Analyzing and Tackling Challenges in NMT

Marc'Aurelio Ranzato

Facebook AI Research - New York City

ranzato@fb.com

https://ranzato.github.io/

in collaboration with: M. Auli, A. Conneau, S. Edunov, L. Denoyer, D. Grangier, H. Jegou, G. Lample, M. Ott

Harvard CS 287, 1 March 2018

Seattle
Montreal
Paris
Menlo Park
New York City
Tel Aviv

FAIR

Machine Translation

- Case-study for sequence to sequence transduction.
- It works in practice and has lots of applications.
- Some challenges:
    - input and output are discrete sequences of variable length
    - alignment
    - large vocabulary, large hypothesis space, need to search
    - one-to-many mapping / uncertainty, metric
    - domain shift
    - some language pairs may have little parallel data

Six challenges for neural machine translation, Koehn et al. Workshop NMT, ACL 2017

Machine Translation

- Case-study for sequence to sequence transduction.
- It works in practice and has lots of applications.
- Some challenges:
    - input and output are discrete sequences of variable length
    - alignment
    - large vocabulary, large hypothesis space, need to search
    - **one-to-many mapping / uncertainty, metric**
    - domain shift
    - **some language pairs may have little parallel data**

Six challenges for neural machine translation, Koehn et al. Workshop NMT, ACL 2017

Goal

To propose:

- Tools to analyze such challenges.
- Methods to tackle some of these challenges,

Neural Machine Translation

(in 3 slides)

Example:

ITA (source) : Il gatto si e' seduto sul tappetino.

↓

EN (target) : The cat sat on the mat.

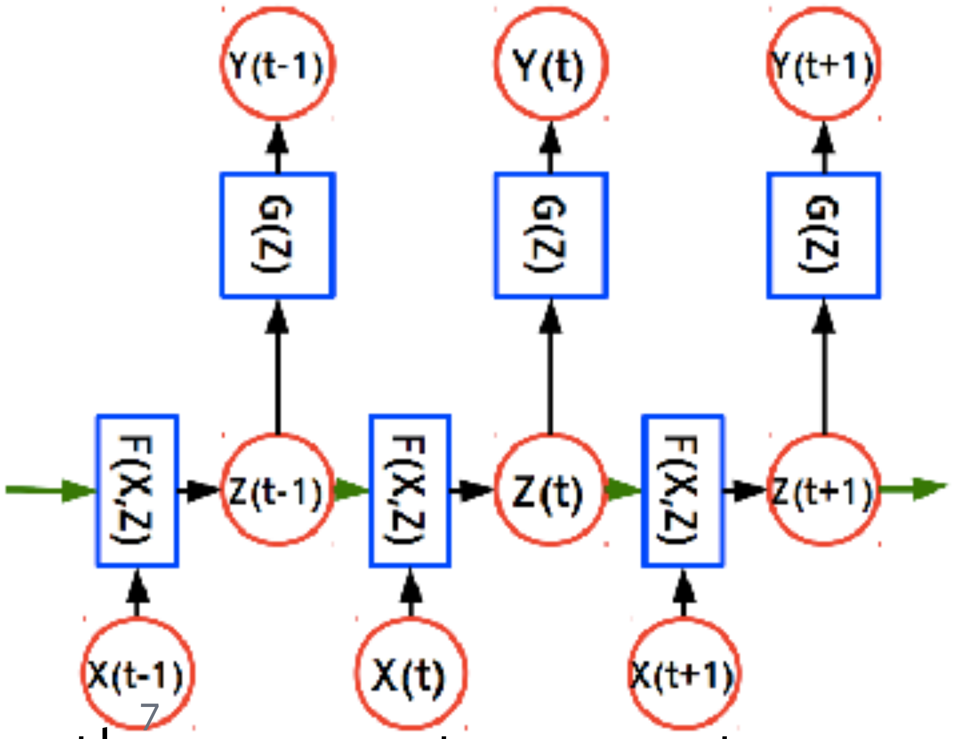
Approach:

Have one RNN/CNN to encode the source sentence, and another RNN/CNN/MemNN to predict the target sentence.

The target RNN learns to (soft) align via attention.

Neural machine translation by jointly learning to align and translate, Bahdanau et al. ICLR 2015

cat sat on



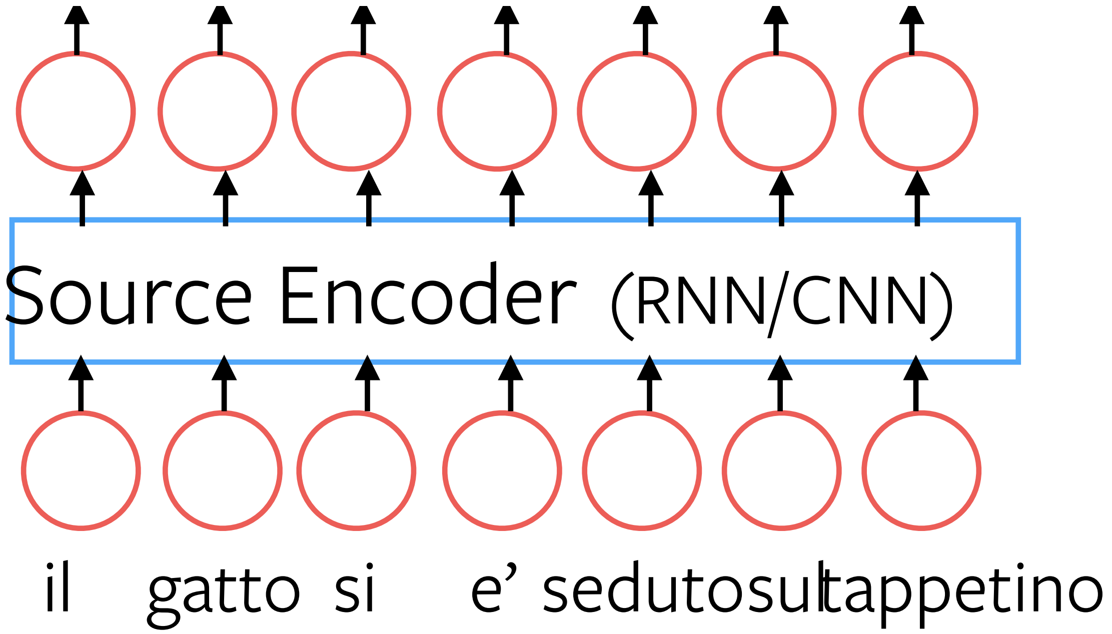
the cat sat

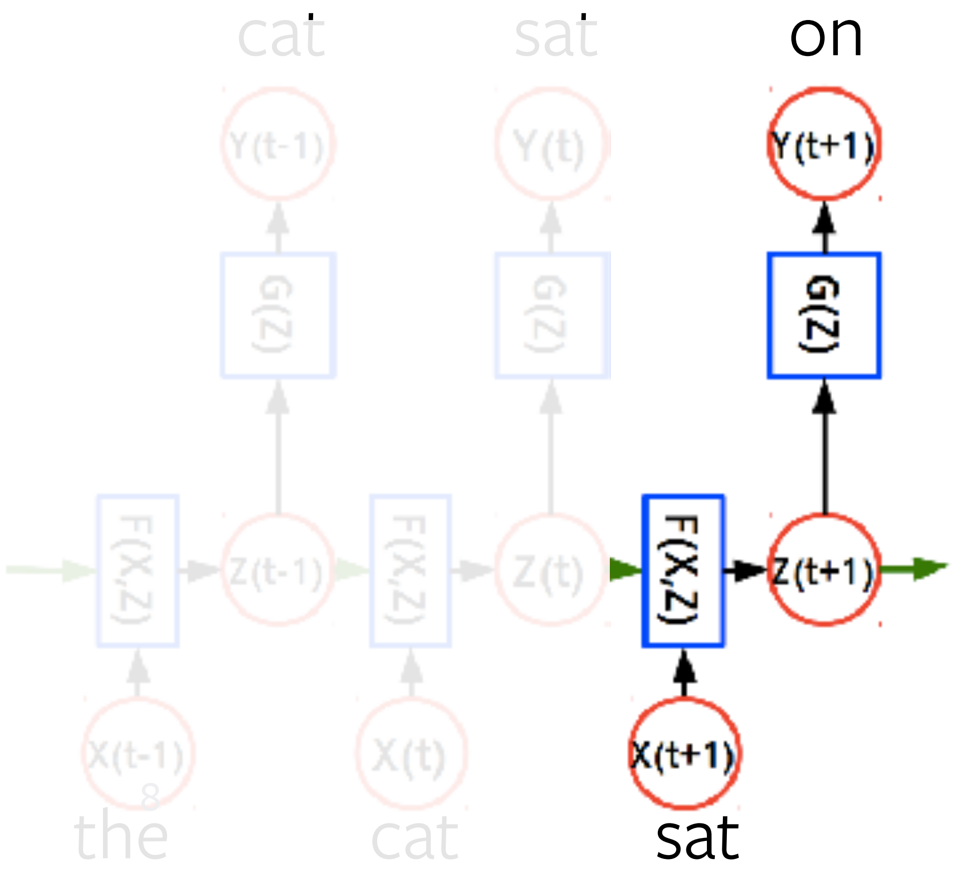
Y. LeCun’s diagram

Source

Target

1) Represent source

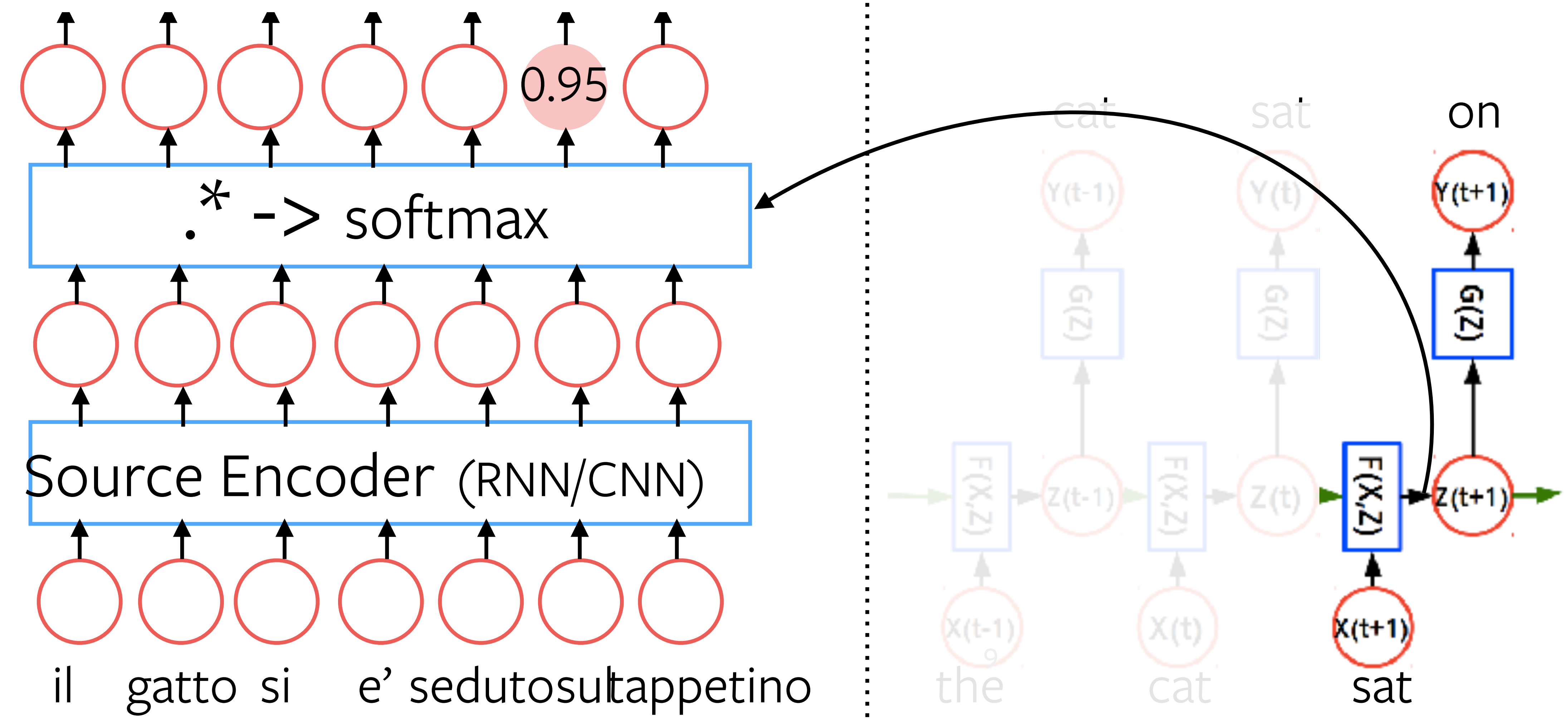




Source

Target

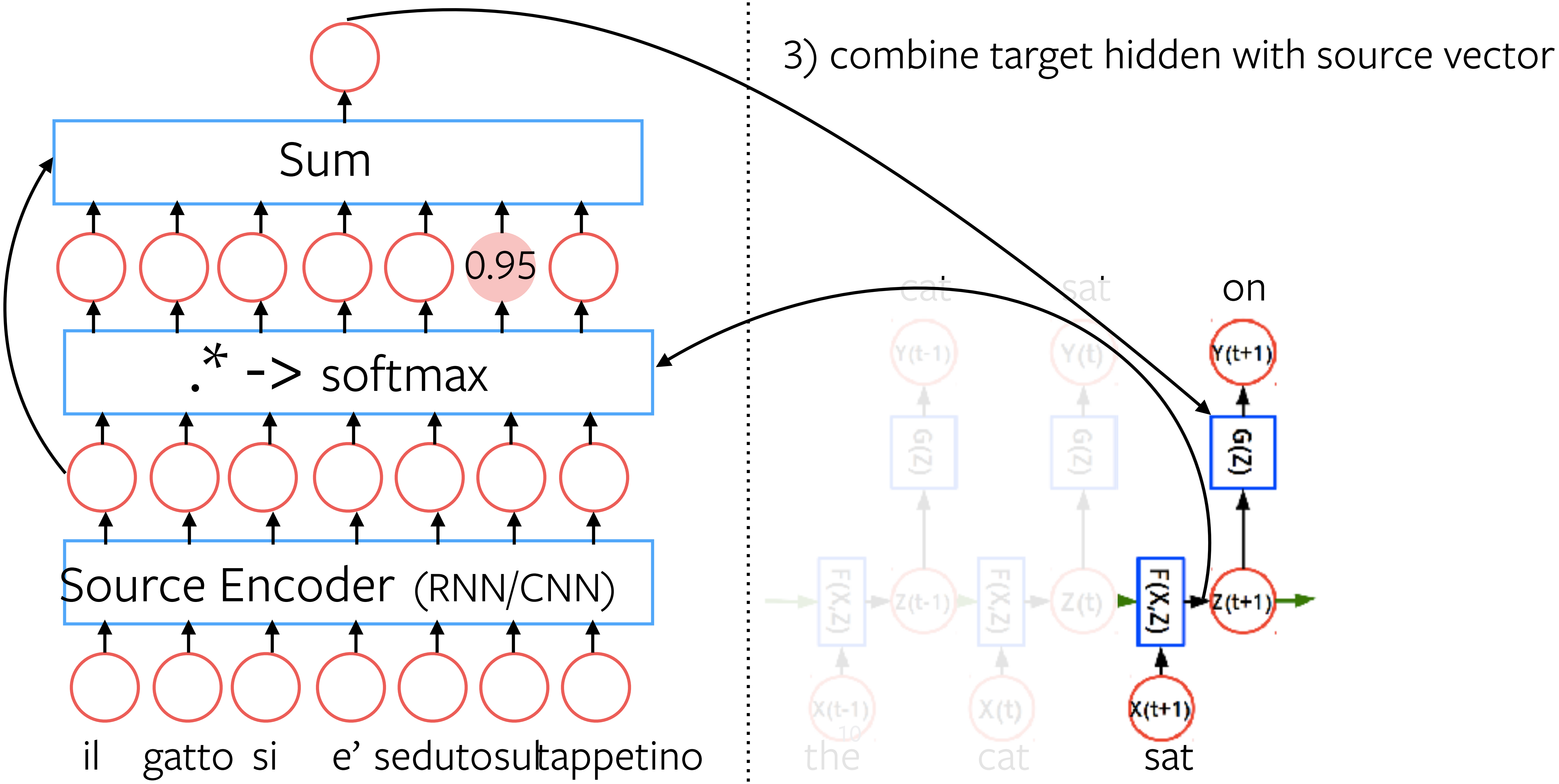
2) score each source word (attention)



M. Ranzato

Source

Target



3) combine target hidden with source vector

NMT Training & Inference

Training: predict one target token at the time and minimize cross-entropy loss.

Inference: find the most likely target sentence (approximately) using beam search.

Evaluation: BLEU at inference time.

Lecture Outline

- Exposure bias/Loss Mismatch: Training at the Sequence Level.
  - how do classical structured prediction losses fare against recent proposals?
  - how much to be gained by fixing this inconsistency?
- Analyzing Uncertainty: model fitting and effects on search.
  - why do larger beam perform worse?
  - why is the model under-estimating rare words?
- Training Without Supervision.
  - how to leverage monolingual data?
  - can we learn without any parallel sentence?

Lecture Outline

- Exposure bias/Loss Mismatch: Training at the Sequence Level.
  - how do classical structured prediction losses fare against recent proposals?
  - how much to be gained by fixing this inconsistency?

Classical Structured Prediction Losses for Sequence to Sequence Learning
Sergey Edunov*, Myle Ott*, Michael Auli, David Grangier, Marc'Aurelio Ranzato
NAACL 2018
https://arxiv.org/abs/1711.04956

Lecture Outline

- Exposure bias/Loss Mismatch: Training at the Sequence Level.
  - how do classical structured prediction losses fare against recent proposals?
  - how much to be gained by fixing this inconsistency?

Classical Structured Prediction Losses for Sequence to Sequence Learning
Sergey Edunov*, Myle Ott*, Michael Auli, David Grangier, Marc'Aurelio Ranzato
NAACL 2018
https://arxiv.org/abs/1711.04956

credit: Several slides borrowed from Sergey.

Problems

- Exposure bias: training and testing are inconsistent. At training time, model has never observed its own predictions at input.
- At training time, we optimize for a different loss.
- Evaluation criterion is not differentiable.

Selection of Recent Literature

- RL-inspired methods
  - MIXER **Ranzato et al. ICLR 2016**
  - Actor-Critic **Bahdanau et al. ICLR 2017**
- Using beam search at training time:
  - BSO **Wiseman et al. ACL 2016**
  - Distillation based **Kim et al. EMNLP 2016**

Question

How do classical structure prediction losses compare against these recent methods?

Classical losses were often applied to log-linear models and/or other problems than MT.

Bottou et al. "Global training of document processing systems with graph transformer networks" CVPR 1997

Collins "Discriminative training methods for HMMs" EMNLP 2002

Taskar et al. "Max-margin Markov networks" NIPS 2003

Tsochantaridis et al. "Large margin methods for structured and interdependent output variables" JMLR 2005

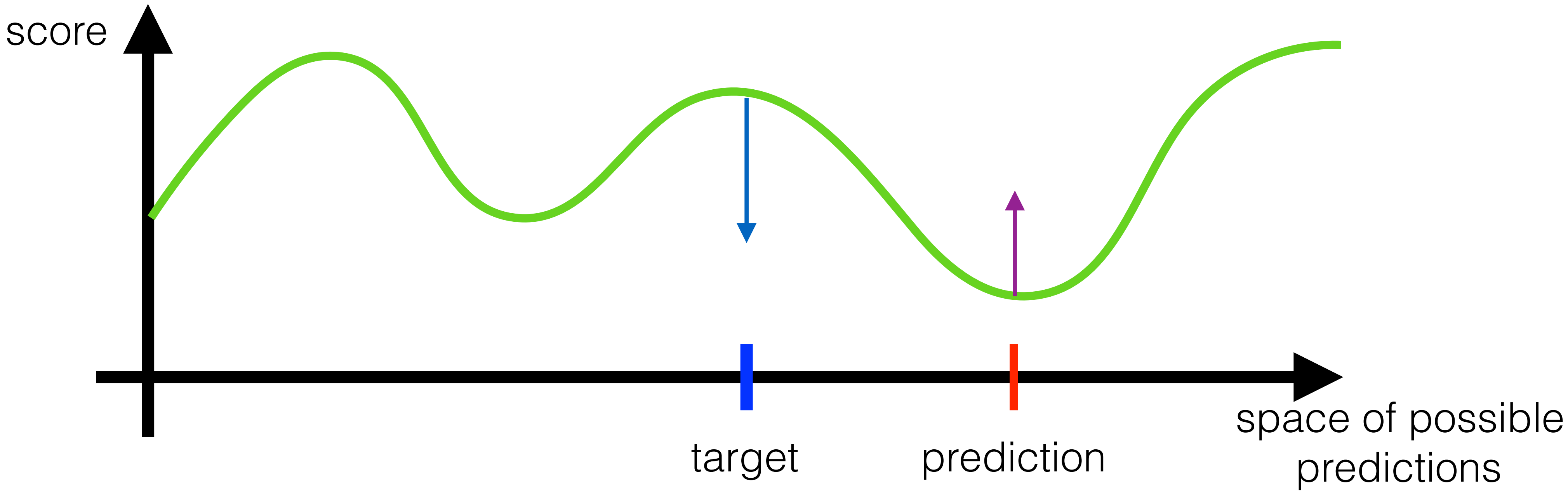
Och "Minimum error rate training in statistical machine translation" ACL 2003

Smith and Eisner "Minimum risk annealing for training log-linear models" ACL 2006

Gimpel and Smith "Softmax-margin CRFs: training log-linear models with cost functions" ACL 2010

Energy-Based Learning

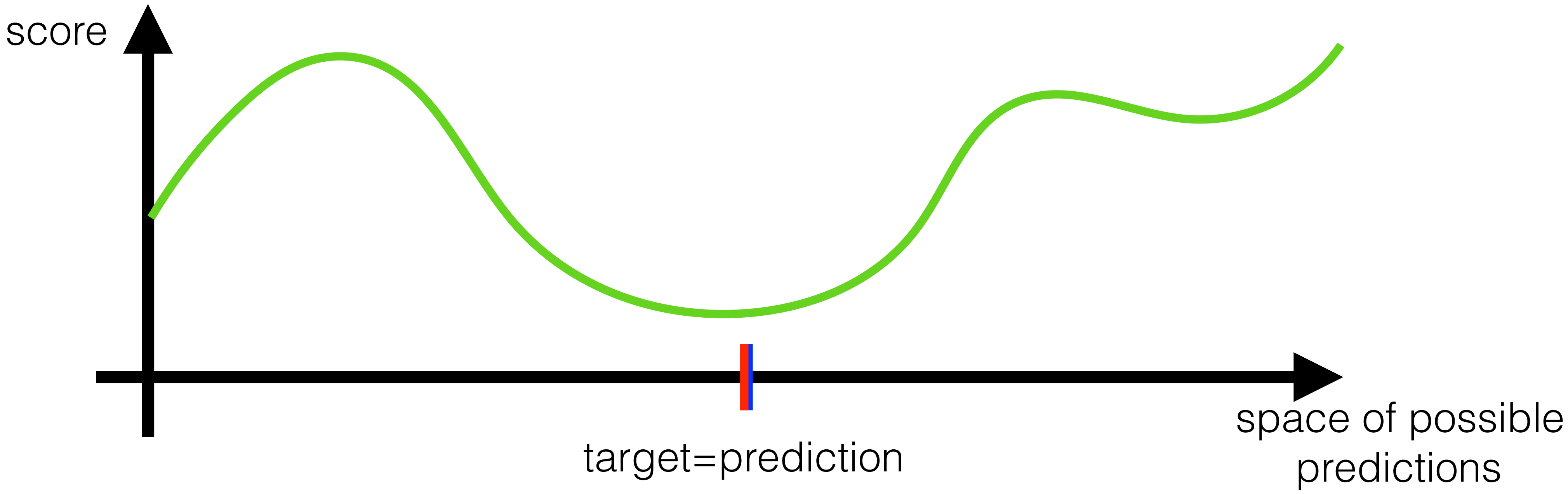
During training



LeCun et al. "A tutorial on energy-based learning" MIT Press 2006

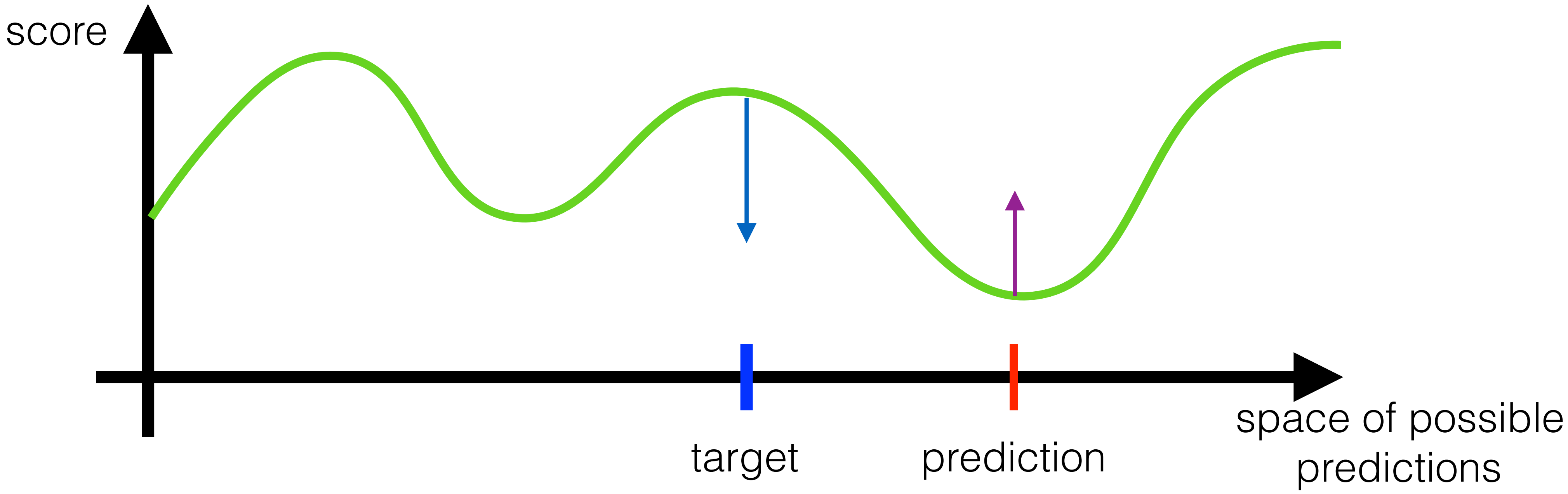
Energy-Based Learning

After training



LeCun et al. “A tutorial on energy-based learning” MIT Press 2006

Energy-Based Learning



Key questions if we want to extend this to structured outputs:

- how to search for most likely output? Enumeration & exact search are intractable.
- how to deal with uncertainty?
- what if target is not reachable?

LeCun et al. "A tutorial on energy-based learning" MIT Press 2006

M. Ranzato

Notation

$\mathbf{x} = (x_1, \ldots, x_m)$ input sentence

Notation

$\mathbf{x}$	input sentence
$\mathbf{t}$	target sentence

Notation

$\mathbf{x}$	input sentence
$\mathbf{t}$	target sentence
$\mathbf{u}$	hypothesis generated by the model

Notation

$\mathbf{x}$	input sentence
$\mathbf{t}$	target sentence
$\mathbf{u}$	hypothesis generated by the model

$\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \text{cost}(\mathbf{u}, \mathbf{t})$ oracle hypothesis

Notation

$\mathbf{x}$	input sentence
$\mathbf{t}$	target sentence
$\mathbf{u}$	hypothesis generated by the model
$\mathbf{u}^*$	oracle hypothesis
$\hat{\mathbf{u}}$	most likely hypothesis

Baseline: Token Level NLL

$$\mathcal{L}_{\text{TokNLL}} = -\sum_{i=1}^{n} \log p(t_i | t_1, \ldots, t_{i-1}, \mathbf{x})$$

for one particular training example and omitting dependence on model parameters.

Sequence Level NLL

$$\mathcal{L}_{\text{SeqNLL}} = -\log p(\mathbf{u}^*|\mathbf{x}) + \log \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} p(\mathbf{u}|\mathbf{x})$$

The sequence log-probability is simply the sum of the token-level log-probabilities.

Sequence Level NLL

$$\mathcal{L}_{\text{SeqNLL}} = -\log p(\mathbf{u}^*|\mathbf{x}) + \log \sum_{\mathbf{u}\in\mathcal{U}(\mathbf{x})} p(\mathbf{u}|\mathbf{x})$$

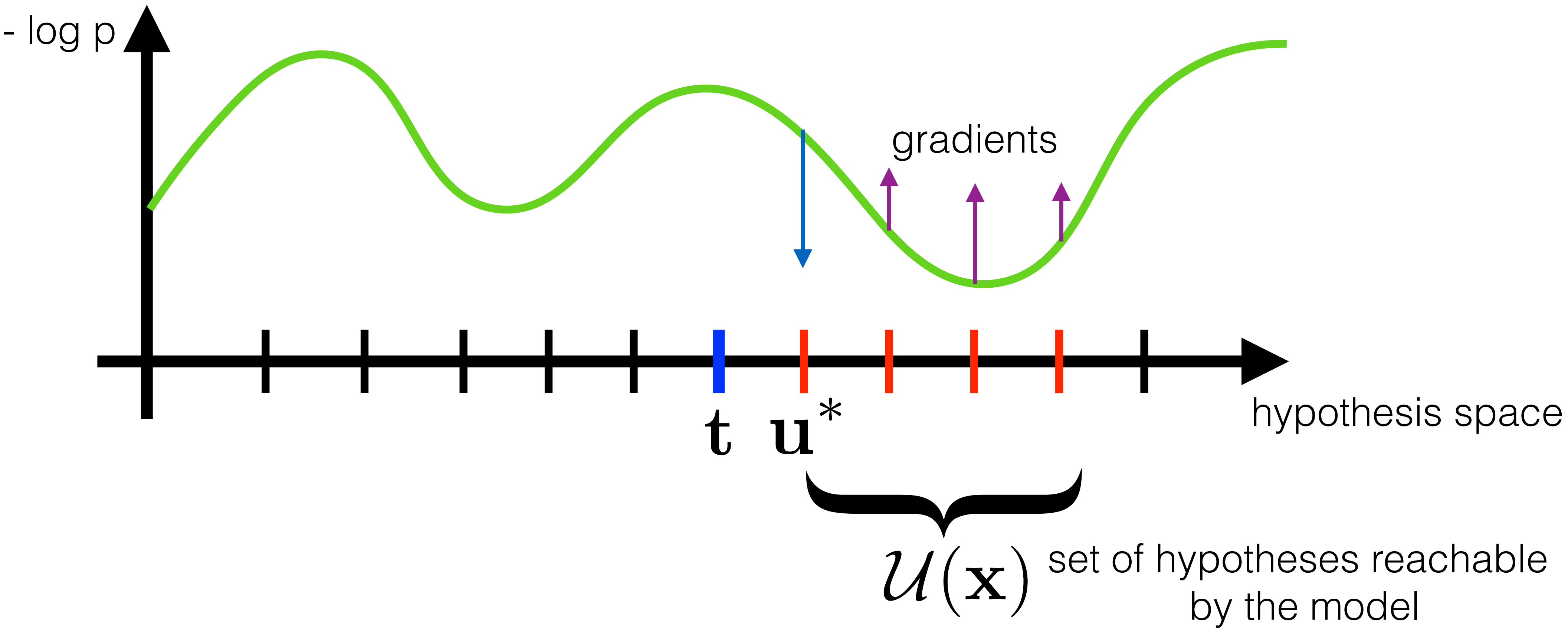
The sequence log-probability is simply the sum of the token-level log-probabilities.

Two key differences: choice of target and hypothesis set.

Homework: compute gradients of loss w.r.t. inputs to token level softmaxes.

Sequence Level NLL

$$\mathcal{L}_{\text{SeqNLL}} = -\log p(\mathbf{u}^*|\mathbf{x}) + \log \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} p(\mathbf{u}|\mathbf{x})$$


- log p
gradients
hypothesis space
$\mathbf{t}$
$\mathbf{u}^*$
$\mathcal{U}(\mathbf{x})$
set of hypotheses reachable by the model

Example

Source:
Wir müssen unsere Einwanderungspolitik in Ordnung bringen.

Target
We have to fix our immigration policy.

Beam:

BLEU	Model score	
75.0	-0.23	We need to fix our immigration policy.
100.0	-0.30	We have to fix our immigration policy.
36.9	-0.36	We need to fix our policy policy.
66.1	-0.42	We have to fix our policy policy.
66.1	-0.44	We've got to fix our immigration policy.

Example

Source:
Wir müssen unsere Einwanderungspolitik in Ordnung bringen.

Target
We have to fix our immigration policy.

Beam:

BLEU	Model score		
75.0	-0.23	↓	We need to fix our immigration policy.
100.0	-0.30	↑	We have to fix our immigration policy.
36.9	-0.36	↓	We need to fix our policy policy.
66.1	-0.42	↓	We have to fix our policy policy.
66.1	-0.44	↓	We've got to fix our immigration policy.

Observations

- Important to use oracle hypothesis as surrogate target as opposed to golden target. Otherwise, the model learns to assign very bad scores to its hypotheses but is not trained to reach the target.

- Evaluation metric only used for oracle selection of target.

- Several ways to generate $\mathcal{U}(\mathbf{x})$.

- Similar to token level NLL but normalizing over (subset of) hypotheses. Hypothesis score: average token level log-probability.

Expected Risk

$$\mathcal{L}_{\text{Risk}} = \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \text{cost}(\mathbf{t}, \mathbf{u}) \frac{p(\mathbf{u}|\mathbf{x})}{\sum_{\mathbf{u}' \in \mathcal{U}(\mathbf{x})} p(\mathbf{u}'|\mathbf{x})}$$

- The cost is the evaluation metric; e.g.: 100-BLEU.
- REINFORCE is a special case of this (a single sample Monte Carlo estimate of the expectation over the *whole* hypothesis space).

Homework: compute gradients of loss w.r.t. inputs to token level softmaxes.

Example

Source:
Wir müssen unsere Einwanderungspolitik in Ordnung bringen.

