Japanese-Chinese Phrase Alignment Using Common Chinese Characters Information

Chenhui Chu, Toshiaki Nakazawa and Sadao Kurohashi
Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku
Kyoto, 606-8501, Japan
{chu, nakazawa}@nlp.kuee.kyoto-u.ac.jp kuro@i.kyoto-u.ac.jp

Abstract

We describe a method to detect common Chinese characters between Japanese and Chinese automatically by means of freely available resources and verify the effectiveness of the detecting method. We use a joint phrase alignment model on dependency trees and report results of experiments aimed at improving the alignment quality between Japanese and Chinese by incorporating the common Chinese characters information detected by proposed detecting method into the alignment model. Experimental results of Japanese-Chinese phrase alignment show that our approach could achieve 0.73 points lower AER than the baseline system.

1 Introduction

Chinese characters are used both in Japanese and Chinese. In Japanese the Chinese characters are called Kanji, while in Chinese they are called Hanzi. Hanzi can be divided into two groups, Simplified Chinese (used in mainland China and Singapore) and Traditional Chinese (used in Taiwan, Hong Kong and Macao). The number of strokes needed to write characters have been largely reduced in Simplified Chinese, and the shapes may be different from the ones in Traditional Chinese. Table 1 gives some examples of Chinese Characters in Japanese, Traditional Chinese and Simplified Chinese.

Because Kanji characters are originated from ancient China, there exist common Chinese characters between Kanji and Hanzi. Actually, the visual forms of the Chinese characters retain certain level of similarity, and many Kanji are identical to Simplified Chinese (e.g. "snow" and "country" in Table 1), some Kanji are identical to Traditional Chinese but different from Simplified Chinese (e.g. "love" in Table 1), but there also exist some visual variations in Kanji (e.g. "begin" and "hair" in Table 1).

On the other hand, Chinese characters contain significant semantic information, and the meanings do not change in most cases between characters in different shapes. For example, the shapes of three characters "発", "發" and "发" in Table 1 are quite different, but all of them have the same meaning "begin".

Based on the characteristics of Kanji and Hanzi described above, we thought that Kanji and Hanzi information may be valuable in machine translation, especially in word/phrase alignment between Japanese and Chinese. Parallel sentences contain equivalent meanings in each language, and we can assume common Chinese characters appear in the sentences. In this paper, we focus on word/phrase alignment between Japanese and Simplified Chinese, where common Chinese characters often have different shapes and it is hard to detect them automatically. We accomplish the detection by means of freely available resources. In addition, we incorporate common Chinese characters information into a joint phrase alignment model on dependency trees.

2 Related Work

Common Chinese characters information have been employed for a number of Japanese-Chinese related tasks. Tan et al. (1995) availed the occurrence of common Chinese characters as a feature of Japanese-Traditional Chinese sentence pair to find a

Meaning	snow	country	love	begin	hair
Kanji	雪	国	愛	発	髪
TC	雪	國	愛	發	髮
SC	雪	国	爱	发	

Table 1: Examples of Chinese characters (TC denotes Traditional Chinese and SC denotes Simplified Chinese).

SC	说	钱	干	故	仿	...
TC	說	錢	干, 幹, 乾	故	仿, 彷, 倣	...

Table 2: Hanzi converter version 3.0 standard conversion table.

direct correspondence in automatic sentence alignment task. Goh et al. (2005) built a Japanese-Simplified Chinese dictionary partly using direct conversion of Japanese into Chinese for the Japanese words that all the characters in the word are made up of Kanji only, namely Kanji words. They did the conversion using a Chinese encoding converter[1] which can convert Traditional Chinese into Simplified Chinese. It works because most Kanji are identical to Traditional Chinese. And for the Kanji with visual variations that cannot be automatically converted using the converter, they manually converted them by hand.

In the context of machine translation, Kondrak et al. (2003) incorporated cognates (words or languages which have the same origin) information in European languages into the translation models of Brown et al. (1993). They arbitrarily selected a subset from the Europarl corpus as training data and extracted a list of likely cognate word pairs from the training corpus on the basis of orthographic similarity, and appended to the corpus itself in order to reinforce the co-occurrence count between cognates. The results of experiments conducted on a variety of bitexts showed that cognate identification can improve word alignments without modifying the statistical training algorithm. Common Chinese characters are kinds of cognates, and it may be possible to improve alignment quality by incorporating common Chinese characters information into Japanese-Chinese alignment models.

