Correction of Errors in a Modality Corpus Used for Machine Translation Using Machine-learning

Masaki Murata, Masao Utiyama, Kiyotaka Uchimoto, Qing Ma,
and Hitoshi Isahara
Communications Research Laboratory
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, JAPAN
{murata,mutiyama,uchimoto,qma,isahara}@crl.go.jp

Abstract

We performed corpus correction on an annotated corpus for machine translation using machine-learning methods such as the maximum-entropy method. We thus constructed a high-quality annotated corpus based on corpus correction. We compared several different methods of corpus correction in our experiments and developed a suitable method for correction. Recently, corpus-based machine translation has been investigated. Since corpus-based machine translation uses corpora, the corpus correction we discuss in this paper should prove to be significant.

1 Introduction

In recent years, various types of tagged corpora have been constructed and used extensively in research. However, tagged corpora include errors which impede the progress of research. The correction of these errors is therefore an important research issue.[1]

We have researched error correction in corpora using an annotated corpus that we are currently constructing. This corpus consists of supervised learning data used for research on translating Japanese tense, aspect, and modality into English (Murata et al. 1999; Murata et al. 2001). We call this a modality corpus. (In this paper, we regard the word *modality* in the broad sense of including tense and aspect.) Tense, aspect, and modality are known to present difficult problems in machine translation. In traditional approaches, tense, aspect, and modality have been translated using manually constructed heuristic rules. Recently, however, corpus-based approaches, such as the example-based method, have also been applied. The modality corpus we consider in this paper is necessary for corpus-based machine translation.

We describe the modality corpus in Section 2, the method of corpus correction in Section 3, and our experiments on corpus correction in Section 4.

2 Modality Corpus for Machine Translation

In this section, we describe the modality corpus. A part of the modality corpus is shown in Figure 1. It is composed of a Japanese-English bilingual corpus; each English sentence can include two types of tags:

[1]There is no previous paper on error correction in corpora. In terms of error detection in corpora, there has been research using boosting or anomaly detection (Abney et al. 1999; Eskin 2000).

, *kono kodomo wa aa ieba kou iu kara koniku-rashii*
This child always talks back to me, and this <v>is</v> why I <vj>hate</vj> him.

d *kare ga aa okubyou da to wa omowana-katta*
I <v>did not think</v> he was so timid.

c *aa isogashikute wa yasumu hima mo nai hazu da*
Such a busy man as he <v>cannot have</v> any spare time.

Figure 1: Part of the modality corpus

- The English main verb phrase is tagged with <v>.
- The English verb phrase corresponding to the Japanese main verb phrase is tagged with <vj>.

The symbols at the beginning of each Japanese sentence, such as "c" and "d", indicate a category of tense, aspect, and modality for the sentence. For example, "c" and "d" indicate "can" and "past tense", respectively. The first symbol in Figure 1 is ",". This symbol is used when <vj> is used; the left part indicates the category of the verb phrase tagged with <v> and the right part indicates the category of the verb phrase tagged with <vj>. In this corpus, there is a large number of examples of the present tense, so the symbol for the present tense is a null expression (i.e., ""). <vj> is tagged when the verb phrase with <v> does not correspond to the Japanese main verb.

We use the following 34 categories for tense, aspect, and modality. These categories are determined by the surface expressions of the English verb phrases.

1. all combinations of {present tense, past tense}, {progressive, not-progressive}, and {perfect, not-perfect} (eight categories)
2. imperative mood (one category)
3. auxiliary verbs ({present tense, past tense} of "be able to", {present tense, past tense} of "be going to", "can", "could", {present tense, past tense} of "have to", "had better", "may", "might", "must", "need", "ought", "shall", "should", "used to", "will", "would") (19 categories)
4. noun phrases (one category)
5. participial construction (one category)
6. verb ellipsis (one category)
7. interjection or greeting sentences (one category)

8. when a Japanese main verb phrase does not correspond to an English verb phrase (one category)

9. when tagging cannot be performed (one category)

These categories of tense, aspect, and modality are defined on the basis of the surface expressions of the English sentences. This means that if we can estimate the correct category for a Japanese sentence, we should be able to translate the Japanese tense, aspect, and modality into English. Therefore, in researching the translation of modality expressions based on the machine-learning method, only the tags indicating the categories of tense, aspect, and modality and the Japanese sentences were used.

We commissioned a sub-contractor to construct the modality corpus according to the above conditions. We used about 40,000 example sentences from the Kodansha Japanese-English dictionary (Shimizu & Narita 1976) as a bilingual corpus. The sub-contractor performed the tagging of <v> and the corresponding categories of modality by hand. The work was inspected at least twice, until the subcontractor considered that the corpus contained no errors.

