Original–Transcribed Text Alignment for Man'yōsyū Written by Old Japanese Language v1.1

Teruaki Oka† **Tomoaki Kono**†
† National Institute for Japanese Language and Linguistics

Abstract

We are constructing an annotated diachronic corpora of the Japanese language. In part of this work, we construct a corpus of Man'yōsyū, which is an old Japanese poetry anthology. In this paper, we describe how to align the transcribed text and its original text semiautomatically to be able to cross-reference them in our Man'yōsyū corpus. Although we align the original characters to the transcribed words manually, we preliminarily align the transcribed and original characters by using an unsupervised automatic alignment technique of statistical machine translation to alleviate the work. We found that automatic alignment achieves an F1-measure of 0.83; thus, each poem has 1–2 alignment errors. However, finding these errors and modifying them are less work-intensive and more efficient than fully manual annotation. The alignment probabilities can be utilized in this modification. Moreover, we found that we can locate the uncertain transcriptions in our corpus and compare them to other transcriptions, by using the alignment probabilities.

1 Introduction

National Institute for Japanese Language and Linguistics (NINJAL) is constructing an annotated diachronic corpora of the Japanese language.[1] As part of this work, we are constructing a corpus of **Man'yōsyū** (萬葉集, "Collection of myriad leaves"), which is an old Japanese poetry anthology complied about 8th–9th century AD. Since it is a worldwide very rare example of literature written more than 1,000 years ago, Man'yōsyū is an major source for those who study old Japanese language (**OJ**). This anthology is composed of 20 volumes and contains more than 4,500 poems.[2] Our corpus is based on the transcribed version of the text from original text (see Figure 1), and a large amount of information is annotated semiautomatically by utilizing NLP tools. For example, word boundaries, part-of-speech (POS) tags, pronunciations, cross-references to original characters, and so on are included in this information. Table1 shows the statistics of our Man'yōsyū corpus.

In this paper, we describe how to align the transcribed text and its original text semiautomatically to be able to cross-reference them in our Man'yōsyū corpus. This is because researchers of OJ frequently reference and consult the original texts. Eventually, we align the original characters to the transcribed words manually. However, to alleviate this work, we preliminarily align the transcribed and original characters by using an unsupervised automatic alignment technique of statistical machine translation and then modify the mistakes manually with less work.

2 Transcription

Most OJ researchers use some type of transcribed version of old Japanese texts. Therefore, we also employed the transcribed version of Man'yōsyū (Kojima et al., 1994) as the base text of our corpus. This

[1] http://pj.ninjal.ac.jp/corpus_center/chj/overview-en.html

[2] Although Man'yōsyū consists of several volumes (books), we deem the anthology to be one text and treat it as singular for clarification in this paper.

Table 1: The statistics of our Man'yōsyū corpus.

Number of poems	4,516
Number of syllabic units	29,489
Total number of words (transcribed)	101,313
Total number of transcribed characters	148,352
Total number of original characters	128,063

