Current trends and tricks of the trade in NMT

Barry Haddow

University of Edinburgh

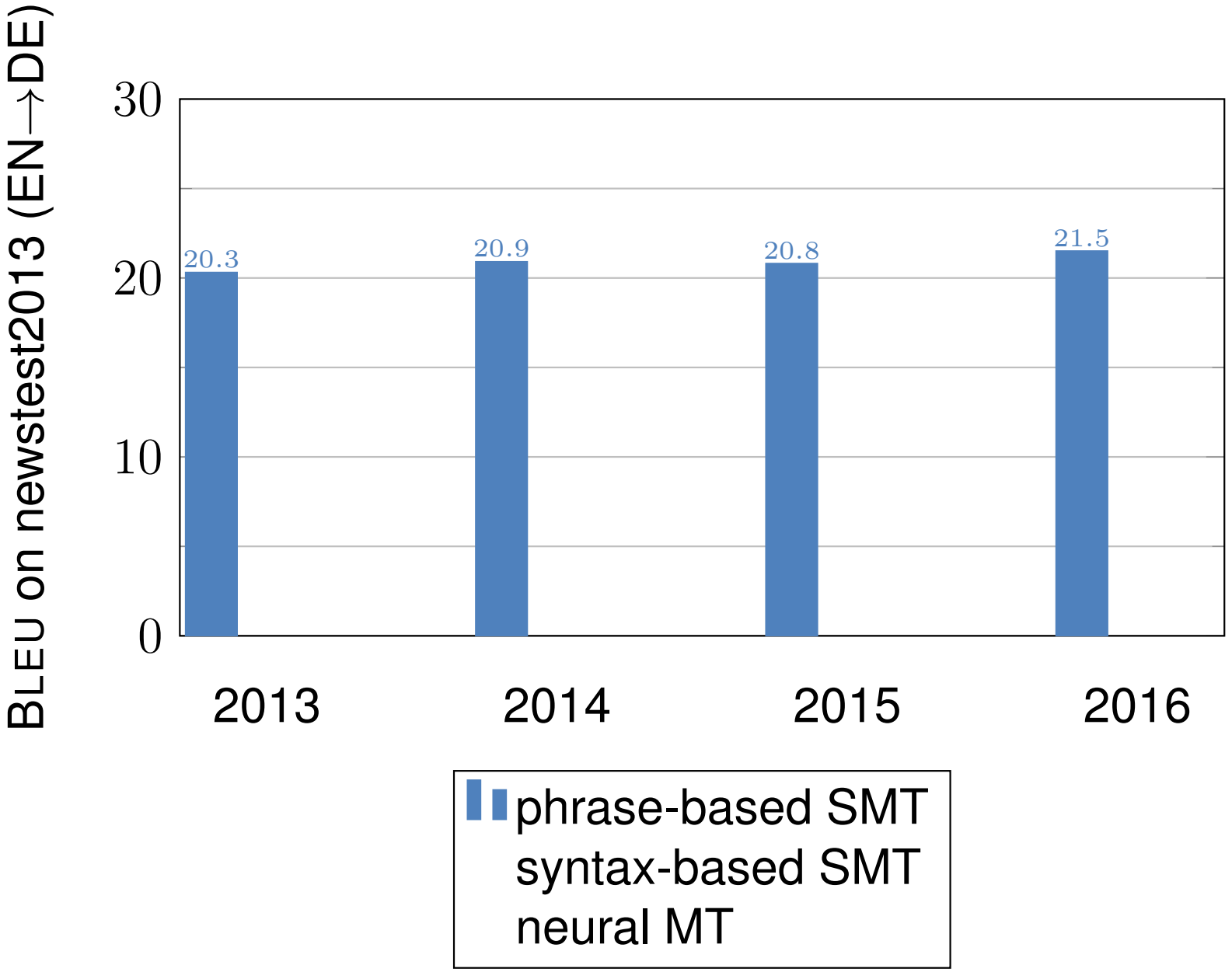
FINMT
September 12th, 2016

Collaborators

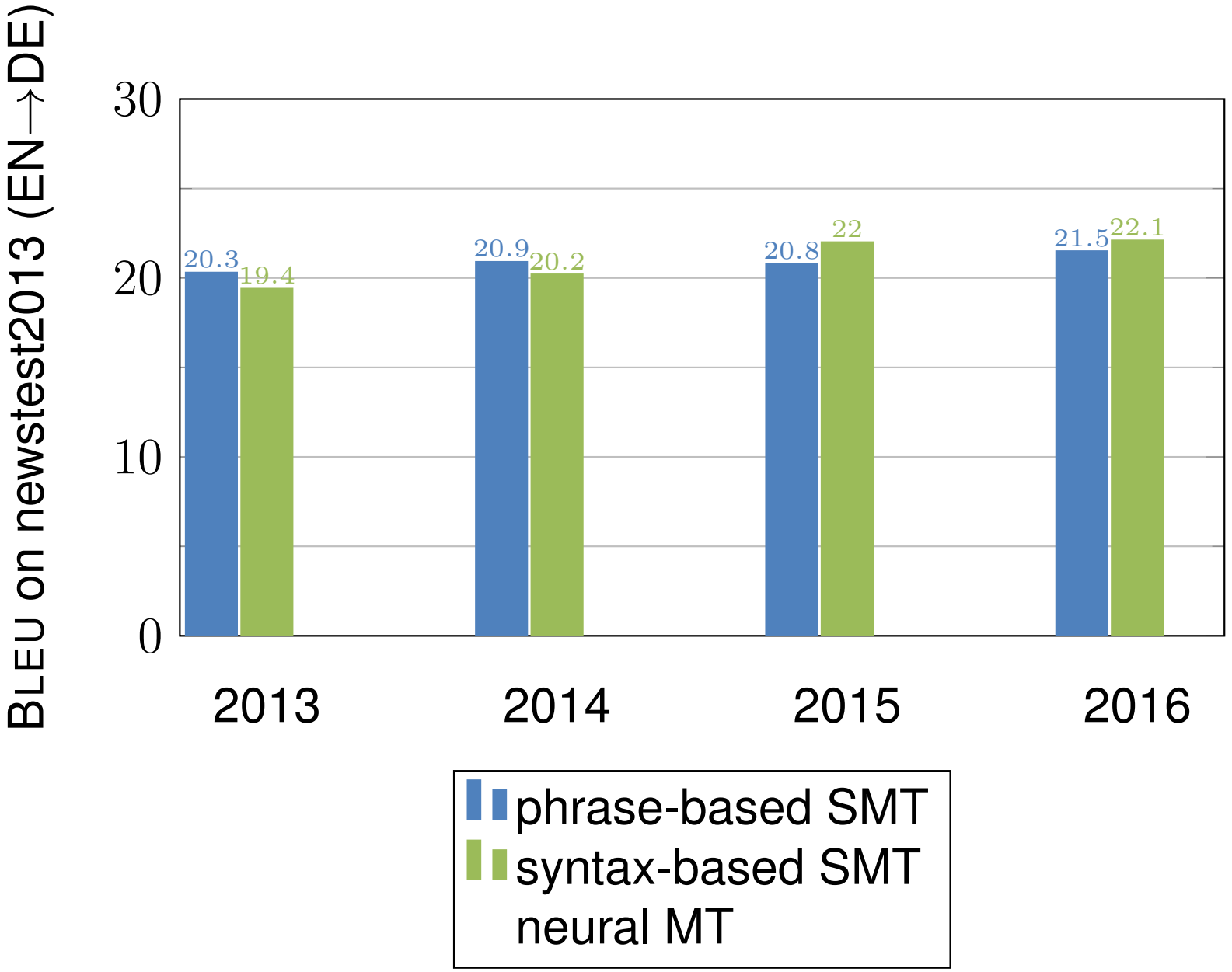
Rico Sennrich

Alexandra Birch

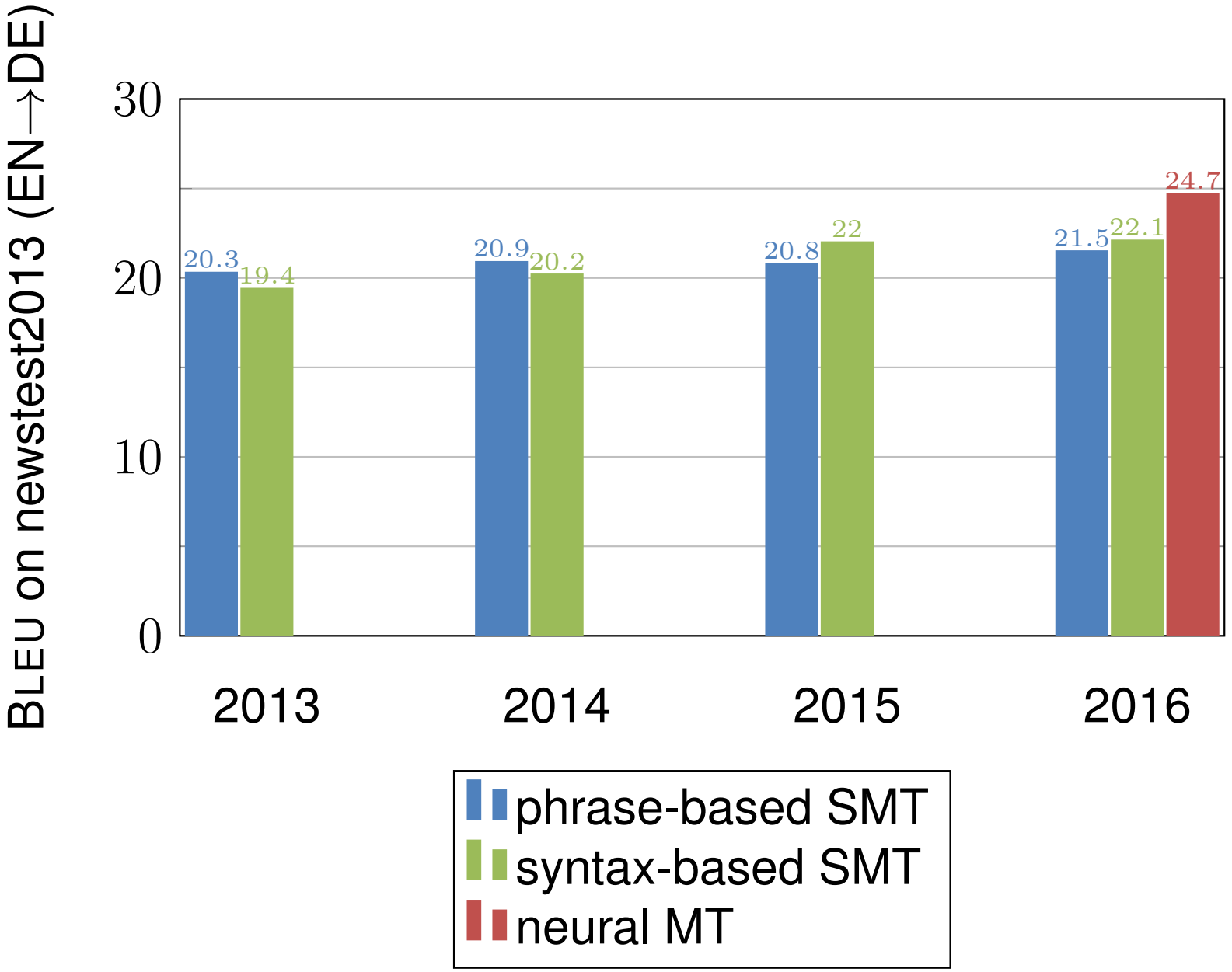
Edinburgh's WMT Results Over the Years



Edinburgh's WMT Results Over the Years



Edinburgh's WMT Results Over the Years



WMT16: English→German

#	score	range	system
1	0.49	1	UEDIN-NMT
2	0.40	2	METAMIND
3	0.29	3	UEDIN-SYNTAX
4	0.17	4	NYU-MONTREAL
5	−0.01	5-10	ONLINE-B
	−0.01	5-10	KIT-LIMSI
	−0.02	5-10	CAMBRIDGE
	−0.02	5-10	ONLINE-A
	−0.03	5-10	PROMT-RULE
	−0.05	6-10	KIT
6	−0.14	11-12	JHU-SYNTAX
	−0.15	11-12	JHU-PBMT
7	−0.26	13-14	UEDIN-PBMT
	−0.33	13-15	ONLINE-F
	−0.34	14-15	ONLINE-G

WMT16: English→German

#	score	range	system
1	0.49	1	UEDIN-NMT
2	0.40	2	METAMIND
3	0.29	3	UEDIN-SYNTAX
4	0.17	4	NYU-MONTREAL
5	−0.01	5-10	ONLINE-B
	−0.01	5-10	KIT-LIMSI
	−0.02	5-10	CAMBRIDGE
	−0.02	5-10	ONLINE-A
	−0.03	5-10	PROMT-RULE
	−0.05	6-10	KIT
6	−0.14	11-12	JHU-SYNTAX
	−0.15	11-12	JHU-PBMT
7	−0.26	13-14	UEDIN-PBMT
	−0.33	13-15	ONLINE-F
	−0.34	14-15	ONLINE-G

Neural MT

WMT16: English→German

#	score	range	system
1	0.49	1	UEDIN-NMT
2	0.40	2	METAMIND
3	0.29	3	UEDIN-SYNTAX
4	0.17	4	NYU-MONTREAL
5	−0.01	5-10	ONLINE-B
	−0.01	5-10	KIT-LIMSI
	−0.02	5-10	CAMBRIDGE
	−0.02	5-10	ONLINE-A
	−0.03	5-10	PROMT-RULE
	−0.05	6-10	KIT
6	−0.14	11-12	JHU-SYNTAX
	−0.15	11-12	JHU-PBMT
7	−0.26	13-14	UEDIN-PBMT
	−0.33	13-15	ONLINE-F
	−0.34	14-15	ONLINE-G

Neural MT Neural components

Outline

- Brief overview of NMT
- Recent Advances
  - Subword representations
  - Using monolingual data
  - Linguistic features for NMT
- Current problems and outlook
- Practical considerations

Neural versus Phrase-based MT

phrase-based SMT

Learn segment-segment correspondences from bitext

- → training is multistage pipeline of heuristics
- → strong independence assumptions
- → “fixed” trade-off between features

