Deep Encoder, Shallow Decoder: Reevaluating Non-autoregressive MT

Jungo Kasai, Nikolaos Pappas, Hao Peng, James Cross, Noah A. Smith
Paul G. Allen School of CSE, University of Washington
Facebook AI

Summary

- Non-autoregressive MT (**NAR**) is a recent fast alternative to **AR** MT.

Summary

- Non-autoregressive MT (**NAR**) is a recent fast alternative to **AR** MT.
- Parallel generation **underperforms** yet **outpaces** left-to-right generation on a GPU.

Summary

- Non-autoregressive MT (**NAR**) is a recent fast alternative to **AR** MT.
- Parallel generation **underperforms** yet **outpaces** left-to-right generation on a GPU.
- Reexamines the speed-accuracy tradeoff.

Summary

- Non-autoregressive MT (**NAR**) is a recent fast alternative to **AR** MT.
- Parallel generation **underperforms** yet **outpaces** left-to-right generation on a GPU.
- Reexamines the speed-accuracy tradeoff.
    - Suboptimal Layer Allocation
    - Insufficient Speed Measurement
    - Lack of Knowledge Distillation for AR Baselines

Fast, Accurate Machine Translation

Fast, Accurate Machine Translation

- MT's generation quality improved in the past 10 years, but can we make it faster?
- Speed is important!
  - Need for massive amount of translation
    - Google translates 100B+ words a day
    - EU translates 1M pages every year

Fast, Accurate Machine Translation

- MT's generation quality improved in the past 10 years, but can we make it faster?
- Speed is important!
  - Some applications particularly need fast translation
    - Simultaneous Translation
    - Conversations

Fast, Accurate Machine Translation

- Many techniques to speed up NMT
    - Vocabulary Reduction (Shi&Knight 2017), Lightweight Attention (Zhang et al., 2018, Peng et al. 2021), Eager Translation (Press et al., 2018)...

Fast, Accurate Machine Translation

- Many techniques to speed up NMT
  - Vocabulary Reduction (Shi&Knight 2017), Lightweight Attention (Zhang et al., 2018, Peng et al. 2021), Eager Translation (Press et al., 2018)...
- Non-autoregressive MT (NAR, Gu et al., 2018)
  - Parallel Computation in Word Generation

Why can NAR be faster?

- Generation $P(\mathbf{Y}) \quad \mathbf{Y} = y_1, y_2, \cdots, y_T$
- Typically given **X** = source language (MT), text (text2speech) etc

Why can NAR be faster?

- Generation $P(\mathbf{Y}) \quad \mathbf{Y} = y_1, y_2, \cdots, y_T$
- Autoregressive $$P(\mathbf{Y}) = \prod_{i=1}^{T} P(y_i | \mathbf{Y}_{<i})$$

Why can NAR be faster?

- Generation

$$P(\mathbf{Y}) \quad \mathbf{Y} = y_1, y_2, \cdots, y_T$$

- Autoregressive

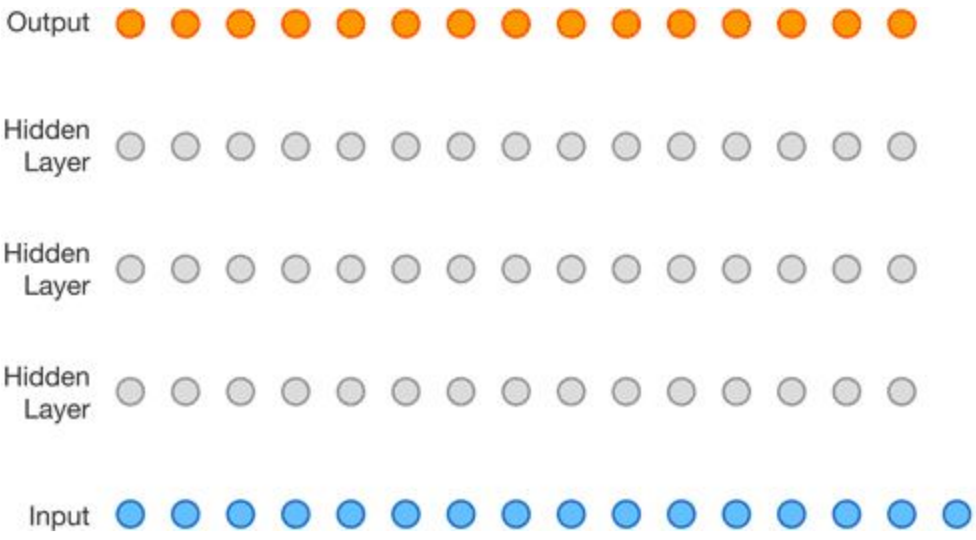
$$P(\mathbf{Y}) = \prod_{i=1}^{T} P(y_i | \mathbf{Y}_{<i})$$

- Non-autoregressive

$$P(\mathbf{Y}) = \prod_{i=1}^{T} P(y_i)$$

Why NAR?

- Speed Overhead in Autoregressive Generation.



https://deepmind.com/blog/article/high-fidelity-speech-synthesis-wavenet

Why NAR?

- Speed up **Generation** by Parallelism.

https://deepmind.com/blog/article/high-fidelity-speech-synthesis-wavenet

Speed-Accuracy Tradeoff in NAR

- **Multimodality** (Gu et al. 2018)
  - Language is highly multimodal. Can’t mix two sentences (modes).
  - You’re welcome <-> My pleasure

Flurry of Recent Work in NAR

- Iterative Methods
  - Lee et al., 2018; Ghazvininejad et al., 2019; Gu et al. 2019; Kasai et al. 2020...