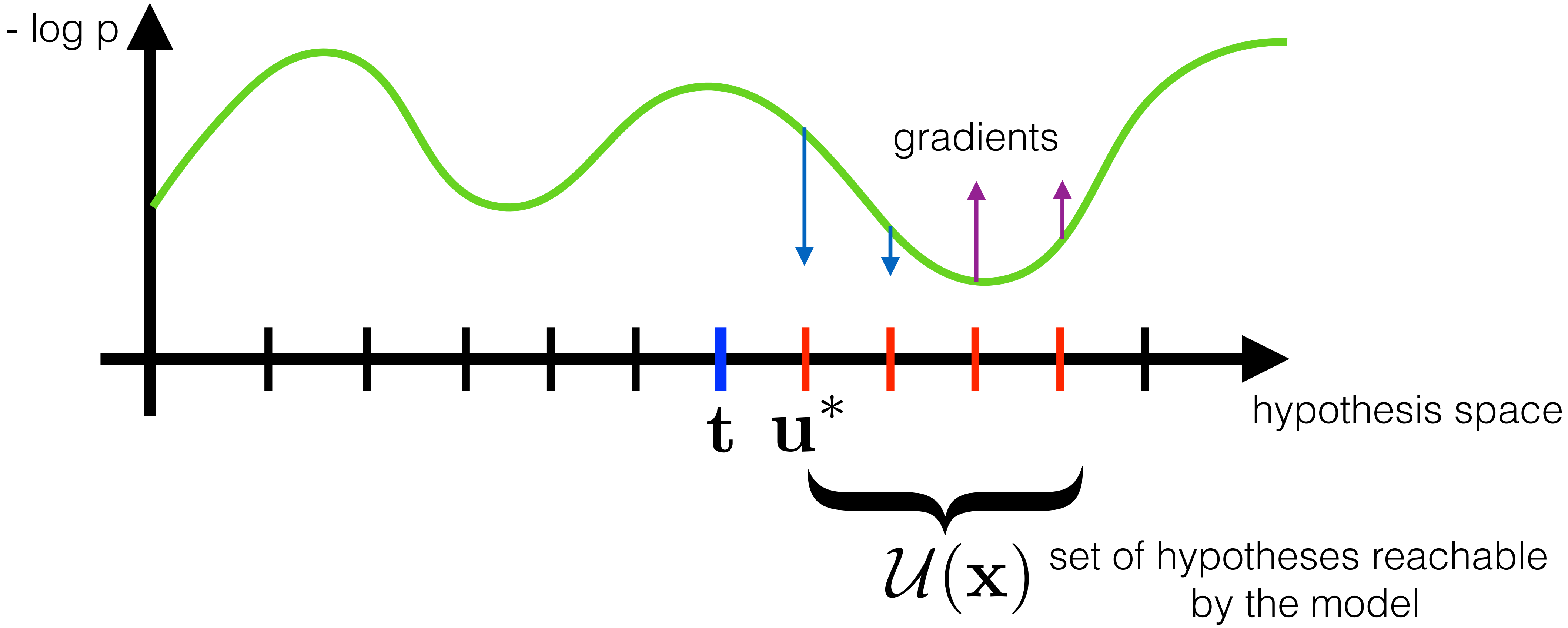
Target
We have to fix our immigration policy.

Beam:

BLEU	Model score		
75.0	-0.23	↑	We need to fix our immigration policy.
100.0	-0.30	↑	We have to fix our immigration policy.
36.9	-0.36	↓	We need to fix our policy policy.
66.1	-0.42	↓	We have to fix our policy policy.
66.1	-0.44	↓	We've got to fix our immigration policy.

(expected BLEU=69)

Example


- log p
gradients
hypothesis space
$\mathbf{t}$
$\mathbf{u}^*$
$\mathcal{U}(\mathbf{x})$
set of hypotheses reachable by the model

Max-Margin

$$\mathcal{L}_{\text{MaxMargin}} = \max\left[0, \text{cost}(\mathbf{t}, \hat{\mathbf{u}}) - \text{cost}(\mathbf{t}, \mathbf{u}^*) - s(\mathbf{u}^*|\mathbf{x}) + s(\hat{\mathbf{u}}|\mathbf{x})\right]$$

- The score is average token level log-probability (or un-normalized score).
- The cost is our evaluation metric; e.g.: 100-BLEU.
- Increase score of oracle hypothesis, while decreasing score of most likely hypothesis.

Homework: compute gradients of loss w.r.t. inputs to token level softmaxes.

Max-Margin

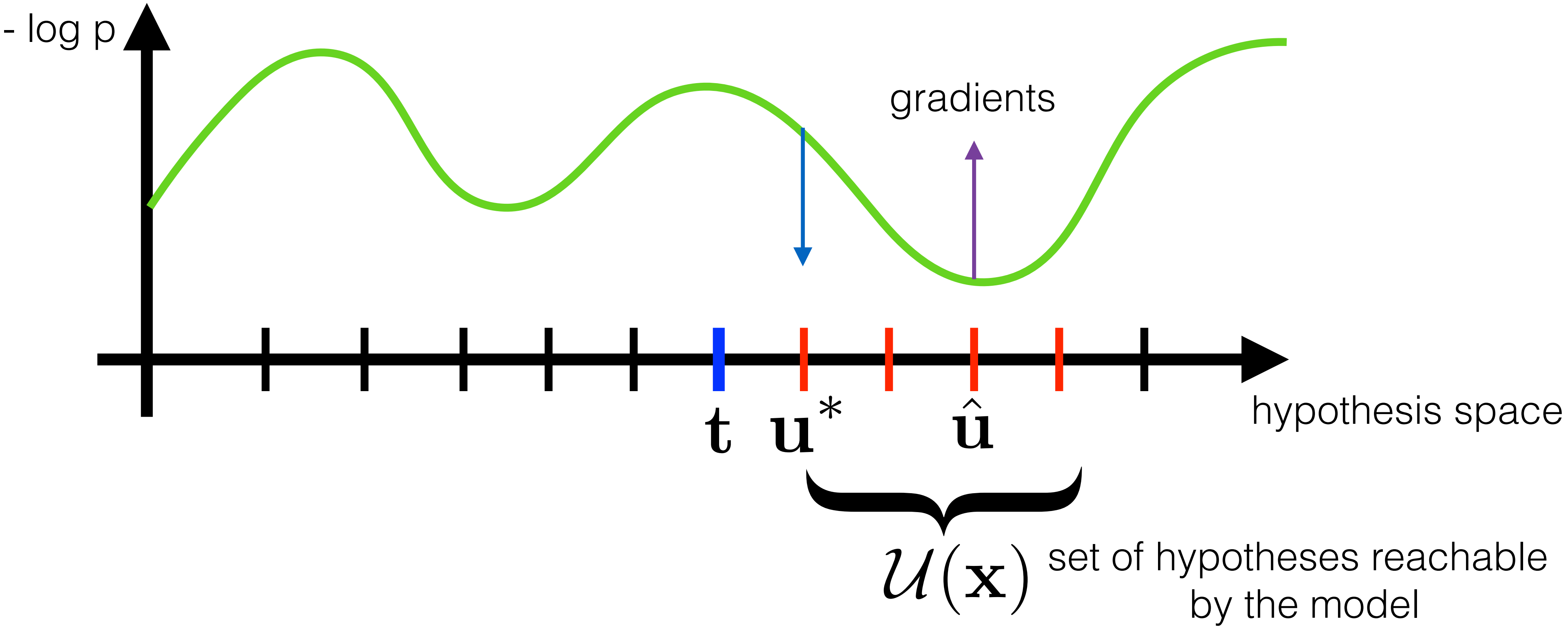
Source:
Wir müssen unsere Einwanderungspolitik in Ordnung bringen.

Target
We have to fix our immigration policy.

Beam:

BLEU	Model score		
75.0	-0.23	↓	We need to fix our immigration policy.
100.0	-0.30	↑	We have to fix our immigration policy.
36.9	-0.36		We need to fix our policy policy.
66.1	-0.42		We have to fix our policy policy.
66.1	-0.44		We've got to fix our immigration policy.

Max-Margin


- log p
gradients
hypothesis space
$\mathbf{t}$
$\mathbf{u}^*$
$\hat{\mathbf{u}}$
$\mathcal{U}(\mathbf{x})$
set of hypotheses reachable by the model

Check out the paper for more examples of sequence level training losses!

Practical Tips

- Start from a model pre-trained at the token level. Training with search is excruciatingly slow…

- Even better if pre-trained model had label smoothing.

- Accuracy VS speed trade-off: offline/online generation of hypotheses.

- Cost rescaling.

- Mix token level NLL loss with sequence level loss to improve robustness.

- Need to regularize more.

Results on IWSLT’14 De-En

	TEST
TokNLL (Wiseman et al. 2016)	24.0
BSO (Wiseman et al. 2016)	26.4
Actor-Critic (Bahdanau et al. 2016)	28.5
Phrase-based NMT (Huang et al. 2017)	29.2

Results on IWSLT'14 De-En

	TEST
TokNLL (Wiseman et al. 2016)	24.0
BSO (Wiseman et al. 2016)	26.4
Actor-Critic (Bahdanau et al. 2016)	28.5
Phrase-based NMT (Huang et al. 2017)	29.2
our TokNLL	31.7
SeqNLL	32.7
Risk	**32.9**
Max-Margin	32.6

Observations

- Sequence level training does improve evaluation metric (both on training and) on test set.
- There is not so much difference between the different variants of losses. Risk is just slightly better.
- In our implementation and using the same computational resources, sequence level training is 26x slower per update using online beam generation of 5 hypotheses.

Observations

- Sequence level training does improve evaluation metric (both on training and) on test set.

- There is not so much difference between the different variants of losses. Risk is just slightly better.

- In our implementation and using the same computational resources, sequence level training is 26x slower per update using online beam generation of 5 hypotheses.

- *Hard comparison since each paper has a different baseline!*

Fair Comparison to BSO

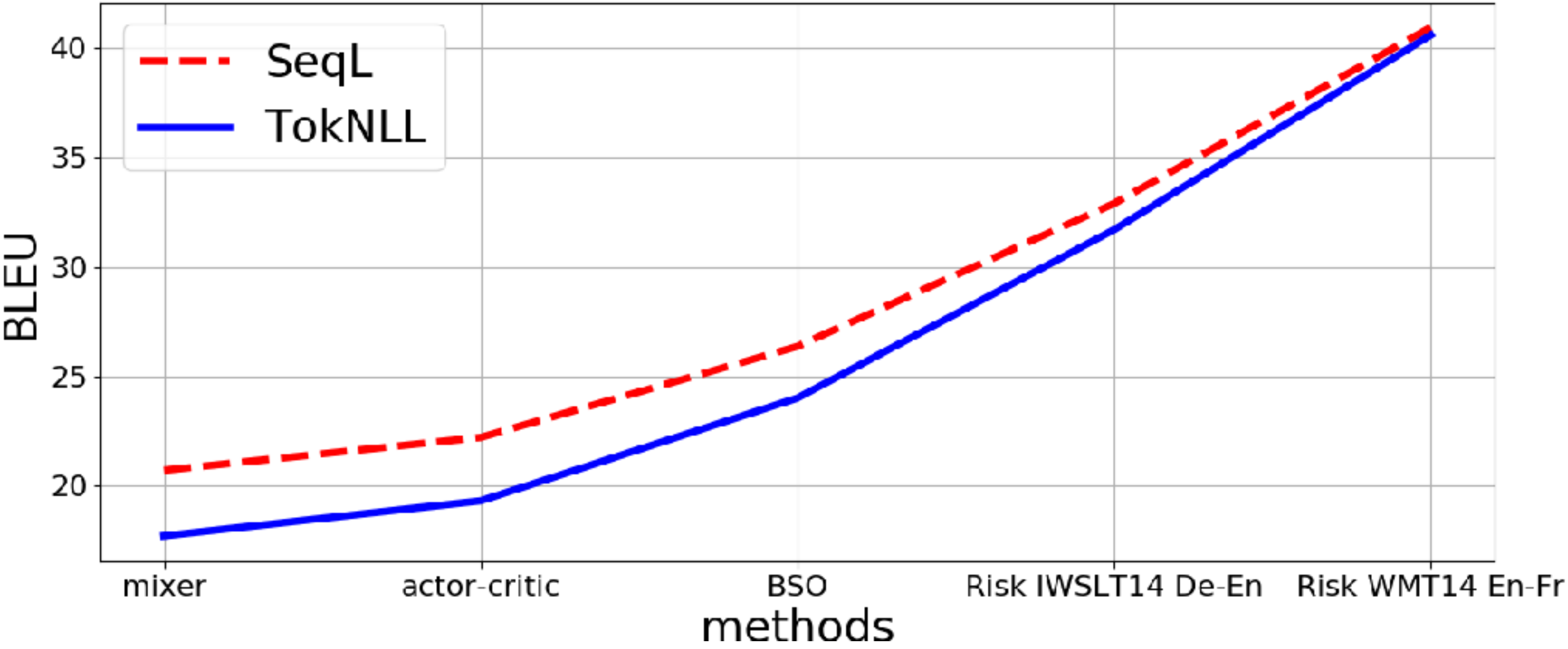
	TEST
TokNLL (Wiseman et al. 2016)	24.0
BSO (Wiseman et al. 2016)	26.4
Our re-implementation of their TokNLL	23.9
Risk on top of the above TokNLL	26.7

Fair Comparison to BSO

	TEST
TokNLL (Wiseman et al. 2016)	24.0
BSO (Wiseman et al. 2016)	26.4
Our re-implementation of their TokNLL	23.9
Risk on top of the above TokNLL	26.7

These methods fare comparably once the baseline is the same…

Diminishing Returns



On WMT’14 En-Fr, TokNLL gets 40.6 while Risk gets 41.0
The stronger the baseline, the less to be gained.

Conclusion

- Sequence level training does improve, but with diminishing returns. It's computationally very expensive.

- The particular method to train at the sequence level does not really matter.

- It's important to use as target the hypothesis in the reachable set that is closest to the reference, as opposed to the reference itself which may not be reachable.

- Sequence level training is more prone to overfitting.

- We should expect big improvements when search is crippled by token level optimization, or if model puts mass int the wrong place or if there is little uncertainty… *but, is this true in NMT?*

Questions?
Вопросы?
¿Preguntas?

Lecture Outline

- Exposure bias/Loss Mismatch: Training at the Sequence Level.
  - how do classical structured prediction losses fare against recent proposals?
  - how much to be gained by fixing this inconsistency?
- Analyzing Uncertainty: model fitting and effects on search.
  - why do larger beam perform worse?
  - why is the model under-estimating rare words?
- Training Without Supervision.
  - how to leverage monolingual data?
  - can we learn without any parallel sentence?

Lecture Outline

- Analyzing Uncertainty: model fitting and effects on search.
  - why do larger beam perform worse?
  - why is the model under-estimating rare words?

Analyzing uncertainty in neural machine translation
Myle Ott, Michael Auli, David Grangier, Marc'Aurelio Ranzato
https://arxiv.org/abs/1803.00047

Questions

- what are the sources of uncertainty in the data?
- do NMT models capture such uncertainty?
- does uncertainty hinder search?
- what tools can we use to measure uncertainty?

Goal

BETTER UNDERSTANDING

E.g.:

- rare word under-estimation
  - artifact of beam search (argmax)?
  - due to exposure bias?
  - due to poor estimation?
- wider beam degradation
  - due to heuristic nature of beam search?
  - is the model poorly trained?
- model fitting
  - are NMT models calibrated?
  - what do NMT models over/under-estimate?

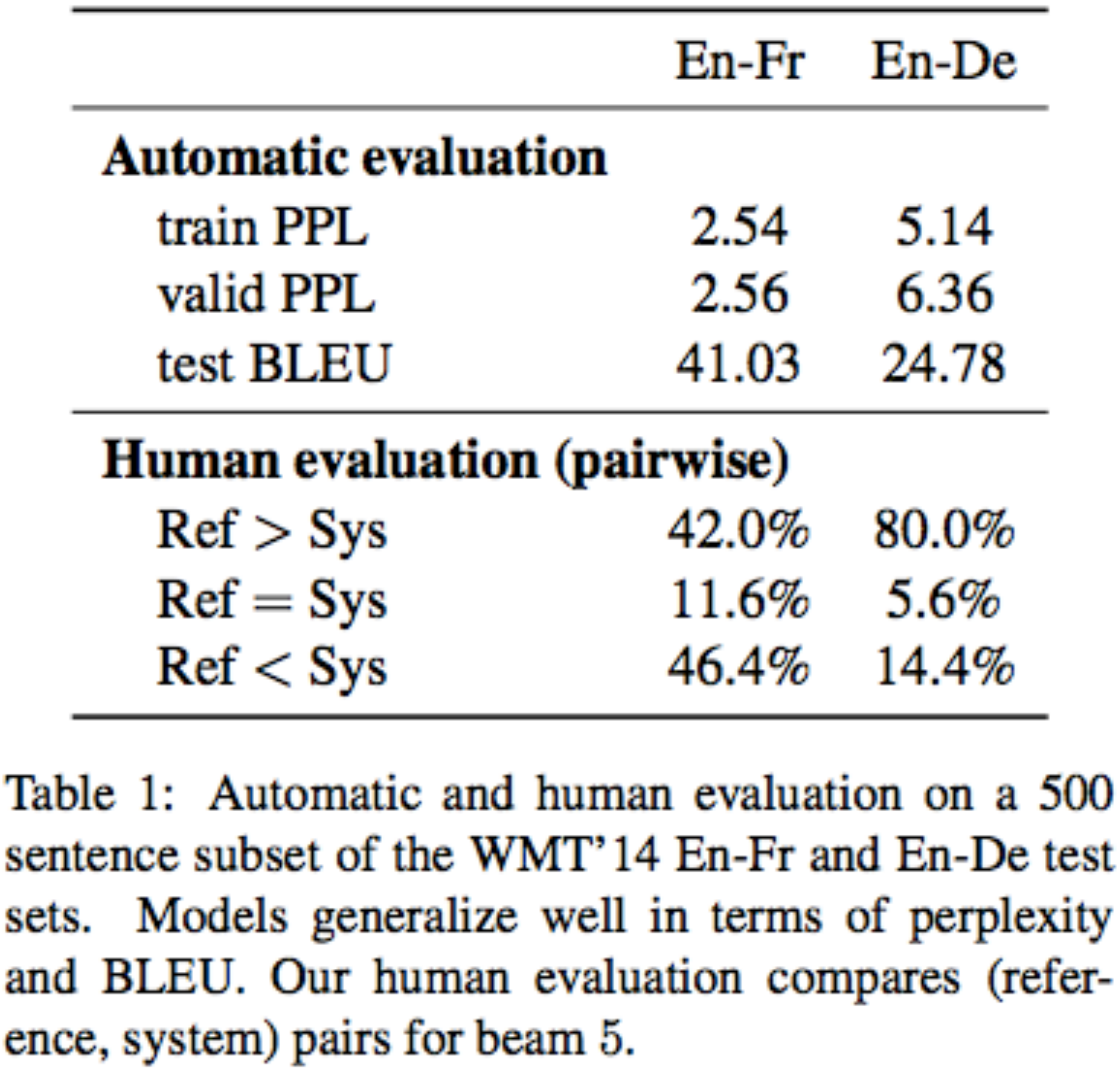
Datasets

	Nr. Sentences	Vocab Size (BPE)
WMT14 En-De	4.5M	40K
WMT17 En-De	5.9M	40K
WMT14 En-Fr	35.5M	40K

Model

- Convolutional NMT with attention
- 15 layers
- 768D embeddings
- ~250M parameters

Evaluating NMT

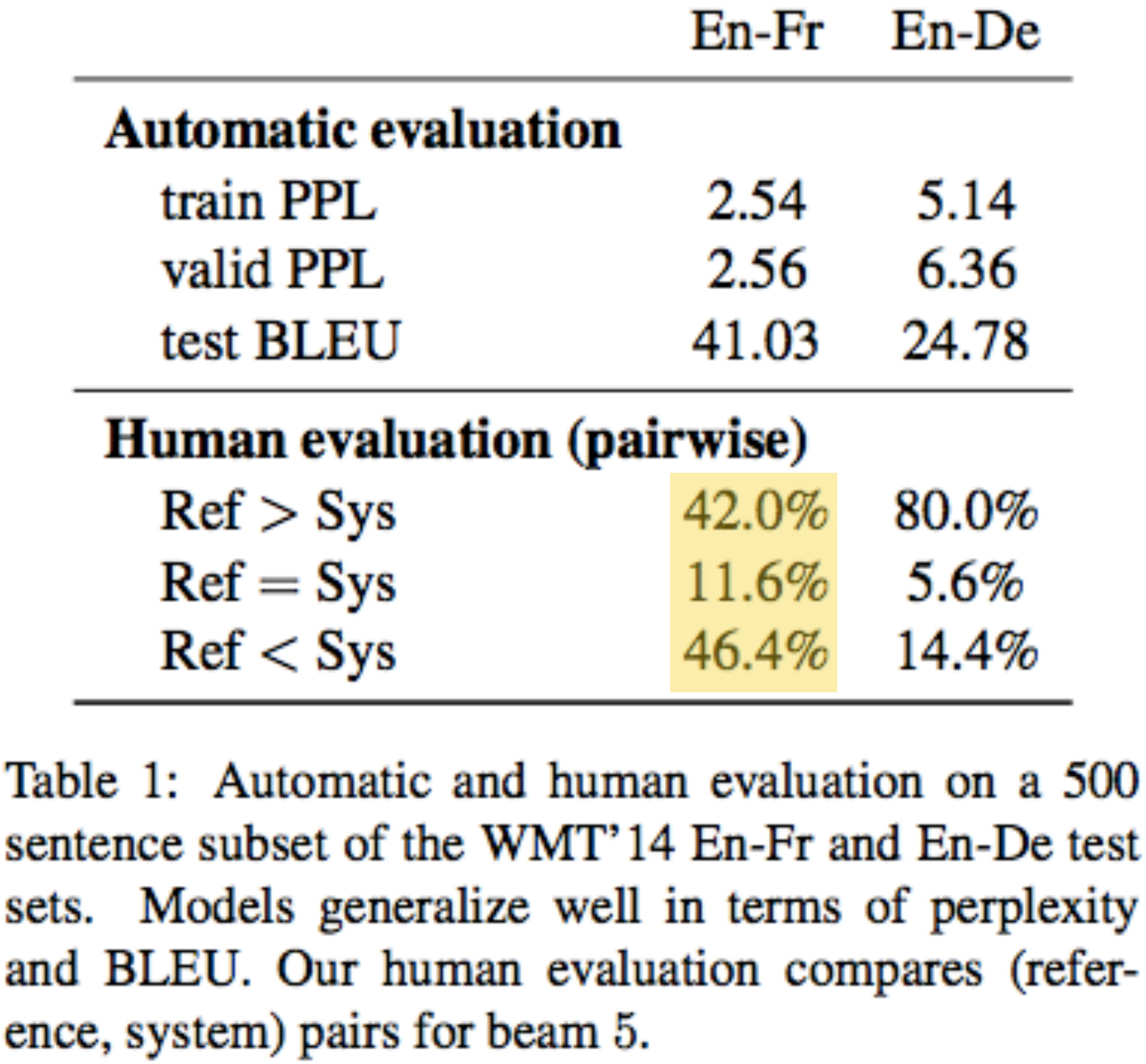


	En-Fr	En-De
Automatic evaluation		
train PPL	2.54	5.14
valid PPL	2.56	6.36
test BLEU	41.03	24.78
Human evaluation (pairwise)		
Ref > Sys	42.0%	80.0%
Ref = Sys	11.6%	5.6%
Ref < Sys	46.4%	14.4%

Table 1: Automatic and human evaluation on a 500 sentence subset of the WMT'14 En-Fr and En-De test sets. Models generalize well in terms of perplexity and BLEU. Our human evaluation compares (reference, system) pairs for beam 5.

Evaluating NMT

Model is very well trained, particularly in En-Fr dataset.



	En-Fr	En-De
Automatic evaluation		
train PPL	2.54	5.14
valid PPL	2.56	6.36
test BLEU	41.03	24.78
Human evaluation (pairwise)		
Ref > Sys	42.0%	80.0%
Ref = Sys	11.6%	5.6%
Ref < Sys	46.4%	14.4%

Table 1: Automatic and human evaluation on a 500 sentence subset of the WMT'14 En-Fr and En-De test sets. Models generalize well in terms of perplexity and BLEU. Our human evaluation compares (reference, system) pairs for beam 5.

Outline

- Data uncertainty
- Search
- Analyzing the model distribution

Data Uncertainty

- Intrinsic
    - there are many semantically equivalent translations of the same sentence. E.g.: style, skipping prepositions, choice of words, structural choices (active/passive tense), etc.

EXAMPLE
Source: The night before would be practically sleepless .

Target #1: La nuit qui précède pourrait s’avérer quasiment blanche .
Target #2: Il ne dormait pratiquement pas la nuit précédente .
Target #3: La nuit précédente allait être pratiquement sans sommeil .
Target #4: La nuit précédente , on n’a presque pas dormi .
Target #5: La veille , presque personne ne connaitra le sommeil .

Data Uncertainty

- Intrinsic
  - there are many semantically equivalent translations of the same sentence. E.g.: style, skipping prepositions, choice of words, structural choices (active/passive tense), etc.
  - under-specification. E.g.: gender, tense, number, etc.

EXAMPLE
Source: nice .

Target #1: chouette .
Target #2: belle .
Target #3: beau .

Data Uncertainty

- Intrinsic
  - there are many semantically equivalent translations of the same sentence. E.g.: style, skipping prepositions, choice of words, structural choices (active/passive tense), etc.
  - under-specification. E.g.: gender, tense, number, etc.
- Extrinsic
  - noise in the data. E.g.: partial translation, copies of the source, etc.

Example: on WMT between 1 and 2% of the training target sentences are copies of the source.

Data Uncertainty

- Intrinsic
  - there are many semantically equivalent translations of the same sentence. E.g.: style, skipping prepositions, choice of words, structural choices (active/passive tense), etc.
  - under-specification. E.g.: gender, tense, number, etc.

HOW DOES THIS AFFECT NMT?

- Extrinsic
  - noise in the data. E.g.: partial translation, copies of the source, etc.

Outline

- Data uncertainty
- **Search**
- Analyzing the model distribution

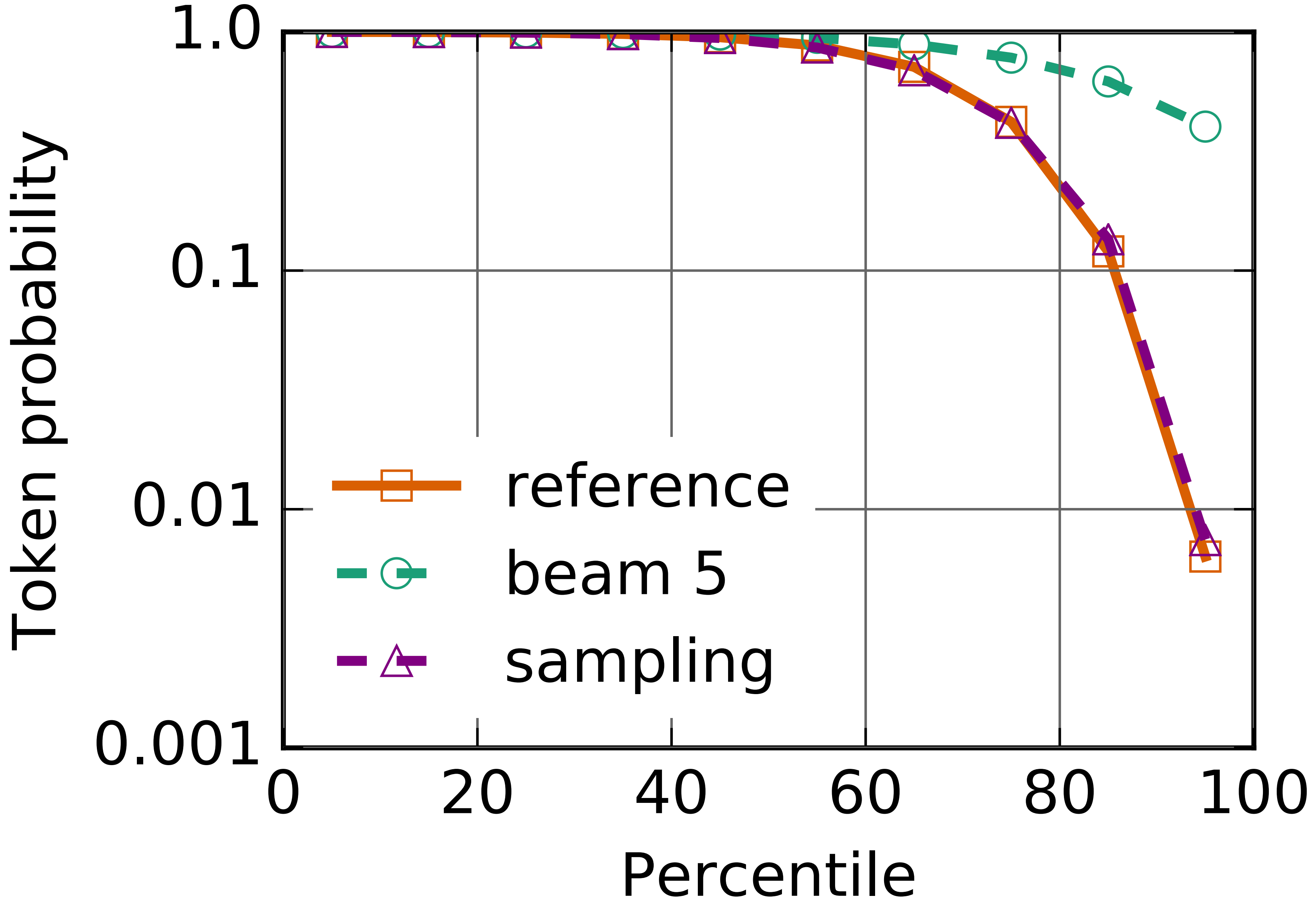
Search

Search aims at finding the most likely sequence according to the model: $\arg\max_{y} p(y|x;\theta)$

Preliminary questions:

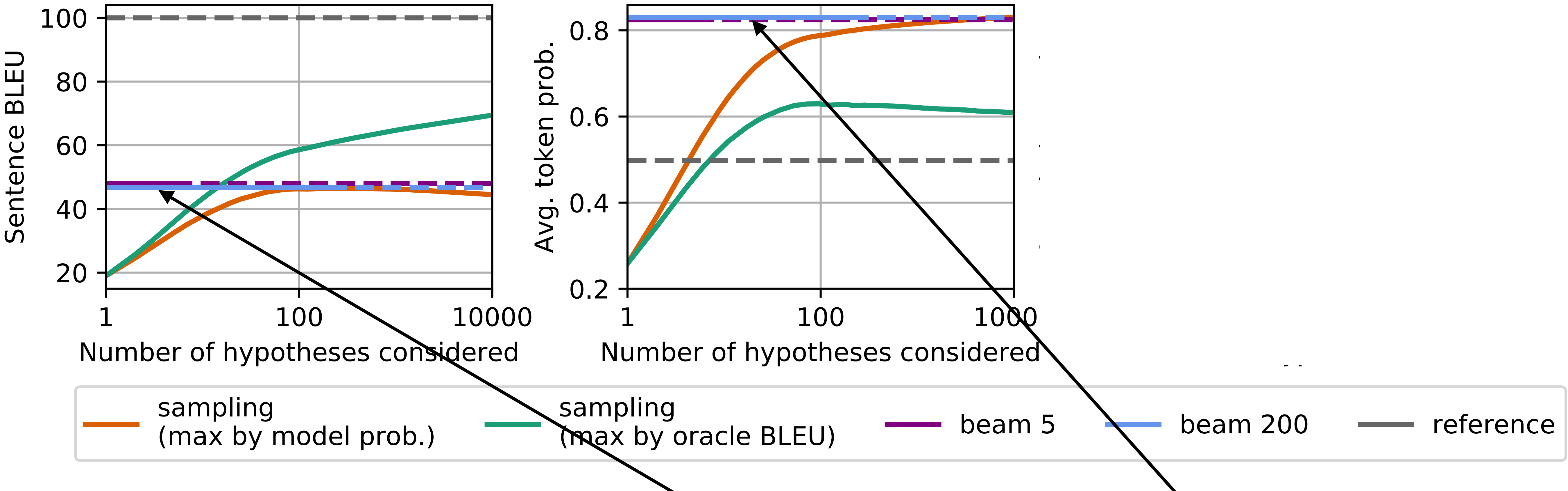
- is beam search effective?
- is beam search efficient?
- are there better search strategies?

Search



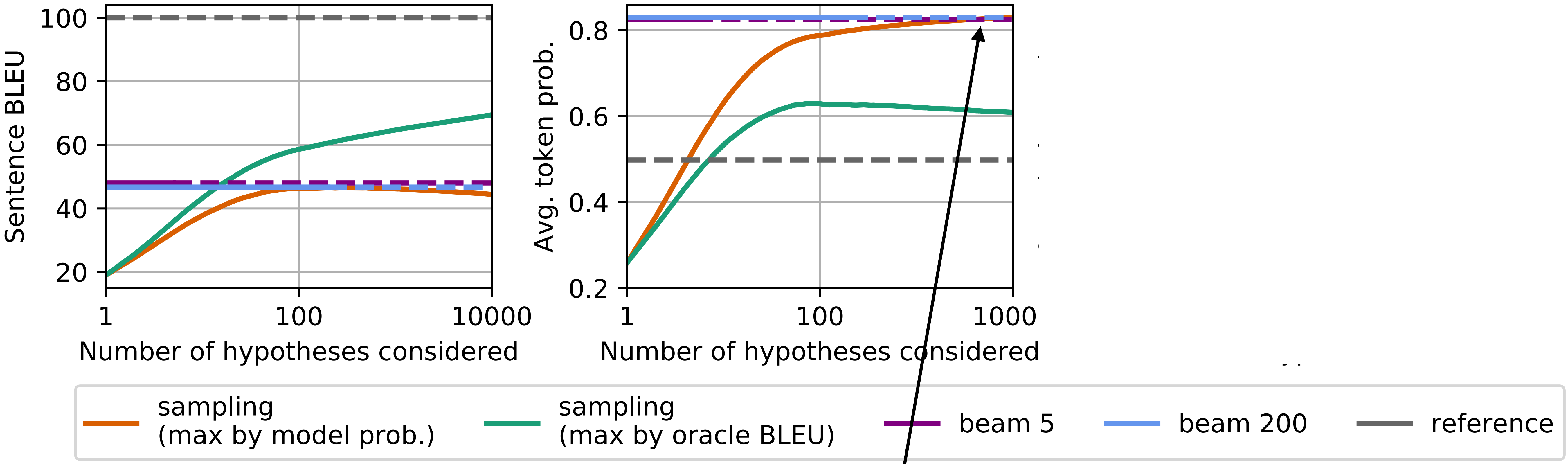
Beam search is very effective; only 20% of the tokens with probability < 0.7 (despite exposure bias)!

