# 万葉集を対象とした原文と読み下し文のアライメント

山田 祐実 †　大村 舞 †　鴻野 知暁 ‡　Kevin Duh †
小木曽 智信 ‡　松本 裕治 †
(† 奈良先端科学技術大学院大学　‡ 国立国語研究所)

# Word Alignment between Original Text and Its Reading in *Man'yōshū*

Yumi Yamada †, Mai Omura †, Tomoaki Kouno ‡, Kevin Duh †,
Toshinobu Ogiso ‡, Yuji Matsumoto †
(†Nara Institute of Science and Technology
‡ National Institute for Japanese Language and Linguistics)

**要旨**

国立国語研究所で開発中の『日本語歴史コーパス』（CHJ）では，『萬葉集』の歌の原文を古文の読み下し文（訓読文）と関連付けて扱えるようにする予定である．しかし，人手でこの作業を行うには量が膨大であるため，自動で行えることが望ましい．本稿では，IBM モデルを用いた原文と訓読文の自動対応付け（アライメント）を行った．IBM モデルによる自動アライメントの結果，概ね正しい対応結果が得られることが分かった．これらの不適切な対応関係を自動で修正するために，読み仮名の情報を用いる手法を用いて対応のない訓読文側の文字を原文側の文字へ対応させた．また，品詞の情報を用いる手法により不要な対応が付いている訓読文側の格助詞の対応を除去した．これにより，アライメントの改善が見られた．今後の課題として，誤った対応の付いたものを正解の対応へ修正することが必要であることが分かった．

## 1　はじめに

日本語の歴史研究において校訂作業が行われた資料を扱う場合，校訂作業前の原文でどのように表記されていたかという情報は重要である．何故なら多くの場合，校訂作業後の古文の読み下し文（訓読文）は原文で書かれた資料から一段離れたものとなるためである．特に，『萬葉集』のように原文と訓読文の表層形が大きく離れている場合，原文を参照することは必須となる．たとえば，小学館『日本国語大辞典』などの研究用の辞典では，『萬葉集』の原文と訓読文の情報が両方明記されている．