Flurry of Recent Work in NAR

- Iterative Methods
  - Lee et al., 2018; Ghazvininejad et al., 2019; Gu et al. 2019; Kasai et al. 2020...
- Light Autoregressive
  - Kaiser et al. 2018; Stern et al. 2018, 2019; Sun et al. 2019

Flurry of Recent Work in NAR

- Iterative Methods
  - Lee et al., 2018; Ghazvininejad et al., 2019; Gu et al. 2019; Kasai et al. 2020...
- Light Autoregressive
  - Kaiser et al. 2018; Stern et al. 2018, 2019; Sun et al. 2019
- Latent Variables
  - Ma et al. 2019; Shu et al. 2020

Flurry of Recent Work in NAR

- Iterative Methods
  - Lee et al., 2018; Ghazvininejad et al., 2019; Gu et al. 2019; Kasai et al. 2020...
- Light Autoregressive
  - Kaiser et al. 2018; Stern et al. 2018, 2019; Sun et al. 2019
- Latent Variables
  - Ma et al. 2019; Shu et al. 2020

NAR MT is strictly better now?

- Some of these recent works claim NAR outspaces AR with equivalent accuracy.

NAR MT is strictly better now?

- Some of these recent works claim NAR outspeeds AR with equivalent accuracy.
- Wait, we're being unfair to AR!
  - Speed Measurements
  - Layer Allocation
  - Knowledge Distillation

Reevaluating NAR

Speed Measure

- **S1 (Most NAR Works)**
  - 1 sentence (utterance) at a time
  - Instantaneous Translation, Simultaneous Translation,...

Speed Measure

- **S1 (Most NAR Works)**
  - 1 sentence (utterance) at a time
  - Instantaneous Translation, Simultaneous Translation,...
- **Smax**
  - Maximum Batch Size
  - Translate Wikipedia, EU Documents, ...

Speed Measure

- Translation Services related to Smax
  - Batched Translation for Large Web Text
    - Amazon, Google Cloud etc

Knowledge Distillation

- Mitigates Multimodality (Gu et al. 2018).
    - Almost all NAR models need KD.
    - AR MT output is less diverse than human (Shen et al. 2019).

Knowledge Distillation

- Mitigates Multimodality (Gu et al. 2018)
    - IWSLT EN-DE Validation

Distillation		Decoder Inputs			Fine-tuning			BLEU
b=1	b=4	+uniform	+fertility	+PosAtt	$+\mathcal{L}_{\text{KD}}$	$+\mathcal{L}_{\text{BP}}$	$+\mathcal{L}_{\text{RL}}$	
				✓				≈ 2
		✓		✓				16.51
			✓	✓				18.87

Knowledge Distillation

- Mitigates Multimodality (Gu et al. 2018)
    - IWSLT EN-DE Validation

Distillation		Decoder Inputs			Fine-tuning			
b=1	b=4	+uniform	+fertility	+PosAtt	$+\mathcal{L}_{\text{KD}}$	$+\mathcal{L}_{\text{BP}}$	$+\mathcal{L}_{\text{RL}}$	BLEU
				✓				≈ 2
		✓		✓				16.51
			✓	✓				18.87
✓		✓		✓				20.72
	✓	✓		✓				21.12
✓			✓					24.02
✓			✓	✓				25.20

Knowledge Distillation for MT

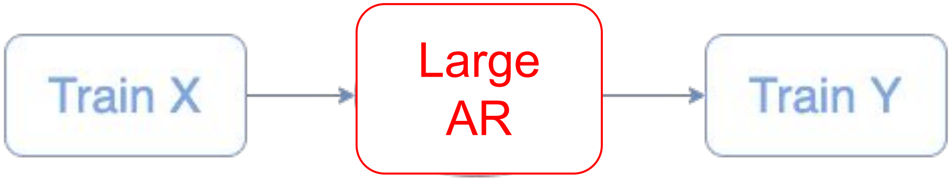
- **Sequence-Level Knowledge Distillation** (Kim & Rush 2016)

Train X

Train Y

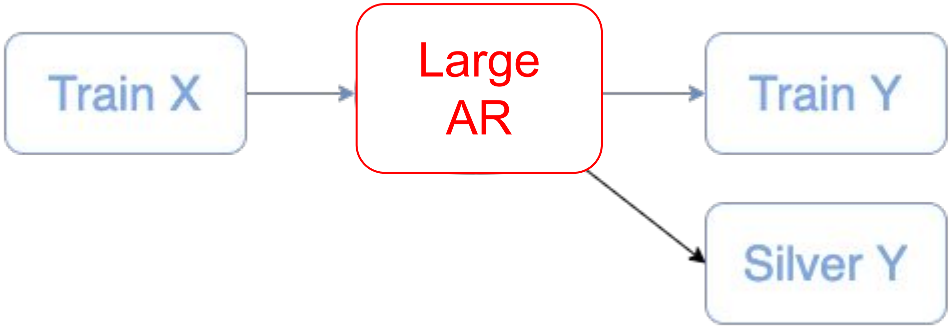
Knowledge Distillation for MT

- **Sequence-Level Knowledge Distillation** (Kim & Rush 2016)



Knowledge Distillation for MT

- **Sequence-Level Knowledge Distillation** (Kim & Rush 2016)

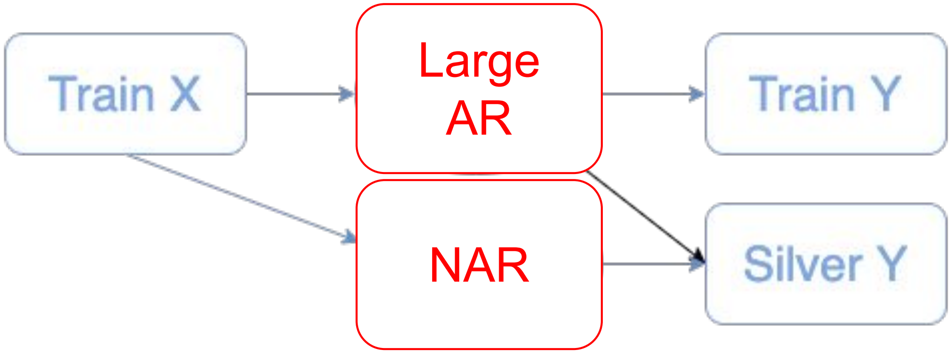


John really likes Mary

John loves Mary

Knowledge Distillation for MT

- **Sequence-Level Knowledge Distillation** (Kim & Rush 2016)

