BBN's Systems for the Chinese-English Sub-task of the NTCIR-10 PatentMT Evaluation

Zhongqiang Huang, Jacob Devlin, Spyros Matsoukas, and Rich Schwartz

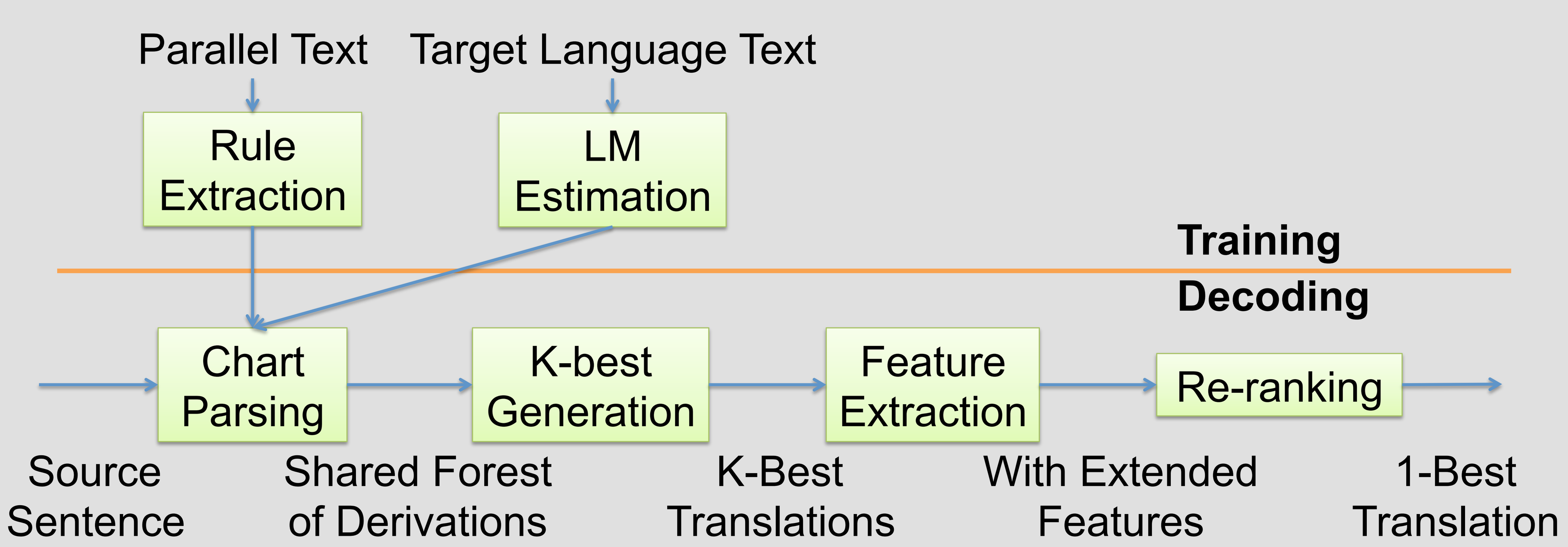
Raytheon BBN Technologies

Introduction

- BBN's statistical translation system for Patent MT
  - Initially developed for newswire, and later for broadcast news, web forums, etc.
  - Best performing system in MT evaluations under DARPA's GALE, BOLT, and other MT-related programs
  - All techniques initially developed for other domains work well on patents
  - Special handling for patents helps
- Lots of potential
  - Patents are easier to translate
  - State-of-the-art accuracies in both automatic and manual evaluations
  - Helpful in real patent examination and possibly other tasks

Statistical Machine Translation

- Translation framework



- String-to-dependency hierarchical translation model
  - Extract only hierarchical rules with well-formed dependencies on the target side:

X_L : X_1 出发 去 X_2 → VB_L : NR_1 leaves for NN_2

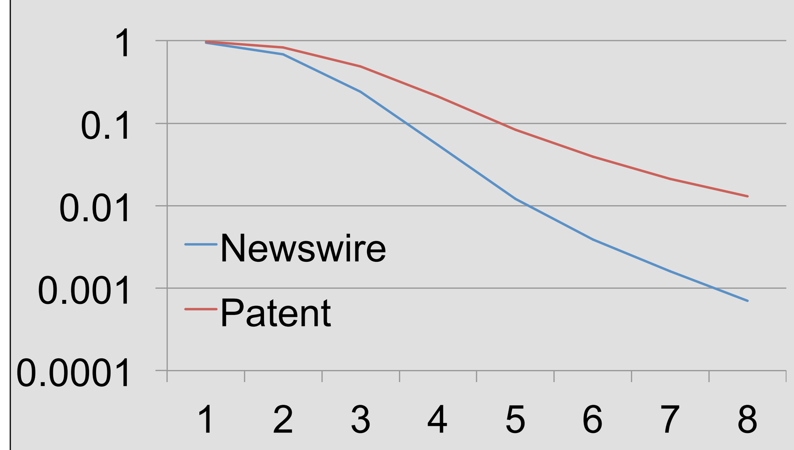
  - Use POS tag of head word as non-terminal labels on the target side
  - Extract all phrasal rules, ignoring dependency
- Features:
  - 10+ core features
  - ~50K sparse binary features