現在国立国語研究所で開発中の『日本語歴史コーパス』（CHJ）[小木曽ら 2013] では，訓読

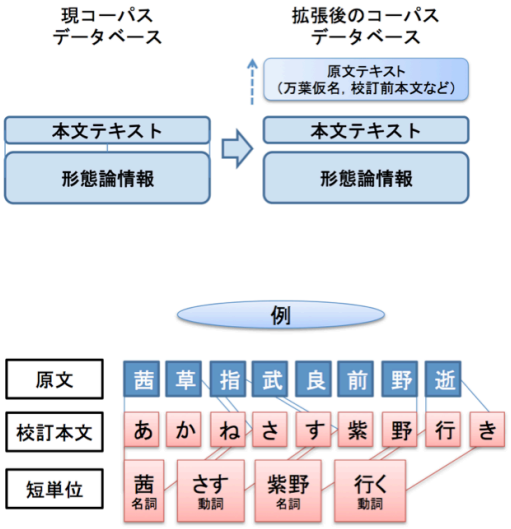


図 1: 日本語歴史コーパスの拡張

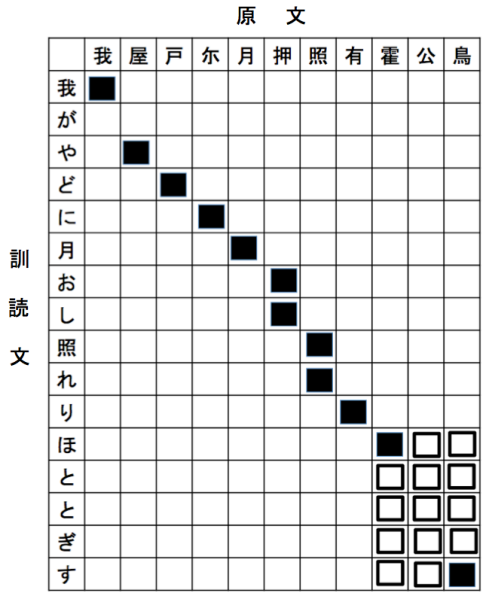


図 2: アライメントの例
黒四角と白抜きの四角がそれぞれ S アライメントと P アライメントを表す

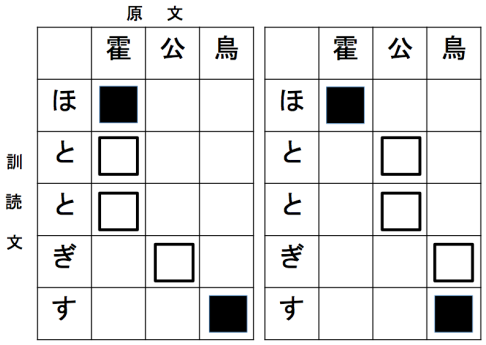


図 3: P アライメントによる評価

文を扱うことができる一方で，原文の情報は訓読文と関連付けて扱えていない．今後，図 1 のようにコーパス内に原文レイヤーを設け，『萬葉集』の原文の情報も同時に扱えるようにする予定である．しかし人手でアライメントを付与するには量が膨大であるため，自動で行えることが望ましい．そこで，本研究では訓読文と原文を文字単位で対応付け（アライメント）することによって，訓読文を原文と結びつけて扱えるようにする．

本稿では，『萬葉集』の歌の原文と古文による訓読文で語の対応付けを自動で行う手法を提案する．具体的には自動で語同士のアライメントを付与する手法である IBM モデル [Brown et al. 1993] を用いて語の対応付けを行う．

提案手法によって対応付けを行った結果，概ね正しい対応付けを行うことが分かった．しかし一方で，一部不適切な対応関係が見られた．本稿ではさらに，不適切な対応関係を自動で修正する手法について検討し追加実験を行った．

## 2　原文と読み下し文の対応付け

ここでは歌の原文と訓読文の対応付けについて概略を述べる．『萬葉集』における原文と訓読文の対応関係の例を図 2 に示す．図中に黒い四角で示した通り，基本的には原文一文字に対して訓読文一文字以上のアライメントを付与する．黒い四角で表したアライメントのように，対応に曖昧性がなく一つに決まるものを本稿では S アライメントとして表す．「我が」の「が」など，補読の助詞は対応する漢字がないものと見なす．「照れ-照」，「鳴き-鳴」のように，送り仮名の対応は原文の漢字一文字に含める．また，「ほととぎす-霍公鳥」，「とよもせ－響令」のように，文字同士の対応付けが正確には難しく，曖昧になるもの（熟字訓）も存在する．そのような曖昧な対応関係を本稿では P アライメントと表現する．図 2 の白抜きの四角のマスと黒い四角のマスの合計が P アライメントである．P アライメントを用いることで，図 3 の「ほととぎず-霍公鳥」の対応はいずれも正解であるとみなす．

## 3　IBM モデルを用いたアライメント（提案手法 1）

本章では，使用したコーパス及び IBM モデルを用いて語の対応付けを行う方法（提案手法 1）と，その結果について述べる．

### 3.1　使用したコーパス

『日本語歴史コーパス』の一部として収録予定の小学館『新編日本古典文学全集』の『萬葉集』（以下、CHJ 万葉集データ）を用いた．本研究では，CHJ 万葉集データの万葉仮名による原文と古文の訓読文を用いた．原文には，漢字の文字列に加え，書き下し文に戻すことができるように，レ点，上下点，一二点といった返り点がそれぞれの漢字に付けられている．

### 3.2　手法

ベースラインとして，自動アライメントの方法である IBM モデルを用いて実験を行った．IBM モデルは Brown ら (1993) が提案した機械翻訳のための手法である．IBM モデルは統計分布を基にして統計値を計算し自動的にアライメントを求める手法である．訓読文と原文のペアを入力として与えることで自動的にアライメント結果を得ることができる．

原文と訓読文は語順が違うため，語順の違いがアライメント結果に悪影響を与える可能性がある．そのため原文は IBM モデルに与える前に，コーパス中に記載されている返り点を基に語順を入れ替えておく．提案手法 1 では原文の語順を入れ替えた後に，IBM モデルに原文と訓読文を与えることで自動的にアライメント結果を得る．提案手法 1 のイメージを図 4 に示す．