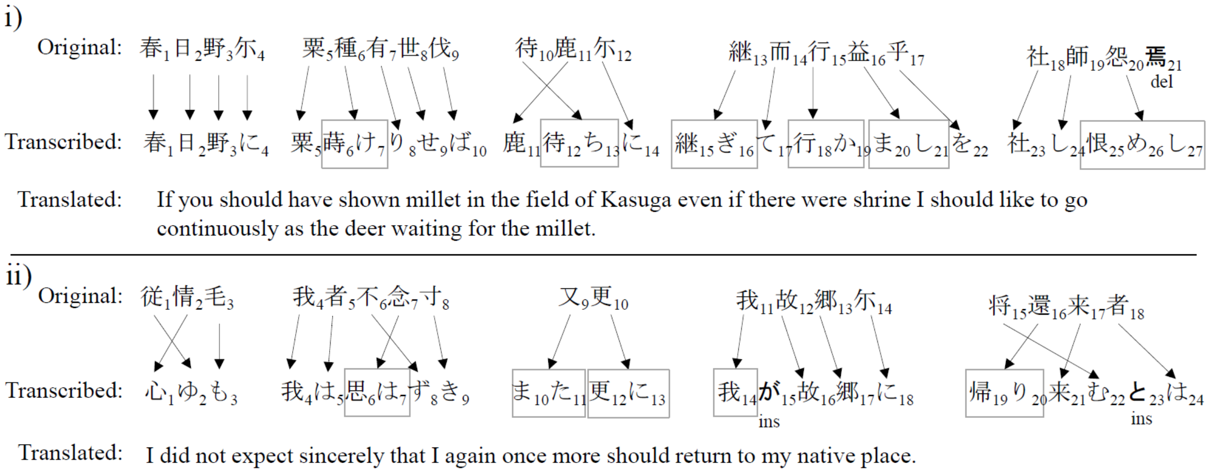

i)
Original: 春1日2野3尓4 粟5種6有7世8伐9 待10鹿11尓12 継13而14行15益16乎17 社18師19怨20焉21
del
Transcribed: 春1日2野3に4 粟5蒔6け7り8せ9ば10 鹿11待12ち13に14 継15ぎ16て17行18か19ま20し21を22 社23し24恨25め26し27
Translated: If you should have shown millet in the field of Kasuga even if there were shrine I should like to go continuously as the deer waiting for the millet.
ii)
Original: 従1情2毛3 我4者5不6念7寸8 又9更10 我11故12郷13尓14 将15還16来17者18
Transcribed: 心1ゆ2も3 我4は5思6は7ず8き9 ま10た11更12に13 我14が15故16郷17に18 帰19り20来21む22と23は24
ins
ins
Translated: I did not expect sincerely that I again once more should return to my native place.

Figure 1: Examples of transcribed poems from Man'yōsyū (Kojima et al., 1994). For clarification, each character was indexed with subscript Arabic numerals. A bold character in an original poem that was deleted in the transcribed poem is indicated by "del," and bold characters in the transcribed poem that were inserted are indicated by "ins." The translated poems were cited from (Pierson, 1929–1963).

text was provided by Syogakukan, a major Japanese publishing company. The provided text is marked up using XML to digitally replicate the paper books and was already annotated with some information (e.g., page number; poem number; ruby, which is explained further in Section 4.1; and original text). This text is a transcription of a reading of the original text into a mixture of *kanji* and *kana* characters used in the writing of the modern Japanese language. The original Man'yōsyū text is written in OJ with only kanji characters, which are used in two different ways: logographically and phonographically (the latter use is known as *man'yōgana*).

In transcription works, the phonographic characters are replaced with kana characters,[3] and some logographical ones are also replaced with more suitable kanji characters or kana characters. Since several kanji characters have been used in the modern Japanese language, they are sometimes not replaced. In addition, since the original poems were sometimes written in the writing style of the Chinese language, the transcribed texts contain character-order replacements, deletions, and insertions with respect to the original poems, as in Chinese–Japanese translation (see Figure 1).

3 Related work

Techniques for automatic alignment between electronic parallel texts are mainly used in the field of statistical machine translation, and many NLP tools are available. The most popular alignment tool is GIZA++ (Och and Ney, 2003), which can align one source token (e.g., a word) to some number of target tokens (1-to-n alignment) in each type of unit (e.g., a sentence) in a parallel corpus by using IBM models (Brown et al., 2003) and HMM model (Brown et al., 2001). GIZA++ allows token-order replacements, deletions, and insertions among a source/target unit pair.

These techniques are used not only in the fields of machine translation, but also digital humanities. For example, (Moon and Baldridge, 2007) used them to induce POS taggers for Middle English text.

[3] If the kana character(s) can be additionally replaced with more easy-to-read logographic kanji character(s), the kana character(s) are replaced with the kanji characters (e.g., "波 奈" are replaced with "は な", and additionally replaced with "花.").

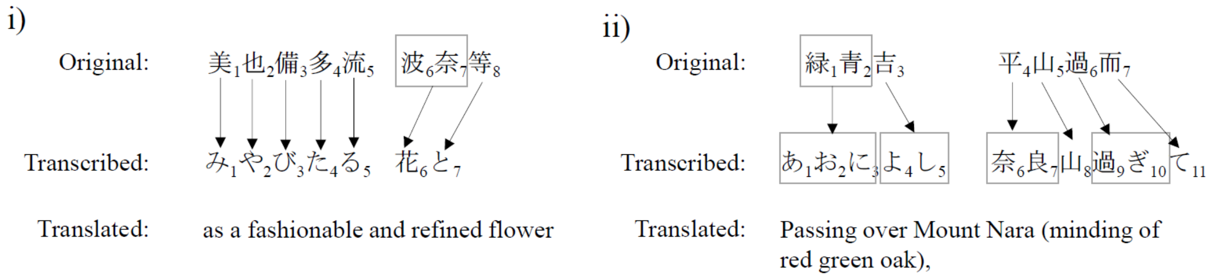

i)
Original:
美1也2備3多4流5 波6奈7等8
Transcribed:
み1や2び3た4る5 花6と7
Translated:
as a fashionable and refined flower
ii)
Original:
緑1青2吉3 平4山5過6而7
Transcribed:
あ1お2に3よ4し5 奈6良7山8過9ぎ10て11
Translated:
Passing over Mount Nara (minding of red green oak),

Figure 2: Examples of many-to-1 and many-to-many transcription (alignment) in Man'yōsyū(Kojima et al., 1994). The translated texts were cited from (Pierson, 1929–1963).