Application to Patent MT

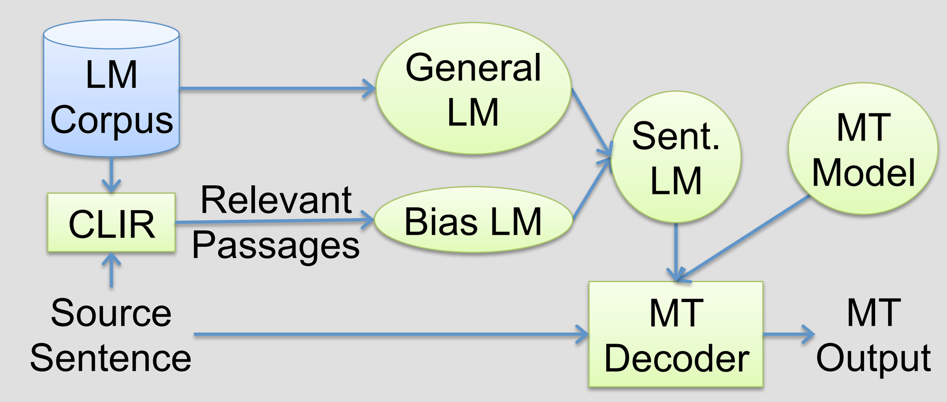
- Data preparation
  - Parallel data: 45M words of Chinese-English sentence pairs
  - Extra LM data: 14B words of US patents in English
  - Development data: 2K Chinese-English sentence pairs; split into tuning and test sets
- Model training
  - Translation model: trained on the 45M parallel corpus
  - Language models:
    - 45M LM: trained on the target side of the 45M parallel corpus
    - 14B LM: trained on the 45M LM data plus 14B English patents
- Addressing issues related to patent data (NTCIR-9)
  - Consistent tokenization of ASCII strings in source and target
    - e.g., "IS-1000" vs. "IS – 1000"
  - Special token sharing in translation and language model
    - One special token for each category: numbers (e.g., 2,596), patent IDs (e.g., No.5,400,788), math expressions (e.g., p=0.004), material names (e.g., C15H23N2O5P), and labeled names (e.g., 3.05kg)
  - Patent case-LM
    - Retrain the case-LM on 45M LM data
  - Word segmentation lexicon
    - Re-optimize on 45M parallel corpus
  - Use only 100 features of the highest weights in each category of the 50K sparse features
    - Address over-fitting due to smaller tuning set
  - Document-level CLIR-based LM adaptation
    - Retrieve most relevant passages for a test document in the 14B LM data using CLIR
    - Bias LM for sentences in the test document to these passages

Recent Advances

- Miscellaneous features
  - Model target bigrams given source and vise versa
  - Trait features: model general properties of translation hypotheses, e.g., *percent of words that re-order*
- Sentence-level LM adaptation instead of document-level
  - Patent documents tend to use well-structured sentences and re-use n-grams in other patent documents



Percentage of test source n-grams observed in training



Sentence-level LM adaptation

- Robust context-dependent modeling
  - Sparse high-order context-dependent translation models

$$P\left(t_{s_i}, t_{s_{i-1}} \mid s_i, s_{i-1}, s_{i+1}, s_{i-2}\right)$$

  - Solution: interpolate all possible back-off components
    - Sparse context types are added independently of each other

$$P\left(t_{s_{i-1}} \mid t_{s_i}, s_i, s_{i-1}, s_{i+1}, s_{i-2}\right)$$
$$= \omega_0 P\left(t_{s_{i-1}} \mid t_{s_i}, s_i, s_{i-1}, s_{i+1}, s_{i-2}\right) + \omega_1 P\left(t_{s_{i-1}} \mid t_{s_i}, s_i, s_{i-1}, s_{i+1}\right) + \cdots + \omega_{30} P\left(t_{s_{i-1}} \mid t_{s_i}\right)$$

  - Each weight ω_j is a function of the marginal count

$$\omega_j P\left(t_{s_i} \mid s_i, s_{i-1}\right) = \frac{1}{Z} \alpha_j \log\left(C\left(s_i, s_{i-1}\right)\right) \frac{C\left(t_{s_i}, s_i, s_{i-1}\right)}{C\left(s_i, s_{i-1}\right)}$$