Search



- Increasing the beam width does not increase BLEU, while probability increases.

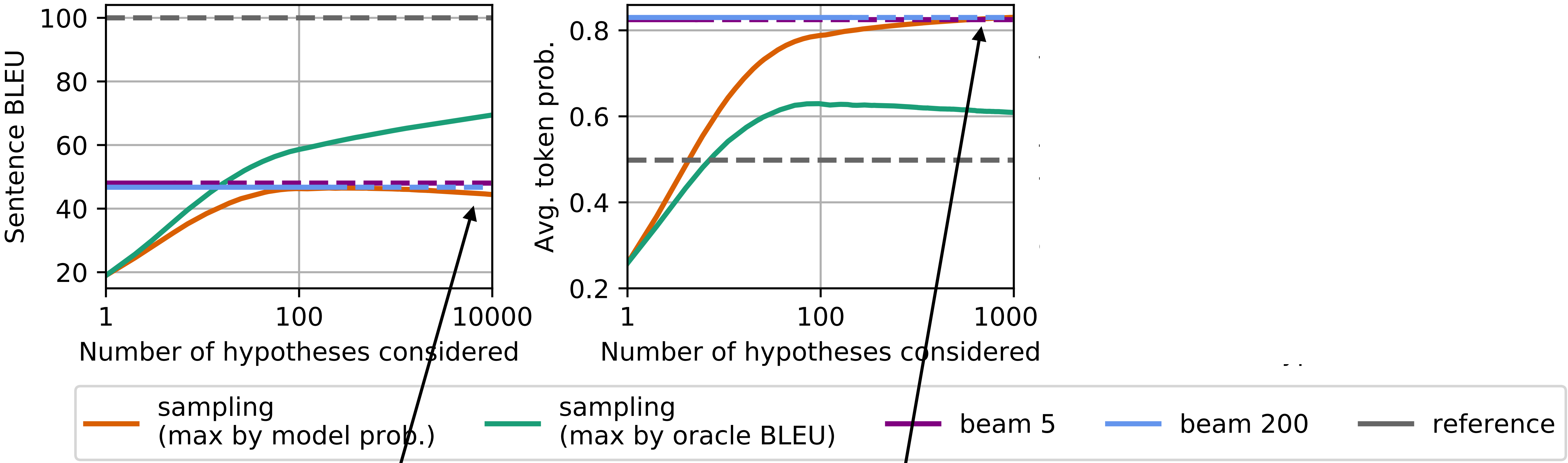
Search



- Increasing the beam width does not increase BLEU, while probability increases.

- Sampling can find hypotheses with similar logprob but:

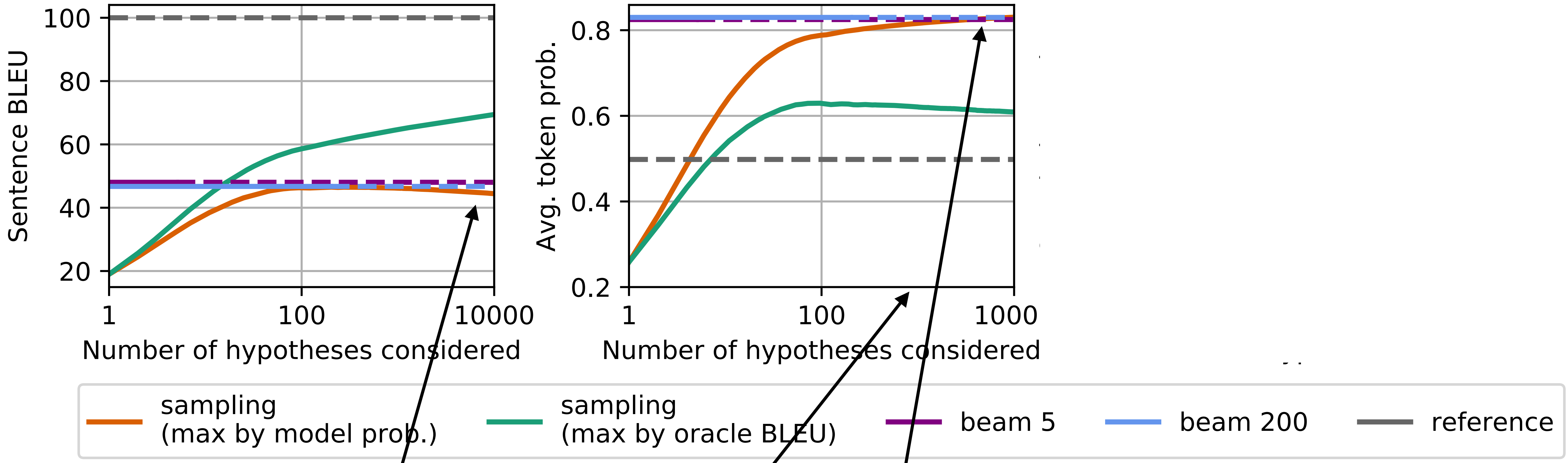
Search



- Increasing the beam width does not increase BLEU, while probability increases.

- Sampling can find hypotheses with similar logprob but:
    - lower BLEU

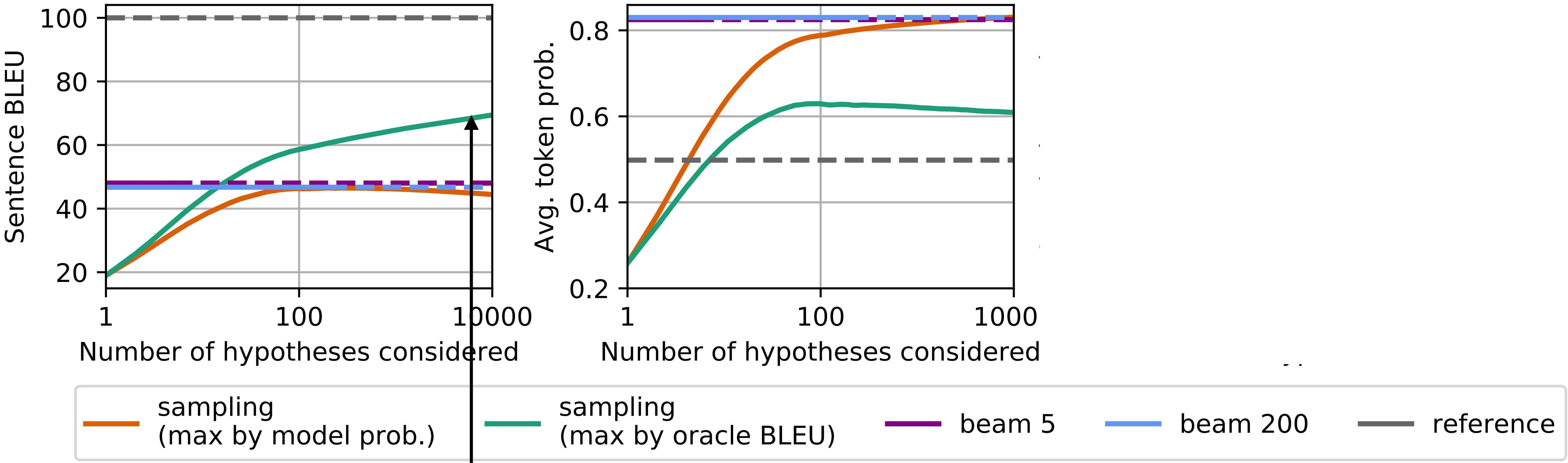
Search



- Increasing the beam width does not increase BLEU, while probability increases.
- Sampling can find hypotheses with similar logprob but:
  - lower BLEU
  - it’s 20x less efficient

M. Ranzato

Search

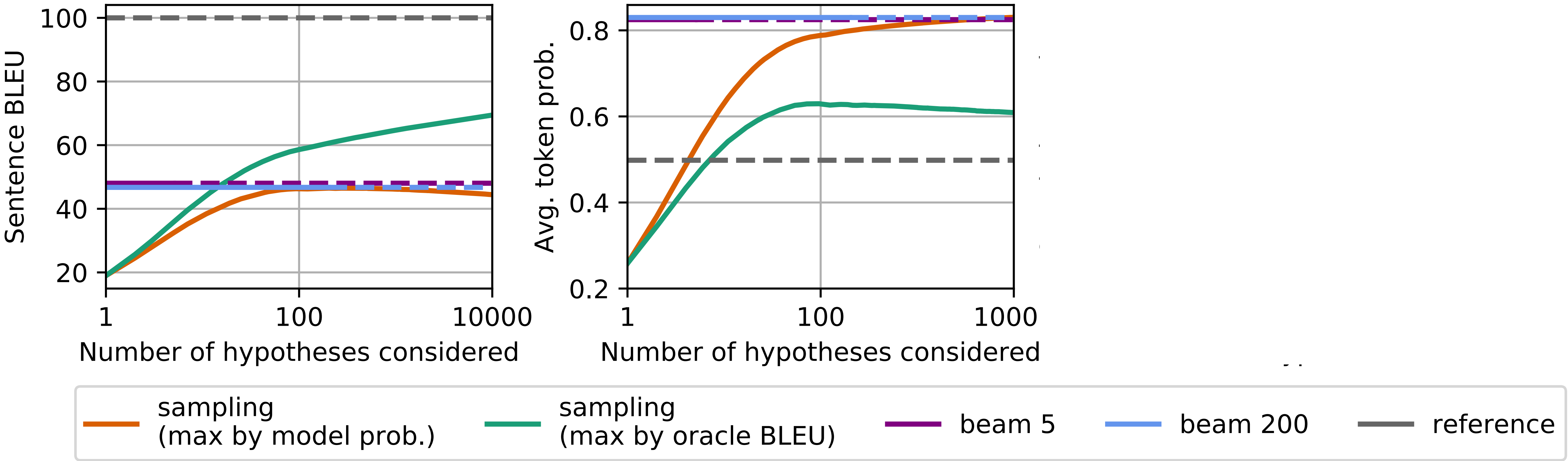


- Increasing the beam width does not increase BLEU, while probability increases.

- Sampling can find hypotheses with similar logprob but…

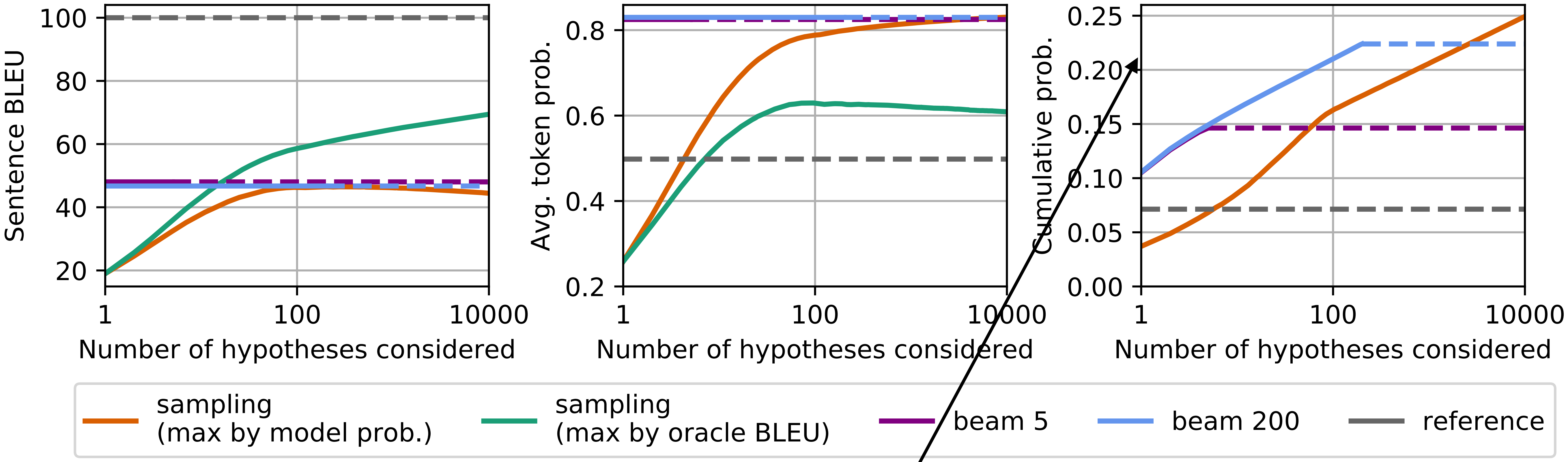
- Among the generated hypotheses, there exist at least one that is pretty close to the reference.

Search



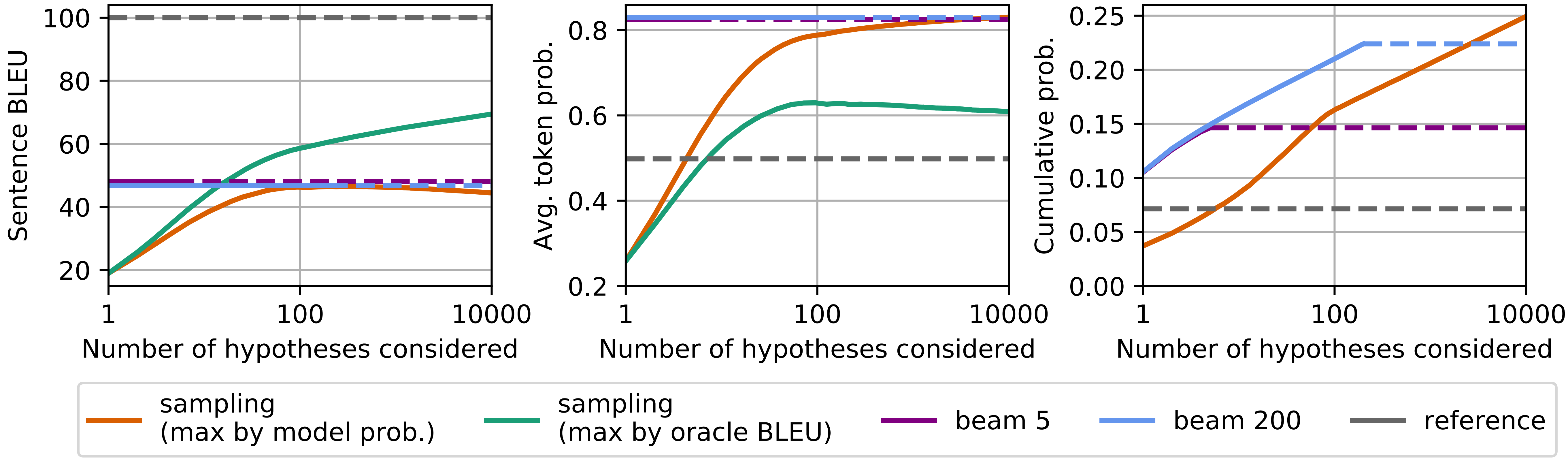
Beam search is very effective and efficient.
However, large beams yield worse BLEU!

Search



- Beam 200/sampling 10K cover only about 22% of the total probability mass; where is the rest?

Search

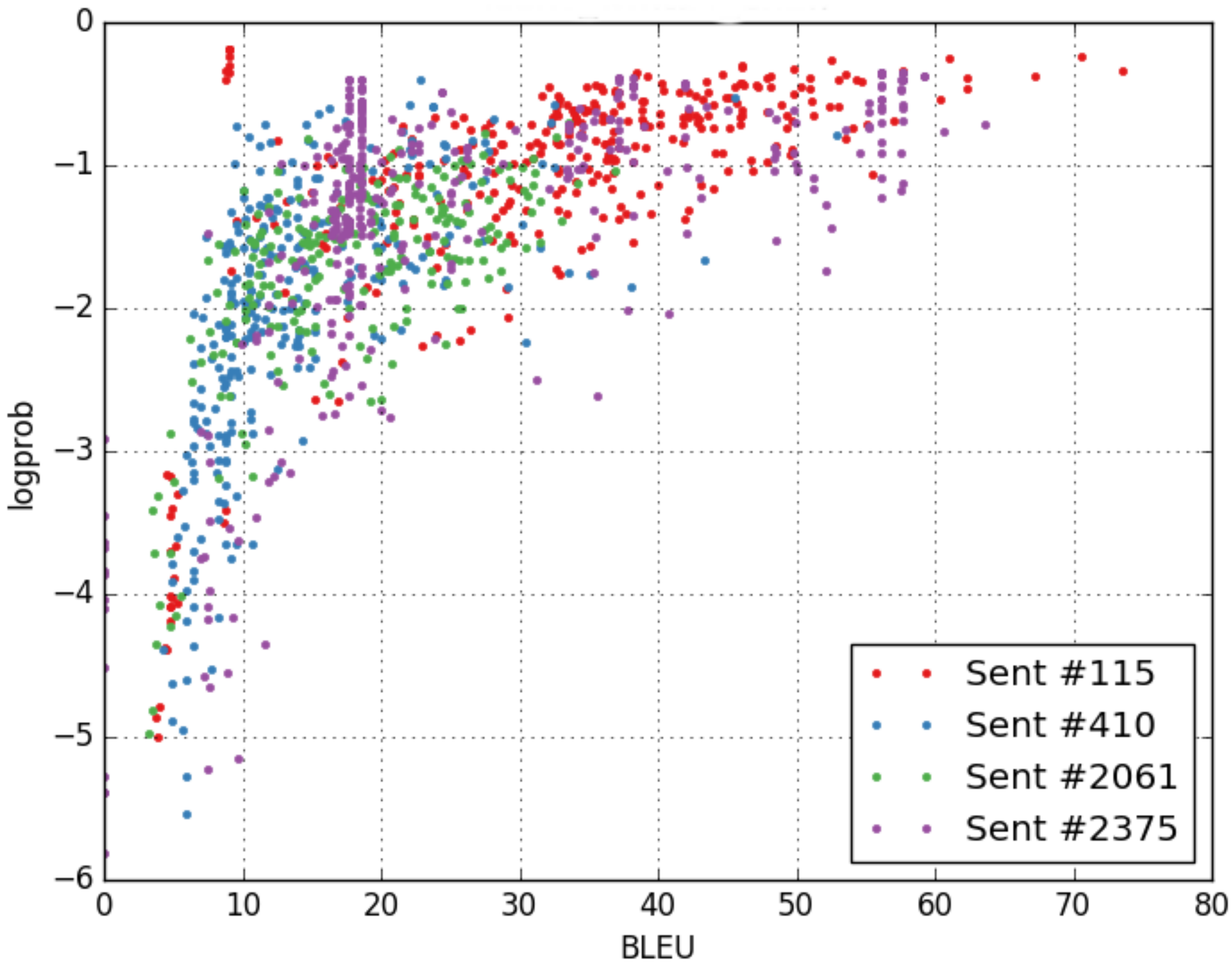


Model distribution has a lot of uncertainty.

Puzzling Observations

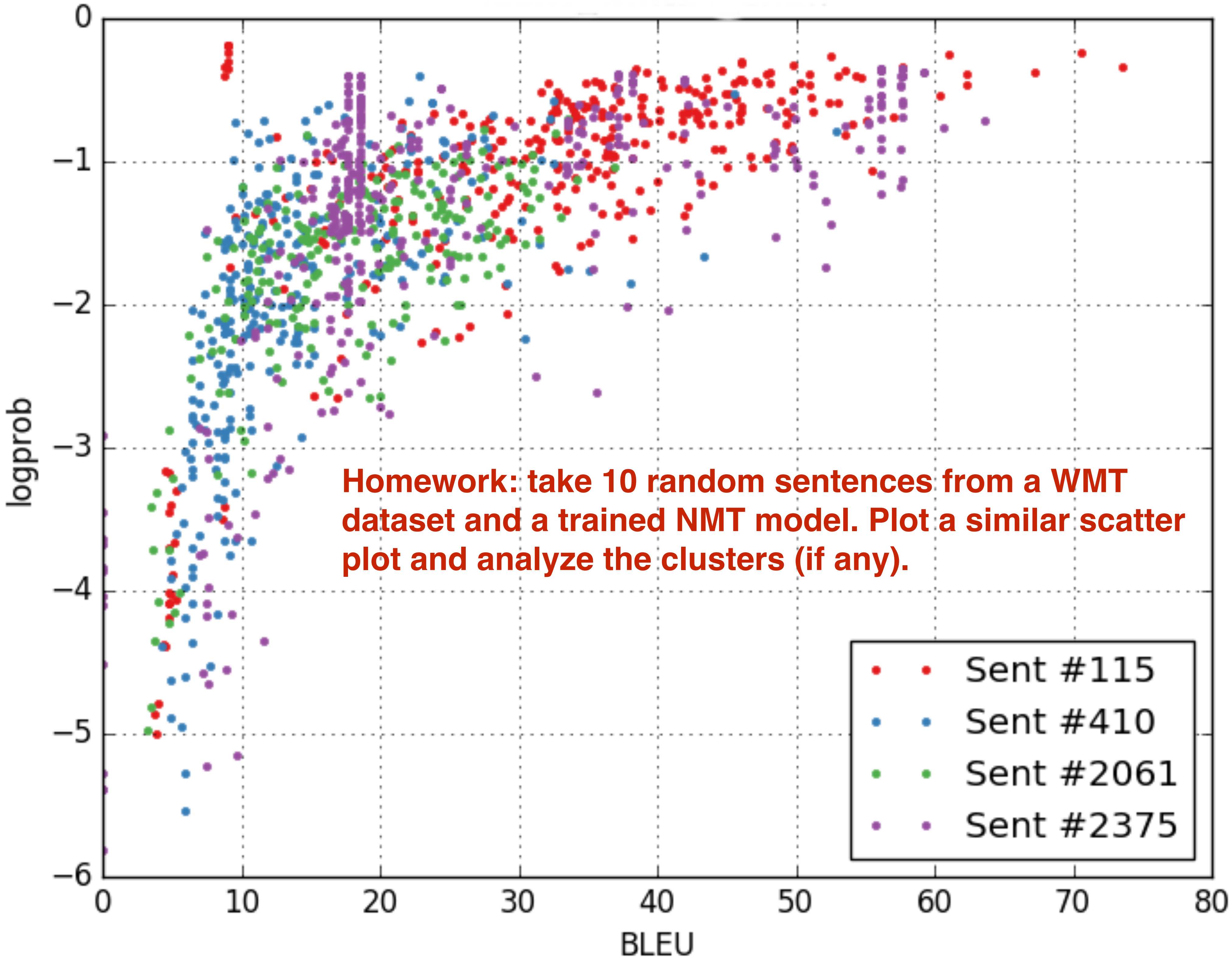
- Increasing beam width after a certain point hurts performance in terms of BLEU.
- Large beam accounts only for fraction of total probability mass.

Hint: Scatter Plot of Samples



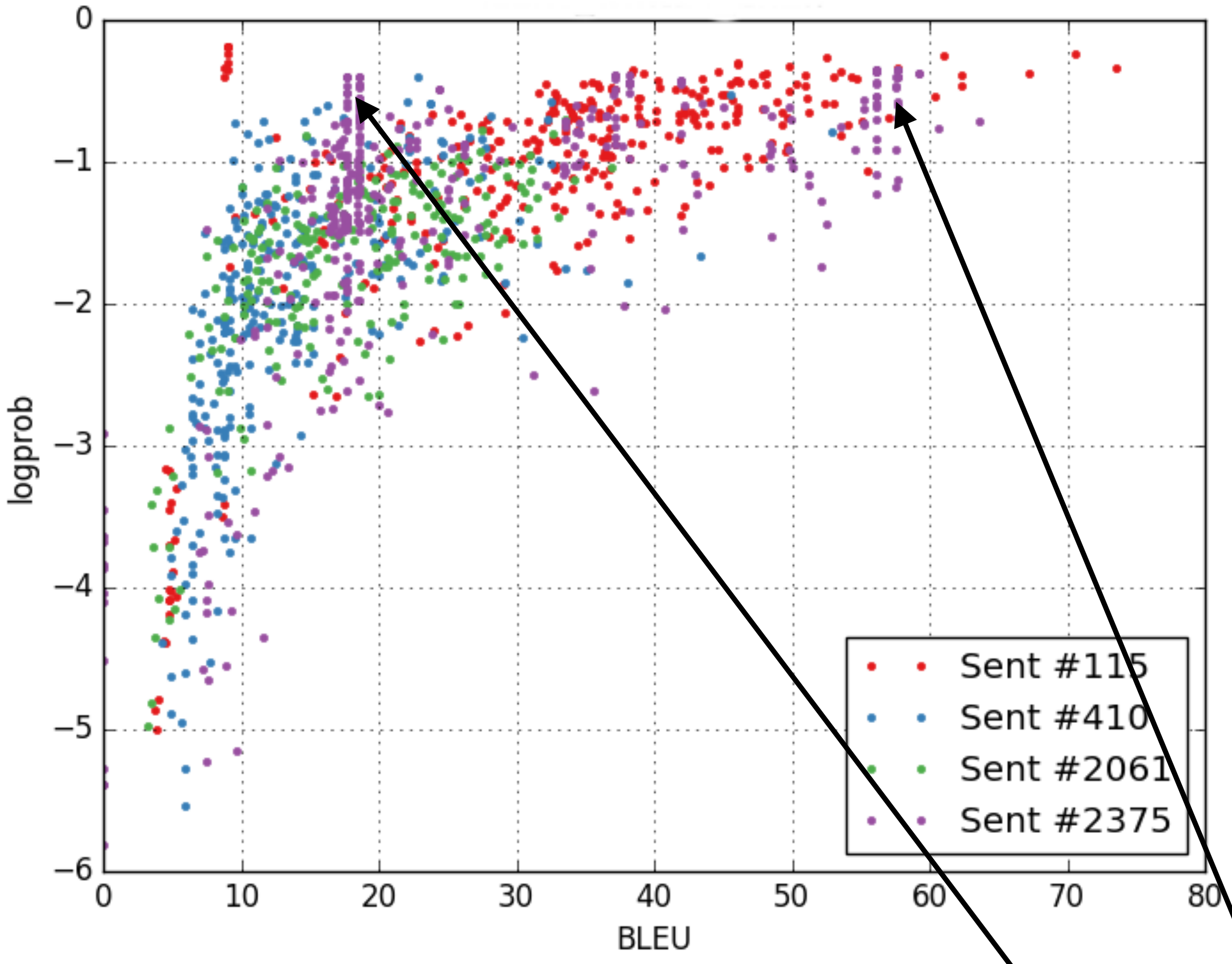
M. Ranzato

Hint: Scatter Plot of Samples



Homework: take 10 random sentences from a WMT dataset and a trained NMT model. Plot a similar scatter plot and analyze the clusters (if any).

Hint: Scatter Plot of Samples



Source #2375 (purple):

Should this election be decided two months after we stopped voting?

Target #2375 (purple):

Cette élection devrait-elle ëtre décidé deux mois après que le vote est terminé?

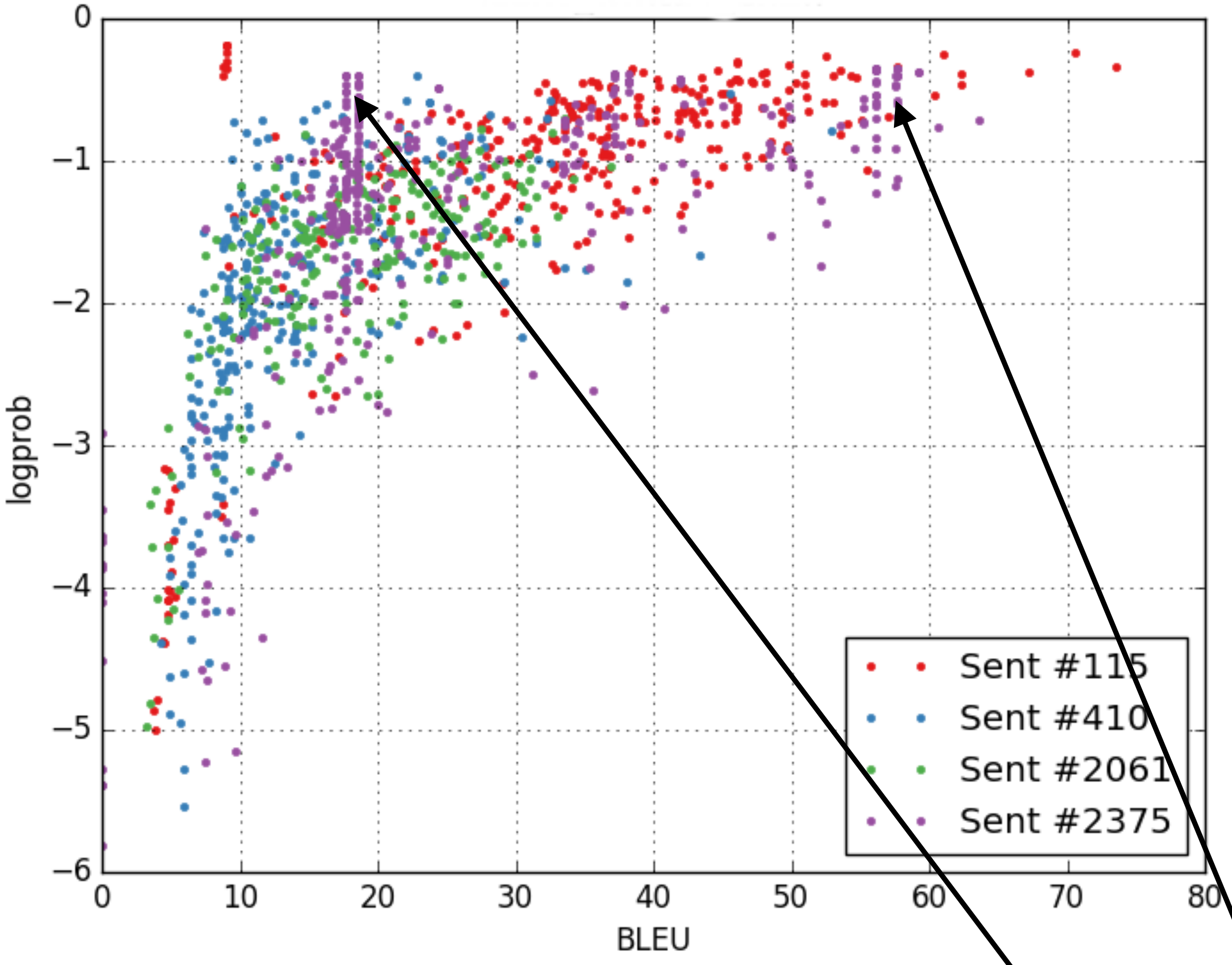
High-BLEU sample:

Cette élection devrait-elle ëtre décidée deux mois après l'arrêt du scrutin?

Low-BLEU sample:

Ce choix devrait-il ëtre décidé deux mois après la fin du vote?

Hint: Scatter Plot of Samples



Source #2375 (purple):

Should this election be decided two months after we stopped voting?

Target #2375 (purple):

Cette élection devrait-elle ëtre
décidé deux mois après que le vote est terminé?

High-BLEU sample:

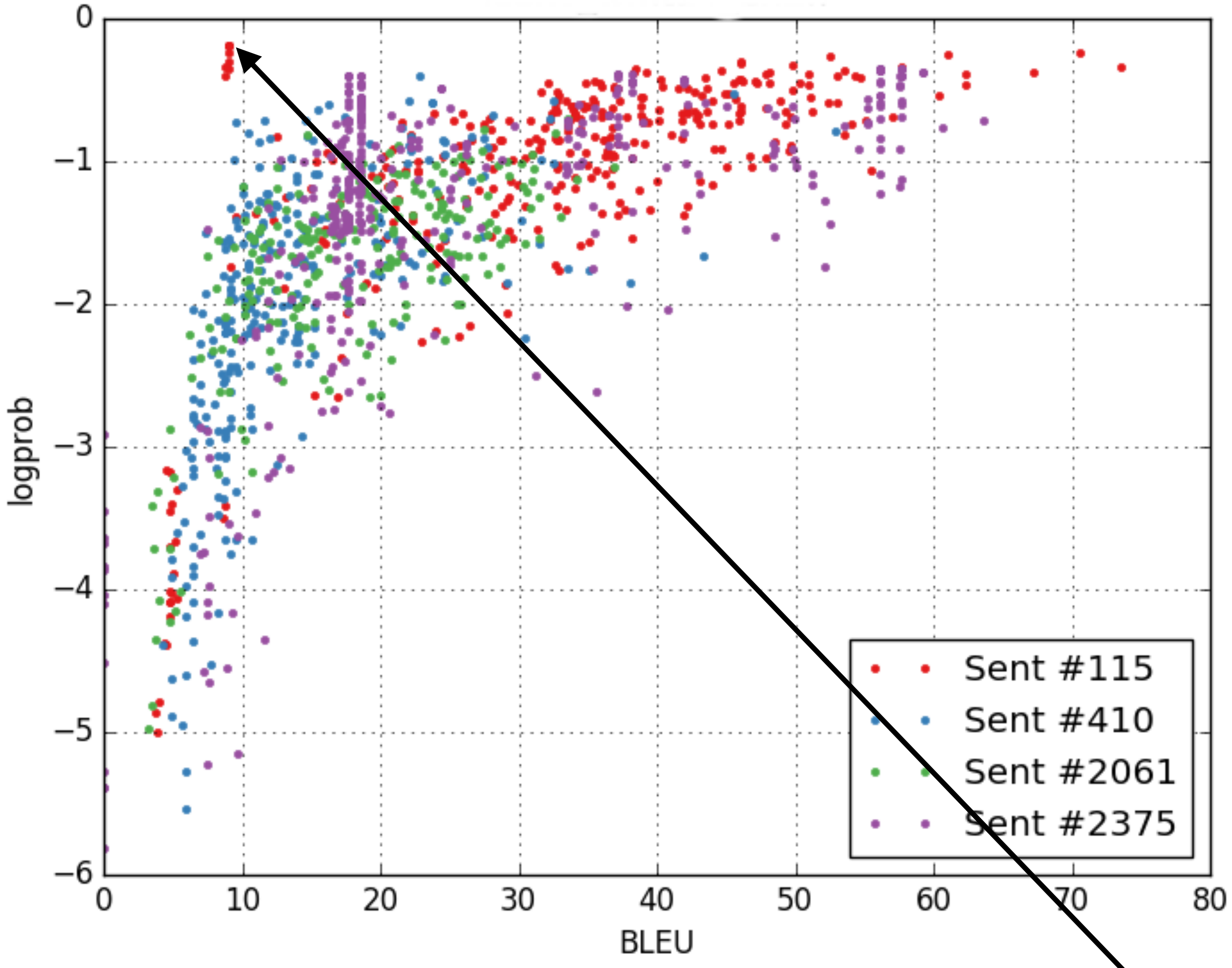
Cette élection devrait-elle ëtre
décidée deux mois après l'arrêt du scrutin?

