**Machine Translation System: TSUKU**

**Motivation**
- Utilizing dependency information in large scale
- But also keep the tree structures in phrase level, which are extracted from CFG tree
- High speed decoding

**Tree Combining**
- Currently, we made a bunch of rules to combine dependency and CFG parse tree
- For sibling nodes in CFG parse tree, we determine head node according to dependency parse tree, and adjust the positions of chunks by dependency relationships of head nodes

```
[NP] the [VP] [NP] [NP] [NP]
```

- Replace the tags in CFG parse tree with tags of dependency heads

---

**Results**

<table>
<thead>
<tr>
<th>Systems / Auto Evaluation Scores</th>
<th>BLEU</th>
<th>NIST</th>
<th>RIBES</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSUKU-ej-int-1 (Proposed Model + Small LM with KenLM)</td>
<td>0.3141</td>
<td>8.126</td>
<td>0.7555</td>
</tr>
<tr>
<td>TSUKU-ej-int-2 (Proposed Model + Large LM with LSHLM)</td>
<td>0.319</td>
<td>8.1894</td>
<td>0.7565</td>
</tr>
<tr>
<td>TSUKU-ej-int-3 (Proposed Model + Large LM with KenLM)</td>
<td>0.3176</td>
<td>8.1769</td>
<td>0.7566</td>
</tr>
<tr>
<td>TSUKU-ej-int-1 uu ( + Word Penalty )</td>
<td>0.3306</td>
<td>8.0849</td>
<td>0.7242</td>
</tr>
<tr>
<td>BASELINE HPBMT</td>
<td>0.3298</td>
<td>8.0837</td>
<td>0.7231</td>
</tr>
<tr>
<td>BASELINE PBMT</td>
<td>0.3361</td>
<td>8.1816</td>
<td>0.7042</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Systems / Subjective Evaluation Scores</th>
<th>Adequacy</th>
<th>Acceptability</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSUKU-ej-int-1</td>
<td>2.7933</td>
<td>0.4088</td>
</tr>
<tr>
<td>BASELINE HPBMT</td>
<td>2.69</td>
<td>-</td>
</tr>
<tr>
<td>BASELINE PBMT</td>
<td>2.5333</td>
<td>-</td>
</tr>
</tbody>
</table>

**Translation Example**

**INPUT:** specifically, the ff amount is calculated at every second calculation timing based on an average value of fuel injection amounts, as will be described later.

```
[S] . → [X0].
|--specifically . [NP] . すなわち . [XO]
|--the ff amount [VP] .  FF 量 [XO]
|--is [NP] . は . [XO]
|--calculated at [NP] . で 算出 した .
|--every second calculation timing [NP] . → [X1] 第 2 の 演算 タイミング 毎
|--based on [NP] . → [X0] に基づいて
|--an average value of [NP] . → [XO] の 平均 値
|--fuel injection amounts . → 燃料 噴射 量
|--as will [NP] . → [XO] ように
|--be described later . 後述 する
```

**Conclusion**
- This tree-to-string translation model could achieve higher translation accuracy when handling long input sentences
- High decoding speed
- Producing about 1/5 translation rules comparing with Moses HPB

**LSH language model (LSHLM)** is a kind of lossy language model which has a high compression ratio and a low probability of information loss triggered by falsepositives. We used the LSH function to generate a group of similar ngrams(called a “bucket”). The LSH function can map similar ngrams to the same hash value. When a falsepositive occurs, similar ngrams are in same bucket and, therefore, incorrect value should be similar to correct one.

**Result of proposed model**

すなわち、FF量は、後述するように燃料噴射量の平均値に基づいて第2の演算タイミング毎で算出した。

**Result of Moses HPB**

具体的には、FF量を第2の演算タイミング毎に算出された燃料噴射量の平均値に基づいて、後述するようにされている。