A Chinese-English Machine Translation System
Based on Micro-Engine Architecture

Liu Qun

Institute of Computing Technology, Chinese Academy of Sciences

Institute of Computational Linguistics, Peking University

Introduction

Despite the use of a number of technologies in the design of machine translation systems, none of them has produced an optimal output on free text. A multi-engine MT approach has therefore been proposed to integrate several MT engines in one system (Frederking and Nirenburg, 1994). Such an approach, as shown in the following diagram (Figure 1), has been successfully used by a number of MT systems (Frederking *et al*., 1994:95-100; Frederking, Rudnicky and Hogan, 1997:61-65; Nirenburg, 1996:96-105; Rayner and Carter, 1997:107-10). Experiments have shown the results of using multi-engine MT system are indeed better than any of the single MT engines in the system (Hogan and Frederking, 1998).

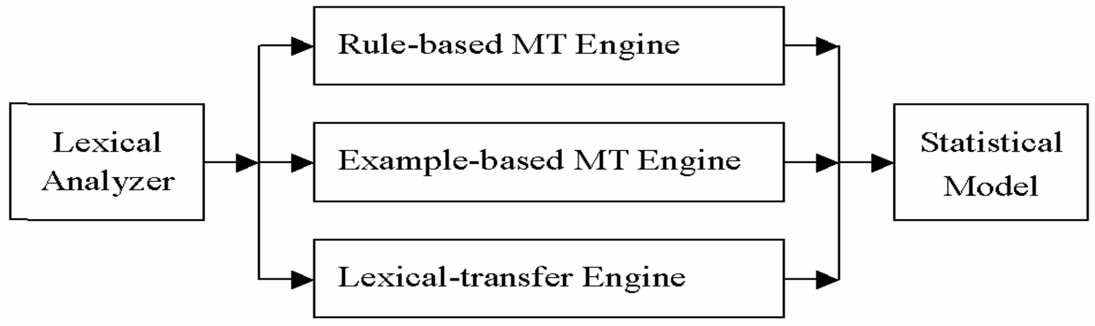

Fig. 1 Structure of Multi-Engine MT

In such a system, each engine tries to translate the source sentence separately, gives a series of translations of the phrases in the source sentence, and then puts the resulting output segments into a shared chart-like data structure. All the partial translations can then be given an internal quality score. A chart-walk algorithm is used to find the best combination of the partial translation.

In multi-engine architecture, the engines work independently. That means, an engine cannot make use of the result of other engines. For example, an example-based MT (EBMT) engine can translate a Chinese sentence “我喜歡看電影,” because there is a sentence “我喜歡看電視劇” in the corpus. But for the sentence “我喜歡看成龍演的這部電影,” the EBMT engine cannot give the result, because there is no sample in the corpus can match the phrase "成龍演的這部電影.” It is possible that a rule-based MT (RBMT) engine can translate this phrase correctly. But in the multi-engine system, the EBMT engine cannot use the result given by RBMT engine.

Here we give a micro-engine approach to machine translation and introduce a Chinese-English machine translation system using such an approach. Similar with the multi-engine approach, it can synthesize the result of the deferent MT engine. What is more, engines in the micro-engine system can interact with each other.

The Micro-engine Architecture

A micro-engine MT system consists of several micro-engines and an engine manager. All the micro-engines share a chart data structure. The engine manager also maintains an active constituent list. An active constituent is a constituent recognized by a micro-engine but has not been used to generate new constituents. The engine manager selects the best active constituent from the active constituent list and sends it to all the micro engines. The micro-engines recognize new constituents using this active constituent and the existing inactive constituents (the edges in the chart). The engine manager will add these new constituents to the active constituents list and the previous selected active constituent will be moved from the active constituent list to the chart. This process repeats itself until a constituent cover the whole input sentence is recognized.