Low-BLEU sample:

Ce choix devrait-il ëtre décidé deux
mois après la fin du vote?

BLEU is just a poor metric.

Hint: Scatter Plot of Samples



Source #115 (red):

The first nine episodes of Sheriff [unk]'s Wild West will be available from November 24 on the site [unk] or via its application for mobile phones and tablets.

Target #115 (red):

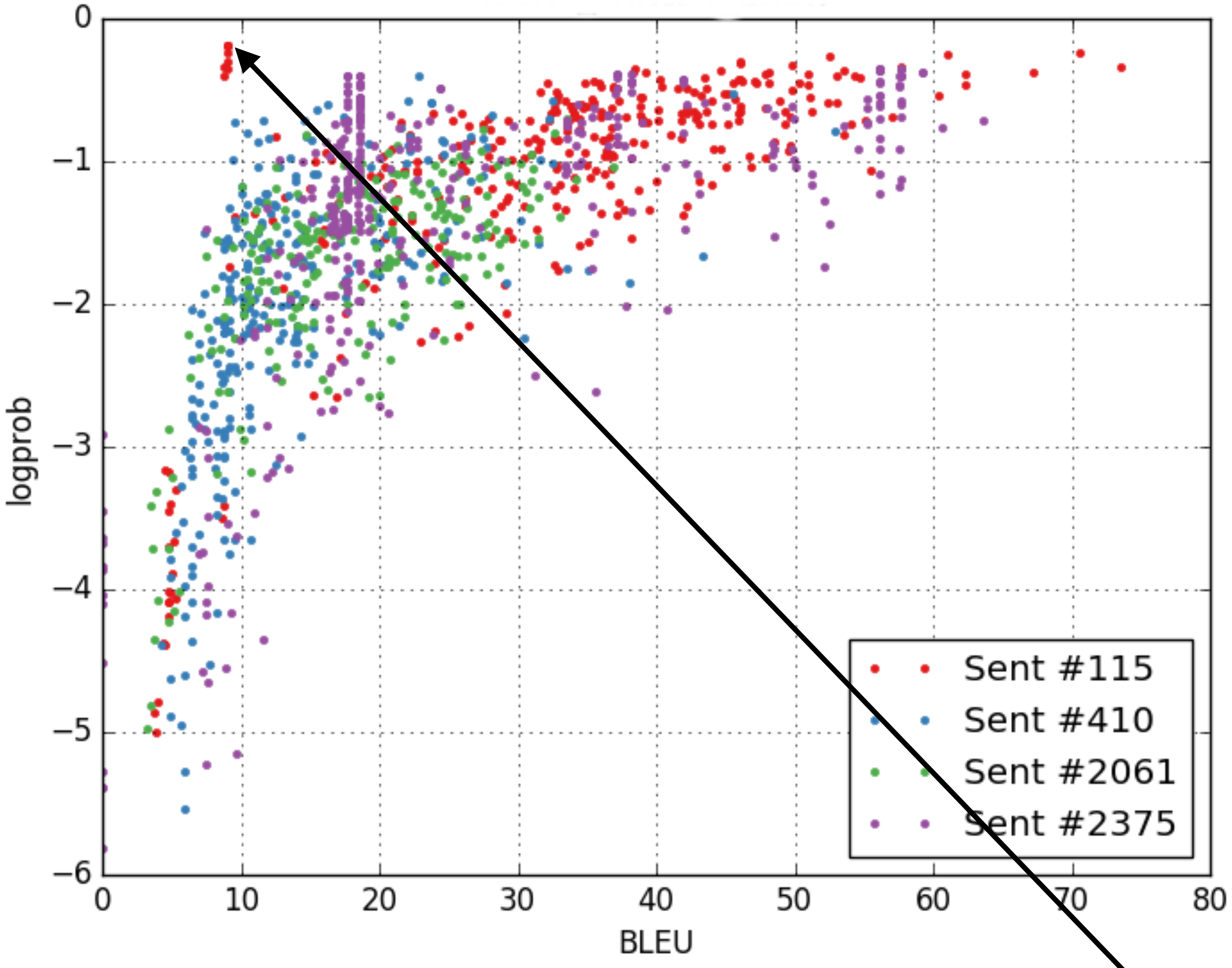
Les neuf premiers épisodes de [unk] [unk] s Wild West seront disponibles à partir du 24 novembre sur le site [unk] ou via son application pour téléphones et tablettes.

High-logp low BLEU sample:

The first nine episodes of Sheriff [unk] s Wild West will be available from November 24 on the site [unk] or via its application for mobile phones and tablets.

M. Ranzato

Hint: Scatter Plot of Samples



Source #115 (red):

The first nine episodes of Sheriff [unk]'s Wild West will be available from November 24 on the site [unk] or via its application for mobile phones and tablets.

Target #115 (red):

Les neuf premiers épisodes de [unk] [unk] s Wild West seront disponibles à partir du 24 novembre sur le site [unk] ou via son application pour téléphones et tablettes.

High-logp low BLEU sample:

The first nine episodes of Sheriff [unk] s Wild West will be available from November 24 on the site [unk] or via its application for mobile phones and tablets.

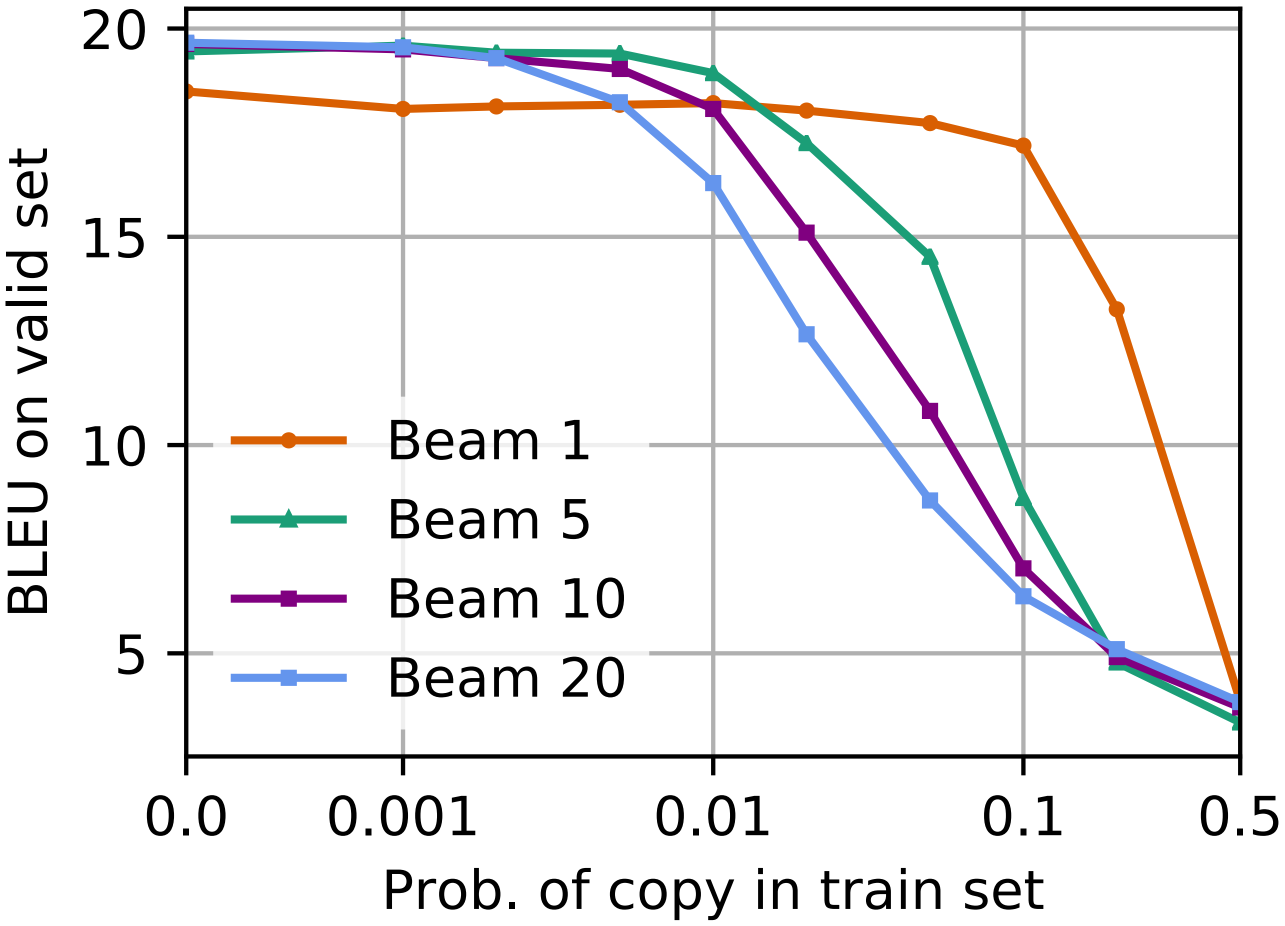
Model generates copies of source sentence!
Why does beam find this?

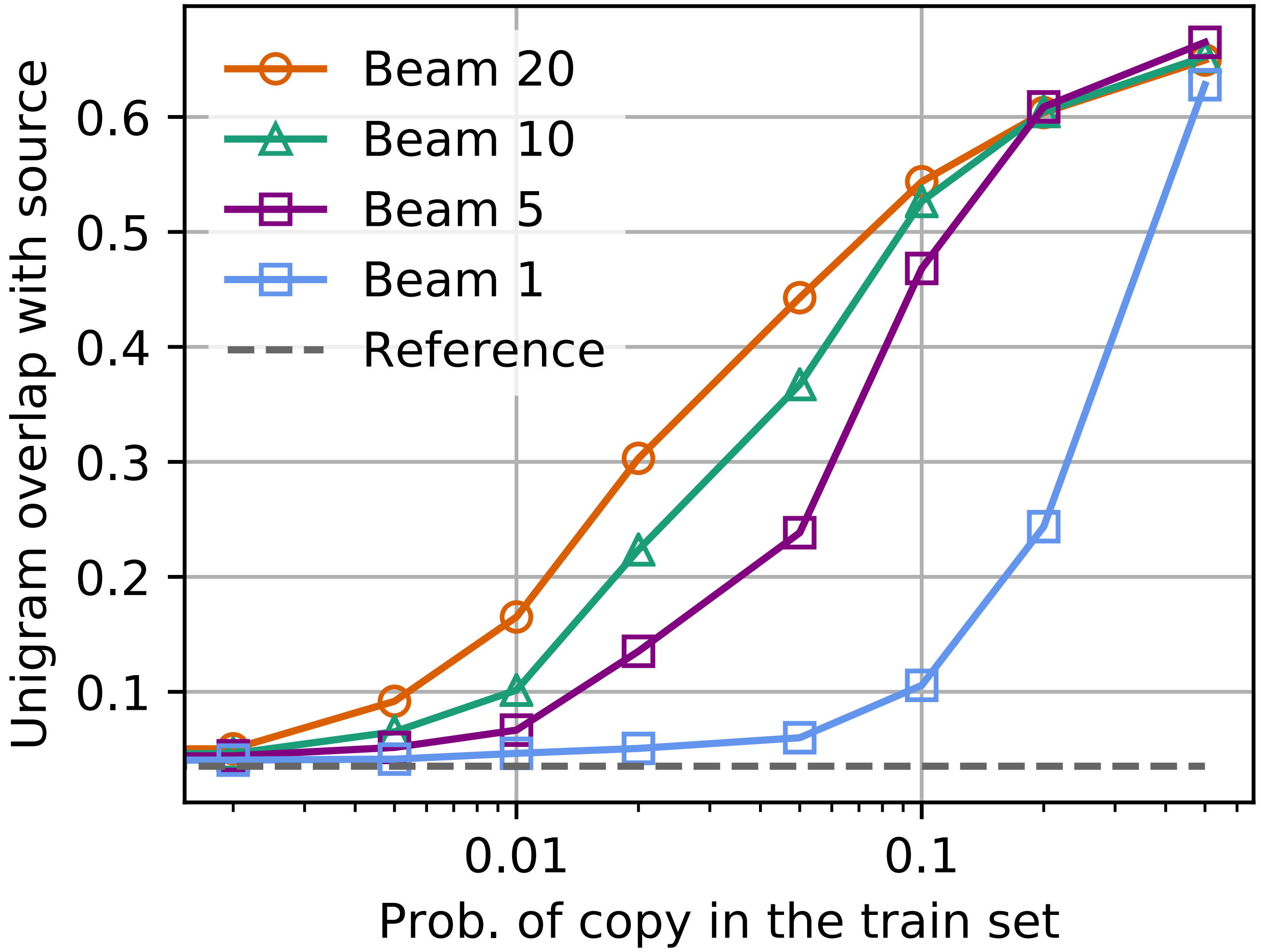
M. Ranzato

Uncertainty <—> Search

- Hard to characterize how uncertainty affects search in general.
- We can however simulate (extrinsic) uncertainty:
  - add fraction of “copy noise” and check effects on search.

Uncertainty <—> Search





Large beams are more prone to copy the source, hence the lower BLEU.

M. Ranzato

Uncertainty <—> Search

- Source: The first nine episodes of Sheriff <unk> ’s Wild West will be available from November 24 on the site <unk> or via its application for mobile phones and tablets .
- Target (reference): Les neuf premiers épisodes de <unk> <unk> s Wild West seront disponibles à partir du 24 novembre sur le site <unk> ou via son application pour téléphones et tablettes .
- Sample: The first nine episodes of Sheriff <unk> s Wild West will be available from November 24 on the site <unk> or via its application for mobile <unk> and tablets .

Uncertainty <—> Search

- **Source**: The first nine episodes of Sheriff <unk> ′s Wild West will be available from November 24 on the site <unk> or via its application for mobile phones and tablets .
- **Target (reference)**: Les neuf premiers épisodes de <unk> <unk> s Wild West seront disponibles à partir du 24 novembre sur le

log probs: -4.53 -0.02 -0.28 -0.11 -0.01 -0.001 -0.004 -0.002 -0.001 -0.005

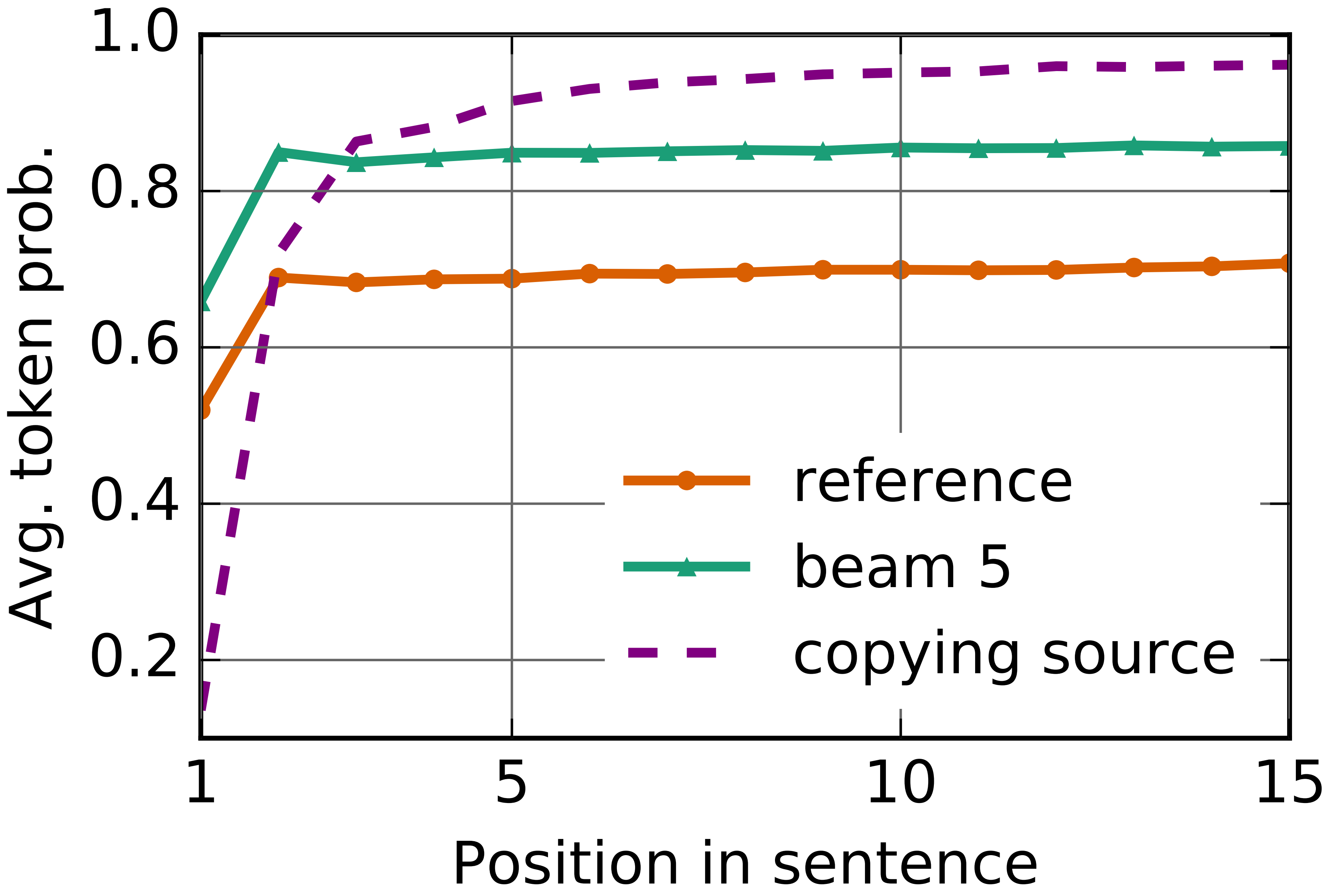
tablettes .

- **Sample**: The first nine episodes of Sheriff <unk> s Wild West will be available from November 24 on the site <unk> or via its application for mobile <unk> and tablets .

Inductive bias alert:

NMT + attention has easy time to learn how to copy!

Uncertainty <—> Search



Initial tokens pay big penalty, but afterwards copying the source is cheap. Only large beams can discover this.

Uncertainty <—> Search

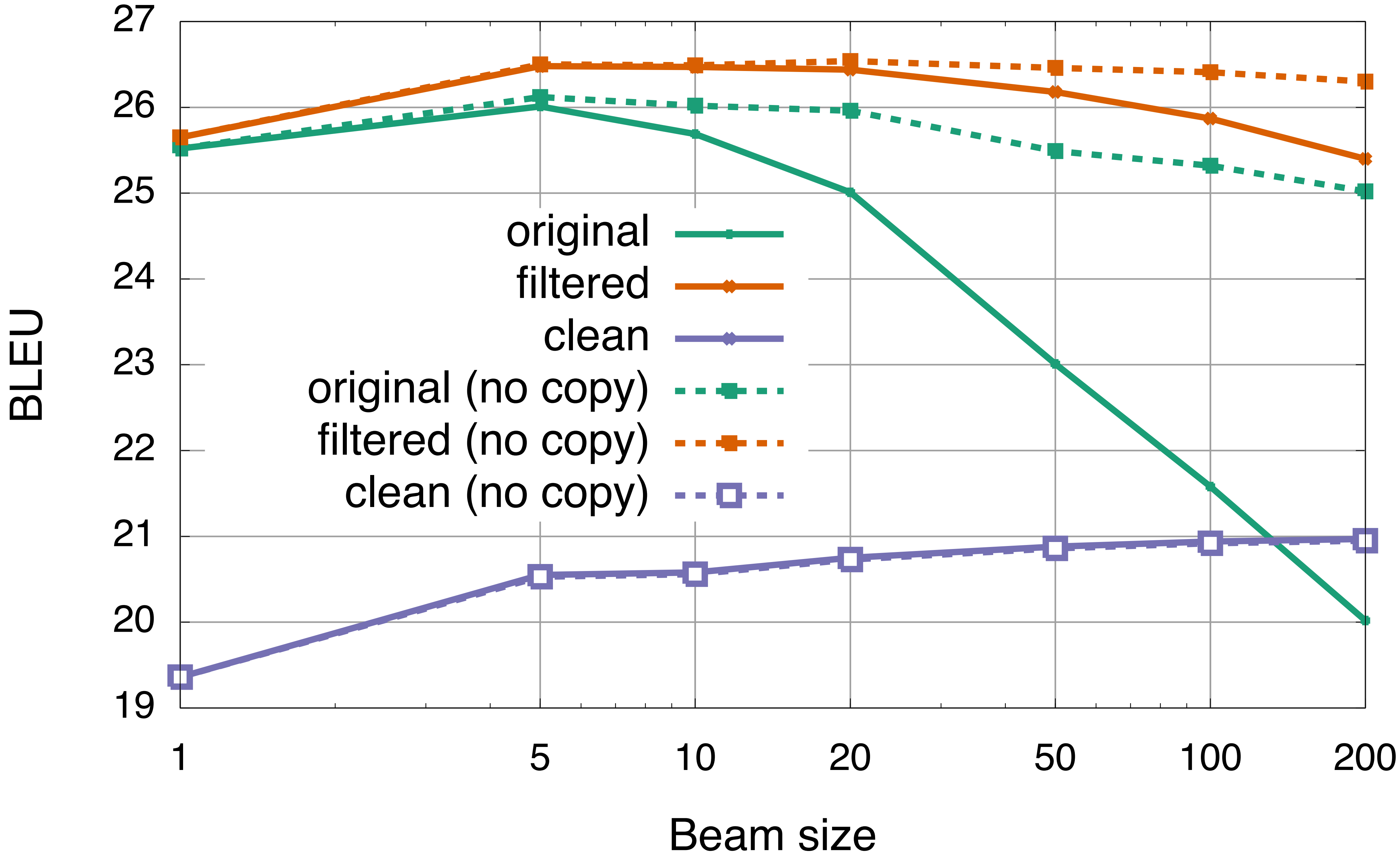
On WMT'14 En-Fr, we estimate that ~2% of the training target sentences are copies of the corresponding source.

Beam@1 yields copies 2.6% of the times.
Beam@20 yields copies 3.5% of the times.

Fixing Search

- Filtering the data with model trained on “clean data” to remove copies from training set.
- Constraining beam search not to output too many words from the source sentence.

Fixing Search



Search & Uncertainty

- Search works very well, i.e. beam finds likely hypotheses according to the model.
- However, it can find spurious sentences (model is wrong), that are merely due to noise in the data collection process.
- This explains why BLEU deteriorates for large beams.
- There are easy fixes.

Puzzling Observations

- Increasing beam width after a certain point hurts performance in terms of BLEU.

Understood

- Large beam accounts only for fraction of total probability mass.

Outline

- Data uncertainty
- Preliminaries
- Search
- **Analyzing the model distribution**

Model Distribution

- Checking match between model and data distribution is challenging because:
    - For a given source sentence, we typically observe only one sample from the data distribution (the provided reference).
    - Enumeration of all possible sequences using the data distribution is intractable anyway.

Model Distribution

We would like to:

- check how closely model and data distribution match

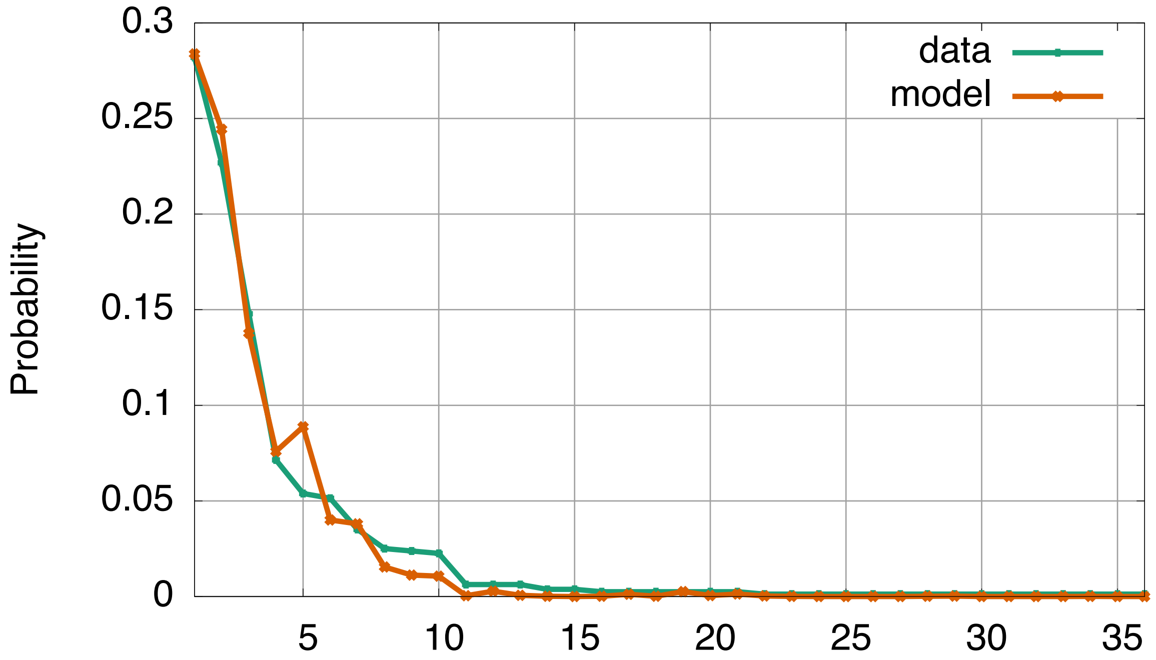
- understand when they differ and why

Anecdotal Example

In the training set there are some source sentences that appear multiple times. Use corresponding targets to estimate the underlying distribution!

EXAMPLE
Source: (The president cutoff the speaker) .

Appears 798 times on the training set with 36 unique translations.

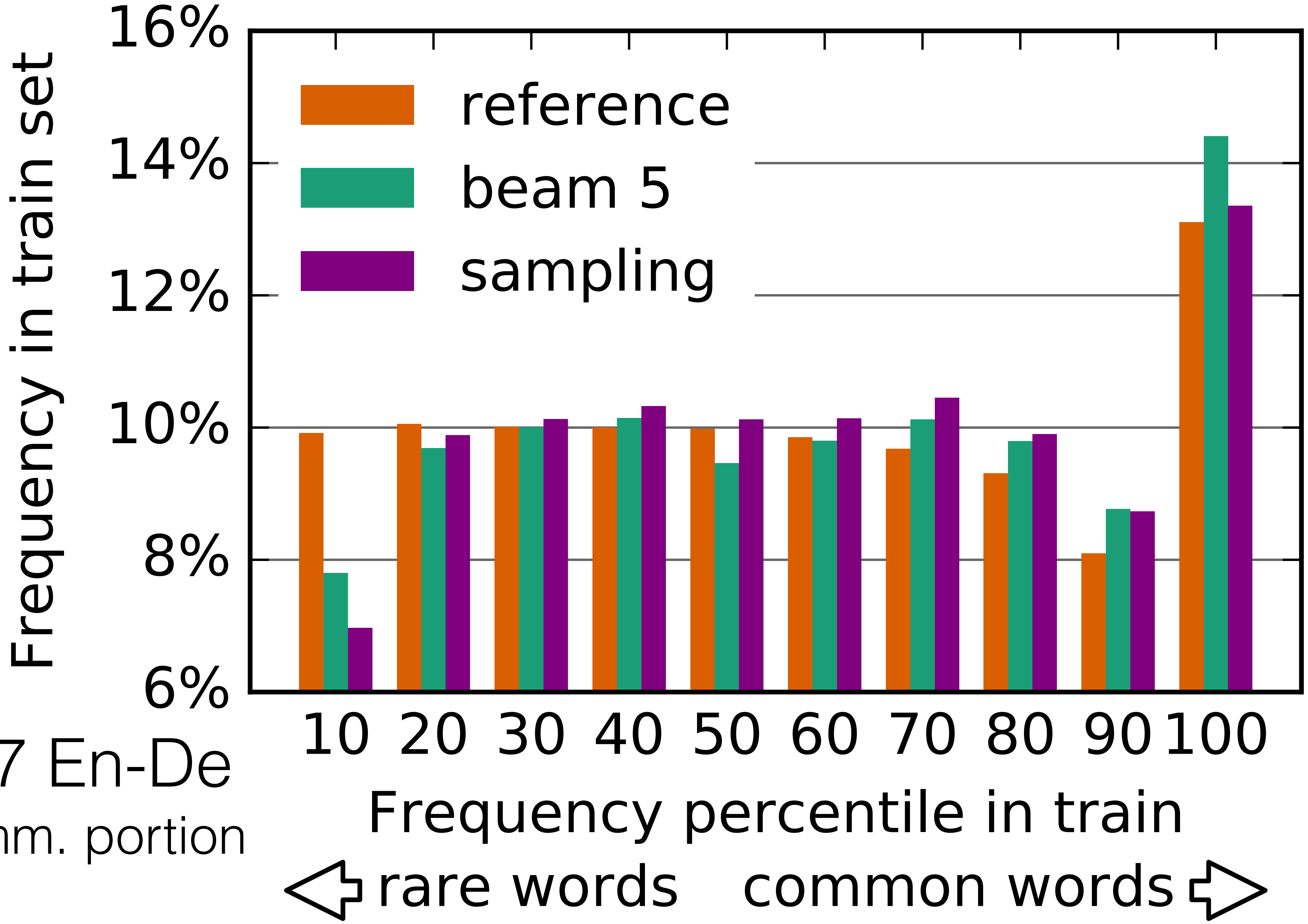


For this source sentence, model and data distribution match very well!

Analysis Tools

- Token level fitting
- Sentence level calibration
- Set level calibration
- Other necessary conditions

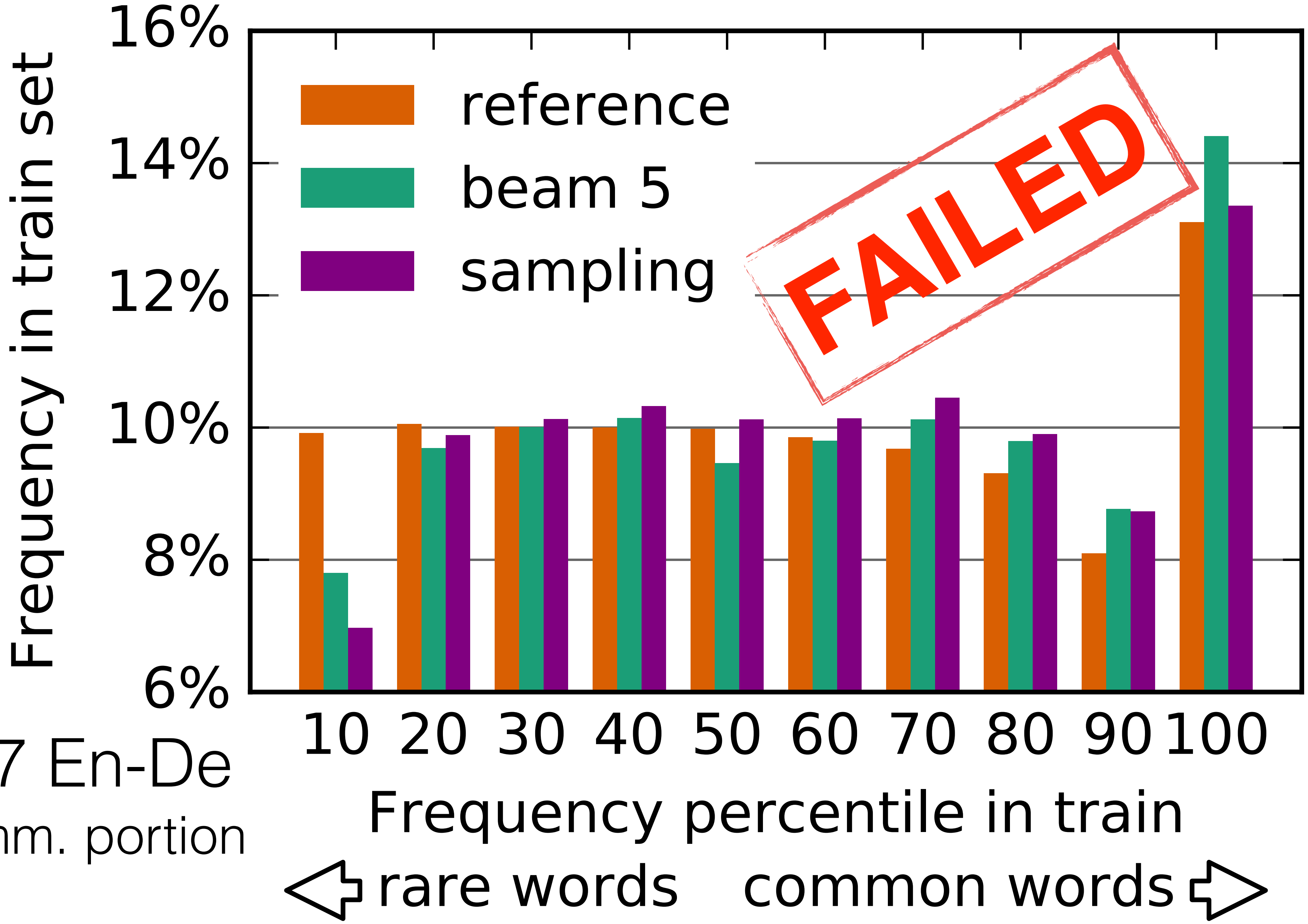
Token Level: Matching Unigram Stats



WMT'17 En-De
news-comm. portion

Model grossly under-estimate rare words.
Beam over-estimates frequent words, as expected.

