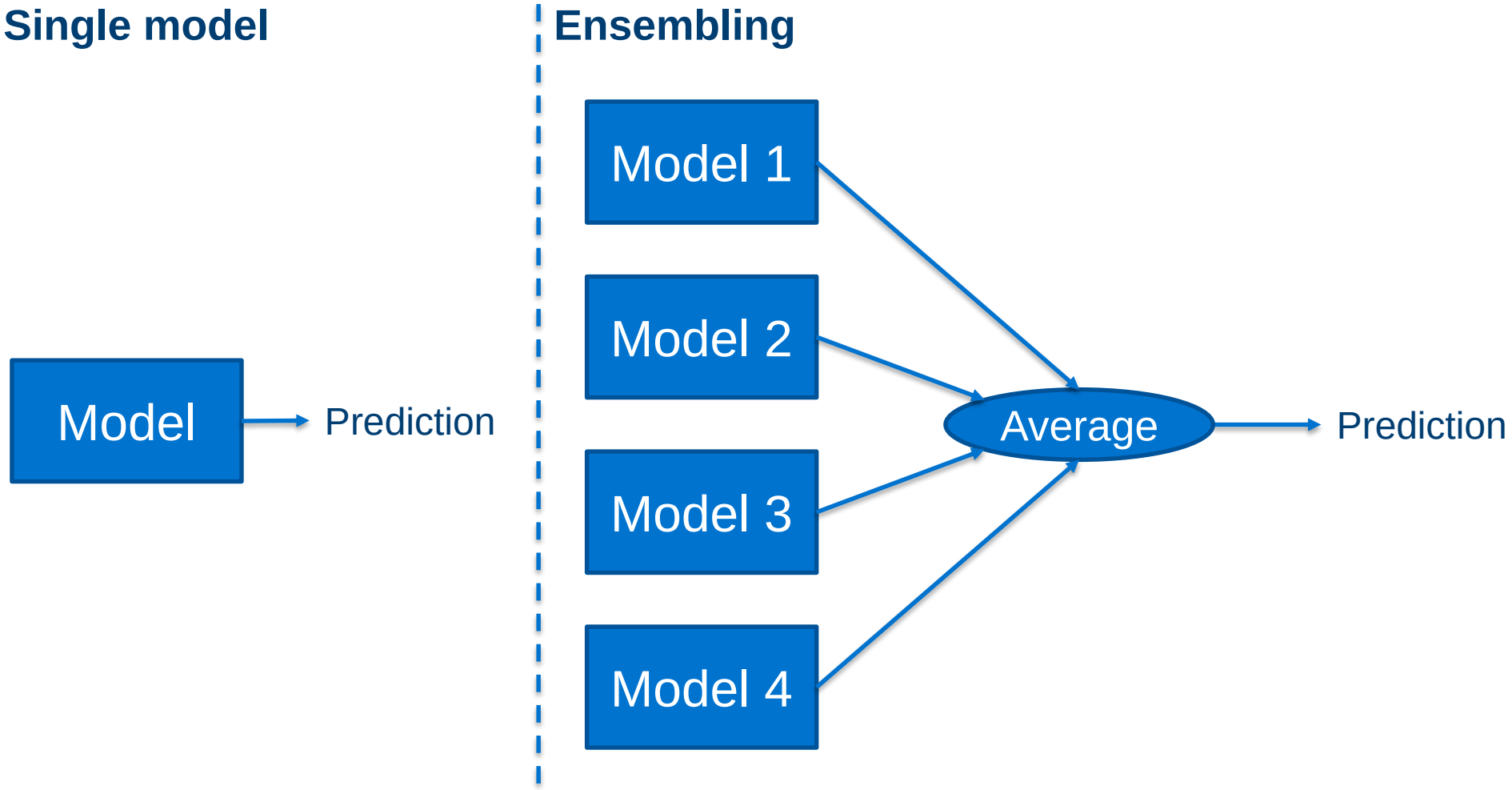
Unfolding and Shrinking Neural Machine Translation Ensembles

Felix Stahlberg and Bill Byrne

Department of Engineering

Ensembling in neural machine translation



Gains through ensembling

WMT top systems (UEdin)

