Tackling Intrinsic Uncertainty with SCONES

Felix Stahlberg

@ MT Marathon | Sep 8, 2022

Google Research

Exact inference in NMT is impossible.

Exact inference in NMT is impossible.

Wrong!

(Stahlberg and Byrne, 2019)

Monotonicity of NMT model scores

NMT left-to-right factorization:

$$\log P(\mathbf{y}|\mathbf{x}) = \sum_{j=1}^{J} \underbrace{\log P(y_j|y_1^{j-1},\mathbf{x})}_{<0}$$

NMT scores are monotonically decreasing:

$$\forall j \in [2, J] : \log P(y_1^{j-1}|\mathbf{x}) > \log P(y_1^{j}|\mathbf{x})$$

Exact decoding for NMT

- 1.) Run beam search
  - $\gamma = P(\mathbf{y}_{\text{beam}}|\mathbf{x})$ is a lower bound on the global best score: $\gamma \leq \log P(\hat{\mathbf{y}}|\mathbf{x})$
- 2.) Run depth-first search
  - Prune if a partial hypothesis score exceeds γ
  - Update γ if a better complete hypothesis is found
  - Child nodes are ordered such that EOS is expanded first

$\mathbf{y}_{\text{beam}}$: beam hypothesis
$\hat{\mathbf{y}}$: global best hypothesis
γ : lower bound

Empty translations

Search	BLEU	Ratio	#Search errors	#Empty
Greedy	29.3	1.02	73.6%	0.0%
Beam-10	30.3	1.00	57.7%	0.0%
Exact	2.1	0.06	0.0%	51.8%

Search error: decoder returns a hypothesis with a lower likelihood than that found by exact inference

In the absence of search errors, NMT often prefers the empty translation.

But Why?

- "Long sentences sum over more log-probabilities (which are negative), so they result in lower scores"
  - **But:** The left-to-right factorization is correct.
- "It doesn't really matter - we use small beams / length normalization in practice"
  - **But:** Length normalization is a remedy, not a cure
  - **But:** What is the head room? Is beam search obscuring other model errors?
- "Just train bigger models longer and on more data"
  - **But**: Problem is reduced, but not solved

Model uncertainty

- The neural model cannot decide which output is correct
- Example:
  - `en`: The pixel will receive updates until October 2026.
  - `de1`: **Der** Pixel wird bis Oktober 2026 Updates erhalten.
  - `de2`: **Das** Pixel wird bis Oktober 2026 Updates erhalten.

Model uncertainty

- The neural model cannot decide which output is correct
- Example:
  - `en`: The pixel will receive updates until October 2026.
  - `de1`: **Der** Pixel wird bis Oktober 2026 Updates erhalten.
  - `de2`: **Das** Pixel wird bis Oktober 2026 Updates erhalten.

Intrinsic uncertainty

- The same input has multiple acceptable outputs
- Example:
  - `de`: Das Pixel wird bis Oktober 2026 Updates erhalten.
  - `en1`: The pixel will receive updates until October 2026.
  - `en2`: The pixel phone gets updates till 10/2026.

Google

Model uncertainty

- The neural model cannot decide which output is correct
- Example:
  - `en`: The pixel will receive updates until October 2026.
  - `de1`: **Der** Pixel wird bis Oktober 2026 Updates erhalten.
  - `de2`: **Das** Pixel wird bis Oktober 2026 Updates erhalten.

Intrinsic uncertainty

- The same input has multiple acceptable outputs
- Example:
  - `de`: Das Pixel wird bis Oktober 2026 Updates erhalten.
  - `en1`: The pixel will receive updates until October 2026.
  - `en2`: The pixel phone gets updates till 10/2026.

Softmax models spread out probability mass across multiple candidates (Eikema and Azis, 2020)

Intrinsic uncertainty in softmax models

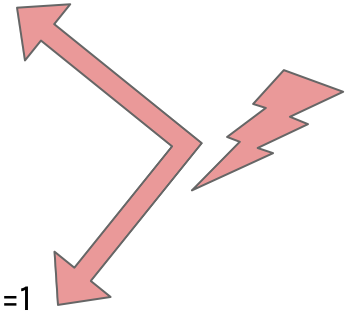
- Conventional NMT learns a distribution P(**y**|**x**) over all translations **y** given the source sentence **x**.
- Thus, it cannot represent intrinsic uncertainty
- Intrinsic uncertainty in the training data leads to contradictions since there is a built in assumption that there is exactly one “correct” translation for a source sentence.