Token Level: Matching Unigram Stats

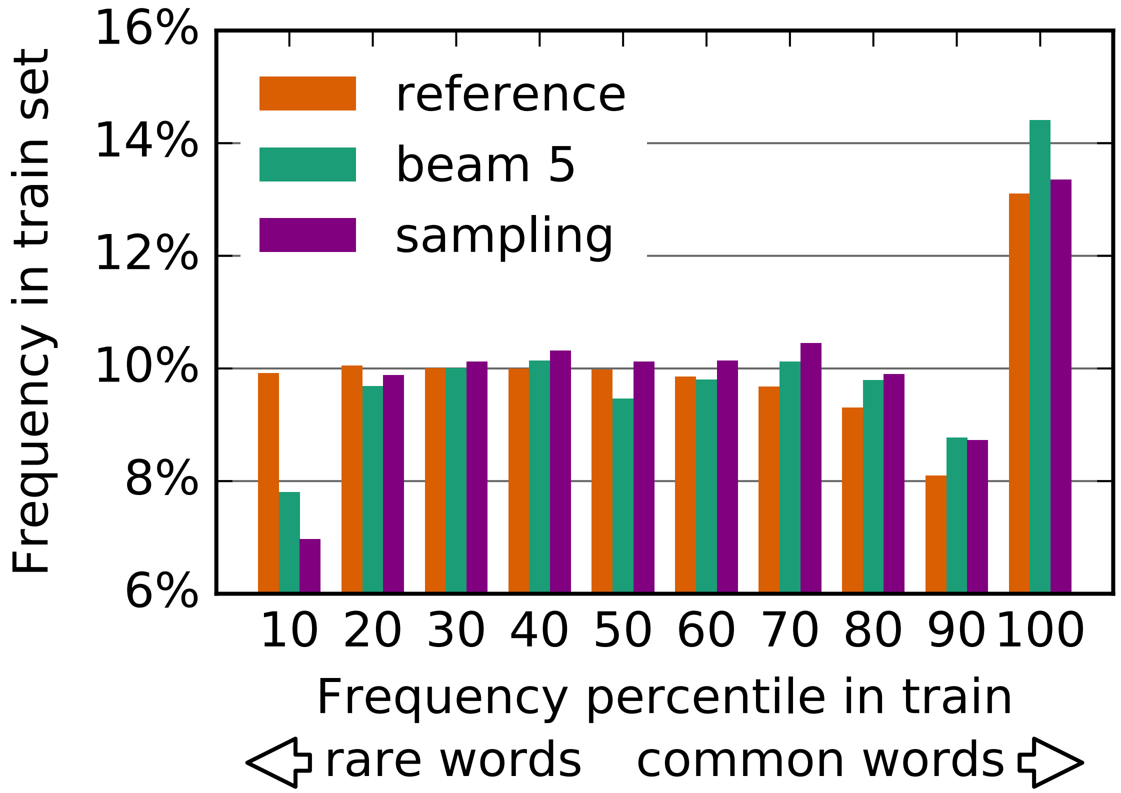


WMT’17 En-De
news-comm. portion

Model grossly under-estimate rare words.
Beam over-estimates frequent words, as expected.

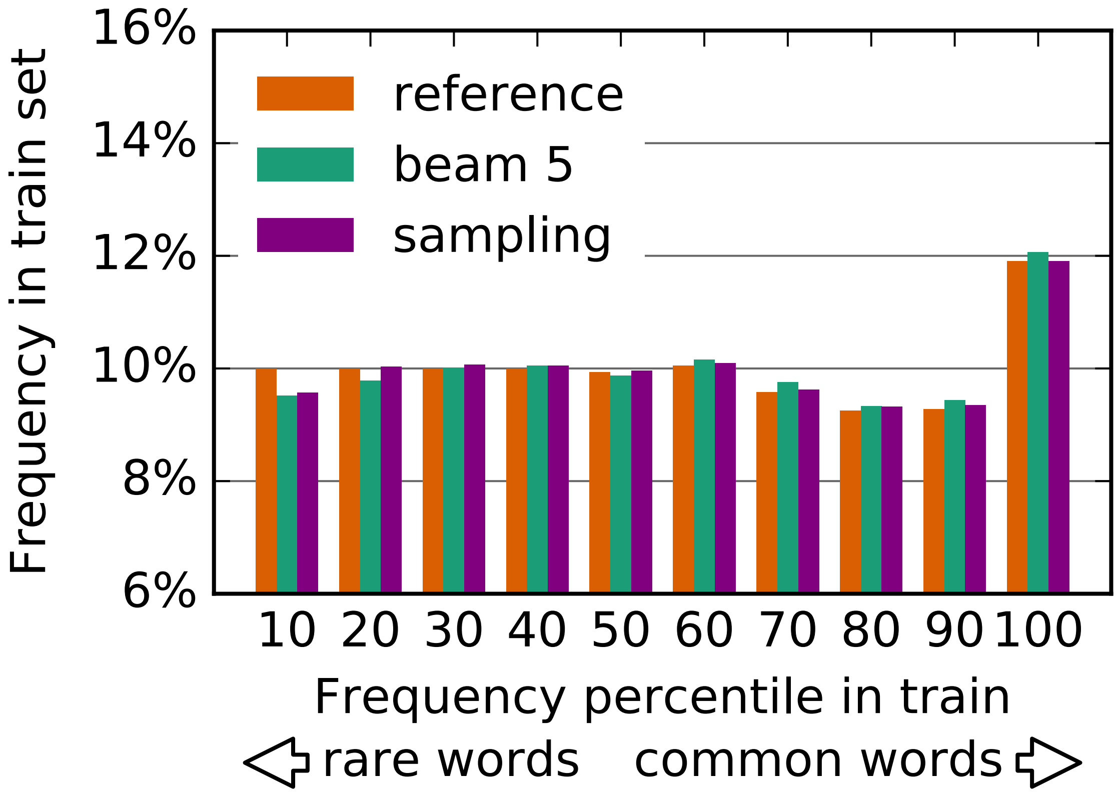
Token Level: Matching Unigram Stats

WMT'17 En-De news-comm. portion



- ~300K parallel sentences
- 21 BLEU on test
- median freq. in 10% bin: 12

WMT'14 En-Fr

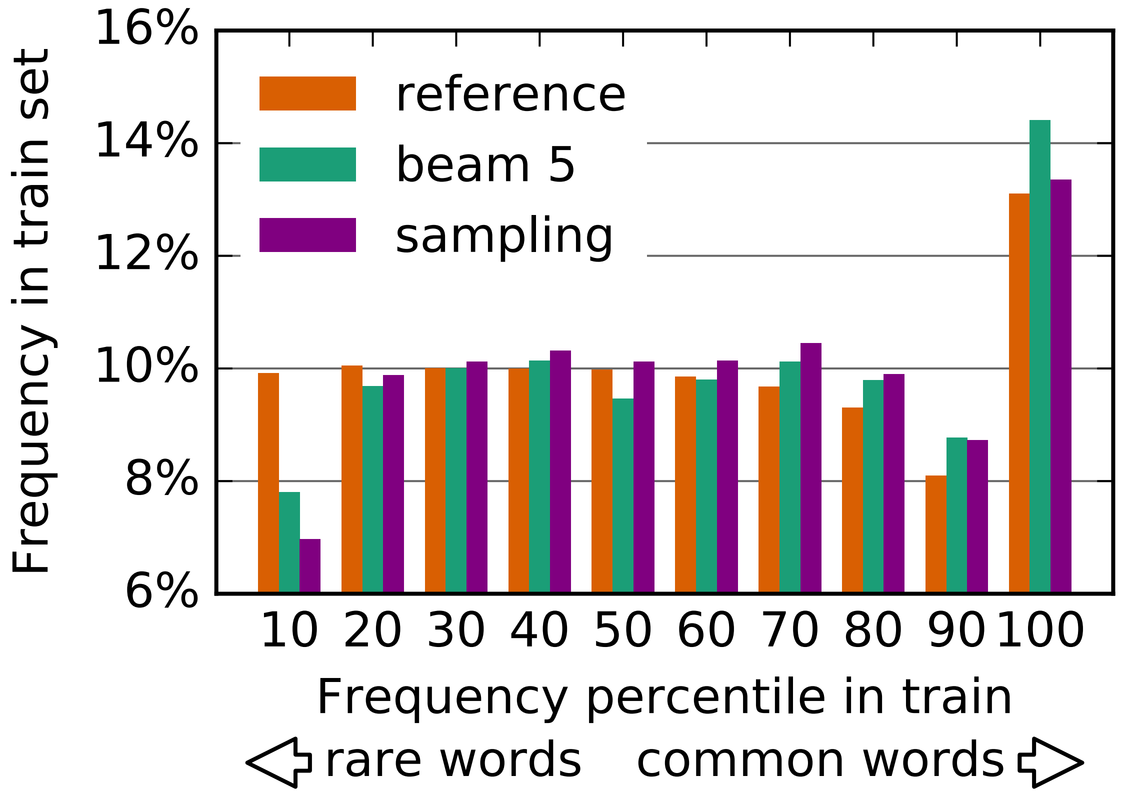


- ~35M parallel sentences
- 41 BLEU on test
- median freq. in 10% bin: 2500

More data & better model close the gap, but rare words are still under-estimated.

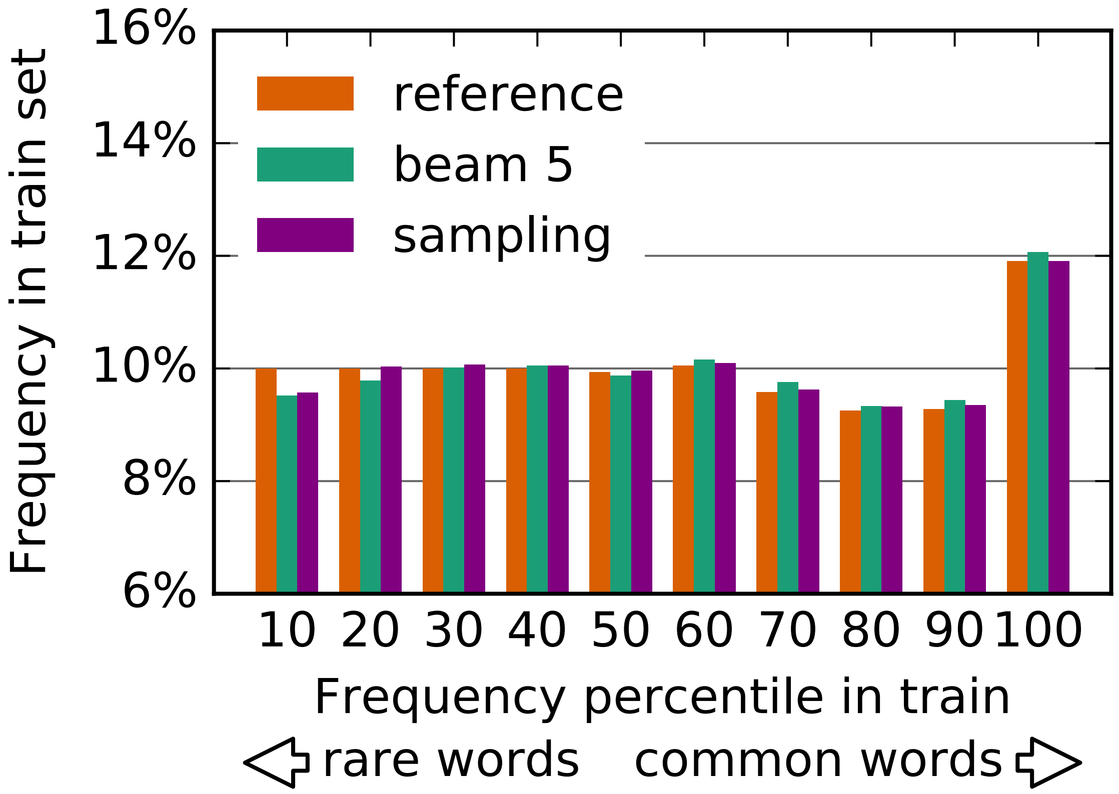
Token Level: Matching Unigram Stats

WMT'17 En-De news-comm. portion



- ~300K parallel sentences
- 21 BLEU on test
- median freq. in 10% bin: 12

WMT'14 En-Fr

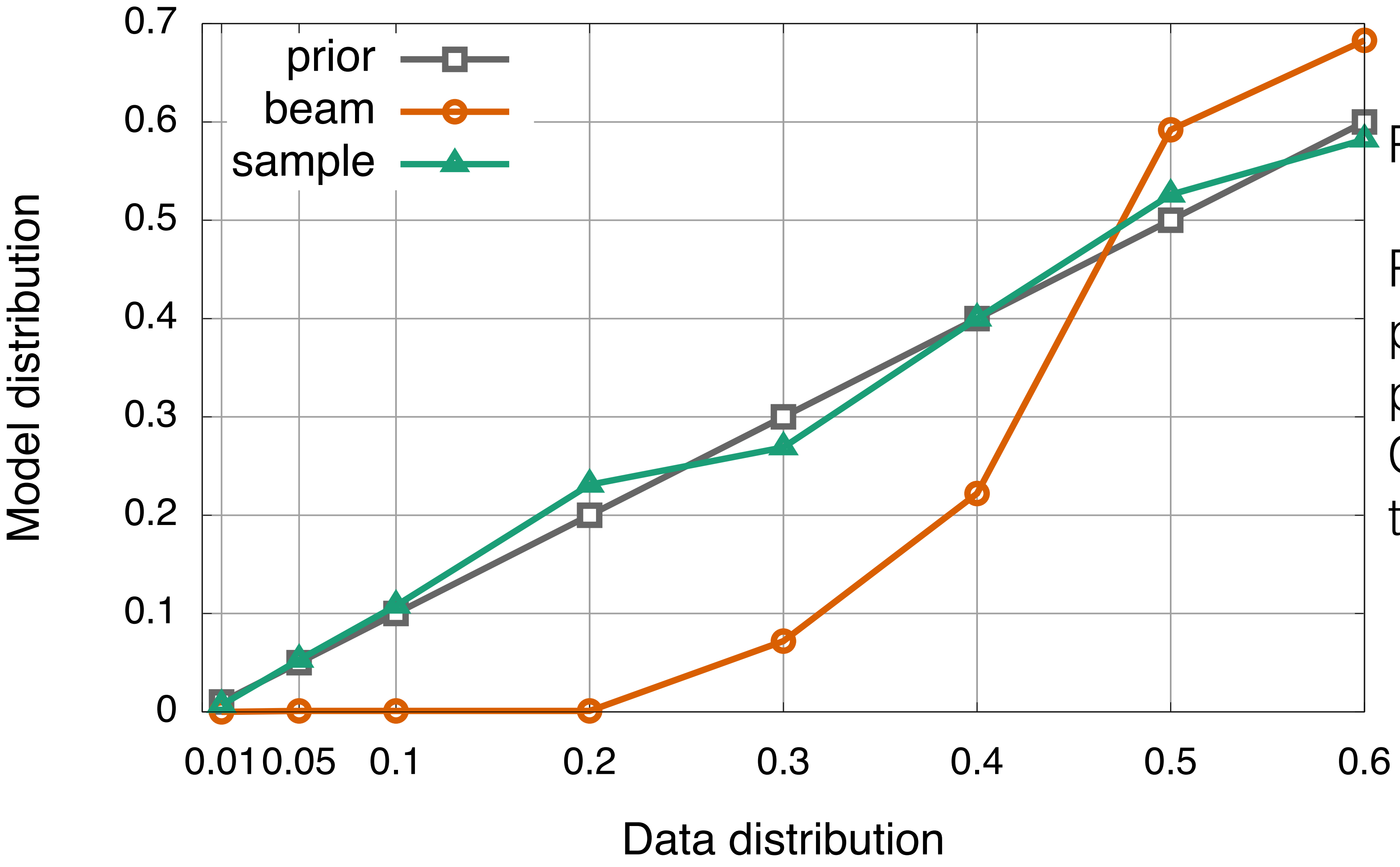


- ~35M parallel sentences
- 41 BLEU on test
- median freq. in 10% bin: 2500

Match may look better than it is if model shifts probability mass within each of these buckets, let's take a closer look then…

M. Ranzato

Token Level Fitting #2

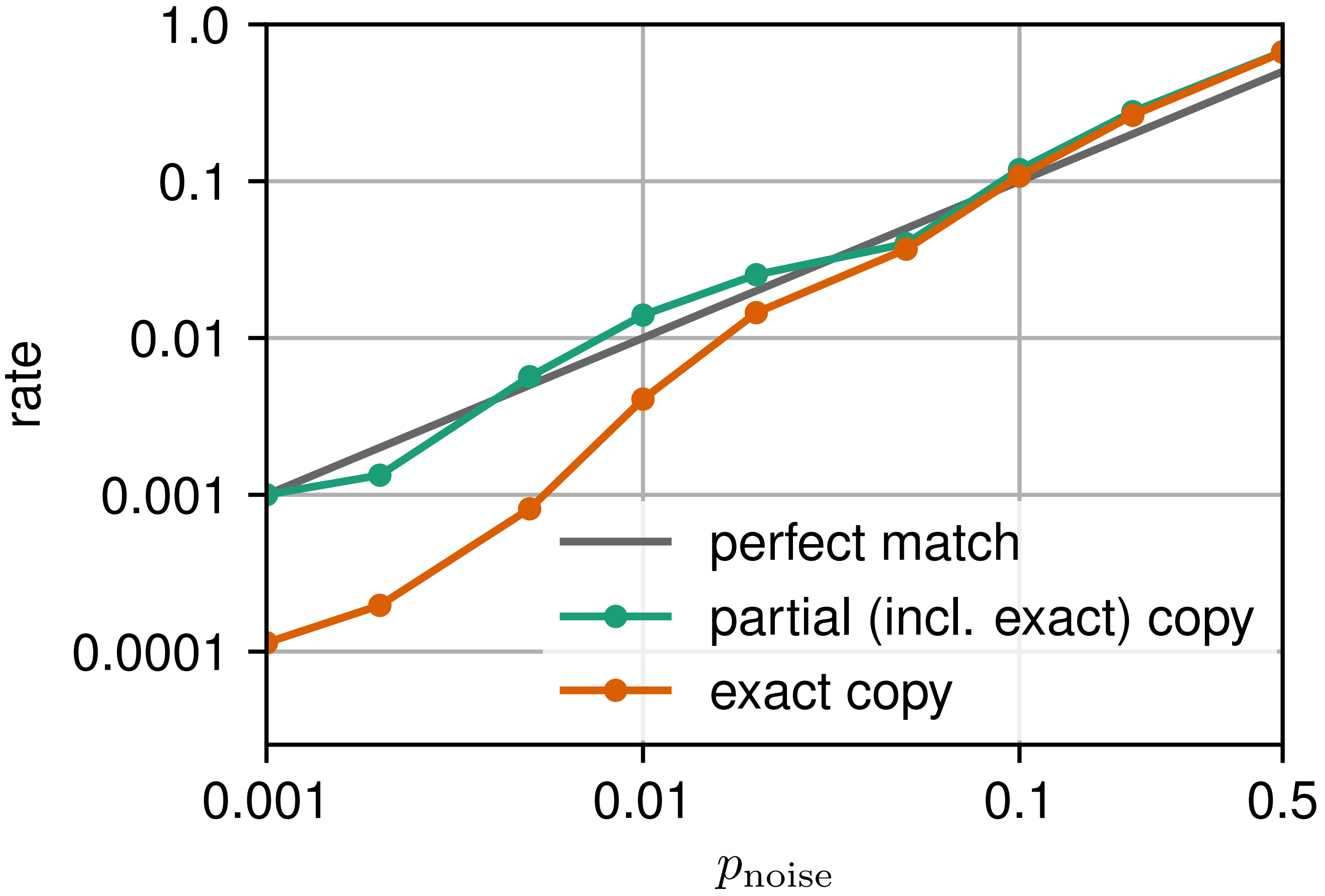


Pick mid-frequency words.

Replace word type w by w1 with probability p, and by w2 with probability (1-p).
Check whether model generates tokens with the correct ratio.

Model fits fairly well at the token level for mid-frequency words. Beam under/over-estimates.

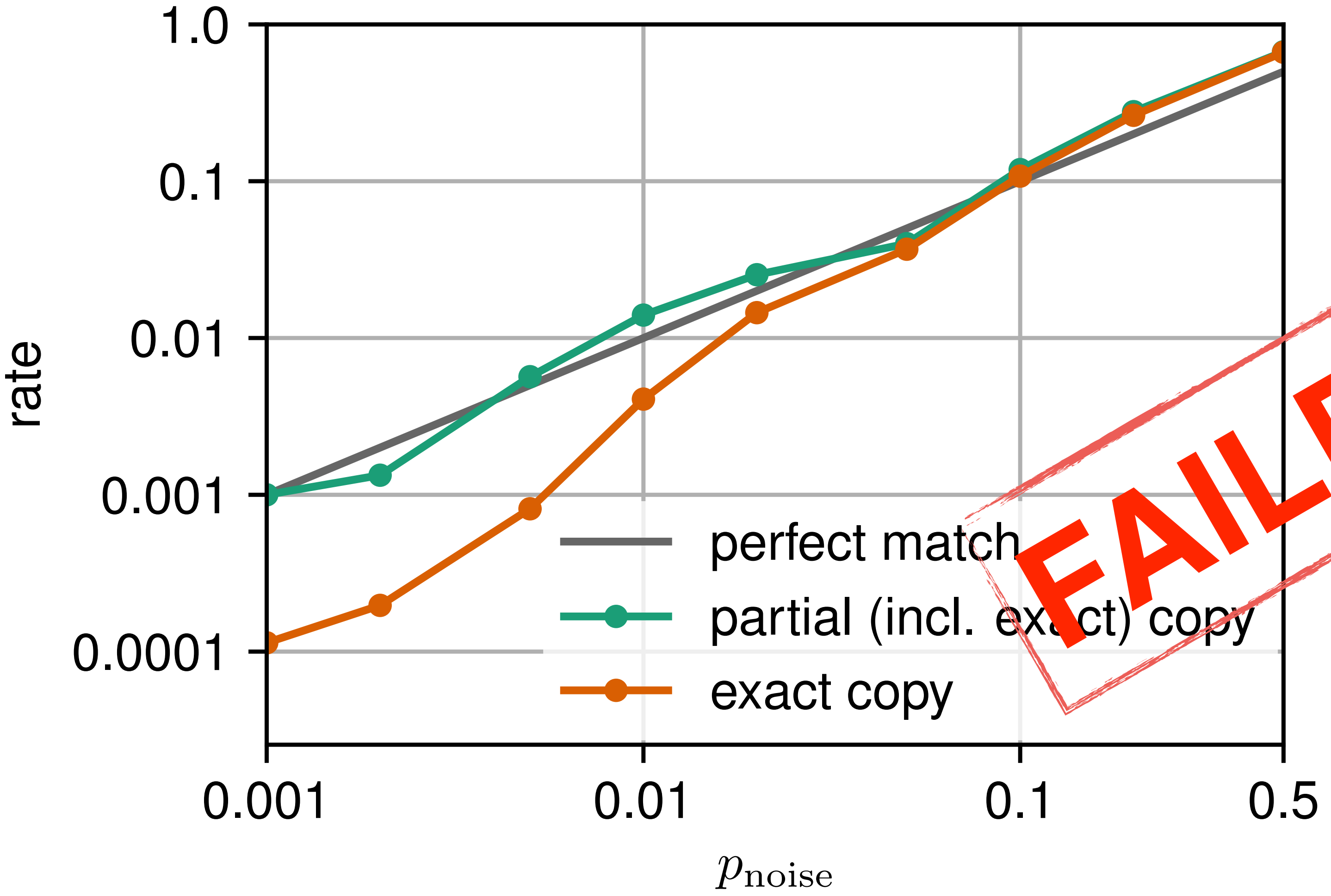
Sentence Level Calibration



Copy source sentences at a given rate during training, check whether probability assigned by the model to copies matches the copy production rate.

NMT model under-estimates copy probability at low rates, while it over-estimates it at high rates.
Model spills probability mass on partial copies.

Sentence Level Calibration

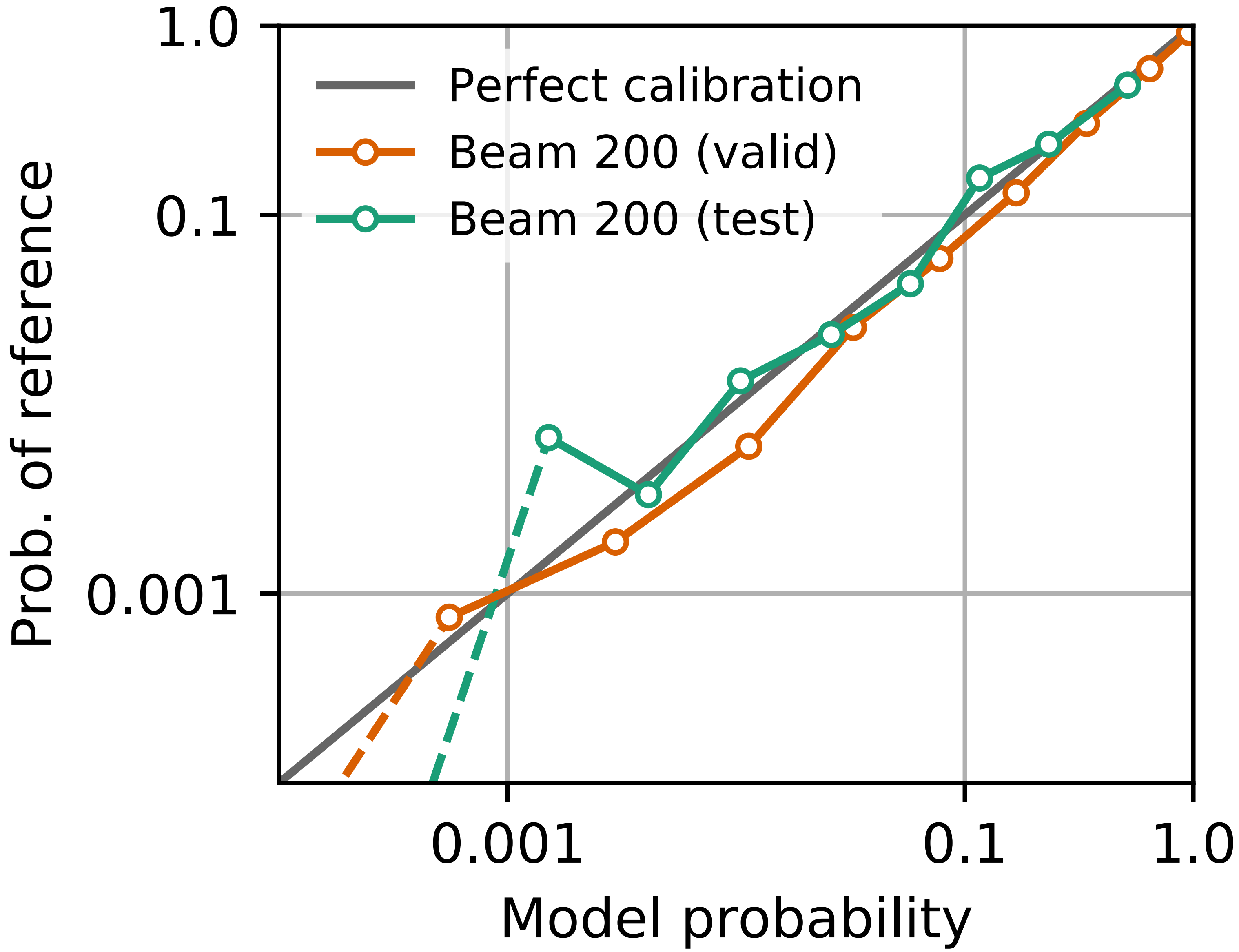


Copy source sentences at a given rate during training, check whether probability assigned by the model to copies matches the copy production rate.

FAILED

NMT model under-estimates copy probability at low rates, while it over-estimates it at high rates.
Model spills probability mass on partial copies.

Set Level Calibration



$$\mathop{\mathbb{E}}_{x \sim p_d} [\mathbb{I}\{x \in S\}] = p_m(S)$$

where S is the set of hypotheses produced by beam.

NMT model is very well calibrated at the set level.

Distance Matching

$$\mathop{\mathbb{E}}_{y \sim p_d, y' \sim p_d}[BLEU(y, y')] \overset{?}{=} \mathop{\mathbb{E}}_{y \sim p_m, y' \sim p_m}[BLEU(y, y')]$$

	En-Fr	**En-De**
human	44.5	32.1
NMT	28.6	24.2

NMT model produces samples that have low BLEU and that are too diverse. Model spreads probability mass.

Distance Matching

$$\mathbb{E}_{y\sim p_d, y'\sim p_d}[BLEU(y,y')] \overset{?}{=} \mathbb{E}_{y\sim p_m, y'\sim p_m}[BLEU(y,y')]$$

	En-Fr	**En-De**
human	44.5	32.1
NMT	28.6	24.2

FAILED

NMT model produces samples that have low BLEU and that are too diverse. Model spreads probability mass.

Multi-Reference Experiments

We collected 10 additional references for 500 randomly selected source sentences from the test set.

We then measure:

- BLEU with *oracle reference,* which is the reference yielding the largest BLEU score.
- BLEU of *average oracle*: compute the above for every hypothesis produced by beam/sampling, and then average.
- *coverage*: number of unique references hypotheses are matched to.

Multi-Reference Experiments

	Beam@5	Beam@200	200 Samples
single reference	41.4	36.2	38.2
oracle reference	70.2	61.0	64.1
average oracle	65.7	56.4	39.1
coverage	1.9	5.0	7.4

Multi-Reference Experiments

	Beam@5	**Beam@200**	**200 Samples**
single reference	41.4	36.2	38.2
oracle reference	70.2	61.0	64.1
average oracle	65.7	56.4	39.1
coverage	1.9	5.0	7.4

Beam produces outputs close to an actual reference.
Lower scoring hypotheses are not far from a reference.
However, they often map to the same reference.

Multi-Reference Experiments

	Beam@5	**Beam@200**	**200 Samples**
single reference	41.4	36.2	38.2
oracle reference	70.2	61.0	64.1
average oracle	65.7	56.4	39.1
coverage	1.9	5.0	7.4

Sampling is more diverse but several samples poorly match any given reference. Mass is spread too much.

Multi-Reference Experiments

	Beam@5	Beam@200	200 Samples
single reference	41.4	36.2	38.2
oracle reference	70.2	61.0	64.1
average oracle	65.7	56.4	39.1
coverage	1.9	5.0	7.4

Homework: if two continuous and uniform p.d.f. matched, how many samples would we need to draw in order to get full coverage with high probability?

Model/Data Distribution Match

- they do not match in general, although anecdotally they might.
- model spreads probability mass too much (see results using sampling and pair-wise BLEU, for instance):
    - it’s impossible for NMT to assign 0 probability to any sequence; low coverage of probability mass.
    - spill-over to “nearby” hypotheses.
- [conjecture] although model may under-estimate copies at low rates, these may be on the top of the beam, just because probability mass is too spread.
- copy noise is over-estimated at high rates.
- model’s most likely outputs (or their proxy) are usually very concentrated (little diversity), possibly also due to probability spread over similar hypotheses.

Conclusions

- Uncertainty in data: intrinsic/extrinsic
  - Search: works really well. For large beams, beam finds spurious modes, but we know how to fix it!
    [not so surprising, since we did model selection using beam search!]
- Model & Data distribution: model is surprisingly well calibrated. In general, it spreads probability mass too much compared to the data distribution.
- More parallel data helps a lot…

Actionable Items

- There are easy fixes to the copy problem [done]
- It would be interesting to find ways to manipulate the model to avoid the spread of probability mass while diversifying beam. [ongoing]

Questions?
Вопросы?
¿Preguntas?

Lecture Outline

- Exposure bias/Loss Mismatch: Training at the Sequence Level.
  - how do classical structured prediction losses fare against recent proposals?
  - how much to be gained by fixing this inconsistency?
- Analyzing Uncertainty: model fitting and effects on search.
  - why do larger beam perform worse?
  - why is the model under-estimating rare words?
- Training Without Supervision.
  - how to leverage monolingual data?
  - can we learn without any parallel sentence?

Lecture Outline

Word Translation Without Parallel Data
Alexis Conneau*, Guillaume Lample*, Marc'Aurelio Ranzato, Ludovic Denoyer, Herve Jegou
ICLR 2018
https://arxiv.org/abs/1710.04087 CODE: https://github.com/facebookresearch/MUSE

Unsupervised Machine Translation Using Monolingual Corpora Only
Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato
ICLR 2018
https://arxiv.org/abs/1711.00043

- Training Without Supervision.
  - how to leverage monolingual data?
  - can we learn without any parallel sentence?

Lecture Outline

Word Translation Without Parallel Data
Alexis Conneau*, Guillaume Lample*, Marc'Aurelio Ranzato, Ludovic Denoyer, Herve Jegou
ICLR 2018
https://arxiv.org/abs/1710.04087 CODE: https://github.com/facebookresearch/MUSE

Unsupervised Machine Translation Using Monolingual Corpora Only
Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato
ICLR 2018
https://arxiv.org/abs/1711.00043

credit: several slides borrowed from Guillaume.

Motivation

- NMT models work very well, provided a lot of parallel data.
- For many language pairs, parallel data is however very scarce, or even inexistent.
- Professional translators are very expensive and hard to find for some language pairs.
- We need a scalable approach to be able to translate in any language pair.

Motivation

- Resources we could use:
  - Limited amount of parallel data.
  - Parallel data from other language pairs.
  - Large amount of monolingual data, which is often more easily available.

Goal

- Training an NMT system without supervision, using monolingual data only.
  - Admittedly, unrealistic but…
  - Baseline for extensions using parallel data (from language pair of interest or others).
  - Scientific endeavor, towards our quest for a good unsupervised learning algorithm.

Unsupervised Word Translation

- Motivation: A pre-requisite for unsupervised sentence translation.