Neural versus Phrase-based MT

phrase-based SMT

Learn segment-segment correspondences from bitext

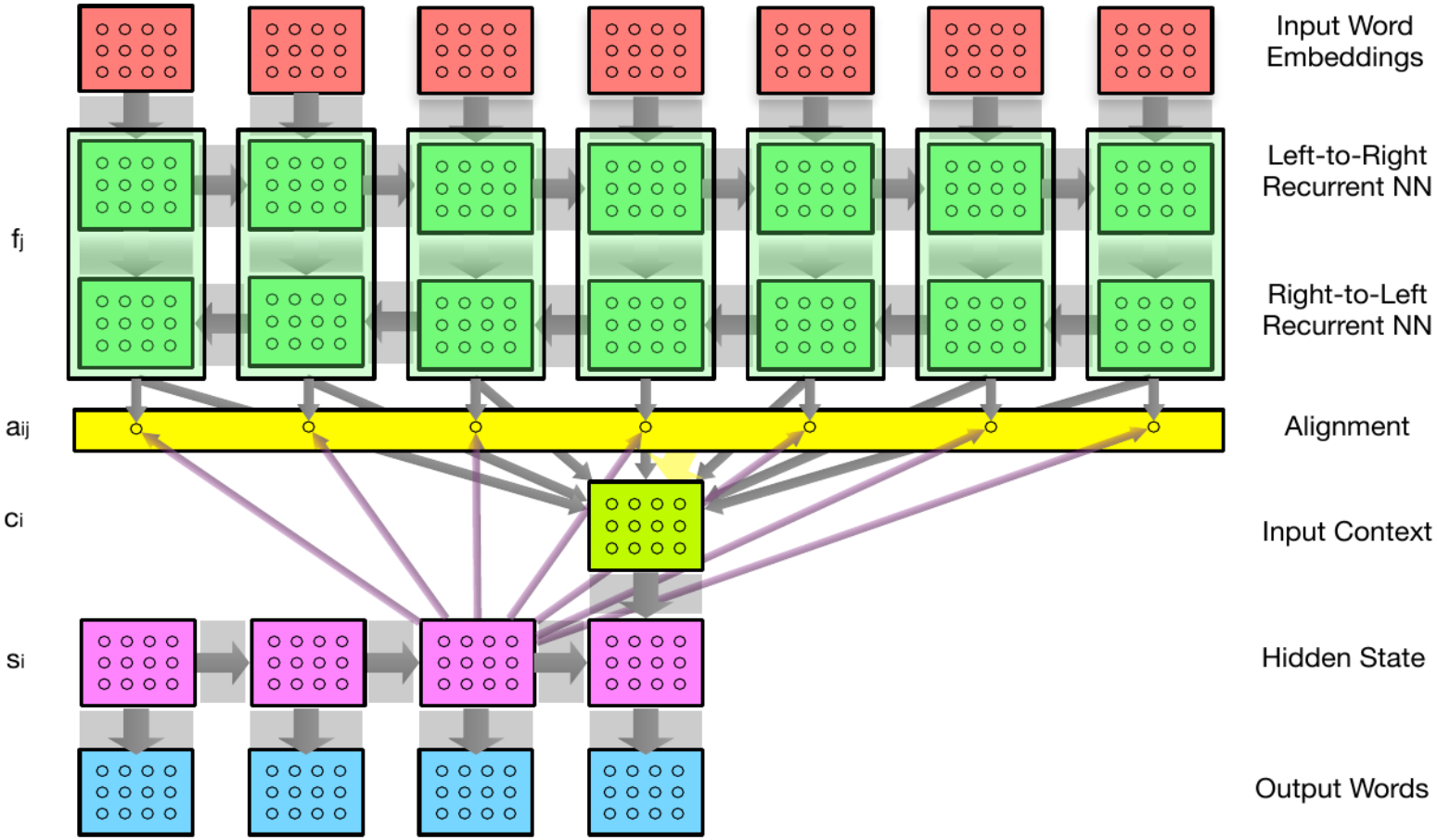
- → training is multistage pipeline of heuristics
- → strong independence assumptions
- → “fixed” trade-off between features

neural MT

Learn mathematical function on vectors from bitext

- → end-to-end trained model
- → output conditioned on full source text and target history
- → non-linear dependence on information sources

Neural Machine Translation: Encoder-Decoder-with-Attention



[Image: Philipp Koehn]

Simplified Equations

Encoder

$$\overrightarrow{h}_j = RNN(x_j, \overrightarrow{h}_{j-1})$$
$$\overleftarrow{h}_j = RNN(x_j, \overleftarrow{h}_{j+1})$$
$$h_j = (\overrightarrow{h}_j; \overleftarrow{h}_j)$$

Decoder-with-Attention

$$e_{ij} = f(s_{i-1}, h_j)$$
$$c_i = \sum_j \text{softmax}_j(e_{ij}) h_j$$
$$s_i = RNN(y_{i-1}, s_{i-1}, c_i)$$

Readout

$$y_i = \text{softmax}(g(s_i, c_i, y_{i-1}))$$

See `https://github.com/emjotde/amunmt/blob/master/notebooks/dl4mt.ipynb`

Training and Inference

Training

- Minimise cross-entropy loss function
- Gradient descent using back-propagation
- Process training data in mini-batches

Inference

- Beam search with small beam (e.g. 12)
- Typically use an ensemble of models

Recent Advances in Neural MT

- some problems:
  - networks have fixed vocabulary
    $\rightarrow$ poor translation of rare/unknown words
  - models are trained on parallel data; how do we use monolingual data?
  - how do we incorporate linguistic mark-up?
- recent solutions:
  - subword models allow translation of rare/unknown words
  - train on back-translated monolingual data
  - linguistic input features

Subwords for NMT: Motivation

MT is an open-vocabulary problem

- compounding and other productive morphological processes
  - they charge a carry-on bag fee.
  - sie erheben eine Hand|gepäck|gebühr.
- names
  - Obama(English; German)
  - Обама (Russian)
  - オバマ (o-ba-ma) (Japanese)
- technical terms, numbers, etc.

... but Neural MT architectures have small and fixed vocabulary

Subword units

segmentation algorithms: wishlist

- **open-vocabulary NMT**: encode *all* words through small vocabulary
- encoding generalizes to unseen words
- small text size
- good translation quality

our experiments

- after preliminary experiments, we use:
  - character n-grams (with shortlist of unsegmented words)
  - segmentation via *byte pair encoding*

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'l o w </w>'	5
'l o w e r </w>'	2
'n e w e s t </w>'	6
'w i d e s t </w>'	3

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'l o w </w>'	5
'l o w e r </w>'	2
'n e w es t </w>'	6
'w i d es t </w>'	3

('e', 's') $\rightarrow$ 'es'

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'l o w </w>'	5
'l o w e r </w>'	2
'n e w est </w>'	6
'w i d est </w>'	3

('e', 's') → 'es'
('es', 't') → 'est'

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'l o w </w>'	5
'l o w e r </w>'	2
'n e w est</w>'	6
'w i d est</w>'	3

('e', 's')	→	'es'
('es', 't')	→	'est'
('est', '</w>')	→	'est</w>'

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'lo w </w>'	5
'lo w e r </w>'	2
'n e w est</w>'	6
'w i d est</w>'	3

('e', 's')	→	'es'
('es', 't')	→	'est'
('est', '</w>')	→	'est</w>'
('l', 'o')	→	'lo'

Byte pair encoding for word segmentation

bottom-up character merging

- iteratively replace most frequent symbol pairs ('A','B') with 'AB'
- apply on dictionary, not on full text (for efficiency)
- output vocabulary: character vocabulary + one symbol per merge

word	frequency
'low </w>'	5
'low e r </w>'	2
'n e w est</w>'	6
'w i d est</w>'	3

('e', 's')	→	'es'
('es', 't')	→	'est'
('est', '</w>')	→	'est</w>'
('l', 'o')	→	'lo'
('lo', 'w')	→	'low'
...		

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'l o w e s t </w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'l o w es t </w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'l o w est </w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'l o w est</w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'lo w est</w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Byte pair encoding for word segmentation

why BPE?

- trade-off between vocabulary size and text length
  (≈10% increase in text length for vocabulary size 60000)
- open-vocabulary:
  learned operations can be applied to unknown words
- alternative view: character-level model on compressed text

	('e', 's')	→	'es'
	('es', 't')	→	'est'
'low est</w>'	('est', '</w>')	→	'est</w>'
	('l', 'o')	→	'lo'
	('lo', 'w')	→	'low'

Evaluation: data and methods

data

- WMT 15 English→German and English→Russian

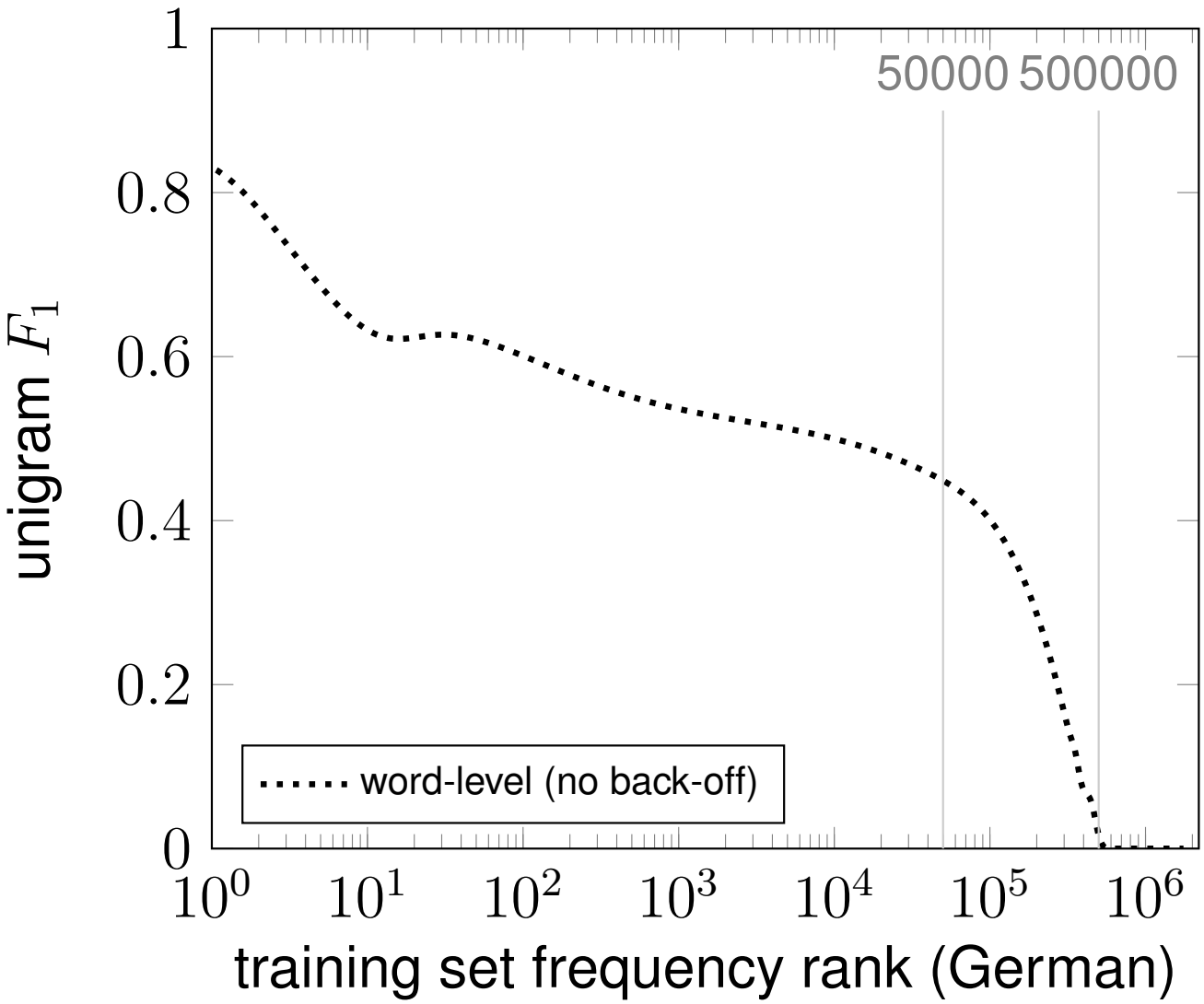
model

- attentional encoder–decoder neural network
- parameters and settings as in [Bahdanau et al, 2014]

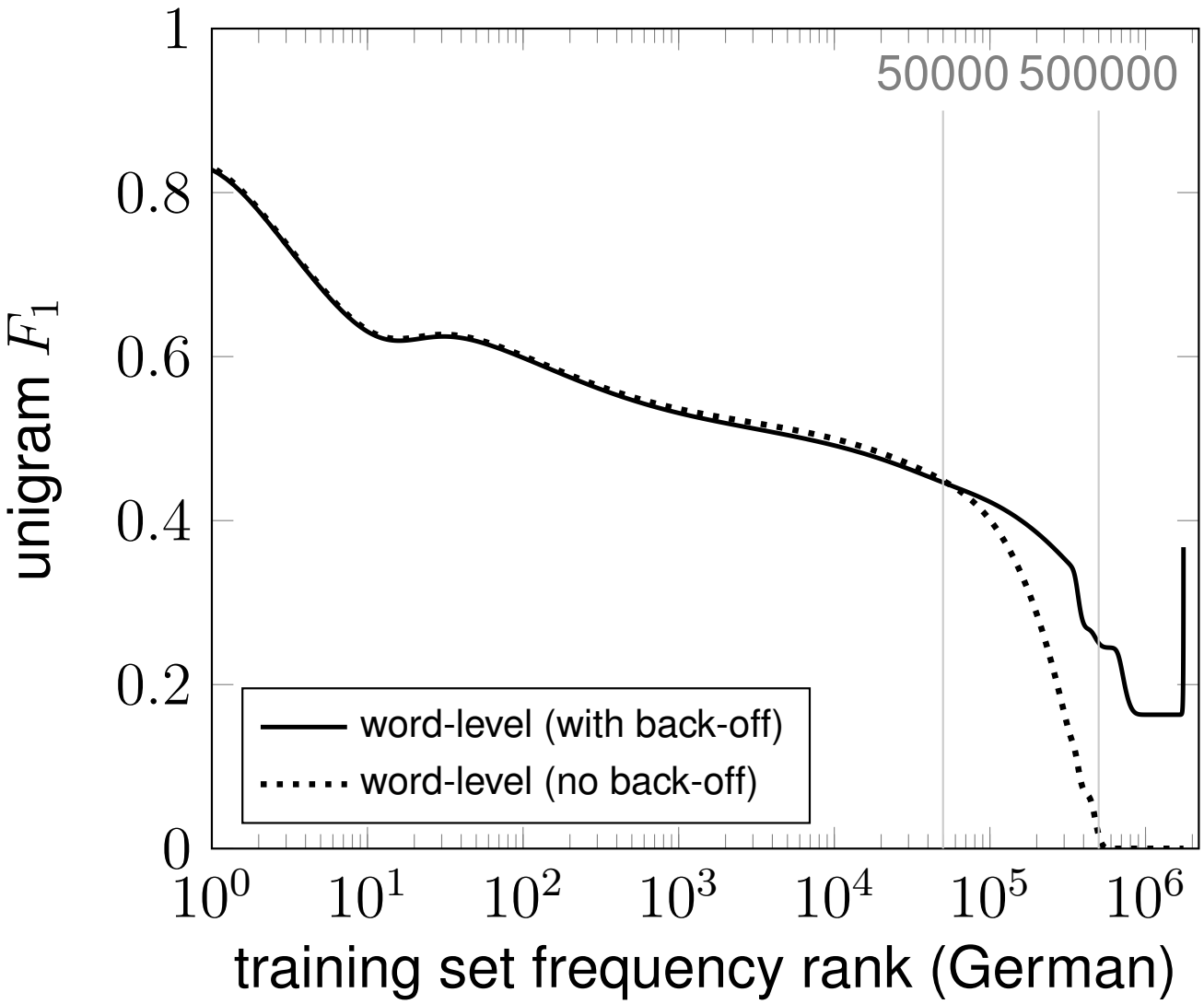
Translation quality

	system	BLEU
en-de	syntax-based (EMNLP15)	24.4
	word-level (with back-off)	22.0
	character bigrams	**22.8**
	BPE	**22.8**
	joint BPE (ensemble)	24.7
en-ru	phrase-based (WMT15)	24.3
	word-level (with back-off)	19.1
	character bigrams	**20.9**
	joint BPE	20.4
	joint BPE (ensemble)	24.1