	WMT'16 (En-De)
Single	31.6
Ensemble	34.2

+2.6 BLEU

	WMT'17 (En-De)
Single	26.6
Ensemble	28.3

+1.7 BLEU

http://matrix.statmt.org/

Google's NMT system

	WMT'14 (En-De)
Single	24.6
Ensemble	26.3

+1.7 BLEU

	WMT'14 (En-Fr)
Single	40.0
Ensemble	41.2

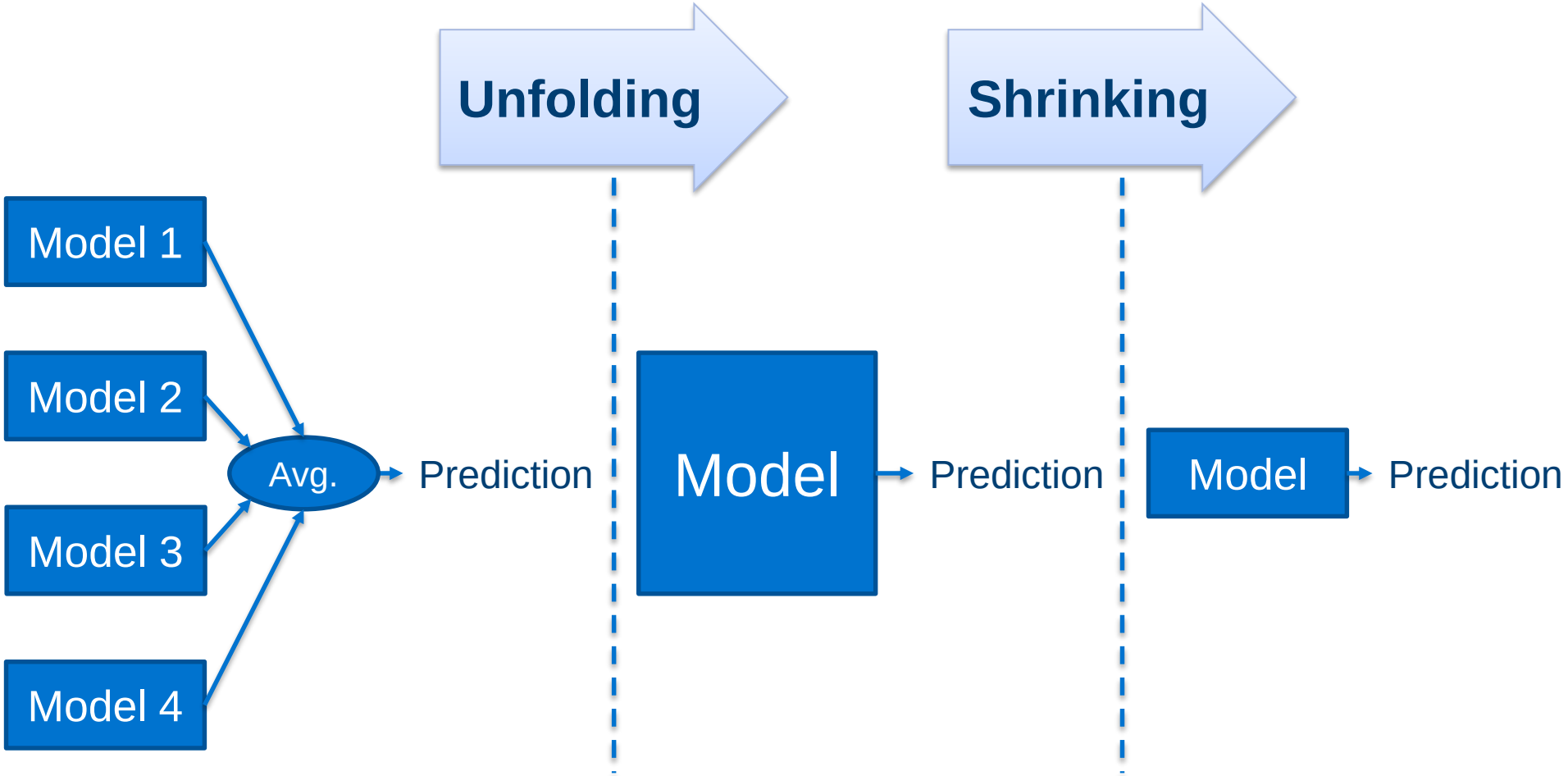
+1.2 BLEU

Wu, Yonghui, et al. "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation." arXiv preprint arXiv:1609.08144 (2016).

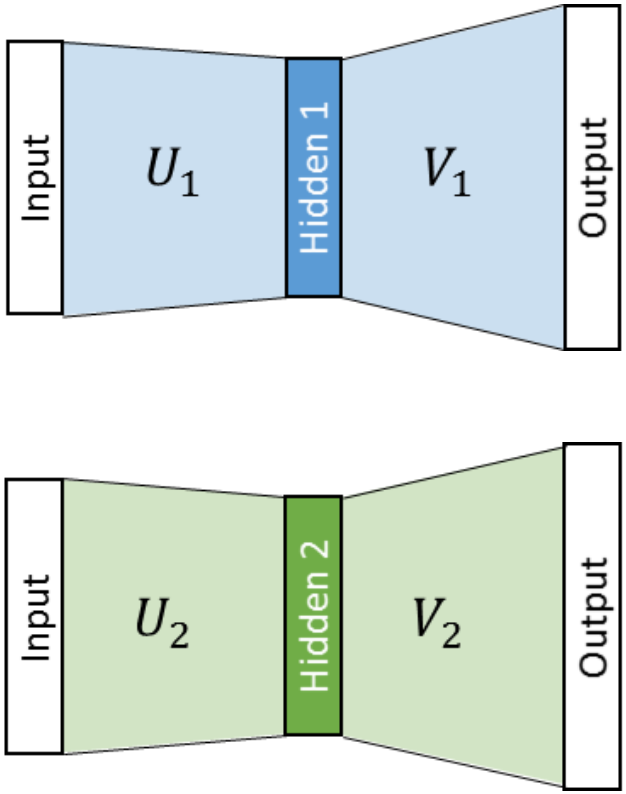
Disadvantages of ensembling

- Decoding with n-ensembles is slow
  - More CPU/GPU switches
  - n times more passes through the network at each decoding step
  - Applying softmax function n more times at each decoding step
- Ensembles are cumbersome
  - Often more difficult to implement