They aligned Modern English and Middle English bibles and then projected the POS tags from words in the Modern English bible to words in the Middle English bible for use as a training corpus for Middle English bigram POS taggers. This projection approach was proposed in (Yarowsky and Ngai, 2001) and has been used to create POS taggers or parsers for low-resource languages or domains, and so on (Drábek and Yarowsky, 2005; Ozdowska, 2006).

We also use GIZA++ because there are character-order replacements, insertions, and deletions between the transcribed and original text. Since (Moon and Baldridge, 2007) mainly intended to create POS taggers, they did not evaluate their autoalignment performance. However, our objective is annotating alignments themselves; thus, we need to evaluate and attempt to improve our automatic alignment performance. In addition, (Moon and Baldridge, 2007) treated texts written in English, which is a word-segmented language, and employed a word as an alignment token. In contrast, Japanese language does not use a space between words; moreover, in our Man'yōsyū corpus, although the transcribed text is already word-segmented because of our policy for creating the corpus, the original text is not segmented because it is merely additional information. Therefore, we employ a character as an alignment token; thus, our automatic alignment is characters-to-characters.

4 Original–transcribed character alignment

We start with the computerized parallel texts of Man'yōsyū (Kojima et al., 1994), in which each transcribed poem is associated with its original poem. In addition, both forms are spaced by caesura space marks. Therefore, we employ each syllabic unit[4], spaced by caesura space marks, as our alignment unit and each character as our alignment token. Since the number of (transcribed) words in the Man'yōsyū corpus is not very large, we can avoid the data sparseness problem as an advantage of employing characters-to-characters alignment. Although (Moon and Baldridge, 2007) employed 1-to-n alignment, our transcriptions have n-to-1 and m-to-n alignment pairs (see Figure 2). However, these are rare cases. Most alignment pairs are 1-to-n, as in Figure 1; thus, we also employ the 1-to-n (one original character to several transcribed characters) alignment of GIZA++[5] and utilize post-processing to cope with the n-to-1 and m-to-n alignments via rules. In addition, our task only has a character sequence-to-a character sequence alignments as a restriction.[6] This is because, most minimum m-to-n alignment pairs between the original and transcribed texts follow this restriction in the transcription works, as in Figures 1 and 2, and eventually, we want to align one original character sequence to one transcribed word (see Table 2).

4.1 Additional data

To improve alignment performance, we use original-transcribed (**or-tr**) unit pairs and other parallel units. First, we use ruby tags. A "ruby" is a small kana character (or characters) attached to the (mainly kanji) character (or characters) in the body text, generally to represent the pronunciation of the body character(s). Man'yōsyū (Kojima et al., 1994) also has ruby characters in both the transcribed and original

[4] Syllabic unit is the equivalent of a "line" in an English poem.

[5] We used *mkcls* for using IBM-Model 4 and HMM Model.

[6] We allow that this "sequence" consists of only one character.

Table 2: Examples of alignment one original character sequence to one transcribed word our Man'yōsyū corpus cross references. Upper example is the case of Figure 1 ii) and under example is the case of Figure 2 ii).

Word	POS tag	Original caharcters
心$_1$	noun	情$_2$
ゆ$_2$	particle	従$_1$
も$_3$	particle	毛$_3$
我$_4$	pronoun	我$_4$
は$_5$	particle	者$_5$
思$_6$は$_7$	verb	念$_7$
ず$_8$	auxiliary verb	不$_6$
き$_9$	auxiliary verb	寸$_8$
ま$_{10}$た$_{11}$	adverb	又$_9$
更$_{12}$に$_{13}$	adverb	更$_{10}$
我$_{14}$	pronoun	我$_{11}$
が$_{15}$	particle	NULL
故$_{16}$郷$_{17}$	noun	故$_{12}$郷$_{13}$
に$_{18}$	particle	尓$_{14}$
帰$_{19}$り$_{20}$	verb	還$_{16}$
来$_{21}$	verb	来$_{17}$
む$_{22}$	auxiliary verb	将$_{15}$
と$_{23}$	particle	NULL
は$_{24}$	particle	者$_{18}$
あ$_1$を$_2$に$_3$よ$_4$し$_5$	noun	緑$_1$青$_2$吉$_3$
奈$_6$良$_7$	noun	平$_4$
山$_8$	noun	山$_5$
過$_9$ぎ$_{10}$	verb	過$_6$
て$_{11}$	particle	而$_7$