3 Common Chinese Characters Detection

Aiming to detect common Chinese characters between Japanese and Simplified Chinese, we do a conversion of Japanese into Chinese. The difference between the study of Goh et al. (2005) and ours is that our proposed method also can convert the Kanji with visual variations into Chinese automatically, while they did it by hand.

3.1 Kanji to Hanzi Conversion

According to the characteristics of Kanji, we divide Kanji into three categories, conversion is needed for Category 2 and 3:

- Category 1: identical to Simplified Chinese
- Category 2: identical to Traditional Chinese but different from Simplified Chinese
- Category 3: visual variations

For category 2, we do a Kanji to Hanzi conversion using the data provided by Chinese encoding converter. Chinese encoding converter is a open source system. It is implemented using a "Hanzi converter version 3.0 standard conversion table" which contains 6,740 corresponding Simplified Chinese and Traditional Chinese character pairs. This table can be downloaded from the web site. Table 2 is a portion of the table. Looking at Table 2, we may notice that a single Simplified Chinese form may correspond to multiple Traditional Chinese forms.

To convert the Kanji in category 3, we use a resource from Unihan database[2]. Unihan database is the repository for the Unicode Consortium's collective knowledge regarding the CJK (Chinese-Japanese-Korean) Unified Ideographs contained in the Unicode Standard[3].

In Unihan database, there is a "Unihan_Variants.txt" file containing character pairs that are unified with some formal rules. The Unicode Standard has adopted a three-dimensional model for determining the relationship between ideographs. The model uses three axes: x-axis to represent meaning, y-axis to represent *abstract* shape, and z-axis to represent visual variations. We are interested in the differences in z-axis. Some common Chinese characters have the same meaning and the same *abstract* shape, and so have the same positions on both the x- and y-axes but different

[1] http://www.mandarintools.com/zhcode.html

[2] http://unicode.org/charts/unihan.html

[3] The Unicode Standard is a character coding system for the consistent encoding, representation and handling of text expressed in most of the world's writing systems. The latest version of the Unicode Standard is Version 6.0.0.

Kanji	説	発	銭	検	経	焼	...
TC	說	發	錢	檢	經	燒	...

Table 3: Visual variations table.

	Japanese	Simplified Chinese
Full Match	説明	说明
Part Match	安否	安全
None Match	言い当てる	猜

Table 4: Example of three word matching types.

positions on the z-axis. Such characters are called z-variants in the database and are identified in the "kZVariant" data field.

We extracted the "kZVariant" data field from "Unihan_Variants.txt" file and obtained a 2,566 visual variational character pairs table. Table 3 is a portion of this table. With this table, we can do Kanji to Hanzi conversion for the Kanji in category 3 by simultaneously using Hanzi standard conversion table.

3.2 Word-to-Word Matching

The Kanji to Hanzi conversion method described above is a character-based one, but machine translation methods always use larger units (words or phrases) than characters. Therefore, we extend the conversion results to word-to-word matching. Figure 1 gives an example of word-to-word matching.

Although Chinese words are basically composed of Kanji only, Japanese words are composed of not only Kanji but Kana (Hiragana and Katakana) and sometimes Kana only. Therefore, the number of Chinese characters in corresponding words do not always equal. We define three types of word-to-word matching: if all the converted Kanji can be found in the Chinese word, we call it Word Full Match; If only part of the converted Kanji can be found in the Chinese word, we call Word Part Match; Otherwise, we call it Word None Match. Table 4 gives an example of these three word matching types.

Of course we can think of other sets of definitions. For example, Word Full Match if all the Chinese characters both in Japanese and Chinese have their counterparts in the other language. However, the definitions described above showed the best performance among various sets of definitions in the preliminary experiments.

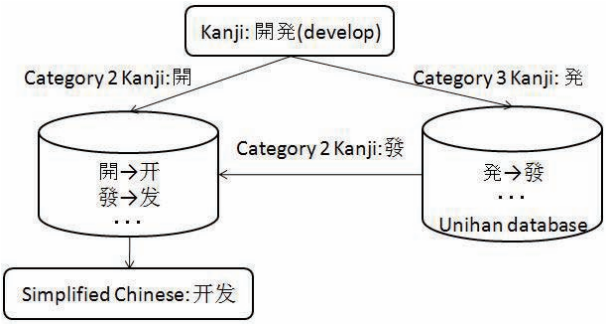

Figure 1: Example of word-to-word matching.