Training data

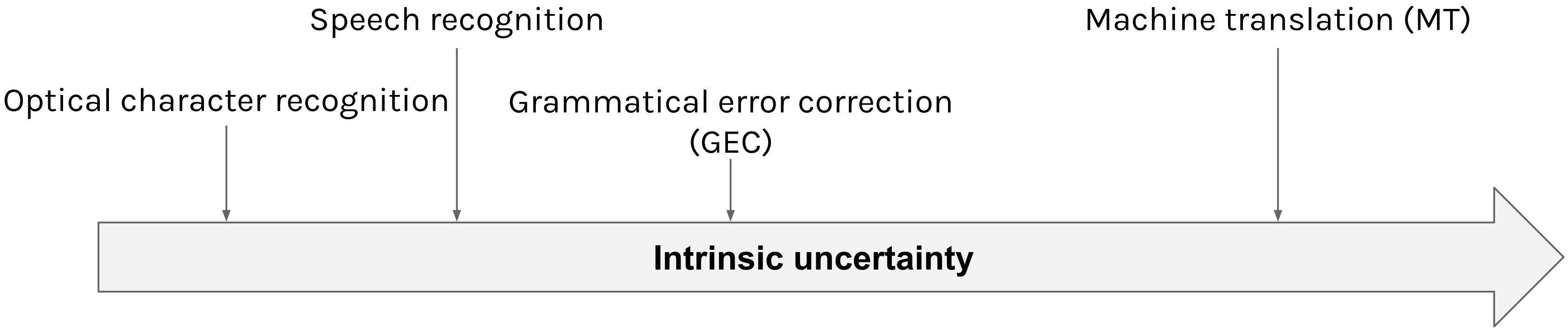
de1	en1
de2	en2
de3	en3
…	
de1	en243

P(en1|de1)=1

P(en243|de1)=1



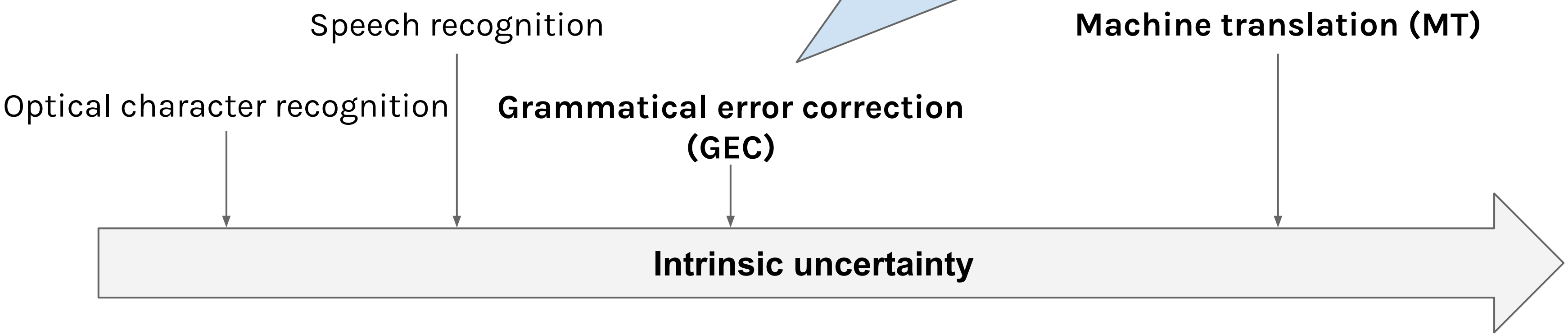
Intrinsic uncertainty of NLP tasks



Google

Intrinsic uncertainty of NLP tasks

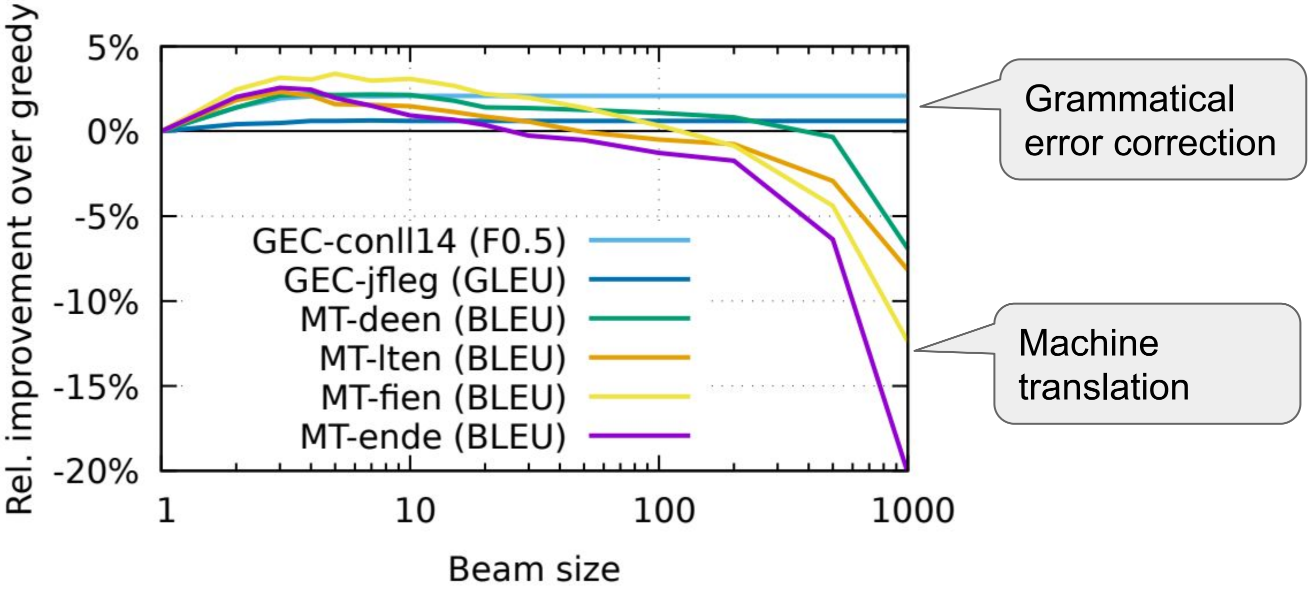
Felix Stahlberg, Ilia Kulikov, and Shankar Kumar. 2022. [Uncertainty determines the adequacy of the mode and the tractability of decoding in sequence to-sequence models](). ACL



Google

Beam search curse (Koehn and Knowles, 2017)

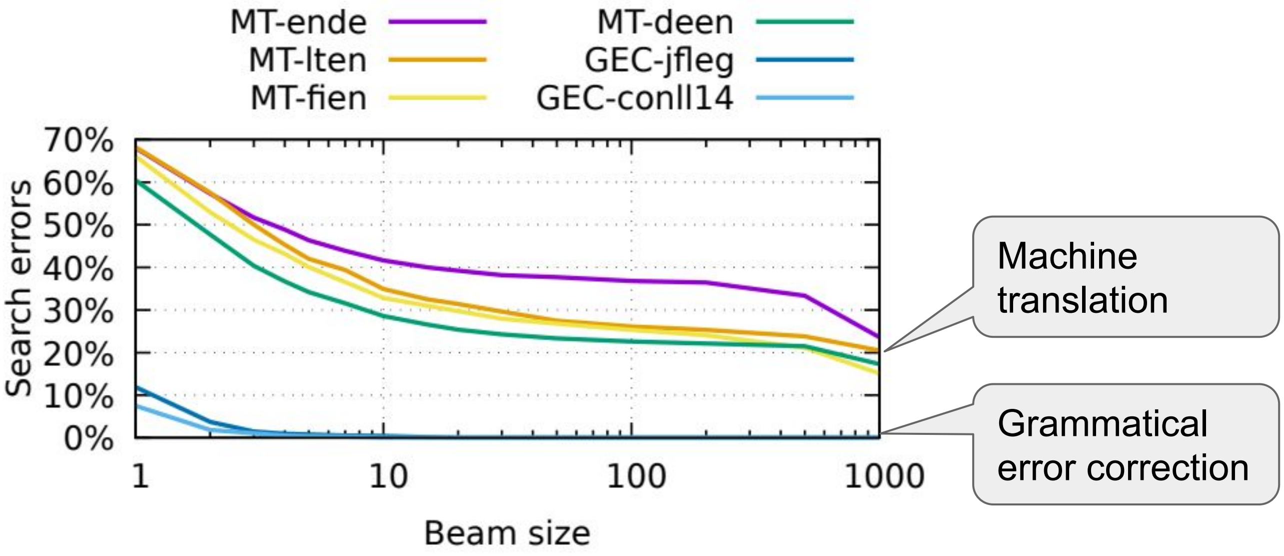
- MT quality degrades for large beam sizes, **but GEC quality saturates**.