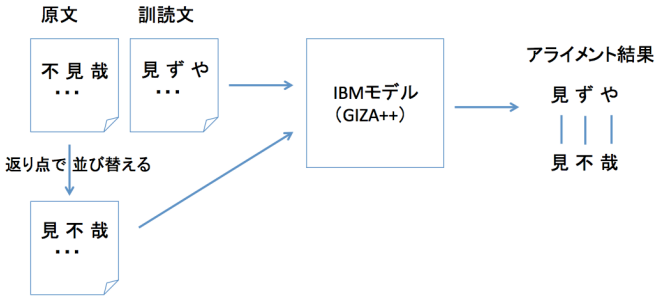


図 4: 提案手法 1 の流れ

表 1: 実験 1 評価値

| | マイクロ F 値 | 適合率 | 再現率 | 出力アライメント数 |
|---|---|---|---|---|
| 並び替えなし | 0.9478 | 0.9422 | 0.9522 | 1,384 |
| 並び替えあり | 0.9589 | 0.9524 | 0.9654 | 1,386 |

### 3.3 実験設定

テストデータとして，CHJ 万葉集データからランダムに 50 首選んだ．本稿では選んだ 50 首のテストデータを対象にして実験，評価を行った．このテストデータには，正解の S アライメントの数は 1 ,360 箇所，P アライメントの数は 1,407 箇所存在する．IBM モデルの実装として，GIZA++ v1.0.7[Gao et al. 2008] を用いた．GIZA++ のパラメータは，デフォルト値に設定した．

IBM モデルでは，計算量を減らすため，翻訳される元の文 (原言語) から翻訳後の言語 (目的言語) への対応は最大 1 単語までという仮定を置いて統計値を計算している．原文から訓読文へは一文字以上の対応が付けられるが，その逆は殆ど起こらないことが分かった．そこで本稿では，原言語を訓読文，目的言語を原文として実験を行っている．

### 3.4 評価方法

評価のために，ランダムに選んだ 50 首の歌に対して人手で正解データを作成した．単語アライメントの評価値として F 値を用いる．F 値は以下の式のように適合率と再現率のマクロ平均として計算される．適合率と再現率は S アライメントの数 $a_s$ と P アライメントの数 $a_p$ に基づいて求められる．適合率はアライメント出力全体数 $a$ に対する P アライメントの正解出力数 ($|a \cap a_p|$) の割合を表す．再現率は S アライメント出力全体数 $a_s$ に対する S アライメントの正解出力数 ($|a \cap a_s|$) の割合を表す．P アライメントと S アライメントを用いることで，アライメントの曖昧性を考慮した評価をすることになる．

$$\text{F 値} = \frac{2 \times \text{適合率} \times \text{再現率}}{\text{適合率} + \text{再現率}}, \quad \text{適合率} = \frac{|a \cap a_p|}{|a|}, \quad \text{再現率} = \frac{|a \cap a_s|}{|a_s|}$$

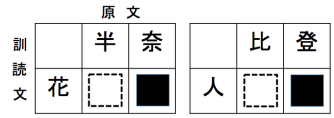


図 5: エラー 1
破線の四角は実際には対応が取れなかったアライメントを表す

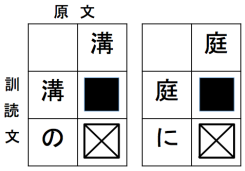


図 6: エラー 2
クロス記号の四角は誤って出力したアライメントを表す

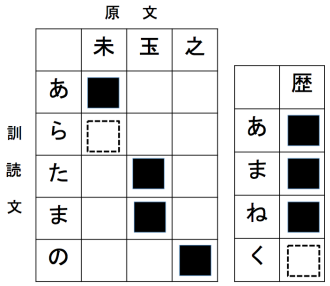


図 7: エラー 3

### 3.5　結果・考察（提案手法 1）

提案手法 1 の実験結果の評価値を表 1 に示す．比較のため，原文を並び替えずに IBM モデルを適応した手法 (並び替えなし) と原文を並び替えた後 IBM モデルを適応した手法 (並び替えあり) の結果を載せている．「並び替えあり」の方が，「並び替えなし」よりもアライメントの評価値が高いことが分かった．