Meaning	envious	wonton	self
Kana	うらやましい	ワンタン	おのれ
Kanji	羨ましい	餛飩	己
TC	羨慕	餛飩	自己
SC	羡慕	馄饨	自己

Table 5: Examples of Kana-Kanji conversion pairs and their corresponding Chinese words.

3.3 Kana-Kanji Conversion

Currently, there are many Japanese words written in Kana even if they have corresponding Kanji expressions, which are accustomed to be used. The Chinese characters in Kanji expressions are again useful as clues to find word-to-word matchings. We can use Kana-Kanji conversion techniques to get the Kanji expressions from Kana expressions, but here, we simply consult a Japanese dictionary of JUMAN (Kurohashi et al., 1994). Table 5 gives some examples of Kana-Kanji conversion results. We only do Kana-Kanji conversion for content words because it is proved that do Kana-Kanji conversion for function words may lead to wrong alignment in the alignment experiments we did.

4 Alignment Model

We used an alignment model proposed by Nakazawa and Kurohashi (2011) which is an extension of the one proposed by Denero et al. (2008). Two main drawbacks of the previous model are the lack of structural information and a naive distortion model. For similar language pairs such as French-English (Marcu and Wong, 2002) or Spanish-English (DeNero et al., 2008), even a simple model that handles sentences as a sequence of words works adequately. This does not hold for distant language pairs such as Japanese-English or Japanese-Chinese, in which word orders differ greatly. The model we used incorporates depen-

dency relations of words into the alignment model (Nakazawa and Kurohashi, 2009) and defines the reorderings on the word dependency trees.

4.1 Generative Story Description

Similar to the previous works (Marcu and Wong, 2002; DeNero et al., 2008), the model we used first describes the generative story for the joint alignment model.

1. Generate ℓ concepts from which subtree pairs are generated independently.
2. Combine the subtrees in each language so as to create parallel sentences.

Here, subtrees are equivalent to phrases in the previous works. One subtree in a concept can be NULL, which represents an unaligned subtree. The model restricts the unaligned subtrees to be composed of exactly one word, because of its simplicity (NULL-alignment restriction).

The number of concepts ℓ is parameterized using a geometric distribution:

$$P(\ell) = p_c \cdot (1 - p_c)^{\ell - 1}. \tag{1}$$

Each concept c_i generates a subtree pair $\langle e_i, f_i \rangle$ from an unknown distribution θ_T, and then they are combined in each language. The model denotes the combinations of subtrees in English as $D_E = \{(j \rightarrow k)\}$, where $(j \rightarrow k)$ denotes that subtree e_j depends on subtree e_k, and in the foreign language as D_F. D refers to D_E and D_F as a whole.

With these notations, the joint probability for a sentence pair is defined as:

$$P(\{\langle e, f \rangle\}, D) = P(\ell) \cdot P(D|\{\langle e, f \rangle\}) \cdot \prod_{\langle e,f \rangle} \theta_T(\langle e, f \rangle). \tag{2}$$

4.2 Subtree Generation

When generating subtrees, the model first decides whether to generate an unaligned subtree (with probability p_ϕ) or an aligned subtree pair (with probability $1 - p_\phi$). DeNero et al. (2008) used $p_\phi = 10^{-10}$ to strongly discourage NULL alignment, but this is not reasonable for some language pairs. Taking Japanese and English as an example, English determiners (a, an, the) and Japanese case markers (*ha*, *ga*, *wo*, etc.) rarely have counterparts. In addition, if the corpus is less clean and sentence pairs often contain a different amount of information, the strict restriction may lead to alignment errors. Therefore, the model uses $p_\phi = 0.33$.

Aligned subtree pairs are generated from an unknown probability distribution θ_A, which obeys the Dirichlet process (DP):

$$\theta_A(\langle e, f \rangle) \sim DP(M_A, \alpha_A), \tag{3}$$

where M_A is the base distribution and α_A is a concentration parameter. The base distribution is defined as:

$$\begin{aligned} M_A(\langle e, f \rangle) &= [P_f(f) P_{WA}(e|f) \cdot P_e(e) P_{WA}(f|e)]^{\frac{1}{2}} \\ P_f(f) &= p_t \cdot (1 - p_t)^{|f|-1} \cdot \left(\tfrac{1}{n_f}\right)^{|f|} \\ P_e(e) &= p_t \cdot (1 - p_t)^{|e|-1} \cdot \left(\tfrac{1}{n_e}\right)^{|e|}, \end{aligned} \tag{4}$$

where P_{WA} is the IBM model1 likelihood (Brown et al., 1993), and n_f and n_e are the numbers of word types in each language. θ_A gives a non-zero weight to aligned subtree pairs only.