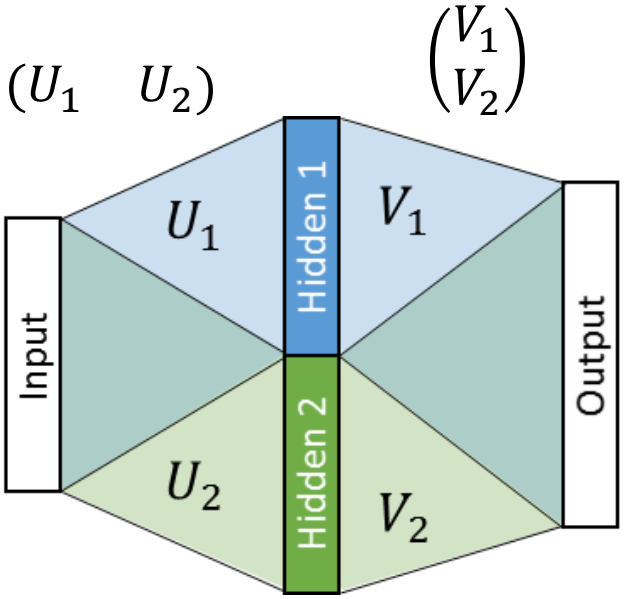
Unfolding and shrinking



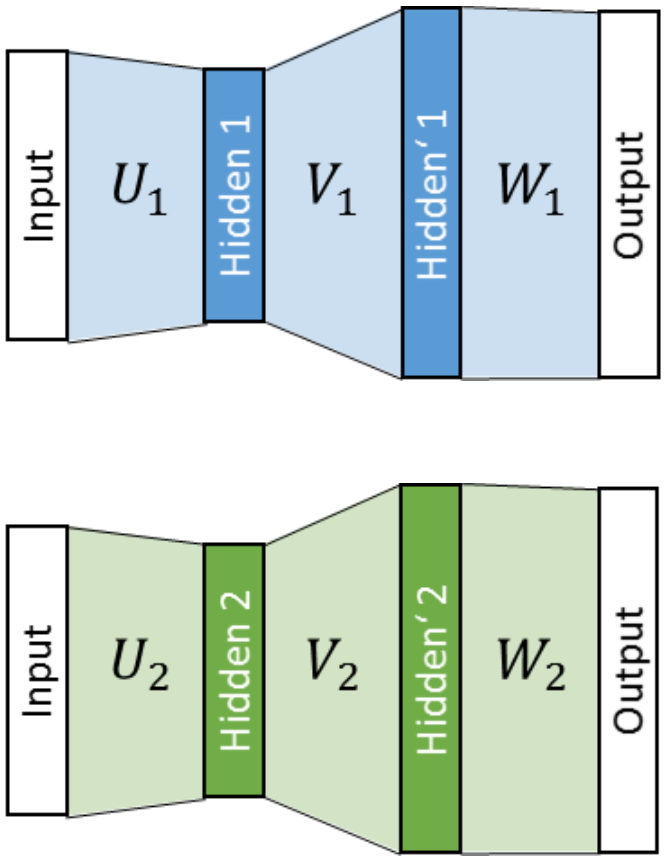
Unfolding a single layer

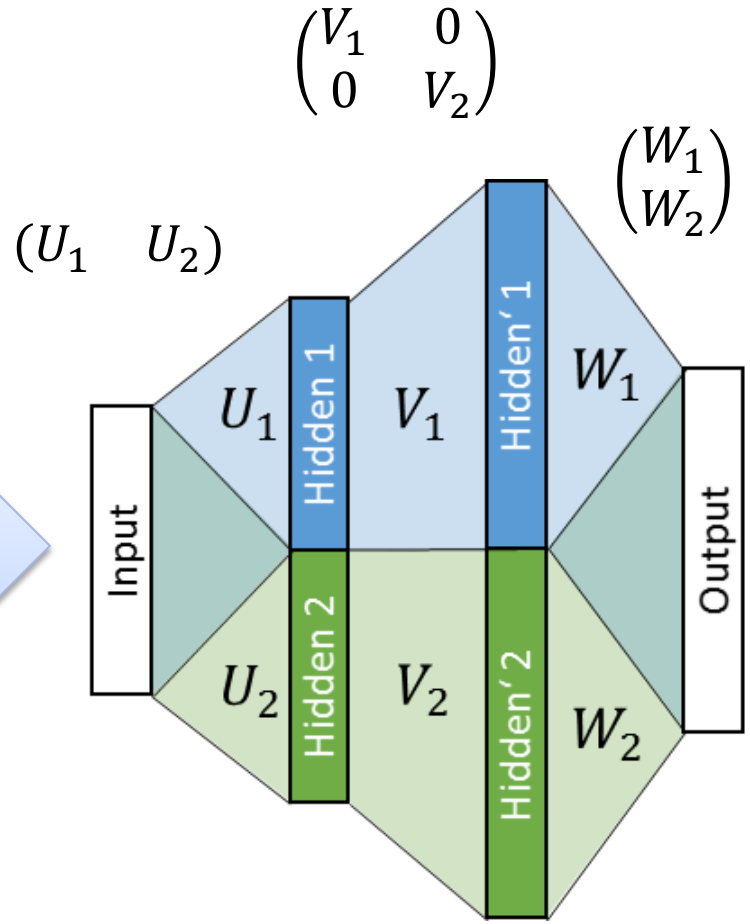


Unfolding



Unfolding multiple layers

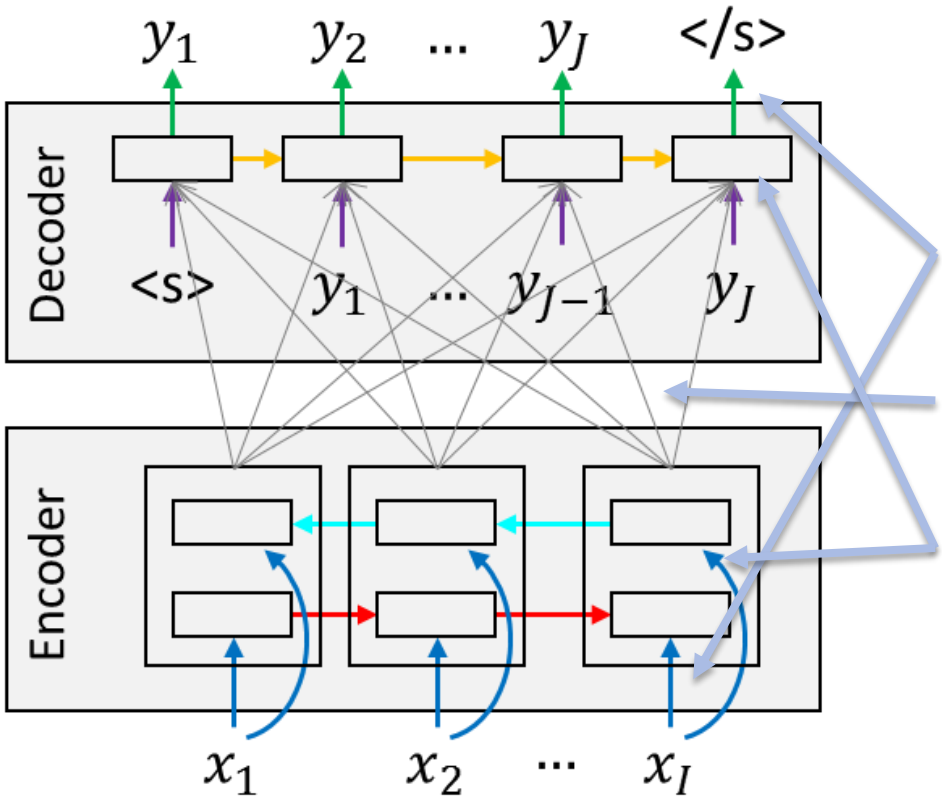




Shrinking – wish list

- Shrinking reduces the dimensionality of layers
  - Objective: Do not affect the behavior of the next layer
- Remove whole neurons rather than individual weights
  - Smaller model **and** faster decoding
  - Network layout is the same, ie. inference code remains unchanged
- Previous work is unsuitable
  - Weight pruning (LeCun et al., 1989; Hassibi et al., 1993; Han et al., 2015; See et al., 2016; …)
  - Approximating non-linear neurons with linear neurons (White, 2008)
  - Network compression methods based on low rank matrix factorization (Denil et al., 2013; Denton et al., 2014; Xue et al., 2013; Prabhavalkar et al., 2016; Lu et al., 2016; ...)