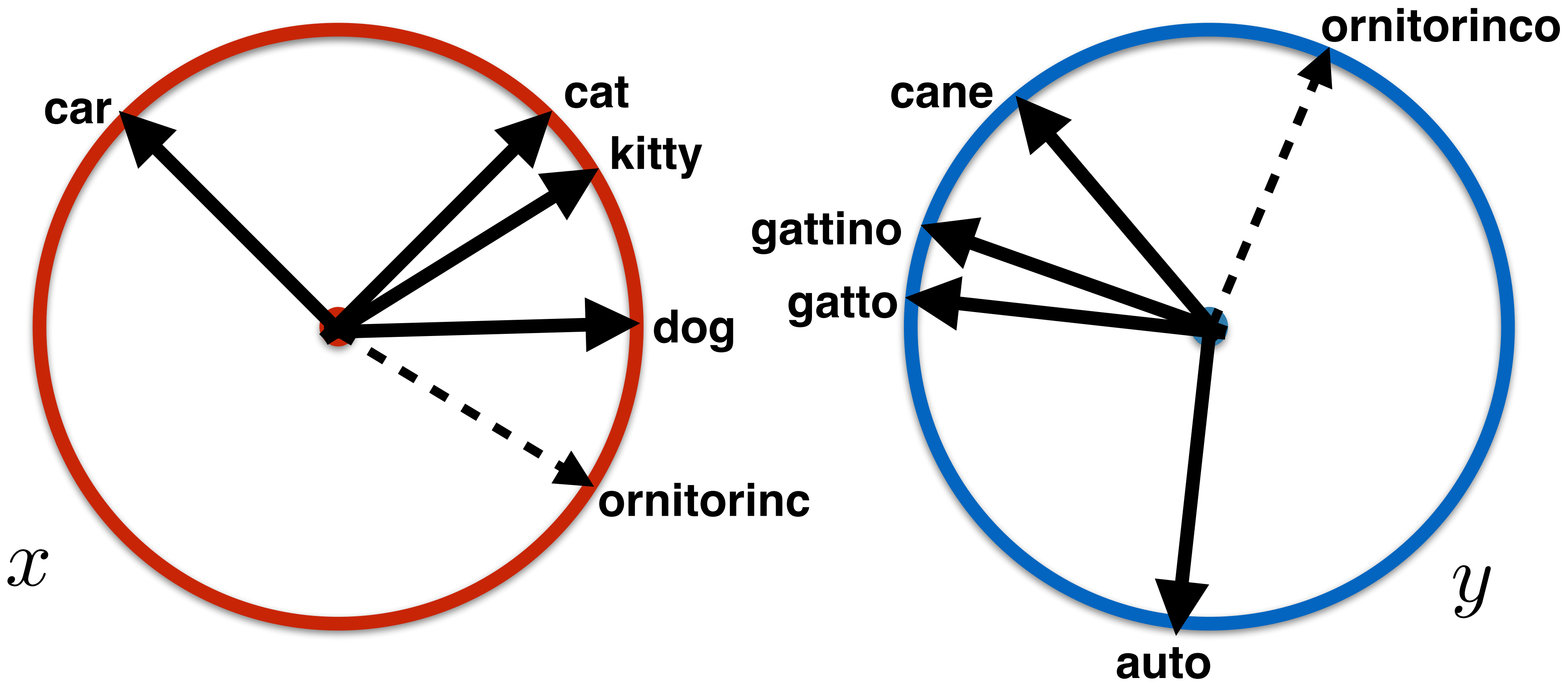
- Problem: given two monolingual corpora in two different languages, estimate bilingual lexicon.

- Hint: the context of a word, is often similar across languages since each language refers to the same underlying physical world.

Method

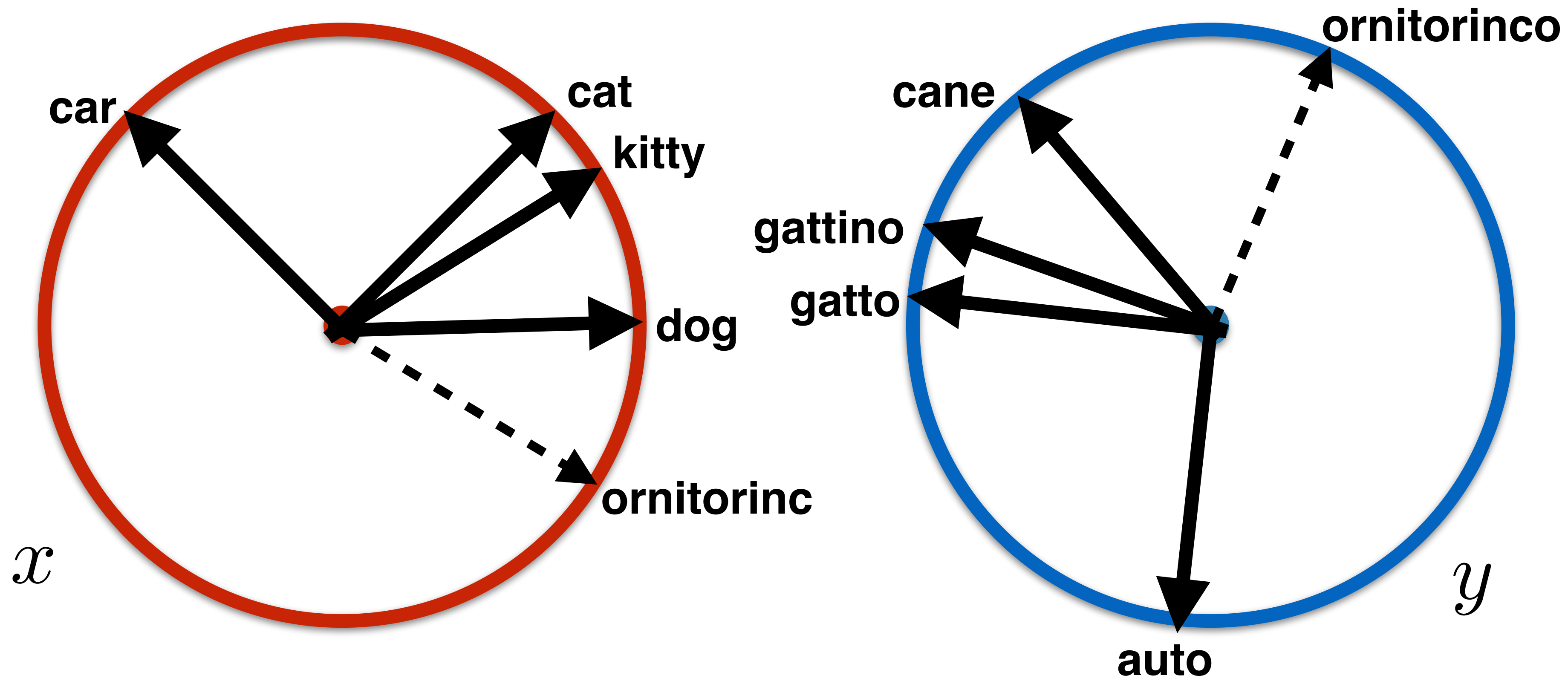
1) learn word embeddings (word2vec) separately on each language using lots of monolingual data.

Method



1) learn word embeddings (word2vec) separately on each language using lots of monolingual data.

Method



2) learn a rotation matrix to roughly align the two domains.

E.g., via adversarial training: pick a word at random from each language, embed them, project one of the two, and make sure distributions match.

x_i embedding i-th word in En

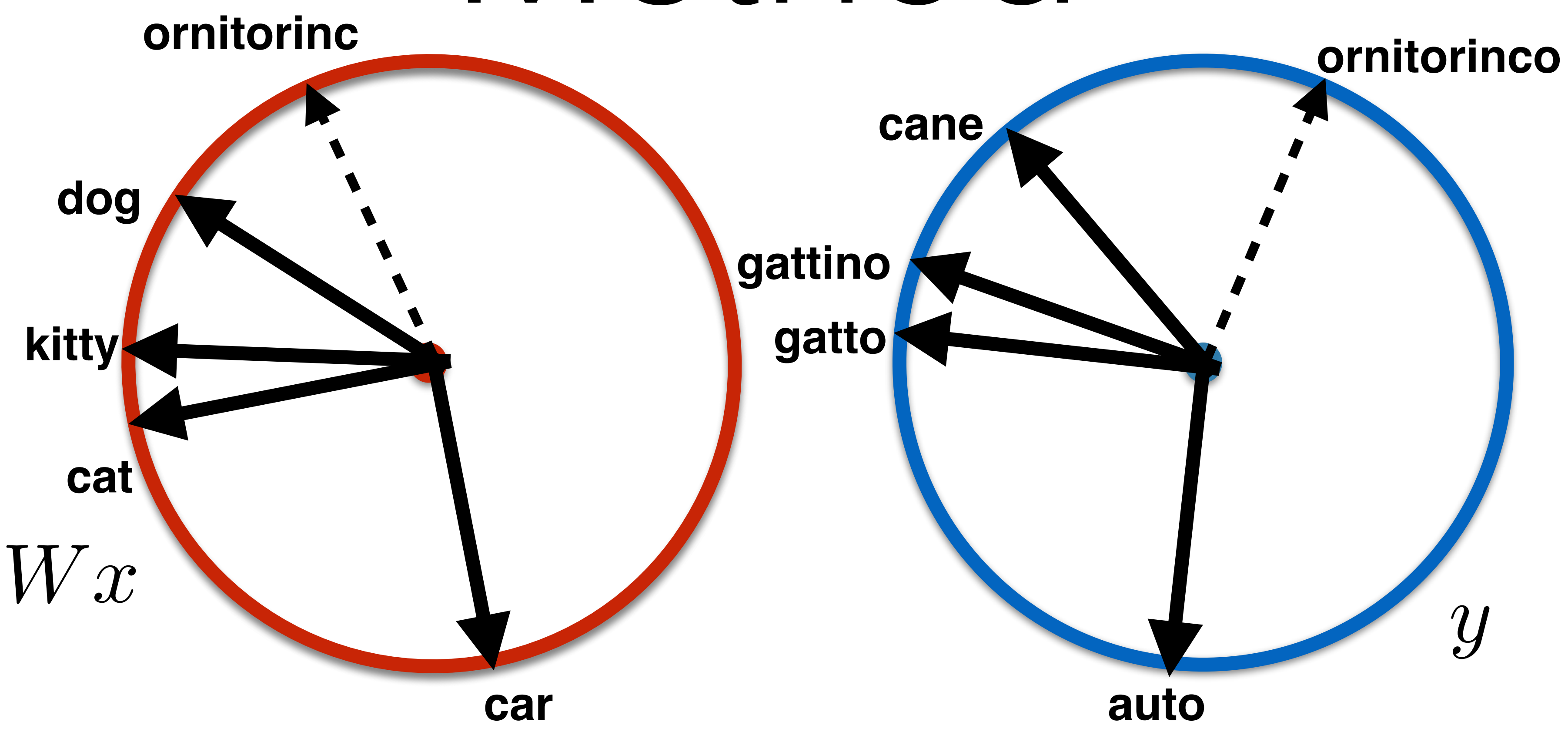
y_j embedding j-th word in It

W orthogonal matrix

$$\mathcal{L}_D(\theta_D|W) = -\mathbb{E}_x\left[\log p(\mathrm{En}|Wx;\theta_D)\right] - \mathbb{E}_y\left[\log p(\mathrm{It}|y;\theta_D)\right]$$

$$\mathcal{L}_W(W\theta_D) = -\mathbb{E}_x\left[\log p(\mathrm{It}|Wx;\theta_D)\right] - \mathbb{E}_y\left[\log p(\mathrm{En}|y;\theta_D)\right]$$

Method



2) learn a rotation matrix to roughly align the two domains.
E.g., via adversarial training: pick a word at random from each language, embed them, project one of the two, and make sure distributions match.

x_i embedding i-th word in En

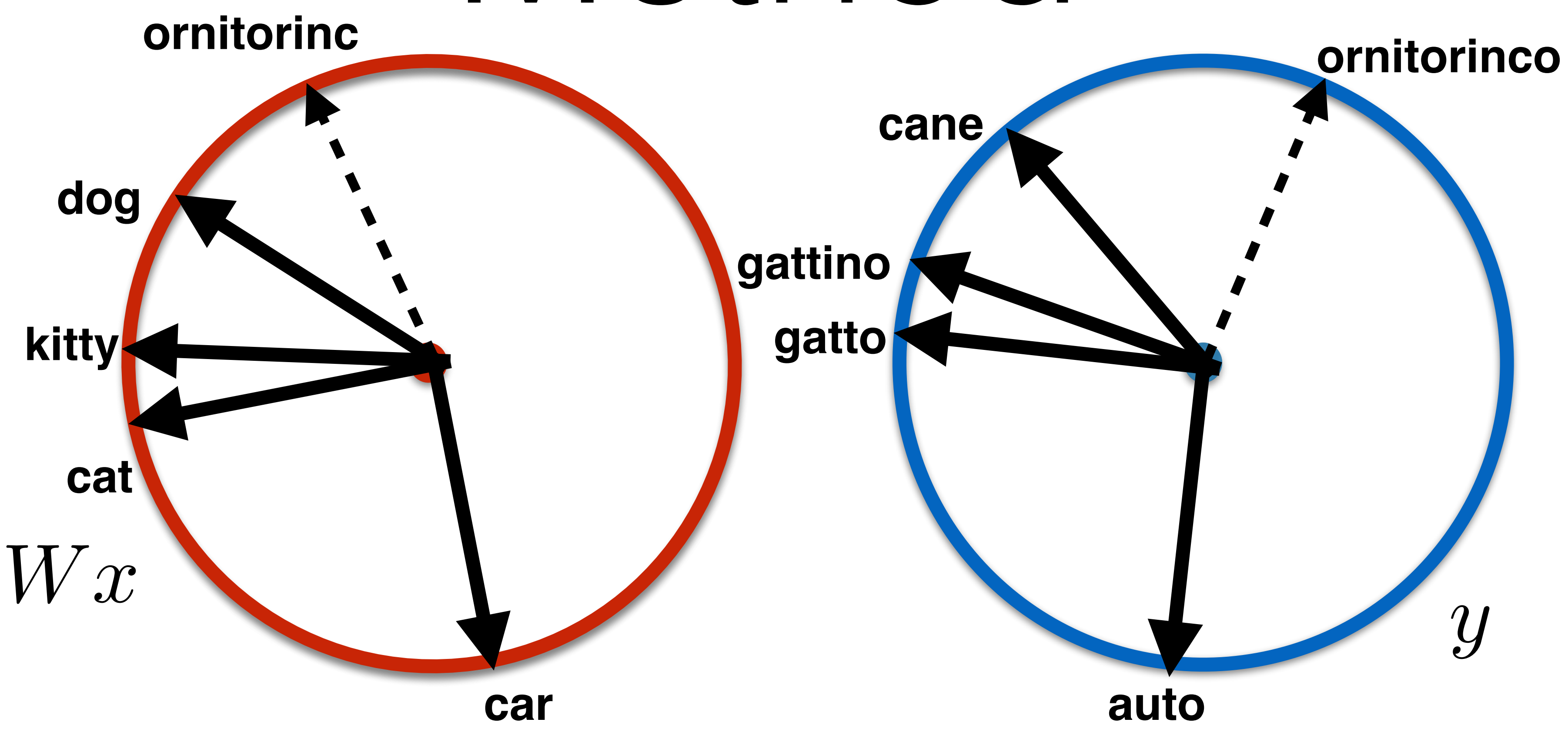
y_j embedding j-th word in It

W orthogonal matrix

$$\mathcal{L}_D(\theta_D|W) = -\mathbb{E}_x\left[\log p(\mathrm{En}|Wx;\theta_D)\right] - \mathbb{E}_y\left[\log p(\mathrm{It}|y;\theta_D)\right]$$

$$\mathcal{L}_W(W\theta_D) = -\mathbb{E}_x\left[\log p(\mathrm{It}|Wx;\theta_D)\right] - \mathbb{E}_y\left[\log p(\mathrm{En}|y;\theta_D)\right]$$

Method



3) Iterative refinement via orthogonal Procrustes, using the most frequent words.

Pick most frequent words, translate them via nearest neighbor, solve least square, and iterate.

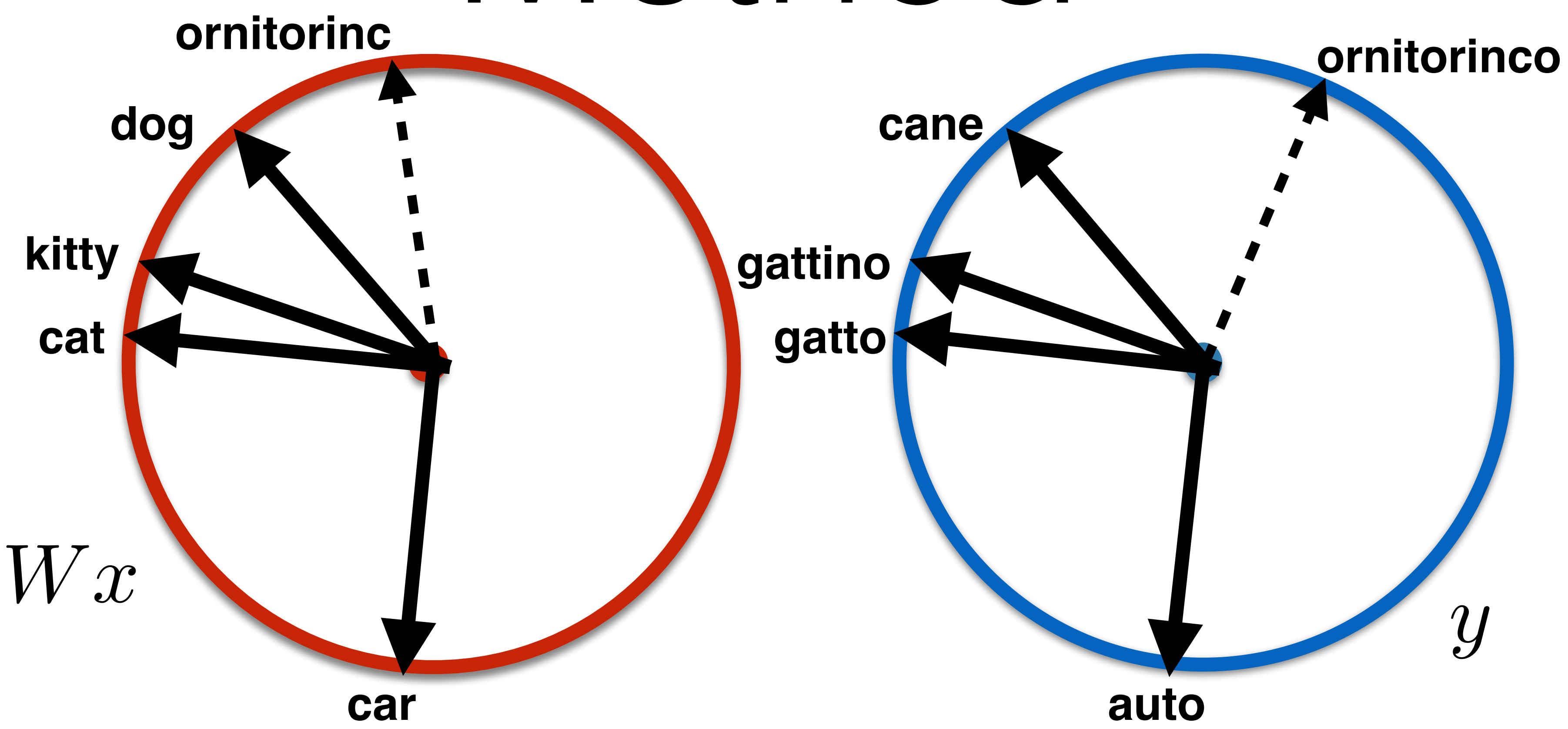
x_i embedding i-th word in En

y_j embedding j-th word in It

W orthogonal matrix

$$W_t = \arg\min ||W_{t-1}X - Y||^2, \text{s.t. } W_t W_t^T = I$$

Method



3) Iterative refinement via orthogonal Procrustes, using the most frequent words.

Pick most frequent words, translate them via nearest neighbor, solve least square, and iterate.

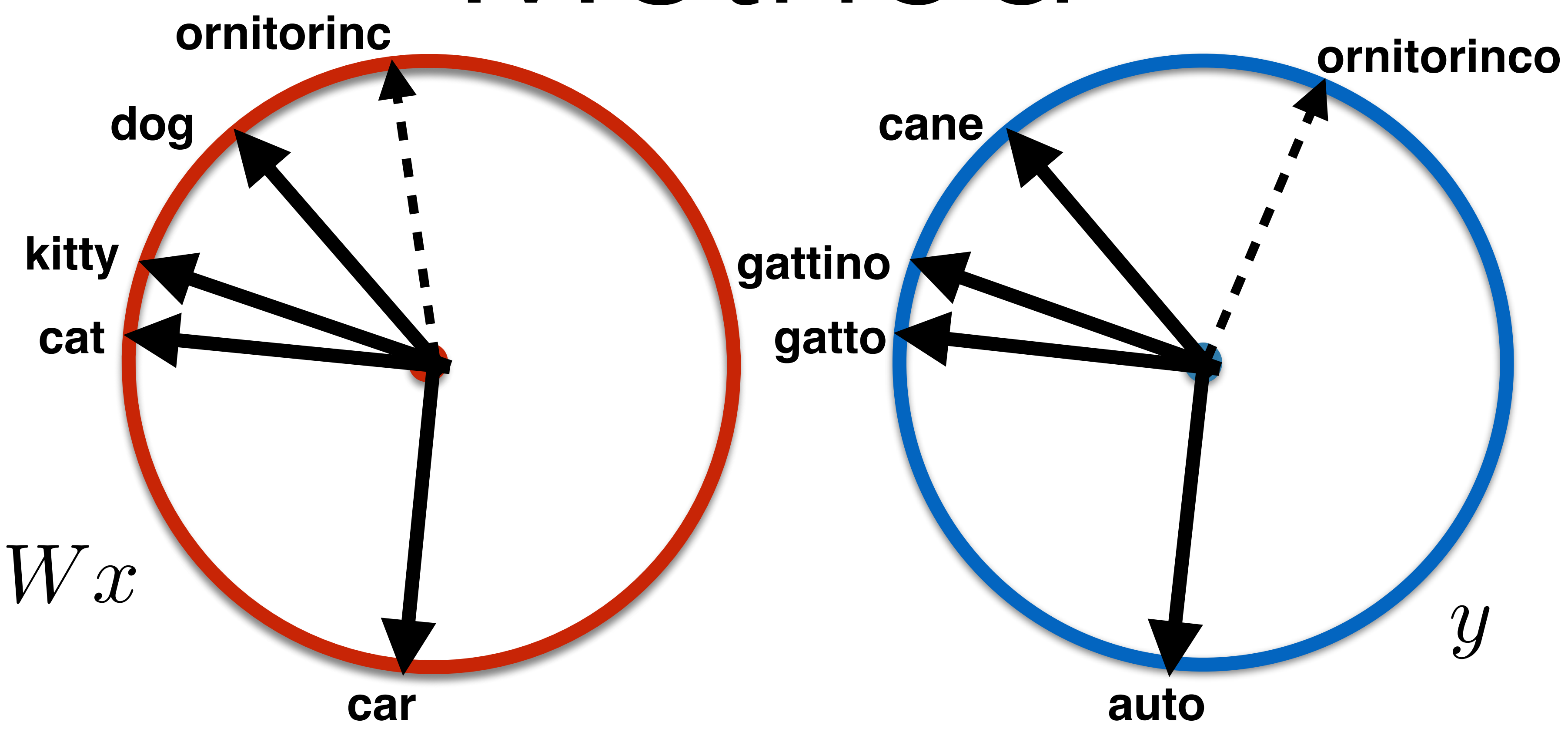
x_i embedding i-th word in En

y_j embedding j-th word in It

W orthogonal matrix

$$W_t = \arg\min ||W_{t-1}X - Y||^2, \text{s.t. } W_t W_t^T = I$$

Method



4) Build lexicon using metric that compensates for hubness.

There are words that have lots of neighbors, while others that are not neighbors of anybody.

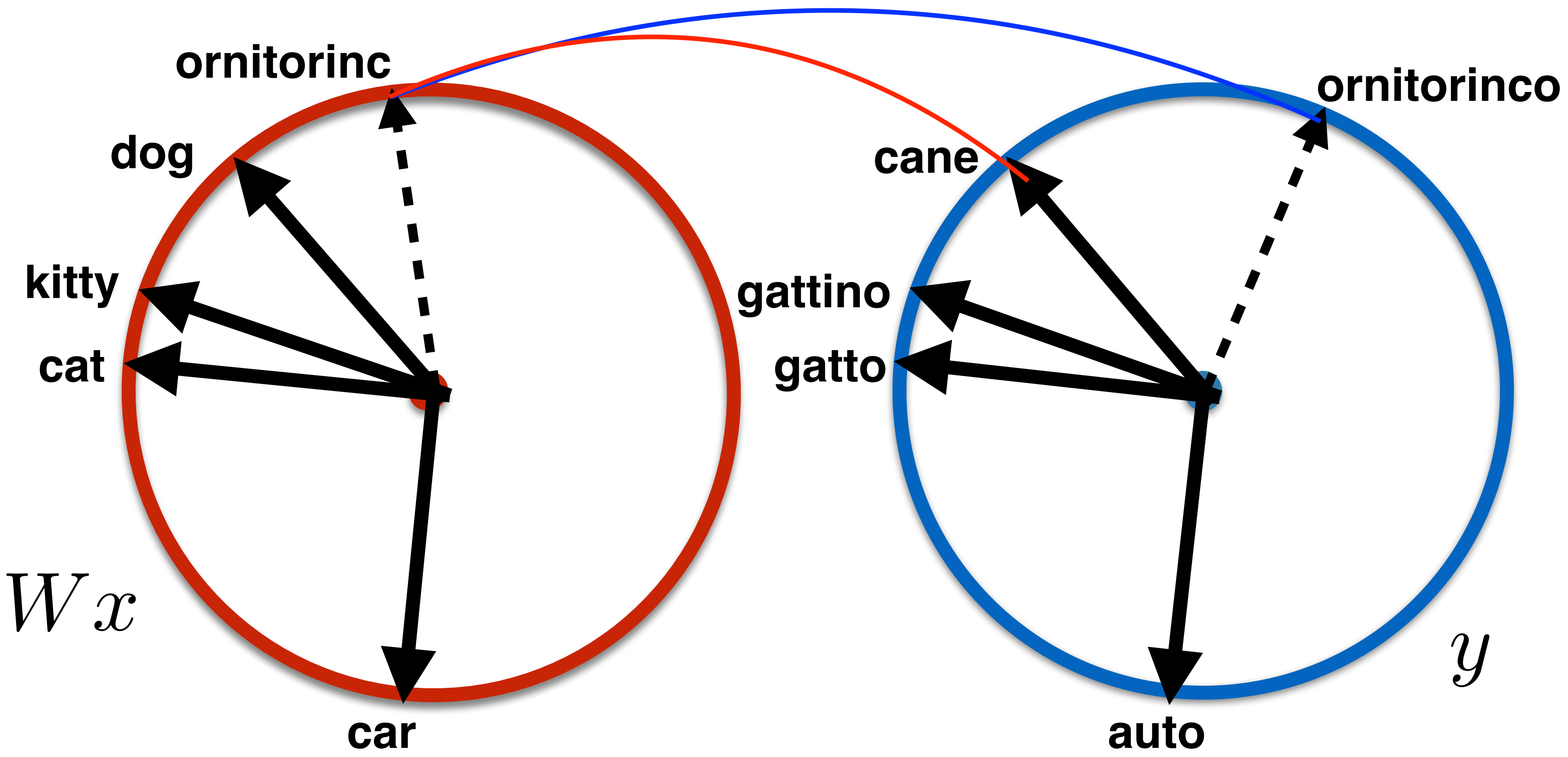
x_i embedding i-th word in En

y_j embedding j-th word in It

W orthogonal matrix

$$\mathrm{CSLS}(Wx, y) = 2\cos(Wx, y) - r_{\mathrm{En}}(Wx) - r_{\mathrm{It}}(y)$$

$$r_{\mathrm{En}}(Wx) = \frac{1}{K} \sum_{y_t \in \mathcal{N}_{\mathrm{En}}(Wx)} \cos(Wx, y_t)$$



4) Build lexicon using metric that compensates for hubness.

There are words that have lots of neighbors, while others that are not neighbors of anybody.

x_i embedding i-th word in En

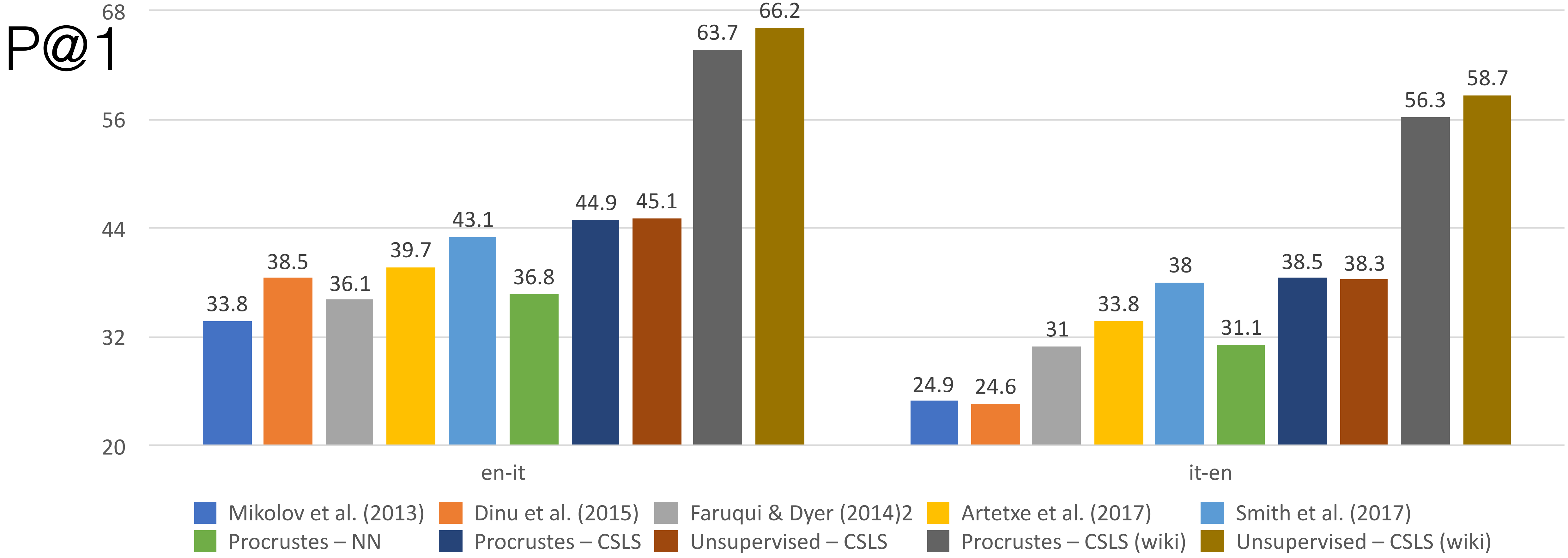
y_j embedding j-th word in It

W orthogonal matrix

$$\text{CSLS}(Wx, y) = 2\cos(Wx, y) - r_{\text{En}}(Wx) - r_{\text{It}}(y)$$

$$r_{\text{En}}(Wx) = \frac{1}{K} \sum_{y_t \in \mathcal{N}_{\text{En}}(Wx)} \cos(Wx, y_t)$$

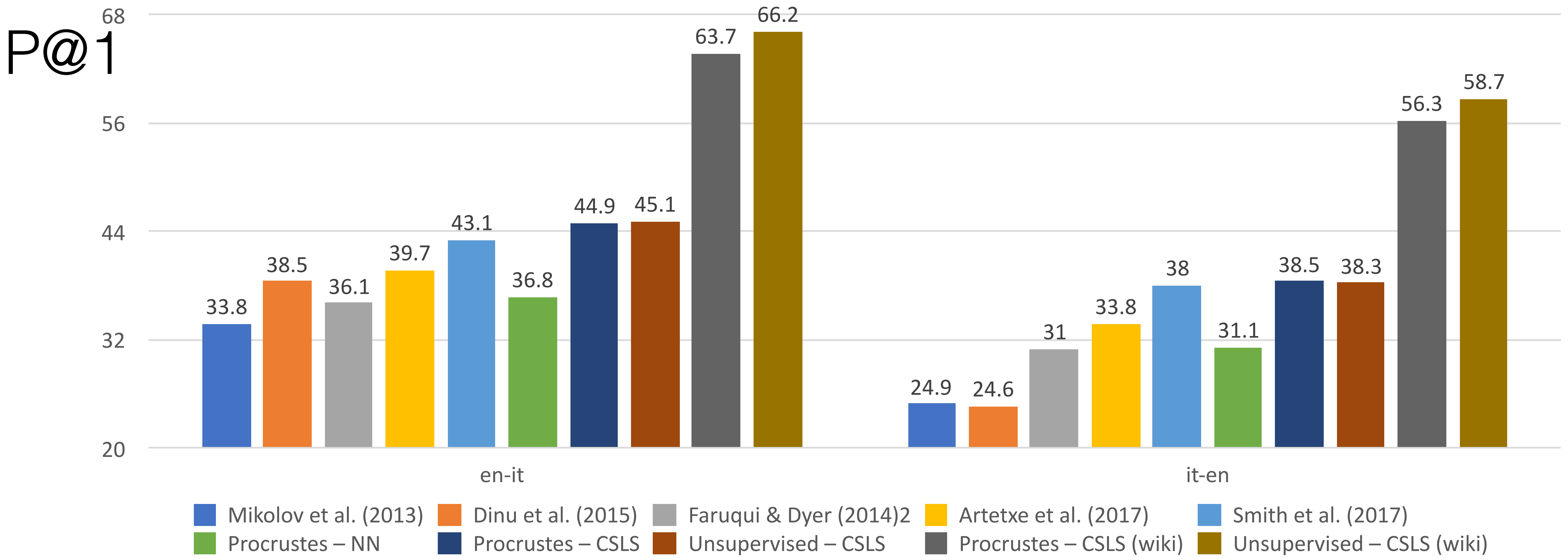
Results on Word Translation



More results on several language pairs, analysis and other tasks in the paper.

By using more anchor points and lots of unlabeled data, we even outperform supervised approaches!

Results on Word Translation



Homework 1: how accurate does the adversarial alignment need to be? Can more refinement steps compensate for poor initial alignment?
Homework 2: apply the same method to sentences from the Multi30K-Task1 image caption dataset.

Key Idea

- Learn representations of each domain.
- Force representations to match in order to translate.
- How to apply this principle to sentences?

Naïve Application

- In general, this may not work on sentences because:
  - Without leveraging compositional structure, space is exponentially large.
  - Need good sentence representations.
  - Unlikely that a linear mapping is sufficient to align sentence representations of two languages.

From Words to Sentences

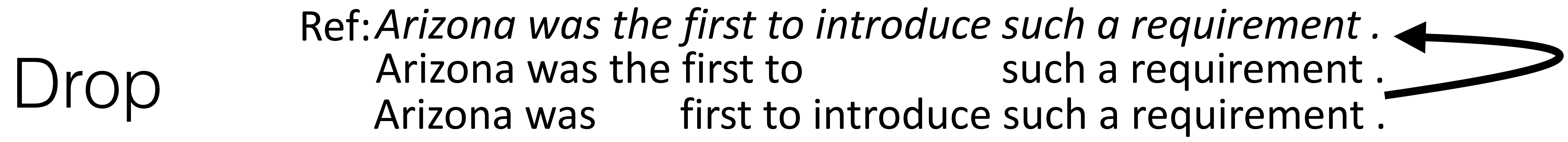
How to learn good sentence representations?
We want to train usual sep2seq architecture (as that achieves the best MT results), but without supervision.