3 Method of Corpus Correction

In this section, we describe the method used to correct errors in the manually constructed modality corpus. We calculated the probabilities of tags, which are objects for error correction in a corpus, and then performed corpus correction using those probabilities. In this paper, we only consider tags for modality categories, not "<v>" and "<vj>" tags.

We tested two different methods for calculating the probability of each tag: the maximum-entropy method, and the decision-list method.[2]

- Method based on the maximum-entropy method (Ristad 1997; Ristad 1998)

  In this method, the distribution of probabilities $p(a, b)$ is calculated for the case when Equation (1) is satisfied and Equation (2) is maximized. The desired probabilities $p(a|b)$ are then calculated using the distribution of probabilities $p(a, b)$:

$$\sum_{a\in A, b\in B} p(a,b) g_j(a,b) = \sum_{a\in A, b\in B} \tilde{p}(a,b) g_j(a,b) \quad for\ \forall f_j\ (1 \le j \le k) \tag{1}$$

$$H(p) = -\sum_{a\in A, b\in B} p(a,b)\ log\left(p(a,b)\right), \tag{2}$$

  where A, B, and F are sets of categories, contexts, and features $f_j (\in F, 1 \le j \le k)$, respectively; $g_j(a, b)$ is a function defined as 1 when context b has feature

[2] In this paper, we used the maximum-entropy method and the decision-list method to calculate the probabilities of each tag. However, we may use a more accurate method to calculate the probabilities for corpus correction.

f_j and the category is a, or defined as 0 otherwise; and $\tilde{p}(a, b)$ is the rate of occurrence of (a, b) in the training data.

In general, the distribution of $\tilde{p}(a, b)$ is very sparse. We cannot use it directly, so we must estimate the true distribution of $p(a, b)$ from the distribution of $\tilde{p}(a, b)$. We assume that the estimated values of the frequency of each category/feature pair as calculated from $\tilde{p}(a, b)$ are the same as those from $p(a, b)$ (this corresponds to Equation (1).) These estimated values are not so sparse. We can thus use the above assumption for calculating $p(a, b)$. Furthermore, we maximize the entropy of the distribution of $\tilde{p}(a, b)$ to obtain one solution of $\tilde{p}(a, b)$, because using only Equation 1 produces several solutions for $\tilde{p}(a, b)$. Maximizing the entropy has the effect of making the distribution more uniform and is considered to be a good solution for data sparseness problems.

- Method based on the decision-list method (Yarowsky 1994)

  In this method, the probability of each category is calculated using one of the features, $f_j (\in F, 1 \leq j \leq k)$. The probability that produces category a in context b is given by the following equation:

$$p(a|b) = p(a|f_{max}), \tag{3}$$

  such that f_{max} is defined by

$$f_{max} = argmax_{f_j \in F} \; max_{a_i \in A} \; \tilde{p}(a_i|f_j), \tag{4}$$

  where $\tilde{p}(a_i|f_j)$ is the occurrence rate of category a_i when the context has feature f_j.

In this paper, we used the following items as features, which are the context when the probabilities are calculated; 26 $(= 5 + 10 + 10 + 1)$ features appear in each English sentence:

- the strings of 1-gram to 5-grams just to the left of <v> in the sentence.

  (e.g. I <v>did not think</v> he was so timid.)

- the strings of 1-gram to 10-grams just to the right of <v>.

  (e.g. I <v>did not think</v> he was so timid.)

- the strings of 1-gram to 10-grams just to the left of </v>.

  (e.g. I <v>did not think</v> he was so timid.)

- the 1-gram string at the end of the sentence.

  (e.g. I <v>did not think </v> he was so timid.)

When the verb phrase was divided into two parts, as in an interrogative sentence, the above extraction of features was performed after eliminating the words between the first </v> and the second <v>.

Because the corpus used in this paper was designed to estimate the modality of the English sentence from the Japanese sentence, readers might think that we should have extracted the features from the Japanese sentence. That would be true if we wanted to infer English modalities from Japanese sentences. What we wanted to do, however, was to correct the English modality tags. Thus, we had to use all the information available. Since the category of the modality expression of the English sentence was tagged and the verb phrase of the English sentence was examined during the manual construction of the corpus, it was reasonable to use the English verb phrase in corpus correction based on machine-learning.

Next, we describe the method used to judge whether each tag in the corpus was correct or incorrect. We first calculated the probabilities of the category of the tag, and of the other categories. We judged that the tag was correct when its category had the highest probability and incorrect when one of the other categories had the highest probability. Next, we corrected the tag if it was judged incorrect. This correction was performed by changing the tag to the tag of the category with the highest probability (corrections were checked by annotators.) [3]

Corpus correction must be checked by human beings, which makes it very time consuming. However, when the probabilities of each tag can be calculated, we can define the confidence value of the corpus correction, as described below. It is thus more convenient to sort the error candidates in the corpus by confidence values and begin by correcting the errors with higher confidence values.