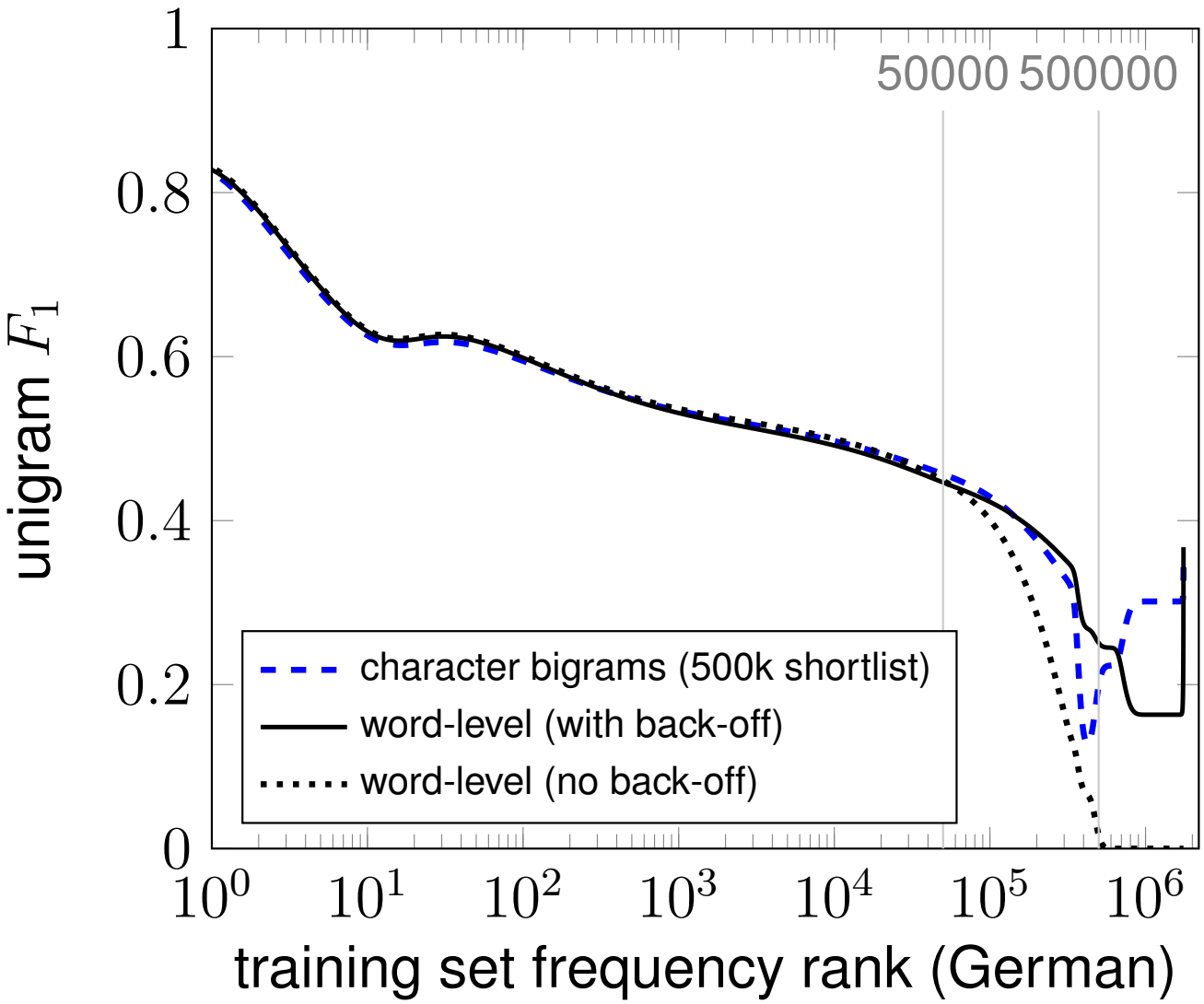
Unigram F_1 EN→DE



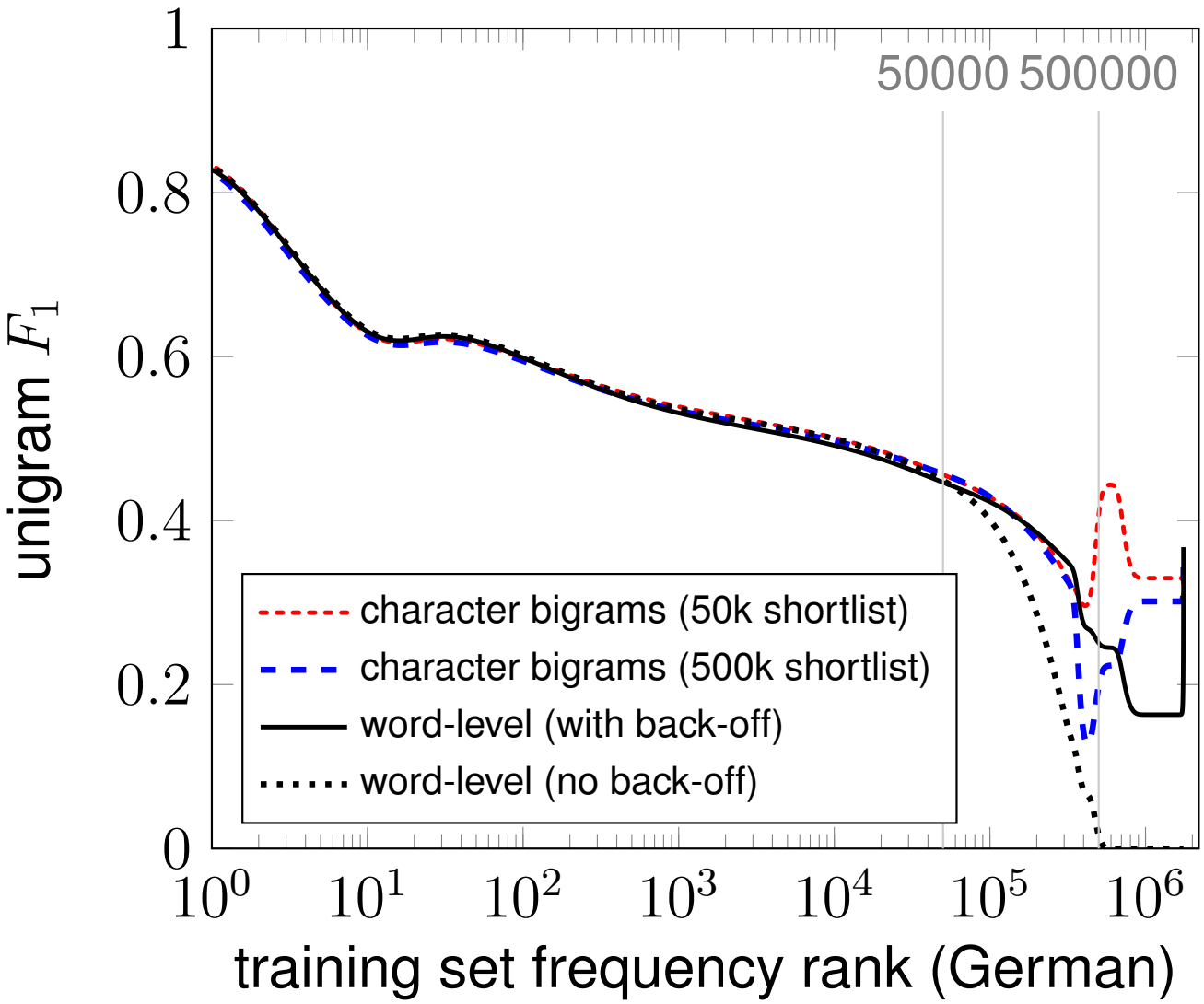
Unigram F_1 EN→DE



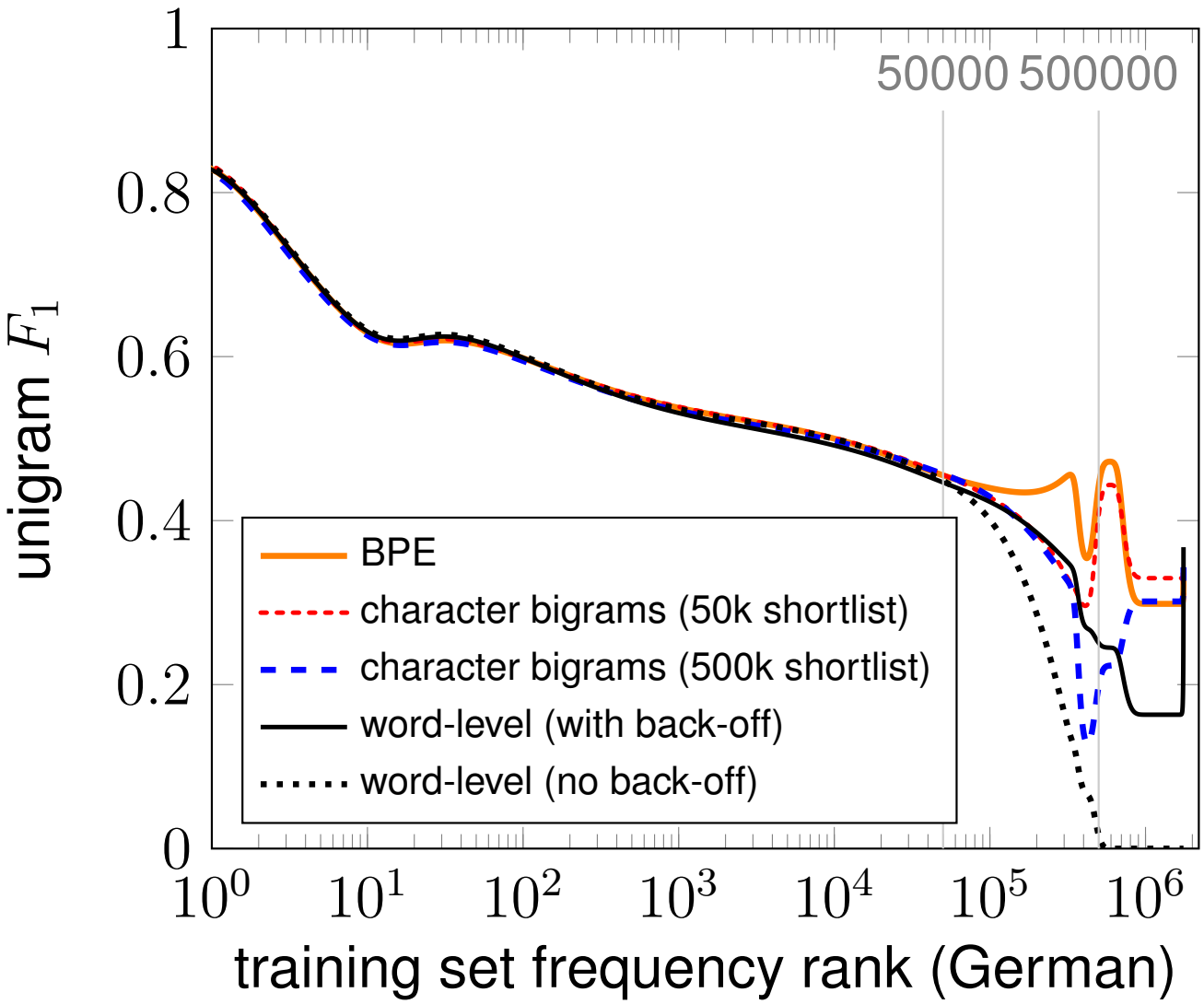
Unigram F_1 EN→DE



Unigram F_1 EN→DE



Unigram F_1 EN→DE



Examples

system	sentence
source	health research institutes
reference	Gesundheitsforschungsinstitute
word-level (with back-off)	Forschungsinstitute
character bigrams	Fo|rs|ch|un|gs|in|st|it|ut|io|ne|n
BPE	Gesundheits|forsch|ungsin|stitute
source	rakfisk
reference	ракфиска (rakfiska)
word-level (with back-off)	rakfisk → UNK → rakfisk
character bigrams	ra|kf|is|k → ра|кф|ис|к (ra|kf|is|k)
BPE	rak|f|isk → рак|ф|иска (rak|f|iska)

Aside: BPE for Phrase-based MT

WMT15 translation task

- Built phrase-based fi→en system with all OPUS data
- Scored well on BLEU, less well on human eval.

Aside: BPE for Phrase-based MT

WMT15 translation task

- Built phrase-based fi→en system with all OPUS data
- Scored well on BLEU, less well on human eval.

WMT16 translation task

- Applied BPE to source (Finnish) only
- Gained $+1.2$ BLEU on test set
- Ranked same as best online systems

Aside: BPE for Phrase-based MT (Examples)

source	yös Intian on sanottu olevan kiinnostunut puolustusyhteistyösopimuksesta Japanin kanssa.
base	India is also said to be interested in puolustusyhteistyösopimuksesta with Japan.
bpe	India is also said to be interested in defence cooperation agreement with Japan.
reference	India is also reportedly hoping for a deal on defence collaboration between the two nations.

source	Balotelli oli vielä kaukana huippuvireestään.
base	Balotelli was still far from huippuvireestään.
bpe	Baloo, Hotel was still far from the peak of its vitality.
reference	Balotelli is still far from his top tune.

Conclusion – Subwords in NMT

- translation of many rare/unknown words is transparent
- subword units enable open-vocabulary NMT
- improved translation quality for rare words
- model-independent: simple pre-/postprocessing

Monolingual Data in NMT

Why Monolingual Data for Phrase-based SMT?

- more training data ✓
- relax independence assumptions ✓
- more appropriate training data (domain adaptation) ✓

Why Monolingual Data for NMT?

- more training data ✓
- relax independence assumptions ✗
- more appropriate training data (domain adaptation) ✓

Monolingual Data in NMT

encoder-decoder already conditions on previous target words

$\Downarrow$

no architecture change required to learn from monolingual data

Monolingual Training Instances

Output prediction

- $p(y_i)$ is a function of hidden state s_i, previous output y_{i-1}, and source context vector c_i
- only difference to monolingual RNN: c_i

Problem

we have no source context c_i for monolingual training instances

Monolingual Training Instances

Output prediction

- $p(y_i)$ is a function of hidden state s_i, previous output y_{i-1}, and source context vector c_i
- only difference to monolingual RNN: c_i

Problem

we have no source context c_i for monolingual training instances

Solutions

- two methods to deal with missing source context:
  - empty/dummy source context c_i
    $\rightarrow$ danger of unlearning conditioning on source
  - produce synthetic source sentence via back-translation
    $\rightarrow$ get approximation of c_i

Monolingual Training Instances

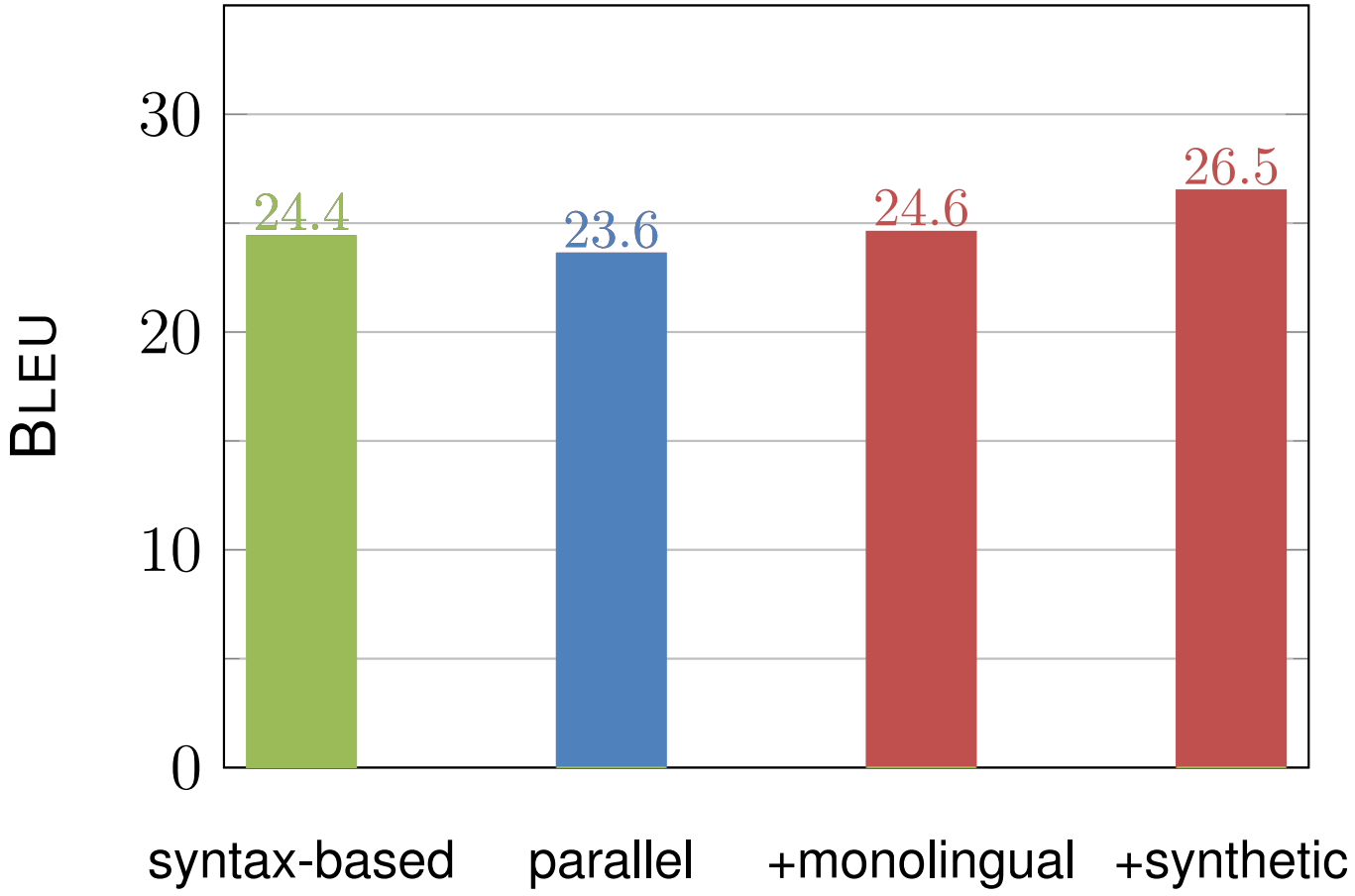
Dummy source

- 1-1 mix of parallel and monolingual training instances
- randomly sample from monolingual data each epoch
- freeze encoder/attention layers for monolingual training instances

