CASIA Phrase-Based SMT System for IWSLT’07

Yu ZHOU
Institute of Automation
Chinese Academy of Sciences (CASIA)
yzhou@nlpr.ia.ac.cn

2007-10-15
Outline

- Overview
- Data
- Word Alignments
- Phrase Extraction
- Language Model
- Decoder
- Other Process
- Experiments
- Conclusion & Future Work
Overview

- Using the Phrase-Based SMT
- Using a log-linear model

\[ e^* = \arg \max_{e} \sum_{m=1}^{M} \lambda_m h_m(e, f) \]

Chinese-English translation task
Data

--Data Collection

- Downloading all the open resources from the web;
- Filtering these corpus which is highly correlative with the released train data by IWSLT 2007.
Data

--Data Preprocessing

- Chinese word segmenting using the software ICTCLAS3.0 (http://www.nlp.org.cn)
- Removing the noises words or characters in the training data
- Transforming the SBC case into DBC case in Chinese corpus
- Tokenizing the English words
Word Alignments

- Obtaining the initial word alignments by GIZA++ (http://www.fjoch.com/GIZA++.html)
- Using the method grow-diag-final method to modify the initial alignments
- Using our method to modify the word alignments by using the dictionary and jumping-distance
Dictionary:

- The bilingual dictionary download from the open resources on the web (http://isslt07.itc.it/menu/resources.html)
- From the bi-directional dictionaries generated by GIZA++, only using such word pairs with the highest probabilities.
Word Alignments (Cont.)

◆ If the word pair \((f_i, e_j)\) which is inexistent in the bilingual dictionary but existent in the word alignments of the sentence pair;

◆ If the word pair \((f_i, e_j)\) is inexistent both in the bilingual dictionary and the word alignments, we will not deal with such case;

◆ If the word pair \((f_i, e_j)\) is existent in the bilingual dictionary but inexistent in word alignments we will add the word pair alignment information;

◆ If the word pair \((f_i, e_j)\) is existent both in the bi-lingual dictionary and in the word alignments, we will keep the word pair alignment information.
Word Alignments (Cont.)

For example:

在中国哪个窗口预定？

Can I make a reservation?
Phrase-Extraction

- Och’s method:
  - Simple and easy to realized
  - Totally consistent with word alignments

- Extend Och’s method:
  - Using a flexible scale to extract the phrase pairs
Phrase-Extraction (Cont.)

\((\tilde{f}, \tilde{e}) \in BP \iff \forall f_i \in \tilde{f} : (f_i, e_j) \in A \rightarrow e_j \in \tilde{e} \)

AND \( \forall e_j \in \tilde{e} : (f_i, e_j) \in A \rightarrow f_i \in \tilde{f} \)
Phrase-Extraction

\[(\tilde{f}, \tilde{e}) \in BP \iff \forall f_i \in \tilde{f} : (f_i, e_j) \in A \rightarrow e_j \in \tilde{e} \]

\[\forall e_j \in \tilde{e} : (f_i, e_j) \in A \rightarrow f_i \in \tilde{f} \]

\[\{e_j \mid (e_j, f_i) \in A\} \geq \text{Threshold,} \]

\[\{e_j \mid e_j \in \tilde{e}\} \]

\[e_j \text{ is not a functional word,} \]

\[\rightarrow \arg \max_{\{f_i \mid (f_i, e_j) \in A\}} p(f_i \mid e_j) \in \tilde{f} \]
Phrase-Extraction (Cont.)

- Computing the percentage of consistent target words in $\hat{e}$;
- Judging if these non-consistent target words are functional words;
- Checking if the source words that the non-consistent and non-functional target word is aligned to are all outside $\hat{f}$;
- Extracting the phrases.
Phrase-Extraction (Cont.)

- For example:

```
it depends on the traffic conditions
```

```
那 要 看 交通 状况
```

```
it depends on the traffic conditions
```
Phrase-Extraction (Cont.)

