

SubMerge: Merging Equivalent Subword Tokenizations for Subword Regularized Models in Neural Machine Translation



Haiyue Song¹, Francois Meyer², Raj Dabre¹, Hideki Tanaka¹, Chenhui Chu³, Sadao Kurohashi^{3,4}

¹NICT, Japan ²University of Cape Town, South Africa ³Kyoto University, Japan ⁴NII, Japan

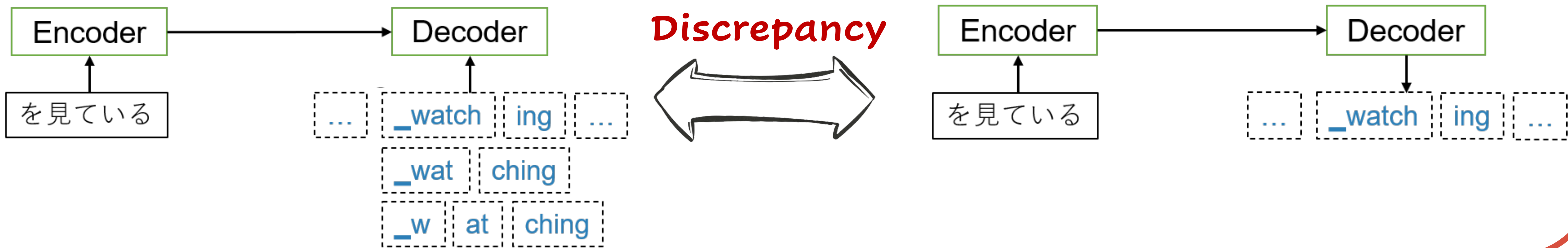
Background

Stochastic subword tokenization, e.g. BPE-dropout, is a strong data augmentation method in NMT.