Synthetic source

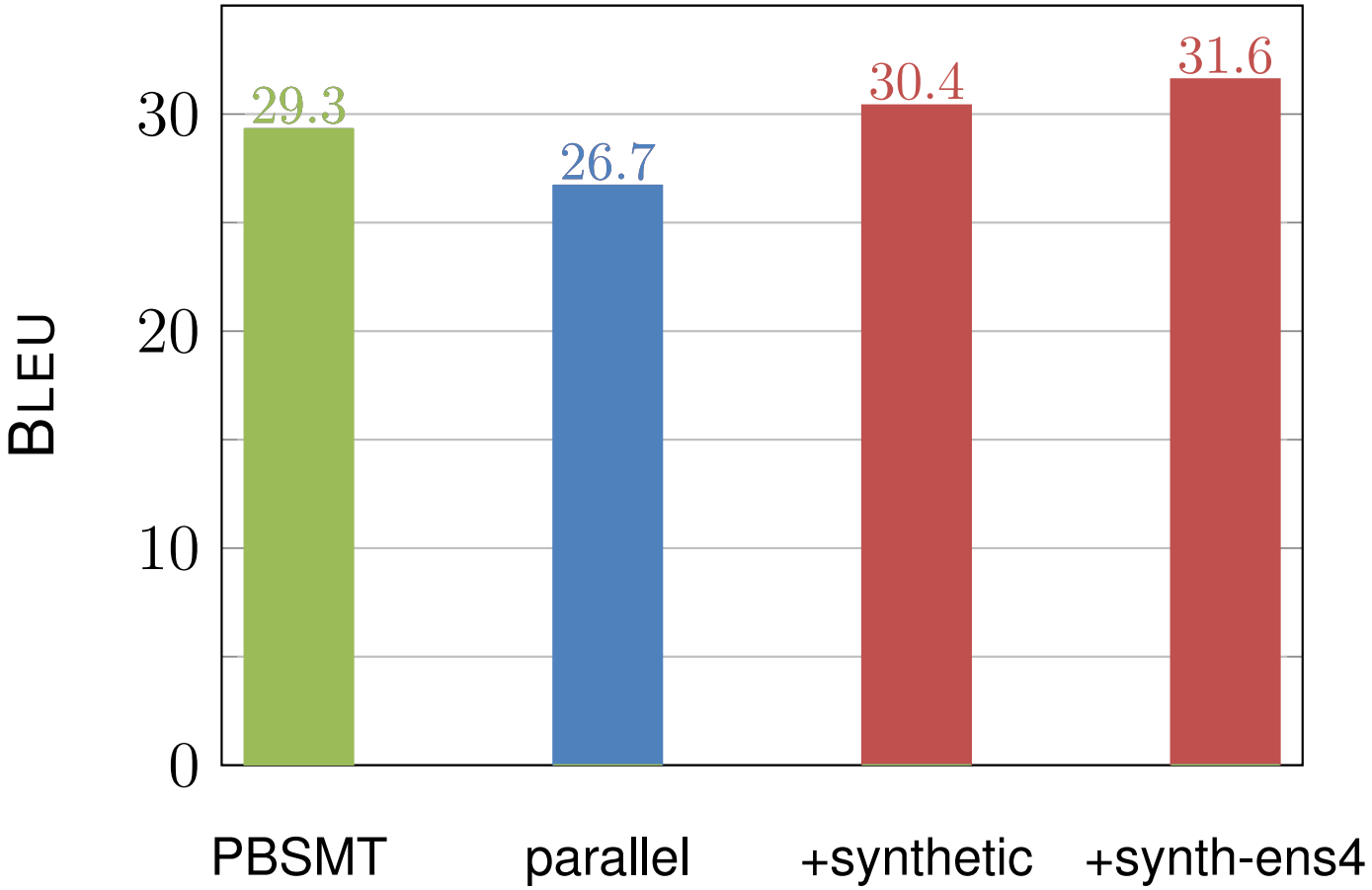
- 1-1 mix of parallel and monolingual training instances
- randomly sample from back-translated data
- training does not distinguish between real and synthetic parallel data

Evaluation: WMT 15 English→German



(NMT systems are ensemble of 4)

Evaluation: WMT 15 German→English



Why is monolingual data helpful?

- Domain adaptation effect
- Reduces over-fitting
- Improves fluency

(See our ACL paper for more analysis.)

Why Linguistic Features?

disambiguate words by POS

English	German
$close_{verb}$	schließen
$close_{adj}$	nah
$close_{noun}$	Ende

source	*We thought a win like this might be* $close_{adj}$.
reference	*Wir dachten, dass ein solcher Sieg nah sein könnte.*
baseline NMT	**Wir dachten, ein Sieg wie dieser könnte schließen.*

Why Linguistic Features?

better generalization; combat data sparsity

word form
liegen (lie)
liegst (lie)
lag (lay)
läge (lay)

Why Linguistic Features?

better generalization; combat data sparsity

word form	lemma	morph. features
liegen (lie)	liegen (lie)	(3.p.pl. present)
liegst (lie)	liegen (lie)	(2.p.sg. present)
lag (lay)	liegen (lie)	(3.p.sg. past)
läge (lay)	liegen (lie)	(3.p.sg. subjunctive II)

Neural Machine Translation: Multiple Input Features

Use separate embeddings for each feature, then concatenate

baseline: only word feature

$$E(close) = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.3 \\ 0.1 \end{bmatrix}$$

$|F|$ input features

$$E_1(close) = \begin{bmatrix} 0.4 \\ 0.1 \\ 0.2 \end{bmatrix} \qquad E_2(adj) = \begin{bmatrix} 0.1 \end{bmatrix} \qquad E_1(close) \parallel E_2(adj) = \begin{bmatrix} 0.4 \\ 0.1 \\ 0.2 \\ 0.1 \end{bmatrix}$$

Linguistic Features

- lemmas
- morphological features
- POS tags
- dependency labels
- BPE tags

Evaluation

Data

- WMT16 training/test data
- English↔German and English→Romanian

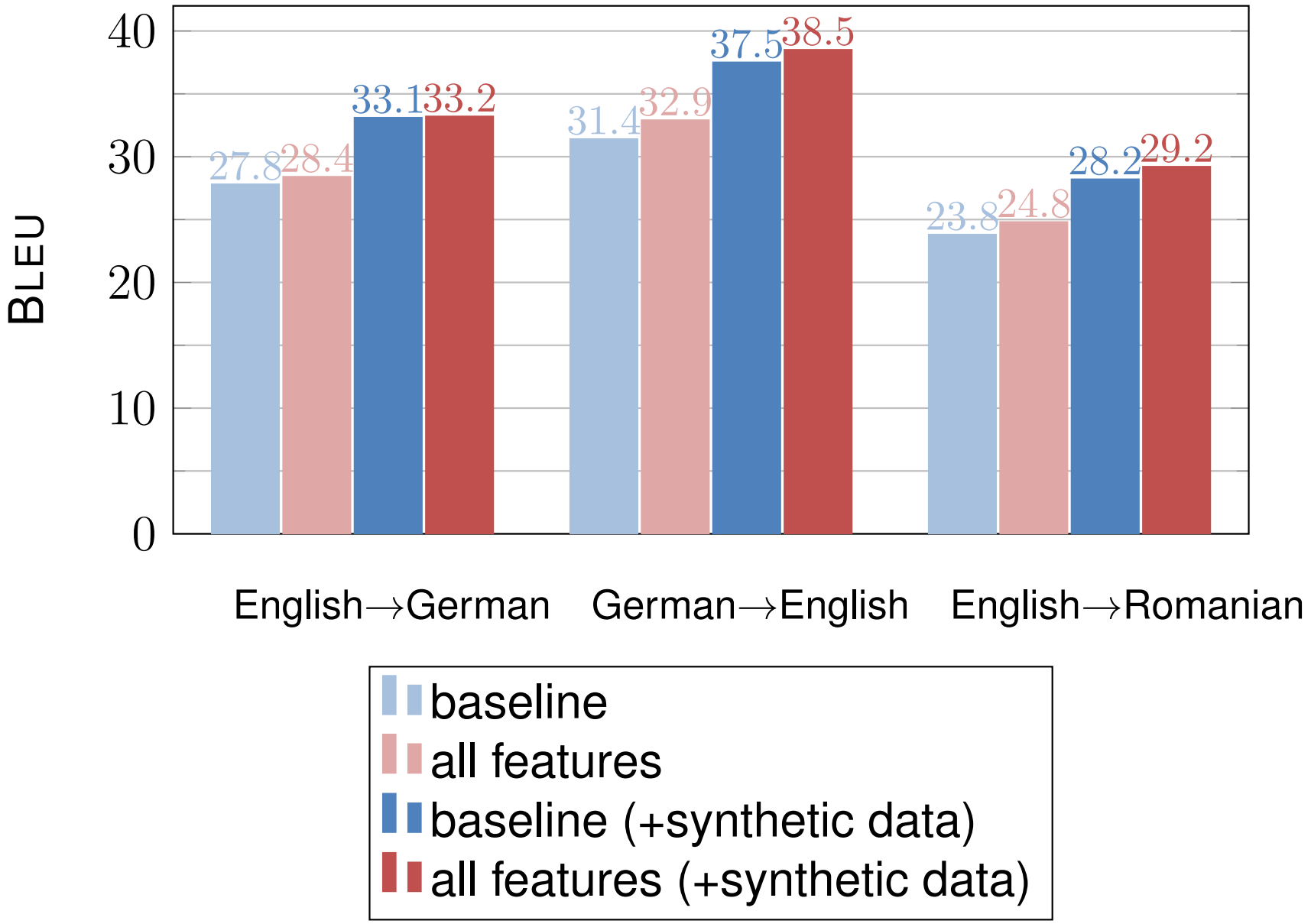
Tools

NMT tool: Nematus (fork of dl4mt-tutorial)

German annotation: ParZu

English annotation: Stanford CoreNLP

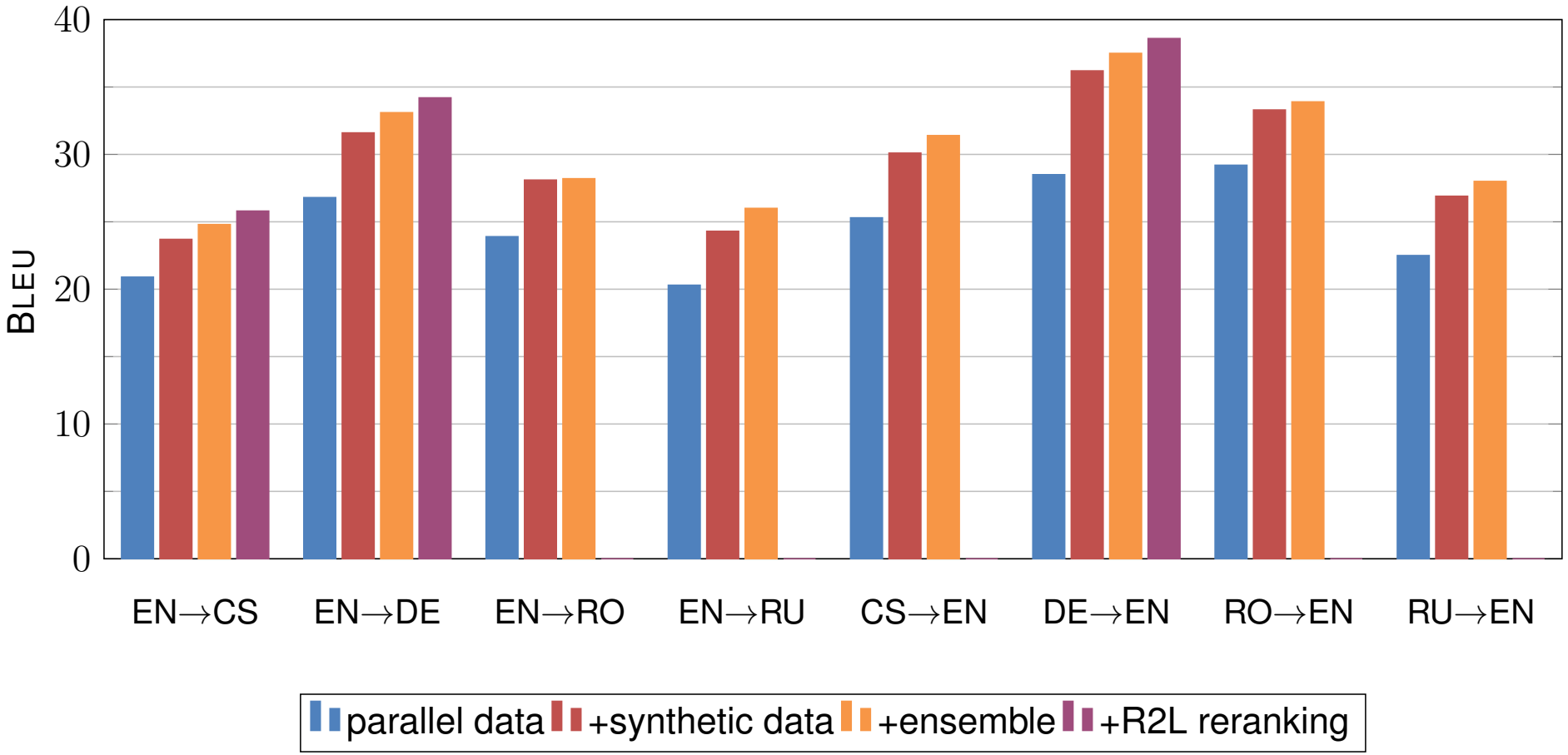
Results: BLEU ↑



Linguistic features: Summary

- Linguistic input features improve neural MT over strong baselines
- Many possibilities for source-side features
  - Domain/topic adaptation
  - Semantically-motivated tags
  - ...
- Target side features (cf multi-task learning)