texts and were computerized with ruby tags (see Figure 3). We use rt tags (body text) as transcribed units and rb tags (ruby text) as original units accessorily, because these tag annotations (computerizations) are not trusted and not every kanji character has a ruby. We call units from the ruby tags in the transcribed text **tr-ruby** and those in the original text **or-ruby**. To avoid data sparseness, only in *mono-ruby*[7] cases, we replace the kanji characters in the rt tags with the kana characters in the rb tags in the transcription text at or-tr as a preprocessing step.[8] These are replaced with rt characters after GIZA++ alignment step. These steps create m-to-1 alignments from 1-to-n alignments of GIZA++ outputs (see Figure 4).

Second, original units include characters that have been used in the modern Japanese language since the OJ, and these characters are sometimes not replaced in transcription work. Therefore, to successfully align these characters, we also use pairs consisting of a character and itself (e.g., “粟–粟”, “種–種”), called **character-self**. Table 3 shows some examples of simplified input data for GIZA++, and Table 4 shows the numbers of units, source characters, and target characters.

4.2 Post-processing rules

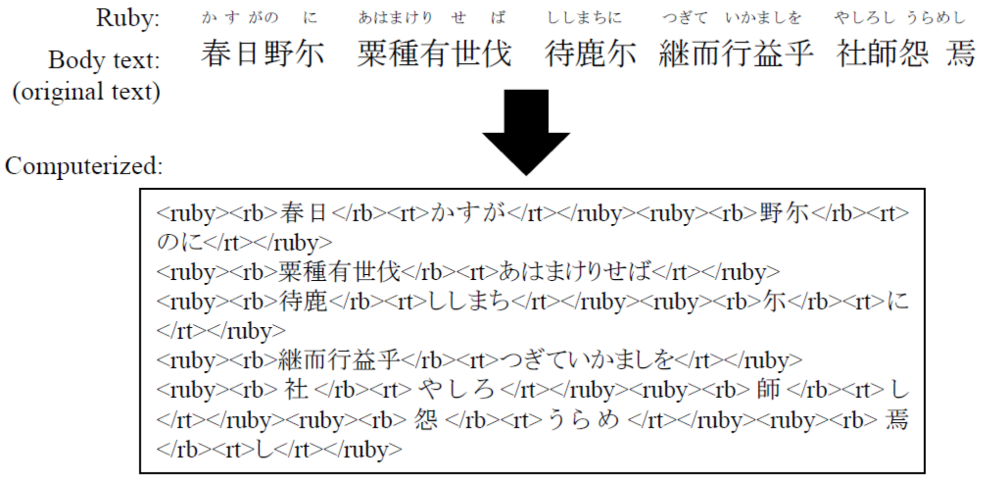
For post-processing the GIZA++ output, we apply the following rules in order (see Figure 5 a)-g)). We note that since the transcribed text has already been word-segmented and POS-tagged, we can refer to the POS tags of all subscribed characters.

a) **Rule 1. Interpolating for alignments 1:** If a character in the transcribed unit is NULL-aligned and the POS tag of the character is not a particle, we assign it with the same character alignment of its leftmost character that is not NULL-aligned in the same unit.

b) **Rule 2. Interpolating for alignments 2:** If a lead character(s) in the transcribed unit is NULL-aligned, we assign it (them) with the same character alignment of its (their) rightmost character that is not NULL-aligned in the same unit.

[7] This is a particular ruby that is attached to only one character in the body text.

[8] Actually, we also replace all *odoriji* characters (〻) that represent iteration of the previous character(s), to the corresponding previous character(s) when we use GIZA++.


Ruby:
かすがの　に　あはまけり　せ　ば　ししまちに　つぎて　いかましを　やしろし　うらめし
Body text:
(original text)
春日野尓　粟種有世伐　待鹿尓　継而行益乎　社師怨　焉
Computerized:

```
<ruby><rb>春日</rb><rt>かすが</rt></ruby><ruby><rb>野尓</rb><rt>
のに</rt></ruby>
<ruby><rb>粟種有世伐</rb><rt>あはまけりせば</rt></ruby>
<ruby><rb>待鹿</rb><rt>ししまち</rt></ruby><ruby><rb>尓</rb><rt>に
</rt></ruby>
<ruby><rb>継而行益乎</rb><rt>つぎていかましを</rt></ruby>
<ruby><rb> 社 </rb><rt> やしろ </rt></ruby><ruby><rb> 師 </rb><rt> し
</rt></ruby><ruby><rb> 怨 </rb><rt> うらめ </rt></ruby><ruby><rb> 焉
</rb><rt>し</rt></ruby>
```

Figure 3: An example of ruby computerization for the original body text.

