Goal

Building a machine translation system that translates Modern Chinese into Classical Chinese.

Data

I use Sima Qian's *Records of the Grand Historian*, also known by its Chinese name *Shiji* (史記), except for the *Table* (表) volumes. The original text contains more than five hundred thousand Chinese characters.

The original text I use comes from several sources. I mainly use the text from Gutenberg Project, which comes as a nice single plain text file[1]. However, about 1,000 characters are corrupted, which are found with the following regular expression:

```
[^\u2E80-\u9FFF，。；：“”‘’《》！？、·「」『』…\r\n]
```

I recovered each corrupted character by cross-referencing Wikisource[2] and the Chinese Text Project[3]. The original text is then converted from traditional Chinese characters to simplified Chinese characters using ConvertZ[4].

The translation I use mainly comes from read126.cn[5]. (Obviously this is not originally where the translation was published, but I cannot find the original source.) There are also corrupted characters in translations, which are harder to deal with. Here's what I did:

1. Try to find the same translation from other sources by Googling the nearby uncorrupted sentence;
2. Try to find other translations by Googling the corresponding original text in classical Chinese;
3. If the corrupted character is a name of some person / location / plant /animal, try to recover by consulting the corresponding original sentence in classical Chinese;
4. Occasionally when all the above fail, try to fill in a translation (as a native speaker).

I also tried to correct the wrong punctuation by looking for consecutive ，。；：！？、• characters. For each case, I checked whether it indicated a missing sentence or was just a typo.

The data is divided into three parts: `train` (4192 paragraphs, 1.29MB), `dev` (524 paragraphs, 259KB) and `test` (524 paragraphs, 200KB) for model training, tuning and evaluation respectively.

[1] http://www.gutenberg.org/files/24226/24226-0.txt

[2] http://zh.wikisource.org/wiki/史記

[3] http://ctext.org/shiji

[4] http://dl.pconline.com.cn/html_2/1/76/id=495&pn=0.html

[5] http://www.read126.cn/194c6894-51d5-4df3-a4bc-fa1282139f82!c0856342-2132-4498-921c-d81450904044!66ee2899-4ffa-41e9-ae09-126ca0281c65.html

Evaluation

I use the per-paragraph average character-level BLEU score as evaluation metric.

I choose character-level BLEU rather than word-level BLEU for two reasons. Firstly, according to *Li et al. 2011*[6], character-level metrics correlate better with human assessment for evaluating Chinese translation. Secondly, the method and even the necessity of classical Chinese word segmentation is still an open question – I will come back to this later.

I manually performed paragraph alignment instead of using automatic sentence alignment algorithms, because the performance of such algorithms needs to be evaluated, and consequently the validity of our evaluation metric would become questionable – this is especially true in our case, and again I will come back to this later. The reason why I use BLEU score averaged over paragraphs rather than sentences is that I found the workload unacceptable having tried to manually align sentences of a volume.

In the calculation of BLEU, punctuation marks are treated indiscriminately, and are not included in n-grams.

1-Hour Baseline

Without a classical-modern Chinese dictionary, the best we can do in an hour is to replace/remove some most frequent words. I replaced 的 with 之, 说 with 曰, 到 with 至, 秦/楚/齐国 with 秦/楚/齐. I also removed the following words and characters: 了, 他们, 所以, 军队, 在, 他, 就, 是, 这. I also tried with other frequent words and characters but didn’t work out well.

The performance of this simple baseline is as follows:

	Dev	Test
1-BLEU	0.4221	0.4516
2-BLEU	0.0818	0.1076
3-BLEU	0.0119	0.0226
4-BLEU	0.0031	0.0060

Method

Sentence Alignment

Although I avoided sentence alignment in evaluation, it is still unavoidable for training the model. I use hunalign to perform sentence alignment.

There are two major challenges:

[6] Li, M., Zong, C., & Ng, H. T. (2011). Automatic evaluation of Chinese translation output: word-level or character-level?

1. Translation from classical Chinese to modern Chinese is often not literal. Sometimes the classical text is paraphrased; sometime additional contextual information is filled in. As a result, the length of classical sentences and corresponding modern sentences often doesn't correlate very well.
2. The original text doesn't have punctuation marks. There are different versions of punctuation by modern scholars that sometimes don't agree with each other, and a translation may adopt any of them, or even have its own version.