High resource: MT-deen (German-English) MT-ende (English-German)
Medium resource: MT-fien (Finnish-English)
Low resource: MT-lten (Lithuanian-English)

High number of beam search errors

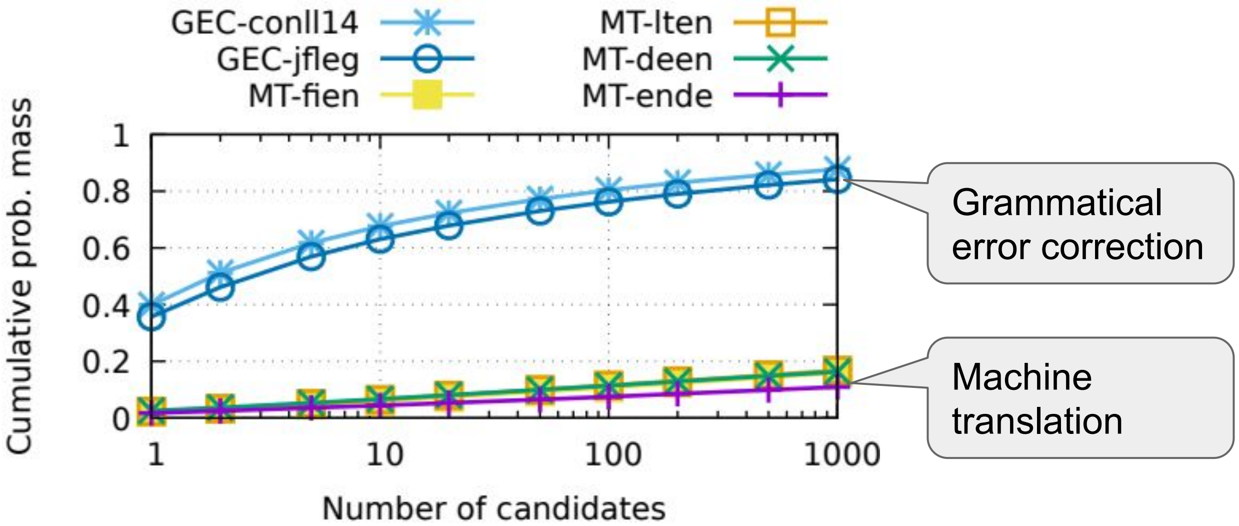
- MT has a high number of beam search errors, **but GEC does not**.



High resource: MT-deen (German-English) MT-ende (English-German)
Medium resource: MT-fien (Finnish-English)
Low resource: MT-lten (Lithuanian-English)

Cumulative probability mass of beam search n-best

- MT n-best lists cover a tiny fraction of the probability mass, **but GEC covers much more.**

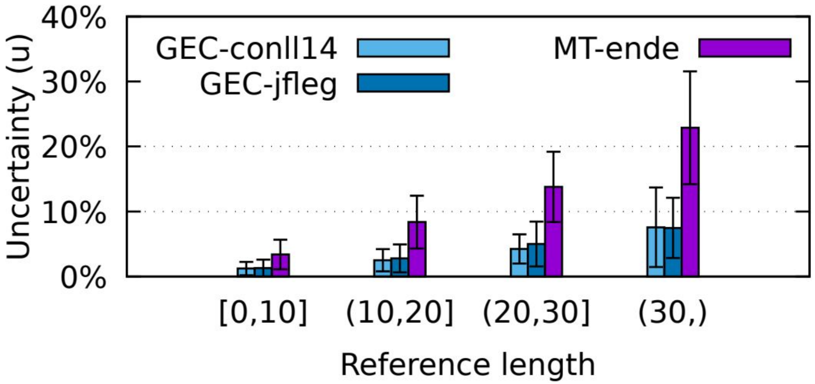


High resource: MT-deen (German-English) MT-ende (English-German)
Medium resource: MT-fien (Finnish-English)
Low resource: MT-lten (Lithuanian-English)

Google

Sentence-level uncertainty measure *u*

- For an *n*-way annotated source sentence with references $\mathbf{y}_1, \ldots, \mathbf{y}_n$ we define the uncertainty measure *u* as the relative edit distance averaged across all the reference pairs



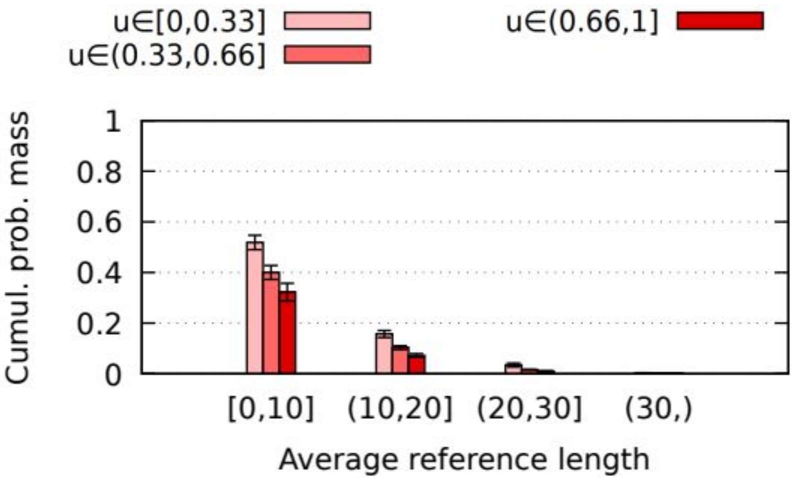
Blue: Grammatical error correction (GEC)
Purple: Machine translation (MT)

