを音節ごとに対応付けすることによってカタカナをローマ字化する変換ルール候補を生成する．次に，②変換ルール候補から与えられた基準に従って最適なローマ字化変換ルールを選択する．私たちは，与えられた対訳がトランスリテレーションであるか否か分類するタスクの精度で，提案手法と従来手法を評価し，提案手法で開発したローマ字化システムが従来法のそれを上回る精度を達成することを確認した．

1. はじめに

1.1 ローマ字化システム

ローマ字化システムとは日本語をローマ字に変換するルールの集合であり，コンピュータにおけるユーザインターフェースによるテキスト入力や非常用漢字や滅多に使用しない難しい漢字また人名や地名などの固有名詞のルビ振りに利用されている．日本語をローマ字で表現することで日本人だけでなく日本語に不慣れな外国人でも日本語を発音することができる．

日本語には複数のローマ字化システムが存在し[1]，主に英語式であるヘボン式ローマ字化システムと日本式ローマ字化システムの 2 種類が代表的である．日本式ローマ字化システム例を以下に示す．

我孫子	祇園祭
Abiko	Gion maturi

複数あるローマ字化システムは現在，統一されていない．その原因はローマ字の発音は母国語に影響されるものであり母国語の発音に近いローマ字の方が便利であるため，統一されたローマ字化システムが浸透しなかったこととローマ字化システムが混在しても使用できる環境であるので統一する必要がなかったことが挙げられる．

しかし近年，コンピュータで言語を扱う自然言語処理などの研究分野において，ローマ字化システムもコンピュータ上で扱うようになり，その際にどのローマ字化システムを使用するべきかが問題となっている．しかしシステムの全体数を把握できずシステム同士の比較が困難であるため，使用すべきローマ字化システムを一つに決定することは難しい．本稿では統計的な手法による利用用途に適したローマ字化システムを構築する．

1.2 トランスリテレーションマイニング

まずトランスリテレーションとはある特定の言語を記した文字表記を異なる言語の文字によって表記することを指す．

ナス	チョウ
Eggplant	Butterfly
ヒラリークリントン	インターネット
Hillary Clinton	Internet

上段は翻訳ペアの例を下段はトランスリテレーションペアの例を示す．このようにトランスリテレーションは言語変換として文字表記を変換しなければならない．またローマ字化システムのように一意的にルールが決まっておらずより複雑であり，さまざまな研究が進められている．トランスリテレーションで前処理として原言語と目的言語の文字体系の統一のためにローマ字化システムが使用されている[2, 3]．トランスリテレーションペアはよくカタカナと英語とで対訳がつけられており本稿でもそれに従う．

トランスリテレーションマイニングは単語ペアを翻訳ペアかトランスリテレーションペアか判別することであり，学習データから翻訳ペアを取り除くことでトランスリテレーションの学習精度が向上する．トランスリテレーションマイニングの研究[4, 5]においてもトランスリテレーションと同様にローマ字化システムは利用されている．

トランスリテレーションマイニングは単語ペアの類似性から翻訳/トランスリテレーションを識別する．しかし従来のローマ字化システムによるローマ字と英語を比較した場合，スペリングが異なる部分があり単語の類似性を正しく評価できず，トランスリテレーションマイニングの性能低下を招いている．本稿では 対訳コーパスから統計的にローマ字化システムを開発することで目的言語の綴りに近いローマ字を生成しトランスリテレーションマイニングの性能の向上を図る．

2. 関連研究

S. Jiampojamarm らの研究[6]のように統計的観点から従来のローマ字化システムを使用せずにトランスリテレーションマイニングのためのローマ字化システムがすでに開発されている．

そのローマ字化システムでは二言語間でカタカナ１文字に対し単一のアルファベット文字で対応が取られており，アラビア語やロシア語などあらゆる言語でも適用できる汎用性の高いシステムが実現されている．しかし問題点として中国語や日本語などのカタカナ１文字に対し複数のアルファベットの対応を取るべき言語では上手くローマ字化ができずにトランスリテレーションマイニングの性能の低下を招いている．