We tested the two following methods for determining the confidence value for corpus correction:

- **Method 1** — the probability of the category with the highest probability is used as the confidence value for corpus correction.
- **Method 2** — the non-probability of the tag originally defined is used as the confidence value for corpus correction.

In this paper, the non-probability is defined as the value obtained by subtracting the probability from 1.

We then explain the methods of using data for calculating probabilities. There are two methods for calculating the probabilities using the machine-learning method:

- calculation of probabilities for closed data,
- calculation of probabilities for open data.

The first method calculates probabilities using all the tags in the corpora including the tag currently being judged. The second method does not use this tag. In this paper, 10-fold cross validation was used for calculating the probabilities for open data. [4]

[3] This method of corpus correction is equivalent to re-estimating the tag in the corpus using a machine-learning method and re-tagging the newly estimated tag.

[4] When the probabilities are calculated using open data in the decision-list method, the probability

4 Experiments on Corpus Correction

We carried out experiments on corpus correction using the methods described in the previous section. These experiments were performed after eliminating the sentences given tags indicating that tagging could not be performed. Thus, these experiments were performed on 39,718 modality tags. The results are shown in Tables 1 to 4; "random 300" indicates the precisions for 300 tags extracted randomly from among the tags corrected by our system; and "top X" indicates the precisions for the top X tags sorted by Method 1 or Method 2. "Precision for detection" indicates the percentage of tags for which detection of an error succeeded in causing the tag to be corrected by our system, while "Precision for correction" indicates the percentage of tags for which correction of an error succeeded in causing the tag to be corrected by our system.

We came to the following conclusions based on the experimental results:

- Throughout all the experiments, the precisions for detection and correction were almost the same. Thus, we found it more convenient to perform both correction and detection, rather than detection only.

  From the viewpoint of manual modification, when we modify tags by hand, it is also more convenient for the system to produce a candidate category that is tagged to the corpus after corpus correction. This tells us how the original tag was incorrect and how we should change it. In other words, when only detection is performed, a candidate category is not presented and an annotator may not know why the tag is incorrect.

- In general, the maximum-entropy method produced higher precision than the decision-list method. However, when closed data was used to calculate the probabilities, the precisions of the top items were almost the same for both methods.

- In terms of the precisions of top items, using closed data to calculate the probabilities was better than using open data. However, in terms of the total number of extracted items, using open data was better.

- In terms of sorting by Method 1 or Method 2, Method 1 generally produced higher precisions for the top items than Method 2.

- In terms of comparing "random 300" and "top X", "top X" produced much higher precisions for the top items than "random 300". We thus found that sorting by confidence values for corpus correction is very important.

On the basis of the above results, we prefer the following strategy:

1. We first perform high-quality corpus correction using the probability calculation for closed data and Method 1.

of the category of the original tag is apt to be 0; the probability of the category of the tag defined after corpus correction is apt to be 1, because the calculation is performed without using the original tag. Thus, when there are several such tags, many of them have the same probability and sorting by probabilities becomes difficult. In this case, we sorted the tags by arranging those whose probability was calculated from features that had many tags in descending order of confidence value for corpus correction.

Table 1: Precision of corpus correction using the maximum-entropy method (probabilities were calculated using closed data. 184 candidate errors were extracted.)

			Precision for detection		Precision for correction	
random 300			69%	(127/184)	68%	(126/184)
Method 1	top	50	100%	(50/ 50)	100%	(50/ 50)
	top	100	92%	(92/100)	92%	(92/100)
	top	150	77%	(116/150)	77%	(116/150)
	top	200	69%	(127/184)	68%	(126/184)
	top	250	—	—	—	—
	top	300	—	—	—	—
Method 2	top	50	88%	(44/ 50)	88%	(44/ 50)
	top	100	81%	(81/100)	81%	(81/100)
	top	150	74%	(112/150)	74%	(111/150)
	top	200	69%	(127/184)	68%	(126/184)
	top	250	—	—	—	—
	top	300	—	—	—	—

Table 2: Precision of corpus correction using the maximum-entropy method (probabilities were calculated using open data. 694 candidate errors were extracted.)

			Precision for detection		Precision for correction	
random 300			28%	(84/300)	26%	(78/300)
Method 1	top	50	88%	(44/ 50)	88%	(44/ 50)
	top	100	88%	(88/100)	88%	(88/100)
	top	150	80%	(121/150)	79%	(119/150)
	top	200	68%	(136/200)	67%	(134/200)
	top	250	60%	(151/250)	59%	(149/250)
	top	300	53%	(160/300)	52%	(157/300)
Method 2	top	50	72%	(36/ 50)	72%	(36/ 50)
	top	100	74%	(74/100)	71%	(71/100)
	top	150	70%	(106/150)	68%	(102/150)
	top	200	67%	(135/200)	65%	(131/200)
	top	250	60%	(152/250)	58%	(147/250)
	top	300	52%	(157/300)	50%	(152/300)

Table 3: Precision of corpus correction using the decision-list method (probabilities were calculated using closed data. 383 candidate errors were extracted.)