u increases with sentence length and intrinsic uncertainty of the task

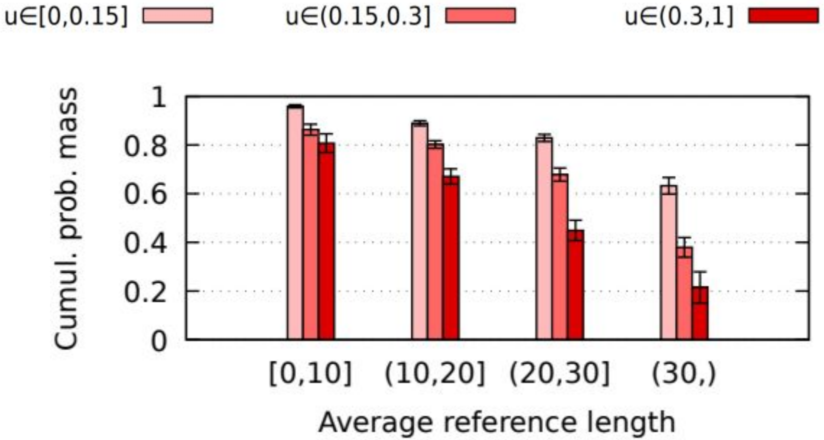
Probability mass on the sentence level

- For both GEC and MT, the probability mass is more spread out for uncertain sentences (in terms of uncertainty measure *u*).

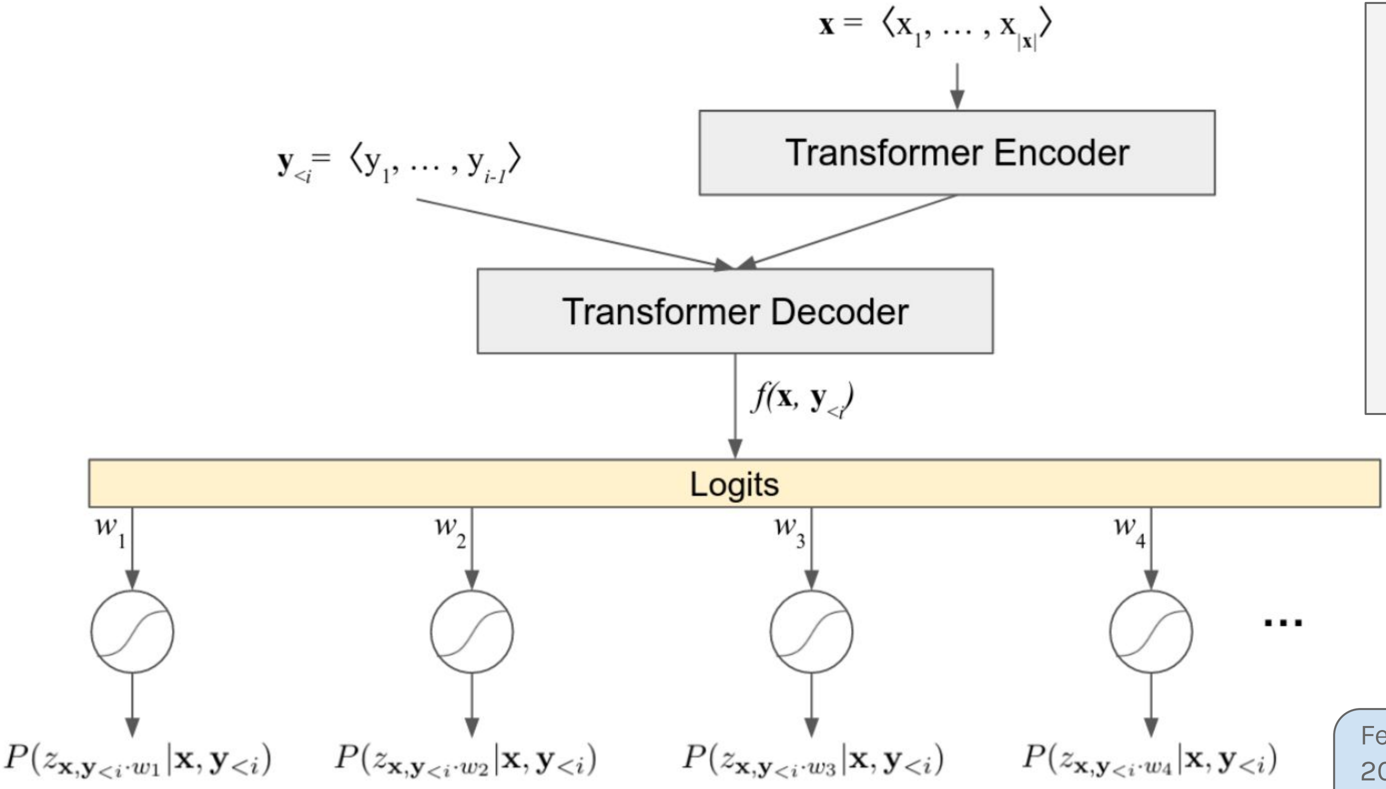
MT-ende



GEC-jfleg



SCONES: Framing seq2seq as a multi-label problem



- Replaces softmax activation with separate sigmoids for each item in the vocabulary
- No summation over the full vocabulary
- Tokens do not share probability mass
- All other parts of the model remain unchanged

Felix Stahlberg and Shankar Kumar. 2022. Jam or Cream First? Modeling Ambiguity in Neural Machine Translation with SCONES. NAACL

SCONES decoding

- Idea: Learn separate binary classifiers for each (**x**, **y**) pair that indicate whether **y** is a valid translation (prefix) of **x**:
  - $z_{\mathbf{x},\mathbf{y}}$ is 1 iff. there is a continuation of **y** to a valid translation

$$t(\mathbf{x},\mathbf{y}) := \begin{cases} \texttt{true} & \text{if } \mathbf{y} \text{ is a translation of } \mathbf{x} \\ \texttt{false} & \text{otherwise} \end{cases} \qquad z_{\mathbf{x},\mathbf{y}} := \begin{cases} 1 & \exists \mathbf{y}' \in \mathcal{V}^* : t(\mathbf{x}, \mathbf{y}\cdot\mathbf{y}') = \texttt{true} \\ 0 & \text{otherwise} \end{cases}$$