ほとんどの場合で適切にアライメントができていることが分かったが，一部で不適切なアライメントが存在することが分かった．「並び替えあり」について，対応付けが正確にできなかったものは大きく分けて 3 種類に分類できた．一つ目のエラー（エラー 1）は，図 5 における「花—半奈」の「半」，「人—比登」の「比」のように，原文側から訓読文に対応するべき語が対応していないものである．全アライメント中，エラー 1 は 13 箇所あった．逆に，2 つ目のエラー（エラー 2）は，訓読文から原文へ対応がないにも関わらず，他の文字に誤った対応が付いているものである．エラー 2 は 40 箇所見つかった．図 6 に示したように，訓読文の補読語である「の」や「に」まで「溝」や「庭」に対応が付いてしまっている．3 つ目のエラー（エラー 3）は，図 7 のように，訓読文から原文へ対応するべき文字が誤った文字に対応している，もしくはどの文字にも対応していないものである．エラー 3 は 33 箇所あった．

4 節以降はエラー 1，エラー 2 について対処するための手法について説明する．エラー 1 を改善するために，読み仮名の情報を用いてアライメントを修正した．エラー 2 に対しては，品詞の情報を用いてアライメントの修正を行った．エラー 3 については，今回は対処方法を考案していないため，今後の課題とする．

## 4　読み仮名の情報を用いた手法（提案手法 2）

ここでは，3.5 節のエラー 1 で示した，読み仮名の情報を用いたアライメントの修正方法を提案し，実験結果について述べる．

### 4.1　手順（提案手法 2）

原文側から訓読文側へ対応すべき語の対応がないエラー 1 には読みの情報が有効であると考える．たとえば，図 5 で示したエラーの例を見ると，原文の「半奈」も訓読文の「花」も

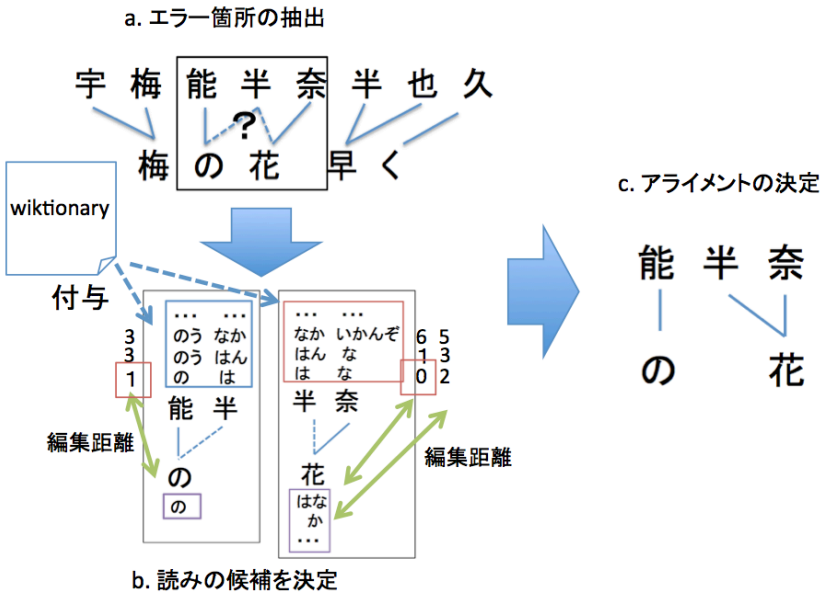


図 8: 読み仮名の情報を用いた手法

「はな」と読むことができる．「半」は「ハン」の他に「は」という読み方もできるためである．同様に，「比登」と「人」においても，どちらも「ひと」と読むことができる．このように，原文と訓読文に共通する読み仮名の情報を用いることで，訓読文の「花」に対応していない「半」も「はな」の一部であることが判断できる．そこで，エラー 1 に対しては読み仮名の情報を用いて対応付けの修正を行う．漢字に対応する読み仮名の情報は，ウェブ上で利用可能な Wiktionary[*1]から取得した．

エラー 1 を修正するためには，まず始めにエラーの箇所を自動で検出する必要がある．置き字（「焉」，「也」）の例外を除き，原文側の語から訓読文側の語へ必ず一文字以上の対応があることが分かっている．そこで，提案手法 1 の結果の中で原文側から訓読文へ一文字も対応を持たない原文の語があれば，そこにエラーがあるとみなした．この仮定に基づいて，対応語を持たない原文の語とその前後の語，及びそれにアライメントが付与されている訓読文の語を検出した．例えば図 8 の a では，原文側の「半」は訓読文側のいずれにも対応が存在していないため，「半」を修正対象として検出する．

