A Practical of Memory-based Approach for Improving Accuracy of MT

Sittha Phaholphinyo, Teerapong Modhiran, Nattapol Kritsuthikul and Thepchai Supnithi
Text Processing Section, Division of Research and Development on Information,
National Electronics and Computer Technology Center
112 Paholyothin Road, Klong 1, Klongluang, Pathumthani, 12120, Thailand
{Sittha.Phaholphinyo, Teerapong.Modhiran,
Nattapol.Kritsuthikul, Thepchai.Supnithi}@nectec.or.th

Abstract

Rule-Based Machine Translation (RBMT) [1] approach is a major approach in MT research. It needs linguistic knowledge to create appropriate rules of translation. However, we cannot completely add all linguistic rules to the system because adding new rules may cause a conflict with the old ones. So, we propose a memory based approach to improve the translation quality without modifying the existing linguistic rules. This paper analyses the translation problems and shows how this approach works.

1 Introduction

“ParSit” is English to Thai ruled-based Machine Translation [2],[3]. It has been launched in Thailand since 2000. This approach requires much linguistic knowledge. However, it is difficult to completely create all rules. Moreover, adding new rules may conflict with the old ones. Because of this limitation, we propose another approach to improve the translation quality.

In this paper, we will discuss the ParSit's errors in details. Then we show the overview of Post-Editing tool and how Post-Edit technique improves on ParSit's translation quality.

2 Characteristics of ParSit Errors

Charoenpornsawat et al. [4] claim that the common errors of machine translation can be classified into two types: incorrect meaning and incorrect ordering. The former, covering 81.74% of all errors, can be classified again into missing some words, generating over words and using incorrect words.

As said by Meechoonuk and Rakchonlatee [5], there are 12 types of linguistic problems found in ParSit translation. Table 1 shows a summarization of the mentioned problems

| Types of Linguistic problems | Source Sentence | Human Translation | ParSit Translation | Analysis |
|-----------------------------|----------------|-------------------|-------------------|----------|
| Mismatched concepts         | You will be ten years old next year. | คุณ-คุณ|จะเป็น-will be|10ปี-ten years old|ปี-year|จะเป็น-will be|10ปี-ten years old|ปี-year|จะเป็น-will be|10ปี-ten years old|ปี-year|วัย-next |
| Misplaced modifiers         | He lives quite near here. | เขา-he|อยู่-live|ใกล้-near|อยู่-live|ใกล้-near|อยู่-live|quilte |
| Inappropriate literal translation | Until recently, the company was in the red. | จน-untill|ล่าสุด-recently|บริษัท-the company|อยู่-was|ใน-in|บริษัท-the company|อยู่-was|ใน-in|บริษัท-the company|อยู่-was|ใน-in|RED in this sentence refers to "lost" but ParSit translates it as "color". |
| Addition of words or phrases | The native of Java are like the Malays. | ชนเผ่า-ethnic|ของ-Java|น่า-like |ชนเผ่า-ethnic|ของ-Java|น่า-like |ที่-Malays. |

ARE LIKE is used in present simple tense but ParSit translates it in the form of present continuous tense.
We can map the 12 types of linguistic problems found on ParSit translation with the concept idea of machine translation errors. The re-arrangement in table 2 shows that using incorrect words is comprised of 9 types of linguistic problems. We suppose that Post-Edit technique can improve translation quality by reducing the errors of incorrect words as well as incorrect ordering. We will describe all the processes of our system, experiment and result in the next sections.
3 ParSit with Post-Edit module

3.1 System overview

Figure 1 shows the original ParSit. The English sentence is analyzed and generated to target sentence in Thai. We improve translation quality by adding Post-Edit module to the system as shown in Figure 2. The Post-Edit module detects some translation errors and corrects them to the appropriate form before sending the final output to users.

3.2 Alignment tool

To construct Post-Edit rules, Alignment tool is developed to create input and output alignment. This tool is originated from Cairo program [6], used in Statistical Machine Translation (SMT) [7]. Cairo program concentrates only on the translation pair, while our tool concentrates on translation error correction.

ParSit generates both acceptable and unacceptable output. The input of the tool, called Alignment file is a pair of an English sentence and its unacceptable translation in Thai. We use Alignment tool to modify the translation result which is used to improve the translation quality.

We can open these files with Alignment tool as shown in Figure 3.

