# Document-Level Machine Translation with Hierarchical Attention

*Experiments with statistical machine translation*

Presented to Prof. Chris Pollett
Department of Computer Science
San Jose State University

In Partial Fulfillment
Of the Requirements of the Class
Fall 2022: CS 297

By
Yu-Tang Shen
September 2022

## Table of Contents

# I. Introduction on Statistical Machine Translation

While rule-based machine translation (RBMT) was thriving from 1930s to 1950s, Warren Weaver proposed an idea of utilizing statistics in languages to produce translations [1]. Although RBMT could provide satisfying translation, it required complex rule definition, pre-editing, and post-editing. In contrast, statistical machine translation (SMT) proposed to simply look at the relationship between the statistics of the source language (SL) and the probability of all the possible candidates in target language (TL). With SMT, it was not necessary for programmers to define every single rule between SL and TL, instead, the task became gathering abundant amount of data to obtain an unbiased statistic for every word.

To mathematically describe SMT task, the job can be described as find the best candidate $\hat{t}$ for the source language vocabulary $s$ from the TL distribution $T$.

$$\hat{t} = argmax_{t \in T} P(t|s)$$

With more complicated model, which will be covered in section II, parameters $p$ will be added to the equation to improve the translation quality.

$$\hat{t} = argmax_{t \in T} P(t|s;p)$$

## II. Statistical Machine Translation Types

There are various designs in SMT, including implementation from Charniak [2], IBM [3], etc.

### A. *Greedy algorithm*

To understand how SMT works, this paper presents a simplest SMT implementation: a greedy translation algorithm. As the name suggests, it chooses the best candidate based on its current options and disregards all other information. A simple greedy algorithm can be described as follows:

```
translate(S)
  result = φ
  for s in S
    candidates = getCandidate(s)
    bestCandidate, bestScore = φ, -1
    for candidate in candidates
      if(probability(s, candidate) > bestScore)
        bestCandidate = candidate
        bestScore = probability(s, candidate)
      end if
    end for
    result = result + bestCandidate
  end for
  return result
```

In `translate`, the algorithm first retrieves all the candidates for the vocabulary `s` in the given input sentence `S`, then compare the probability of word `s` translated into `candidate` for all `candidates`.

There are several disadvantages in this method, such as overlooking information in adjacent words, TL ordering, and TL usages. This method merely considers the probability of given SL word translated into some TL word, which completely ignores words' neighbors, which two or more of them can become a phrase. Another common issue in machine translation is the output ordering, for example, TL grammar dictates that subjects should be in front of verbs, while verbs go before subjects in SL grammar. The greedy method preserves the original ordering SL and can fail to reorder the words to produce a grammatically correct TL sentence.

Still, this method provides a more concise way to build a simple machine translator, where rules need not to be specified.

## B. *Charniak's model*

In Charniak's SMT model, more information is considered when choosing the best candidate for a translation. In Charniak's model, not only the relationship between SL and TL is considered but also the usage of the TL [2].

Charniak developed an English parser that describes the probability of specific word being which part of speech [4]. For example, given a word "said", the probability of it being followed by two consecutive nouns is low, while for the word "gave" the probability of it being followed by two consecutive nouns is high. This parser describes the usage of a language; in other words, it is easy to lookup if a word is used correctly by its probability.

With this parser that describes a language, Charniak utilizes it to optimize the selection process during SMT. As described in the first section, SMT selects best candidate by $\hat{t} = argmax_{t \in T} P(t|s)$, and Charniak expanded this formula to $\hat{t} = argmax_{t \in T} P(s|t)P(t)$. Since the distribution of $P(s)$ is fixed once the input is given, two equations are equivalent. Recall that the parser is a function that gives the probability of a given word being which part of speech, the parser serves as $P(t)$ in Charniak's SMT implementation.

Yamada and Knight [5] published a translation model that takes an English parsing and outputs a translated Chinese sentence. Charniak reversed the process to improve the translation quality from Chinese to English. Instead of retrieving a Chinese sentence from an English parsing, all possible parsing that can yield the input Chinese sentence are retrieved. Charniak's SMT model then search among all the possible parsing for the highest probability. For example, a given Chinese sentence "你好嗎" can be the result from these parsing: "you good", "how are you", "how do you", etc., in Charniak's parser, P("how are you") should have the highest probability, making it the final output of the model.

## III. EXPERIMENT

A modified version of greedy SMT is implemented in this report, where in addition to considering the relationship between SL and TL, the adjacent words information and ordering issue are explored.

