Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

小規模なコーパスを用いた仮名漢字混じり文と仮名文の対応づけ

山　口　文　彦[†1]

イースター島には Rongorongo と呼ばれる未解読文字が遺されている．Rongorongo を歌を歌うように読んでいたという記録があるため，Rongorongo の記号列と現地の古い歌が対応するか否かを統語的に調べる研究を行った．しかし，未解読言語が対象であるために正解を設定できず，結果を評価することが難しい．そこで，既知の言語を用いた同様の問題を設定し，手法の評価を行った．

本論文では，日本語の仮名漢字混じりの文と，片仮名のみの文を用いて，それらの対応付けを見つけるある手法について報告する．この手法は，文を文字単位に分割する以上の言語に関する知識をできるだけ用いないように留意している．結果として，仮名漢字混じり文とその読みを正しく表している仮名文との対応付けを見つけることができた．また，出現頻度の高い文字の種類が少ない場合に誤判定されることが分かった．

Correspondence between Kana-Kanji sentence and Katakana sentence using small corpus

Fumihiko Yamaguchi[†1]

Undeciphered script called Rongorongo is remained in Easter Island. There are some records that Rongorongo is read as singing. Thus, the correspondence between Rongorongo and the old local chants are researched syntactically. However, as the collect answer is unknown, it's difficult to evaluate the results. Therefore, by setting similar problem in known language, the method is thought to be evaluated.

In this paper, a method is reported to find the correspondence between Japanese sentence, which contains both kana and Kanji, and the sentence, which contains Katakana. As results, the collect correspondence is found which is between a Kanji sentence and the Katakana sentence which represents its reading. And it is clarified that the method misjudges when there are few number of kinds of frequent characters.

1. はじめに

イースター島には Rongorongo と呼ばれる未解読な文字と考えられている記号列がある (図 1)．タヒチに派遣されていた神父 Jaussen の記録によると，Metoto という名のイースター島出身者が Rongorongo を読む際の様子が歌を歌うようであったとされている．そこで，Rongorongo の記号列と，同じくイースター島に残された歌をラテン文字によって記録した資料を用いて，両者に含まれる記号の出現順序に対応付けが見つけられるか否かを調べ，5 月の自然言語処理研究会において報告した[2]．結果として，頻度 2 以上の記号が出現する順序が一致するような，Rongorongo 符号列とラテン文字列 (歌詞) の組がいくつか見つかった．しかし，対象が未解読言語であるために，正解となる対応付けを設定することができず，したがって定量的な評価ができなかった．そこで本論文では，手法の評価をするために，正解を決めることができるような同様の問題を設定する．

同様な問題設定をするために，まず Rongorongo の特徴について述べる．Rongorongo は主に木片に刻まれた記号の列であり，それらの木製品は 18 世紀から 19 世紀に製作されたと考えられている[4]．Rongorongo はイースター島で独自に発達した文字体系であるとも言われているが，他の文化圏から隔絶した状態で発達したため，ヒエログリフにおける Rosetta Stone のような対訳コーパスが存在せず，未解読である．Rongorongo が刻まれた木製品は現在 26 個が残っている．Mètraux は，Rongorongo の記号が，約 120 の記号に分類できるとし，文字の種類が 120 というのは，表音文字であると考えるには多過ぎ，表意文字であると考えるには少な過ぎると指摘している[3]．Barthel は記号をさらに細かく約 630 種類に分類し，それぞれを 3 桁の数字で表した．これは Barthel 符号と呼ばれる．Barthel 符号を用いることで，Rongorongo を計算機可読なテキスト情報として扱うことができる．Rongorongo の記号の列には，空白などの区切り文字と思われるものは，一つの例外を除いて現れず[⋆1]，(ヨーロッパの言語のように) 単語ごとの区切りは見られない．また，句読点に

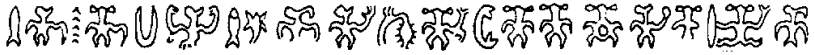


図 1　Aruku Kurenga と呼ばれる Rongorongo の Verso 側の 1 行目

†1 東京理科大学 大学院 理工学研究科 情報科学専攻
Department of Information Sciences, Graduate School of Tokyo University of Science

⋆1 Santiago Staff と呼ばれる棒状の木製品には，Barthel 符号で 2474 個の記号が刻まれており，そのうち 97

Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

相当する記号も不明であり，列の途中で記号が途切れていることもないため，文や段落ごとに区切られている様子も見られない．なお，26 個の木製品全体でも，刻まれた記号の個数は約 15,000 個であり，Rongorongo は小さなコーパスであると言える．