3. 提案手法

3.1 アプローチ

本研究では対訳単語コーパスから統計に基づき利用用途に適したローマ字化システムを構築する．提案手法の利点は対訳コーパスからローマ字化システムを学習できる点と適用するタスクに合わせてローマ字化の選択基準を自由に変更できる点である．

まず対訳コーパスからベイジアンアライメント[7]を用いて日本語(カタカナ)の音節ごとに英語の対応付けを行い，ローマ字化変換ルール候補を得る．その変換ルール候補の中から期待編集距離（Expected Edit Distance：EED）を基準として最も適切なローマ字化変換ルールを一意的に決定する．ベイジアンアライメントと EED の詳細を以下に示す．

3.2 ベイジアンアライメント

ベイジアンアライメントは音節単位の対応付けを行うノンパラメトリックな手法である． ベイジアンモデルは過学習なしで一貫性のあるアライメントが可能である．そのため従来アライメントが難しかった一対多または多対多アライメントが可能である．またノイズを含むデータでも精度の高いアライメントができる．

本研究では 2 種類のローマ字化システムを開発した．unigram システムはカタカナ一文字に対し複数のアルファベットを対応する一対多アライメントをとり，n-gram システムは拗音や促音などを含む複数のカタカナに対し複数のアルファベットを対応する多対多アライメントをとる．アライメントの例を図 1 に示す．

ユ X 0.00684931506849315	ト o 0.0022883295194508
ユ YOU 0.00684931506849315	ッフェ ffe 0.666666666666667
ユ LLOU 0.00684931506849315	ッフェ uffet 0.333333333333333
プ P 0.771929824561403	ッダー der 0.5
プ B 0.0567595459236326	ッダー ddha 0.5
プ PE 0.0567595459236326	ショ tio 0.732142857142857
プ PU 0.0412796697626419	ショ sho 0.0892857142857143
プ PP 0.0123839009287926	ショ i 0.0892857142857143
プ PF 0.0103199174406605	ショ sh 0.0357142857142857
プ POU 0.0103199174406605	ショ e 0.0178571428571429

(a) unigram (一対多アライメント)　　(b) n-gram (多対多アライメント)

図 1　日英対訳ペアによるアライメント例

図 1 はカタカナ－ローマ字対とその発生確率を示す．アライメントをとった全てのカタカナとローマ字対がローマ字化変換ルール候補である．

3.3 ローマ字化の選択基準（Expected Edit Distance：EED）

ローマ字化変換ルールは一意的でなければならないのでベイジアンアライメントで求めたルール候補から唯一のルールを決定する．本稿ではトランスリテレーションマイニングの単語の類似性の計測によく用いられる編集距離(レーベンシュタイン距離)の要素をローマ字化システムの選択基準に取り入れることで更なる分類精度の改善を目指し，発生確率とレーベンシュタイン距離を組み合わせた EED を基準として変換ルールを選択する．

対訳コーパスの原言語(カタカナ)と目的言語(英語)を $S = (s_1, s_2, \ldots, s_I)$ と $T = (t_1, t_2, \ldots, t_I)$ とする．各 s_i と t_i はそれぞれの文字体系における書記素とする．

Π と Ω はそれぞれの文字体系における音節とアルファベットの集合である．ローマ字化変換ルール R はタプル(o_j, r_j)の集合である．o_j と r_j は日本語の音節と英語のアルファベットである：$\forall j\ o_j \in \Pi,\ r_j \in \Omega$

$$R = \{(o_1, r_1), (o_2, r_2), \ldots, (o_J, r_J)\} \quad (1)$$

r_j はベイジアンアライメントによって o_j とアライメントがとられたアルファベット候補集合 C_j から選択する：$C_j = (c_1, c_2, \ldots, c_K)$．　変換ルール$(o_j, r_j)$は式(2)に表す期待編集距離が最小値をとる r_j を選択する．