Unaligned subtrees are generated from another unknown probability distribution θ_N:

$$\begin{aligned} \theta_N(\langle e, f \rangle) &\sim DP(M_N, \alpha_N) \\ M_N(\langle e, f \rangle) &= \begin{cases} P_{WA}(e|\text{NULL}) & \text{if } f = \text{NULL} \\ P_{WA}(f|\text{NULL}) & \text{if } e = \text{NULL} \end{cases}. \end{aligned} \tag{5}$$

θ_N gives a non-zero weight to unaligned subtrees only. Note that unaligned subtrees are always composed of only one word in the model. Finally, θ_T can be decomposed as:

$$\theta_T(\langle e, f \rangle) = p_\phi \theta_N(\langle e, f \rangle) + (1 - p_\phi) \theta_A(\langle e, f \rangle). \tag{6}$$

4.3 Dependency Relation Probability

Instead of the naive reordering model in the previous work, the model we used considers dependency relations between subtrees and assigns a weight to each relation. Suppose subtree f_j depends on subtree f_k (parent subtree), which means $(j \rightarrow k) \in D_F$, and both f_j and f_k are aligned subtrees. Their counterparts, e_j and e_k respectively, are somewhere on the dependency tree of the other side. The model assumes that e_j tends to depend on e_k because the dependencies between concepts hold across languages. The dependency relation probability reflects this tendency.

Formally, the model extracts a tuple $(N(f_j), rel(f_j, f_{j'}))$ for subtree f_j, and assigns the dependency relation probability to that tuple. For unaligned subtrees, the dependency relation probability is not taken into consideration.

If the parent subtree is an unaligned subtree, it ascends the dependency tree to the root node until an aligned subtree is found. The model calls the nearest aligned subtree a *pseudo parent*. The pseudo parent for subtree f_j is denoted as $f_{j'}$, and the number of unaligned subtrees from f_j to $f_{j'}$ is denoted as $N(f_j)$. The model considers an imaginary root node as a pseudo parent for the root subtree. Japanese function words are often unaligned, but the dependency relations between subtrees stepping over the function words are assumed to hold on the other side. Therefore the model introduces a pseudo parent to capture the relations.

Function $rel(f_j, f_{j'}))$ returns a dependency relation between the counterparts of the two arguments. Note that the counterparts of f_j and $f_{j'}$ are e_j and $e_{j'}$, respectively. The model expresses a dependency relation as the shortest path from one subtree to another. For simplicity, it indicates the path with a pair of non-negative integers, where the first is the number of steps going up (Up) the dependency tree and the other is the number going down ($Down$). It also requires one additional step for going through unaligned subtrees. Consequently, the tuple is represented as a triplet of non-negative integers $R_f = (N, Up, Down)$.

The dependency relation probabilities for the foreign language side are drawn from an unknown probability distribution θ_{fe} and for the English side from θ_{ef}, with both obeying the DP:

$$\begin{aligned} \theta_{fe}(R_f) &\sim DP(M_{fe}, \alpha_{fe}) \\ M_{fe}(R_f) &= p_{fe} \cdot (1 - p_{fe})^{N+Up+Down-1} \\ \theta_{ef}(R_e) &\sim DP(M_{ef}, \alpha_{ef}) \\ M_{ef}(R_e) &= p_{ef} \cdot (1 - p_{ef})^{N+Up+Down-1}. \end{aligned} \tag{7}$$

Using the notations and definitions above, the dependency tree-based reordering model $P(D|\{\langle e, f\rangle\})$ is decomposed as:

$$P(D|\{\langle e, f\rangle\}) = \prod_{\langle e,f\rangle} \theta_{fe}(R_f) \cdot \theta_{ef}(R_e). \tag{8}$$

The model is trained by Gibbs sampling using similar samplers described in DeNero et al. (2008). We skip the detail of the model training here.