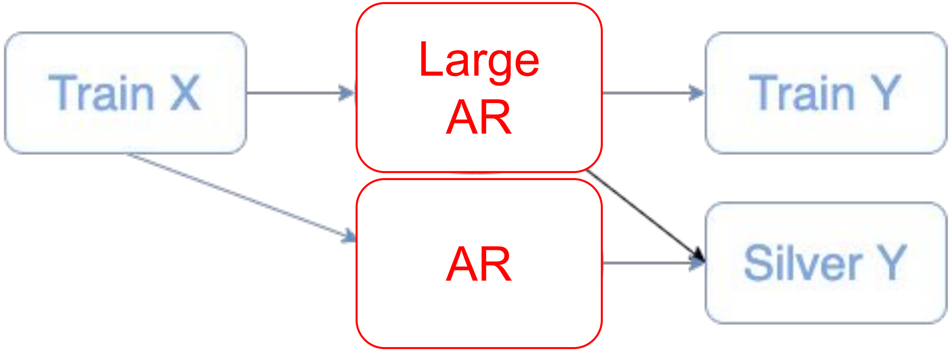


John really likes Mary

John loves Mary

Knowledge Distillation for MT

- **This work applies KD to AR baselines as well.**



John really likes Mary

John loves Mary

Layer Allocation

- Equal depths in the encoder and decoder are typically assumed.
- They have different accuracy and speed implications.

Layer Allocation

- Equal depths in the encoder and decoder are typically assumed.
- They have different accuracy and speed implications.
- Experiments with varying depths.
- **Deep-Shallow** speeds up AR MT with accuracy retained.
    - AR’s speed disadvantage is overestimated.

Implications on Speed

- Speed up **S1** with a shallow decoder. Increasing the encoder depth only causes a mild slowdown.

	Full Model		
	AR E-D	AR E-1	NAR E-D
Total Operations	$\mathcal{O}(EN^2 + DN^2)$	$\mathcal{O}(EN^2 + 1 \cdot N^2)$	$\mathcal{O}(EN^2 + DTN^2)$
Time Complex.	$\mathcal{O}(EN + DN^2)$	$\boxed{\mathcal{O}(EN + N^2)}$	$\mathcal{O}(EN + DTN)$

Implications on Speed

- Speed up **S1** with a shallow decoder. Increasing the encoder depth only causes a mild slowdown.
- With large batches, the increased total operation in NAR can slow down **Smax**.

	Full Model		
	AR E-D	AR E-1	NAR E-D
Total Operations	$\mathcal{O}(EN^2 + DN^2)$	$\mathcal{O}(EN^2 + 1 \cdot N^2)$	$\boxed{\mathcal{O}(EN^2 + DTN^2)}$
Time Complex.	$\mathcal{O}(EN + DN^2)$	$\mathcal{O}(EN + N^2)$	$\mathcal{O}(EN + DTN)$

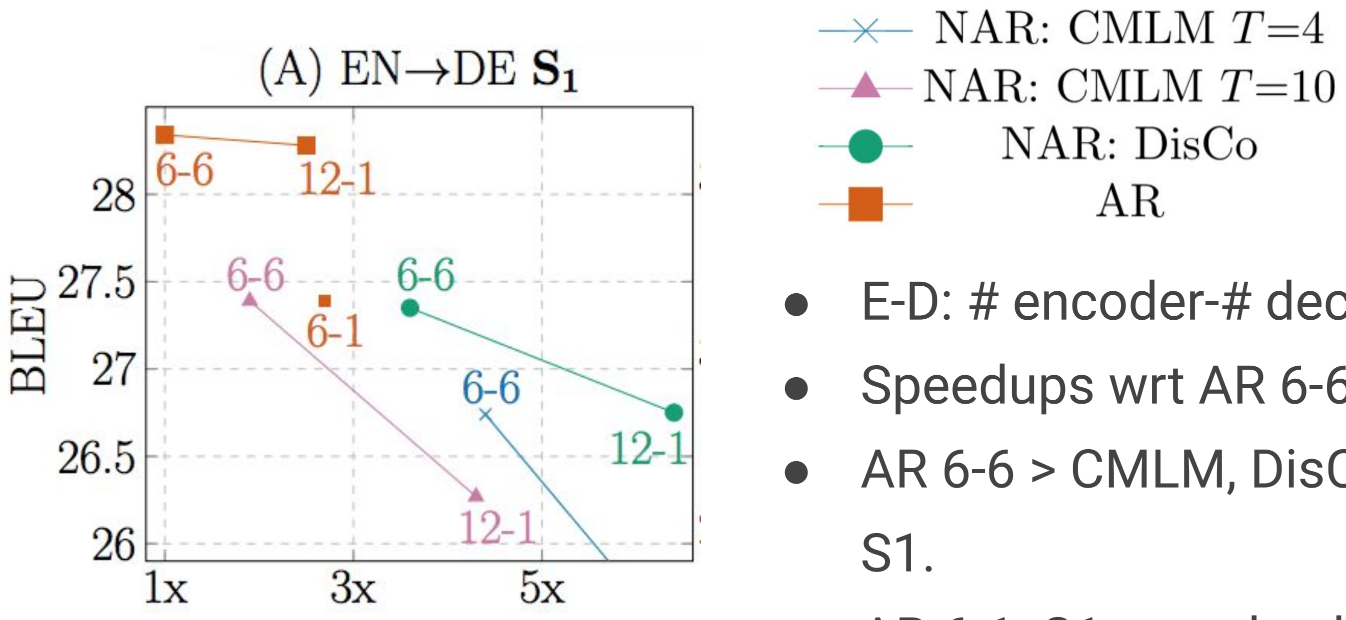
Experiments

Setups: Benchmarks

- Follow prior NAR works (Ghazvininejad et al., 2019; Kasai et al., 2020)
- BPE subwords