$\Phi : \Pi \rightarrow \Omega$ は R によって定義されるローマ字化関数とする．

$$\phi(o_j) = \underset{c_k \in C_j}{\operatorname{argmin}}\ E[D(c_k)] \quad (2)$$

$D(c_k)$は変換ルール候補(o_j, c_k)のレーベンシュタイン距離のコストを表す．対訳コーパス中の o_j,に対して候補が 1 つしかない場合は，そのコストは変換ルール候補 c_k と o_j とアライメントをとる目的言語のアルファベット $\psi(o_j)$とのレーベンシュタイン距離 $LD(c_k, \psi(o_j))$とする．

対訳コーパスにおけるこのコストの期待値は以下のように計算される．

$$E[D(c_k)] = \sum_{l=1..K} p(c_l) LD(c_k, c_l) \quad (3)$$

レーベンシュタイン距離のコストの期待値，つまり発生確率とレーベンシュタイン距離を組み合わせた期待編集距離が最も小さくなる変換ルールを選択することで生成されるローマ字の偏りをなるべく抑え，より英語に近いスペリングで日本語をローマ字化する．

4. 開発結果

4.1 実験データ

ローマ字化システムの学習データは，[8]の日英トランスリテレーションマイニング対訳コーパスを用いる．このデータはウィキペディアの他の言語版の記事への内部リンクから抽出されたタイトル 4339 組で構成されている．日本語タイトルは全てカタカナのみで記述されている．さらに各単語ペアはトランスリテレーションペアであるか否か注釈がつけられている．全 4339 単語ペアのうち 3800 組が音訳ペア，539 組が翻訳ペアである．提案手法はノイズを含むデータでも学習が可能であるため全データで学習を行い，またトランスリテレーションマイニングにおいても全データを利用した．

4.2 提案システム

興味深いことに提案システム(unigram, n-gram)は従来システムであるヘボン式・日本式システムと 3 割ほど異なる変換ルールを学習した：n-gram システムとヘボン式シ

ステムの変換ルールは 32%異なっている．表 1 に提案システムと従来システムの変換ルールの主な違いを表す．

人間と機械が生成したローマ字化システムの大きな違いとして「ル」が挙げられる．提案システムは“R”よりも“L”を変換ルールとして選択している．これは“R”よりも“L”の方が英語で頻繁に「ル」と対訳が取られていることを意味する．さらに英語の綴りでは多くのウ段の変換ルールにおいて“U”対応付けられないので提案システムでは“U”が欠落しているローマ字変換ルールもあった．

表 1　ローマ字化システムの主な変換ルールの違い

Kana	Hepburn (Nihon-shiki)	N-gram	Unigram
カ	KA	CA	CA
ク	KU	C	K
グ	GU	G	G
ケ	KE	CE	KE
コ	KO	CO	CO
シ	SHI (SI)	SI	S
ジ	JI (ZI)	GI	G
ス	SU	S	S
ズ	ZU	S	S
ゼ	ZE	SE	SE
ツ	TSU (TU)	TS	TS
ト	TO	T	T
ド	DO	D	D
フ	FU (HU)	F	F
ブ	BU	B	B
プ	PU	P	P
ム	MU	M	M
ユ	YU	U	U
ヨ	YO	JO	JO
ル	RU	L	L
キャ	KIYA(KYA)	CA	-
クィー	KUII	QUEE	-

5. 評価実験

5.1 トランスリテレーションマイニング性能

提案手法で開発したローマ字化システムがトランスリテレーションマイニングに適しているかどうか評価実験を行った．ここではトランスリテレーションマイニングは対訳単語コーパスの単語ペアをトランスリテレーションペアであるか否か判断する二項分類とする．本稿では，分類の基準として標準化編集距離(Normalized Edit Distance : NED)を使用した．類似した方法が[4, 5]でも採用されている．英単語と各システムによって生成されたローマ字列との NED を計測し分類を行った．ここでは NED はレーベンシュタイン距離を編集経路の和で割ったものである．編集経路の和で割ることで測定する 2 つ文字列の長さの違いによる値の偏りを除去することができ NED の値域を[0,1]に留めることができる．