Table 3: Examples of simplified input data for GIZA++.

	Source: Original unit	Target: Transcribed unit
or-tr	粟 種 有 世 伐	粟 蒔 け り せ ば
tr-ruby	社 師 怨 焉	や し ろ し う ら め し
	社	や し ろ
	師	し
	怨	う ら め
or-ruby	粟 種 有 世 伐	あ は ま け り せ ば
	社	や し ろ
	恨	う ら
	焉	し
character-self	粟	粟
	種	種

c) **Rule 3. Interpolating for alignments 3:** If a character is NULL-aligned and not in { 而, 於, 乎, 于, 矣 , 焉, 也, 兮 }[9] in the original units, we assign it with the same alignment of its rightmost character that is not NULL-aligned in the same unit. If such a character is absent, we assign it same character alignment of the leftmost character that is not NULL-aligned in the same unit.

d) **Rule 4. Remove intersections:** If an m-to-n alignment is either not one original character sequence or one transcribed character sequence (or both), we remove all alignments for those characters without alignment between the leftmost sequences.

e) **Rule 5. Remove initial or final particles alignments:** If a transcribed character sequence start or end with a particle character (or characters) on an m-to-n alignment, we remove the connections to these characters from the original characters, unless the transcribed sequence consists only particle character(s).

f) **Rule 6. Remove "が" alignments:** If an original character is aligned with "我が" in the transcribed unit, we remove the connections from the original character to "が" and its right side characters. Although this "我が" is pos-tagged with noun, strictly speaking the "が" means a case particle.

g) **Rule 7. Remove "み" alignments:** If an original character is not only aligned with "み" whose POS tag is "suffix-substantive-general" in the transcribed unit, we remove the connections from the original character to"み" and its right side characters.

Rules a–c assign some non-NULL connection(s) to NULL-aligned characters (see examples of Figure 5 a–c). Furthermore, rule c makes m-to-1 or m-to-n alignments from 1-to-1 or 1-to-n align-

[9] These original characters are sometimes NULL-aligned, as in Figure 1, and called "置字 (Okiji)."

Table 4: The number of units, source characters, and target characters.

	Number of units	Number of original characters	Number of transcribed characters
or-tr	29,489	128,063	161,561
tr-ruby	25,187	35,543	56,901
or-ruby	44,778	126,646	181,597
character-self	2,196	2,196	2,196

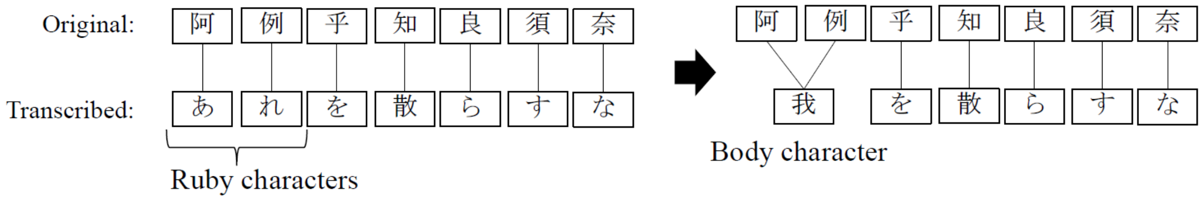


Figure 4: An Example of a created many-to-1 alignment, after mono-ruby characters were restored to a body character.

ments (see examples of Figure 5 c). Eventually, since we want to align one original character sequence to one transcribed word (see Table 2), we remove the intersections between several m-to-n alignments using Rule d. On the other hand, an original character sequence that is not a particle or suffix does not include the meaning of a particle or suffix. Therefore, we remove such alignments using Rules e–g. Such transcribed particle or suffix characters are called "読み添え (Yomisoe)."

5 Evaluation of automatic alignment performance

In our experiment, we compared the precision, recall and F1-measure of our approach across eight datasets.[10] To evaluate alignment performance, we use 79 randomly selected poems from our Man'yōsyū corpus. Two professional researchers of Man'yōsyū probatively annotated the correct alignments for the poems. Table 5 presents the results of the evaluation.