However, it suffers from **training-inference discrepancy**

Training

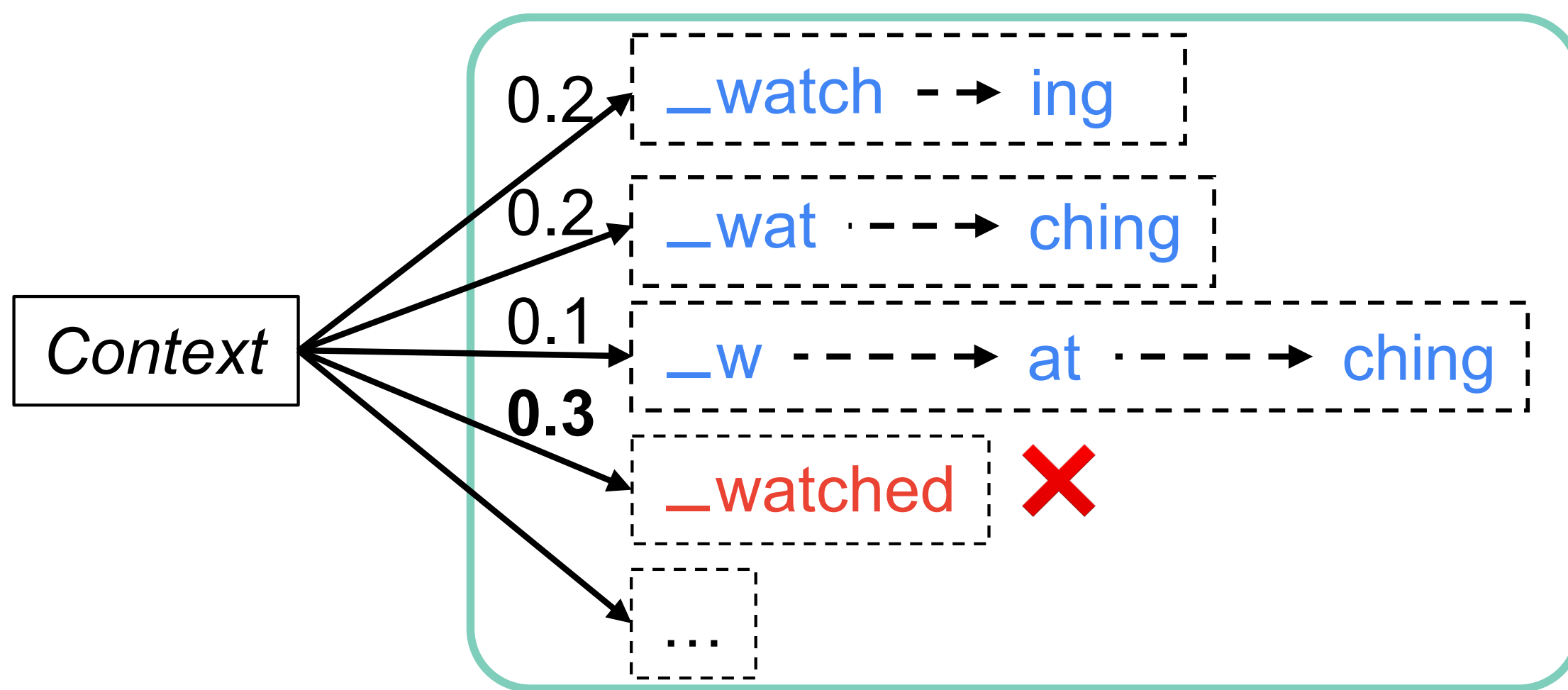
Inference



Overview

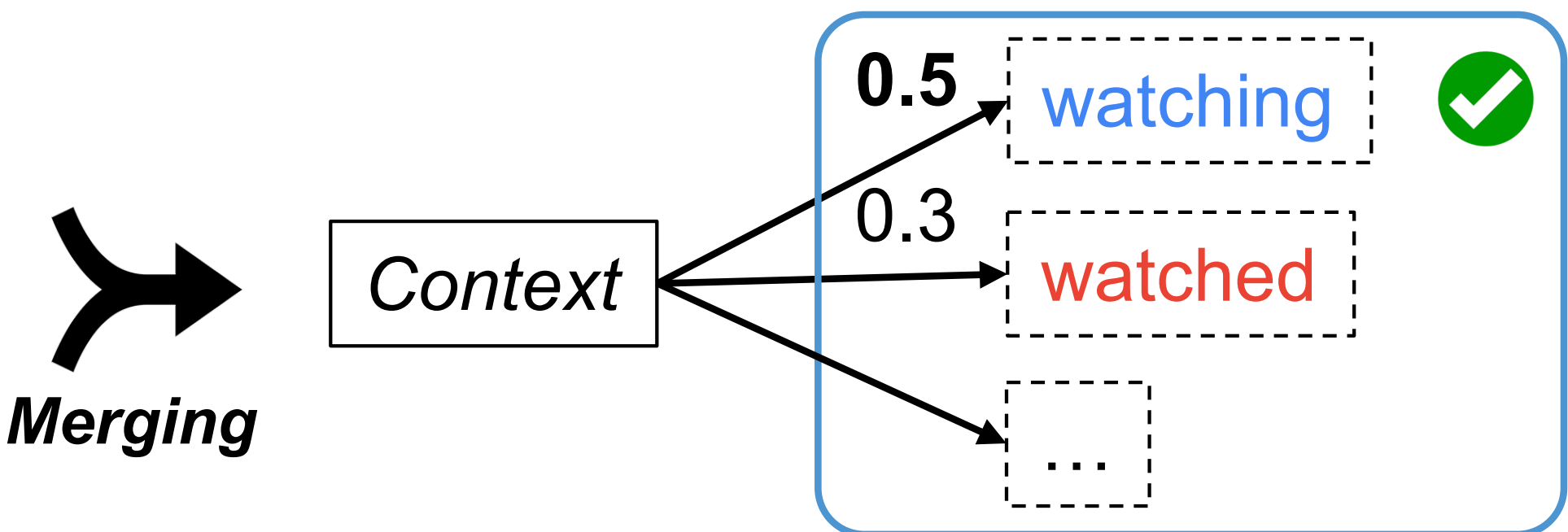
Previous