4.4 Common Chinese Characters Information Incorporation

We incorporate common Chinese characters information into the alignment model in two ways. One is adjusting the base distribution to reflect the information, and the other is exploiting the information directly into the alignment model. Note that common Chinese characters information has an effect on non-NULL alignments only.

4.4.1 Base Distribution Adjustment

Base distribution for phrase pair generation is derived from IBM model 1 likelihood:

$$p(e|f) = \frac{\epsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} t(e_j|f_i), \tag{9}$$

where $t(e_j|f_i)$ is lexical translation probability estimated by EM algorithm. In the first method, we adjust the lexical translation probability distribution utilizing common Chinese characters information.

For all the lexical translation probabilities, we multiply a weight w according to the information:

$$t(e_j|f_i) = t(e_j|f_i) \cdot w. \tag{10}$$

Then we normalize them

$$t(e_j|f_i) = \frac{t(e_j|f_i)}{\sum_{i=1}^{n} t(e_j|f_i)}, \tag{11}$$

the weight varies with the three word matching types.

4.4.2 Model Modification

In the second method, we define three phrase-to-phrase matching types: if all the words in the Japanese phrase belong to Word Full Match, we call it Phrase Full Match; If all the words in the Japanese phrase belong to Word None Match, we call it Phrase None Match; Otherwise we call it Phrase Part Match. We assign the weight w outside of the base distribution. We modify the model by incorporating the weight w into the subtree generation distribution and redefine the joint probability for a sentence pair as:

$$P(\{\langle e, f\rangle\}, D) = P(\ell) \cdot P(D|\{e, f\}) \cdot \prod_{\langle e,f\rangle} w \cdot \theta_T(\langle e, f\rangle), \tag{12}$$

the weight varies with the three phrase matching types.

5 Experiments

5.1 Coverage of Common Chinese Characters Detection

We investigated the coverage of proposed common Chinese characters detecting method on Japanese-

	Ja	Zh
# of sentences	680k	
# of words	21.8M	18.2M
# of CC	14.0M	24.2M
# of CC(+Kana-Kanji)	14.6M	24.2M
ave. sen. length	32.9	22.7

Table 6: Statistics of the Japanese-Chinese corpus (CC denotes Chinese characters).

	character		word	
	Ja	Zh	Ja	Zh
Category 1	52.41%	30.48%	26.27%	32.09%
+Category 2	68.43%	39.80%	30.87%	37.27%
+Category 3	75.33%	43.81%	32.52%	39.04%
+Kana-Kanji	75.74%	45.82%	34.66%	41.46%
Kanconvit	75.77%	45.82%	34.79%	41.67%

Table 7: Results of detecting method experiments.

Chinese corpus. The corpus is a paper abstract corpus provided by JST[4] and NICT.[5] This corpus was made in the project in Japan named "Development and Research of Japanese-Chinese Natural Language Processing Technology". The statistics of this corpora is shown in Table 6.

We measured the coverage based on both characters and words. For comparison, we used a publicly available tool Kanconvit[6] which uses a small table of equivalent Kanji-Simplified Chinese characters pairs extracted from internet. There are five kinds of experimental settings:

- Category 1: no detecting method used (only exactly the same characters are found)
- +Category 2: proposed Kanji to Hanzi conversion for Kanji in category 2
- +Category 3: proposed Kanji to Hanzi conversion for Kanji both in category 2 and 3
- +Kana-Kanji: both proposed Kanji to Hanzi conversion and Kana-Kanji conversion (number of Chinese characters after Kana-Kanji conversion in the corpus is shown in Table 6)
- Kanconvit: Kanconvit Kanji to Hanzi conversion and Kana-Kanji conversion

The results shown in Table 7 verified the effectiveness of our proposed detecting method. Also, there is complementation between Kanconvit and our proposed detecting method.

[4]http://www.jst.go.jp

[5]http://www.nict.go.jp/

[6]http://kanconvit.ta2o.net/

5.2 Alignment

We conducted alignment experiments on Japanese-Chinese corpus to show the effectiveness of using common Chinese characters information.

5.2.1 Settings

The training corpus we used is the same one we used in Subsection 5.1.

As gold-standard data, we used 510 sentence pairs for Japanese-Chinese which were annotated by hand. There are two types of annotations, sure (S) alignments and possible (P) alignments (Och and Ney, 2003). The unit of evaluation was word. We used precision, recall and alignment error rate (AER) as evaluation criteria. All the experiments were run on original forms of words. We set the weight w to 6000 for both Word and Phrase Full Match, 3000 for both Word and Phrase Part Match and 1 for both Word and Phrase None Match. These weights showed the best performance in the preliminary experiments for tuning the weights.

