

CSCSE 638 Natural Language Processing Foundation and Techniques

Lecture 21: Non-Autoregressive Generation

Kuan-Hao Huang

Spring 2025



Invited Talk



- **Speaker:** [Pan Lu](#), Postdoctoral Scholar at Stanford University
- **Title:** Advancing Complex Reasoning with Language Models and Agentic Systems
- **Date:** 4/16
- **Online @ Zoom:**
 - <https://tamu.zoom.us/my/khhuang?pwd=oAdWOKVOCGPAPqDbJnVtktdW2AE6nb.1>

Invited Talk

Abstract: Complex reasoning is fundamental to human intelligence and plays a crucial role in advancing education, science, and technology. This talk explores the development of language model systems that exhibit robust mathematical reasoning and facilitate scientific reasoning, marking a significant step toward general artificial intelligence. We introduce novel multi-modal and knowledge-intensive benchmarks designed to assess the reasoning capabilities of large language models (LLMs) and vision-language models (VLMs) in real-world scenarios, including those involving visual data, tabular information, and scientific applications. The talk highlights recent advancements in mathematical reasoning within visual contexts and addresses key unresolved challenges. Additionally, we present cutting-edge retrieval and tool-augmented algorithms that significantly enhance LLM performance in mathematical reasoning tasks. Finally, we explore how agentic systems, leveraging test-time optimization and external tools, can further advance mathematical reasoning and scientific discovery.

Quiz 3

- Date: 4/14
 - 15 minutes before the end of the lecture
 - 5 questions focusing on high-level concepts
- Pay attention to how methods work and the difference between them

W10	3/17	L15	Multilingual NLP [slides]
	3/19	L16	Vision-Language Models [slides]
W11	3/24	L17	Adversarial Attack and Defense [slides]
	3/26	L18	AI-Generated Text Detection [slides]
W12	3/31		Invited Talk (Minhao Cheng)
	4/2	L19	Bias Detection and Mitigation [slides]
W13	4/7	L20	Hallucinations and Misinformation Control
	4/9	L21	Controlled Generation, Non-Autoregressive Generation

Recap: Autoregressive Language Models

$$\begin{aligned}P(w_1, w_2, w_3, \dots, w_l) &= P(w_1)P(w_2, w_3, \dots, w_l|w_1) \\&= P(w_1)P(w_2|w_1)(w_3, \dots, w_l|w_1, w_2) \\&= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)(w_4, \dots, w_l|w_1, w_2, w_3) \\&= \prod_{i=1}^l P(w_i|w_1, w_2, \dots, w_{i-1})\end{aligned}$$

$$\begin{aligned}P(\textit{She likes to go hiking}) &= P(\textit{She}) \cdot P(\textit{likes}|\textit{She}) \cdot P(\textit{to}|\textit{She likes}) \\&\quad \cdot P(\textit{go}|\textit{She likes to}) \cdot P(\textit{hiking}|\textit{She likes to go})\end{aligned}$$

Recap: Autoregressive Language Models

Binge ... on | - | and | of | is

Binge **drinking** ... is | and | had | in | was

Binge drinking **may** ... be | also | have | not | increase

Binge drinking may **not** ... be | have | cause | always | help

Binge drinking may not **necessarily** ... be | lead | cause | results | have

Binge drinking may not necessarily **kill** ... you | the | a | people | your

Binge drinking may not necessarily kill **or** ... even | injure | kill | cause | prevent

Binge drinking may not necessarily kill or **even** ... kill | prevent | cause | reduce | injure

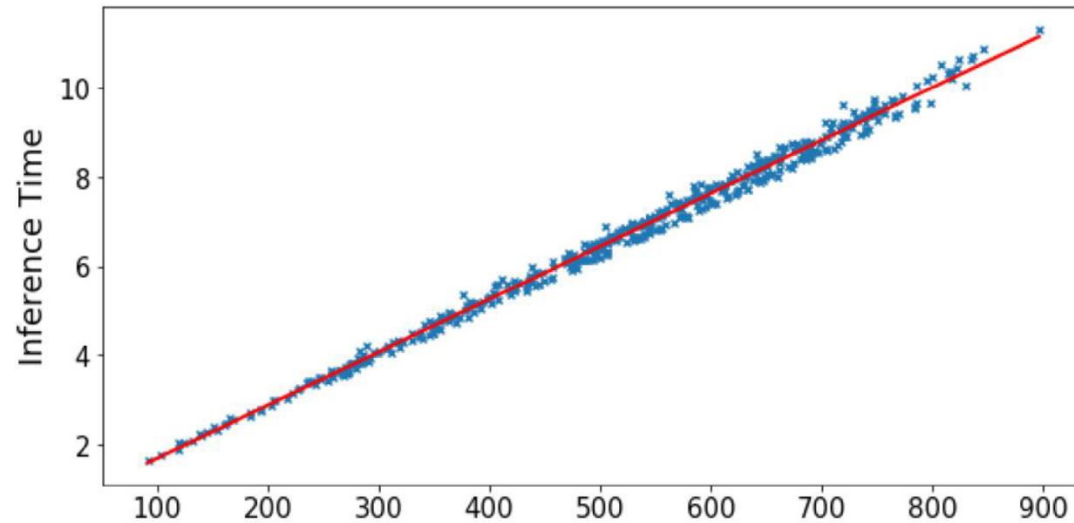
Binge drinking may not necessarily kill or even **damage** ... your | the | a | you | someone

Binge drinking may not necessarily kill or even damage **brain** ... cells | functions | tissue | neurons

Binge drinking may not necessarily kill or even damage brain **cells,** ... some | it | the | is | long

Non-Autoregressive Generation

- Can we generate text in ways other than word by word?
 - Other order
 - Parallel decoding
- Why?
 - Faster decoding



Non-Autoregressive Generation

- Can we generate text in ways other than word by word?
 - Other order
 - Parallel decoding
- Why?
 - Faster decoding
 - More similar to human writing (revision)

There is a bird on the tree.

There is a red bird on the green tree.

There is a red bird with a big beak on the green tree.

Document Editing

INTRODUCTION

Several initiatives have been launched in many countries with the aim of modernizing ~~the~~ public services. In this sense different reports and documents have indicated the need for investing in technologies ~~for to~~ offering better services to citizens and organizations (Department of Public Expenditure and Reform, 2011; United Nations, 2012) and thus ~~reduc~~ing the burden for them. This trend is named “electronic ~~g~~Government (e-~~g~~Government).” and it can be defined as the “the use of information technology to enable and improve the efficiency with which government services are provided to citizens, employees, businesses, and agencies” (Carter & Belanger, 2005).

Although several countries have improved their services through the use of more sophisticated ~~web~~ Web pages, according to the Department of Public Expenditure and Reform (2012), countries still need to introduce more technologies and automated processes with the aim of reducing the burden of processes currently needed for citizens and organizations. In several public processes the citizens and organization ~~must have to~~ present physical documents ~~which that~~ are delivered manually from one section to ~~an~~other section, ~~sometimes~~ producing ~~sometimes~~ delays in the delivery due to human causes such as illnesses, oversight, or overwork of the public worker. By means of automated processes, this documentation is immediately available to the next section or administration in the business process once the documentation has been analyzed and completed by the corresponding section or administration. In this sense public bodies ~~must have to~~ ensure that the sharing of data between different public organizations provides a reduction ~~of~~ ~~in~~ the number of times citizens or businesses ~~must have to~~ ask for data.