Selecting one tokenization



Proposed

Recovering word-level probability



Method

SubMerge consists of

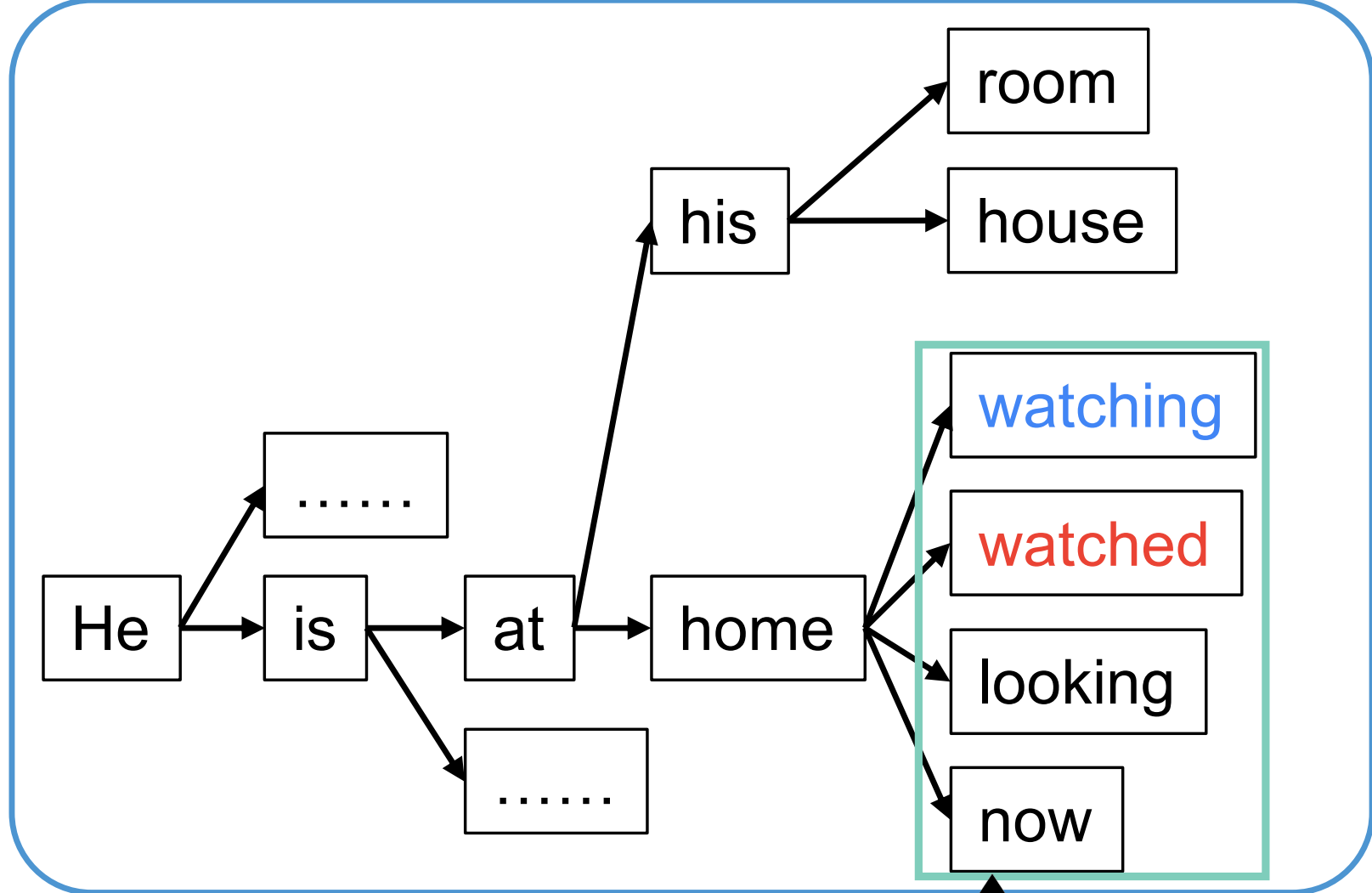
■ **Outer beam search**

- which views words as minimal units.