次に，対応を持たない原文の文字がその前後一文字のどちらと単語を成すのかを決定する．これを決定することで，対応を持たない原文の文字を訓読文のどの文字へ対応付ければ良いかを決める．たとえば，図 8 の場合，「半」が前後の「能」と「奈」のどちらに付くかを判断することにより，訓読文側の「の」と「花」のどちらに対応付けられるかを決定する．つまり，「能半—の」と「半奈—花」のどちらの対応関係が適切かという問題になる．これを判断するために，各文字に付与した読み仮名の類似度をレーベンシュタイン距離 (以降編集距離) を用いて計算する．

編集距離は二つの文字列がどのくらい離れたものであるかを数値（コスト）を用いて表す指

[*1] Wiktionary：漢字索引 音訓 https://ja.wiktionary.org/wiki/Wiktionary:漢字索引_音訓

標である．二つの文字列のうち片方の文字列について，もう一方の文字列と等しくなるまで文字の挿入・削除・置換のいずれかの操作を繰り返す．一度の操作のコストは，挿入・削除では 1 点，置換は 2 点とし，コストの合計を二つの文字列間の編集距離とする．原文と訓読文の読みの候補の全ての組み合わせについて編集距離を計算する．

図 8 の b に示したように，Wiktionary を用いて原文と訓読文の漢字に全ての読みの候補を付与した．読みの全ての組み合わせについて編集距離を計算し，最小のものを前後それぞれの読みの候補として決定する．最後に前後について，最も小さいコストをとる対応関係を選ぶ．図 8 の b において，「の は」と「の」の編集距離と「はな」と「はな」の編集距離を比較すると 1 ＞ 0 となるので，編集距離のより小さい後者を対応関係として選ぶ．以上の手順を踏まえることで，図 8 の c のように「半 奈」と「花」が適切な対応として選ばれる[*2]．このように読みの情報を用いることで対応が取れなかった原文側の漢字のアライメントを得ることができる．

### 4.2 結果・考察（提案手法 2）

3.3 節で述べたランダムに選んだ 50 首について，原文側で対応がとれていなかった例は 13 個存在した．この対応が取れていなかった漢字に対して読みの編集距離を用いて適切なアライメントを付与した結果を表 2 に示す．二重山括弧で示したものが対応の取れていなかった漢字であり，山括弧で示したものが上述の方法で選ばれた適切なアライメントである．表 2 に示したように，13 個あるエラーのうち 11 個は改善された．改善されなかったものは，「真田葛―まくず」と「古保志―こほし」の 2 つである．この 2 つは前後の編集距離のスコアが同点になったため，この方法では適切な対応を選択することができなかった．「真田葛―まくず」のように，文字ごとに読みを対応させることが難しい熟字訓の場合，読み仮名を手掛かりに対応を取ることができないことがある．また，「古保志―こほし」は，訓読文の「恋し（こほし）」に対応する．「古保―恋」の対応が取れなかった理由は，現代語では「恋し」を「こほし」と読まないためと考えられる．Wiktionary の読み仮名の情報にも「レン，こい，こ」しか存在しない．このように，歴史的仮名遣いに応じた漢字によって原文と訓読文が対応している場合にも，この手法では限界がある．

## 5 品詞の情報を用いた手法（提案手法 3）

次に，3.5 節のエラー 2 で示した，原文へ対応のないはずの補読語が他の文字に対応付いてしまう誤りを修正するための手法について説明し，実験結果について述べる．

### 5.1 手順（提案手法 3）

修正するエラーの対象は，訓読文側の格助詞に原文側へ対応語があるとき，その原文の対応語が訓読文側に複数の対応語を持つ場合である．たとえば 3.5 節の図 6 に示した「溝の」の「の」や「庭に」の「に」などのように，訓読文側の格助詞から原文へ不適切な対応のある場合