Challenges

- How to add words in the middle?
- Parallel decoding → fluency issue

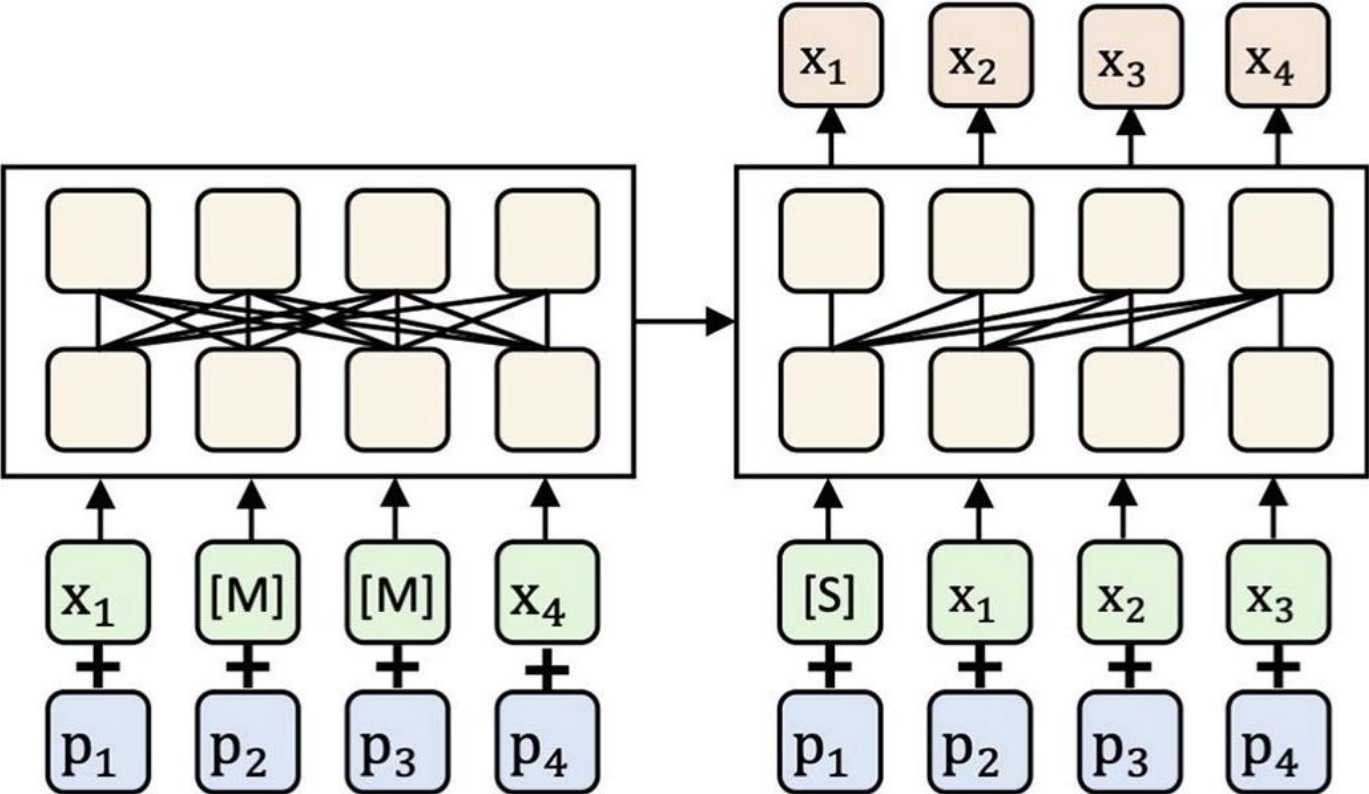
Semi-Autoregressive Neural Machine Translation

Chunqi Wang* **Ji Zhang** **Haiqing Chen**

Alibaba Group

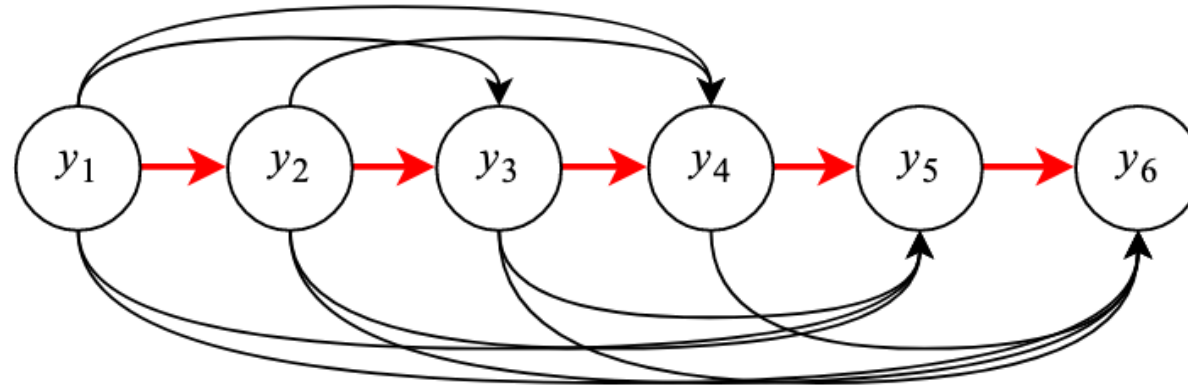
{shiyuan.wcq, zj122146, haiqing.chenhq}@alibaba-inc.com

