### Patents are easy to translate

They contain lots of repetitive and formulaic text ⇒ train a model of **sparse lexicalized features** on a large data set using **multi-task learning**; incorporate \( \ell_1 / \ell_2 \) regularization to find most important features

<table>
<thead>
<tr>
<th>Feature selection</th>
<th>dev1</th>
<th>dev2</th>
<th>dev3</th>
<th>dev1,2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>27.76</td>
<td>28.71</td>
<td>–</td>
<td>–</td>
<td>28.92</td>
</tr>
</tbody>
</table>

#### Rule identifiers: unique rule identifier

- **Rule n-grams**: bigrams in source and target side of a rule, e.g., of \( X_1, X_1 \) requirements
- **Rule shape**: 39 patterns identifying location of sequences of terminal and non-terminal symbols, e.g., (for rule (1))
  - \( NT, term* \), \( NT \), \( NT, term* \), \( NT \)

**Feature selection**

- \( X \to \text{ 光線のとき, } X_1 \): this time, the \( X_1 \) is
- \( X \to \) テキスト メモリ 41 に \( X_1 \) in the text memory 41

#### Gold standard ranking:

- **BLEU+1 scores of translations of kbest lists**

### Multi-task learning, \( \ell_1 / \ell_2 \) regularization and parallelization

#### Data

- Single dev, sparse features
- Single dev, dense features
- Multi-task, sparse features

#### Systems & results:

- **Constrained setup for both JP-EN and ZH-EN subtasks**:
  - **Marton & Resnik’s (2008) soft-syntactic constraints**:
    - \( \{ADJP,ADV,VP,CP,NNP,DT,WP,TO,PP,PP,WR,VP\} \times \{=,+,\} \)
  - **Indicate if spans in decoder derivations**
  - **Match or cross** constituents of syntactic trees
  - **Weights may be tied** (marker: ‘2’) or set independently (marker: ‘\( \)’)
  - **IP2 VP2 NP**, 5 features, \( NP \) tied, IP/VP independent; \( VP2 \) (20 features)

#### Results on dev:

- **34.06 (baseline) → 34.57 (IP2 VP2 NP)** → **34.84 (XP2)**

#### Effects of soft-syntactic constraints

<table>
<thead>
<tr>
<th>baseline</th>
<th>Another option</th>
<th>reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Another option is coupled to both ends of . . . , thereby allowing . . .</td>
<td>XP2</td>
<td>Another alternative is to couple the ends of . . . , thereby allowing . . .</td>
</tr>
<tr>
<td>( XP2 )</td>
<td>Another further option is to optically couple both ends 10 of . . . , thus allowing . . .</td>
<td></td>
</tr>
</tbody>
</table>

### Preprocessing (JP-EN only)

- **JP**: Full-width-latin characters converted to their standard UTF-8 equivalents
- **JP**: **Katakana term splitting** (RWTW NTCIR9) w/ compound splitter (Koenig/Knights, 2003)
- **EN**: Customized tokenizer (avoid splitting of FIG. or PAT. . .)
- **both**: **Consistent tokenization** (BBN NTCIR9): training data aligned using regular expressions; for test/dev sources applied the most common variants

### SMT setup:

- **cdec SCFG decoder** (Dyer, 2010): Hiero grammars (2 non-terminals max, . . .) built w/ impl. of the suffix array extraction technique of (Lopez, 2007); Sgram modified Kneser-Ney smoothed LM built w/ SRILM; lowercased models; high values for cube pruning pop limit (500) and span size limit (100) at test time; Chinese segmentation w/ Stanford Segmenter, Japanese w/ MeCab; parses w/ Stanford Parser; English tokenizing/recasing/truecasing w/ Moses tools

---

The HDU Discriminative SMT System for Constrained Data PatentMT at NTCIR10

Institute for Computational Linguistics, Heidelberg University, Germany
Patrick Simianer, Gesa Stupperich, Laura Jehl, Katharina Wäschle, Artem Sokolov, Stefan Riezler

\{simianer,stupperich,jehl,waeschle,sokolov,riezler\}@cl.uni-heidelberg.de