The addition of tr-ruby or character-self to the dataset improves the performance of our alignment in comparison with the or-tr only or the addition of or-ruby. However, the or-tr+or-ruby+character-self dataset results in the best performance. This is because the data are noisy, even though the number of or-ruby units is the largest in our dataset, as can be seen when comparing Figure 3 to Figure 1. We believe that the addition of character-self reined in this noise as a restriction during unsupervised learning. In addition, the proportion of poems that have identical alignments as the correct alignments is 1/79 at most. Since the F1-measures are about 0.83, each poem has 1–2 alignment errors. However, finding these errors and modifying them are less work-intensive and more efficient than fully manual annotation.

Since GIZA++ uses probabilistic models, we can calculate the probability of each m-to-n alignment pair from the output. We normalized the probabilities and use them as the score of the m-to-n alignment pair. We set a threshold value for the score to predict the correct/wrong of the alignment pair, and then investigated a correlation with actual correct/wrong. Consequently, we found that we can distinguish the correctness of an m-to-n alignment pair with high coefficient of correlation (0.925) when the threshold value is 0.15035 (using the or-tr+tr-ruby+or-ruby+character-self). That is, we can modify the errors more efficiently if we begin our modification by checking the alignment pairs with scores below the threshold. We have already started this modification based on the results of our automatic alignment approach (using the or-tr+tr-ruby+or-ruby+character-self) and two workers have completed 1,023/4,516 poems during a period of five months.

[10] As you can see Table 2, our objective annotation is closely resembles Japanese morphological analysis task like (Kudo et al., 2004). Thus, we did not use evaluating measures (precision, recall, F1-measure) for alignment task and used evaluating measures for Japanese morphological analysis task with the same name that can more appropriately evaluate what we want to evaluate.