In this experiment, English to Chinese mapping is retrieved from Taiwan Panorama[1] organized by National Academy for Educational Research[2], and the usage of Chinese is obtained from [6]. After lemmatizing the input data, the core translation function can be abstracted as follows:

```
1  translate(S)
2      result := get_current_translation(S)
3      for i in S.length
4          //maximize the probability on both SL-TL mapping and TL usage
5          result[i] := argmax_prob(result[i-1], word, result[i+1], dictionary)
6      end for
7      for i in result.length
8          // swap if two TL words gives a higher probability when flipped
9          if(TL_prob(result[i] + result[i+1]) < TL_prob(result[i+1] + result[i])
10              result[i], result[i+1] := result[i+1], result[i]
11          end if
12     end for
13     return result
```

In the beginning, the model will check if there is an available translation to improve on, and when the SL sentence is not available, it simply translates the input by considering the SL-TL relationship. With an initial translation to work on, the model does the following:

$$argmax_{t_i \in T} P(t_i|s_i) + P(t_{i-1}, t_i, t_{i+1})$$

The former term describes the probability of a SL word $s_i$ being translated into the TL word $t_i$, and the latter term defines how probable [$t_{i-1}$ (the TL word before $t_i$), $t_i$, $t_{i+1}$(the TL word after $t_i$)] is in TL usage.

To address the ordering problem, the program examines if swapping items in bigrams increases the bigram probability in TL usage in the loop in line 7. To describe line 7 mathematically, the loop can be illustrated as:

$$argmax_{\pi=[t_i,t_{i+1}]\in T}\ P(\pi)$$

The function `translate` is repeated multiple times until the probability stops increasing.

[1] https://www.taiwan-panorama.com/en/Periodical

[2] http://coct.naer.edu.tw/bc/

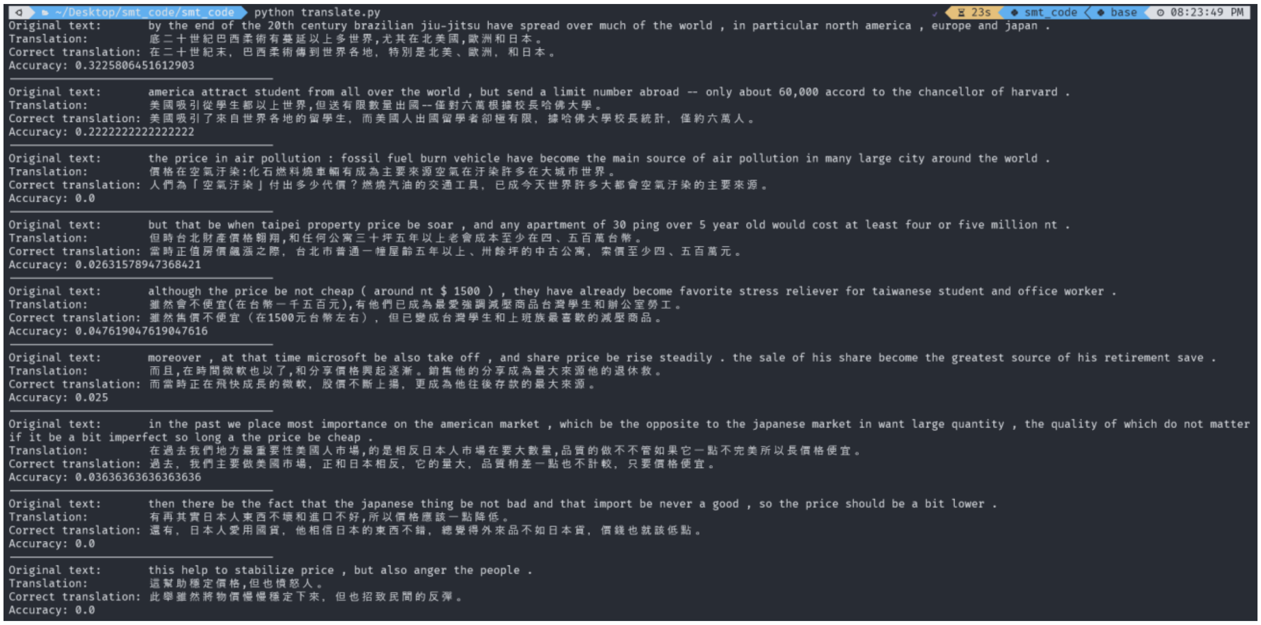
Two additional experiments are tested to improve translation quality: translate to empty and translate based on TL usage.

### A. *Translate to empty*

If a SL word can mean an empty string, there will be a slight probability `translate` selects empty string regardless SL-TL probability and TL usage. Still, this selected translation will compare with all the other translations to ensure that this translation indeed improves the quality.