Putting it all together: WMT16 Results

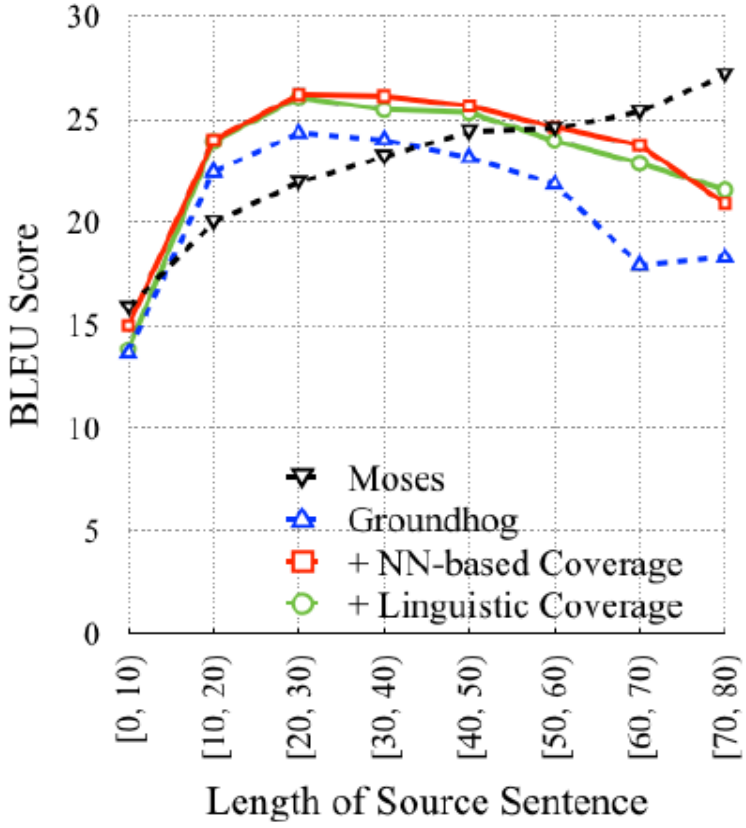


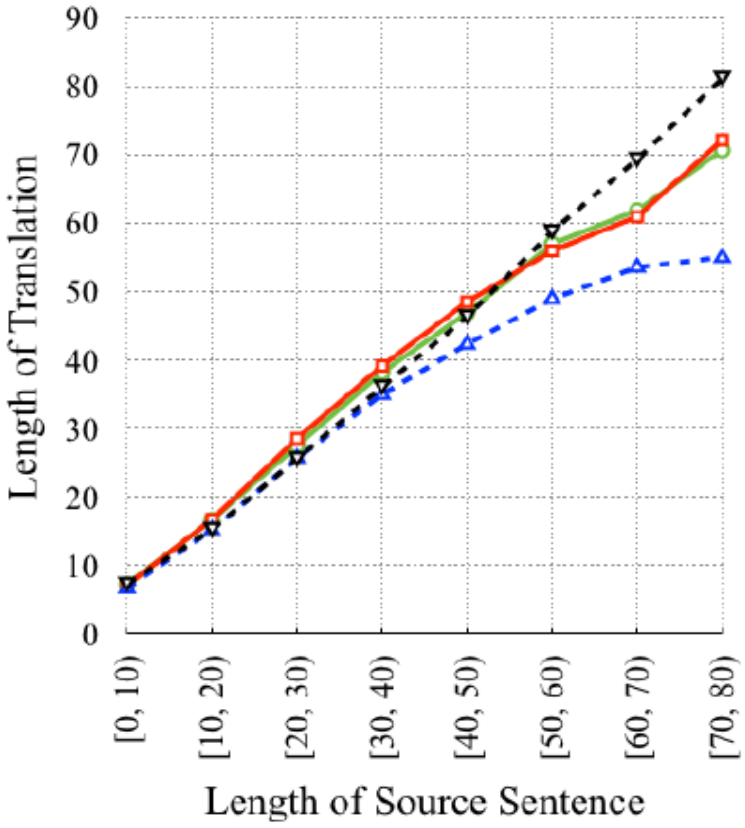
Also New in NMT from ACL

- Coverage mechanism [Tu et al.]
- Minimum risk training [Shen et al.]
- Character-based models [Chung et al., Luong and Manning, Costa-jussá and Fonollosa]
- Copying input to output [Gu et al., Gulçehre et al.]
- Incorporating syntax [Eriguchi et al.]
- Using monolingual data [Cheng et al.]

Current Problems in NMT (and Opportunities)

Length bias





[Tu et al., ACL 2016]

Fluency vs. Adequacy

		Adequacy		Fluency	
ru-en	ONLINE-G	**0.115**	74.2	0.100	69.9
	AMU-UEDIN	**0.103**	73.3	0.178	72.2
	ONLINE-B	0.083	72.8	0.030	67.8
	NRC	0.060	72.7	0.092	69.9
	PROMT-RULE-BASED	0.044	72.1	−0.102	63.8
	UEDIN-NMT	0.011	71.1	**0.245**	74.3
	ONLINE-A	−0.007	70.8	0.020	66.7
	AFRL-MITLL-PHRASE	−0.040	70.1	0.047	68.4
	AFRL-MITLL-CONTRAST	−0.071	69.3	−0.020	66.5
	ONLINE-F	−0.322	61.8	−0.472	54.7

[Bojar et al. WMT 2016]

Other Problems in NMT

- Lack of interpretability
- Choice of training regime, convergence, variability
- Long training times
- Dealing with domain shift
- No alignment between source and output

Dealing with Morphological Complexity

- Good progress so far
  - WMT16 evaluation shows good results in Czech, German . . .
  - Analysis of PBMT vs NMT shows 20% less morph errors [Bentivogli et al, EMNLP 2016]
- Opportunities for Further Progress
  - Lots of flexibility on representations
  - Output conditioned on whole source
  - Source/target linguistic analyses easily added
  - Consider interaction of attention and representation

Practical Considerations

Hardware

- RAM/disk/cpu requirements much more modest than “traditional SMT”
- GPU memory important – at least 12G for our expts.

Training times (WMT16)

	en-cs	en-de
Sentences	52M	7.8M
Training	21d	12d
Fine-tuning	8d	–

- For en-de the synthetic was mixed with parallel
- Convergence monitored by perplexity and BLEU on heldout
- Save every 30,000 steps, ensemble of last 4 savepoints

Our Tools

Nematus (`https://github.com/rsennrich/nematus`)

- Training and decoding of NMT models
- Based on dl4mt by Kyunghyun Cho and others
- Slightly different model than Groundhog/Bahdanau paper
- Adds ensemble, input features, nbest, dropout, ...
- All Python, theano backend

AmuNMT (`https://github.com/emjotde/amunmt`)

- Fast C++ decoder for nematus models
- Directly interfaces CUDA libraries
- Also enables fast CPU decoding

subword-nmt (`https://github.com/rsennrich/subword-nmt`)

- BPE learn and apply ... and chrF metric

WMT16 scripts and models

- `https://github.com/rsennrich/wmt16-scripts`
- `http://data.statmt.org/rsennrich/wmt16_systems/`

NMT Toolkits

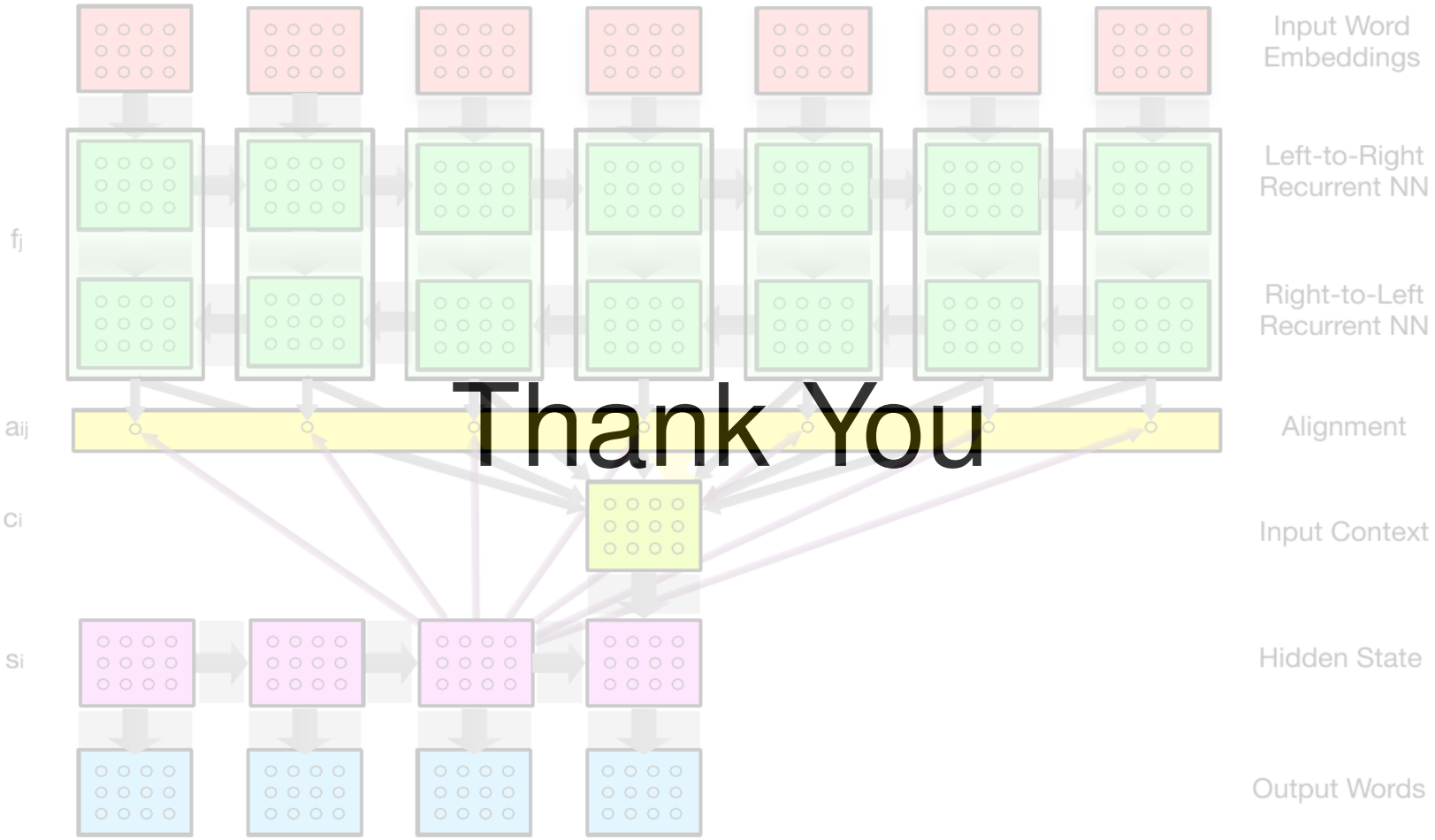
Software	Who	Backend	Comments
AmuNMT decoder	Marcin	Custom	compatible with Nematus models
dl4mt-tutorial	Cho, Orhan	Theano	
lamtram	Graham	CNN	
mantis	Trevor	CNN	
Nematus	Rico, Cho, Orhan	Theano	
nmt.hybrid	Thang	Matlab	also word-based
Paddle	Baidu	Paddle	demo
Keras seq2seq	Fariz	Keras	
TensorFlow seq2seq	Google	TensorFlow	MT demo
seq2seq-attn	Yoon	Torch	
textsum	Google	TensorFlow	text sum demo

https://github.com/jonsafari/nmt-list

Acknowledgments

Some of the research presented here was conducted in cooperation with Samsung Electronics Polska sp. z o.o. - Samsung R&D Institute Poland.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreements 645452 (QT21) and 644402 (HimL).


Input Word Embeddings
Left-to-Right Recurrent NN
Right-to-Left Recurrent NN
Alignment
Input Context
Hidden State
Output Words
fj
aij
ci
si

Thank You

Prior work

back-off models [Jean et al. (2015), Luong et al. (2014)]

- compounds: hard to model 1-to-many relationships
- names: if alphabets differ, we need transliteration

Huffman encoding of rare words [Chitnis and de Nero (2015)]

no generalization to unseen words

subword units in PBSMT

different challenges:

- limited vocabulary in NMT
- strong independence assumptions in PBSMT

Rare words often have transparent translation

100 rare German words (not among top 50000)

56	compounds	Stallhygiene	stable hygiene
21	names	Reuss	Reuss
6	cognates/loanwords	emanzipieren	emancipate
5	transparent affixes	süßlich	sweetish
1	number	2346	2346
1	technical term	ingres.utf8	ingres.utf8
10	other	Vermietern	landlords

hypothesis

we can translate rare/unseen words on subword level

Core idea: transparent translations

transparent translations

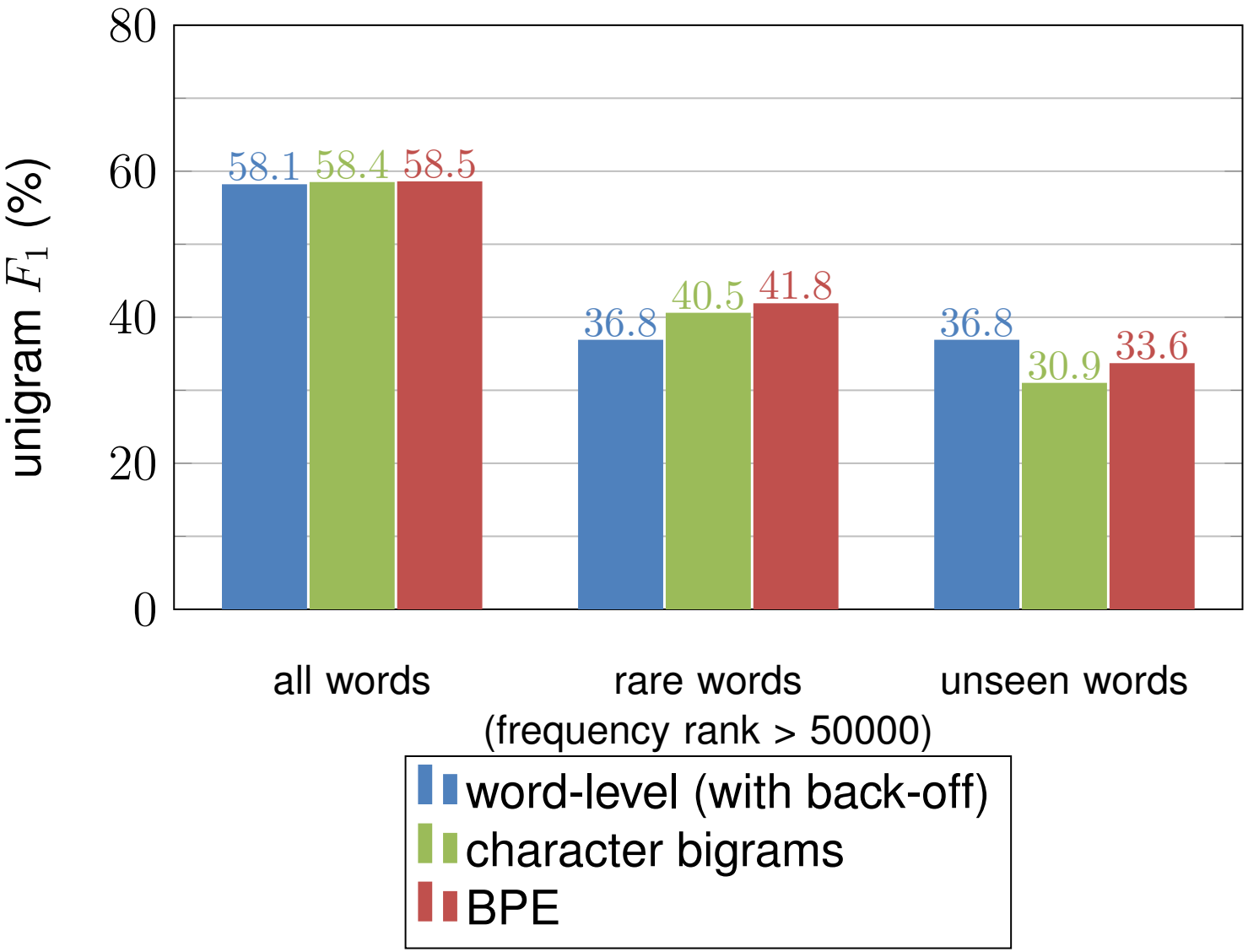
- some translations are semantically/phonologically transparent
- morphologically complex words (e.g. compounds):
  - solar system (English)
  - Sonnen|system (German)
  - Nap|rendszer (Hungarian)
- named entities:
  - Obama(English; German)
  - Обама (Russian)
  - オバマ (o-ba-ma) (Japanese)
- cognates and loanwords:
  - claustrophobia(English)
  - Klaustrophobie(German)
  - Клаустрофобия (Russian)