NED を閾値として各分類器の性能を表す ROC 曲線(Receiver Operating Characteristic curves ）を図 2 に示す．ROC 曲線は弁別閾値を変化させて二項分類器の性能を示すグラフである．図 2 は各ローマ字化システムが用いられた以下の 5 つの分類器でトランスリテレーションマイニングの性能を比較している：

提案システムによる分類器
　Unigram，N-gram
従来システムによる分類器
　Hepburn，Nihon-shiki，Single-character

Single-character は関連研究で紹介した[6]の一対一アライメントをとったローマ字化システムによる分類器である．

ROC 曲線の縦軸は True Positive Rate(TPR)であり，コーパス中の全トランスリテレーションペアのうち二項分類によって正しくトランスリテレーションペアと判断された割合を表し，横軸は False Positive Rate(FPR ）でありコーパス中の全翻訳ペアのうち二項分類によって誤ってトランスリテレーションペアだと判断された割合を表す．性能が高い分類器は TPR は高い値，FPR は低い値をとる．つまり ROC 曲線はグラフの縦軸と上線に沿うような曲線が分類器の性能が最も良いことを示す．

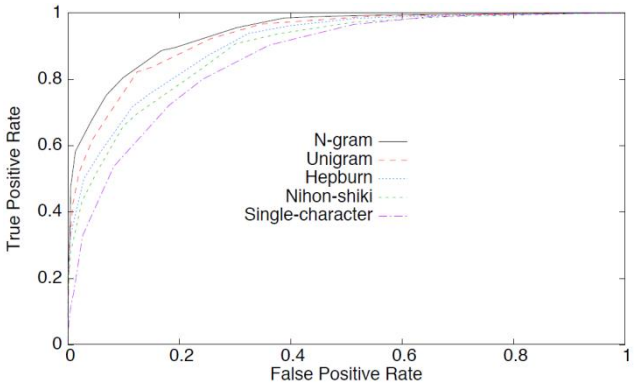
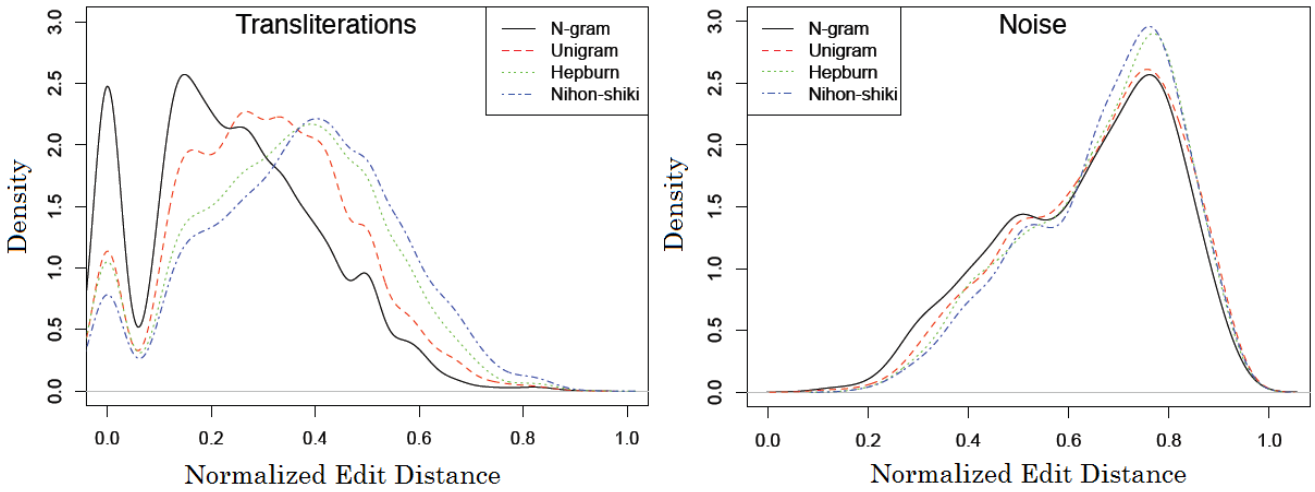