Machine Translation with Seq2Seq

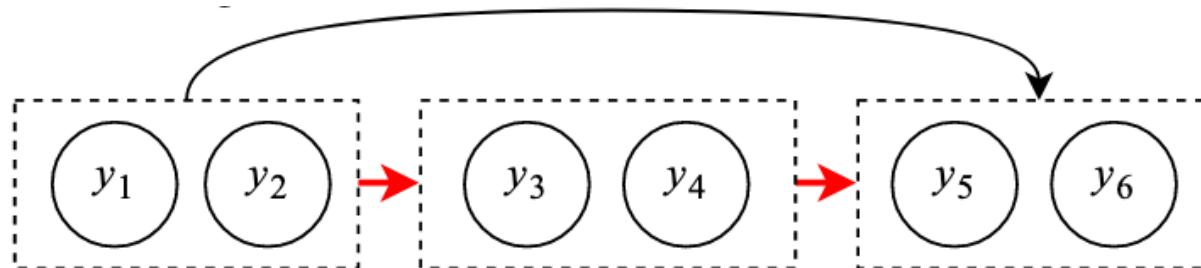


Semi-Autoregressive Decoding

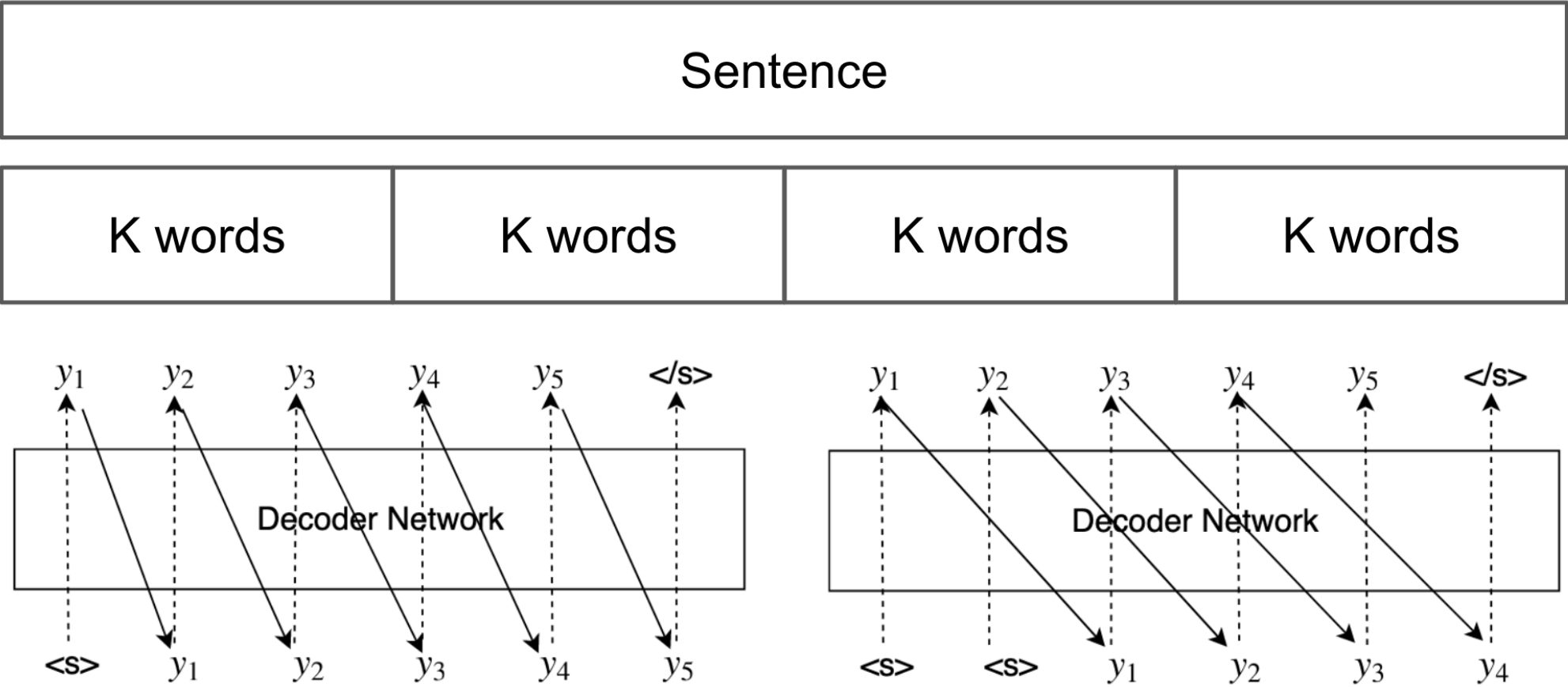
- Autoregressive decoding



- Semi-autoregressive decoding



Semi-Autoregressive Transformer



Mask Modification

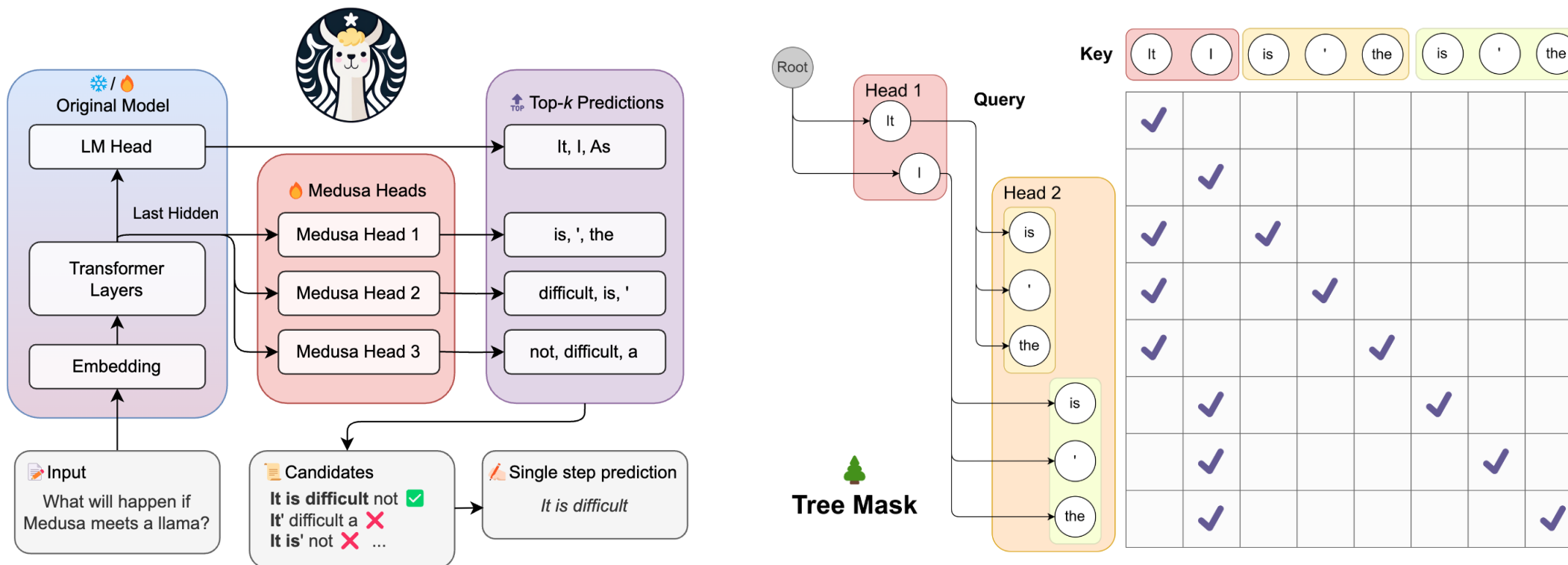
$$\begin{bmatrix} 1 & \mathbf{0} & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & \mathbf{0} & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & \mathbf{0} \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & \mathbf{1} & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & \mathbf{1} & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & \mathbf{1} \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Machine Translation Performance