### B. *Translate based on TL usage*

With a slight probability, TL candidates can be selected in the following manner: given the previous translated TL vocabulary $t_{i-1}$, look at the TL usage and select the most common word that follows $t_{i-1}$, and make it the translation of $s_i$.



```
~/Desktop/smt_code/smt_code  python translate.py                                  23s  smt_code  base  08:23:49 PM
Original text:       by the end of the 20th century brazilian jiu-jitsu have spread over much of the world , in particular north america , europe and japan .
Translation:         底二十世紀巴西柔術有蔓延以上多世界,尤其在北美國,歐洲和日本。
Correct translation: 在二十世紀末，巴西柔術傳到世界各地，特別是北美、歐洲，和日本。
Accuracy: 0.3225806451612903
────────────────────────────────────────

Original text:       america attract student from all over the world , but send a limit number abroad -- only about 60,000 accord to the chancellor of harvard .
Translation:         美國吸引從學生都以上世界,但送有限數量出國--僅對六萬根據校長哈佛大學。
Correct translation: 美國吸引了來自世界各地的留學生，而美國人出國留學者卻極有限，據哈佛大學校長統計，僅約六萬人。
Accuracy: 0.2222222222222222
────────────────────────────────────────

Original text:       the price in air pollution : fossil fuel burn vehicle have become the main source of air pollution in many large city around the world .
Translation:         價格在空氣汙染:化石燃料燒車輛有成為主要來源空氣在汙染許多在大城市世界。
Correct translation: 人們為「空氣汙染」付出多少代價？燃燒汽油的交通工具，已成今天世界許多大都會空氣汙染的主要來源。
Accuracy: 0.0
────────────────────────────────────────

Original text:       but that be when taipei property price be soar , and any apartment of 30 ping over 5 year old would cost at least four or five million nt .
Translation:         但時台北財產價格翱翔,和任何公寓三十坪五年以上老會成本至少在四、五百萬台幣。
Correct translation: 當時正值房價飆漲之際，台北市普通一幢屋齡五年以上、卅餘坪的中古公寓，索價至少四、五百萬元。
Accuracy: 0.02631578947368421
────────────────────────────────────────

Original text:       although the price be not cheap ( around nt $ 1500 ) , they have already become favorite stress reliever for taiwanese student and office worker .
Translation:         雖然會不便宜(在台幣一千五百元),有他們已成為最愛強調減壓商品台灣學生和辦公室勞工。
Correct translation: 雖然售價不便宜（在1500元台幣左右），但已變成台灣學生和上班族最喜歡的減壓商品。
Accuracy: 0.047619047619047616
────────────────────────────────────────

Original text:       moreover , at that time microsoft be also take off , and share price be rise steadily . the sale of his share become the greatest source of his retirement save .
Translation:         而且,在時間微軟也以了,和分享價格興起逐漸。銷售他的分享成為最大來源他的退休救。
Correct translation: 而當時正在飛快成長的微軟，股價不斷上揚，更成為他往後存款的最大來源。
Accuracy: 0.025
────────────────────────────────────────

Original text:       in the past we place most importance on the american market , which be the opposite to the japanese market in want large quantity , the quality of which do not matter
if it be a bit imperfect so long a the price be cheap .
Translation:         在過去我們地方最重要性美國人市場,的是相反日本人市場在要大數量,品質的做不不管如果它一點不完美所以長價格便宜。
Correct translation: 過去，我們主要做美國市場，正和日本相反，它的量大，品質稍差一點也不計較，只要價格便宜。
Accuracy: 0.03636363636363636
────────────────────────────────────────

Original text:       then there be the fact that the japanese thing be not bad and that import be never a good , so the price should be a bit lower .
Translation:         有再其實日本人東西不壞和進口不好,所以價格應該一點降低。
Correct translation: 還有，日本人愛用國貨，他相信日本的東西不錯，總覺得外來品不如日本貨，價錢也就該低點。
Accuracy: 0.0
────────────────────────────────────────

Original text:       this help to stabilize price , but also anger the people .
Translation:         這幫助穩定價格,但也憤怒人。
Correct translation: 此舉雖然將物價慢慢穩定下來，但也招致民間的反彈。
Accuracy: 0.0
```

The translation accuracy is measured by simply comparing the prediction with the ground truth divided by the translation sentence length. The accuracy is within the range of 32% to 0%, but most of the results can convey the meaning of the original SL sentence.

## IV. CONCLUSION

SMT provides an easier way to implement machine translation compared to RBMT. SMT selects the most suitable translation over the TL space by considering either SL statistics, TL statistics, SL-TL mapping, or all of them.

In the experiment section, it shows that it is more powerful than RBMT, where it can translate more sentences without manually adding all the underlying rules and the ability to produce a more grammatically correct order. Still, translation quality from SMT is not optimal, as seen in the experiment in this report, and more advanced technique should be used to capture the complex transition from one language to another.

## REFERENCES

[1] W. J. Hutchins, "Machine translation: A brief history," in *Concise history of the language sciences*, Elsevier, 1995, pp. 431-445.

[2] E. Charniak, K. Knight and K. Yamada, "Syntax-based language models for statistical machine translation," in *Proceedings of Machine Translation Summit IX: Papers*, 2003.

[3] P. Brown, V. Della Pietra, S. Della Pietra and R. Mercer, "The Mathematics of Statistical Machine Translation: Parameter Estimation," *Computational Linguistics,* vol. 19, pp. 263-311, 1993.

[4] E. Charniak, "Statistical Techniques for Natural Language Parsing," *AI Magazine,* 1997.

[5] K. Yanada and K. Knight, "A decoder for syntax-based statistical MT," *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics,* 2002.

[6] T. Emerson, "The Second International Chinese Word Segmentation Bakeoff," in *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, 2005.