- Decompose into conditionals for left-to-right decoding:

$$\begin{aligned} P(z_{\mathbf{x},\mathbf{y}} = 1|\mathbf{x}) &= \prod_{i=1}^{|\mathbf{y}|} P(z_{\mathbf{x},\mathbf{y}_{\leq i}} = 1 | z_{\mathbf{x},\mathbf{y}_{<i}} = 1, \mathbf{x}) \\ &= \prod_{i=1}^{|\mathbf{y}|} P(z_{\mathbf{x},\mathbf{y}_{\leq i}} = 1 | \mathbf{x}, \mathbf{y}_{<i}). \end{aligned}$$

Sigmoid

$$P(z_{\mathbf{x},\mathbf{y}_{<i}\cdot w} = 1|\mathbf{x}, \mathbf{y}_{<i}) = \sigma(f(\mathbf{x}, \mathbf{y}_{<i})_w)$$

Transformer logits

SCONES training loss (token position *i*)

$$\mathcal{L}_{\text{SCONES}}(\mathbf{x}, \mathbf{y}, i) = \mathcal{L}_{+}(\mathbf{x}, \mathbf{y}, i) + \alpha \mathcal{L}_{-}(\mathbf{x}, \mathbf{y}, i)$$

Increase prob. of reference token

Decrease prob. of all other tokens

$$\begin{aligned}\mathcal{L}_{+}(\mathbf{x}, \mathbf{y}, i) &= -\log P(z_{\mathbf{x}, \mathbf{y}_{\leq i}} = 1 | \mathbf{x}, \mathbf{y}_{<i}) \\ &= -\log \sigma(f(\mathbf{x}, \mathbf{y}_{<i})_{y_i}).\end{aligned}$$

$$\begin{aligned}\mathcal{L}_{-}(\mathbf{x}, \mathbf{y}, i) &= -\sum_{w \in \mathcal{V} \setminus \{y_i\}} \log P(z_{\mathbf{x}, \mathbf{y}_{<i} \cdot w} = 0 | \mathbf{x}, \mathbf{y}_{<i}) \\ &= -\sum_{w \in \mathcal{V} \setminus \{y_i\}} \log(1 - \sigma(f(\mathbf{x}, \mathbf{y}_{<i})_w)).\end{aligned}$$

BLEU score improvements (tuned alpha)

	Greedy search						Beam search (beam size = 4)					
	de-en	en-de	fi-en	en-fi	lt-en	en-lt	de-en	en-de	fi-en	en-fi	lt-en	en-lt
Softmax	38.8	38.7	26.9	18.5	26.3	11.5	39.6	39.4	27.7	19.0	26.9	12.0
SCONES	39.9	39.1	27.6	19.5	27.7	12.5	40.3	39.8	28.4	20.0	28.9	12.6
Rel. improvement	**+2.7**‡	**+1.2**	**+2.8**†	**+5.4**‡	**+5.3**‡	**+8.5**‡	**+1.7**†	**+0.9**	**+2.7**†	**+5.5**‡	**+7.4**‡	**+5.7**

- SCONES consistently beats the Softmax baselines
- SCONES with greedy search can often outperform softmax with beam search

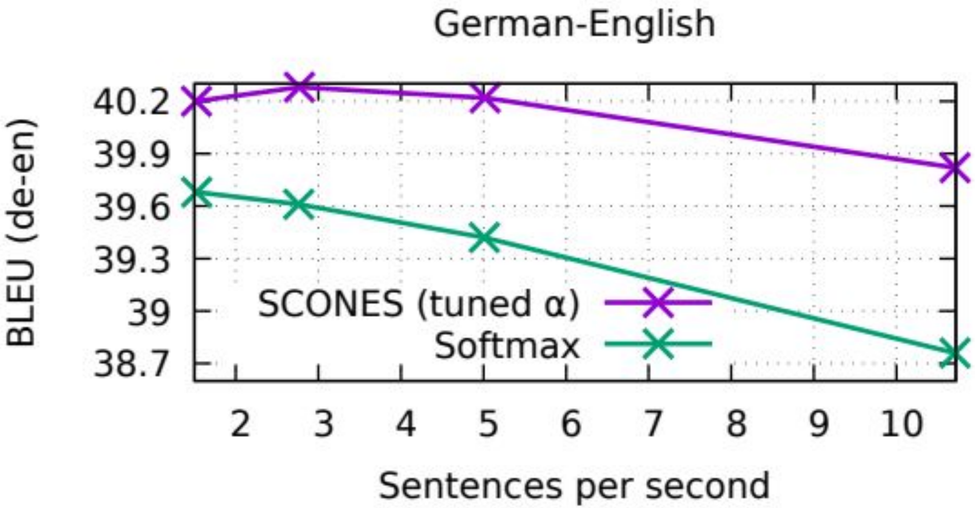
Language pair	α
de-en	0.5
en-de	0.5
fi-en	0.7
en-fi	1.0
lt-en	0.7
en-lt	0.9

BLEURT-20 score improvements (tuned alpha)

	Greedy search						Beam search (beam size = 4)					
	de-en	**en-de**	**fi-en**	**en-fi**	**lt-en**	**en-lt**	**de-en**	**en-de**	**fi-en**	**en-fi**	**lt-en**	**en-lt**
Softmax	70.44	68.08	68.93	66.16	68.52	56.68	70.78	68.48	69.56	66.44	69.20	57.61
SCONES	70.69	67.55	69.28	67.32	68.96	58.68	70.88	67.99	69.72	67.91	69.95	59.48