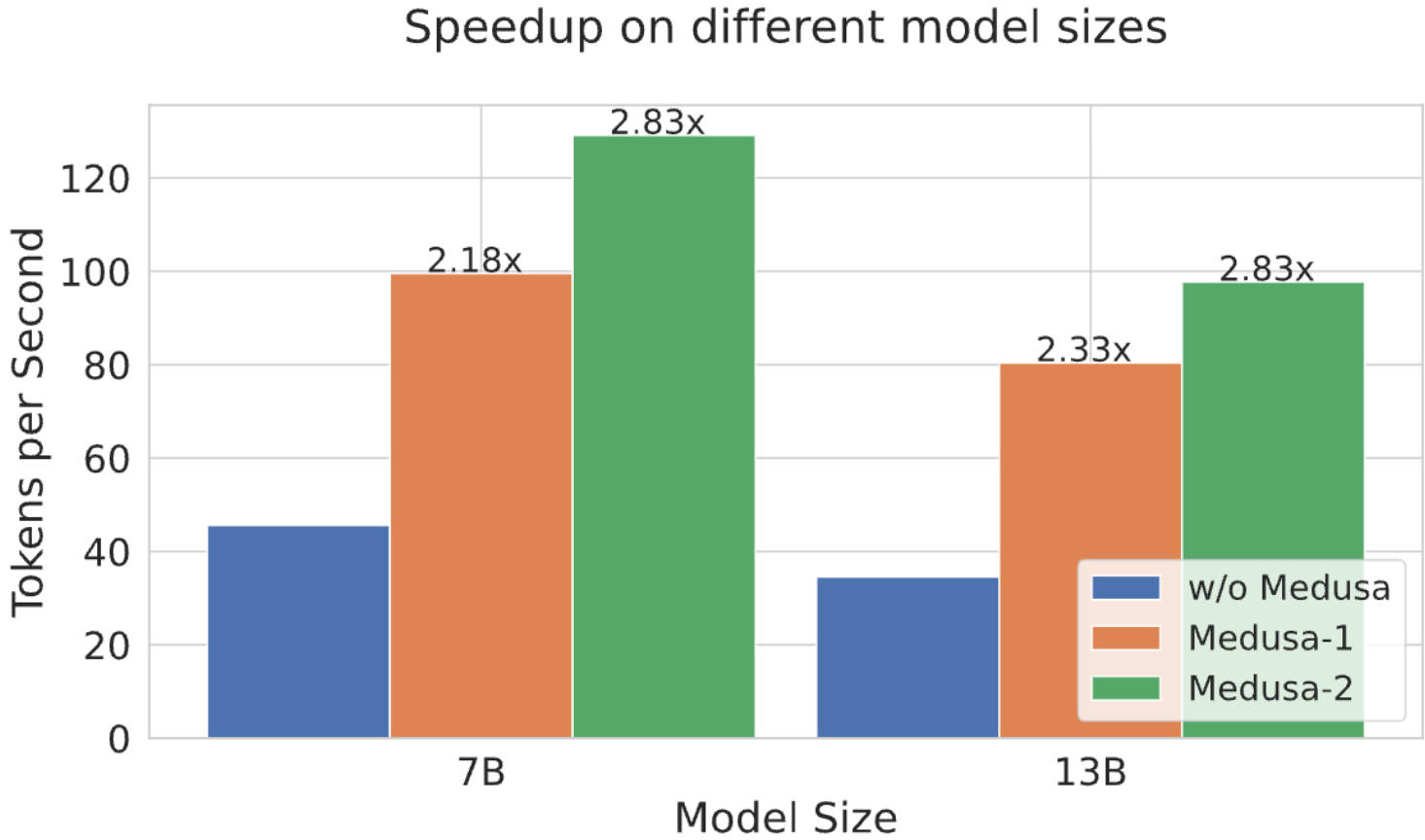
Model	Beam Size	BLEU	Degeneration	Latency	Speedup
Transformer	4	27.11	0%	346ms	1.00×
	1	26.01	4%	283ms	1.22×
Transformer, $N=2$	4	24.30	10%	163ms	2.12×
	1	23.37	14%	113ms	3.06×
<i>This Work</i>					
SAT, $K=2$	4	26.90	1%	229ms	1.51×
	1	26.09	4%	167ms	2.07×
SAT, $K=4$	4	25.71	5%	149ms	2.32×
	1	24.67	9%	91ms	3.80×
SAT, $K=6$	4	24.83	8%	116ms	2.98×
	1	23.93	12%	62ms	5.58×

MEDUSA: Simple LLM Inference Acceleration Framework with Multiple Decoding Heads

Tianle Cai^{*1,2} Yuhong Li^{*3} Zhengyang Geng⁴ Hongwu Peng⁵ Jason D. Lee¹ Deming Chen³ Tri Dao^{1,2}



Decoding Speed



NON-AUTOREGRESSIVE NEURAL MACHINE TRANSLATION

Jiatao Gu^{†*}, James Bradbury[‡], Caiming Xiong[‡], Victor O.K. Li[†] & Richard Socher[‡]

[‡]Salesforce Research

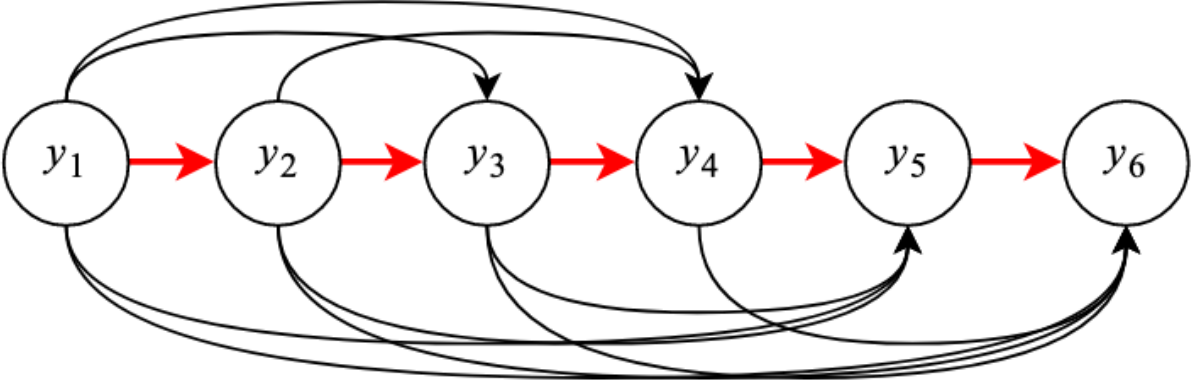
{james.bradbury, cxiong, rsocher}@salesforce.com

[†]The University of Hong Kong

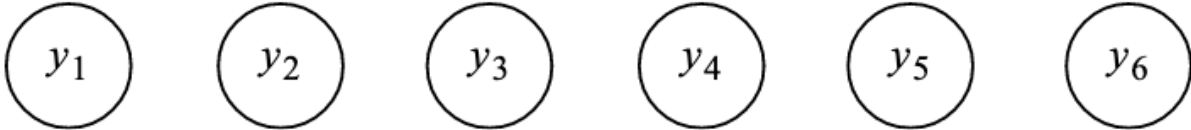
{jiataogu, vli}@eee.hku.hk

Fully Non-Autoregressive Decoding

- Autoregressive decoding

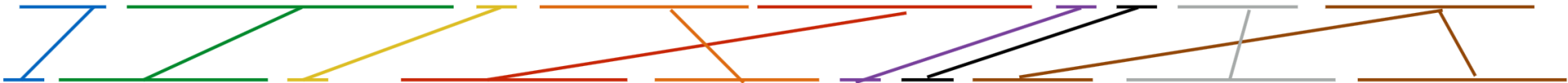


- Fully non-autoregressive decoding



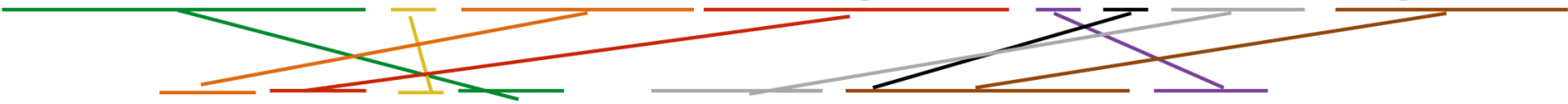
Motivation (Word Alignment)

The development of artificial intelligence is a really big deal.