[*2] スコアが同点の場合はどちらが適切か選択できないためアライメントを追加しない．

表 2: 読みの編集距離を用いた結果

| | エラー | （前）編集距離のスコア | （後）編集距離のスコア |
|---|---|---|---|
| 1 | 能《半》奈 | の，のは　1 | 〈はな，はな　0〉 |
| 2 | 将《尓》焼 | む，すすむそ　3 | 〈をや，そや　2〉 |
| 3 | 真《田》葛 | ま，まや　1 | つずら，やつずら　1 |
| 4 | 比《登》能 | 〈ひと，ひと　0〉 | の，みの　1 |
| 5 | 伊《波》毛 | 〈いわ，いわ　0〉 | も，わも　1 |
| 6 | 安《伎》也 | 〈あき，あき　0〉 | やま，ぎやま　1 |
| 7 | 許呂《母》弖 | 〈ころも，ころも　0〉 | て，もて　1 |
| 8 | 伊《麻》佐 | 〈いま，いま　0〉 | さか，まさか　1 |
| 9 | 毛ゝ《等》利 | も，もゆすりら　4 | 〈とり，とり　0〉 |
| 10 | 古《保》志 | こ，こほ　1 | し，ほし　1 |
| 11 | 岐《多》流 | 〈きた，きた　0〉 | る，なる　1 |
| 12 | 知《可》豆 | 〈ちか，ちか　0〉 | づ，べづ　1 |
| 13 | 芸《可》久 | ぎ，ぎべ　1 | 〈かく，かく　0〉 |

が多いためである．このような格助詞の不適切な対応を修正するためには，はじめに訓読文側の格助詞を見分ける必要がある．

まず訓読文側で品詞の情報を得るために，MeCab v0.98 [Kudo et al. 2004] と中古和文 UniDic v1.4 [小木曽 2013] を用いて品詞の情報を付与した．その後，訓読文の文字の中で「格助詞」と判定された文字に対応しているアライメントを検出する (アライメント 1 とする)．その格助詞からアライメント 1 で対応している原文側のアライメントについて，複数のアライメントが付与されているか確認する．もし複数のアライメントがあった場合，アライメント 1 を除く．例えば，図 6 の「庭に―庭」の対応関係の場合，訓読文側の「に」は格助詞であり，原文側の「庭」に対応が付いている（アライメント 1)．次に，原文側の「庭」を見ると，訓読文側の「庭」にもアライメントが付与されていることが分かるため，アライメント 1 を除く．

また，実験設定として，提案手法 2 の読み仮名の情報を用いた対応付けを行った後に，提案手法 3 の品詞の情報を用いた対応付けを行った．逆に，提案手法 3 の後に提案手法 2 を行うことも試みた．

### 5.2　結果・考察（提案手法 3)

この節では，提案手法 3 の結果について述べる．また，提案手法 2 と 3 の二つの追加実験を行った後に適切な対応の取れていないエラーについて考察を行う．

まず提案手法 3 で品詞の情報を用いた結果，改善されたエラーの例を図 9 に示す．補読語の格助詞が原文の漢字に対応付いていた不適切な対応がなくなった．

読み仮名の情報を用いたものと，品詞の情報を用いたもの，その両方を用いたものの評価値の比較を表 3 に示す．F 値は読み仮名の情報を用いた後に品詞の情報を用いたものが最も高

表 3: 提案手法の比較
「並び替えあり」は返り点によって原文の漢字の文字列を並び替えたもの．「読みの情報のみ」は提案手法２のみを適用したもの．「品詞の情報のみ」は提案手法３のみを適用したもの．「品詞 → 読み」は提案手法３の後に提案手法２を適用したもの．「読み → 品詞」は提案手法２の後に提案手法３を適用したもの．

| | マイクロ F 値 | 適合率 | 再現率 | 出力アライメント数 |
|---|---|---|---|---|
| 並び替えあり | 0.9589 | 0.9524 | 0.9654 | 1,386 |
| 読みの情報のみ | 0.9630 | 0.9528 | 0.9735 | 1,397 |
| 品詞の情報のみ | 0.9665 | 0.9712 | 0.9618 | 1,354 |
| 品詞 → 読み | 0.9688 | 0.9706 | 0.9669 | 1,362 |
| 読み → 品詞 | 0.9706 | 0.9714 | 0.9699 | 1,366 |