図 2　各ローマ字化システムによる分類器の ROC 曲線

グラフより N-gram の二項分類器が最も良い結果であった．また従来のローマ字化システムにおいては日本式よりもヘボン式の方がトランスリテレーションマイニングに適していることがわかった．ヘボン式は外国人が日本語を読むために作られたローマ字化システムであるので，英語と類似したスペリングでローマ字化が行われており，日本式よりも性能の高い二項分類に成功したと考えられる．Single-character は最も性能が低く，日英のトランスリテレーションマイニングにおいては一対多と多対多アライメントが有効であると推測される．

5.2 統計的有意性

分類器の性能を数値で表すために ROC 曲線の曲線化面積(The Area Under Curve : AUC ）を適用した（ 表 2 ）ROC 曲線下面積は分類器の分類の正確さの確率を表す．またローマ字列の平均フレーズ長（ Length ） と平均レーベンシュタイン距離（ Mean LD ）も示す．対訳コーパスの英単語の平均フレーズ長 L = 6 であり，N-gram と Unigram の平均フレーズ長と一致した．Mean LD からも



(a)　トランスリテレーションペアの NED の分布　　(b)　翻訳ペアの NED の分布

図 3　トランスリテレーションペアと翻訳ペアにおける標準化編集距離の分布

N-gram でのローマ字化は英語と綴りが類似していることが分かる．さらに[10]で提唱されている方法で ROC 曲線下面積に有意差検定を行い，各曲線の統計的有意性を検証した．全ての隣接する曲線の AUC において統計的有意差が認められた．（$\alpha < 0.05$）

表 2　ROC 曲線に基づく AUC

Approach	AUC	Length	Mean LD
N-gram	0.942	6	2.6
Unigram	0.927	6	3.1
Hepburn	0.907	7	3.7
Nihon-shiki	0.892	7	4.0
Single-character	0.867	3	4.6

6. 分析

6.1 ローマ字化システムの効果

ローマ字化システムがトランスリテレーションマイニングの性能にどのように貢献するのかそのメカニズムについて検証する．図 3 の（a）・（b）はデータのトランスリテレーションペアと翻訳ペアとを個々に各ローマ字化システムでローマ字化し標準化編集距離（NED）を計測し，その NED の確率密度関数（Probability Density functions：PDF）を求めカーネル密度推定を行ったグラフである．

図 3 の（b）のグラフから翻訳ペア（ノイズペア）における NED の分布はローマ字化システムに関係なく似たような結果となることがわかる．統計学的に分布が等価であるかを評価するために，ノンパラメトリックな検定であるコルモゴロフ-スミルノフ検定（Kolmogorov-Smirnov test）をノイズペアの分布に対して行った．N-gram とヘボン式/日本式を除く，全ての分布同士は α=0.05 で等価であった．NED＝0.5 付近で分布が少し増えるのは以下の例のように本来はトランスリテレーションペアであるが省略や略称により生じる部分的な一致のある翻訳ペアが原因であると考えられる．

パソコン　　ワープロ
Personal Computer　　Word Processor

図 3（a）はトランスリテレーションペアの NED の分布である．開発したローマ字化システムの分布が他の分布に比べて左に位置することから開発したローマ字化システムがより良い性能であることがわかる．図 3（a）のデータに対し統計学的に分布が異なるかどうかを評価するために，ノンパラメトリックな検定であるウィルコクソンの符号順位検定（Wilcoxon signed-rank test）をトランスリテレーションペアの分布に対して行った結果，全ての分布は有意に違いがあった（α＝0.05）．

さらに興味深い点では，英語と日本語から変換したローマ字がまったく同一である NED=0.0 において N-gram システムは他のシステムのうち最も性能の良い Unigram システムの約 2 倍も英単語と同じスペリングでのローマ字化に成功しているという点である．