Shrinking NMT (Bahdanau et al., 2015) networks

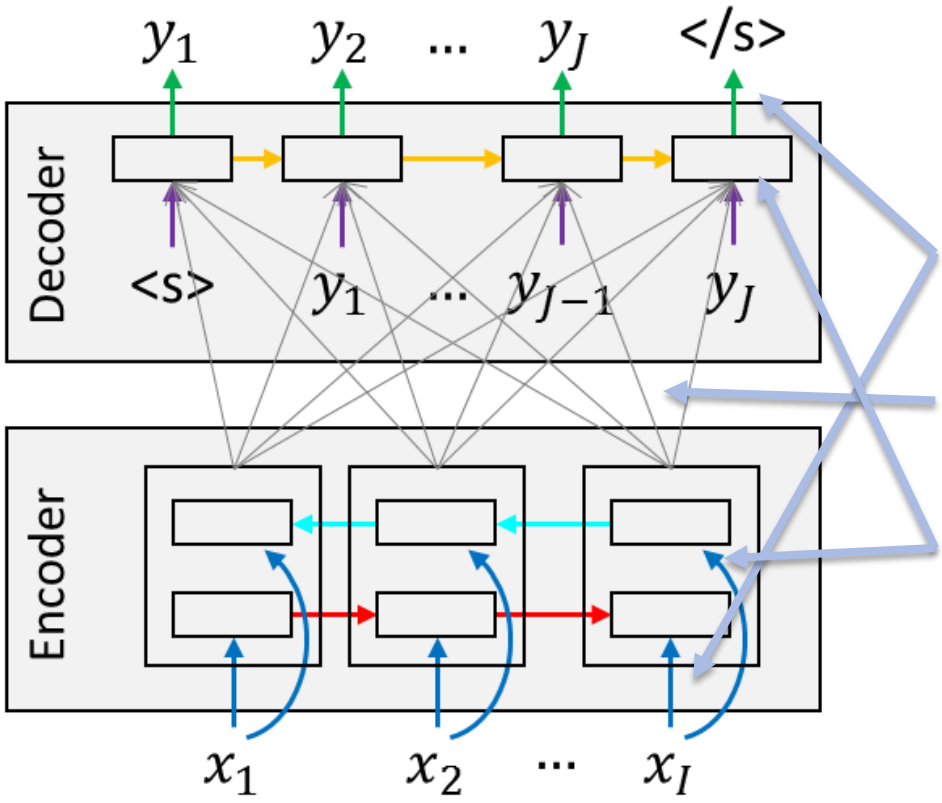


Embedding layers: SVD-based shrinking

Attention: Data-free shrinking

GRU cells: Data-bound shrinking

Shrinking NMT (Bahdanau et al., 2015) networks

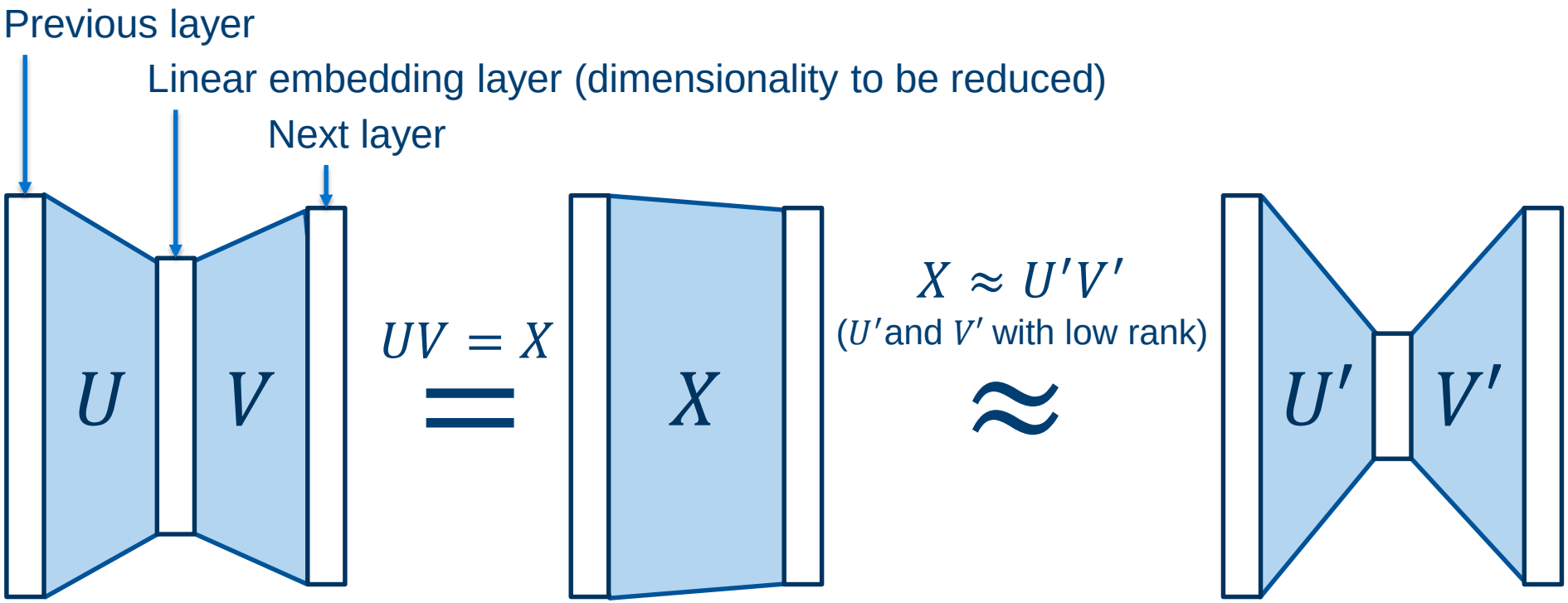


Embedding layers: SVD-based shrinking

Attention: Data-free shrinking

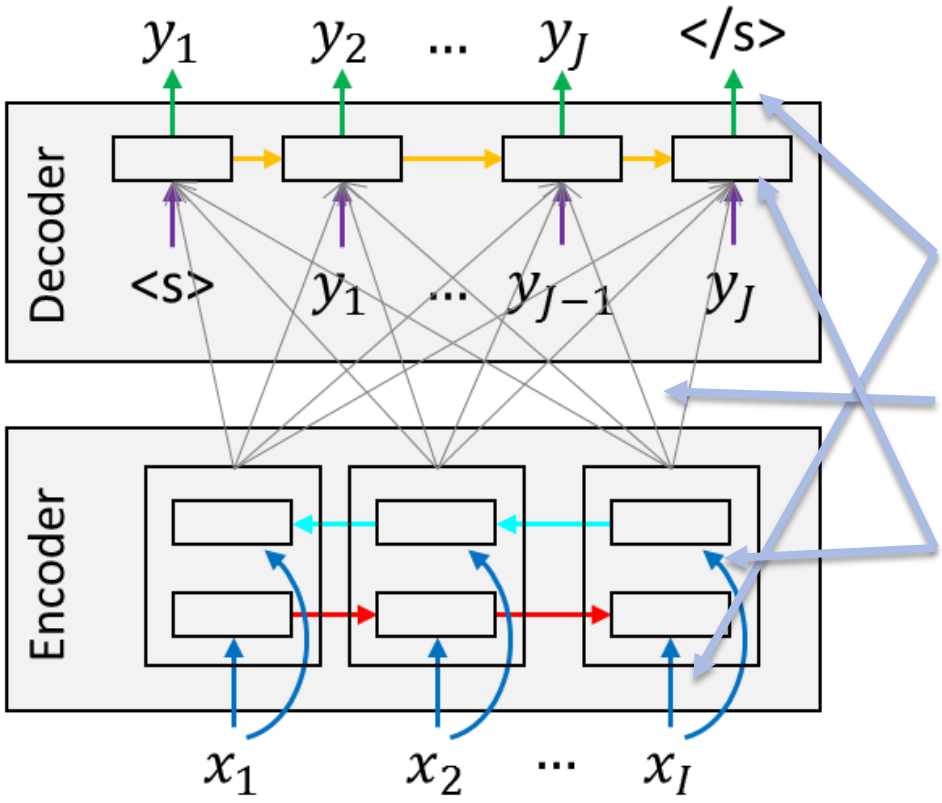
GRU cells: Data-bound shrinking

Shrinking linear layers with low-rank matrix factorization



We use truncated SVD for the factorization