  - Weights α are optimized to maximize likelihood on a held-out set
    - Least useful components are thrown out for efficiency
- Recurrent neural net LM for rescoring
  - Trained on 45M LM data, interpolated with 5-gram LM
- True-casing is treated as a translation problem
  - Trained on 45M LM data, use rule probabilities, true-case LM probability, and sparse features, e.g., *is the word upper cased and a proper noun?*

Results

- On development set

Development in NTCIR-9	BLEU
Initial system with 45M LM	34.01
+ consistent tokenization	34.56
+ more token sharing	34.97
+ patent case-LM	36.47
+ optimized word segmenter	36.95
+ top 100 features	37.71
+ 14B LM	39.14
+ document-level LM adaptation	40.04

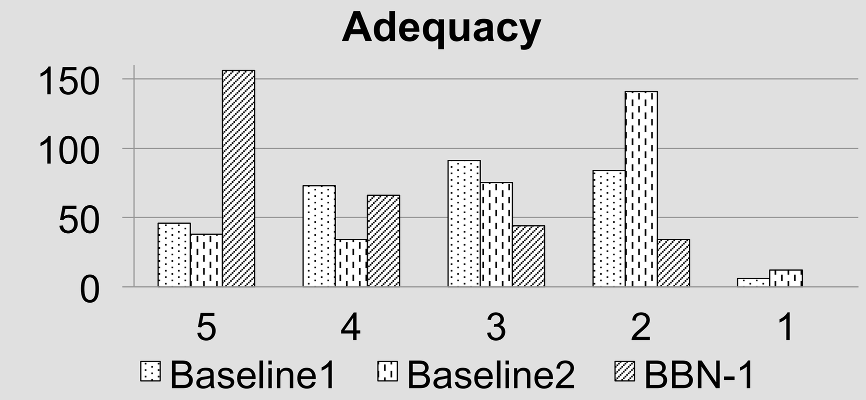
Development in NTCIR-10	BLEU
NTCIR-9 system with 45M LM	37.71
+ miscellaneous features	38.06
+ robust context dep. translation	38.72
+ recurrent neural network LM	39.35
+ translation-based true caser	40.02
NTCIR-9 system with 14B LM	39.14
+ miscellaneous features	39.51
~~+ document-level LM adaptation~~	~~39.94~~
+ sentence-level LM adaptation	40.95
+ robust context dep. translation	41.09
+ recurrent neural network LM	41.43
+ translation-based true caser	42.13

- Official evaluation

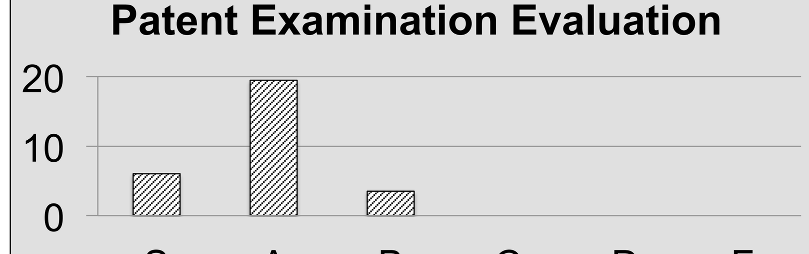
Automatic Evaluation (BLEU)

System	IE	CE	ME
Baseline1	32.52	30.74	17.96
Baseline2	31.34	29.34	18.05
BBN-1	42.68	39.44→41.09	27.62
BBN-2	39.98	36.69→38.93	N/A

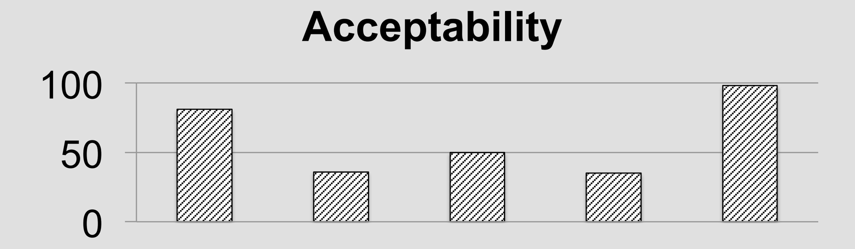
Adequacy



Patent Examination Evaluation



Acceptability



- Typical example

Source: 对于每一像素，着色引擎210使用在以上等式(2)-(4)中陈述的边等式来确定所述像素是否在三角形中。

MT output: For each pixel, the rendering engine 210 uses the edge equation set forth in equations (2) to (4) above to determine whether the pixels in a triangle.

Reference: For each pixel, the shading engine 210 determines whether the pixel is in the triangle using the edge equations set forth in equations (2) - (4) above.