From Words to Sentences

How to learn good sentence representations?
We want to train usual sep2seq architecture (as that achieves the best MT results), but without supervision.

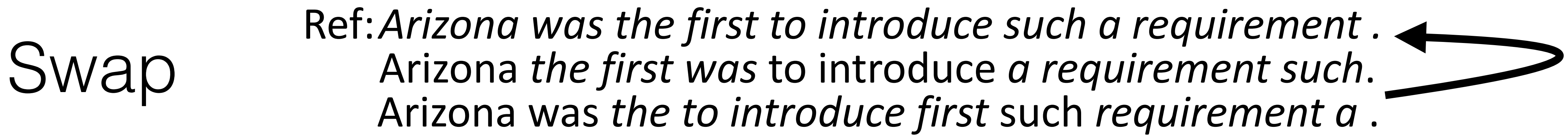
Solution: denoising autoencoding task.

Noise: word drop and word swap.



Drop

Ref: *Arizona was the first to introduce such a requirement .*
Arizona was the first to such a requirement .
Arizona was first to introduce such a requirement .



Swap

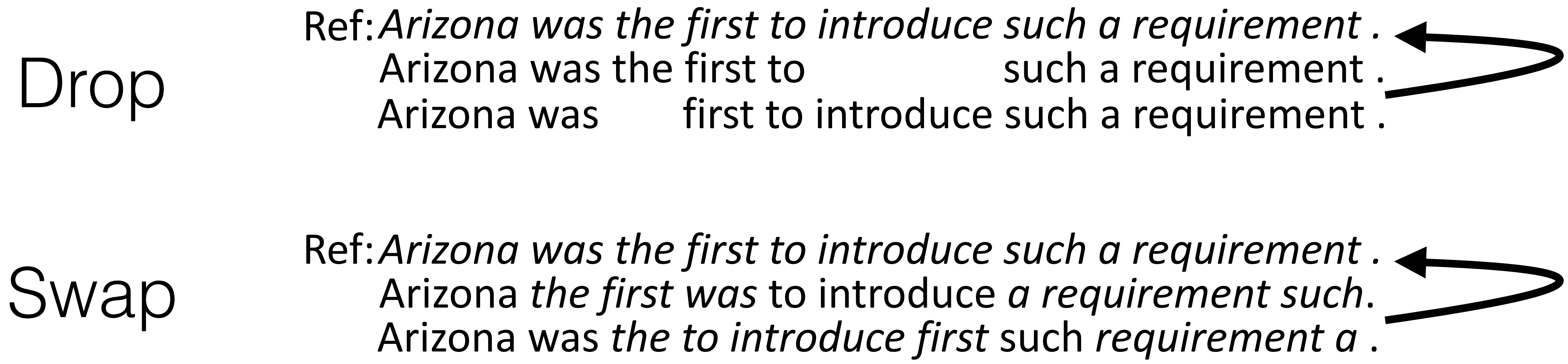
Ref: *Arizona was the first to introduce such a requirement .*
Arizona *the first was* to introduce *a requirement such*.
Arizona was *the to introduce first* such *requirement a* .

From Words to Sentences

How to learn good sentence representations?
We want to train usual sep2seq architecture (as that achieves the best MT results), but without supervision.

Solution: denoising autoencoding task.

Noise: word drop and word swap.



Drop

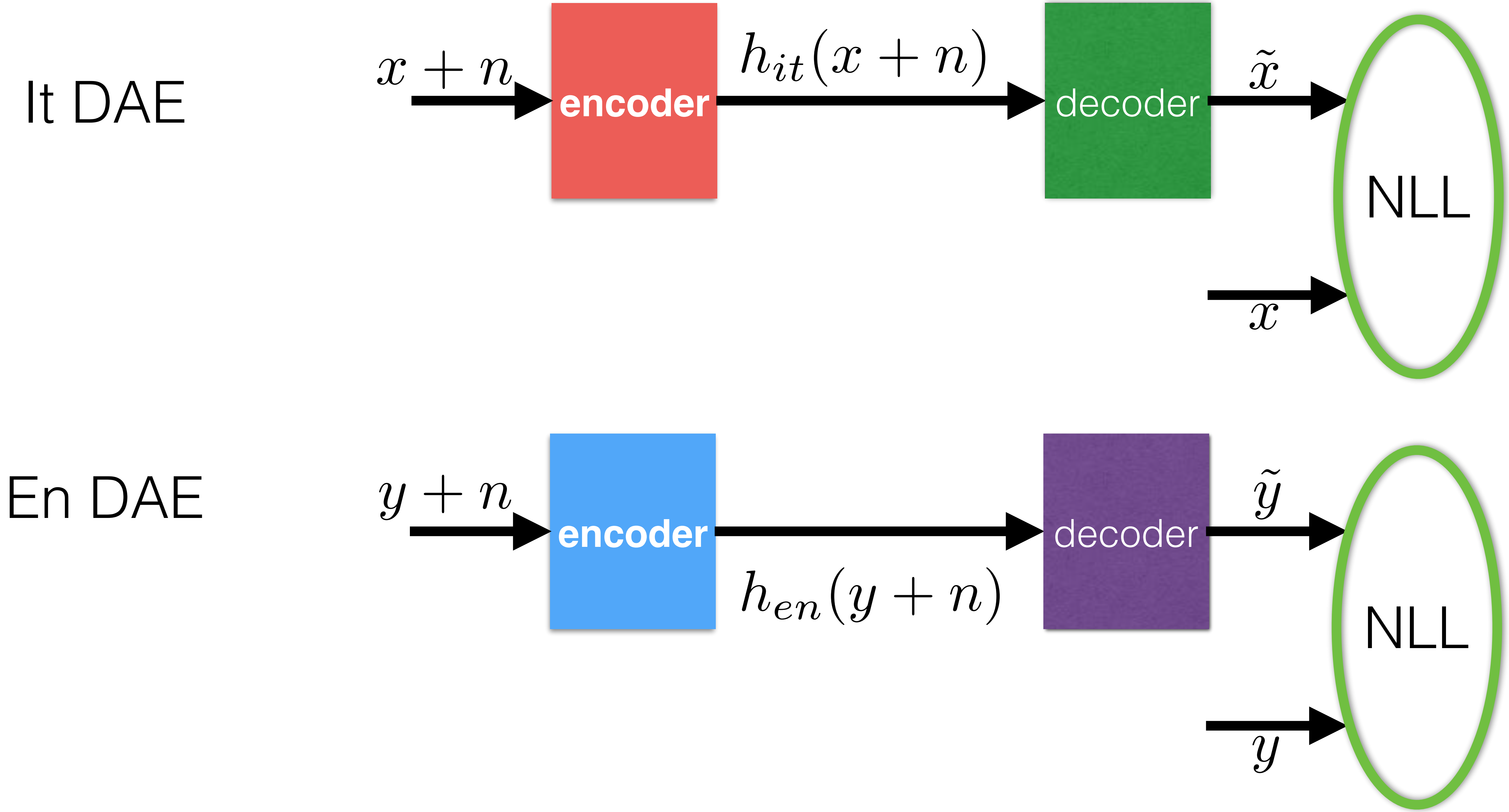
Ref: *Arizona was the first to introduce such a requirement .*
Arizona was the first to such a requirement .
Arizona was first to introduce such a requirement .

Swap

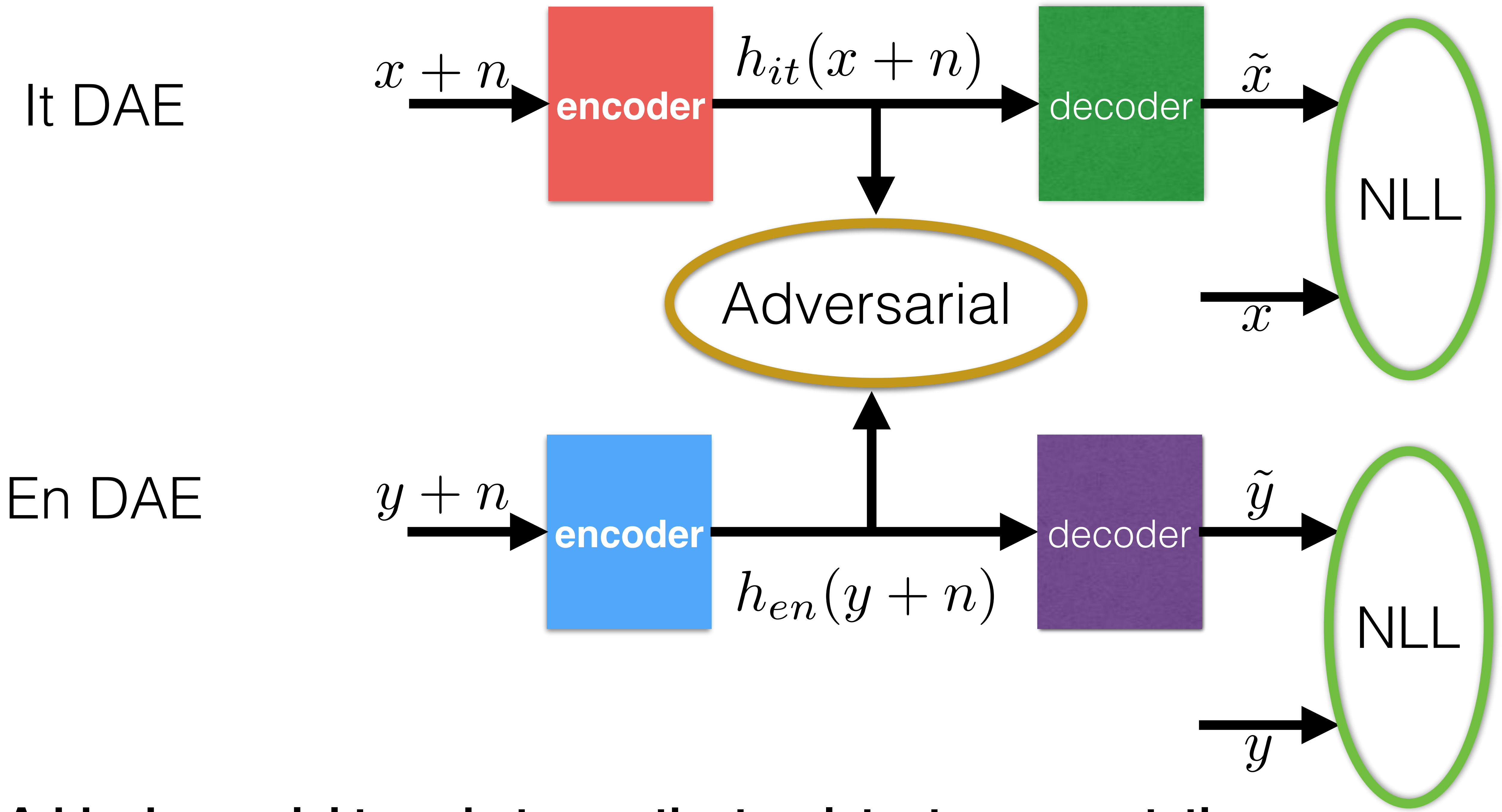
Ref: *Arizona was the first to introduce such a requirement .*
Arizona *the first was* to introduce *a requirement such*.
Arizona was *the to introduce first* such *requirement a* .

Even with attention, the model has to learn regularities in the input (not just copy but a good language model).

Denoising Auto-Encoding

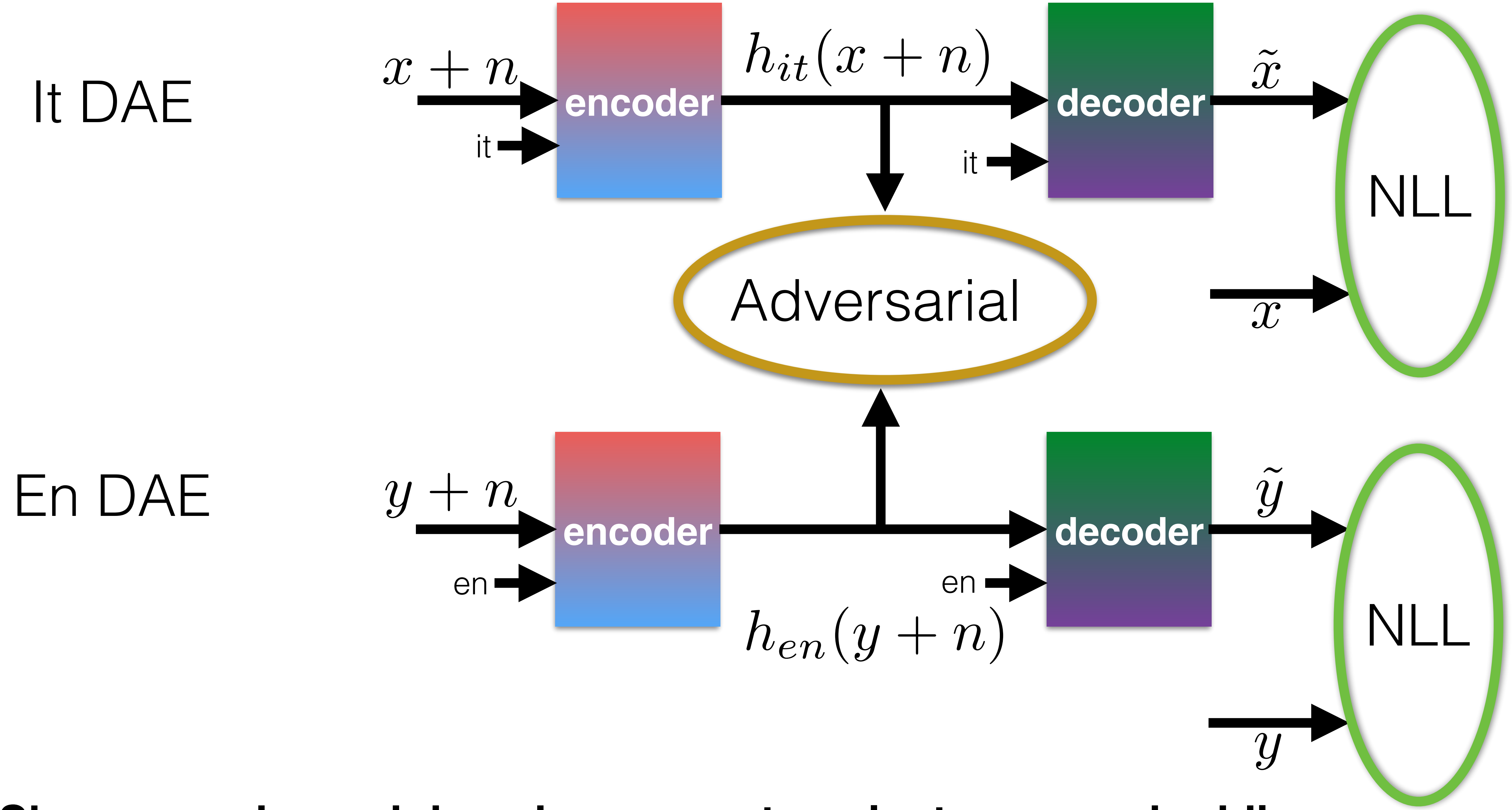


Constraining the Latent Representation



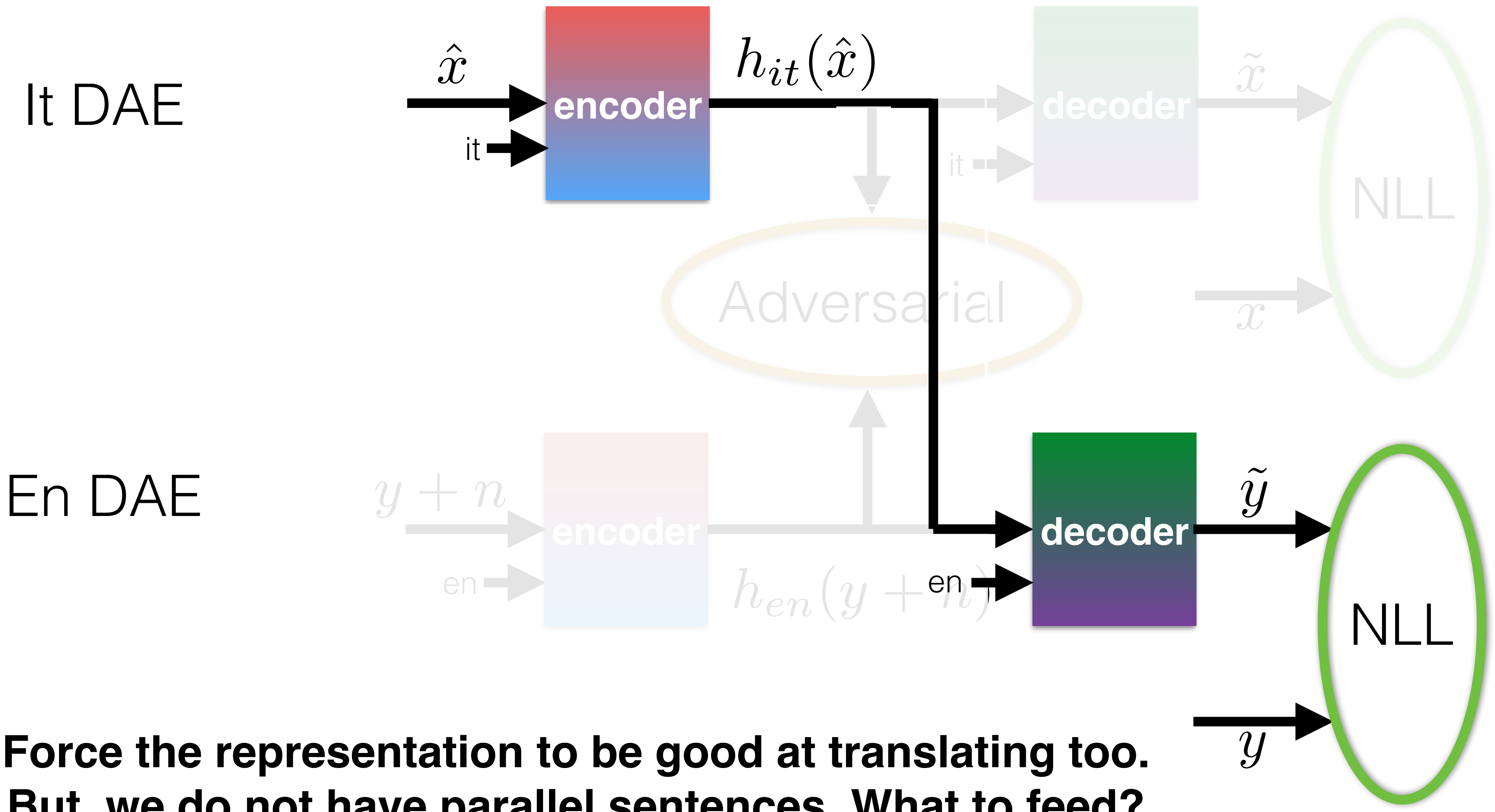
Add adversarial term between the two latent representations.

Constraining the Latent Representation



Share encoder and decoder parameters, just swap embeddings.

Constraining the Latent Representation



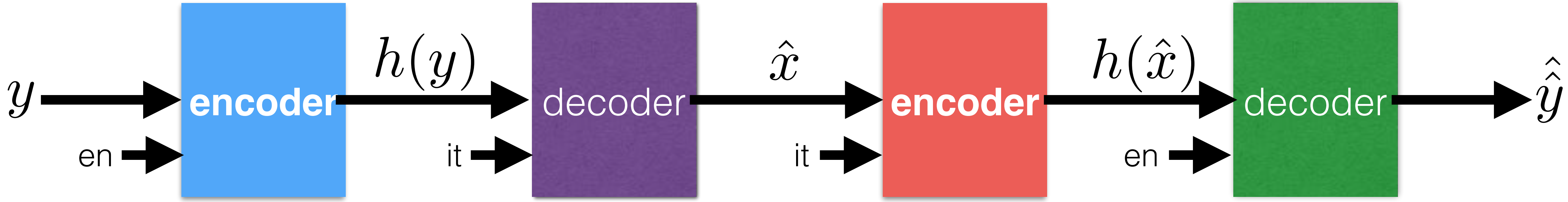
Force the representation to be good at translating too.
But, we do not have parallel sentences. What to feed?

Recap

- Method is a combination of several ingredients:
  - denoising autoencoders
  - translation from artificially generated pairs
  - adversarial loss in latent space
  - parameter sharing
  - initialization from word translation model.
- It's crucial to:
  - somehow share the same latent representation, and
  - to use noise close to actual translation noise.
- If the above two conditions were satisfied and denoising worked well, we could guarantee improvement as we iterate.

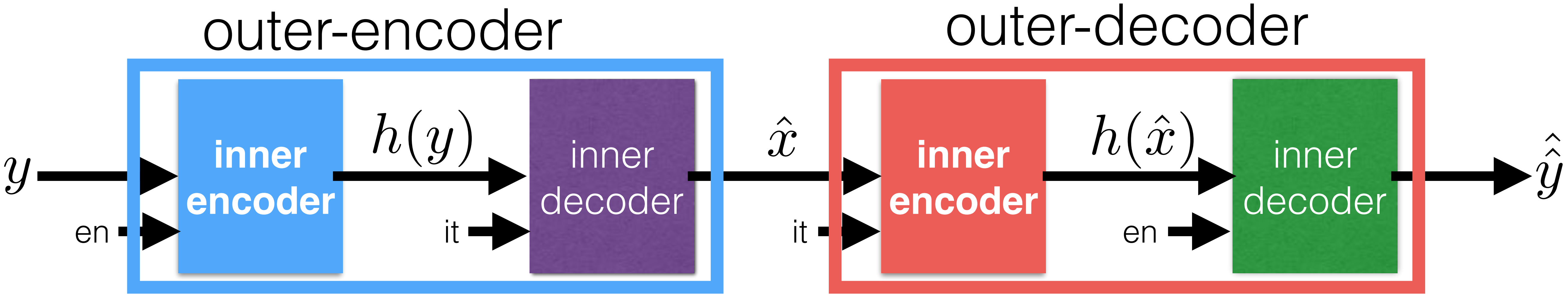
An Alternative View

Since what we are ultimately interested in translation, we can start our construction from the back-translation model and artificially generate parallel sentences.



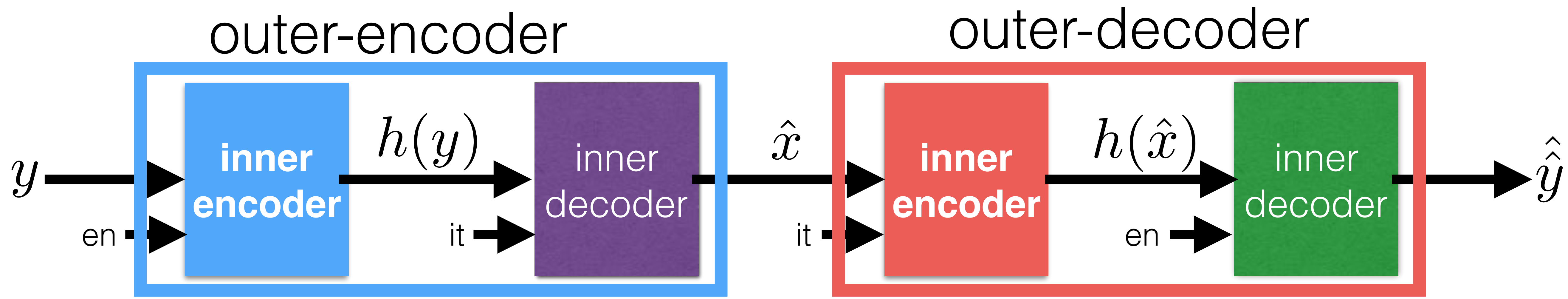
An Alternative View

Since what we are ultimately interested in translation, we can start our construction from the back-translation model and artificially generate parallel sentences.



An Alternative View

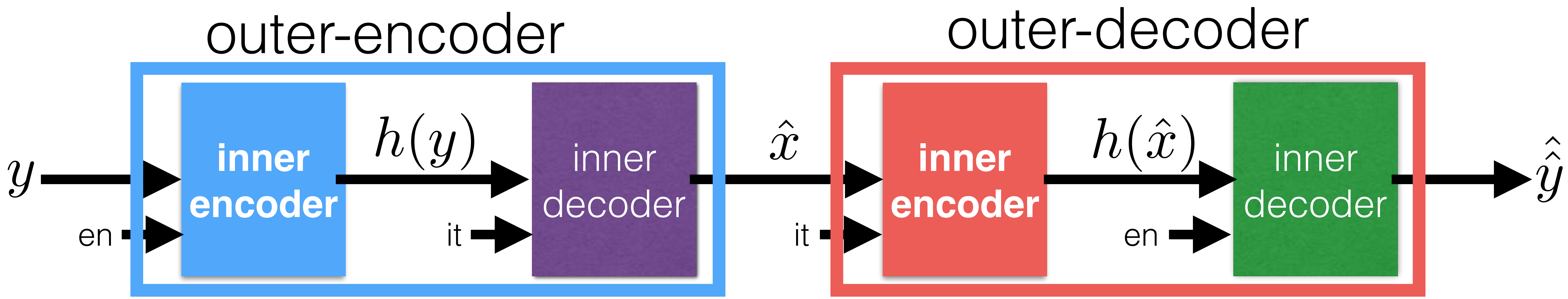
Since what we are ultimately interested in translation, we can start our construction from the back-translation model and artificially generate parallel sentences.



How to constrain the intermediate sentence to be a valid Italian sentence?
It has to be a valid sentence and it has to be a translation.

An Alternative View

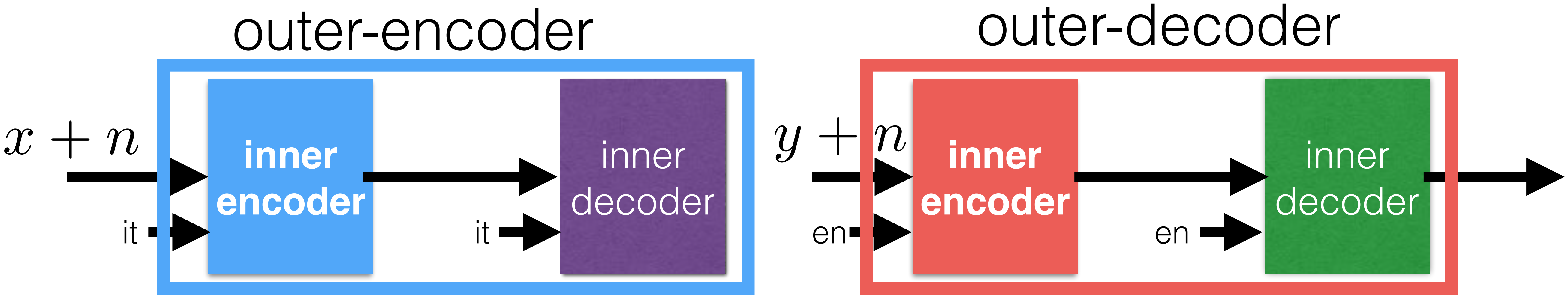
Since what we are ultimately interested in translation, we can start our construction from the back-translation model and artificially generate parallel sentences.



How to constrain the intermediate sentence to be a valid Italian sentence?

- we could add some language modeling constraints directly on $\hat{x}$, but it would be hard to bprop and would be weak constraint on translation.
- instead, we constraint the latent space.

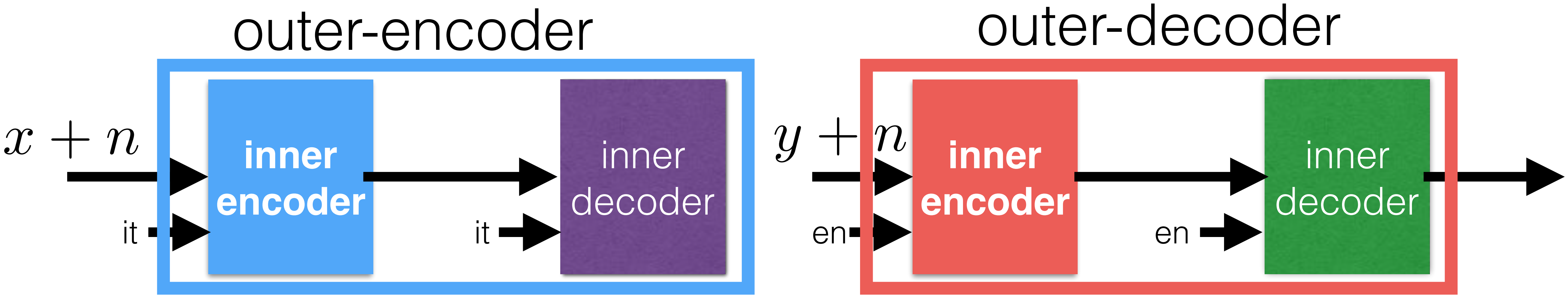
Adding Language Modeling



Since inner decoders are shared between the LM and MT task, it should constraint the intermediate sentence to be fluent.

different.

Adding Language Modeling



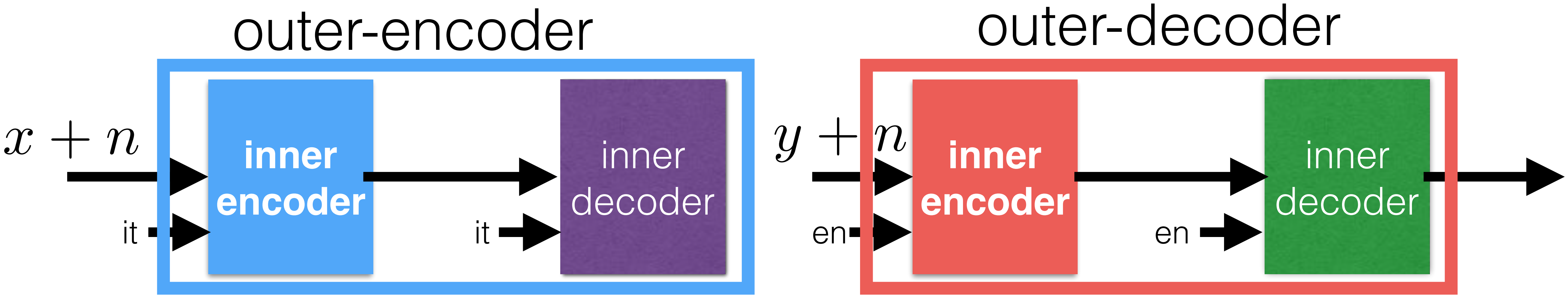
Since inner decoders are shared between the LM and MT task, it should constraint the intermediate sentence to be fluent.

But that's not enough:

- translation noise cannot be exactly reproduced (without parallel data).

→ latent representation may not be robust to translation noise

Adding Language Modeling



Since inner decoders are shared between the LM and MT task, it should constraint the intermediate sentence to be fluent.

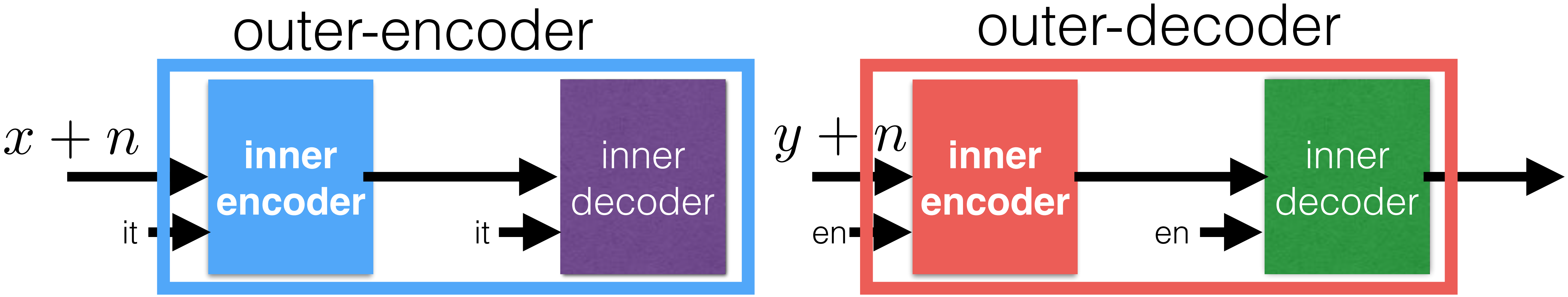
But that's not enough:

- translation noise cannot be exactly reproduced (without parallel data).
- latent representation produced by the "other" inner encoder may be different.

→ NMT won't know how to translate.

M. Ranzato

Adding Language Modeling

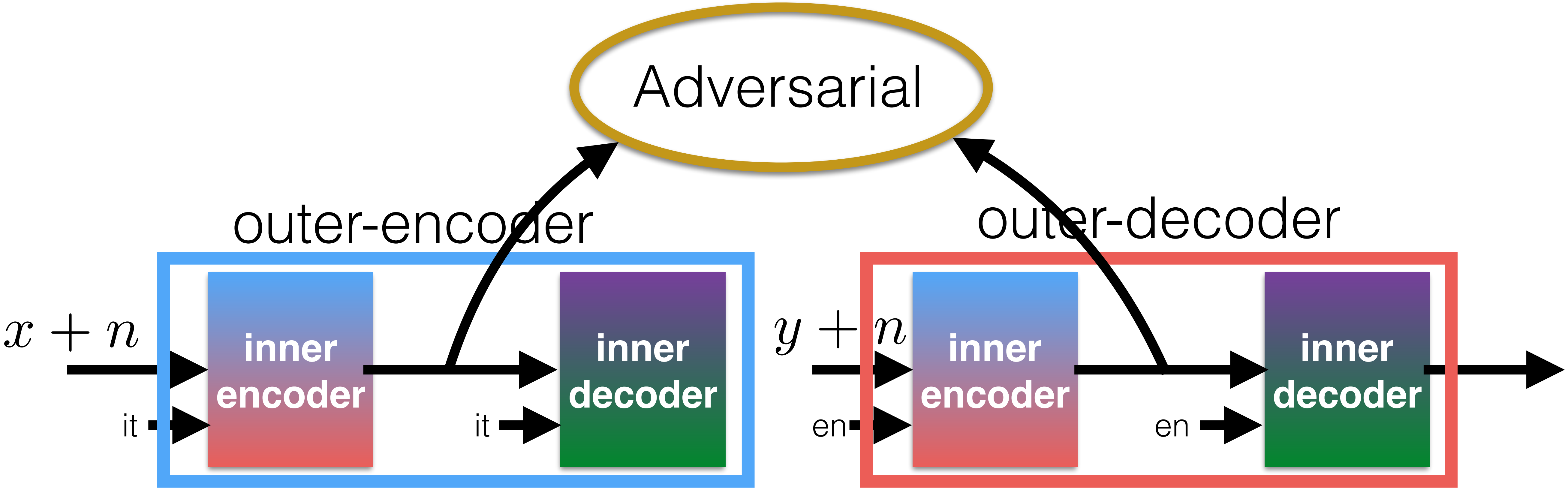


Since inner decoders are shared between the LM and MT task, it should constraint the intermediate sentence to be fluent.