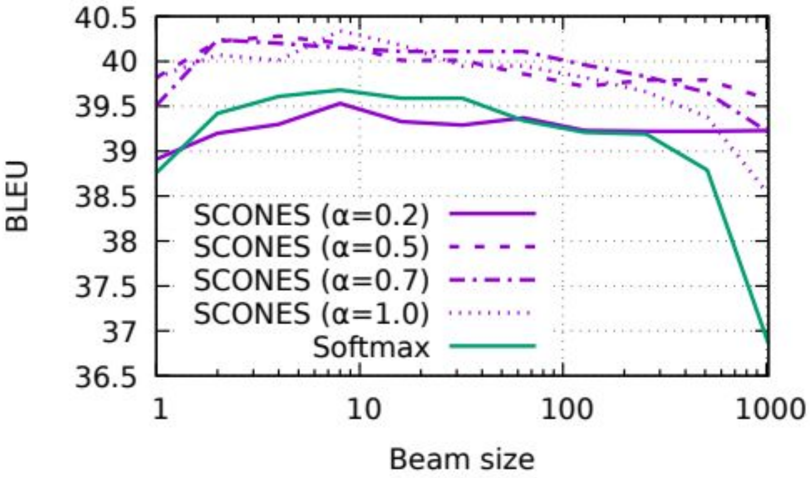
- SCONES beats the Softmax baseline BLEURT scores for all language directions except English-German

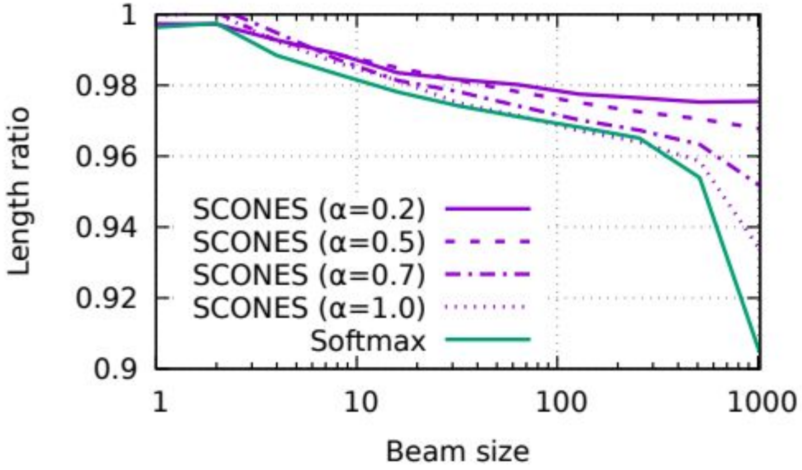
Quality/runtime trade-off



Speed-ups of up to 4x without BLEU degradation

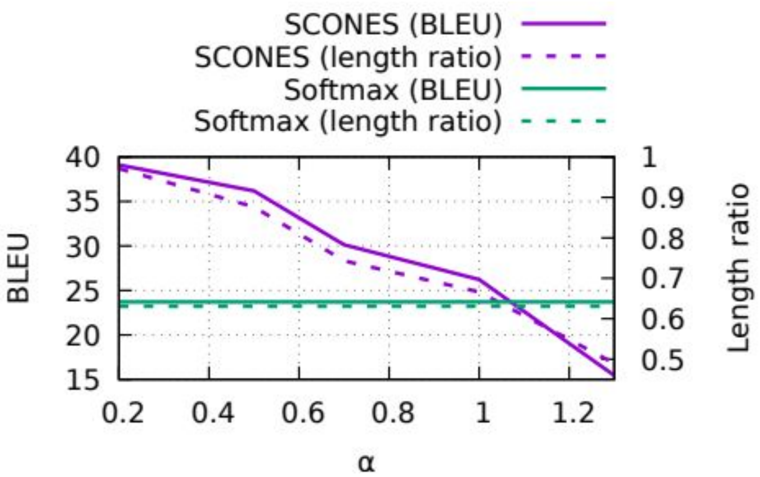
Fixing the beam search curse





A small alpha value mitigates the beam search curse.

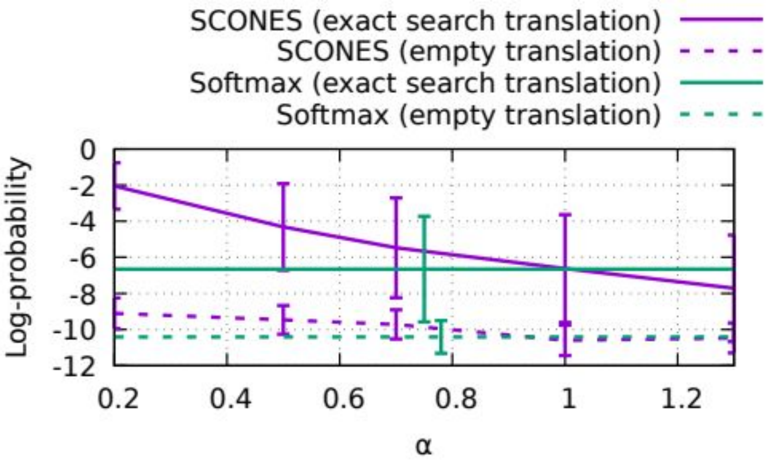
Fixing inadequate highest probability translations



Unlike softmax, SCONES with a small alpha value assigns the highest probability to adequate translations.

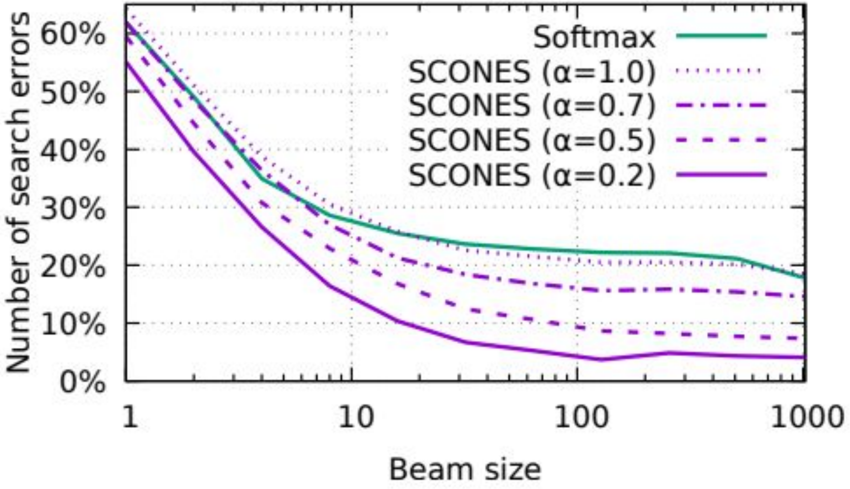
Google

Fixing empty highest probability translations



Unlike softmax, SCONES with a small alpha value achieves good separation between most likely and empty translation.