Figure 3: Alignment tool

Considering Figure 3, the topmost level is English source sentence. The middle part refers to ParSit generated output. The bottommost level is provided for human editing. There are two types of connection lines between levels. The upper ones link between the source sentence and ParSit output. These lines cannot be changed. The lower ones express a relation between ParSit output and human edited translation. Alignment file format are shown in Figure 4. Each file contains WORD, POS, ordering information in each pair sentence.

Figure 4: Alignment file format.

Alignment tool is used to modify words in the bottommost layer and the connection line between ParSit sentence and Human edited sentence. The commands in alignment tool that provides for linguists are shown as follows:

Add word is used to insert a new word.
Delete word is used to remove an unnecessary word.
Edit word is used to change a word.
Add link is used to insert line between the middle to the bottommost layer.
Delete link is used for removing unnecessary line between the middle and the bottommost layer.
The data collection which is rectified by human is called “Post-Edit Corpus”. It is used for creating rules in Post-Edit module.

The main criteria of editing alignment file are:
- Edit only comprehensible sentence.
- Edit only necessary part in each sentence, not exceed three positions.
- Omit long, complex or incomplete sentence.

### 3.3 Post-Edit Rule

As mentioned above, Post-Edit rules are generated from collection of edited alignment files. In the current version, we use three types of relations to construct Post-Edit rules.

**One-to-one relation** is used to create a rule to change any wrong word to the proper one.

**Many-to-one relation** is used to create a rule to combine multiple words into a new single word.

**One-to-many relation** is used to create a rule to split a single word to multiple words.

---

### 3.3.1 Structure of Memory-based Rule Information

We apply Memory-based to assist rule generation. We consider word and POS, in source sentence as necessary features for creating rules.

The structure of information is:

```
Left#Center#Right#Source_word#Target_word#Human_word
```

Figure 8: Example of alignment file.

Figure 8 shows an example of alignment file. We can create a rule to correct the word "ช่วยมาก- much/ ของ-of/ ขนาด-size/" to be "ขนาดใหญ่ -large size/" by 2 types of features.

First, we focus on two left and two right neighboring words. We will get "they," were" and "." as word features. From this example, the information for constructing rule is:

**Left:** They were
**Center:** of a
**Right:**

**Source_word:** (much|POTH)(of|PREP)(a|DOTH size|N)
**Target_word:** ช่วยมาก-ของ-ขนาด
**Human_word:** ขนาดใหญ่

Second, we use POS of two left and two right neighboring POS as feature. The information for constructing rule will be:

**Left:** POTH AUX
**Center:** PREP DOTH
**Right:**

**Source_word:** (much|POTH)(of|PREP)(a|DOTH size|N)
**Target_word:** ช่วยมาก-ของ-ขนาด
**Human_word:** ขนาดใหญ่

In some cases, two types of features can be combined to get better rules.

We store the information in trie structure[8] so that we can access rapidly. Trie structure is also appropriated to large amount of information.

In our data structure, Trie composed of two types of data; keyword and a collection of
Post-Edit rule. keyword is a data that point to its own representative Post-Edit rule collection.

![Diagram](image.png)

Figure 9: Data structure of rule information

We use production rule to represent Post-Edit Rule. It contains `<condition>` terms and `<action>` terms as shown in the following pattern.

```
IF <condition 1> ... <condition n> THEN
  <action 1> ... <action n>
```

The Post-Edit rule pattern is defined as follows:

```
IF Match(Target_word) THEN
  IF( Match(Left_word)
      && Match(Right_word)
      && Match(Center_word))
  THEN
    Assign(Target_word Human_word);

From the example shown in Figure 8. Post-Edit rule pattern (two left and two right neighboring words) will be

```
IF (Match(Target_word "Death is a necessary end.")
THEN
  IF (Match(Left_word "There isn’t any logic to his move."
                           && (Match(Right_word "He nakedly declared his intentions.")
                           && (Match(Center_word "To envy our betters is human nature.")
                           && (Match(Left_word "Success is near at hand.")
                           && (Match(Right_word "There isn’t any logic to envying our betters.")
                           && (Match(Center_word "To envy our betters is human nature.")
                           && (Match(Left_word "We have tested this technique with our 6,000 bilingual sentences and found that Post-Edit technique can reduce some important errors which frequently occur in ParSit’s translation result. These improvements are shown below.

4.1 Improvement in incorrect meaning

The improvement in this section is categorized in two types: incorrect words reduction and over words reduction.