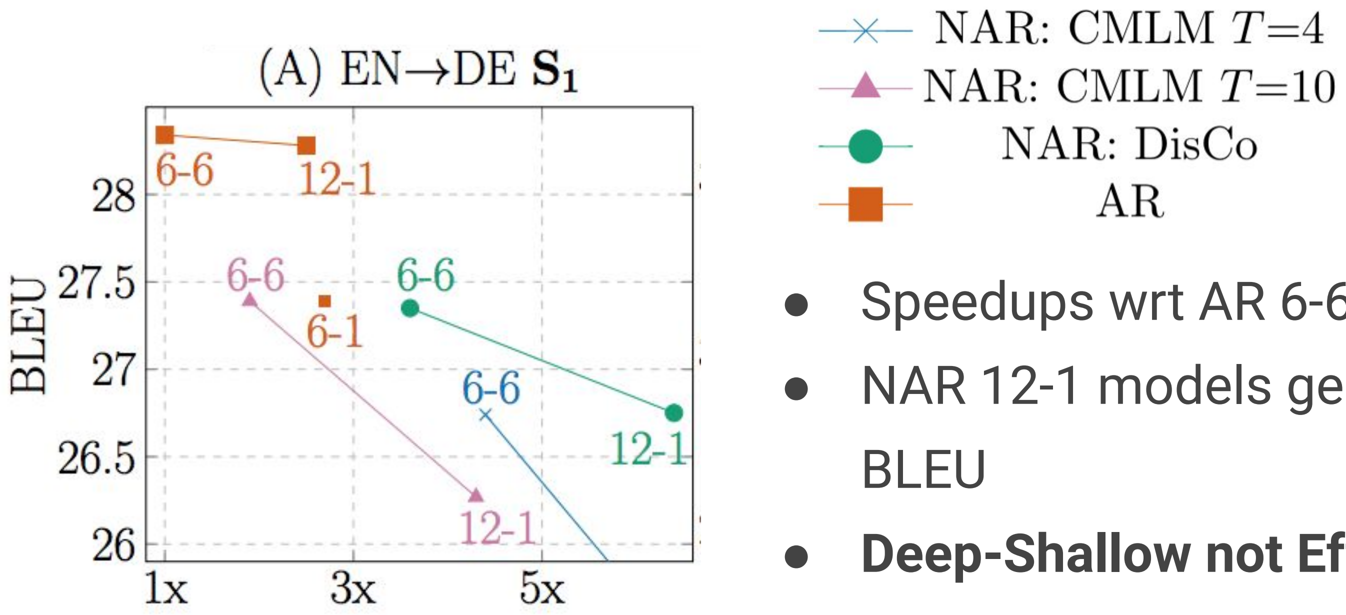
	Train Pairs	Teacher Transformer	Model
WMT 2016 EN-DE	4.5M	Large	Base
WMT 2016 EN-RO	610K	Base	Base
WMT 2017 EN-ZH	20M	Large	Base
WMT 2014 EN-FR	36M	Large	Base

Speed-Accuracy Tradeoff S1



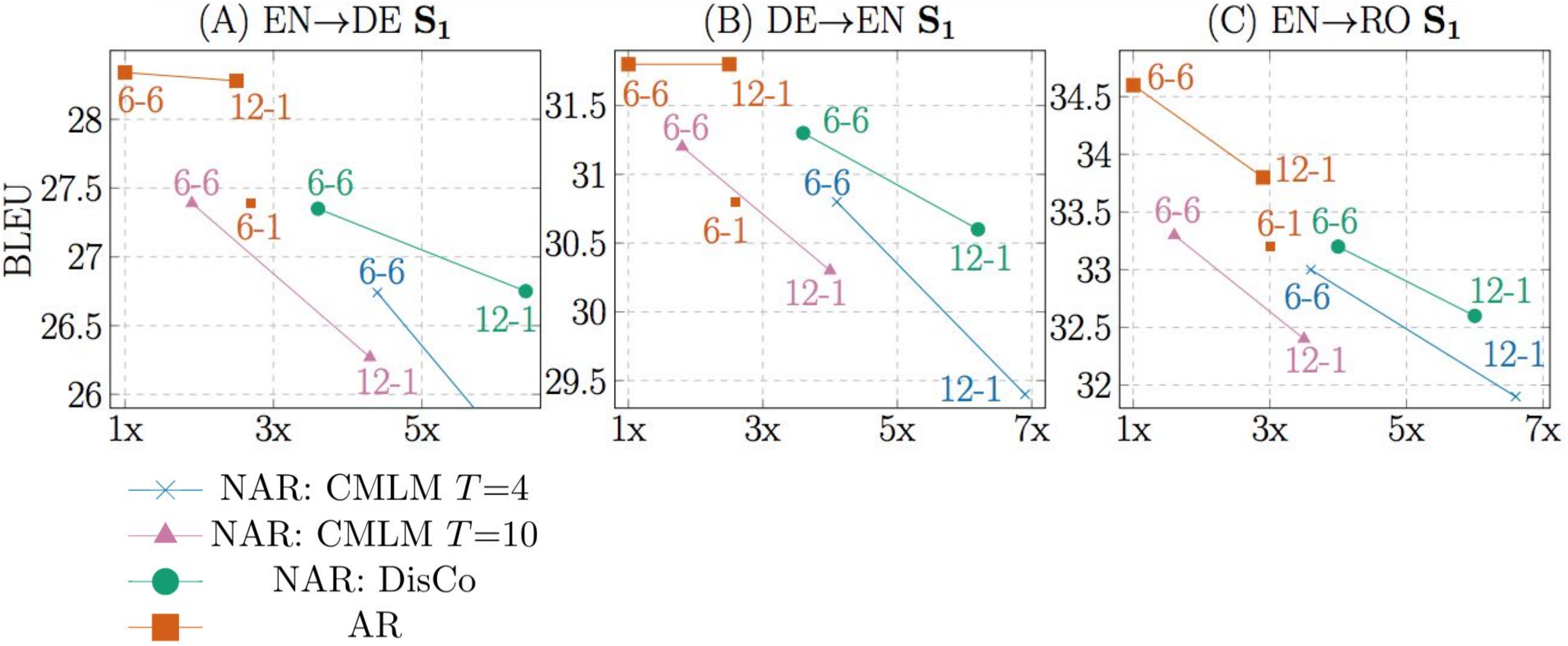
- E-D: # encoder-# decoder
- Speedups wrt AR 6-6 Baseline
- AR 6-6 > CMLM, DisCo but slow in S1.
- AR 6-1: S1 speedup but loss in BLEU.
- **AR 12-1: a balanced middle ground.**