Japanese sentences were converted into dependency structures using the morphological analyzer JUMAN (Kurohashi et al., 1994), and the dependency analyzer KNP (Kawahara and Kurohashi, 2006). Chinese sentences were converted into dependency trees using the word segmentation and POS-tagging tool by Canasai et al. (2009) and the dependency analyzer CNP (Chen et al., 2008).

For comparison, we used GIZA++ (Och and Ney, 2003) which implements the prominent sequential word-base statistical alignment model of IBM models. We conducted word alignment bidirectionally with its default parameters and merged them using grow-diag-final-and heuristic (Koehn et al., 2003). Also, we used BerkelyAligner[7] (DeNero and Klein, 2007) with its default settings for unsupervised training. Experimental results are shown in Table 8. Common Chinese characters information is detected by our proposed detecting method and Kanconvit. The alignment accuracy of the alignment model we used without incorporating the information is indicated as "Baseline", the alignment accuracy after adjusting the base distribution to reflect the information is indicated as "BD", and the alignment accuracy after exploiting the information directly into the

[7]http://code.google.com/p/berkeleyaligner/

	Pre.	Rec.	AER
grow-diag-final-and	83.77	75.38	20.39
BerkelyAligner	**88.43**	69.77	21.60
Baseline	86.78	76.87	18.14
BD (Proposed)	87.22	76.88	17.93
BD (Kanconvit)	87.24	76.81	17.96
MM (Proposed)	86.88	**78.17**	**17.41**
MM (Kanconvit)	86.89	78.16	17.42

Table 8: Results of Japanese-Chinese alignment experiments.

alignment model is indicated as ”MM”.

5.2.2 Discussion

The results showed that the alignment model we used achieve reasonably high alignment accuracy compared to that of GIZA++ and BerkeleyAligner, and furthermore the alignment accuracy can be improved by incorporating common Chinese characters information. Figure 2 shows an example of alignment improvement after incorporating the information. It successfully discovered the alignment between ”規準” and ”规定” (both mean standard), because there is a common Chinese character (”規” and ”规”), and also, the alignment between ”その” and ”该” (both mean that) could be discovered.

However, compared to the baseline system, the improvement after incorporating common Chinese characters information is not so significant. The reasons for this can be summarized in two aspects.

Firstly, although in most cases the information could achieve correct alignment, there also exist some exceptions. Table 9 gives an exception example, which shows the lexical translation probability of word ”エルニーニョ現象(El Nino phenomenon)” estimated by IBM model 1 and after adjustment using the information. Note that the segmentation results of ”El Nino phenomenon” in Japanese and Chinese are different, JUMAN analyzes it as one word, while the Chinese morphological analyzer recognizes it as two words. Because ”現象” and ”现象” (both mean phenomenon) are common Chinese characters, the lexical translation probability between ”エルニーニョ現象(El Nino phenomenon)” and ”现象(phenomenon)” is reinforced, which is undesirable.

Secondly, the alignment model we used suffers from the accuracy of Chinese parser which could affect the effectiveness of incorporating the information. Although the Chinese parser we used is the state-of-the-art in the world (Chen et al., 2008), the accuracy is less than 80%. While the Japanese parser which we used in the experiments can analyze sentences in over 90% accuracy.

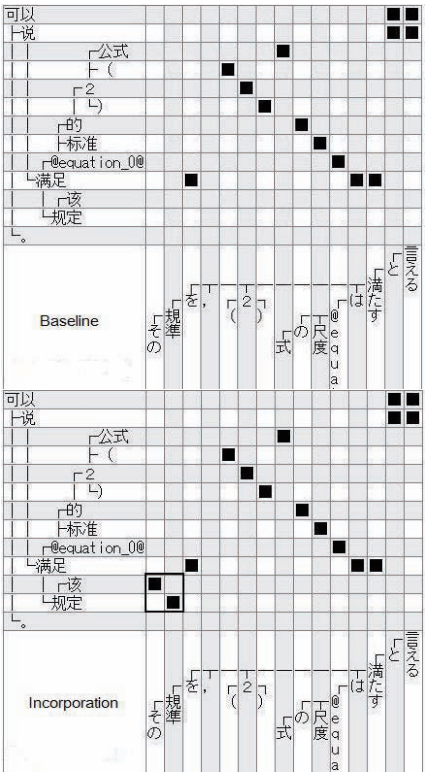

Figure 2: Alignment result from baseline system and after incorporating common Chinese characters information (denoted as Incorporation).

f_i	e_j	model 1	adjustment
エルニーニョ現象	厄尔尼诺	0.317965	0.000998
エルニーニョ現象	现象	0.317965	0.998031
エルニーニョ現象	引起	0.309406	0.000971

Table 9: Lexical translation probability estimated by IBM model 1 and after adjustment.