Beam search errors

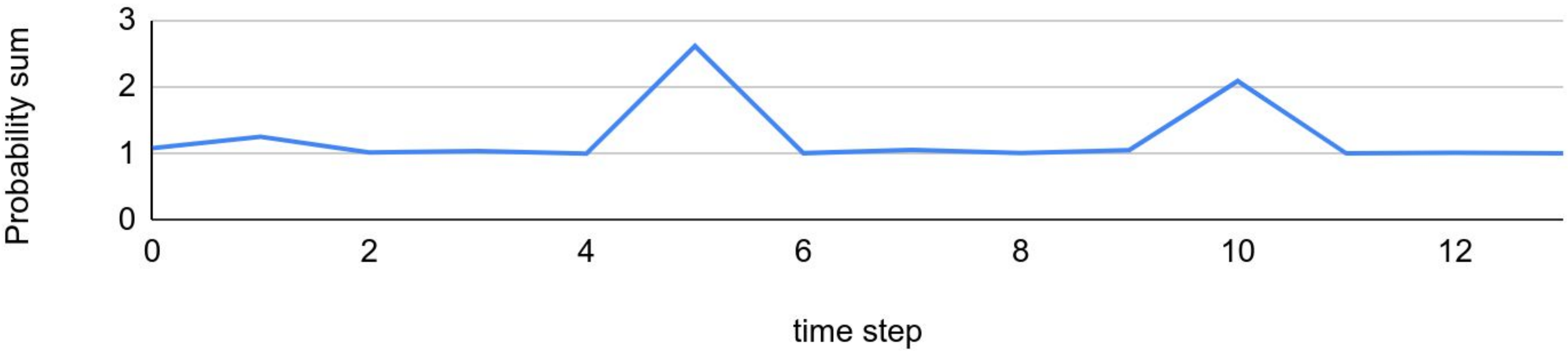


SCONES with a small alpha value reduces the number of beam search errors.

SCONES represents intrinsic uncertainty

French-English example

Input sentence: la stratosphère se trouve à environ 10 à 50 km d' altitude .
Output sentence: The stratosphere is about 10 to 50 km altitude .



SCONES represents intrinsic uncertainty

French-English example

Input sentence: la stratosphère se trouve à environ 10 à 50 km d' altitude .
Output sentence: The stratosphere is about 10 to 50 km altitude .

Time step 0	
Tok.	**Prob.**
_The	0.999

Time step 1	
Tok.	**Prob.**
_st	0.996

...

Time step 5	
Tok.	**Prob.**
_about	0.987
_approx	0.627
_around	0.389

...

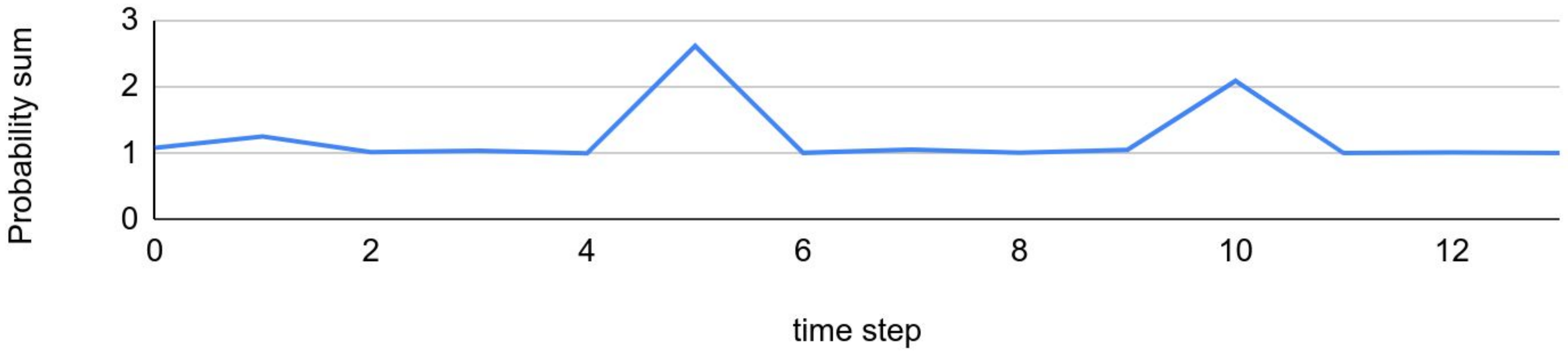
Time step 8	
Tok.	**Prob.**
_50	1.000

...

Time step 10	
Tok.	**Prob.**
_alt	0.918
_above	0.902

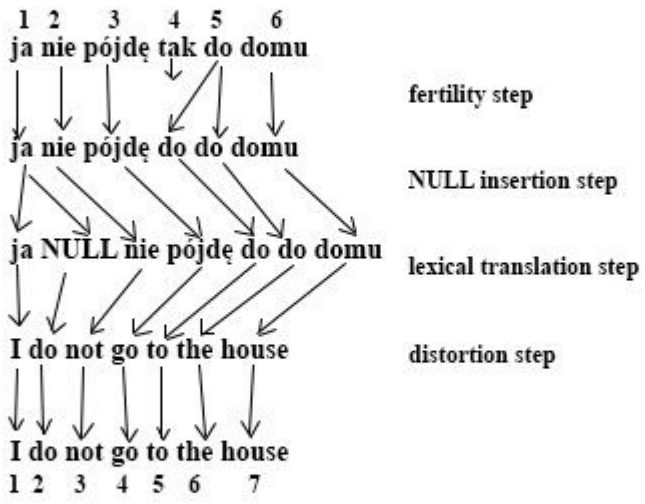
...