Speed-Accuracy Tradeoff S1

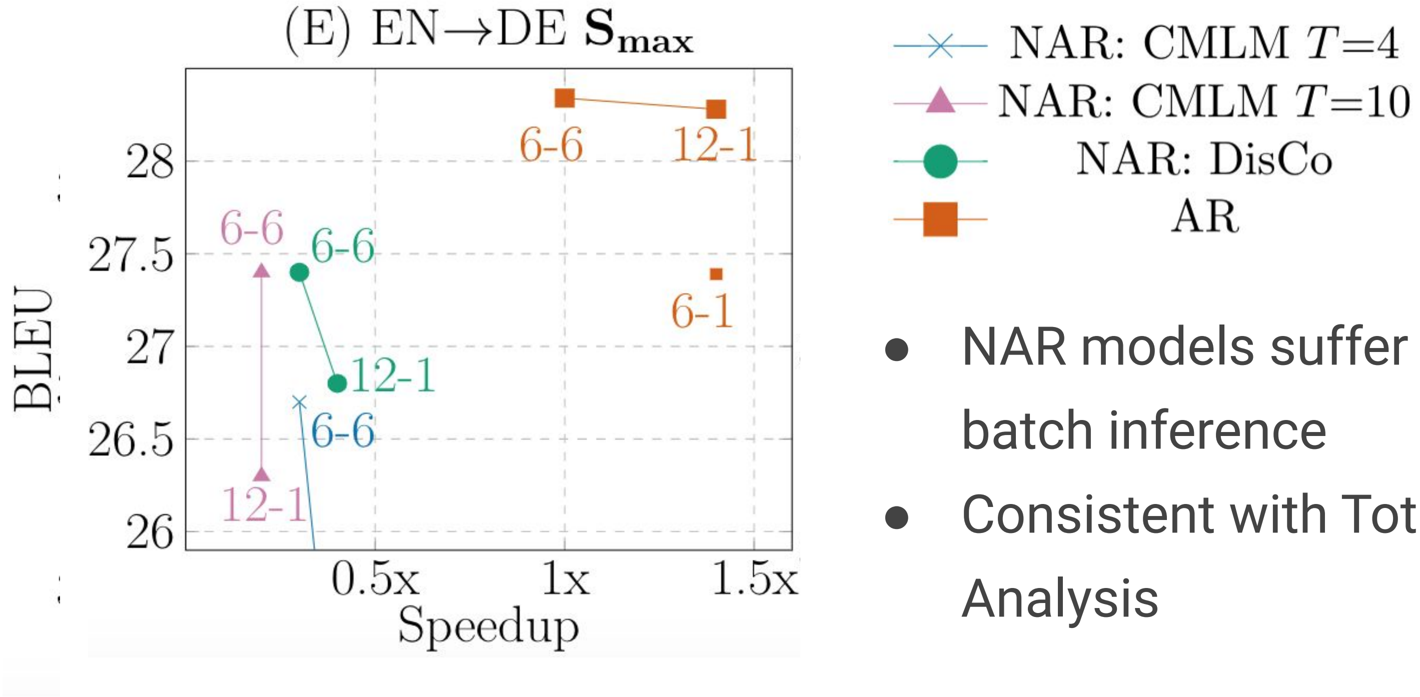


- Speedups wrt AR 6-6 Baseline
- NAR 12-1 models generally suffer in BLEU
- **Deep-Shallow not Effective for NAR**

Speed-Accuracy Tradeoff S1 More Langs.