But that's not enough:

- translation noise cannot be exactly reproduced (without parallel data).
- latent representation produced by the "other" inner encoder may be different. WE NEED TO SHARE LATENT REPRESENTATIONS.

Adding Language Modeling



Since inner decoders are shared between the LM and MT task, it should constraint the intermediate sentence to be fluent.

But that’s not enough:

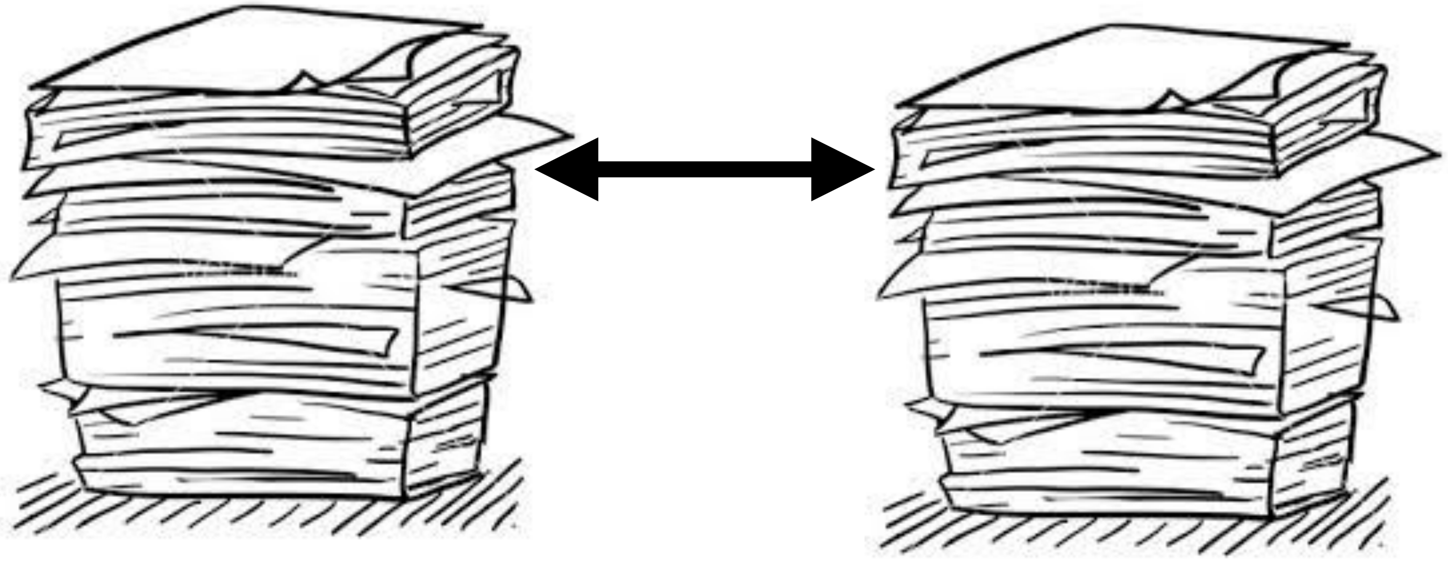
- translation noise cannot be exactly reproduced (without parallel data).
- latent representation produced by the “other” inner encoder may be different. WE NEED TO SHARE LATENT REPRESENTATIONS.

Sharing Representations

Straightforward methods to induce representation sharing between inner encoders when fed with different languages:

- parameter sharing
- adversarial term in latent space.
- Initializing embeddings with word translation mapping.

Methodology

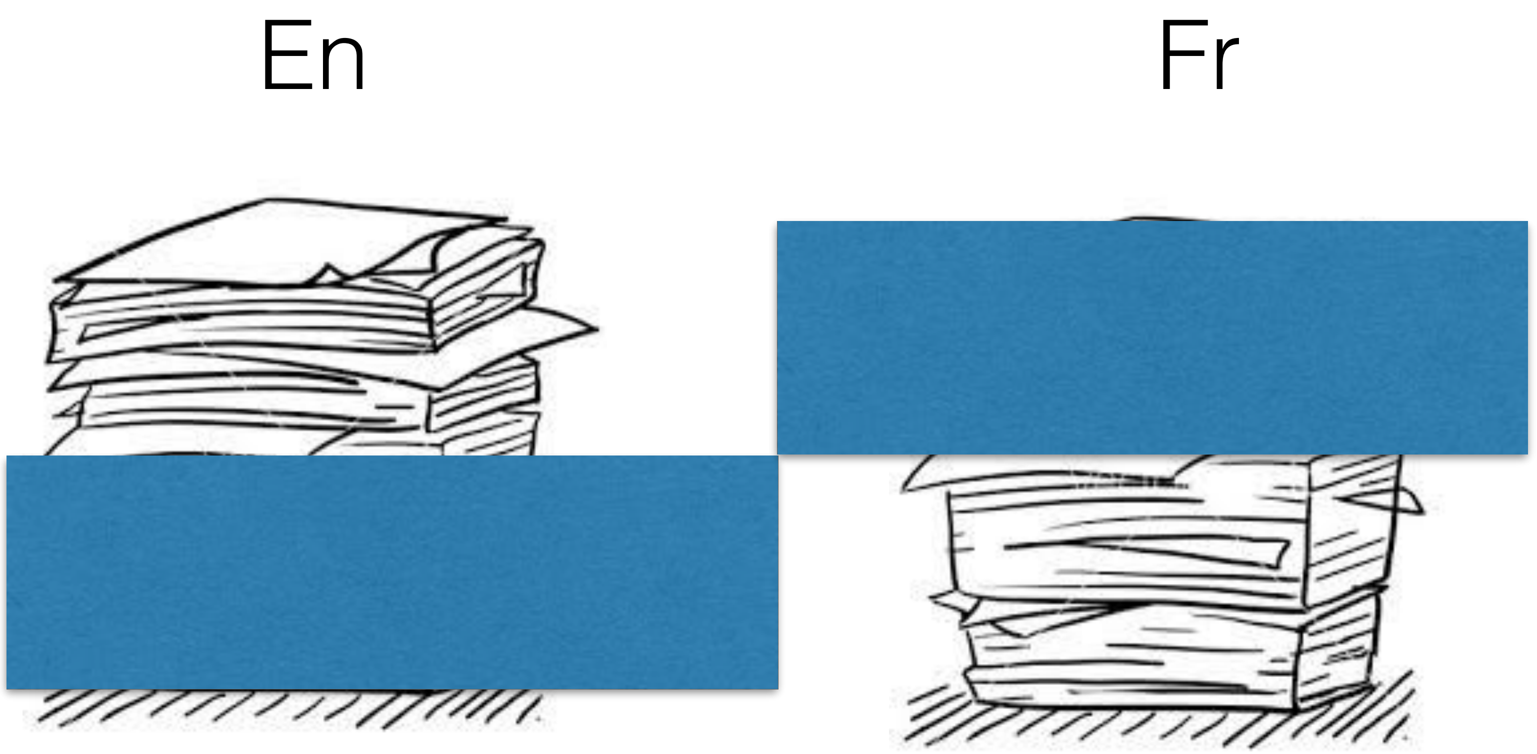


En

Fr

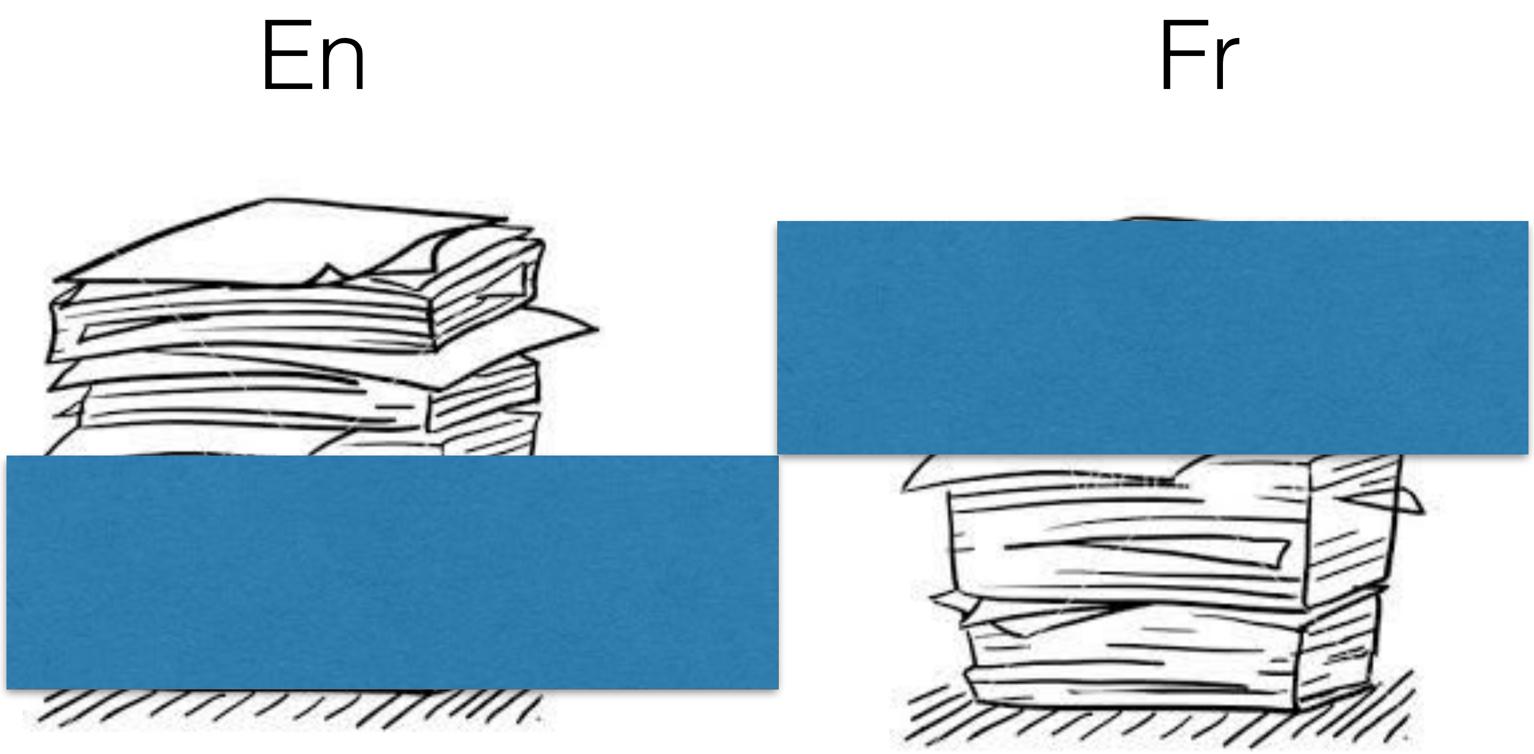
Take commonly used parallel corpus.

Methodology



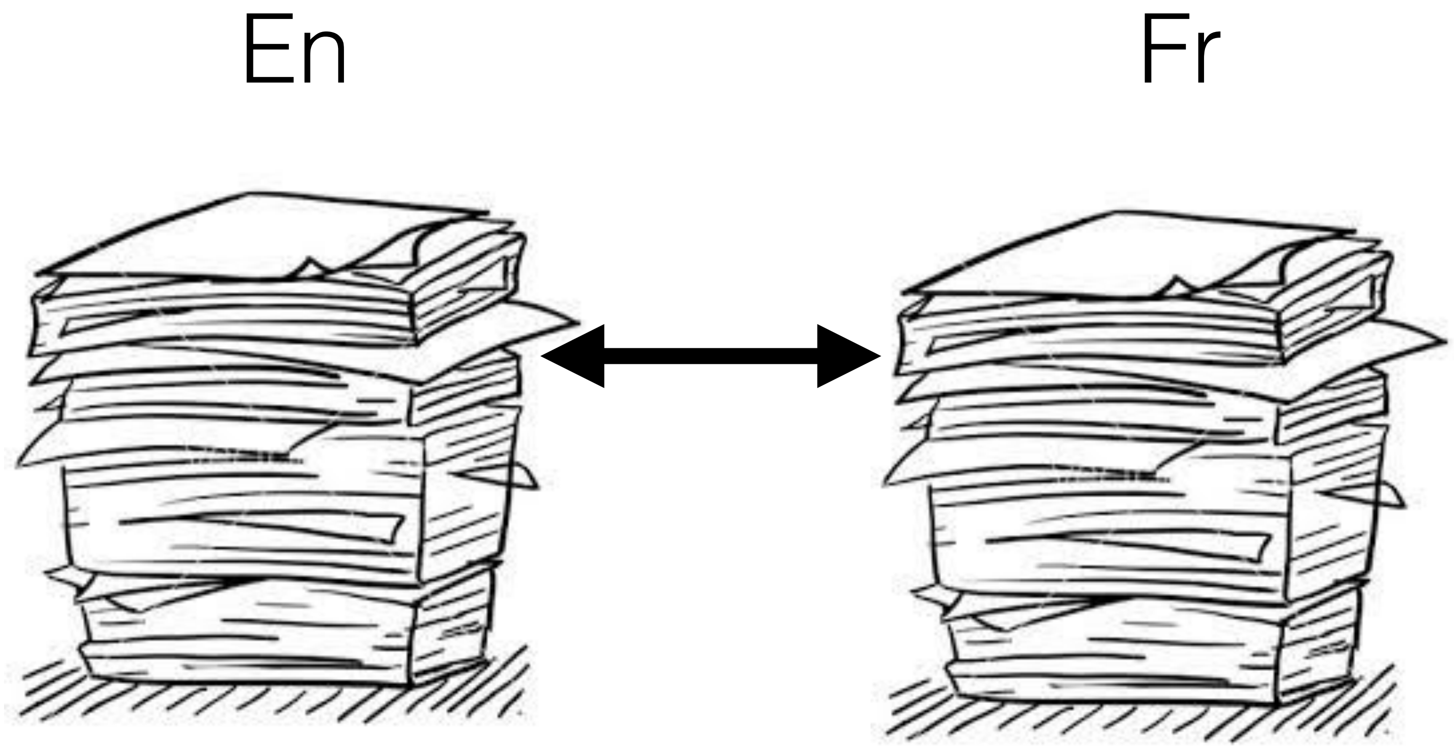
Split training set into two non-overlapping parts to generate monolingual corpora.

Methodology



Train using composite loss
(denoising autoencoding, cross-coding, adversarial)

Methodology

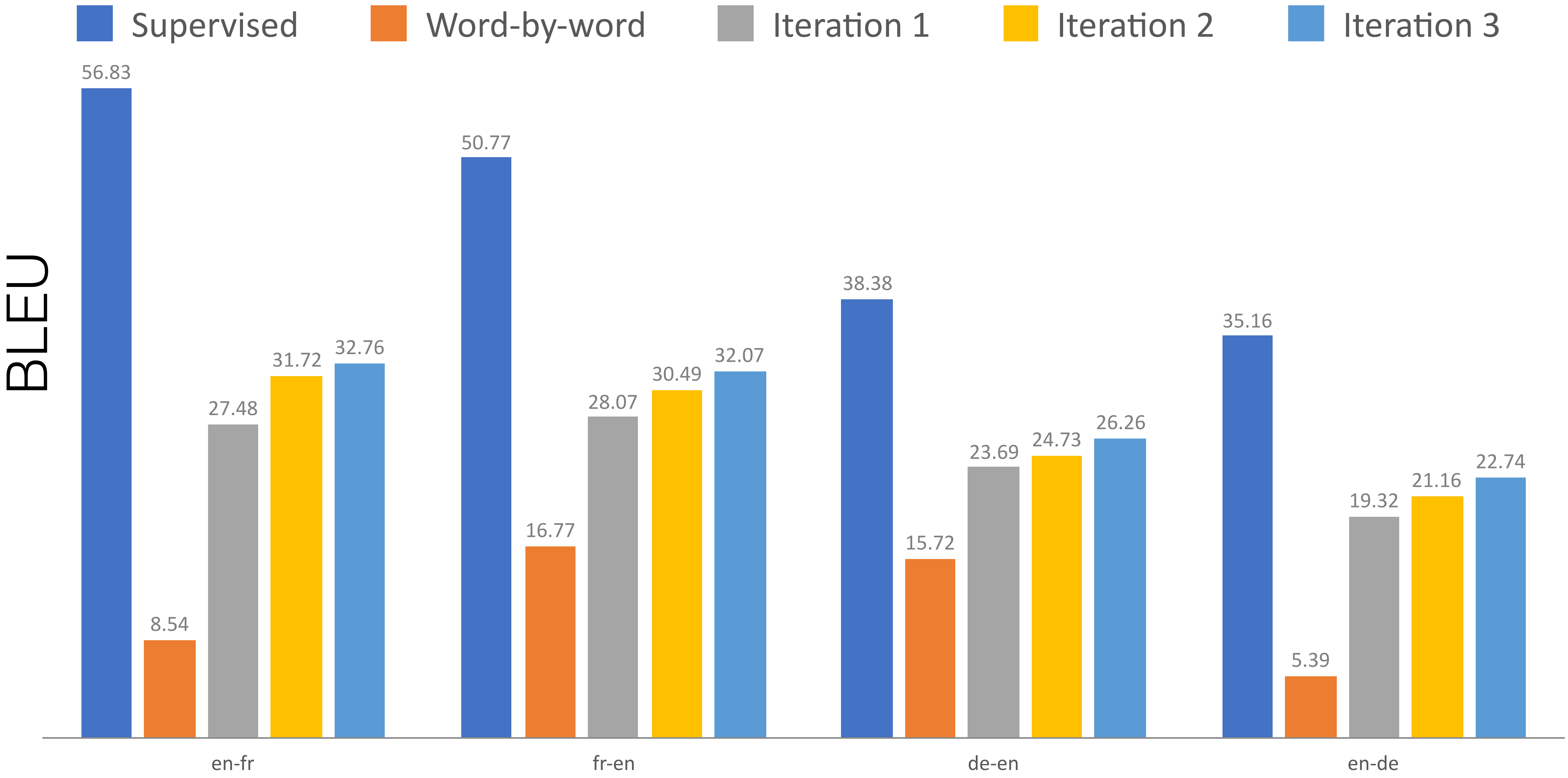


Test on original test set.

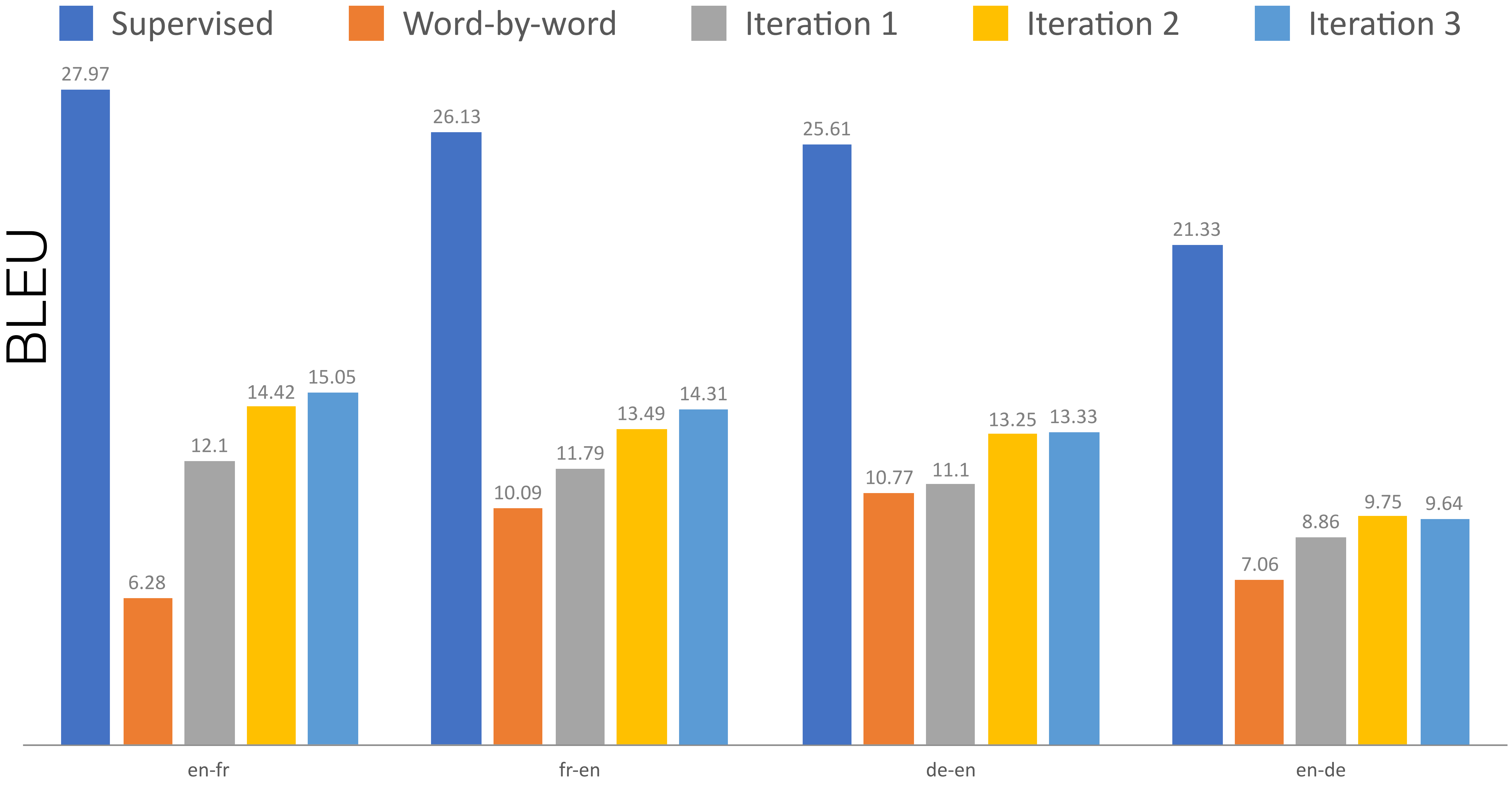
Datasets

- Multi30k-Task1: En-Fr, En-De (without using images)
    - 14.5K captions in each language for training
    - eval on test set
- WMT'14 En-Fr
    - 15M sentences in each language for training
    - eval on newstest2014
- WMT'16 En-De
    - 1.8M sentences in each language for training
    - eval on newstest2016

Results on Multi30K-Task1



Results on WMT



M. Ranzato

Improvements by Iterating

Source	une femme aux cheveux roses habillée en noir parle à un homme .
Iteration 0	
Iteration 1	
Iteration 2	
Iteration 3	
Reference	a woman with pink hair dressed in black talks to a man .

Improvements by Iterating

Source	une femme aux cheveux roses habillée en noir parle à un homme .
Iteration 0	a woman at hair roses dressed in black speaks to a man .
Iteration 1	
Iteration 2	
Iteration 3	
Reference	a woman with pink hair dressed in black talks to a man .

iteration 0 is word-by-word translation using the unsupervised word translation model.

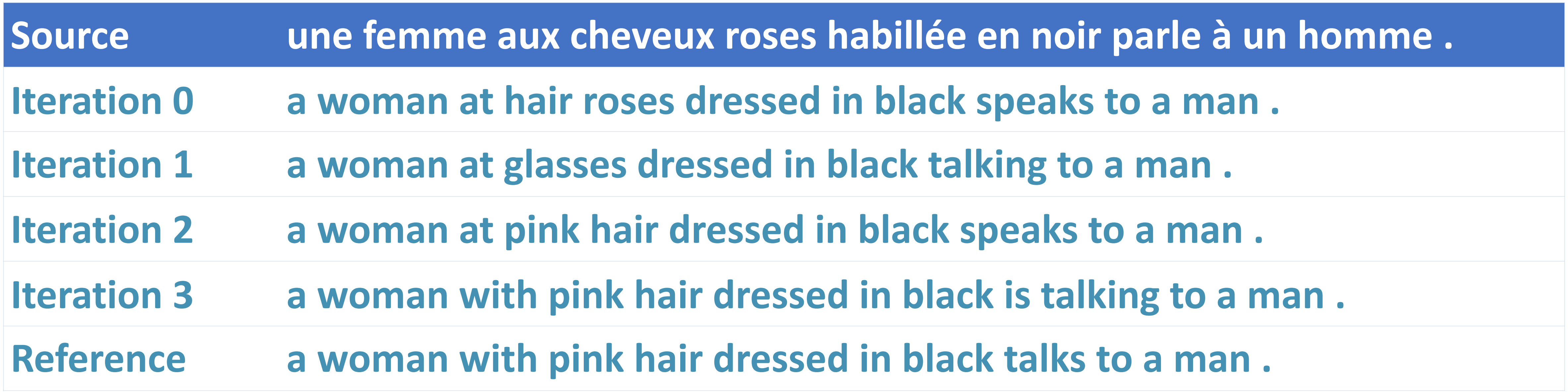
Improvements by Iterating

Source	une femme aux cheveux roses habillée en noir parle à un homme .
Iteration 0	a woman at hair roses dressed in black speaks to a man .
Iteration 1	a woman at glasses dressed in black talking to a man .
Iteration 2	
Iteration 3	
Reference	a woman with pink hair dressed in black talks to a man .

Improvements by Iterating

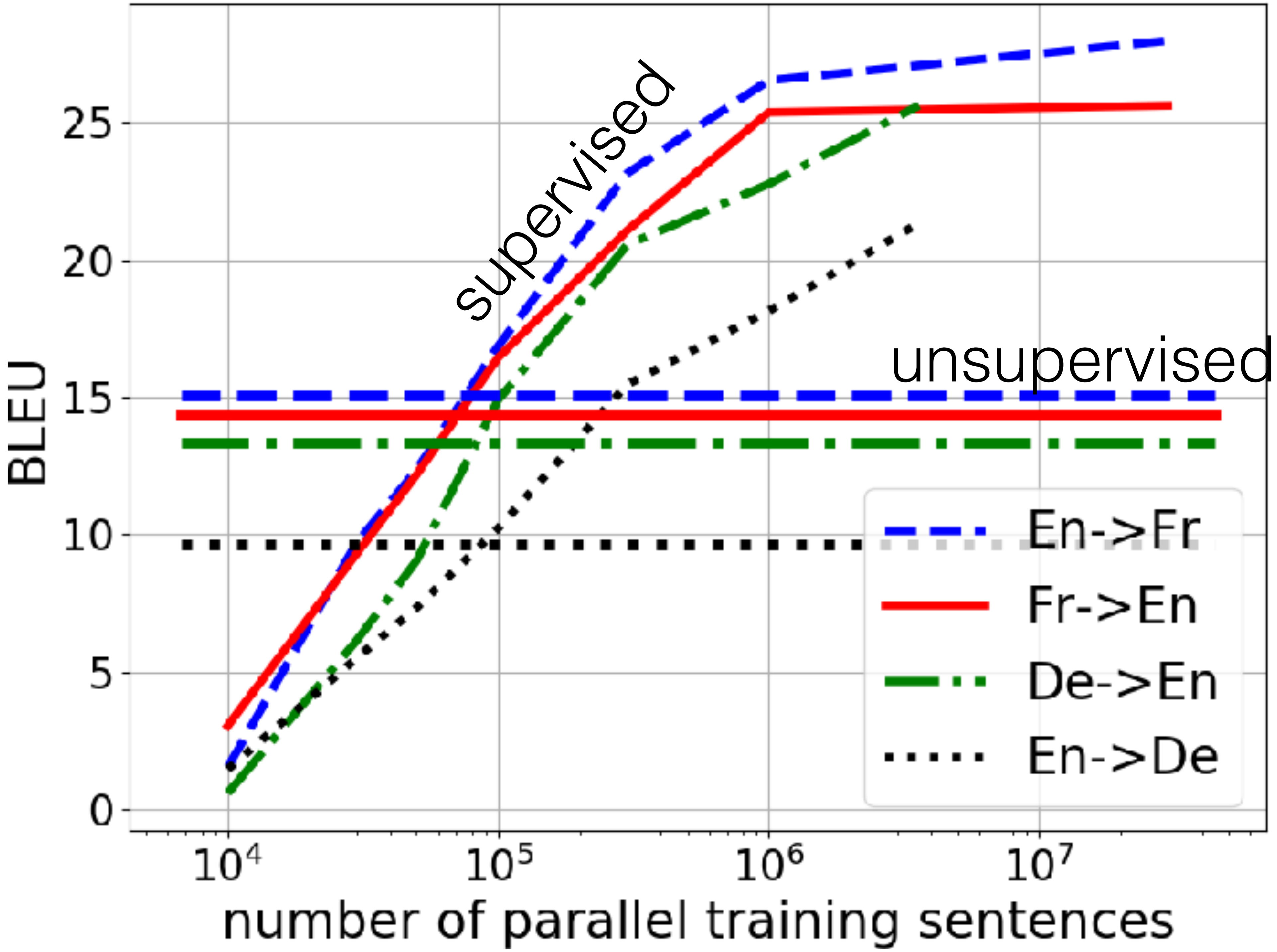
Source	une femme aux cheveux roses habillée en noir parle à un homme .
Iteration 0	a woman at hair roses dressed in black speaks to a man .
Iteration 1	a woman at glasses dressed in black talking to a man .
Iteration 2	a woman at pink hair dressed in black speaks to a man .
Iteration 3	
Reference	a woman with pink hair dressed in black talks to a man .

Improvements by Iterating



Source	une femme aux cheveux roses habillée en noir parle à un homme .
Iteration 0	a woman at hair roses dressed in black speaks to a man .
Iteration 1	a woman at glasses dressed in black talking to a man .
Iteration 2	a woman at pink hair dressed in black speaks to a man .
Iteration 3	a woman with pink hair dressed in black is talking to a man .
Reference	a woman with pink hair dressed in black talks to a man .

Supervised VS Unsupervised



Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
Full	**27.48**	**28.07**	**23.69**	**19.32**

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
$\lambda_{cd} = 0$	25.44	27.14	20.56	14.42
Full	27.48	28.07	23.69	19.32

- λ_{cd} cross-domain loss coefficient. No translation task.

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
$\lambda_{cd} = 0$	25.44	27.14	20.56	14.42
Without pretraining	25.29	26.10	21.44	17.23
Full	27.48	28.07	23.69	19.32

- λ_{cd} cross-domain loss coefficient. No translation task.

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
$\lambda_{cd} = 0$	25.44	27.14	20.56	14.42
Without pretraining	25.29	26.10	21.44	17.23
Without pretraining, $\lambda_{cd} = 0$	8.78	9.15	7.52	6.24
Full	27.48	28.07	23.69	19.32

- λ_{cd} cross-domain loss coefficient. No translation task.

On this dataset, it's important to either initialize the embeddings or to add the translation task.

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
λ_{cd} = 0	25.44	27.14	20.56	14.42
Without pretraining	25.29	26.10	21.44	17.23
Without pretraining, λ_{cd} = 0	8.78	9.15	7.52	6.24
Without noise, C(x) = x	16.76	16.85	16.85	14.61
Full	27.48	28.07	23.69	19.32

- λ_{cd} cross-domain loss coefficient. No translation task.

Noise is important to learn good representations.

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
$\lambda_{cd} = 0$	25.44	27.14	20.56	14.42
Without pretraining	25.29	26.10	21.44	17.23
Without pretraining, $\lambda_{cd} = 0$	8.78	9.15	7.52	6.24
Without noise, $C(x) = x$	16.76	16.85	16.85	14.61
$\lambda_{auto} = 0$	24.32	20.02	19.10	14.74
Full	27.48	28.07	23.69	19.32

- λ_{cd} cross-domain loss coefficient. No translation task.

Ablation Study on Multi30K-Task1

	en-fr	fr-en	de-en	en-de
$\lambda_{cd} = 0$	25.44	27.14	20.56	14.42
Without pretraining	25.29	26.10	21.44	17.23
Without pretraining, $\lambda_{cd} = 0$	8.78	9.15	7.52	6.24
Without noise, $C(x) = x$	16.76	16.85	16.85	14.61
$\lambda_{auto} = 0$	24.32	20.02	19.10	14.74
$\lambda_{adv} = 0$	24.12	22.74	19.87	15.13
Full	27.48	28.07	23.69	19.32

Summary

- To some extent, we can learn to translate without any supervision, using monolingual data only.
- It’s key to constrain the model to produce valid intermediate sentences. We did so in the feature space.
    - Use noise that is proxy of translation noise.
    - Induce sharing of latent representations.

Future Work

- Figure out which constraints are universally useful and efficient.
- Leverage small amounts of labeled data, and large amounts of labeled data from other language pairs.
- Test method on languages for which we really do not have labeled data.

Summary of the Lecture

- To understand what is worth working on, we need to come up with better tools to analyze current models.

- Some well-known challenges of NMT can be easily explained and resolved.

- In general, model fitting needs more attention than search. Training at the sequence level works better but with diminishing returns as the baseline model gets stronger.

- The same principle of aligning domains in feature space can be used to translate words and sentences in a fully unsupervised manner; but there is a lot of room for improvement.

References

[1] **Classical Structured Prediction Losses for Sequence to Sequence Learning**
Sergey Edunov, Myle Ott, Michael Auli, David Grangier, Marc'Aurelio Ranzato
NAACL 2018
https://arxiv.org/abs/1711.04956

[2] **Analyzing Uncertainty in Neural Machine Translation**
Myle Ott, Michael Auli, David Granger, Marc'Aurelio Ranzato
submitted
https://arxiv.org/abs/1803.00047

[3] **Word Translation Without Parallel Data**
Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, Herve Jegou
ICLR 2018
https://arxiv.org/abs/1710.04087 CODE: https://github.com/facebookresearch/MUSE

[4] **Unsupervised Machine Translation Using Monolingual Corpora Only**
Guillaume Lample, Ludovic Denoyer, Marc'Aurelio Ranzato
ICLR 2018
https://arxiv.org/abs/1711.00043

Collaborators

Myle Ott

Sergey Edunov

Michael Auli

David Grangier

Guillaume Lample

Alexis Conneau

Herve Jegou

Ludovic Denoyer

THANK YOU

Questions?
Вопросы?
¿Preguntas?