Shrinking NMT (Bahdanau et al., 2015) networks

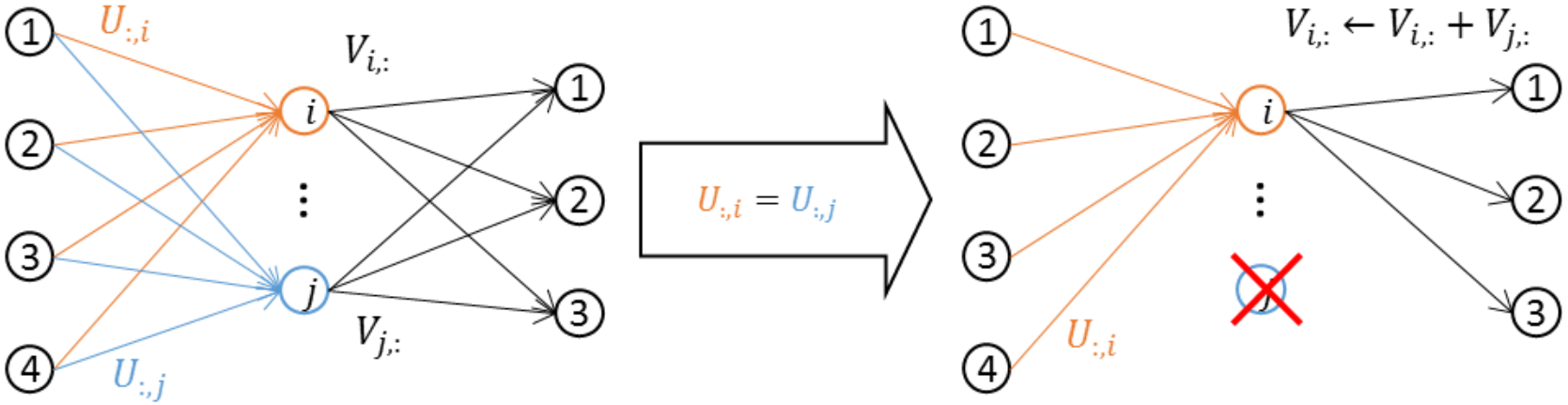


Embedding layers: SVD-based shrinking

Attention: Data-free shrinking

GRU cells: Data-bound shrinking

Approximating a neuron with its most similar neighbor
(Srinivas and Babu, 2015)

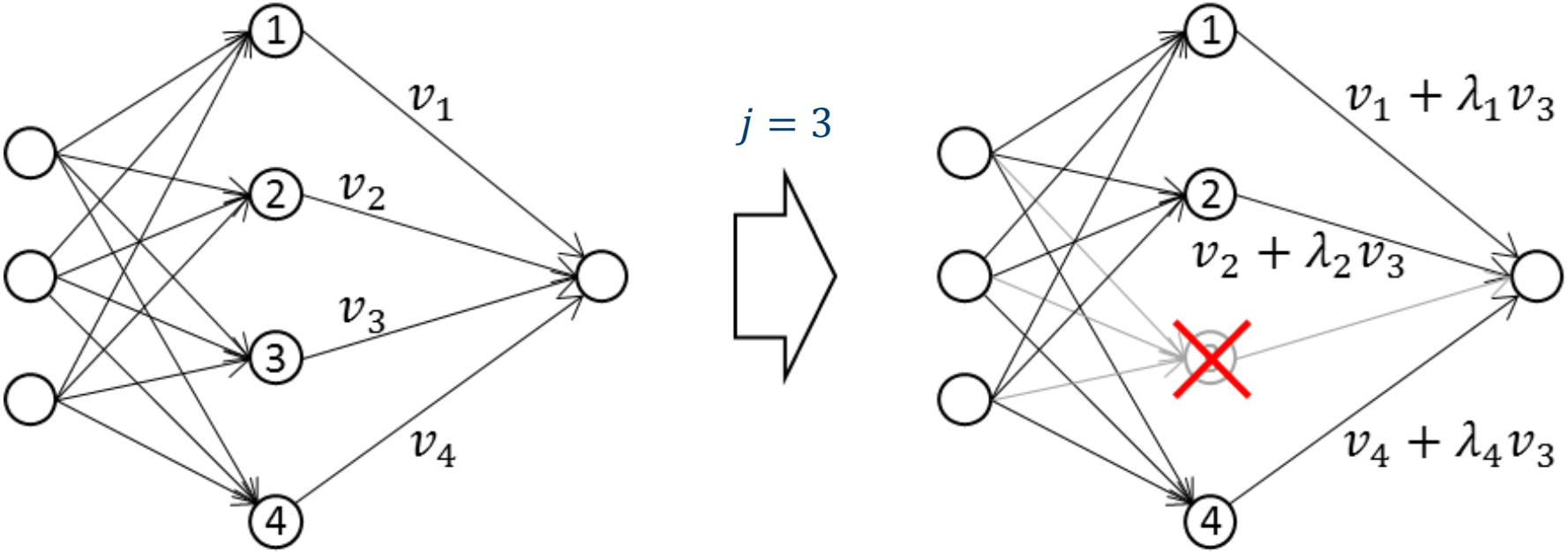


Selection criterion: $\underset{i,j \in [1,m]}{\arg\min} ||U_{:,i} - U_{:,j}||_2^2 ||V_{j,:}||_2^2$

Similar incoming weights

Small outgoing weights

Approximating a neuron with a linear combination of its neighbors



How to estimate λ?

Data-free and data-bound shrinking

U: Incoming weight matrix
λ: Interpolation weights

Data-free shrinking

„Approximate incoming weights“

$$U_{:,\neg j}\lambda = U_{:,j}$$

Data-free and data-bound shrinking

U: Incoming weight matrix
λ: Interpolation weights
A: Neuron activity matrix

Data-free shrinking

„Approximate incoming weights“

$$U_{:,\neg j}\lambda = U_{:,j}$$

Theory: Set the expected error introduced by shrinking to zero assuming a linear activation function.