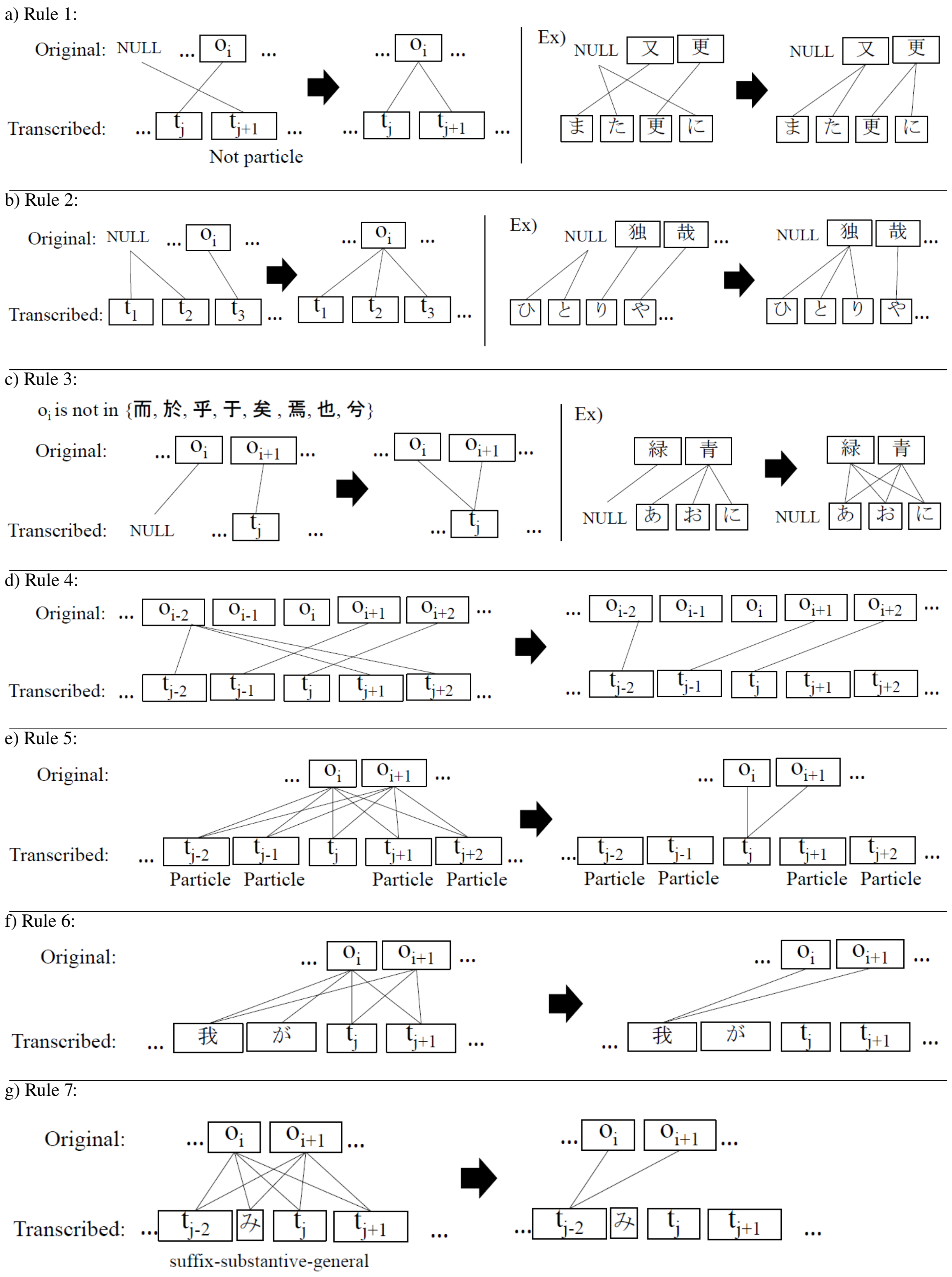


Figure 5: Post-processing rules. The above rules are applied to the GIZA++ output according to the order. The squares represent each character (token), and edges (connections) represent each character alignment between an original character and a transcribed character.

Table 5: Automatic alignment performances for each dataset.

		Perc.	Rec.	F1
(1)	or-tr	0.827	0.826	0.826
(2)	or-tr + tr-ruby	0.830	0.826	0.829
(3)	or-tr + or-ruby	0.827	0.822	0.825
(4)	or-tr + character-self	0.829	0.827	0.828
(5)	or-tr + tr-ruby + or-ruby	0.832	0.827	0.830
(6)	or-tr + tr-ruby + character-self	0.832	0.827	0.829
(7)	or-tr + or-ruby + character-self	**0.834**	**0.828**	**0.831**
(8)	or-tr + tr-ruby + or-ruby + character-self	**0.834**	0.827	0.830

Table 6: The 10 unit-pairs with the highest (normalized) alignment probabilities. The Correct/Wrong column shows whether the unit-pair alignments are completely correct. ［］ shows the mono-ruby (rt) or iterated characters.

Source: Original unit	Target: Transcribed unit	Probability	Correct/Wrong
毛武尓礼乎	もむにれを	0.995	Correct
伊牟礼氐乎礼婆	い群［む］れて居［を］れば	0.993	Correct
乎氐母許乃毛尓	をてもこのもに	0.991	Correct
乎呂能波都乎尓	尾［を］ろのはつをに	0.991	Correct
都芸奈牟毛能乎	継［つ］ぎなむものを	0.991	Correct
保杼呂〻［保］〻［杼］〻［呂］尓	ほどろほどろに	0.991	Correct
伊乎祢受乎礼婆	眠［い］を寝［ね］ず居［を］れば	0.990	Correct
乎良牟等須礼杼	居［を］らむとすれど	0.990	Correct
乎弖毛許乃母尓	をてもこのもに	0.990	Correct
乎弖毛許能母尓	をてもこのもに	0.989	Correct

6 Extra tries

We can calculate the alignment probability of each pair of unit. We normalized and sorted these probabilities (using the or-tr+tr-ruby+or-ruby+character-self). Table 6 shows the 10 best and Table 7 shows the 10 worst unit-pairs. The characters in the original units are all phonographic (1-to-1 alignment) in Table 6. Conversely, in Table 7, most characters in the original units are difficult to read (logographical), which matches our intuition. Despite that both the original and transcribed units consist of only one character and are the same, the numeric characters in Table 7 —六 (six), 二 (two), 四 (four)— are scored poorly. This is because these characters in the original Man'yōsyū are mostly used as phonographic characters, such as "四具礼 (drizzling rain)," rather than for their numerical meanings. These all numeric characters in Table 7 are units for note, and they are exceptional uses. "紫–紫の" also has similar result. However, most uses of "紫" in original units consist of several characters, such as "筑紫奈留"; thus, the case that the original unit consists of only "紫" has low probability. Additionally, "雖小–小［ちひ］さけど" has character-order replacement.

Many OJ researchers have transcribed Man'yōsyū using their own policies. Therefore, many syllable units in the original Man'yōsyū have several transcriptions. We compared the (normalized) probabilities of varied transcriptions that are listed in (Tsuru and Moriyama, 1977) and show this result in Table 8. In this table, we can find transcriptions with higher probabilities than ours. However, these probabilities are calculated from only our transcription; thus, they tell us only, "Which transcription is most likely in our corpus?" At least, from this results, we may as well think that we employ other transcriptions about these transcriptions in our corpus. In these ways, we can find units in our transcription that are difficult to read or uncertain, and then select more likely transcriptions using this comparison.

Table 7: The 10 unit-pairs with the lowest (normalized) alignment probabilities. The Correct/Wrong column shows whether the unit-pair alignments are completely correct. ［］ shows the mono-ruby characters (rt).