Och’s phrase table

Our phrase table

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>那要看</td>
<td></td>
</tr>
<tr>
<td>那要看交通状况</td>
<td></td>
</tr>
<tr>
<td>交通</td>
<td></td>
</tr>
<tr>
<td>交通</td>
<td></td>
</tr>
<tr>
<td>交通状况</td>
<td></td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>那要看</td>
<td></td>
</tr>
<tr>
<td>那要看交通</td>
<td></td>
</tr>
<tr>
<td>那要看交通状况</td>
<td></td>
</tr>
<tr>
<td>要看交通</td>
<td></td>
</tr>
<tr>
<td>要看交通状况</td>
<td></td>
</tr>
<tr>
<td>看交通</td>
<td></td>
</tr>
<tr>
<td>看交通状况</td>
<td></td>
</tr>
<tr>
<td>交通</td>
<td></td>
</tr>
<tr>
<td>交通</td>
<td></td>
</tr>
<tr>
<td>交通状况</td>
<td></td>
</tr>
</tbody>
</table>
Phrase-Extraction (Cont.)

\[
\phi(\tilde{f} | \tilde{e}) = \frac{N(\tilde{f}, \tilde{e})}{\sum_{f'} N(\tilde{f}', \tilde{e})}
\]

\[
\phi(\tilde{e} | \tilde{f}) = \frac{N(\tilde{f}, \tilde{e})}{\sum_{\tilde{e}'} N(f, \tilde{e}')}
\]

\[
\text{lex}(\tilde{f} | \tilde{e}, a) = \prod_{i=i_1}^{i_2} \frac{1}{|\{j | (i, j) \in a\}|} \sum_{\forall (i, j) \in a} p(f_i | e_j)
\]

\[
\text{lex}(\tilde{e} | \tilde{f}, a) = \prod_{j=j_1}^{j_2} \frac{1}{|\{i | (i, j) \in a\}|} \sum_{\forall (i, j) \in a} p(e_j | f_i)
\]
Language Model

- Using the ngram-count tool in the open SRILM toolkit (http://www.speech.sri.com/projects/srilm)
- Training on the English part of the training data in GIZA++
- Using the 4-gram language model with Kneser-Ney smoothing method
Decoder

- Using the beam search algorithm which is similar to the Pharaoh decoder
- Adding the ‘expanding F-zerowords’ model
- Using a new tracing back method;
- Using the monotone search without any distortion and reordering model
Other Process

--Name Entities

- For the person name and location name, we translate them only by looking up its translations in the common phrase pair table.
- For the organization name, we translate them using the model based on a synchronous CFG grammar.
- For the number and date, we adopt the method based on the man-written rules to translate.
Other Process

--Post-processing

- Transforming the lowercase of the first character of the English words into uppercase
- Recombination the separated punctuations with its left closest English words
# Experiments

<table>
<thead>
<tr>
<th>Data</th>
<th>Chinese</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE_train</td>
<td>39,950</td>
<td>39,950</td>
</tr>
<tr>
<td>CE_sent_filtered</td>
<td>188,282</td>
<td>188,282</td>
</tr>
<tr>
<td>CE_dict_filtered</td>
<td>31,132</td>
<td>31,132</td>
</tr>
<tr>
<td>CE_newdev1</td>
<td>24,192</td>
<td>24,192</td>
</tr>
<tr>
<td>CE_newdev2</td>
<td>10,423</td>
<td>10,423</td>
</tr>
<tr>
<td>CE_test</td>
<td>489</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Chinese</td>
<td>English</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>DEV_train Sentences</td>
<td>283,556</td>
<td>283,556</td>
</tr>
<tr>
<td>Words</td>
<td>1,754,932</td>
<td>1,900,216</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>11,424</td>
<td>10,507</td>
</tr>
<tr>
<td>Average Length</td>
<td>6.2</td>
<td>6.7</td>
</tr>
<tr>
<td></td>
<td>Chinese</td>
<td>English</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>TST_train</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sentences</td>
<td>293,979</td>
<td>293,979</td>
</tr>
<tr>
<td>Words</td>
<td>1,890,984</td>
<td>2,051,619</td>
</tr>
<tr>
<td>Vocalbulary</td>
<td>11,661</td>
<td>11,273</td>
</tr>
<tr>
<td>Average Length</td>
<td>6.4</td>
<td>7.0</td>
</tr>
</tbody>
</table>
Experiments (Cont.)

<table>
<thead>
<tr>
<th>System</th>
<th>BLEU4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.2730</td>
</tr>
<tr>
<td>CASIA</td>
<td>0.3648</td>
</tr>
</tbody>
</table>

Baseline means the system with the base methods on word alignments and phrase extraction. The baseline system is only looking the name entities as the common words. CASIA means the system with the new methods described in our paper.
Conclusion & Future Work

- Using several new approaches in this our system: word alignments, phrase extraction, name entity identification and translation
- Adding the semantic information into our model
- Using non-consecutive phrase pair into our decoder
- Adding the reorder model into our decoder
- Re-ranking the N-best of the decoder
- Combining with other translation systems
Thanks!

谢谢！