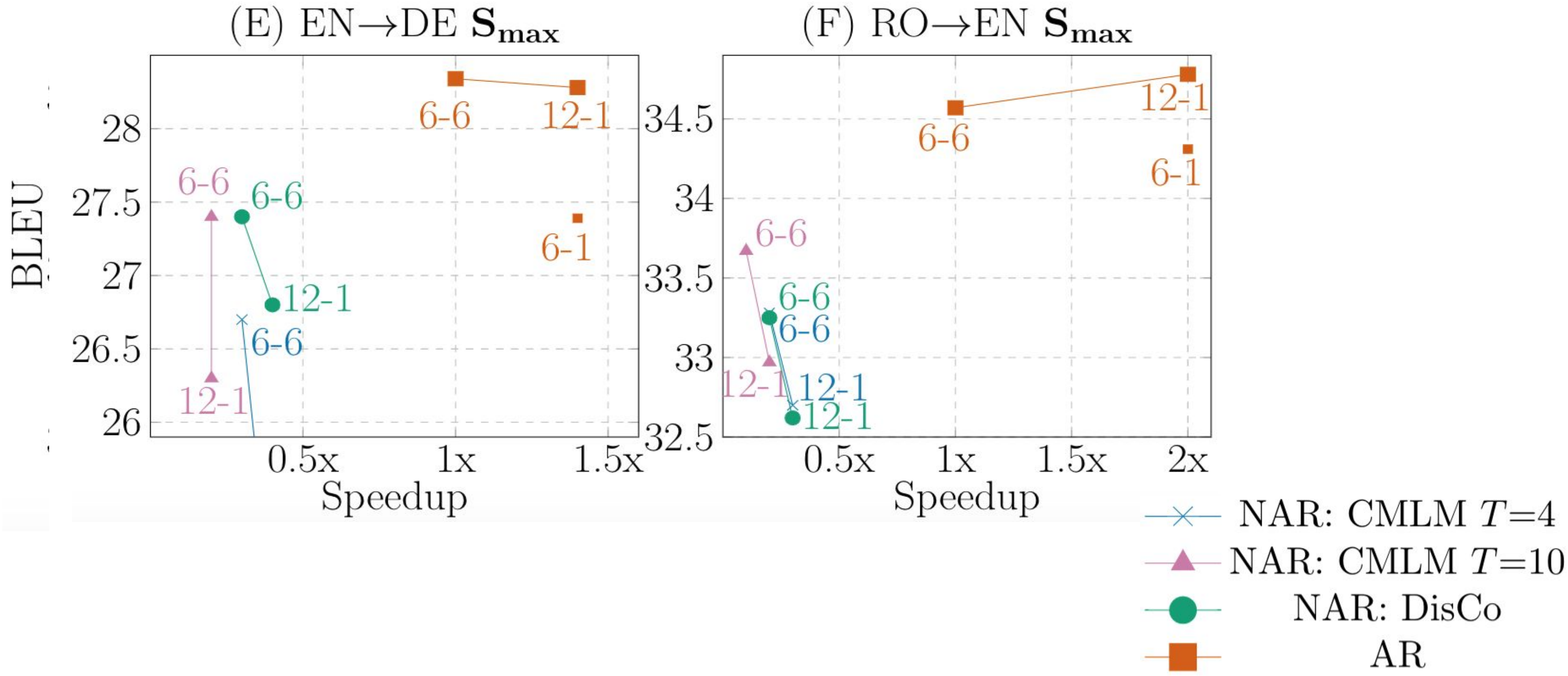


Speed-Accuracy Tradeoff Smax



- NAR models suffer in large batch inference
- Consistent with Total Operations Analysis

Speed-Accuracy Tradeoff Smax More Langs.



High-resource MT

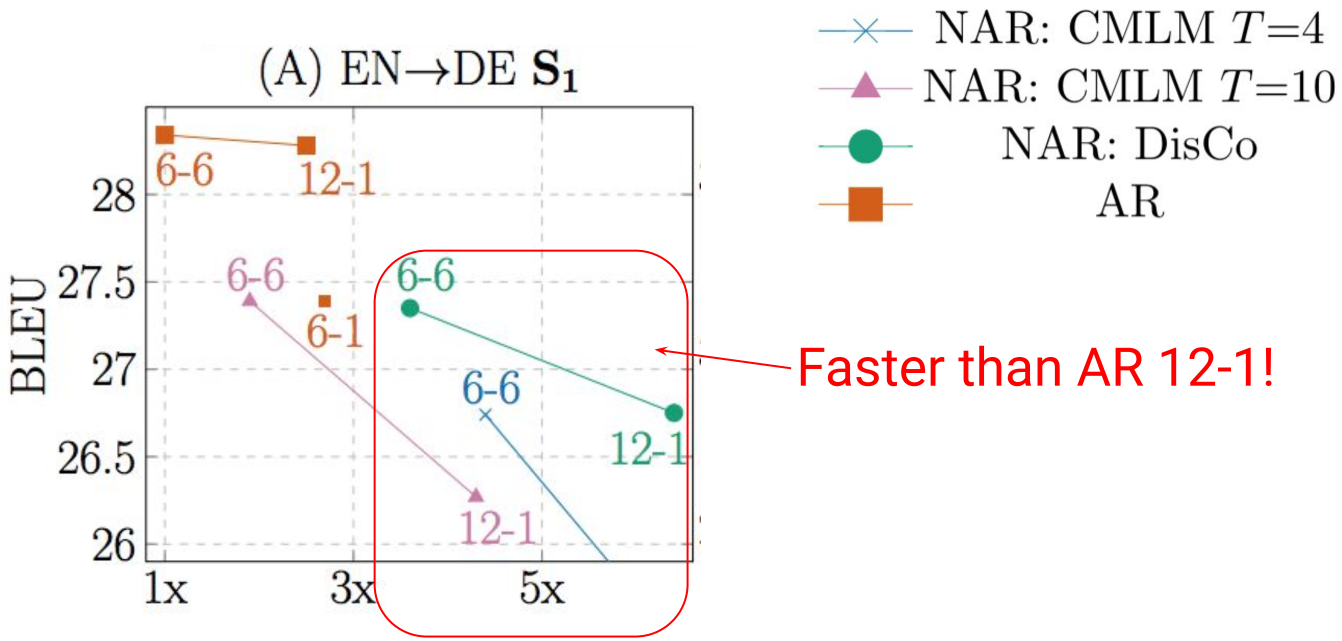
Model	T	E-D	WMT17 EN→ZH BLEU	WMT17 EN→ZH S_1	WMT17 EN→ZH S_{max}	WMT14 EN→FR BLEU	WMT14 EN→FR S_1	WMT14 EN→FR S_{max}
CMLM	4	6-6	33.58	**3.5×**	0.2×	40.21	**3.8×**	0.2×
CMLM	10	6-6	34.24	1.5×	0.1×	40.55	1.7×	0.1×
DisCo		6-6	34.63	2.5×	0.2×	40.60	3.6×	0.2×
AR Deep-Shallow		12-1	34.71	2.7×	**1.7×**	**42.04**	2.8×	**1.9×**
AR		6-6	**35.06**	1.0×	1.0×	41.98	1.0×	1.0×
Dist. Teacher		6-6	35.01	–	–	42.03	–	–

Compare with More NAR Works

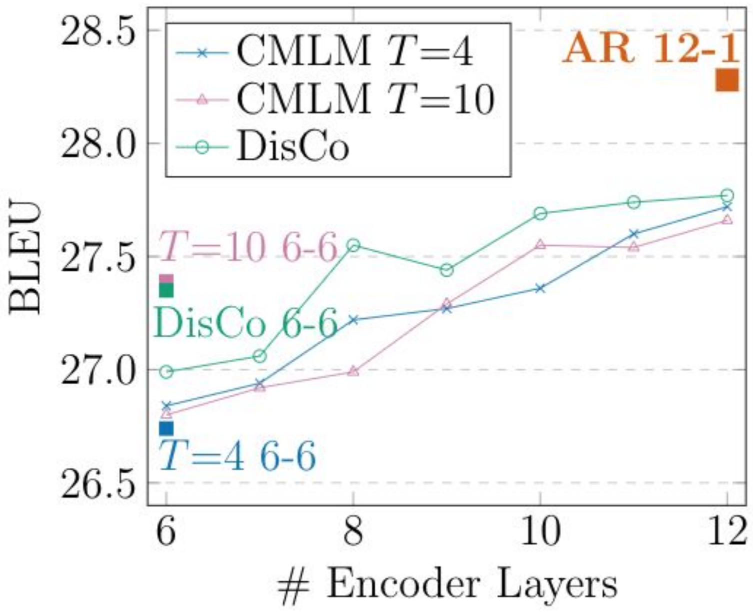
	EN-DE (BLEU)	Avg Iterative Steps
Levenshtein Transformer	27.3	>7
SMART	27.0	10
Imputer	28.0	4 (12-layer NN Only)
AR 6-6	**28.3**	N
AR Deep-Shallow (12-1)	**28.3**	N