以上より，提案手法は英語との編集距離を抑えることによりトランスリテレーションペアと翻訳ペアの 2 つの分布の違いを際立たせトランスリテレーションマイニングの性能を向上させたといえる．

6.2 ローマ字化システムの比較

ローマ字化システムのスペリングと英語のスペリングがどの程度違いがあるか調べるために対訳コーパスの英語に対して n-gram ローマ字化システムと日本式ローマ字化システムで変換したローマ字の発生確率を比較した．図 4 は基準となる英語に対してそれぞれの n-gram システムと日本式システムのローマ字の発生確率の相対的な違いを示す．

図 4 から全体的に n-gram システムの方が英語に近いことが確認できる．日本式システムと英語の大きな違いが生まれる原因は，日本式システムは子音と母音を常にセット

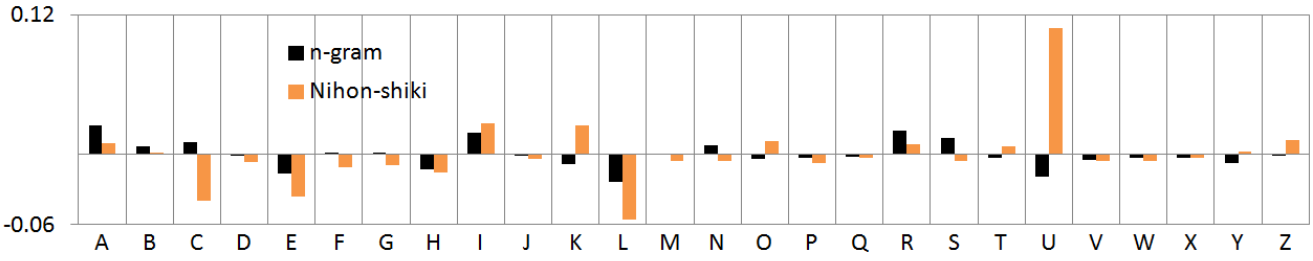


図 4　n-gram システムと日本式システムによるローマ字の発生確率の差異

で生成するので“U”を過剰生成する．また“C”と“L”は日本式システムでは生成する変換ルールを持たないため“C”と“L”は英語よりも大幅に少なく，代わりに生成される“K”と“R”は過剰生成している．2 つのシステムの違いを実際の単語例で表すと，単語「スクール（英：SCHOOL）」は日本式システムでローマ字化すると‘SUKUURU’であり開発された n-gram システムを使用すると‘SCOOL’である．このように n-gram システムは日本式システムよりもより英語のスペリングに近いローマ字列を生成している．

7. 他言語におけるローマ字化システムの適用

日英のトランスリテレーションマイニングでローマ字化システムの有用性を示したが他言語でも同様の方法でトランスリテレーションマイニングに適したローマ字化システムの構築が可能であるか実験を行う．対象言語(文字表記)は中国語(漢字)ー 英語とロシア語(キリル文字) ー英語の 2 種類である．

7.1 中国語ー英語

中国語は Pinyin と呼ばれるローマ字化システムが確立しているが，漢字 1 文字に対しローマ字複数の対応をとる言語であるので英語と中国語でもローマ字化システムを開発しトランスリテレーションマイニングを行った．

NEWS2010 Shared Task[10]で使用された対訳コーパスを使用した．しかし seed データ（トランスリテレーションペア 1000 組）と reference データ（トランスリテレーションペア 621 組）でローマ字化システムを開発したところ，変換ルールの数が seed データでは 214 組，reference データでは 385 組であった．reference データが seed データの約 0.6 倍と少ないにも関わらず，変換ルールは seed データよりも約 1.5 倍多かった．