BPE: increasing segmentation consistency

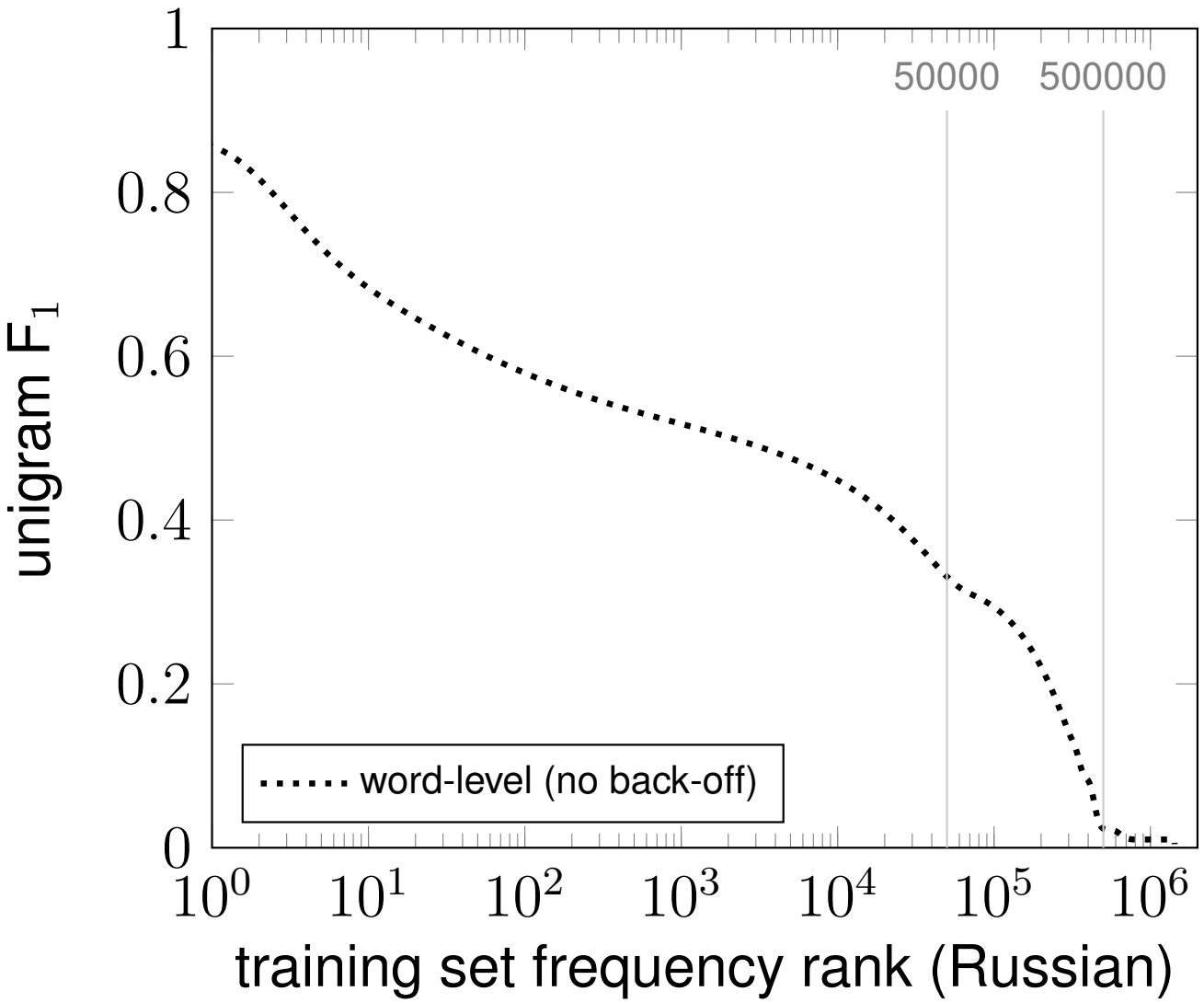
joint BPE

- being able to align subword units 1-to-1 is helpful
- to increase consistency, we concatenate text in both languages to learn BPE
- for Cyrillic, we use ISO 9 Romanization
- example:
  - BPE: Mirz|ayeva → Мир|за|ева (Mir|za|eva)
  - joint BPE: Mir|za|yeva → Мир|за|ева (Mir|za|eva)

