**Language Analysis**

- **English**: Moses tokenizer & Irish parser
- **Japanese**: In-house tokenizer and POS tagger

**Tree-to-String Syntax-based SMT**

- Use parse tree
- Synchronous context-free grammar (SCFG) rules
- Char parsing decoder with cube pruning

**Rule Augmentation**

- Known as Syntax-Augmented Machine Translation (Zollmann and Venugopal, 2006)
- Extract more rules by modifying source parse tree
- Given a parse tree, produce additional nodes

**Handling OOV**

- **Handy on Word Split**
  - Example: nano-laminate → nano laminate

- **Spell Error Correction**
  - Use open-source spell checker, Aspell
  - Correction based on edit distance
  - Select one with the shortest distance among top-3 suggestions

**Neural Machine Translation**

1. Use target character sequence rather than word sequence
   - Removes the need to replace rare words with the unknown word symbol
   - Simpler than other methods recently proposed to address the same issue

2. Bi-directional representation of target characters
   - Better accuracy in preliminary experiment
   - E.g.,  

3. **Modified RNN encoder-decoder**
   - Bi-directional RNN with an attention mechanism (Bahdanau et al., 2015)

   \[ \begin{align*}
   h_t &= \text{f}_t(\text{h}_{t-1}, \text{h}_{t+1}) \\
   \text{c}_t &= \sum_{i=1}^{T} \alpha_i \text{c}_{t,i} \\
   \text{c}_{t,i} &= \text{f}_t(\text{h}_{t,i}, \text{h}_{t+1})
   \end{align*} \]

   - New hidden state of the decoder
   - Prob. of next target word

**Experimental Results**

- **En-Ja SMT**
  - BLEU: 31.34 (290M)
  - #Rules: 19.5K

- **Ko-Ja SMT**
  - BLEU: 70.31 (57M)
  - #Rules: 55M

- **Effect of combination between T2S/PBMT and NMT**
  - NMT outperforms T2S in En-Ja, while it does not outperform PBMT in Ko-Ja
  - NMT reranking give a great benefit between T2S/PBMT and NMT

**Ko-Ja Technical Terms**

- Word-level PBMT
- Char-level tokenization
- 10-gram LM, max-phrase-len=10

**Observation**

- Both Ko-Ja technical terms are usually transliterated from the same foreign word
- Char-level PBMT implicitly learns transliteration rules

**Japanese Machine Translation System for WAT 2015**

Hyoong-Gyu Lee1, Jaesong Lee1, Jun-Seok Kim2 and Chang-Ki Lee2

1NAVER LABS, NAVER Corp. and 2Kangwon National University