統計的自然言語処理の手法は，言語の特徴への依存が少ないので未解読言語にも適用できると考えられる．そうした研究の例としては，Snyder らによる Ugarit 文字 (楔形文字) とヘブライ語の文字の対応付けがある[1)]．これら 2 つの言語は地理的にも時代的にも近く，関連が深いことが知られている．彼らはこれらの言語における対訳ではないコーパスを用いて，文字同士および同根語である単語同士を対応づけることに成功している．どちらの言語においても単語を分かち書きしており，単語の語尾変化に着目するなどの知識を利用している．

しかし，小さなコーパスが対象である場合，統計的な手法は不利であると考えられる．ここで，逆にコーパスが小さいことから，全探索によって対応付けが見つけられるのではないかと考えた．

ある言語について，文字による表記と，発音の記録 (表音文字の列) があり，いずれも計算機可読なテキスト情報となっているとする．それぞれの文字列がどのように読まれるかが分からないとき，文字列の読みとなっている表音文字の列を見つけることが目的である．書かれた文字が表音文字であればもちろん，表意文字である場合にも，それぞれの文字には読み方があると考えられる．ある発音 (読みの列) の記録 Y がある文字列 K の読みになっているとすると，K の中に含まれる文字の読みは Y に含まれる．このとき，複数の文字が K 中に出現する順序と，それらの読みが Y 中に出現する順序は一致すると考えられる．このような対応付けの有無を統語的に見つける方法について考える．

例えば “ABACB” という文字列の中には “A” と “B” が複数回登場する．このとき，“12132” の中の “1” と “2” は，それぞれ “A” と “B” が “ABACB” に現れるのと同じ順序で現れる．しかし “13221” という列の中には，{A, B} と {1, 2} をどのように対応づけても，(“13221” の部分列の中にも) 同じ順序で現れることがない．このとき，“ABACB” に現れる文字の出現順序は “12132” に現れる文字の出現順序と対応し，“13221” に現れる文字の出現順序とは対応しないと考える．

このように，記号の並びだけを見て，与えられた文字列の読みとなっている表音文字列を見つけることを考える．本研究の最終的な目的は Rongorongo と歌の対応付けを見つけることにあるが，本論文では，手法自体を評価するために正解を決めることができる問題として，日本語の仮名漢字混じり文と片仮名のみの文を用いる．

個が一本の縦線であって，この縦線は区切り文字のようにも見える．なお他の木製品にこのような一本の縦線は現れない．

2. 問題の記述

ある言語について，文字による表記 K と，発音の記録 Y があり，いずれも計算機可読なテキスト情報となっているとする．K に含まれる文字と Y に含まれる文字の対応付けを考えたとき，K 中に出現する文字の順序と，対応する文字が Y 中に出現する順序が一致するような対応付けが存在するか否かを調べたい．

ここで，K は仮名漢字混じり文，Y は仮名文を想定している．本論文は，上記のような対応付けの有無を調べることで，K の読みである仮名文 Y を見つけることができるかどうかを実験的に確かめたことの報告である．なお，K と Y に登場する文字集合には重複が無く (句読点も「、」「。」と「,」「.」で分けている)，句読点も単なる文字として扱う．

3. 対応付けを見つけるアルゴリズム

K と Y をそれぞれ仮名漢字混じりの入力文および読みの表記である片仮名の入力文であるとする．

漢字と読みの関係であることを考えると，K 中の 1 文字は Y 中の (長さ 2 以上の) 文字列に対応すると考えるのが自然であるかもしれない．しかし，例えば「問」は「モン」と対応するのではなく,「モ」とだけ対応すると考えてもよい．なぜなら,「問」が出現するのと同じ順序で「モン」が出現するとすれば，やはり同じ順序で「モ」が出現するからである．同様に「問題」が「モンダイ」と対応すると考えるのが自然かも知れないが，K の中に「問」も「題」も「問題」という形でのみ登場する場合には,「題」が「ン」と対応づけられても，出現順序には影響しないと考えられる．

文字同士の対応付けの仮定増やしながら，K と Y をそれぞれ順に読み進めていくアルゴリズムを考える．このとき，ある時点までの計算によって K 中の文字 k と Y 中の文字 y の対応が仮定されているとする．しかしそのような場合でも，y は k 以外の文字の読みとしても用いられている可能性がある．そこで，k と y の対応が仮定されている場合に Y 中に y が出現しても，それが k と対応付けられる場合と，対応しない場合の 2 つの場合を考える必要がある．

出現順序の組み合わせ方が一致するかどうかを調べるので，出現頻度が 1 以下の文字については考慮しない．そこで，K 中に現れる頻度 2 以上のそれぞれの文字について，Y 中

Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

に現れる 1 文字を割り当てることにする．K 中に現れる異なる文字が Y 中の同じ文字と対応することもあるとする．なお，そのような対応付けをすべて列挙しようとすると，ある程度の長さの文字列同士が対応する場合に組合せ論的に対応付けが見つかることになる．例えば「問題」と「モンダイ」では,「モンダイ」の 4 文字から 2 文字選ぶ組合せの数 (6 通り) の対応付けが考えられる．前述のように，そのいずれの対応付けも許容するので，すべての対応付けを列挙することには多くの計算資源を必要とすることが予想される．そこで，ここでは対応付けの有無だけを調べることにし，最初に対応付けが見つかった時点で，対応すると判定して終了するアルゴリズムを考える，逆に，対応付けが無いことは，すべての組合せを調べた上で判定されるものとする．

対応付けられるか否かを判定するアルゴリズムを，疑似コードを用いて図 2 に示す．関数 `corres` は，K と Y の何文字目までを読み込んだかをそれぞれ i_K, i_Y に，計算途中で得られた文字同士の対応付けを H に受け取る．ここで H は K に現れる文字と Y に現れる文字の組を要素とする集合である．また，$A[i_A]$ は A の i_A 番目の文字を表し，l_A は A の長さを表す．A の先頭の文字は $A[0]$ で表されるものとする．$B[i_B]$ と l_B も同様である．

`corres`$(0, 0, \{\})$ の返戻値は true もしくは false であり，true を返すとき K と Y が対応付けられたと判定する．

このアルゴリズムは，仮名漢字混じり文 K に含まれる頻度 2 以上のすべての文字が仮名文 Y に含まれる文字と対応づけられるか否かを単純な全探索によって調べるものである．

両方の入力文を 1 文字ずつ読み進めていき，現在読んでいる K に含まれる頻度 2 以上の文字がどの仮名とも対応していなければ，現在読んでいる仮名文字との対応を仮定した場合 (12 行目) と，仮定せずに Y だけを読み進めた場合 (13 行目) の 2 つの場合を調べる．もし，現在読んでいる K に含まれる頻度 2 以上の文字が，すでにいずれかの仮名文字と対応しているのであれば，現在読んでいる仮名文字が対応するものであるか否かを調べる (6 行目)．対応するものであれば，K と Y の両方を読み進める場合 (7 行目) と，Y だけを読み進めた場合 (8 行目) の 2 つの場合を調べる．もし現在読んでいる文字同士の対応が仮定されていないならば，対応する仮名が登場するまで読み進めるために，Y だけを読み進める (10 行目)．

もし，K に含まれるいずれかの頻度 2 以上の文字に対応する仮名が Y に含まれないとすると，K を読み終える前に Y を最後まで読み終えることになる．このときは対応していないと判定し，false を返す (2 行目)．

こうして K を最後まで読んだとき，K に含まれるすべての頻度 2 以上の文字は Y に含まれる文字と対応することが分かる．このとき K と Y が対応していると判定し，true を返す (1 行目)．

```
corres(i_K, i_Y, H) :=
 1: if i_K ≥ l_K then return true endif;
 2: if i_Y ≥ l_Y then return false endif;
 3: if K[i_K] の頻度が 1 以下 then return corres(i_K + 1, i_Y, H) endif;
 4: if Y[i_Y] の頻度が 1 以下 then return corres(i_K, i_Y + 1, H) endif;
 5: if (K[i_K], X) ∈ H である X が存在する then
 6:     if X = Y[i_Y] then
 7:         if corres(i_K + 1, i_Y + 1, H) then return true endif;
 8:         return corres(i_K, i_Y + 1, H)
 9:     endif;
10:     return corres(i_K, i_Y + 1, H)
11: else
12:     if corres(i_K + 1, i_Y + 1, H ∪ {(K[i_K], Y[i_Y])}) then return true endif;
13:     return corres(i_K, i_Y + 1, H})
14: endif
```

図 2 対応付けを判定するアルゴリズムの疑似コード

なお，読んだ文字の頻度が 1 以下である場合はその文字を読み飛ばしている (3〜4 行目)．

このアルゴリズムは `corres` の再帰呼び出しの形をしているが，呼び出しのたびに i_K と i_Y の和が単調に増加する．それらが l_K または l_Y よりも大きくなったところで停止するため，必ず停止する．