4.1.1 Incorrect words reduction

| Source Sentence | ParSit Translation | ParSit Translation with Post Edit technique | Improvement in incorrect meaning |
|-----------------|--------------------|---------------------------------------------|----------------------------------|
| A serious accident occurred last year. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Inappropriate literal translation |
| An overpowering burst of ammonia assaulted my lungs. | อาชญาชนิว | อาชญาชนิว | Inappropriate literal translation |
| He is a nailer at lying. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Insufficient definition of idioms |
| This subject lies near my heart. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Mismatched concepts |
| Death is a necessary end. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Inappropriate literal translation |

Table 3: Incorrect words reduction

4.1.2 Over words reduction

| Source Sentence | ParSit Translation | ParSit Translation with Post Edit technique | Improvement in incorrect meaning |
|-----------------|--------------------|---------------------------------------------|----------------------------------|
| There isn’t any logic to his move. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Addition of words or phrases |
| He nakedly declared his intentions. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Insufficient dictionary definition |
| To envy our betters is human nature. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Mismatched concepts |
| Success is near at hand. | ข้าตั้นชัยนีนักฉันมีกลุ่ม | ข้าตั้นชัยนีนักฉันมีกลุ่ม | Translation does not conform to Thai grammar |

Table 4: Over words reduction
4.2 Improvement in incorrect ordering

| Source Sentence | ParSit Translation | ParSit Translation with Post Edit technique | Improvement in incorrect ordering |
|-----------------|-------------------|--------------------------------------------|----------------------------------|
| The tree are naked of leaves. | รัติ-is nerve | รัติ-is nerve | - Misplaced modifiers |
| We are a democracy only in name. | นี-are | นี-are | - Insufficient definitions of nouns, two-word verbs, and phrasal verbs. |
| An old man was fascinated by her lily hands. | ชери-are | ชери-are | |
| There is a lack of naval officers. | ชุฉึ-there is | ชุฉึ-there is | |
| Some of the provincial newspaper are as influential as the national newspaper. | แปล-are | แปล-are | |

Table 5: Incorrect ordering reduction

Charoenpornsawat et al. [4] had improved the qualities of translation by applying machine learning to the machine translation results. Focused on verb-to-be, they claimed that the accuracy had been increased about 25.1-29.7 per cent compared to the original results. We assimilate an experiment by applying these rules to Ripper [9] to generalize rule. However, the numbers of rules are too few, because the sparseness of data caused only partial features are analyzed. Finally we got too general rule.

However, Post-Edit Technique cannot improve some error such as missing word(s). Because this technique are based on context information, if the target sentence lack of the meaning or translation, the word's link on alignment tool also cannot link to the source word. Considering the Post-edit rule, we currently use the output from the corpus to generate rules based of word and POS features.

5 Conclusion and Future work

The advantage of this technique is to improve some weak point of machine translation such as incorrect meaning reduction. This technique requires a large size of input data to succeed. Our next step is to collect a larger size of corpus, handle with a long, complicate data set and apply an machine learning technique for automatic rule generation without sparseness problem.

References

[1] Sergei Nirenburg, Jaime Carbonell, Masaru Tomita and Kenneth Goodman. Machine Translation: A Knowledge-Based Approach

[2] Virach Sornlertlamvanich, Paisarn Charoenpornsawat, Mithika Boriboon and Lalida Boonmana. 2000. ParSit: English-Thai Machine Translation Services on Internet. 12th Annual Conference, ECIT and New Economy, National Electronics and Computer Technology Center, Bangkok, June 2000. (in Thai)

[3] ParSit : http://www.suparsit.com

[4] Paisarn Charoenpornsawat, Virach Sornlertlamvanich and Thatsane Charoenporn. 2002. Improving Translation Quality of Rule-based Machine Translation. In the 19th International Conference on Computational Linguistics (Coling 2002): Workshop on Machine Translation in Asia. Taipei, Taiwan.

[5] Manisara Meechoonuk and Somporn Rakchonlatee. 2001. An analysis of text translated by machine. National Institute of Development Administration.

[6] Smith, Noah A. and Michael E. Jahr (2000). "Cairo: An Alignment Visualization Tool," in the proceedings of The Second International Conference on Language Resources and Evaluation (LREC 2000), Athens, Greece.

[7] Al-Onaizan, Yaser, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky (1999). "Statistical Machine Translation: Final Report," Johns Hopkins University 1999 Summer Workshop (WS 99) on Language Engineering, Center for Language and Speech Processing, Baltimore, MD, USA.

[8] Trie algorithm: http://linux.thai.net/~thep/

[9] Cohen, William W. 1995.. Fast effective rule induction, In Proceedings of the Twelfth International Conference on Machine Learning, Lake Taho, California, Morgan Kaufman.