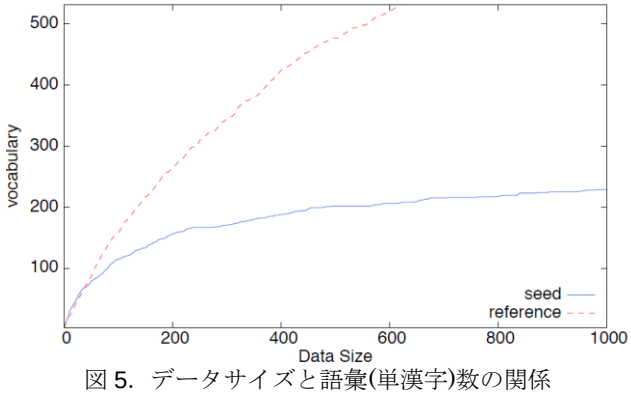


図 5．データサイズと語彙(単漢字)数の関係

対訳コーパスのデータサイズとコーパスに含まれる語彙（単漢字）数の関係を表すグラフを図 5 で示す．変換ルールの数は対訳コーパスに含まれる漢字の種類によって決定される．中国では日常で使用する場合でも 5,000 種類以上の漢字が存在するが全ての漢字が平等に使用される訳ではなく，外国人の名前に使用される漢字の種類が決まっている．seed データは外国人や外国の地名が多く含まれ，使用されている漢字が限定されているためデータサイズが 200 ペアを超えたあたりから含まれる語彙の数が収束し，変換ルールの種類が少なかったのではないかと考えられる．さらに reference データは漢字ー英語の対訳ペアだけでなく漢字と Pinyin の対訳ペアを含んでいる．つまりデータの性質の点においても seed データと reference データは異なっており含まれる漢字の種類が異なり，またその漢字の種類をカバーするほどのデータサイズがなかったために変換ルールの学習が上手くいかなかったと考えられる．提案手法は対訳コーパスから直接学習できるが，コーパスサイズとローマ字化システムの性質の関係についてさらに研究を進める必要がある．

7.2 ロシア語ー英語

ロシア語はキリル文字と呼ばれる文字体系を持つ言語であり，ロシアのローマ字化システムは日本ほど多くはないが複数あり，2010 年にパスポートに使用されるローマ字化システムが新しく採用された．ロシア語と英語のトランスリテレーションマイニングの性能実験でも中国語と同様に NEWS2010 Shared Task のデータを用いた．1000 組のトランスリテレーションペアをもつ seed データで学習しローマ字化システム EnRu システムを開発した．train データは複数の単語列同士をペアとして対訳がとられており各単語ごとに対訳が対応していなかったため，858 組のトランスリテレーションペアをもつ reference データを各言語の単語の順番を並び変えて非トランスリテレーションペアを生成し，トランスリテレーションペア 858 組，非トランスリテレーションペア 3746 組からなるデータを評価データとしてトランスリテレーションマイニングを行った．

図 6 はロシア語と英語の各分類器における ROC 曲線を示す．比較システムは 2010 年に採択されたパスポート規格のローマ字化システム Passport(2010)と一対一アライメントがとられているローマ字化システム Single である．英露のトランスリテレーションマイニングにおいてはどのローマ字化システムを利用してもほぼ同じ高い性能が出ている．ロシア語は日本語より英語に近い言語であり一対一アライメントで対応が可能な言語であることが原因であると考えられる．実際に Passport(2010)のほとんどが一対一アライメントがとられている：キリル文字 33 文字中 27 文字が一対一アライメントをとっている．

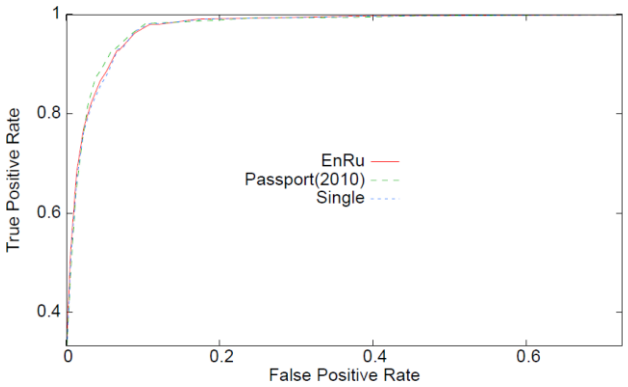


図 6　ロシア語と英語の各分類器における ROC 曲線

8. まとめ

本論文では，対訳コーパスを利用したローマ字化システムの統計的開発技術について記述した．まず単語の対訳コーパスからノンパラメトリック手法であるベイジアンアライメントで音節ごとに対応付けを行い，ローマ字化変換ルールの候補を生成する．そして最も良い変換ルールが予め定められた基準に従って候補から 1 つ選択される．