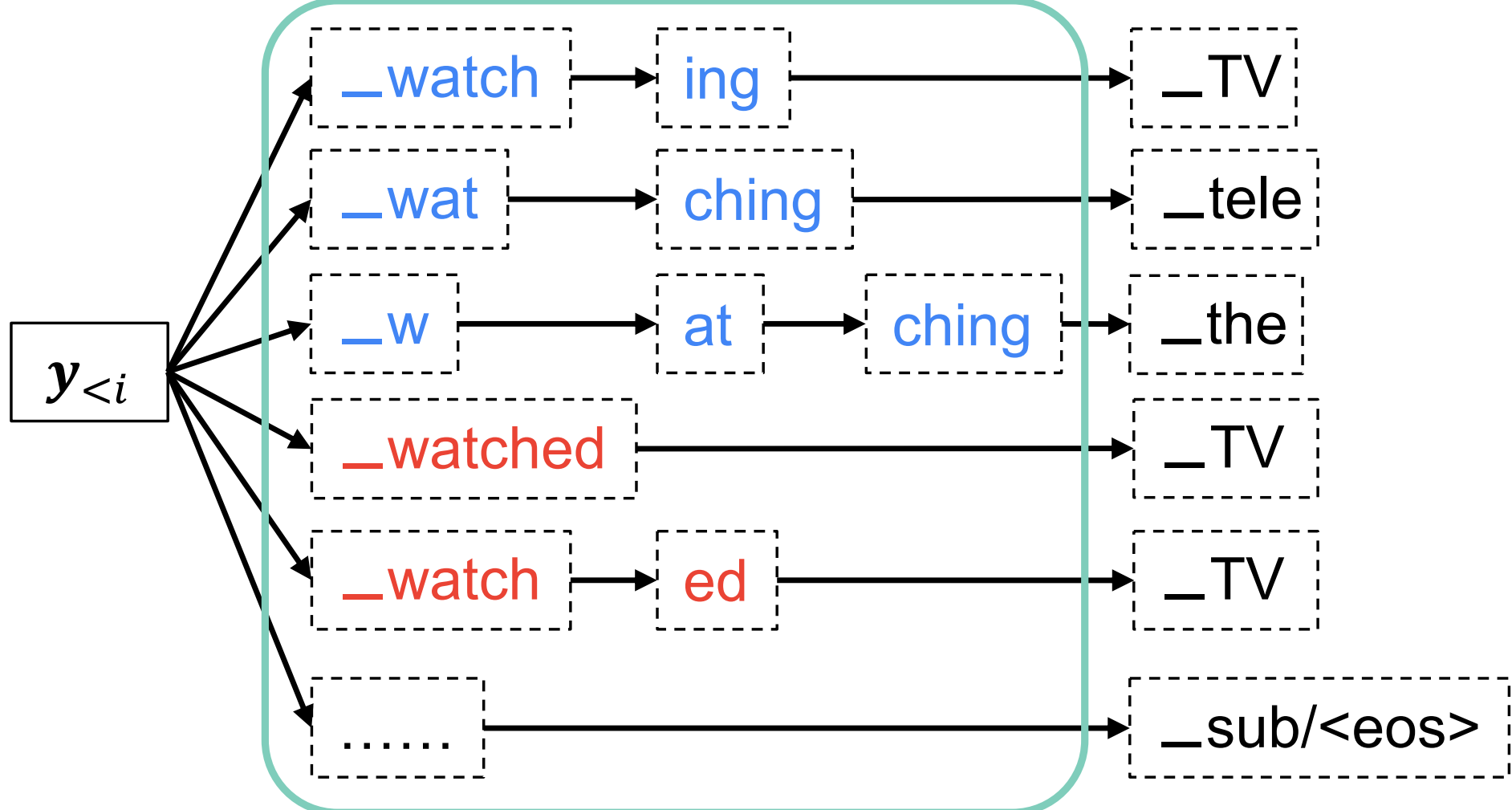
■ **Inner beam search**

- which returns the n -best list of words and their probabilities merged from multiple tokenizations.

Outer beam search



Inner beam search



Merge

Some math:

$$\arg \max_Y P_\theta(Y|X) \quad \text{Tokenizer} \quad \tau_S(X) = \mathbf{x}, \quad \tau_T(Y) = \mathbf{y} \Rightarrow P_\theta(Y|X) = \sum_{\mathbf{x} \in \mathcal{V}_S(X)} \sum_{\mathbf{y} \in \mathcal{V}_T(Y)} P_\theta(\mathbf{y}|\mathbf{x}) P_{\tau_S}(\mathbf{x}|X)$$

Models with deterministic tokenizer $P_\theta(Y|X) \approx P_\theta(\mathbf{y}|\mathbf{x})$ **Models with stochastic tokenizer** **Challenge** $P_\theta(Y|X) \neq P_\theta(\mathbf{y}|\mathbf{x})$