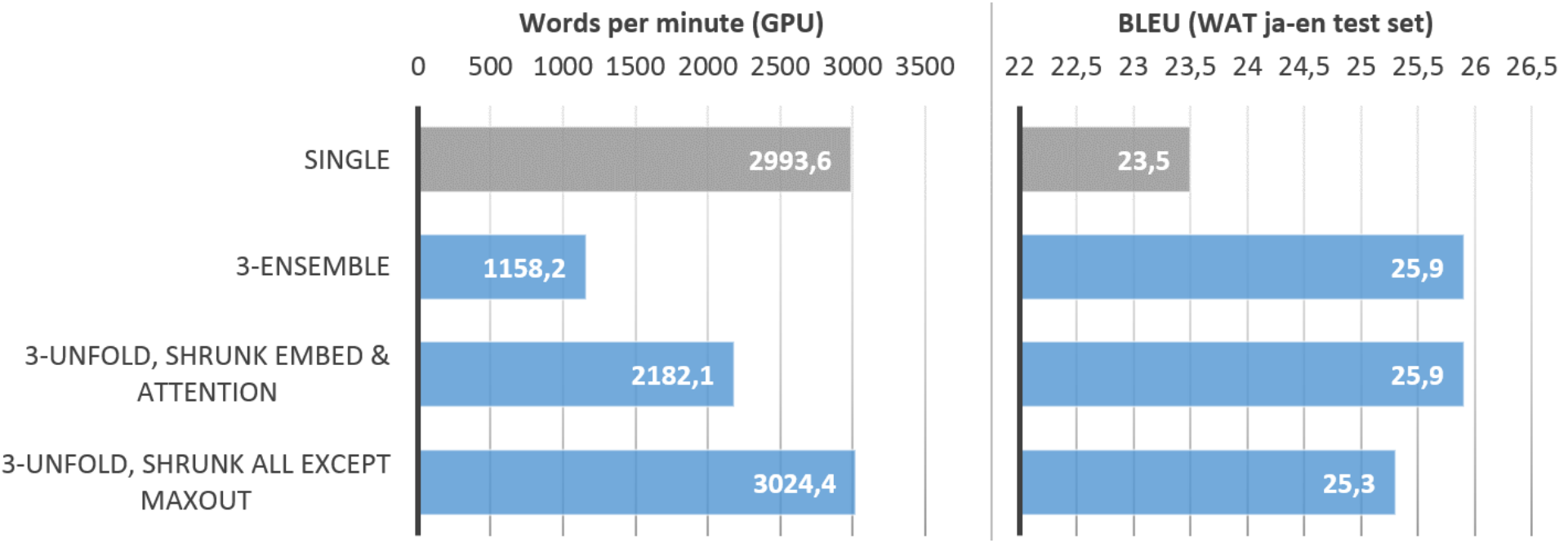
Data-bound shrinking

„Directly approximate neuron activity“

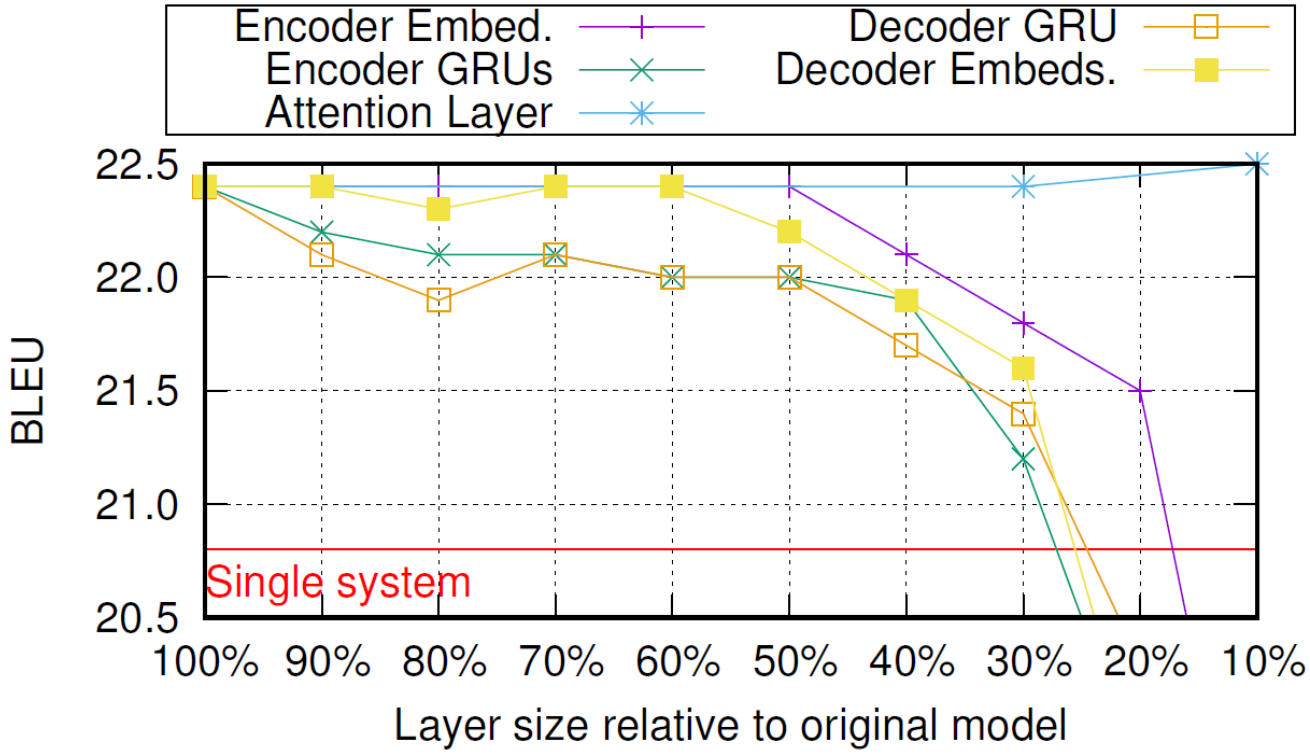
$$A_{:,\neg j}\lambda = A_{:,j}$$

Theory: Set the expected error introduced by shrinking to zero by estimating the expected neuron activities with importance sampling.