Compare AR and NAR

S1 Speed Constraint

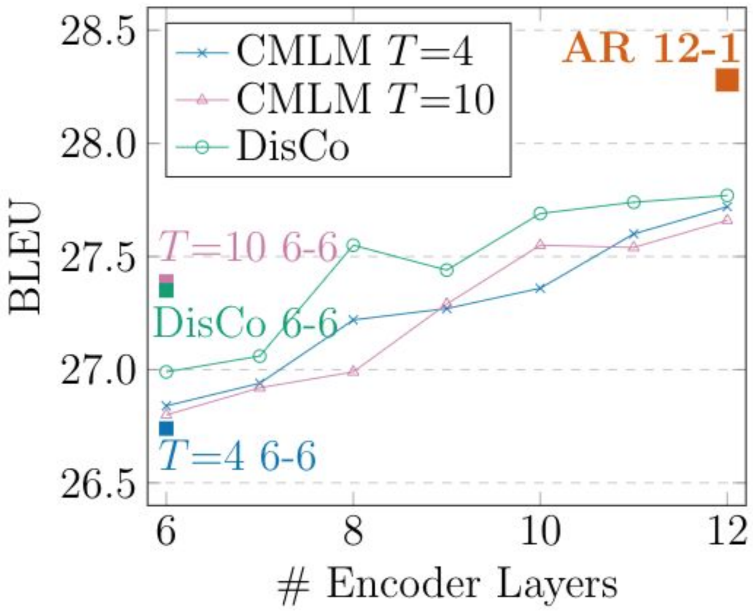


S1 Speed Constraint



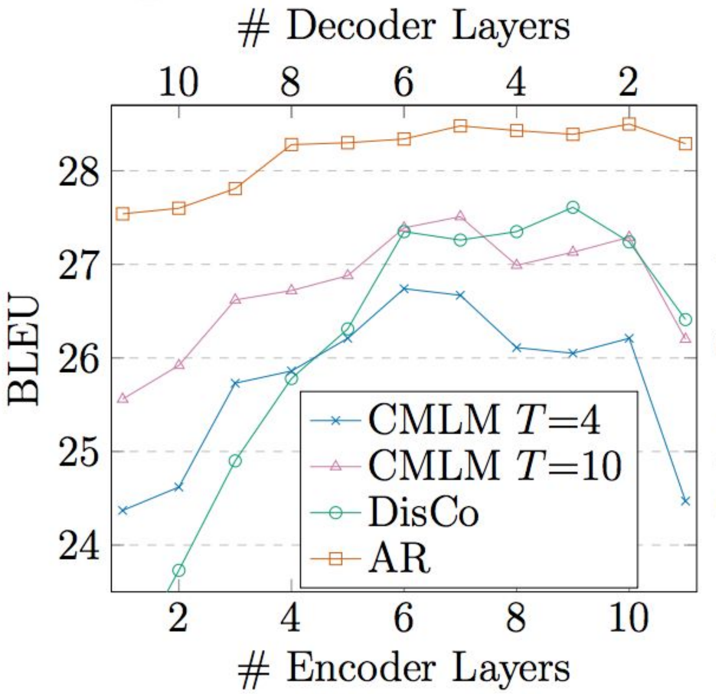
- WMT EN-DE Test
- Maximize Decoder Depth in the budget
  - E.g., DisCo 12-9

S1 Speed Constraint



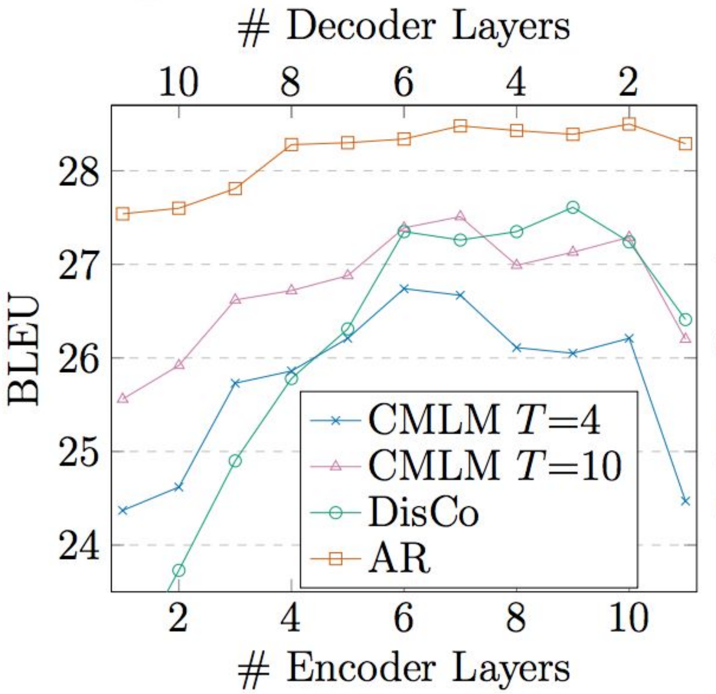
- WMT EN-DE Test
- Maximize Decoder Depth in the budget
  - E.g., DisCo 12-9
- Accuracy still far from AR 12-1 under the same S1 Budget

Total Layers Constraint



- WMT EN-DE Test
- E+D=12

Total Layers Constraint



- WMT EN-DE Test
- E+D=12
- AR: Stable
- NAR: decoders can't be too shallow

Why deep-shallow doesn’t work in NAR?