Approximation: good encoder $P_\theta(Y|X) \approx \sum_{\mathbf{y} \in \mathcal{V}_T(Y)} P_\theta(\mathbf{y}|\mathbf{x})$ Introduce word tokenizer $\tau_W(Y) = \mathbf{w}$ Gather sentence probability word-by-word $P_\theta(w_i|w_{<i}, \mathbf{x}) = \sum_{\mathbf{y}' \in \mathcal{V}_T(w_i)} P_\theta(\mathbf{y}'|w_{<i}, \mathbf{x})$

$$P_\theta(Y|X) = P_\theta(\mathbf{w}|\mathbf{x}) = \prod_{i=1}^n P_\theta(w_i|w_{<i}, \mathbf{x})$$

Results

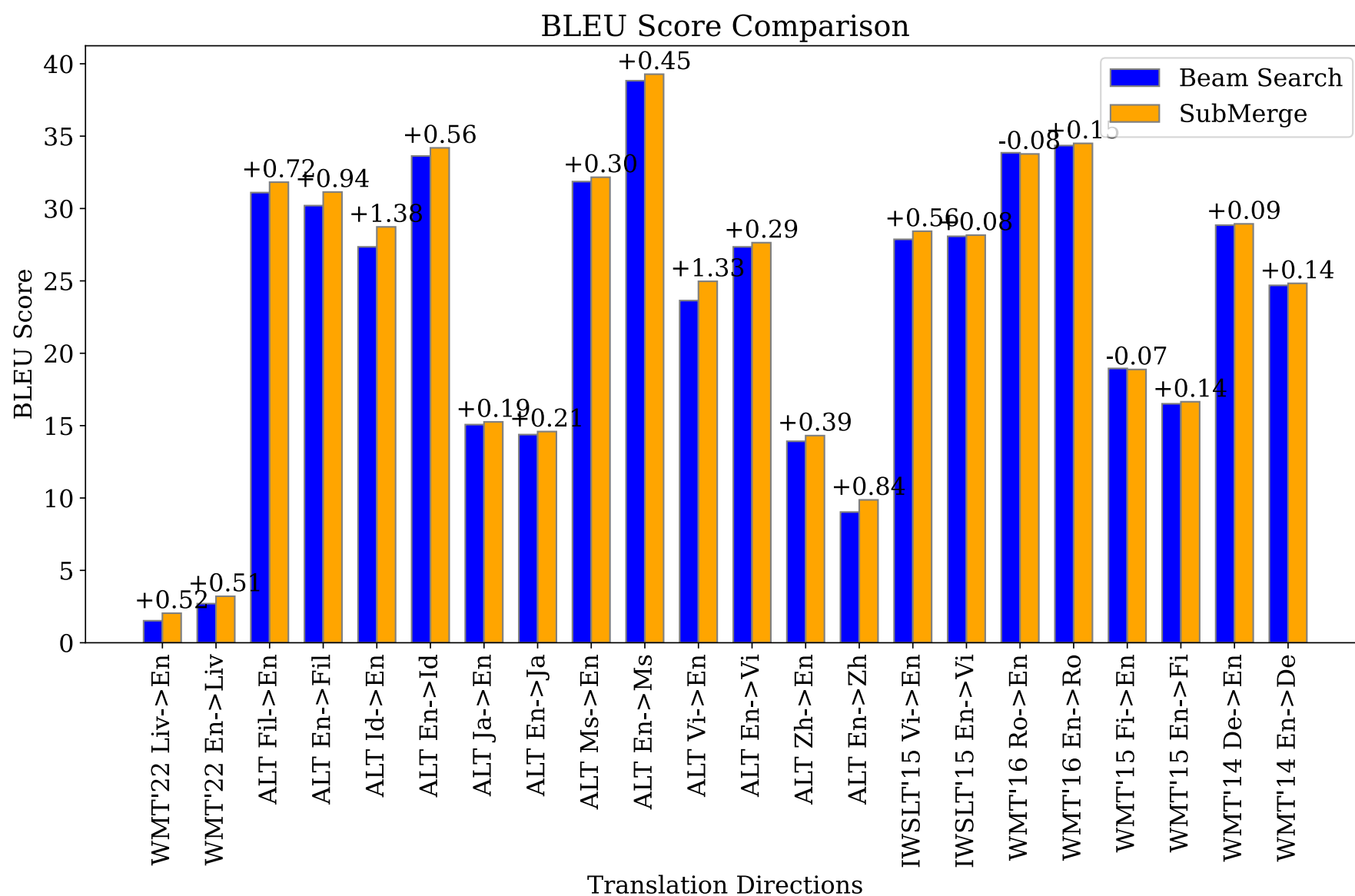
Translation quality

■ **Low-resource**

- Up to 1.3 BLEU $\uparrow$ improvement

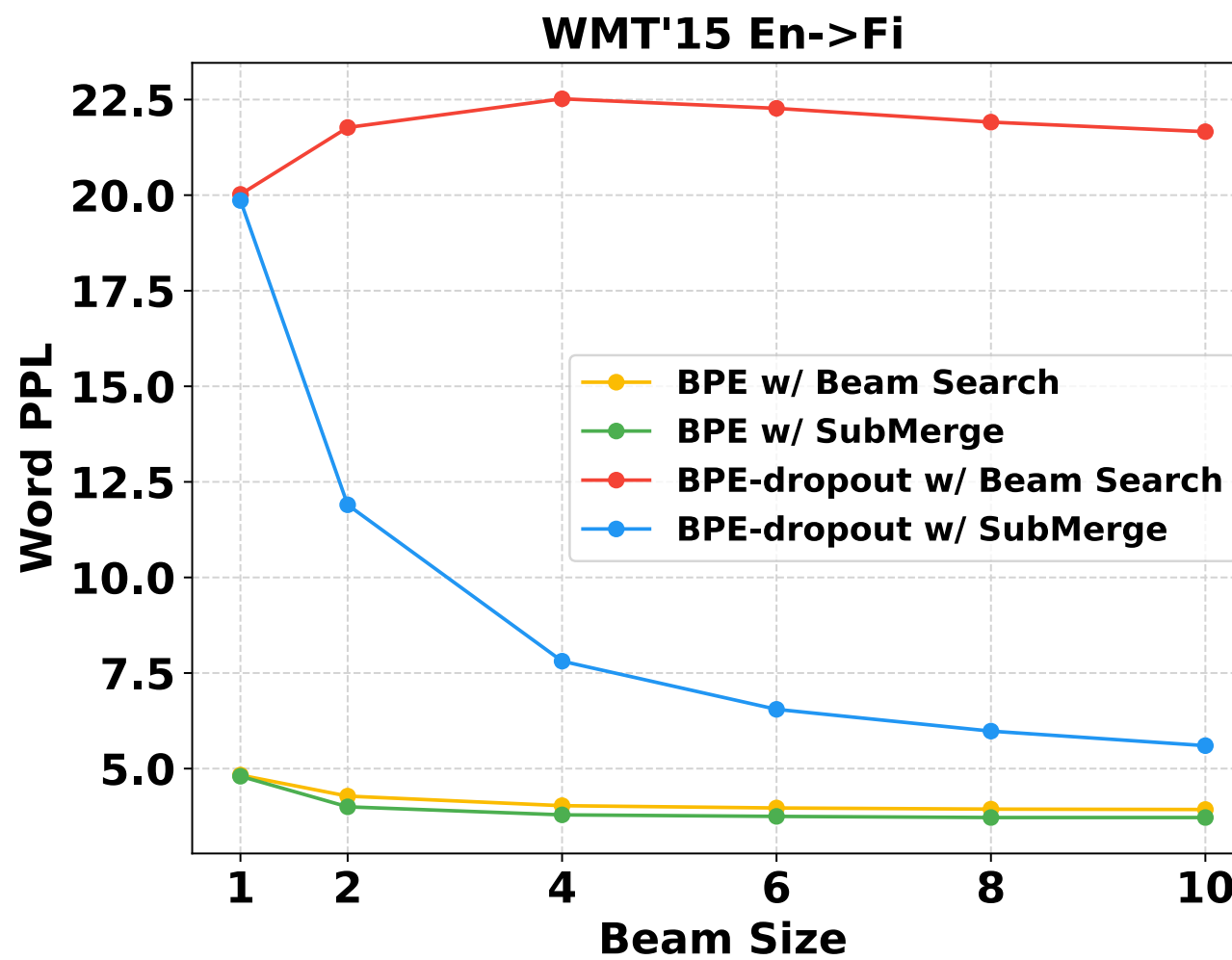
■ **High-resource**

- Comparable with Beam Search



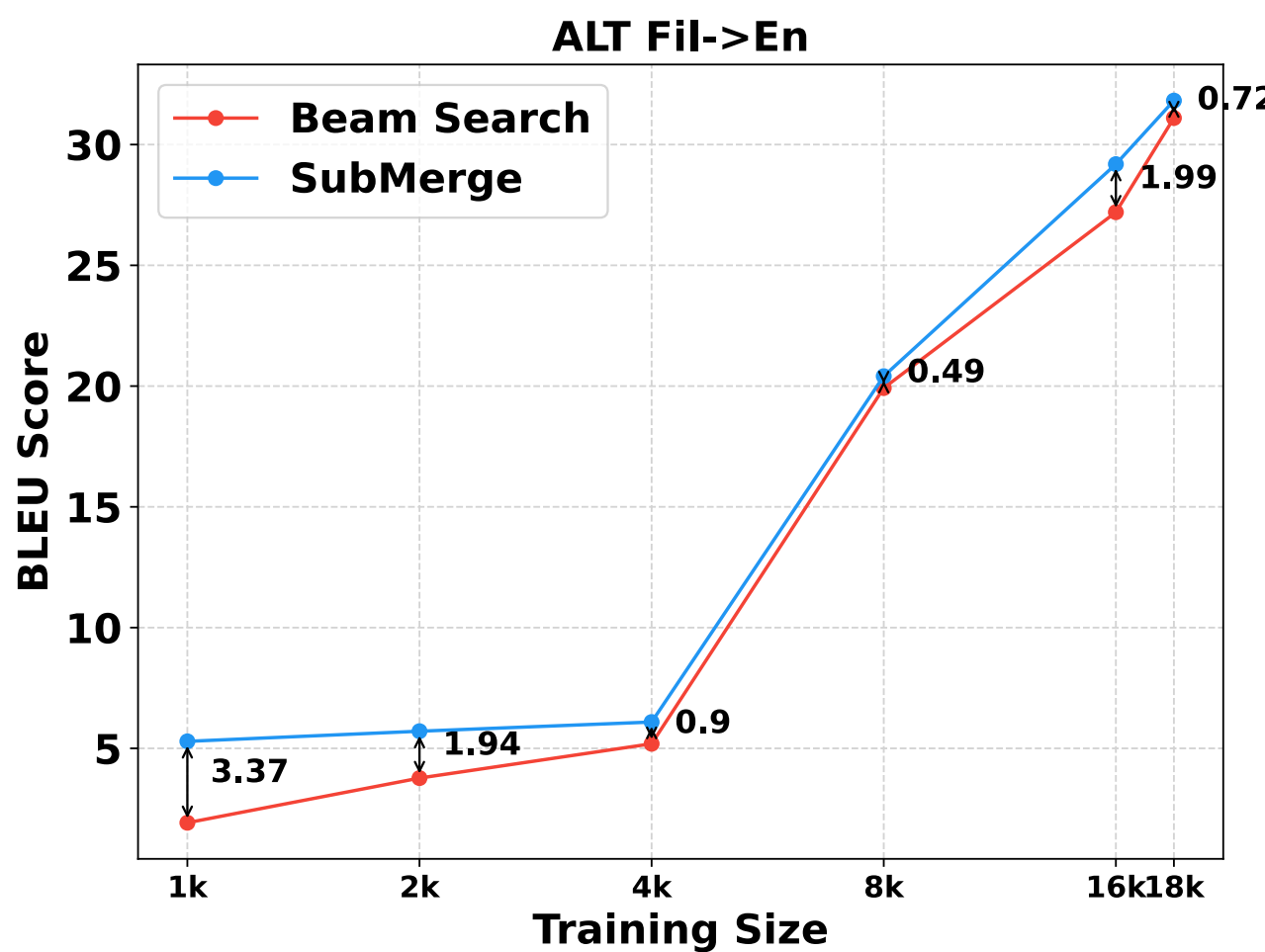
Perplexity

- Stochastic tokenization *enlarges* word perplexity
- SubMerge recovers the correct word perplexity, approaching the non-stochastic method.



Analysis: very-low-resource

- The improvement is large having 1k data, showing the potential to apply it in the decoding of fine-tuned LLMs with little in-domain data.



Analysis: comparing with non-subword-based model

- Although theoretically using word-based model gives 100% accuracy word PPL estimation, the quality is not as good as using BPE-dropout with SubMerge

Models	Vi→En	En→Vi	Ro→En	En→Ro
Char-based	24.72	27.04	30.45	29.82
Word-based	21.40	25.24	26.33	25.67
Subword-based	28.43	28.16	33.77	34.50

BLEU scores