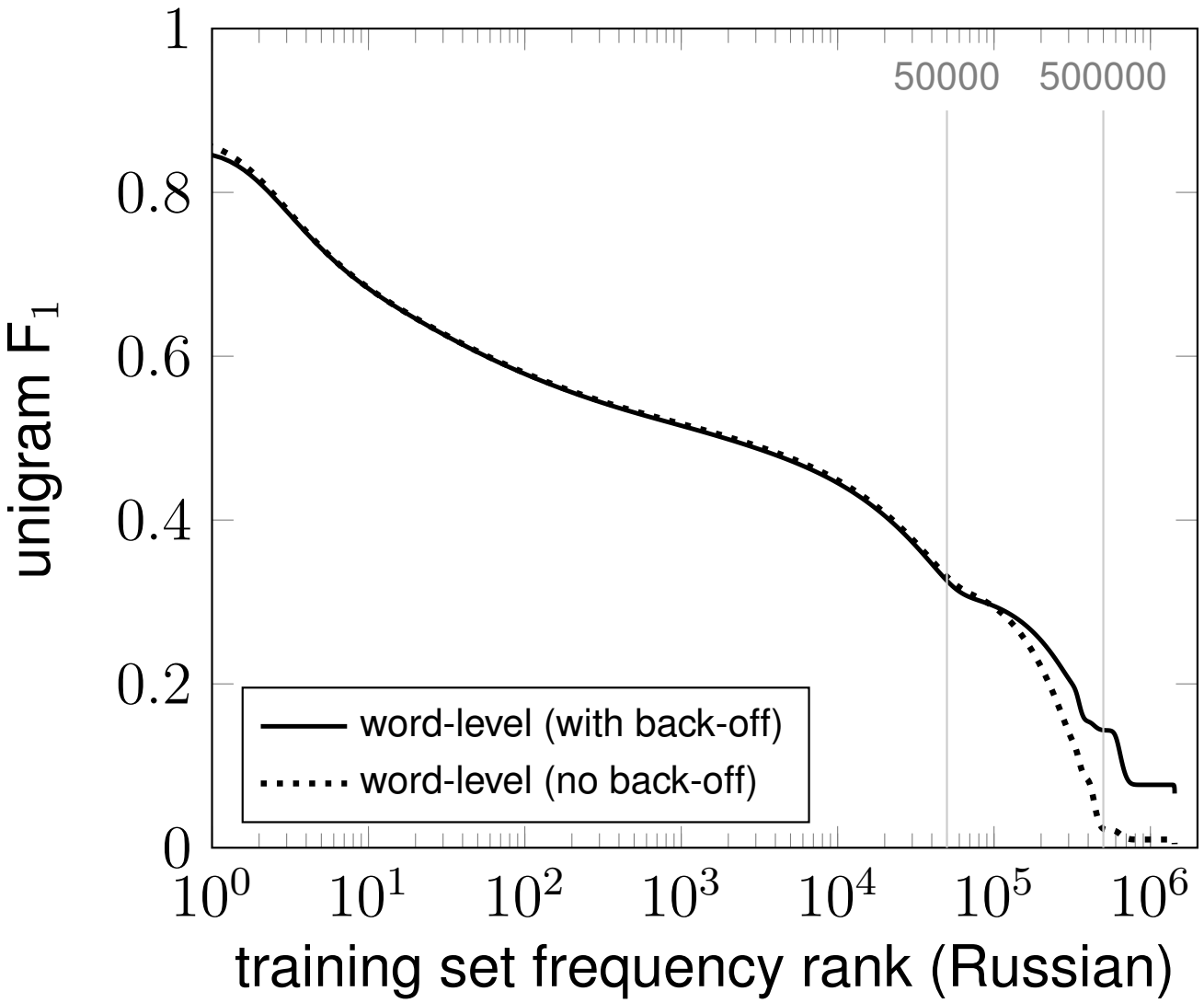
Unigram F_1 EN→DE



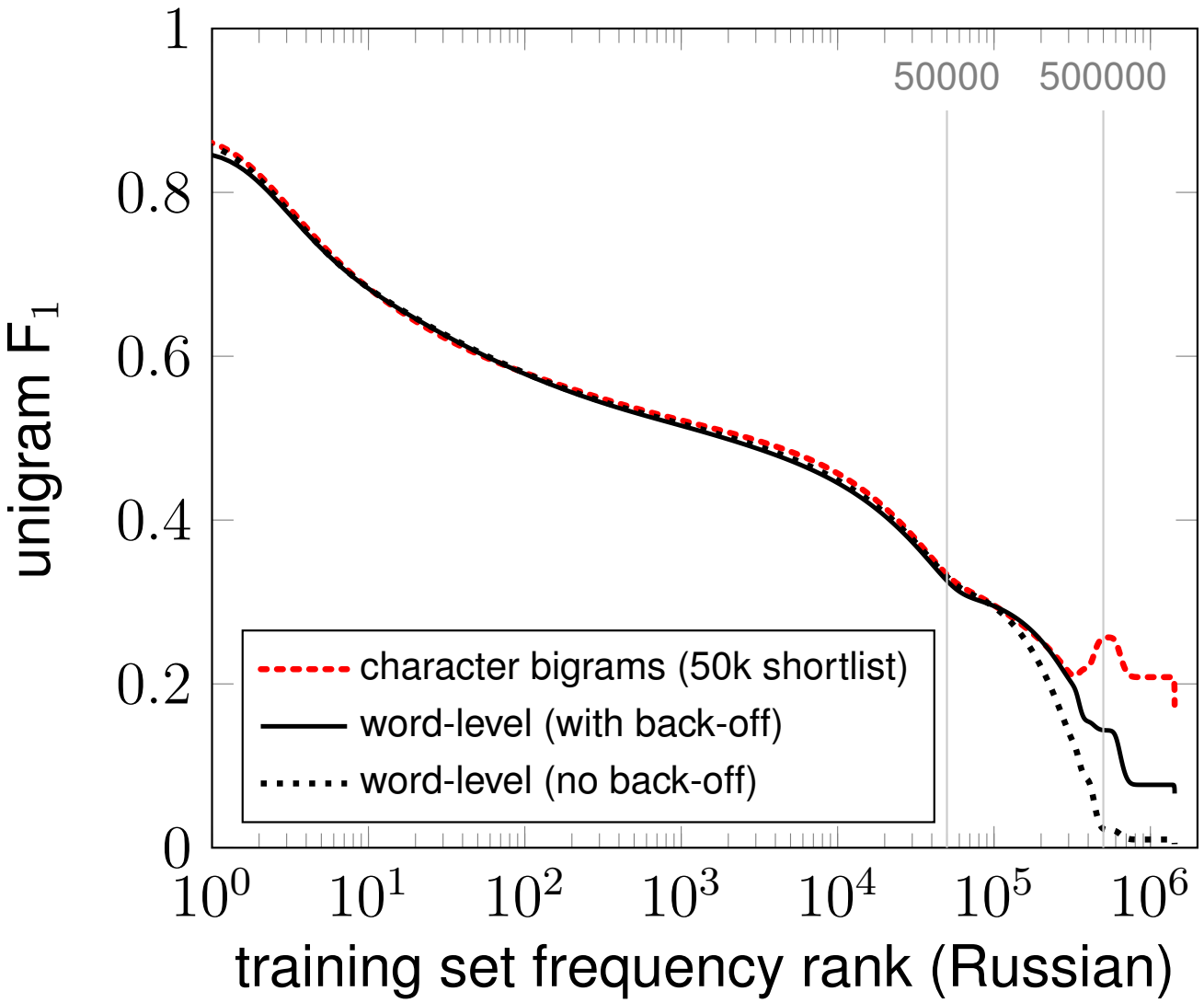
Unigram accuracy EN→RU



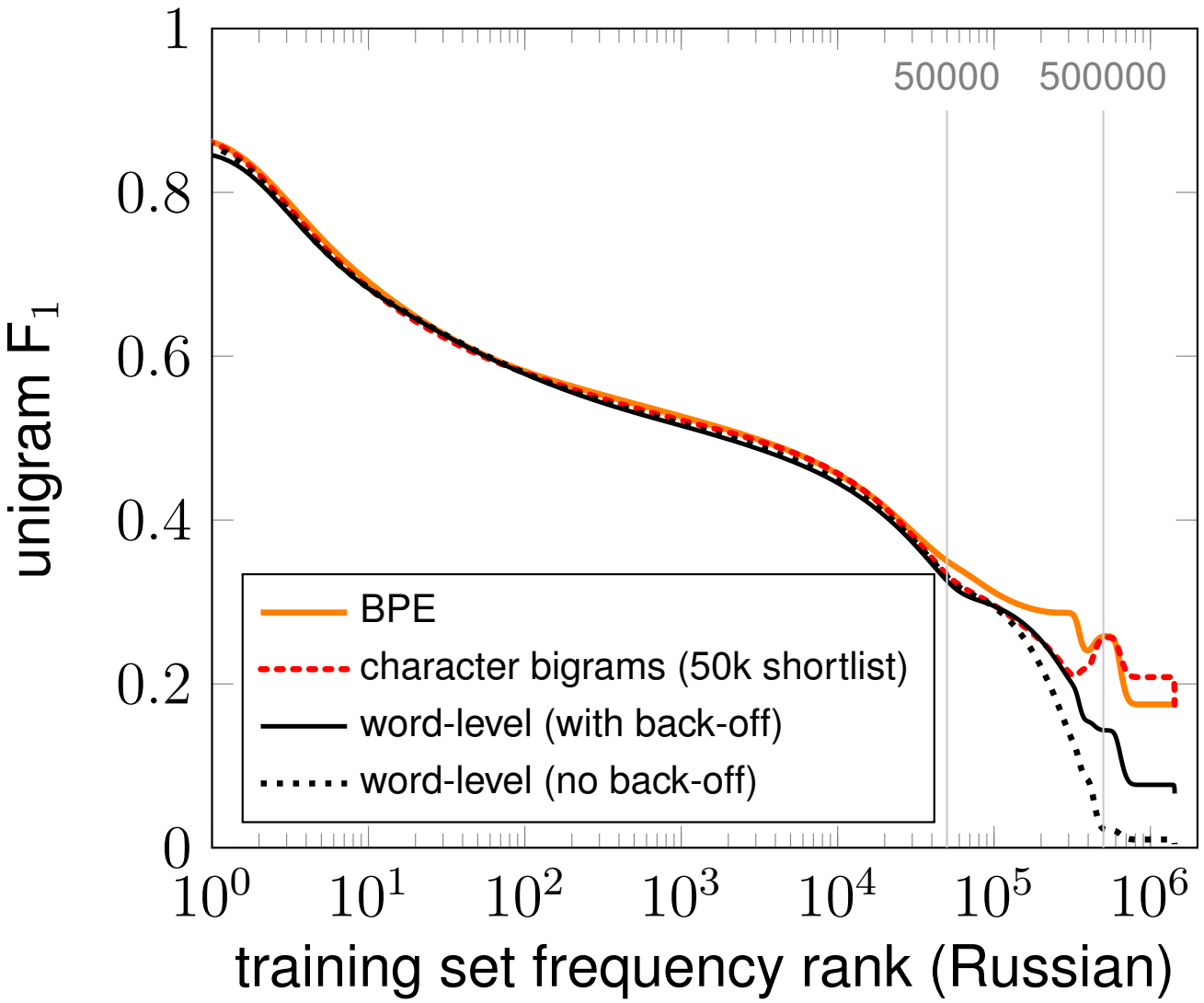
Unigram accuracy EN→RU



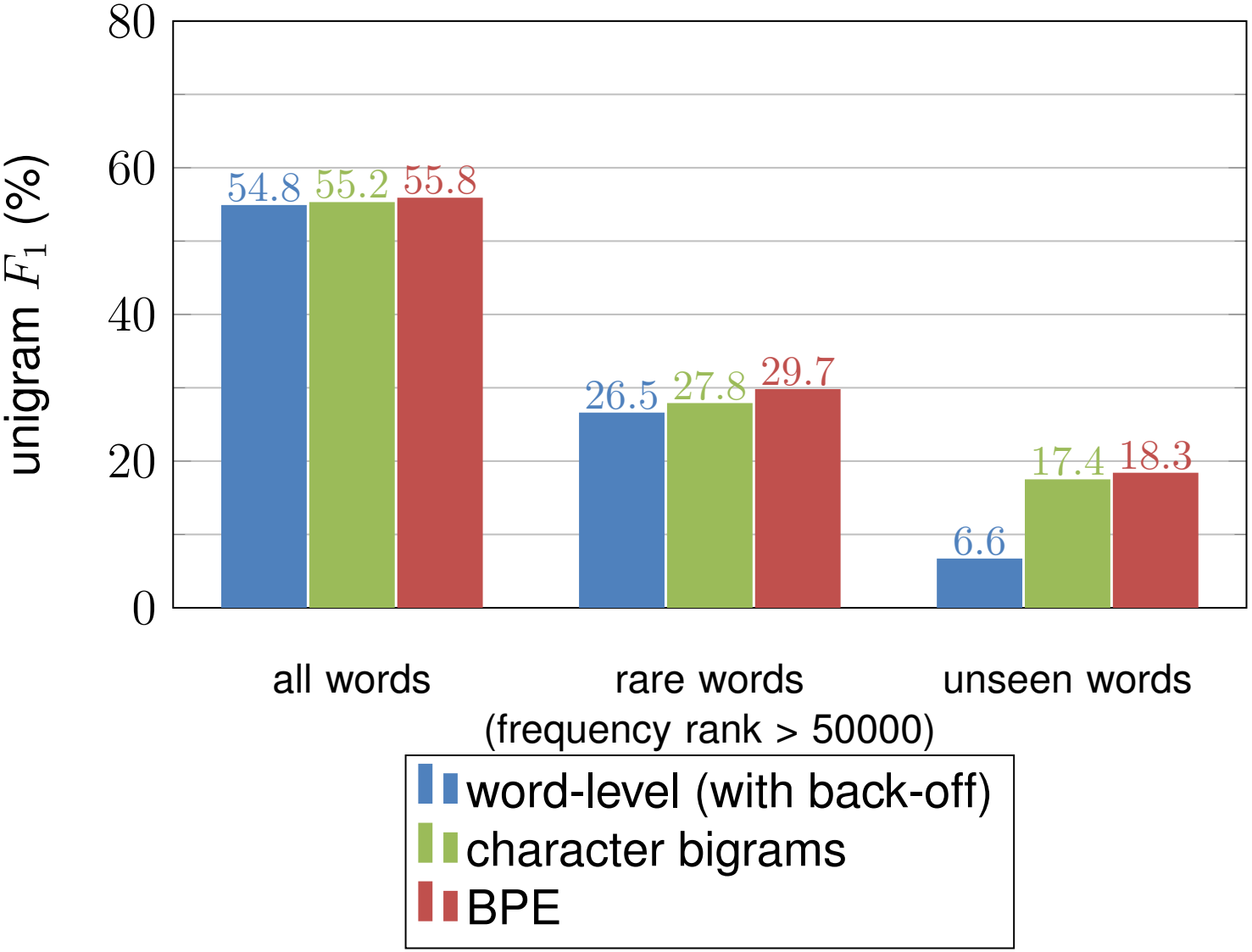
Unigram accuracy EN→RU



Unigram accuracy EN→RU



Unigram accuracy EN→RU



Analysis: Domain Adaptation with Monolingual Data

BLEU	
WMT (in-domain)	IWSLT (out-of-domain)
+2.9	+1.2

Table: Gains on English→German from adding synthetic News Crawl data.

→ domain adaptation explains improvement partially

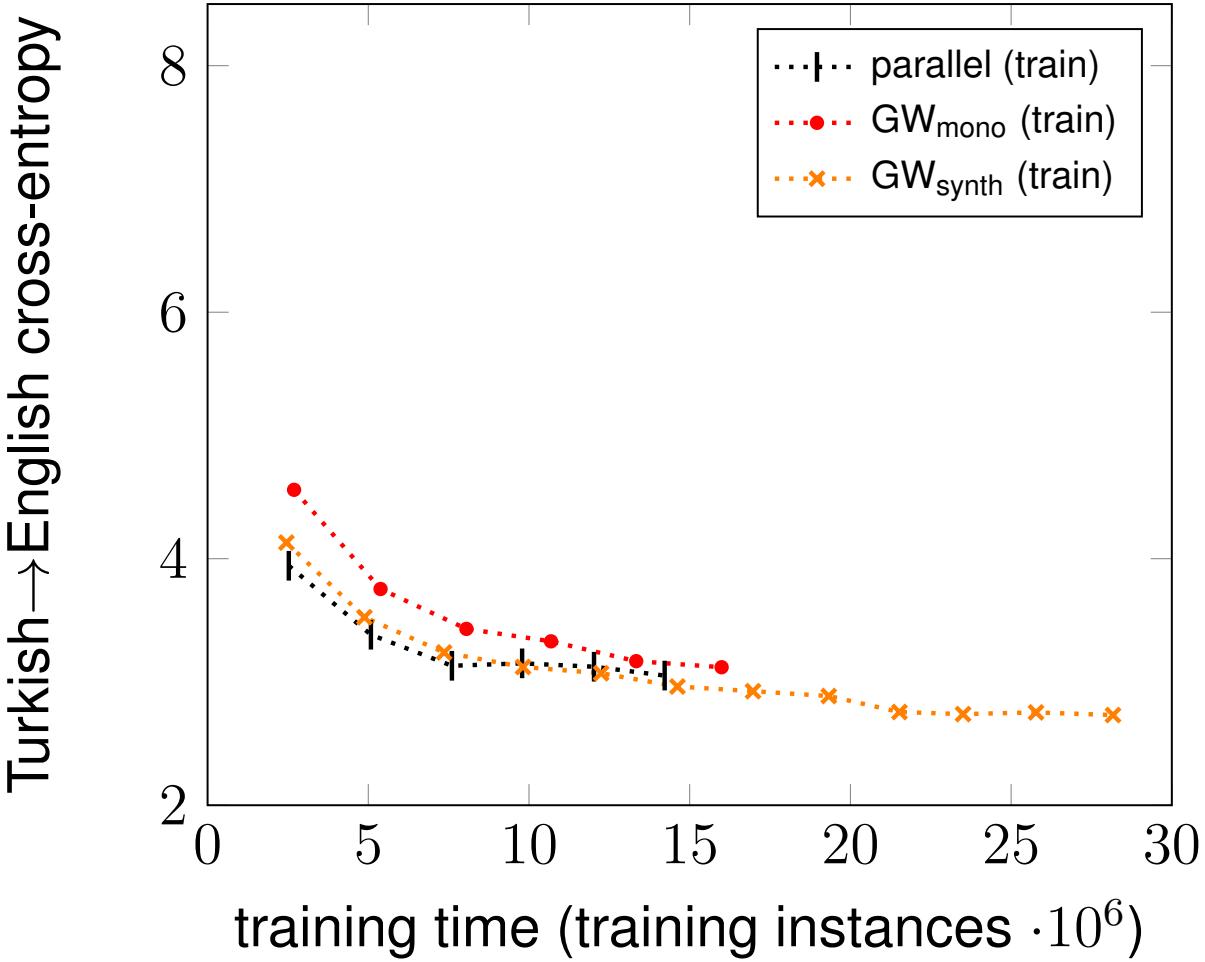
Analysis: Domain Adaptation with Monolingual Data

	system	BLEU IWSLT tst2015
0	phrase-based	25.5
1	NMT out-of-domain	25.5
2	1+in-domain$_{synth}$	26.7
3	1+in-domain$_{parallel}$	**28.4**

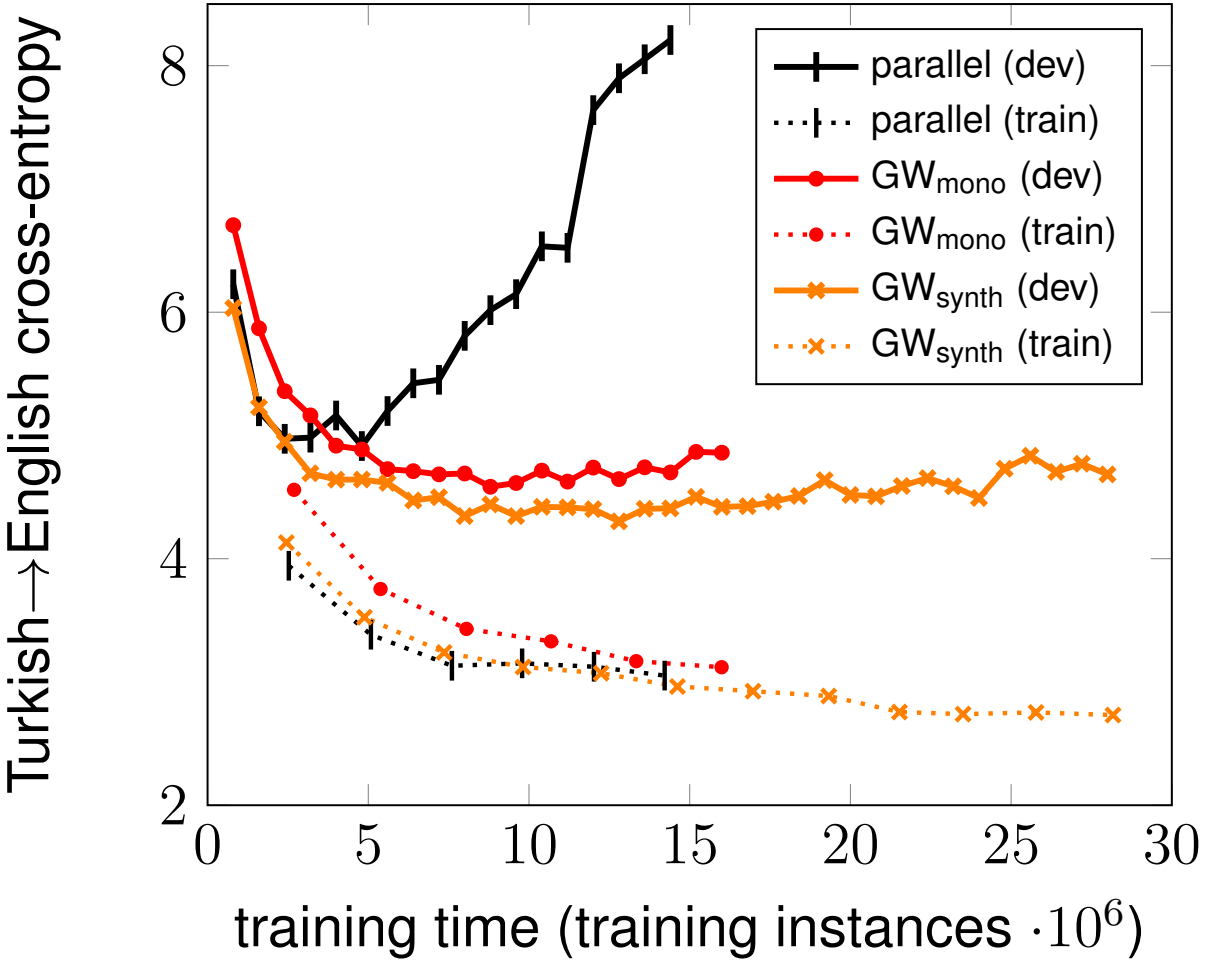
Domain adaptation by fine-tuning on in-domain data

- out-of-domain: WMT (8 million sentence pairs)
- in-domain: TED (200k sentence pairs)

Analysis: Overfitting



Analysis: Overfitting



The Effect of Back-Translation Quality

Use different systems for German→English translation

- baseline system (beam size 1)
- baseline system (beam size 12)
- system that was trained with synthetic data

back-translation	BLEU	
	(DE→EN)	EN→DE
none	-	23.6
parallel (beam 1)	(22.3)	26.0
parallel (beam 12)	(25.0)	26.5
synthetic (beam 12)	(28.3)	26.6

Comparsion to PBSMT

synthetic parallel data

- rationale in PBSMT: phrase-table domain adaptation [**?**, **?**, **?**]
- rationale in NMT: allow training on monolingual data

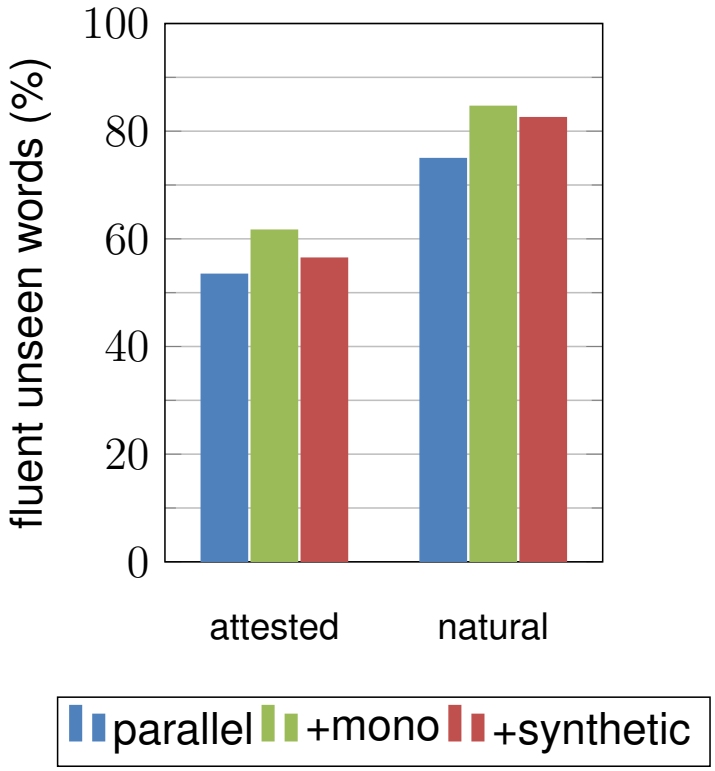
system	BLEU	
	WMT (in-domain)	IWSLT (out-of-domain)
PBSMT gain	+0.7	+0.1
NMT gain	+2.9	+1.2

Table: Gains on English→German from adding synthetic News Crawl data.