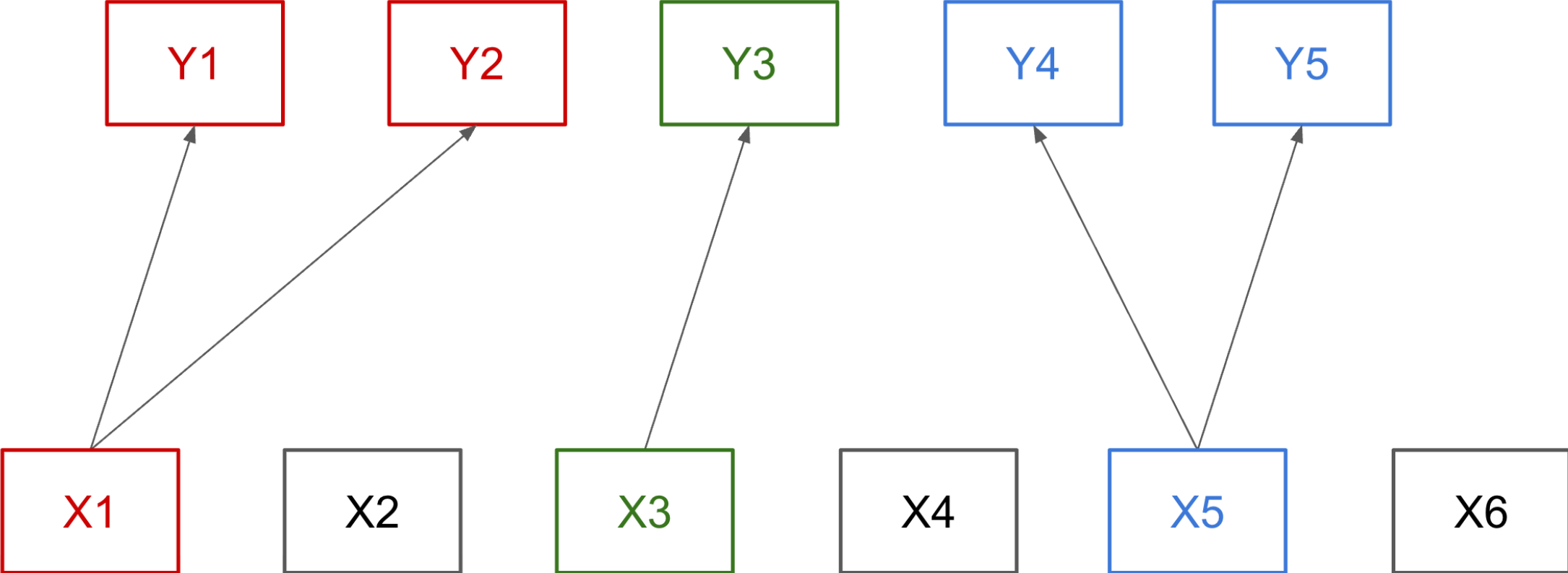
El desarrollo de la inteligencia artificial es un asunto realmente importante.

The development of artificial intelligence is a really big deal.