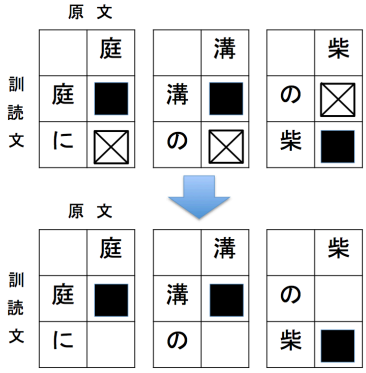


図 9: 提案手法３による改善例（上が改善前，下が改善後）

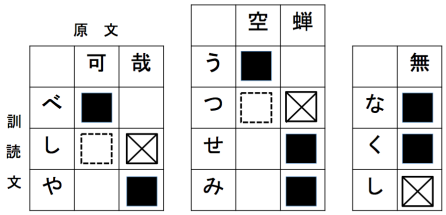


図 10: 今後改善の必要なアライメントの例

い．読みの情報のみを用いた場合，返り点による並び替えのみを行った結果よりも出力の S アライメント・P アライメントの数は 11 箇所増えた．これは，4.2 節で述べたように，原文側へ対応を持たない例の 13 箇所のうち 11 箇所が改善されたためである．

次に，品詞の情報を用いた場合と並び替えのみの結果を比較する．出力アライメント数は 32 箇所減り，S アライメント・P アライメントの数は 5 箇所減った．32 箇所の格助詞を除いたが，そのうちの 5 箇所は誤って除いてしまったことを意味している．格助詞の中でも除いてはいけない対応が 5 箇所存在していたためである．しかし，提案手法 2 と組み合わせることにより，対応が修正できたため，あまり最終的な結果に影響がでなかった．

また，以上の試みの後に，アライメントが改善されていない部分について分析を行った．エラーの例を図 10 に示す．これらの例は 3.5 節で触れたエラー 3 に分類でき，訓読文から原文へ誤った対応を持つものである．図 10 の「べしや-可哉」と「空蝉-うつせみ」は，対応する文字が不適切であるものを示す．「べしや-可哉」の例では，「可-べし」「哉-や」と対応すべきところが，「可-べ」「哉-しや」と対応付いている．「無-なく」は，助詞以外の語で不適切な対応を持つものである．これらの対応関係は複数の歌で出現する語であることから，IBM モデルに対して頻度が高い対応のペアを再度辞書として追加して制約を与えることにより，アライメントを改善する方法が考えられる．

## 6 まとめ

本稿では，IBM モデルの実装である GIZA++ を用いて『萬葉集』の原文と訓読文の文字単位での対応付けを行った．GIZA++ を用いる際，事前に返り点で原文の並び替えを行った方が並び替えを行わないものよりも評価値が高くなった．次に，原文側から訓読文側へ対応する語が対応付いていないエラーに対し，原文と訓読文それぞれの文字に読み仮名を割り当てて類似度を測ることにより，不適切な対応付けの修正を行った．最後に，訓読文側に品詞の情報を用いて格助詞の誤った対応付けを修正した．それぞれの修正により，アライメントは改善されたが，まだ不適切な対応関係が残っていることが分かった．今後は，対応関係に誤りのある文字を正しい対応関係へ修正する方法について考える必要がある．

## 参考文献

[Brown et al. 1993] Brown, Peter F., Vincent J. Della Pietra, and Stephen A. Della Pietra et al. (1993). The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational linguistics* Vol. 19.2, pp.263-311

[Gao et al. 2008] Gao, Qin and Stephan Vogel (2008). Parallel Implementations of Word Alignment Tool. In *Proceedings of Software Engineering, Testing, and Quality Assurance for Natural Language Processing* (ACL2008), pp.49-57

[Kudo et al. 2004] Kudo, Taku, Kaoru Yamamoto, and Yuji Matsumoto (2004). Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of 2004 Conference on Empirical Methods in Natural Language Processing* (EMNLP2004), pp. 230-237

[小木曽 2013] 小木曽智信 (2013)「中古仮名文学作品の形態素解析」日本語の研究，9:4，pp.49-62

[小木曽ら 2013] 小木曽智信、須永哲矢、冨士池優美、他 (2013)「『日本語歴史コーパス 平安時代編』先行公開版について」第 3 回 コーパス日本語学ワークショップ予稿集，pp.269-276