Source: Original unit	Target: Transcribed unit	Probability	Correct/Wrong
石穂菅	厳菅	0.094	Correct
向南山	北山に	0.092	Correct
紫	紫の	0.090	Correct
雖小	小［ちひ］さけど	0.087	Wrong
六	六	0.081	Correct
恵得	愛［うるは］しと	0.078	Wrong
二	二	0.061	Correct
従来	昔より	0.047	Wrong
四	四	0.039	Correct
美	愛［うるは］しみ	0.008	Correct

Table 8: Comparing of the normalized alignment probabilities of various transcriptions.

Original unit	Our transcribed unit	Other transcription	Alignment probability
恋等尓	こひしらに		0.270
		こほしらに	0.134
		こふらくに	0.196
		こふとにし	**0.368**
		こふらむに	0.172
結手懈毛	ゆふ手たゆきも		0.068
		ゆふてたゆきも	0.084
		むすぶてうきも	**0.158**
		ゆふ手たゆしも	0.105
		ゆふてたゆしも	0.141
		ゆふてゆるぶも	0.043

7 Conclusion

In this paper, we described how to semiautomatically align the transcribed and original characters to be able to cross-reference them in our Man'yōsyū corpus. Our approach uses GIZA++, which is used in the field of machine translation, and post-processing rules. We also utilized ruby tags as additional training data, and achieved an F1-measure of about 0.83, meaning that is each poem has only 1–2 alignment errors. However, finding and modifying these errors are cheaper and more efficient than using completely manual annotation. Since the coefficient of correlation between the alignment score and alignment correctness is 0.925, the score can be utilized for increasing error-correction efficiency. We have already begun making modifications based on the result of our automatic alignment approach. In addition, we confirmed that we can find the uncertain transcriptions in our corpus and compare them with other transcriptions by using alignment probabilities. We plan to use this approach to investigate the various transcriptions from a statistical perspective as future work. We hope this research will ease and encourage further study of historical works.

Acknowledgements

The work reported in this paper was supported by the NINJAL collaborative research project, "The Construction of Diachronic Corpora and New Developments in Research on the History of Japanese."

References

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. 1991. *Aligning Sentences in Parallel Corpora*. In proceedings of *the 29th Annual Meeting of the Association for national Linguistics* (*ACL-91*), 169–176.

Peter F. Brown, Vincent J. Della. Pietra, Stephen A. Della. Pietra and Robert L. Mercer. 1993. *The Mathematics of Statistical Machine Translation: Parameter Estimation*. *Computational Linguistics*, 19(2):263–311.

Elliott F. Drábek and David Yarowsky. 2005. *Induction of Fine-grained Part-of-speech Taggers via Classifier Combination and Crosslingual Projection*. In proceedings of *the ACL Workshop on Building and Using Parallel Texts* (*ParaText '05*), 49–56.

Noriyuki Kajima, Masatoshi Kinoshita and Haruyuki Touno. 1994. *Shinpen Nihon Koten Bungaku Zensyu*, volume 6–9. Syougakukan, JP.

Taesun Moon and Jason Baldridge. 2007. *Part-of-speech Tagging for Middle English through Alignment and Projection of Parallel Diachronic Texts*. In proceedings of *the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (*EMNLP-CoNLL*), 390–399.

Taku Kudo, Kaoru Yamamoto and Yuji Matsumoto. 2004. *Applying Conditional Random Fields to Japanese Morphological Analysis*, In proceedings of *the 2004 Conference on Empirical Methods in Natural Language Processing* (EMNLP 2004), pp.230-237 *Proc. the 2004 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2004*), 230–237.

Franz J. Och and Hermann Ney. 2003. *A Systematic Comparison of Various Statistical Alignment Models*. *Computational Linguistics*, 29(1):19–51.

Sylwia Ozdowska. 2006. *Projecting POS Tags and Syntactic Dependencies from English and French to Polish in Aligned Corpora*. In proceedings of *the International Workshop on Cross-Language Knowledge Induction* (In *EACL 2006*), 53–60.

Jan L. Pierson. 1929–1963. *The Manyôśû : Translated and Aannotated*, Book 1–20. Brill, Leiden, NED.

Hisashi Tsuru and Takashi Moriyama. 1977. *Man'yōsyū*, expanded edition. Ohfu, Japan.

David Yarowsky and Grace Ngai. 2001. *Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection Across Aligned Corpora*. In proceedings of *the Second Meeting of the North American Chapter of the Association for Computational Linguistics* (*NAACL '01*), 1–8.