Shrinking layers to their original size (Japanese-English)



Impact on BLEU of shrinking individual layers



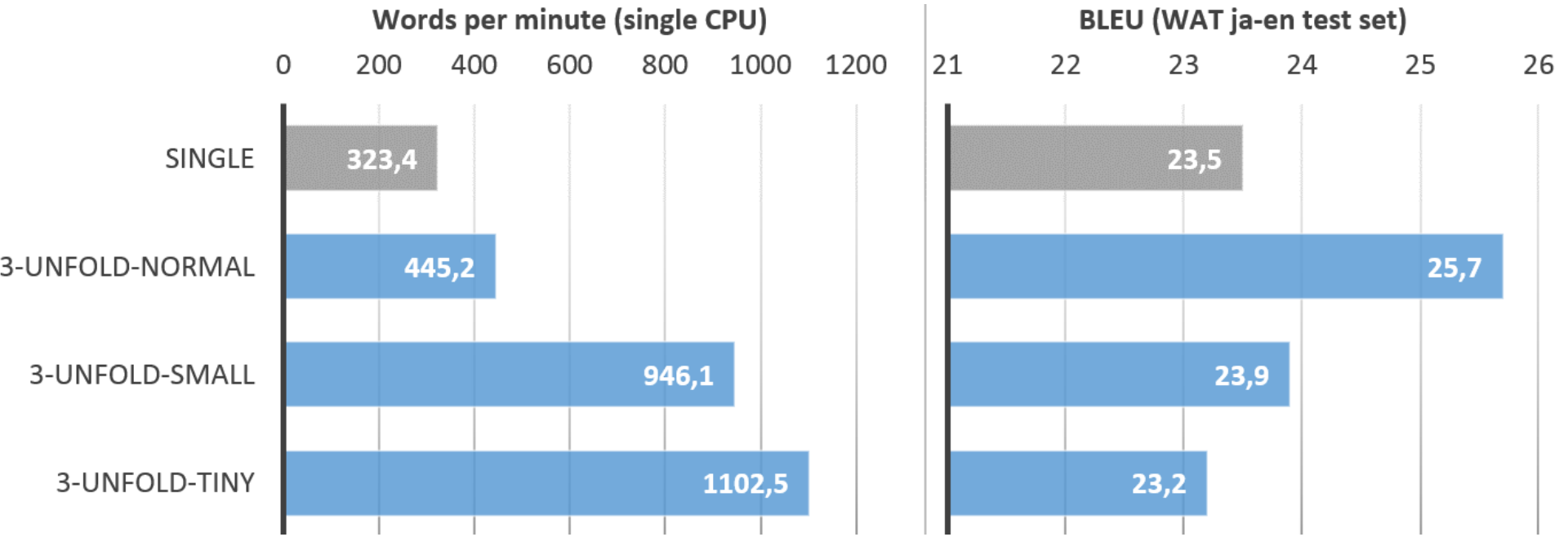
- Individual layers can be shrunk even below their original size
- GRU layers are more sensitive to shrinking than embedding or attention layers

Designing three setups for Japanese-English

Layer sizes

	Single	3-Unfold		
		Normal	Small	Tiny
Enc. Embed.	620	410	310	170
Enc. GRUs	1000	1300	580	580
Attention	1000	100	100	100
Dec. GRU	1000	1350	590	590
Dec. Maxout	500	1500	1500	1500
Dec. Embeds.	620	430	320	170
Size Factor	1.00	1.00	0.50	0.33

Designing three setups for Japanese-English



(Unbatched) GPU decoding speed is roughly constant after unfolding, but shrinking makes batching more effective

Conclusion

- Unfolding yields ensemble level performance with a single network
  - Often faster and easier to deploy
- Shrinking can reduce the size of unfolded networks significantly
  - Depending on the aggressiveness of pruning, shrinking+unfolding yields either
    - +2.2 BLEU at the same decoding speed or
    - 3.4 times CPU speed up with a minor drop in BLEU
- Our work indicates huge amounts of wasted computation
  - High dimensional embedding and attention layers may be needed for training, but are not necessary for inference

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In ICLR. Toulon, France.
- Misha Denil, Babak Shakibi, Laurent Dinh, Nando de Freitas, et al. 2013. Predicting parameters in deep learning. In Advances in Neural Information Processing Systems. pages 2148–2156.
- Emily L. Denton, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. 2014. Exploiting linear structure within convolutional networks for efficient evaluation. In Advances in Neural Information Processing Systems. pages 1269–1277.
- Song Han, Jeff Pool, John Tran, and William Dally. 2015. Learning both weights and connections for efficient neural network. In Advances in Neural Information Processing Systems. pages 1135–1143.
- Babak Hassibi, David G. Stork, et al. 1993. Second order derivatives for network pruning: Optimal brain surgeon. Advances in neural information processing systems pages 164–164.
- Yann LeCun, John S. Denker, Sara A. Solla, Richard E. Howard, and Lawrence D. Jackel. 1989. Optimal brain damage. In NIPS. volume 2, pages 598–605.
- Zhiyun Lu, Vikas Sindhwani, and Tara N. Sainath. 2016. Learning compact recurrent neural networks. In ICASSP, pages 5960–5964.
- Rohit Prabhavalkar, Ouais Alsharif, Antoine Bruguier, and Lan McGraw. 2016. On the compression of recurrent neural networks with an application to LVCSR acoustic modeling for embedded speech recognition. In ICASSP, pages 5970–5974.
- Abigail See, Minh-Thang Luong, and Christopher D. Manning. 2016. Compression of neural machine translation models via pruning. CoNLL 2016 pages 291–299.
- Suraj Srinivas and R. Venkatesh Babu. 2015. Data-free parameter pruning for deep neural networks. arXiv preprint arXiv:1507.06149 .
- Jian Xue, Jinyu Li, and Yifan Gong. 2013. Restructuring of deep neural network acoustic models with singular value decomposition. In Interspeech. pages 2365–2369.
- White, Halbert. “Learning in artificial neural networks: A statistical perspective.” Learning 1.4 (2008)

Thanks