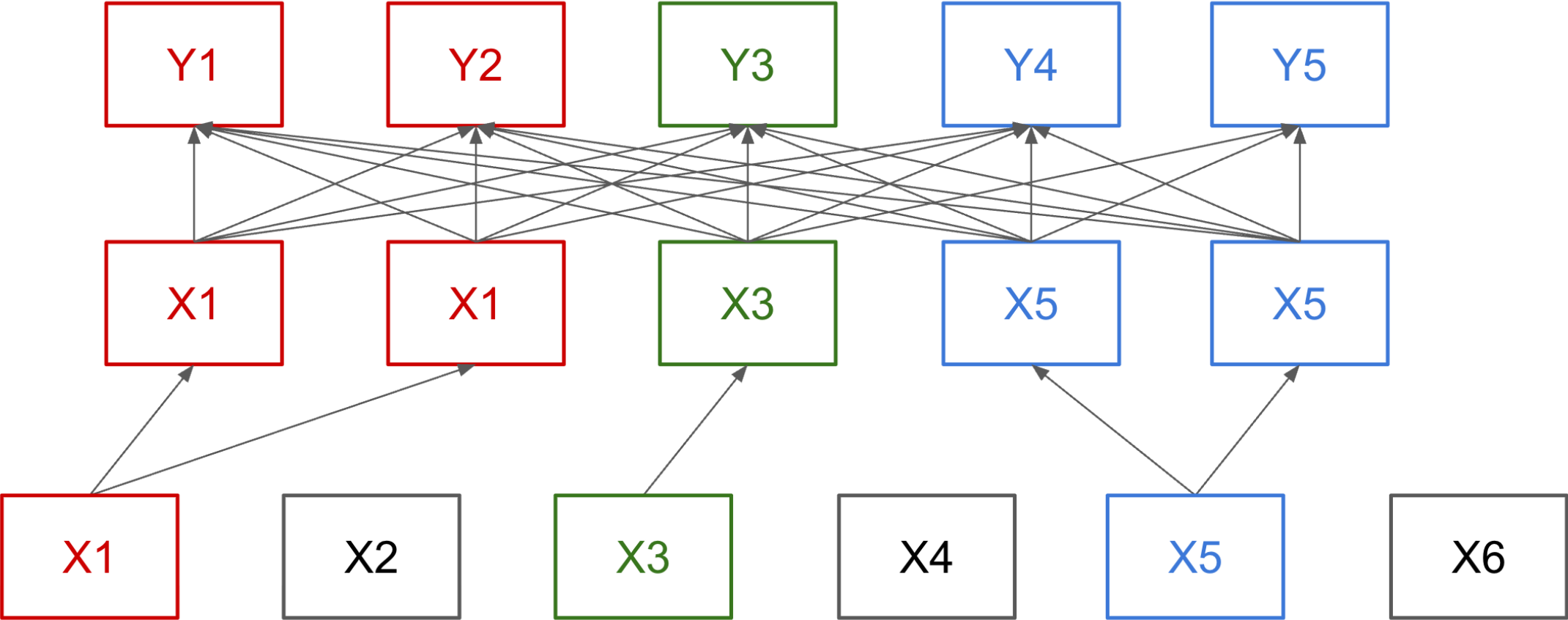


人工知能の発展は本当にすごいことです。

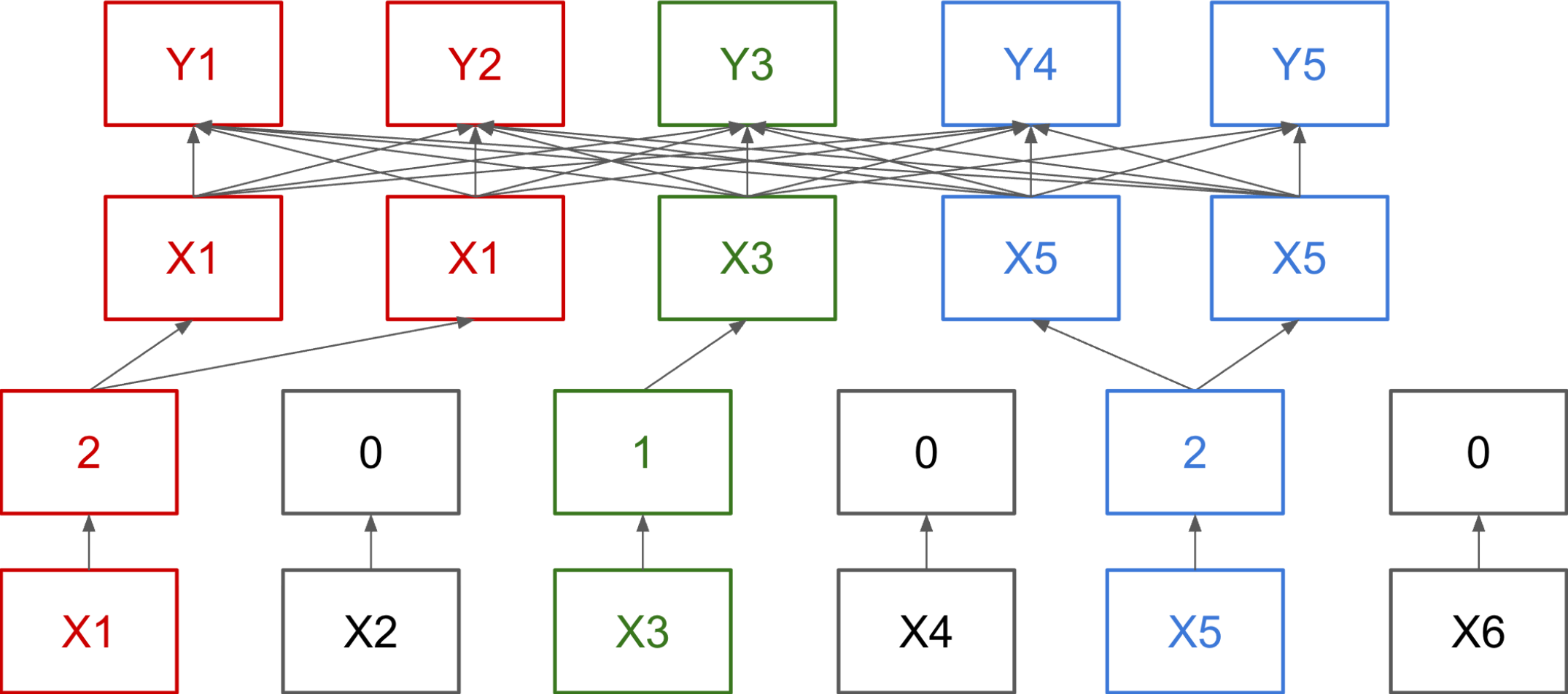
Motivation (Word Alignment)



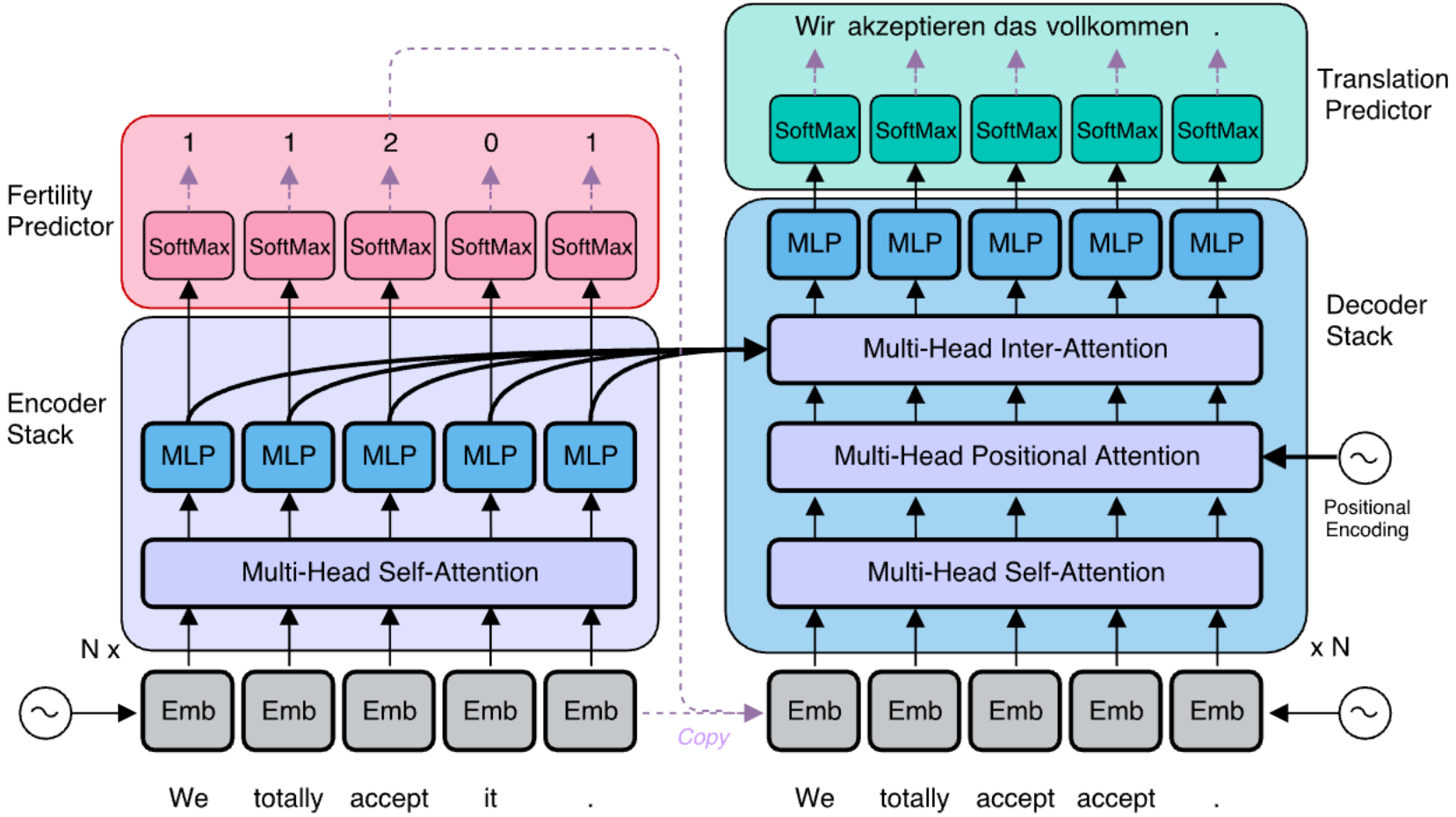
Motivation (Word Alignment)