- Hypothesis: diverging word order btw the source and the target

Why deep-shallow doesn’t work in NAR?

- Hypothesis: diverging word order btw the source and the target
- Test: Reorder input English sentences to have monotonic alignment. Does it help?

Source: I like school -> **I school like**

私は　学校が　好きです

Why deep-shallow doesn't work in NAR?

- Applied fast align (Dyer et al., 2013) to WMT16 EN-DE Test

Model	***E-D***	**Orig.**
CMLM, $T = 10$	6-6	27.4
CMLM, $T = 10$	12-1	26.3
DisCo	6-6	27.4
DisCo	12-1	26.8
AR	6-6	**28.3**
AR Deep-Shallow	12-1	**28.3**

Why deep-shallow doesn't work in NAR?

- Applied fast align (Dyer et al., 2013) to WMT16 EN-DE Test

Model	E-D	Orig.	Reorder	Δ
CMLM, $T = 10$	6-6	27.4	31.7	4.3
CMLM, $T = 10$	12-1	26.3	31.0	4.7
DisCo	6-6	27.4	31.0	3.6
DisCo	12-1	26.8	31.6	**4.8**
AR	6-6	**28.3**	**32.6**	4.3
AR Deep-Shallow	12-1	**28.3**	**32.6**	4.3

Conclusion and Future Prospects

Conclusion

- AR’s speed-accuracy balance improves with deep-shallow configurations.

Conclusion

- AR’s speed-accuracy balance improves with deep-shallow configurations.
- Future work in NAR should consider layer allocation, knowledge distillation, and speed measurement.

Conclusion

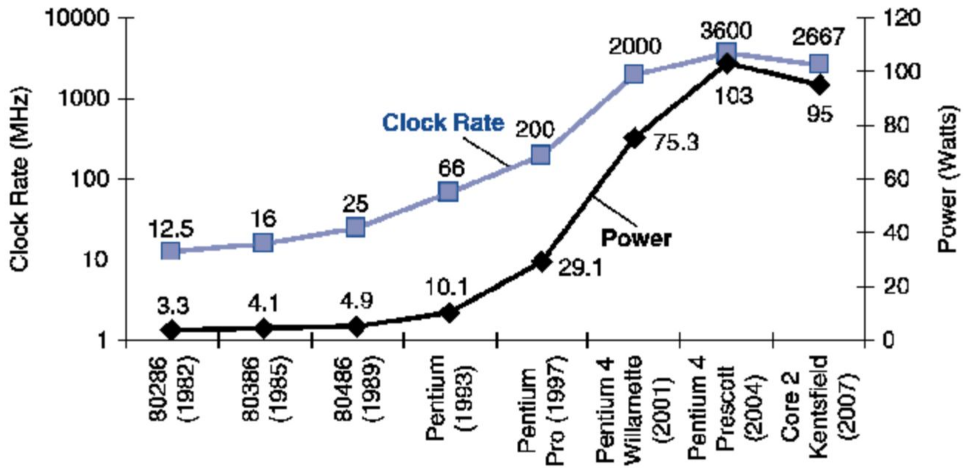
- AR's speed-accuracy balance improves with deep-shallow configurations.
- Future work in NAR should consider layer allocation, knowledge distillation, and speed measurement.
- Deep-shallow configurations for other seq2seq tasks? Seq2seq pretraining like T5 or BART?

Future Prospects of Fast, Accurate MT

- Should we still work on NAR?
  - Many other MT optimization methods

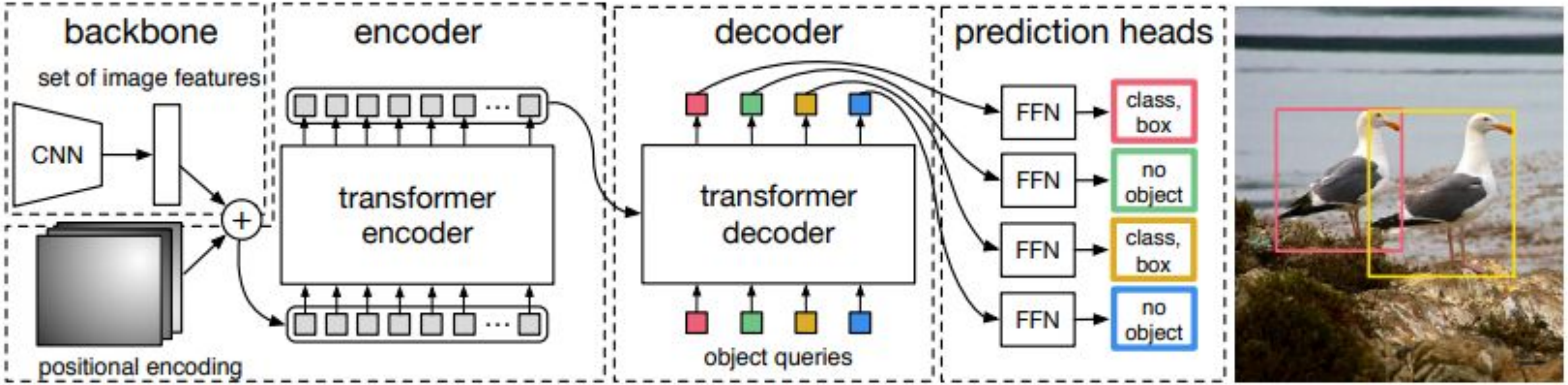
Future Prospects of Fast, Accurate MT

- Should we still work on NAR?
  - **Heat Wall** (Etiemble. 2018)
  - Parallelism is the future



Future Prospects of Fast, Accurate MT

- NAR Transformer Applications beyond MT
  - Speech/Image Generation (Parallel WaveNet, GANs)
  - Image Recognition (DETR)



Thank you!

https://github.com/jungokasai/deep-shallow