Analysis: Fluency

word-level fluency

- models create unseen words via subword units
  civil rights protections → Bürger|rechts|schutzes
- how many of them are fluent?
  - attested in monolingual corpus
  - natural according to native speaker
- example of disfluency:
  asbestos mats → *As|best|atten
  (correct: As|best|matten)

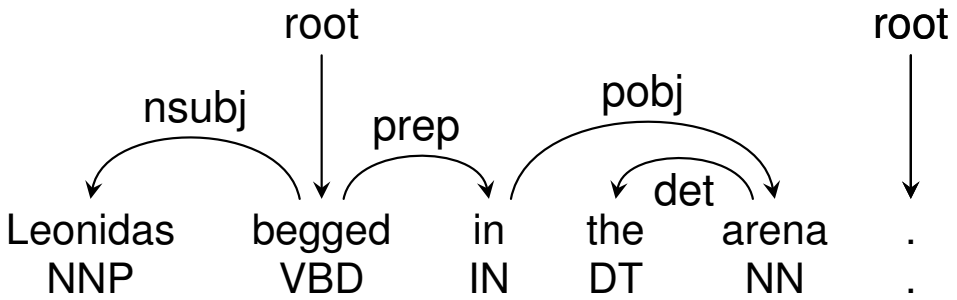


Why Linguistic Features?

guide reordering with syntactic information

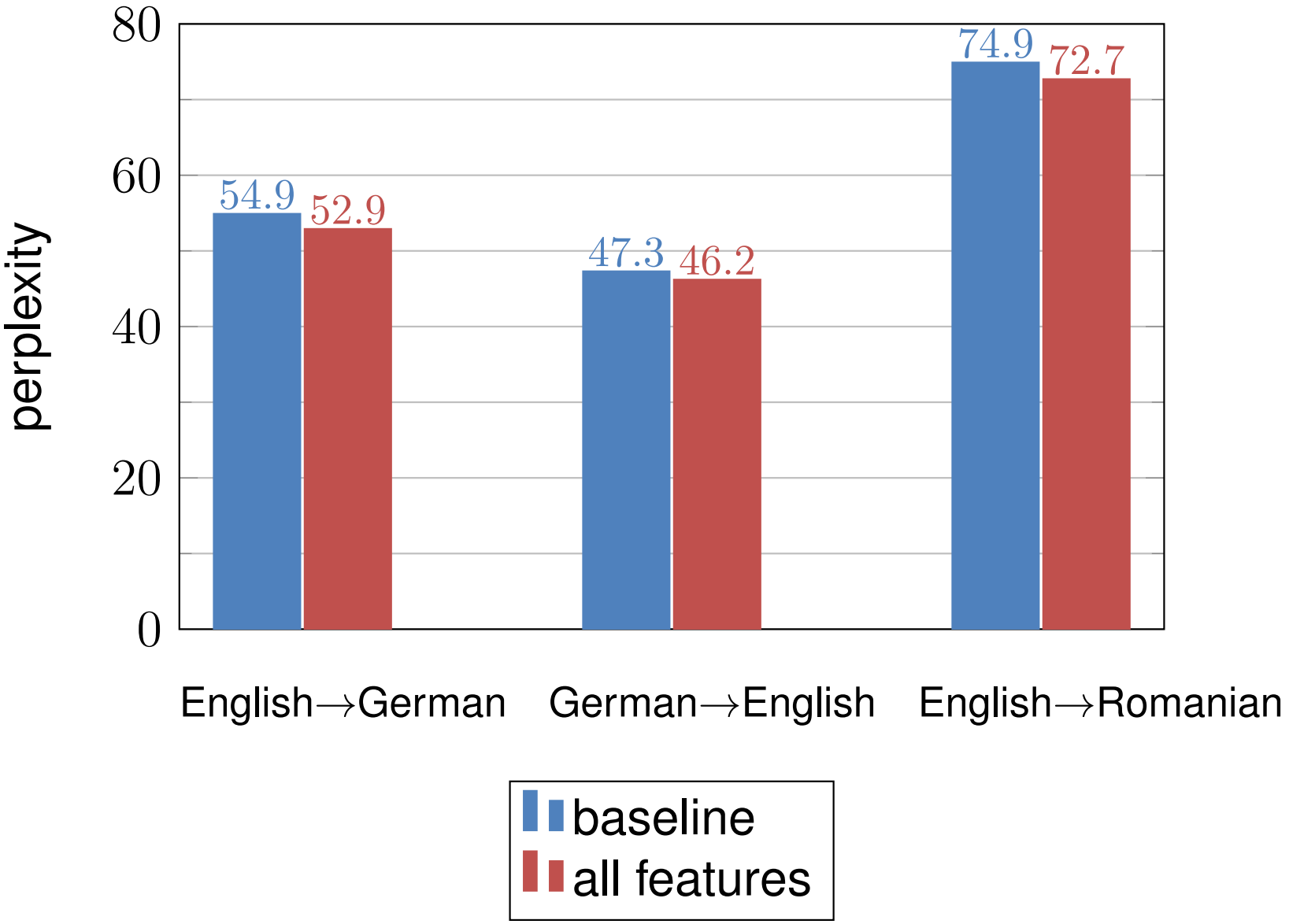
source	*Gefährlich*$_{pred}$ *ist die Route*$_{subj}$ *aber dennoch.*
reference	*However the route is dangerous.*
baseline NMT	**Dangerous is the route, however.*

Linguistic Features: Example

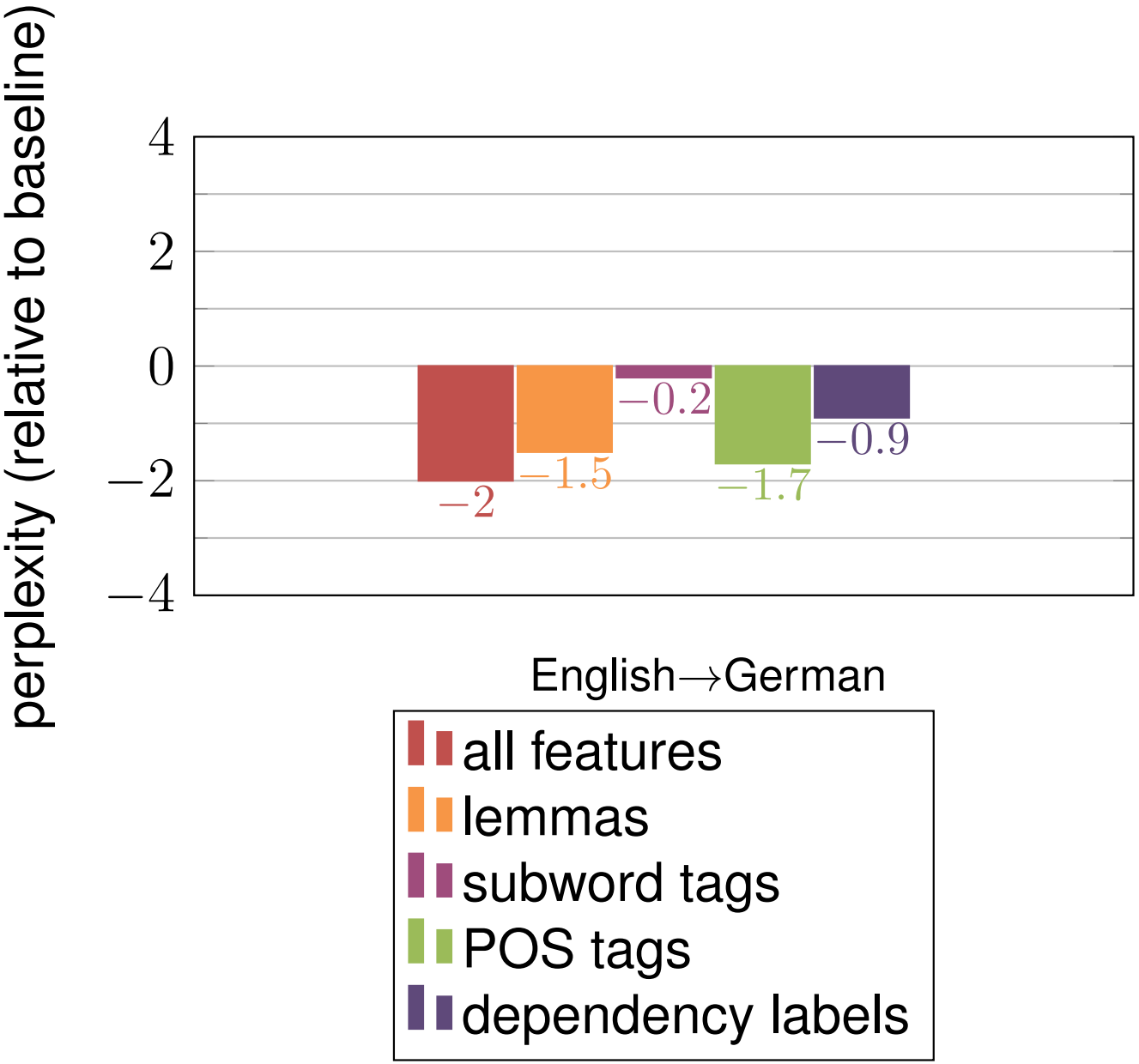


words	Le:	oni:	das	beg:	ged	in	the	arena	.
lemmas	Leonidas	Leonidas	Leonidas	beg	beg	in	the	arena	.
subword tags	B	I	E	B	E	O	O	O	O
POS	NNP	NNP	NNP	VBD	VBD	IN	DT	NN	.
dep	nsubj	nsubj	nsubj	root	root	prep	det	pobj	root

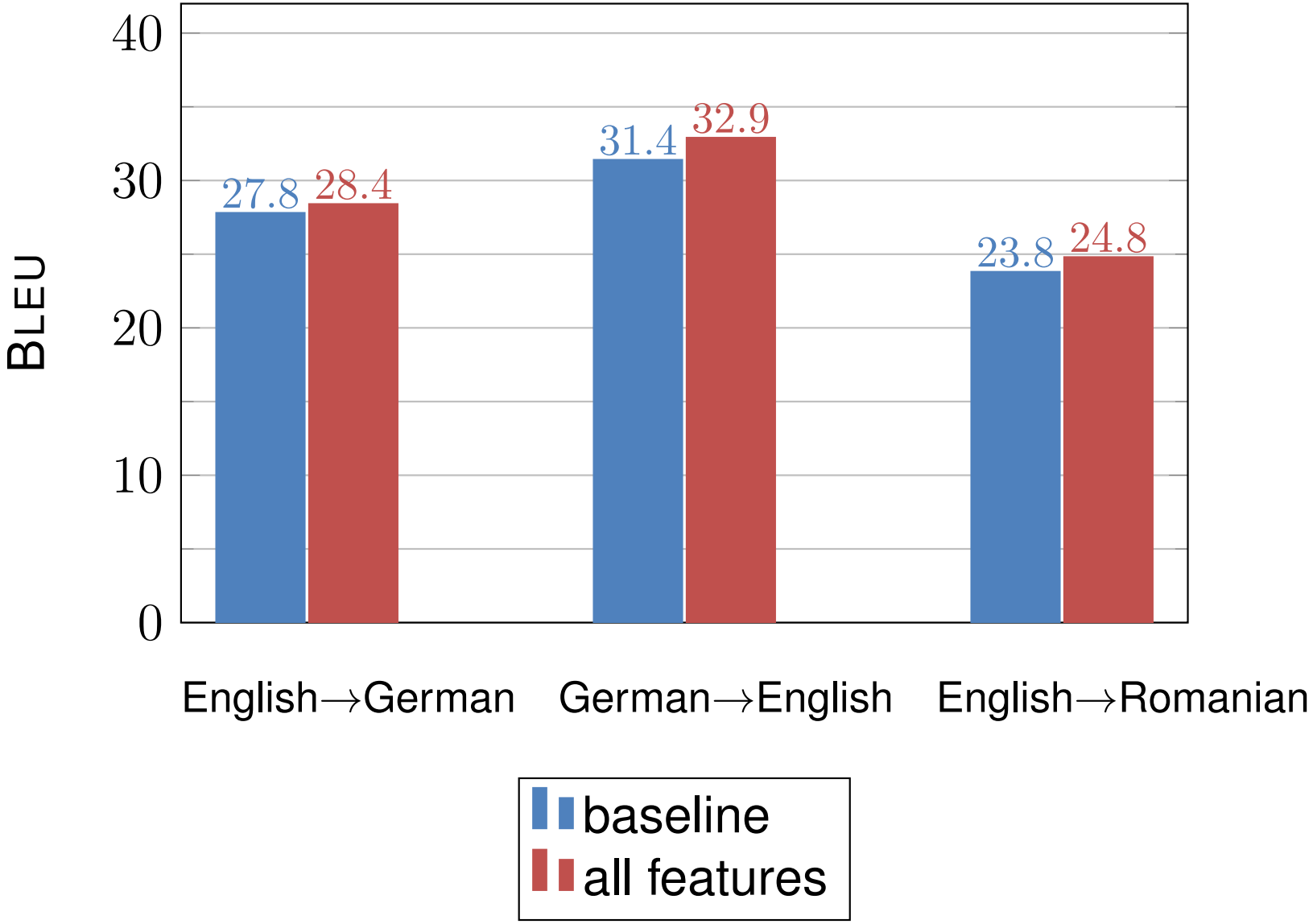
Results: Perplexity ↓



Results: Individual Features (Perplexity ↓)



Results: BLEU ↑

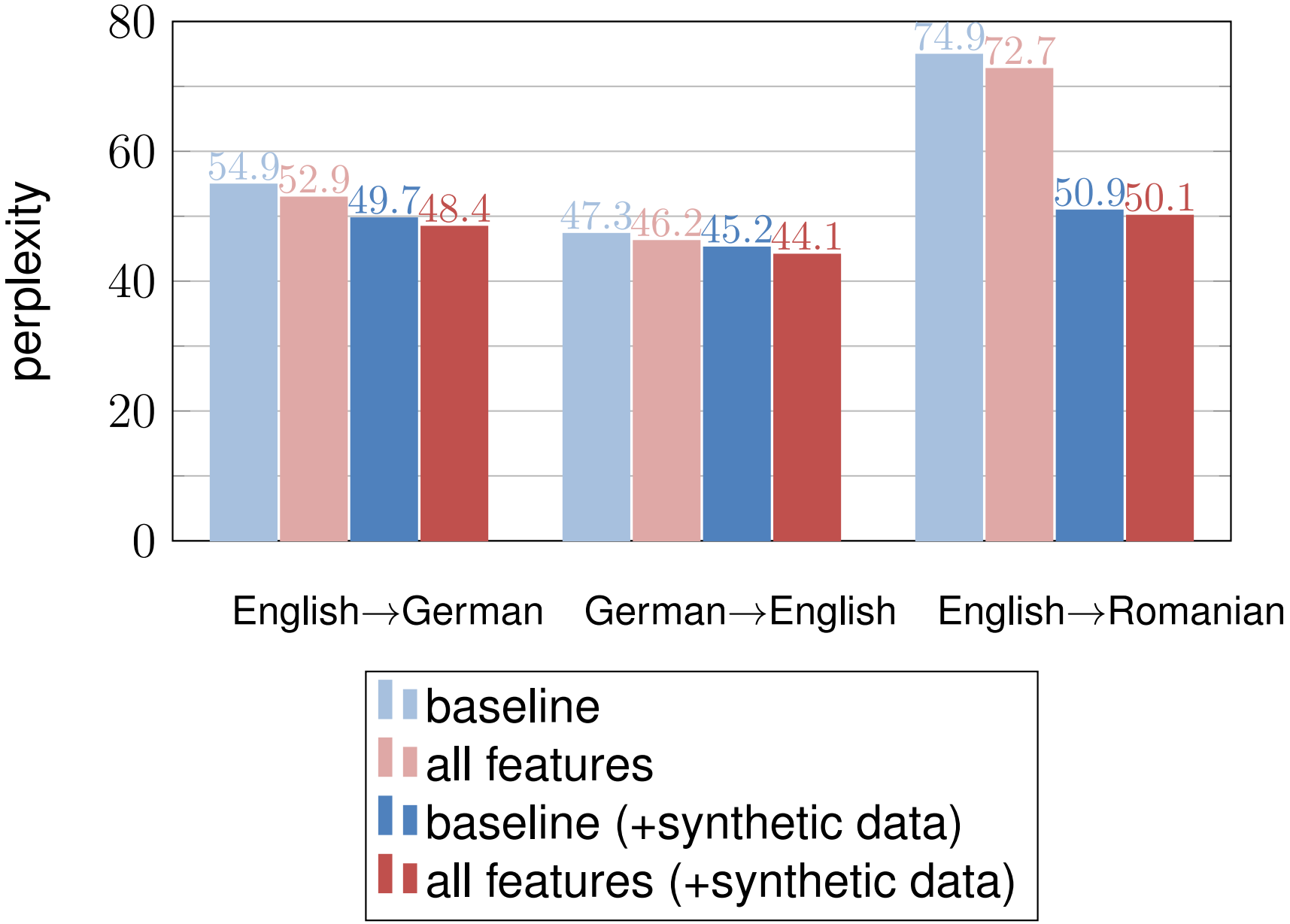


Evaluation: Synthetic Data

use synthetic training data from back-translated monolingual corpus

$\rightarrow$ stronger baseline, annotation of noisy input

Results: Perplexity ↓



Evaluation: Vocabulary and Embedding Size

feature	input vocabulary EN	DE	model	embedding all	single
subword tags	4	4	4	5	5
POS tags	46	54	54	10	10
morph. features	-	1400	1400	10	10
dependency labels	46	33	46	10	10
lemmas	800000	1500000	85000	115	167
words	78500	85000	85000	*	*

*: size is chosen to give total embedding size 500 (=baseline)

Examples

source	*We thought a win like this might be close$_{adj}$.*
reference	*Wir dachten, dass ein solcher Sieg nah sein könnte.*
baseline NMT	*Wir dachten, ein Sieg wie dieser könnte schließen.*
all features	*Wir dachten, ein Sieg wie dieser könnte nah sein.*

source	*Gefährlich$_{pred}$ ist die Route$_{subj}$ aber dennoch.*
reference	*However the route is dangerous.*
baseline NMT	*Dangerous is the route, however.*
all features	*However, the route is dangerous.*