Time step 13	
Tok.	**Prob.**
</s>	1.000

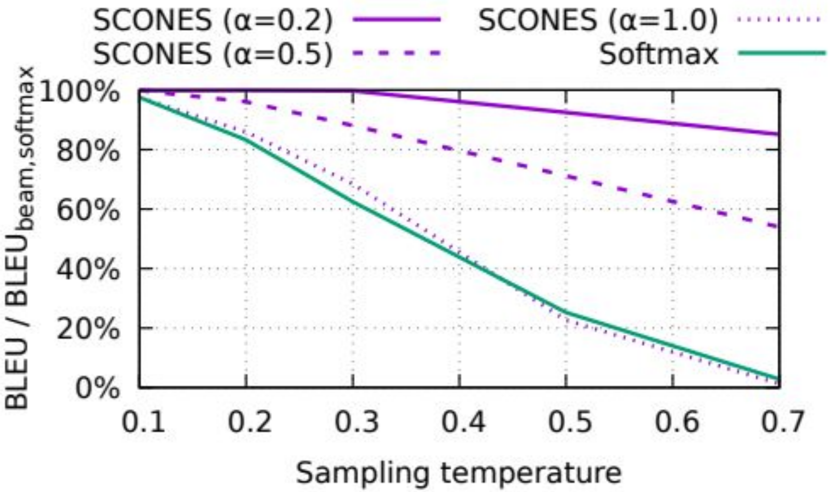


Synthetic language experiments

- Idea: Train an IBM-3 word alignment model with GIZA++
- Generate training data by sampling from IBM-3
- Changing the sampling temperature controls the intrinsic uncertainty of the translation task



German-to-synthetic-English translation



SCONES with a small alpha value is more robust against intrinsic uncertainty.

SCONES - Practical advice

- Using the optimizer hyper-parameters from softmax for SCONES training is often a good starting point, but SCONES may need further hyper-parameter search.
- JAX code for SCONES is in the appendix of the paper. A numerically more stable TensorFlow implementation is in Lingvo.
- SCONES gradient norms are huge in the first few training iterations. Using Lamb instead of Adam can help.
- A good alpha value depends on the task and the vocabulary size. We have trained models with alpha between 0.1 and 6.0.

Summary

- Intrinsic uncertainty causes various issues in conventional seq2seq models
  - Beam search curse (degrading performance at higher beam sizes)
  - Overly spread out probability mass
  - Beam search errors
  - Inadequate modes
- SCONES is designed to equip sequence models with the ability to represent intrinsic uncertainty
  - Has a tunable parameter alpha to balance the importance of correct tokens and incorrect tokens
- SCONES with tuned alpha value yields significant BLEU and/or runtime improvements over softmax baselines
- SCONES with small alpha value fixes various pathologies of seq2seq models for intrinsically uncertain NLP tasks like machine translation

Thank you!

Your scones connection in Prague:

artic BAKEHOUSE

http://www.articbakehouse.cz/

https://flic.kr/p/bm2u5o

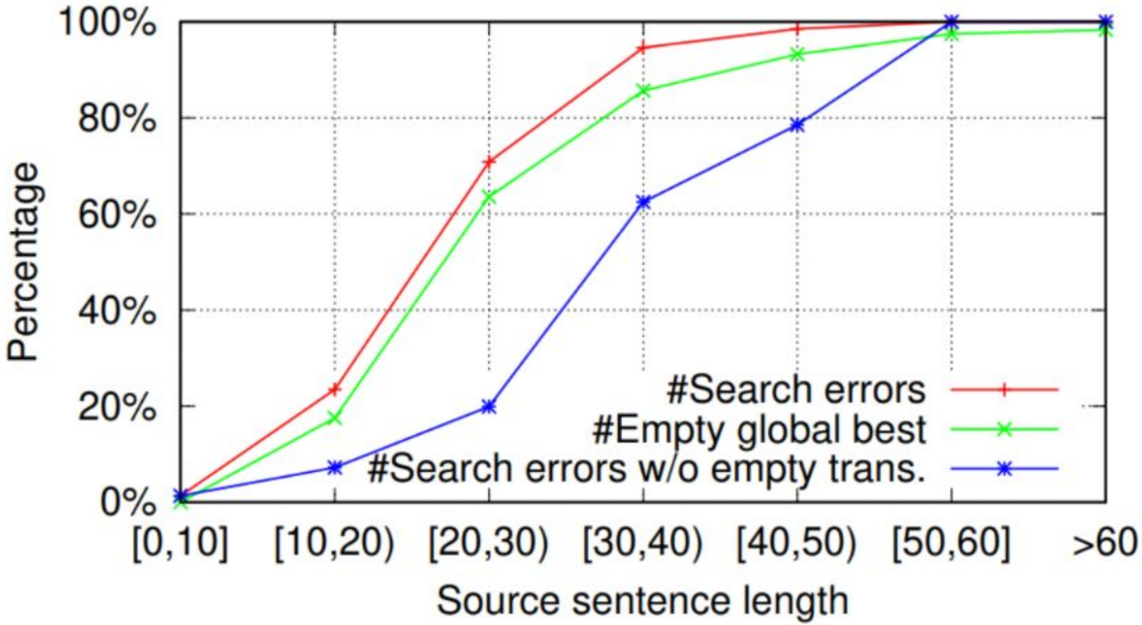
Length normalization

Length-dependent lower bounds: $\gamma_k = (k+1)\frac{\log P(\mathbf{y}_{\text{beam}}|\mathbf{x})}{|\mathbf{y}_{\text{beam}}|+1}$

Search	W/o length norm. BLEU	W/o length norm. Ratio	With length norm. BLEU	With length norm. Ratio
Beam-10	37.0	1.00	36.3	1.03
Beam-30	36.7	0.98	36.3	1.04
Exact	27.2	0.74	36.4	1.03

Length normalization fixes translation lengths but prevents exact search from matching the BLEU score of Beam-10.

Source sentence length



Long source sentences are more affected by search errors and empty translations.

Architectures

Model	Beam-10		Exact
	BLEU	#Search err.	#Empty
LSTM*	28.6	58.4%	47.7%
SliceNet*	28.8	46.0%	41.2%
Transformer-Base	30.3	57.7%	51.8%
Transformer-Big*	31.7	32.1%	25.8%

➡ **The problem of search errors and empty translations is not specific to transformer_base.**