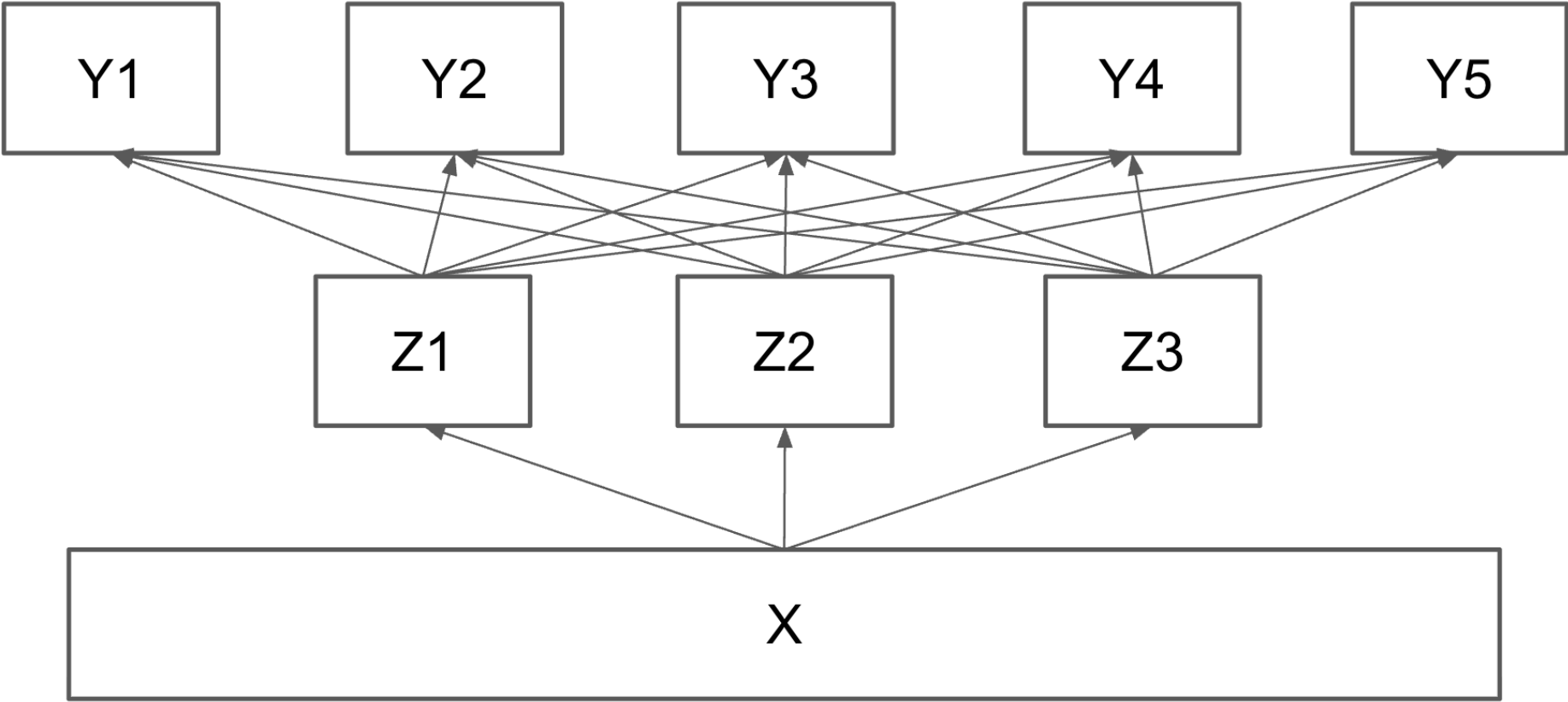
Fertility Predictor



Fertility Predictor



Decode with Latent Variables (A More General Version)



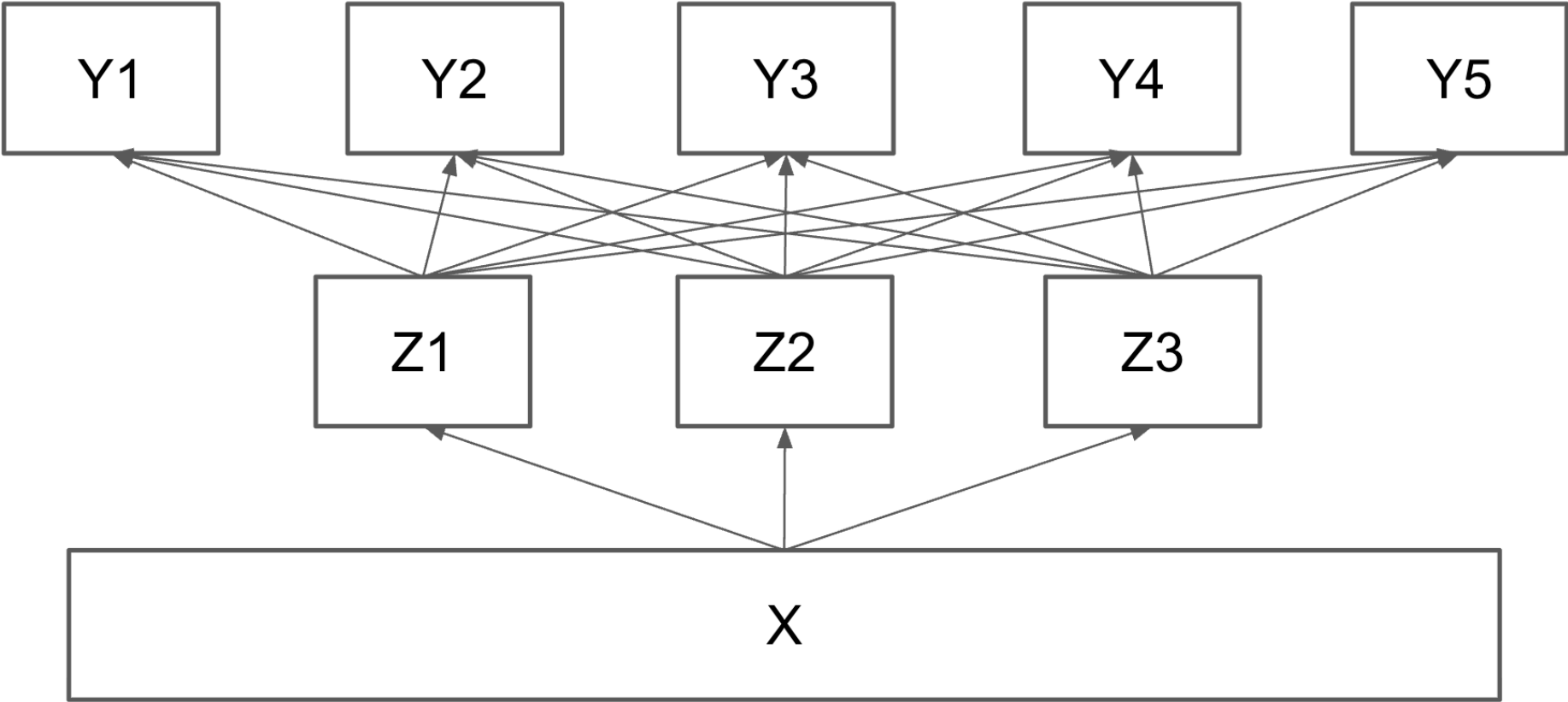
Results

Models	WMT14		WMT16		IWSLT16		
	En→De	De→En	En→Ro	Ro→En	En→De	Latency / Speedup	
NAT	17.35	20.62	26.22	27.83	25.20	39 ms	15.6×
NAT (+FT)	17.69	21.47	27.29	29.06	26.52	39 ms	15.6×
NAT (+FT + NPD $s = 10$)	18.66	22.41	29.02	30.76	27.44	79 ms	7.68×
NAT (+FT + NPD $s = 100$)	19.17	23.20	29.79	31.44	28.16	257 ms	2.36×
Autoregressive ($b = 1$)	22.71	26.39	31.35	31.03	28.89	408 ms	1.49×
Autoregressive ($b = 4$)	23.45	27.02	31.91	31.76	29.70	607 ms	1.00×