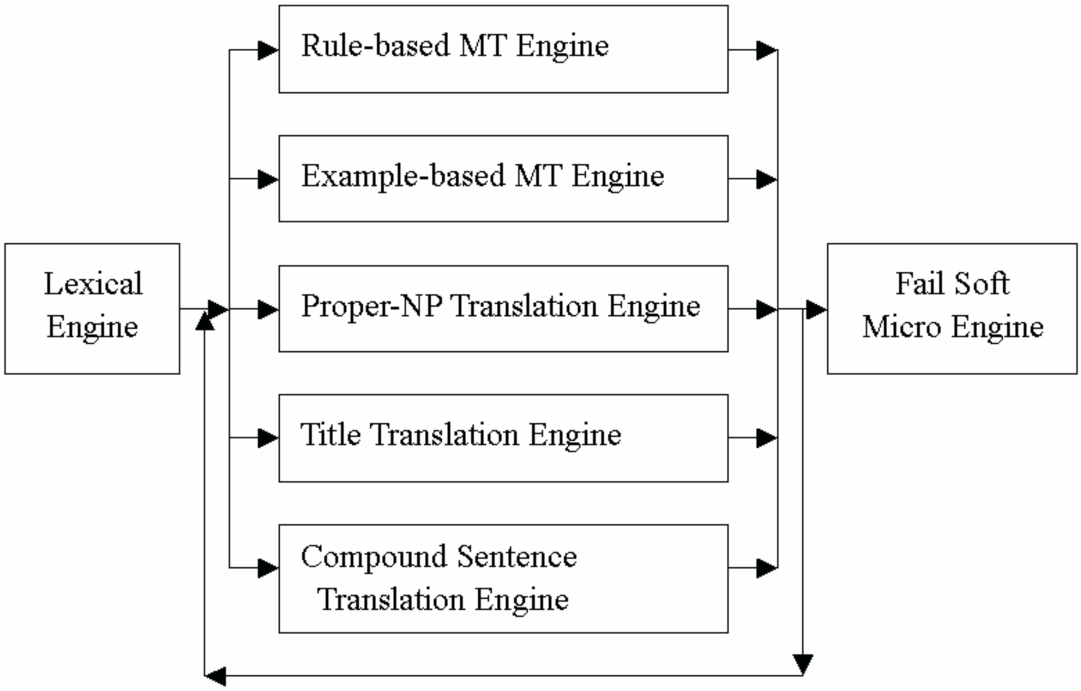

Fig.2 Structure of Micro-Engine MT Architecture

The Micro-engine

A micro-engine is a machine translation engine. Unlike a traditional MT engine, a micro-engine does not try to translate the whole input sentence. A micro-engine is specialized. It only tries to find a specific type of constituent in the input sentence and translate these constituents. All the engines work cooperatively to translate the whole input sentence. That means, an engine can make use of the results generated by other micro-engines.

Normally, a micro-engine should implement two functions:

(1) *Recognition*

The micro-engine accepts an active constituent, combines it with the existing inactive constituents, and generates a list of new constituents.

(2) *Translation*

The engine should translate the constituent it recognized. It may call the Translate function of the micro-engines that recognize the sub-constituents.

The Engine Management Algorithm

Data:

Chart — containing all the inactive constituents

ActiveList — the active constituent list

EngineList — the list of micro-engines

Algorithm:

```
Use the lexical engine to recognize all the words in the source sentence
Add these words into ActiveList
Repeat while ActiveList is not empty
    TheEdge = the constituent with the highest score in ActiveList
    If TheEdge covers the whole input sentence
        Call the Translate function of TheEdge
        Return the result translation text
    EndIf
    For EachEngine in EngineList
        Call the Recognize function of EachEngine using TheEdge as input
        Add all the output constituents to ActiveList
    EndFor
    Remove TheEdge from ActiveList
    Add TheEdge to Chart
    Sort ActiveList according to certain criterion
EndRepeat
If no constituent covering the whole sentence is recognized
    Use the Fail-soft Engine to find a best combination of existing constituents
    Translate the constituents in the combination
    Return the result translation text
EndIf
EndAlgorithm
```

Lexical Engine and Fail-soft Engine

Normally, if the input active constituent is empty, the micro-engine's Recognition function will not do anything. But there are two exceptions. These two micro-engines are called the Lexical and the Fail-soft Engines.

The Lexical Engine is the engine that carries out the lexical analysis. That means, to lookup the dictionary, to segment the Chinese sentence to words, and to recognize Chinese personal names and place names. Its Recognition function works only when the input active constituent and the chart are both empty.

The Fail-soft Engine is used when there is no constituent covering the whole input sentence recognized. It selects the best combination of the existing constituents in the chart and generates the translation based on them. Its Recognition function works only when the input constituent is empty and the chart is not empty.

Other Micro-engines

In addition to the Lexical Engine and Fail-Soft Engine, we use five other micro-engines in our Chinese-English machine translation system.

One of the micro-engines is a rule-based engine. This engine is constructed from a traditional rule-based Chinese-English machine translation system. (Liu and Yu, 1998: 514-17) There are about 300 syntax rules in this engine. It uses a chart-parsing algorithm to parse the sentence.

Another micro-engine is an example-based engine. We collect a bilingual corpus with about 200,000 words. Most texts in the corpus consist of news or editorials of Xinhua news agency or government white books.

The third micro-engine is a proper-NP translation engine. This engine can recognize proper noun phrases from the Chinese sentence and translation them into English. The proper noun phrases include person name phrases, place name phrases, organization name phrases, time phrases, number phrases, money phrases, and so on.

The fourth micro-engine is a title translation engine. The title of Chinese articles usually have a special syntax structure, such as "機器翻譯的預處理研究", "試論網絡黑客的行爲方式" and "魯迅傳". This micro-engine can recognize these kind of titles and translate them properly.

The fifth micro-engine is a compound sentence translation engine. This engine can find the logical relations between the simple sentences in a compound sentence according to the conjunction words, and translate the sentence properly.

An Example

Here we give an example to show how the micro-engine MT system works. For an Chinese sentence:

演員帕特里克·斯威茲在他最近的一部電影中扮演了一個感人的保鏢角色。

(Actor Patrick Swayze played a touch bouncer in one of his recent movies.)