			Precision for detection		Precision for correction	
random 300			34%	(104/300)	33%	(101/300)
Method 1	top	50	100%	(50/ 50)	100%	(50/ 50)
	top	100	92%	(92/100)	92%	(92/100)
	top	150	76%	(115/150)	74%	(112/150)
	top	200	62%	(124/200)	60%	(121/200)
	top	250	51%	(128/250)	50%	(125/250)
	top	300	44%	(132/300)	43%	(129/300)
Method 2	top	50	88%	(44/ 50)	86%	(43/ 50)
	top	100	86%	(86/100)	84%	(84/100)
	top	150	71%	(107/150)	69%	(104/150)
	top	200	59%	(118/200)	57%	(115/200)
	top	250	50%	(126/250)	49%	(123/250)
	top	300	43%	(129/300)	42%	(126/300)

Table 4: Precision of corpus correction using the decision-list method (probabilities were calculated using open data. 694 candidate errors were extracted.)

			Precision for detection		Precision for correction	
random 300			6%	(18/300)	6%	(18/300)
Method 1	top	50	56%	(28/ 50)	52%	(26/ 50)
	top	100	43%	(43/100)	40%	(40/100)
	top	150	31%	(47/150)	29%	(44/150)
	top	200	26%	(52/200)	24%	(48/200)
	top	250	22%	(55/250)	20%	(51/250)
	top	300	20%	(61/300)	19%	(57/300)
Method 2	top	50	66%	(33/ 50)	64%	(32/ 50)
	top	100	48%	(48/100)	46%	(46/100)
	top	150	44%	(66/150)	42%	(63/150)
	top	200	35%	(71/200)	34%	(68/200)
	top	250	30%	(77/250)	29%	(73/250)
	top	300	26%	(80/300)	25%	(76/300)

2. Next, we perform corpus correction for a much larger number of tags using the probability calculation for open data, the maximum-entropy method, and Method 1.

5 Conclusion

We have described corpus correction using a machine-learning method for a modality corpus for machine translation. We constructed a high-quality modality corpus using corpus correction. The resulting modality corpus is very useful for studies on Japanese-English translation of tense, aspect, and modality. Recently, corpus-based machine translation has been studied. Since corpus-based machine translation uses corpora, the corpus correction described in this paper should prove relevant.

Our method of corpus correction has the following advantages:

- There has been no previous paper on error correction in corpora.

  In terms of error detection in corpora, there has been other research using boosting or anomaly detection (Abney et al. 1999; Eskin 2000). We found that the precisions for detection and correction were almost the same. Therefore, we should perform correction in addition to detection.

- Our method calculates the probability of each tag and can sort the error candidates in the corpus using these probabilities as confidence values for corpus correction. Thus, we can begin to correct errors with higher confidence values.

- Our method uses the machine-learning method and inherits its original advantages:

  - Our method has the same wide applicability as the machine-leaning method and can be used to correct various types of corpora.
  - A large amount of human effort is not necessary. Humans only have to provide the appropriate feature sets used in the machine-learning method.

References

Abney, Steven, Robert E. Schapire & Yoram Singer: 1999, 'Boosting applied to tagging and PP attachment', *EMNLP/ VLC-99*.

Eskin, Eleazar: 2000, 'Detecting errors within a corpus using anomaly detection', *NAACL-2000*.

Murata, Masaki, Qing Ma, Kiyotaka Uchimoto & Hitoshi Isahara: 1999, 'An example-based approach to Japanese-to-English translation of tense, aspect, and modality', in *TMI '99*, pp. 66–76.

Murata, Masaki, Kiyotaka Uchimoto, Qing Ma & Hitoshi Isahara: 2001, 'Using a support-vector machine for Japanese-to-English translation of tense, aspect, and modality', *ACL Workshop, the Data-Driven Machine Translation*.

Ristad, Eric Sven: 1997, 'Maximum entropy modeling for natural language', ACL/EACL Tutorial Program, Madrid.

Ristad, Eric Sven: 1998, 'Maximum entropy modeling toolkit, release 1.6 beta', http://www.mnemonic .com/software/memt.

Shimizu, Mamoru & Narimasu Narita, eds.: 1976, *The KODANSHA Japanese-English Dictionary*, Kodansha.

Yarowsky, David: 1994, 'Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French', in *32rd Annual Meeting of the Association for Computational Linguistics*, pp. 88–95.