This put us in a dilemma. If we split the text into sentences only by periods, a one-to-one alignment wouldn't exist for many sentences, as a result many sentences need to be concatenated into overly-long ones, which choke the follow-up word alignment algorithm. If we treat commas and periods indiscriminately, the correlation of sentence length becomes much weaker, which severely harms the performance of sentence alignment algorithm.

I use a two-pass approach to solve the problem. In both passes, to utilize the fact that corresponding sentences very often share same characters, all characters are treated as single tokens. In the first pass, the bi-text is split by periods only, and the sentence alignment algorithm groups them into chunk pairs with quite high quality. In the second path, each chunk is separated by both commas and periods, and (sub-)sentence alignment is performed again within each chunk. If there are still overly-long sentences, they are simply abandoned.

By doing this, fewer sentences are abandoned, and the resulting sentences are shorter, which is friendlier to the word alignment algorithm.

Word Segmentation

The challenge of word segmentation is two-fold. For modern text, we can use an off-the-shelf word segmenter (e.g. Stanford Word Segmenter), but it does poorly when dealing with names of people, locations, ancient official positions, sacrificial vessels, legendary animals and plants, etc. For classical text, neither an off-the-self word segmenter nor any word-segmented text that can be used to train a segmenter is available.

I believe this problem can be solved by learning a vocabulary from the bi-text. The vocabulary can be used as part of the input data of a CRF-based modern Chinese segmenter like Stanford Word Segmenter to improve its performance. For classical Chinese, unlike modern Chinese, most words consist of only one character, thus ambiguity is not a big issue and simple forward maximum matching with the vocabulary would be reasonably good.

I implemented a fairly simple unsupervised algorithm to extract vocabulary. In short, for each 2-gram and 3-gram that appears in both classical and modern texts, we compute its frequency, boundary entropy and mutual information, and if they reach a certain threshold, we put the n-gram into the vocabulary[7]. This gives us a vocabulary with a few thousand words. I then manually remove some of them and add some new ones.

Word Alignment

Before performing word alignment, sentences that have more than 25 words are abandoned. In addition, if a modern sentence is more than 3 times longer than its classical counterpart, it is also abandoned since this often suggests aggressive paraphrasing that doesn't generalized very well.

[7] I got the idea from here: http://www.matrix67.com/blog/archives/5044 , but similar techniques are used in other more formal papers, too.

I implemented a modified IBM Model2 to perform word alignment: for a pair of words that share x common characters, an extra $0.9x$ pseudo-count is collected in the E-step.

Moses is then used to train a phrase-based model, decode and tune on dev data set.

Result

	Dev	Test
1-BLEU	0.5854	0.5636
2-BLEU	0.2011	0.1783
3-BLEU	0.0539	0.0421
4-BLEU	0.0154	0.0096

The model clearly outperforms the baseline, but not as much as expected. (I also tried to use GIZA++ instead of my implementation, and the result is similar.)

I guess part of the reason is that model doesn't know how to handle paraphrase. A random example is shown below. The model's translation is reasonably fine (and much better than the baseline translation), but there are many characters (in parentheses) that are translated literally but doesn't match the original text.

Modern text:

汉文帝后元七年，汉景帝即位，让冯唐去做楚国的丞相，不久被免职。汉武帝即位时，征求贤良之士，大家举荐冯唐。冯唐这年已九十多岁，不能再做官了，于是任用他的儿子冯遂做了郎官。冯遂字王孙，也是杰出的人才，和我友好。

Model translation:

（孝文后）七年，（汉）景帝立，使（冯）唐为楚丞相，（已而）免。（孝）武（皇）帝立（时），征求良士，（皆）任冯唐。（冯）唐是岁（已）九十余，不得仕之，乃用（其）子冯遂为郎。（冯）遂字王孙，亦（杰出）之士，与我善。

Baseline translation:

汉文帝后元七年，汉景帝即位，让冯唐去做楚之丞相，不久被免职。汉武帝即位时，征求贤良之士，大家举荐冯唐。冯唐年已九十多岁，不能再做官，于任用之儿子冯遂做郎官。冯遂字王孙，也杰出之人才，和我友好。

Original classical text:

七年，景帝立，以唐为楚相，免。武帝立，求贤良，举冯唐。唐时年九十余，不能复为官，乃以唐子冯遂为郎。遂字王孙，亦奇士，与余善。