The Lexical Engine will look up the dictionary and cut the sentence into words:

演員/n 帕/g 特/g 里/f 克/v •/w 斯/g 威/g 茲/g 在/p 他/r 最近/a 的/u

一/m 部/q 電影/n 中/f 扮演/v 了/u 一/m 個/q 感人/a 的/u 保鏢/n 角色/n 。/w

The labels following each words, such as "n", "g", "w" and etc., is the part-of-speech tags of the words.

The Proper-NP Translation Engine will recognize the "*帕特里克·斯威茲*" as a transliteration of a foreign name:

帕特里克·斯威茲/n (Patrick Swayze)

The Rule-Based MT Engine will recognize and translate the constituents as below:

Np 演員帕特里克·斯威茲 (Actor Patrick Swayze)

Pp 在他最近的一部電影中 vp (in his recent a movie)

Vp 扮演了一个感人的保鏢角色 np (played a role of a touch bodyguard)

S 演員帕特里克·斯威茲在他最近的一部電影中扮演了一個感人的保鏢角色。

（Actor Patrick Swayze played a role of a touch bodyguard in his recent a movie.）

The result translation is not so grammatical in English.

While the Example-based MT Engine can translation some phrases in other way:

Np 在他最近的一部電影中 vp (in one of his recent movies)

Vp 扮演了一个感人的保鏢角色 np (played a touch bodyguard)

These partial translations are better, because the Example-based MT Engine can translate phrases by the comparing them with the similar examples in the corpus, rather than translate them according to the manually written rules.

Finally the Rule-based MT Engine will synthesis the intermediate result to an accepted translation:

S 演員帕特里克·斯威茲在他最近的一部電影中扮演了一個感人的保鏢角色。

(Actor Patrick Swayze played a touch bodyguard in one of his recent movies.)

In this example, we can see that different micro-engines work cooperatively and the translation is better than what may be generated by any of the single engine.

Conclusion and Future Work

Both micro-engine MT system and multi-engine system can employ different MT technologies in a single system. But there are still differences between them.

The granularity of engines in a micro-engine system is finer than that of engines in a multi-engine system. In a multi-engine system, each engine should be a complete MT system. It tries to translate the whole input sentence. But in a micro-engine system, each engine has its specialty. A micro-engine does not need to try to translate the whole sentence. It just needs to translate the “familiar” part of the sentence and ignore the rest of the sentence.

A micro-engine system is a close coupling system. In such a system, all the engines work cooperatively. The relation between the micro-engines is cooperative, rather than competitive and an engine can take the intermediate results of other engines as its input. In contrast, a multi-engine system is a loose coupling system. Engines in a multi-engine system work separately.

The micro-engine architecture of machine translation has much strength that other engines cannot match. It is easy to develop. It is true that the engines can work cooperatively in a micro-engine system, the programming interface between micro-engines is, however, rather simple. The micro-engines need not handle the complicated communications between them. The system also has good scalability. Adding new micro-engines to a micro-engine system will not cause modification of the old system. And because the micro-engines need not translate the whole sentence, developers can focus their attention on a specific problem such as improving accuracy. Lastly, the engine management algorithm can be easily modified to a parallel algorithm, which will take the advantage of rapid parallel computer that has several CPUs. And it is our intention to develop more micro-engines in the future to improve our MT systems.

References

Frederking, Robert and Sergei Nirenburg (1994). “Three Heads are Better than One.” *Proceedings of the Fourth Conference on Applied Natural Language Processing* (ANLP-94), Stuttgart, Germany, pp.95-100.

Frederking, Robert, *et al*. (1994). “Integrating Translations from Multiple Sources with the Pangloss Mark III Machine Translation System.” *Proceedings of the First Conference for Machine Translation in the Americas* (AMTA), Columbia, Maryland, October, pp.73-80.

Frederking, Robert, Alexander Rudnicky and Christopher Hogan (1997). “Interactive Speech Translation in the DIPLOMAT Project.” Steven Krauwer *et al.*, eds., *Spoken Language Translation: Proceedings of a Workshop,* Association of Computational Linguistics and European Network in Language and Speech, Madrid, Spain, July, pp.61-65.

Hogan, Christopher and Robert Frederking (1998). “An Evaluation of Multi-engine MT

Architecture." David Farwell *et al.*, eds., *Machine Translation and the Information Soup*. New York: Springer-Velagg, pp.113-23.

Liu, Qun and Yu Shiwen (1998). "TransEasy: A Chinese-English Machine Translation System Based on Hybrid Approach." David Farwell *et al*., eds., *Machine Translation and the Information Soup*. New York: Springer-Velagg, pp.514-17.

Nirenburg, Sergei *et al*. (1996). "Two Principles and Six Techniques for Rapid MT Development." *Proceedings of the Second Conference of the Association for Machine Translation in the Americas* (AMTA), Montreal/Quebec, Canada, October, pp.96-105.

Rayner, Manny and David Carter (1997). "Hybrid Processing in the Spoken Language Translator." *Proceedings of ICASSP-97*, Munich, Germany, pp.107-10.