私たちは日本語・中国語・ロシア語の 3 カ国語で提案手法を適用した．本稿では学習データとしてウィキペディアの他の言語版の記事への内部リンクから取り出されたタイトルのコーパスを使用し，選択基準として期待編集距離を使用し，評価実験としてトランスリテレーションペアであるか否かの二項分類を行うトランスリテレーションマイニングタスクにこの技術を適用した．日本語では，トランスリテレーションマイニングの性能はどのローマ字化システムを使用するかに強く依存することが判明した．さらに開発したローマ字化システムは従来のローマ字化システムより有意に判別性能が優れていることが示された．中国語では，日本語よりも遥かに語彙数が多く，データに含まれる漢字の種類が異なっていたことからローマ字化変換ルールはデータサイズに大きく影響されることが分かった．ロシア語では，ロシア語は英語と類似した言語であるために従来のシステムでも充分に高い性能を得ており開発したローマ字化システムによる分類器でも高い性能が得られた．

9. 今後の課題

提案手法はノイズデータから学習が可能であるので理論的には言語に依存せずに適用することができ，必要となる資源は対訳単語コーパスと対応付けされたローマ字化変換ルールの候補からルールを選択するための明確に定義された基準だけである．将来的には異なる基準を用いてローマ字化システムを開発し，ローマ字化システムに与える影響を具体的に研究する予定である．またどのくらいのデータサイズがあれば十分なローマ字化変換ルールが獲得できるか検証を行う．

参考文献

[1] J.Quint, ”Automatic Japanese transliteration with a formalism for presyntactic analysis”, Natural Language Processing and Knowledge Engineering on 2003.International Conference， pp.512-518，2003.

[2] K. Knight and J.Graehl, “Machine transliteration”, *Computational Linguistics*，pp. 599-612，1998.

[3] Y.Qin, “Phoneme strings based machine transliteration”, Natural Language Processing andKnowledge Engineering (NLP-KE) on 2011 7th International Conference，pp. 304-309，2011

[4] W.Aransa, H.Schwenk, L.Barrault, and F.Le Mans, “Semi-supervised transliteration mining from parallel and comparable corpora”, *Proceedings IWSLT 2012*, 2012

[5] O. Htun, A. Finch, E. Sumita and Y. Mikami, ”Improving Transliteration Mining by Integrating Expert Knowledge with Statistical Approaches”, *International Journal of Computer Applications*，57，November 2012.

[6] S. Jiampojamarm, K. Dwyer, S. Bergsma, A. Bhargava, Q. Dou, M. Kim and G. Kondrak, “Transliteration Generation and Mining with Limited Training Resources”, Proceedings of the 2010 Named Entities Workshop, pp.39-47, 2010

[7] A.Finch and E. Sumita. , “A Bayesian Model of Bilingual Segmentation for Transliteration.”, In M. Federico, I. Lane, M. Paul and F. Yvon, editors, *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pp.259-266, 2010.

[8] T. Fukunishi, A. Finch, S. Yamamoto, and E. Sumita,, “Using features from a bilingual alignment model in transliteration mining”, In 2011 Named Entities Workshop，pp.49，2011

[9] J. Hanley and B. McNeil, “The meaning and use of the area under a receiver operating characteristic (roc) cueve”, Radiology, Vol.143, pp.29-36, 1982

[10] A. Kumaran, M. Khapra and L. Haizhou, “Report of NEWS 2010 transliteration mining shared task”, Association for Computational Linguistics , pp.22-28, 2010