Also, we notice that between the two ways of incorporating common Chinese characters information, improvement of Base Distribution Adjustment is smaller than Model Modification. The reason of this is that the Base Distribution Adjustment method only adjust the base distribution $M_A(\langle e, f\rangle)$ which has little effect to $\theta_T(\langle e, f\rangle)$. On the other hand, there is also a problem with the method of Model Modification, because after the modification, the joint probability for a sentence pair will not be probability anymore.

6 Conclusion

In this paper we proposed a method to detect common Chinese characters between Japanese and Chinese by means of freely available resources and verified the effectiveness of our proposed detecting method. We incorporated the information into a joint phrase alignment model on dependency trees. Experimental results showed that incorporating the information could achieve 0.73 points lower AER than the baseline system which proved our assumption that Japanese-Chinese phrase alignment quality could be improved using common Chinese characters information.

Although in most cases incorporating common Chinese characters information could achieve correct alignment, there also exist some exceptions. In the future, we plan to survey the exceptional cases, and find a way to deal with the exceptions. Also, our methods of incorporating the information into the joint phrase alignment model have some drawbacks. We plan to develop a more effective method to incorporate the information.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Association for Computational Linguistics*, 19(2):263–312.

Wenliang Chen, Daisuke Kawahara, Kiyotaka Uchimoto, Yujie Zhang, and Hitoshi Isahara. 2008. Dependency parsing with short dependency relation in unlabeled data. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 88–94.

John DeNero and Dan Klein. 2007. Tailoring word alignments to syntactic machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 17–24, Prague, Czech Republic, June. Association for Computational Linguistics.

John DeNero, Alexandre Bouchard-Côté, and Dan Klein. 2008. Sampling alignment structure under a Bayesian translation model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 314–323, Honolulu, Hawaii, October. Association for Computational Linguistics.

Chooi-Ling Goh, Masayuki Asahara, and Yuji Matsumoto. 2005. Building a Japanese-Chinese dictionary using kanji/hanzi conversion. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 670–681.

Daisuke Kawahara and Sadao Kurohashi. 2006. A fully-lexicalized probabilistic model for Japanese syntactic and case structure analysis. In *Proceedings of Human Language Technology Conference of the NAACL, Main Conference*, pages 176–183, New York City, USA, June. Association for Computational Linguistics.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL 2003: Main Procedings*, pages 127–133.

Grzegorz Kondrak, Daniel Marcu, and Kevin Knight. 2003. Cognates can improve statistical translation models. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 46–48.

Canasai Kruengkrai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentara Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint chinese word segmentation and pos tagging. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 513–521, Suntec, Singapore, August. Association for Computational Linguistics.

Sadao Kurohashi, Toshihisa Nakamura, Yuji Matsumoto, and Makoto Nagao. 1994. Improvements of Japanese morphological analyzer JUMAN. In *Proceedings of the International Workshop on Sharable Natural Language*, pages 22–28.

Daniel Marcu and Daniel Wong. 2002. A phrase-based,joint probability model for statistical machine translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 133–139. Association for Computational Linguistics, July.

Toshiaki Nakazawa and Sadao Kurohashi. 2009. Statistical phrase alignment model using dependency relation probability. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation (SSST-3) at NAACL HLT 2009*, pages 10–18, Boulder, Colorado, June. Association for Computational Linguistics.

Toshiaki Nakazawa and Sadao Kurohashi. 2011. Bayesian subtree alignment model based on dependency trees (to appear). In *Proceedings of the 5th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, November.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Association for Computational Linguistics*, 29(1):19–51.

Chew Lim Tan and Makoto Nagao. 1995. Automatic alignment of Japanese-Chinese bilingual texts. *IEICE Transactions on Information and Systems*, E78-D(1):68–76.