K に含まれる頻度 2 以上の文字の種類数を n, Y に含まれる頻度 2 以上の文字の個数を m とする．K 中に頻度 2 以上の文字が新たに登場するごとに，現在読んでいる仮名と対応する場合としない場合の 2 つの場合に分けており，また，すでに対応している文字については，以降に出現する同じ仮名のどれと対応するかを全て調べている．したがって，計算量は $O(m2^n)$ となる．

Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

4. 実験結果

数学の読みもの[5] から無作為に取り出した 21 文字から 41 文字の長さの仮名漢字混じり文 20 文について，そのそれぞれを片仮名で表記した 20 文を用意した．それらの全組合せ $20 \times 20 = 400$ 通りについて，対応付けが見つかるか否かを判定した．実験の結果，20 通りの仮名漢字混じり文とその仮名表記である仮名文の組合せについては，すべて対応があると正しく判定された．また，対応していない組合せについて，対応していないと正しく判定したものが 211 通りであった．

一方，対応していない組合せであるにもかかわらず対応していると誤判定されたものが 169 通りあった．この 169 通りの中には，次のような例がある．ここで，上段が入力された仮名漢字混じり文，下段が仮名文であり，その間の線は対応を示している．

良い問題は、謎めいていて面白いものである。

コレハタダシイカモシレナイシ，タダシクナイカモシレナイ．

この例で見つかった文字同士の対応付けは次の通りである．

仮名漢字文に出現する文字	い	て
仮名文に出現する文字	シ	ナ

この例では，仮名漢字混じり文に出現する頻度 2 以上の文字が 2 種類しかない．このように，頻度 2 以上の文字が少ない入力については，対応しない組合せについて誤って対応すると判定する傾向にある．ここまでの実験で用いた 20 文を仮名漢字混じり文に出現する頻度 2 以上の文字の種類数で分けると，1,4,5,8 種類のもの各 1 文, 2 種類のもの各 6 文，3 種類のもの各 3 文，6 種類のもの各 2 文，7 種類のもの各 5 文であった．

そこで，仮名漢字混じり文に出現する頻度 2 以上の文字の種類数ごとに入力を分け，それぞれ 10 文ずつを用いて実験した．その結果を表 1 に示す．ここで例えば 2 種類以上とあるものは，2〜8 種類を含むものをほぼ同数ずつ含む 10 文をそれぞれ無作為に選んでいる．この結果をみると，頻度 2 以上の文字の種類が多いものについては誤判定の回数が少ない傾向にあることが分かる．

さらに，頻度 2 以上の文字を比較的多く含むにもかかわらず誤判定する例としては，以下のようなものが挙げられる．

表 1 仮名漢字混じり文に含まれる頻度 2 以上の文字数と誤判定された組合せの数．仮名漢字混じり文 10 文とそのそれぞれを片仮名で表記した 10 文を用い，全組合せ $10 \times 10 = 100$ 通りについて，対応づけが見つかるか否かを判定した．正しく対応しているのは 10 通りであり，対応があるのに無いと判定した回数は 0 であった．この表では，対応しない文について対応があると誤判定された回数を示している．

仮名漢字混じり文に含まれる頻度 2 以上の文字数	誤判定された組合せの数
2 種類以上	36
3 種類以上	39
4 種類以上	24
5 種類以上	8
6 種類以上	14
7 種類以上	6

しかし、そうした経験がないのであれば、

サイショノココロミガセイコウスルトハカギラナイシ，

これは練習ではなく問題となる。

ナンジッカイトシッパイスルコトモアルダロウ．

この例で見つかった文字同士の対応付けは次の通りである．

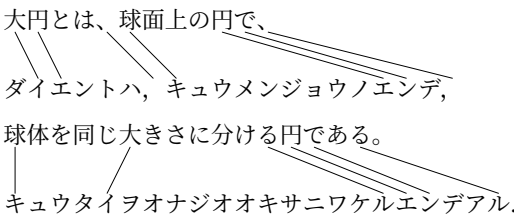
仮名漢字文に出現する文字	し	、	な	で	れ	は
仮名文に出現する文字	コ	イ	ル	ト	カ	イ

この例が対応付けられてしまうのは，仮名漢字混じり文と仮名文のどちらにおいても，頻度 2 以上の文字の出現頻度が比較的多いことが理由と考えられる．

これまでに報告した実験では，仮名漢字混じり文とその読みである仮名文については，すべて対応していると判定できている．用いたアルゴリズムが全探索をしているので，当然の結果とも言える．しかし，意味的には対応している組合せであるにもかかわらず，対応していると判定できない場合もある．例えば,「これは、三色のうちのどの色を選んでもうまくいく。」という仮名漢字混じり文を入力すると,「色」という漢字の読みが「ショク」と「イロ」の 2 通りあるために，この文の読みである仮名文と対応付けることができなかった．一つの文字が複数の読みを持ち，その両方の読みが入力文の中に登場するような例では対応付けることができないと考えられる．

Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

さらに，対応していると判定できたものの中には，上述のような一つの文字が複数の読みを持つ問題があるものの，たまたま成功している例も含まれている．例えば,「大円とは、球面上の円で、球体を同じ大きさに分ける円である。」という仮名漢字混じり文では,「大」の読みが「ダイ」と「オオ」の 2 通りがあるが，この文を片仮名で表した文と対応していると判定された．その対応付けは次に示す通りである．



概ね正しい対応付けになっているが,「大」を「イ」と対応付けており，たまたま「キュウタイ」の「イ」が「大きさ」の前に出現したために対応付けられると判定されていることが分かる．

5. 議　　論

実験から，一つの文字に複数の読みがある場合には，正しく判定できないことがあると分かった．しかし，一つの文字に複数の読みがあると仮定すると，対応していない組み合わせについても，対応があると誤判定する可能性が高くなると考えられる．一つの文の中に登場する複数の読みを持つ文字は余り多くはなく，本論文で実験した中では高々 1 文字であるが，その個数は言語ごとのパラメータとなるように思われる．

また，本論文で用いた手法では，対応する文字同士の間の空き方については考慮していない．そのため，不自然な空き方をする対応付けを見つけて，対応しない組合せを対応すると誤判定することも考えられる．例えば，仮名漢字混じり文において連続する 2 つの文字が，仮名文において遠く離れた 2 つの文字とそれぞれ対応するとしたら，不自然であるように思われる．しかし，これは一文字の読みがいくつのシラブルになるかに依存した問題であり，一文字の読みを構成するシラブルの個数はやはり言語に依存したパラメータであるように思われる．

本手法では，対象言語が日本語であることをできるだけ利用しないよう留意したつもりではあるが，一文の長さなどがパラメータとなっていることが考えられる．今後，対応付けが見つかった場合に，その確からしさを評価する方法について考える必要があるだろう．

なお，本論文で用いた手法は，K の (仮定された) 読みが Y に含まれるか否かを判定するものである．前節における実験で用いた仮名文は，仮名漢字混じり文の読みを表したもので，過不足が無い．しかし，仮名文の前後にいくつかの仮名文字を追加した場合でも，対応がある場合には対応すると正しく判定することができる．ただし，この場合には追加された文字を用いた対応が見つかることが有りうるので，意味的な対応が無いにもかかわらず対応すると誤判定する可能性が高くなるものと考えられる．

6. ま と め

本論文では，仮名漢字混じり文と片仮名文との対応付けを統語的な全探索によって見つける手法について考察した．結果として，対応する組合せのほとんどについて，正しく対応すると判定することができた．一方，対応しない組合せについて，対応すると誤判定することがある．誤判定の率が悪くても 3〜4 割であり，条件が良ければ 1 割程度に抑えられると考えられる．誤判定の原因には，仮名漢字混じり文に含まれる頻度 2 以上の文字の種類が少ないこと，およびそのような文字の頻度が高いことが挙げられると分かった．

本論文で用いた手法は，文を文字単位に分割する以上の言語に関する知識を用いないように留意した．その上で，文字列と，その文字列の読みである表意文字の列を対応付けることに，ある程度成功したと言える．

謝　　辞

本論文中の Rongorongo の図は，Cercle dÉtudes sur l'Île de Pâques et la Polynésie による画像を rongorongo.org[6] よりダウンロードして使わせて頂きました．

参 考 文 献

1) Benjamin Snyder, Regina Barzilay, Kevin Knight, A Statistical Model for Lost Language Decipherment", ACL 2010
2) 山口文彦, "Rongorongo 符号列とイースター島古謡音韻列の対応", 情報処理学会自然言語処理研究会報告, NL196SLP81-20, May 28, 2010
3) Alfred Mètraux, Ethnology of Easter Island, Bishop Museum Press, Bernice P. Bishop Museum Bulletin 160, Honolulu, 1940.
4) Steven Roger Fischer, RONGORONGO — the Easter Island script —, Clarendon Press, Oxford, Oxford Studies in Anthropological Linguistics, vol. 14, 1997.

Vol.2011-IFAT-101 No.6
Vol.2011-NL-200 No.6
2011/1/28

5) Paul Zeits 著，山口文彦，松崎公紀，三橋泉，松永多苗子，伊知地宏 訳，“エレガントな問題解決”, 日本オライリー, 2010
6) Rongorongo or the Hieroglyphs of the Easter Island Tablets, http://www.rongorongo.org/