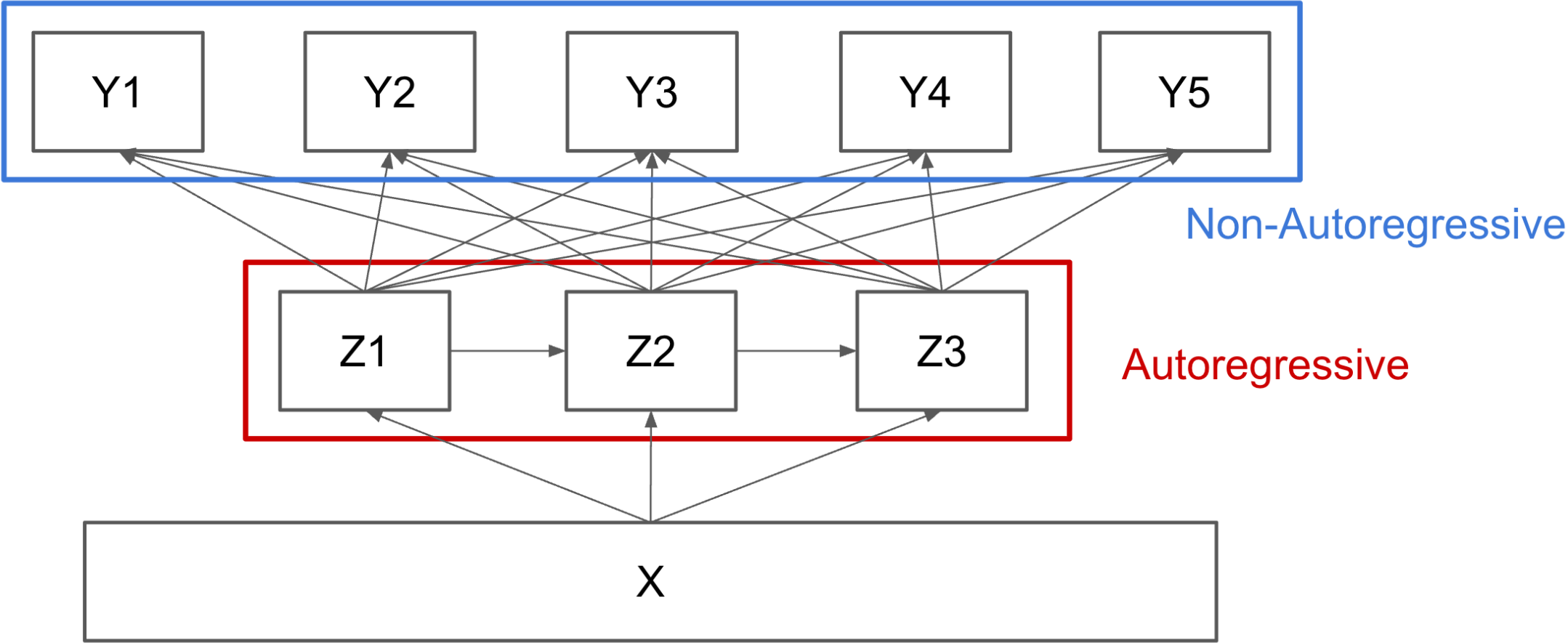
Fast Decoding in Sequence Models Using Discrete Latent Variables

**Łukasz Kaiser¹ Aurko Roy¹ Ashish Vaswani¹ Niki Parmar¹ Samy Bengio¹ Jakob Uszkoreit¹
Noam Shazeer¹**

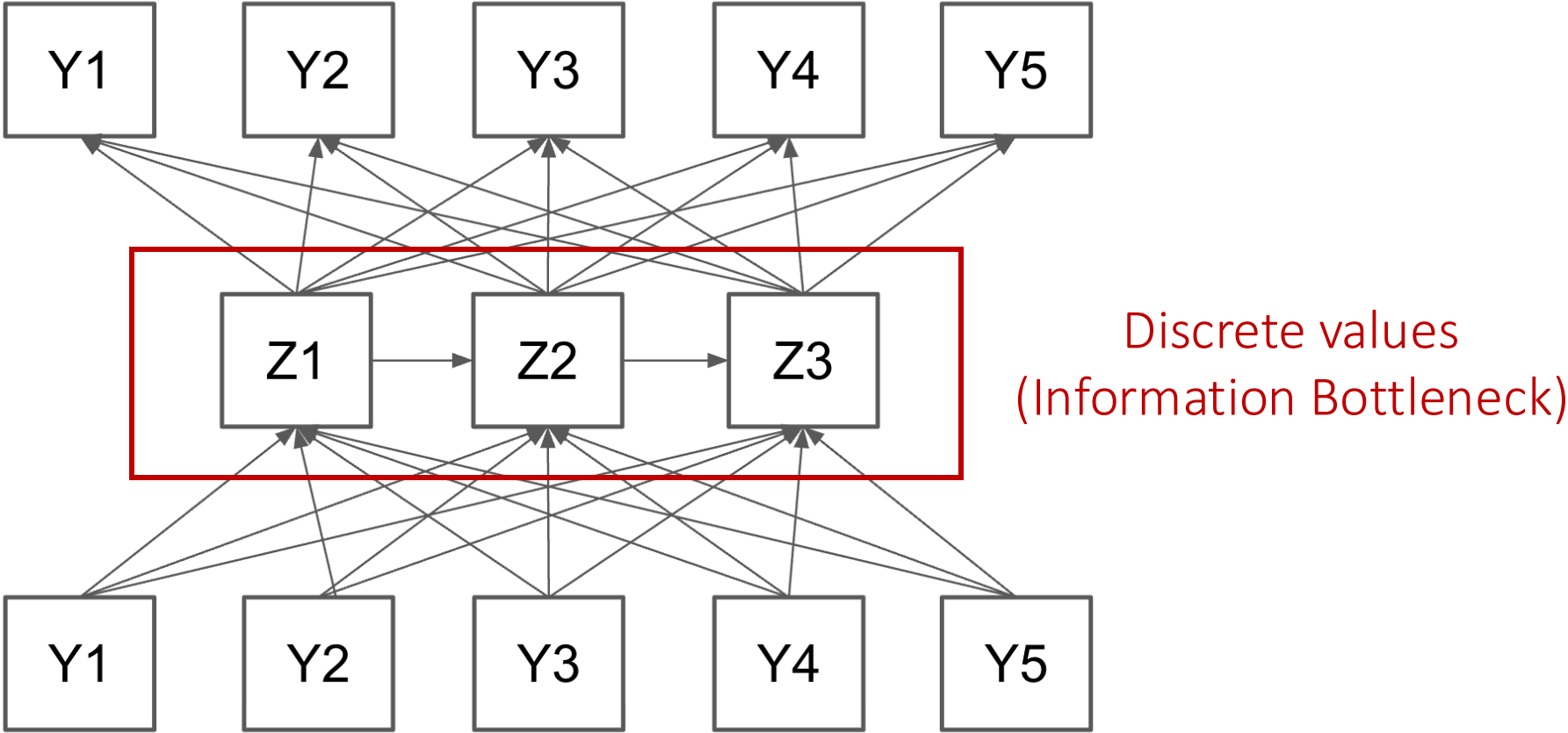
Decode with Latent Variables



Decode with Autoregressive Latent Variables

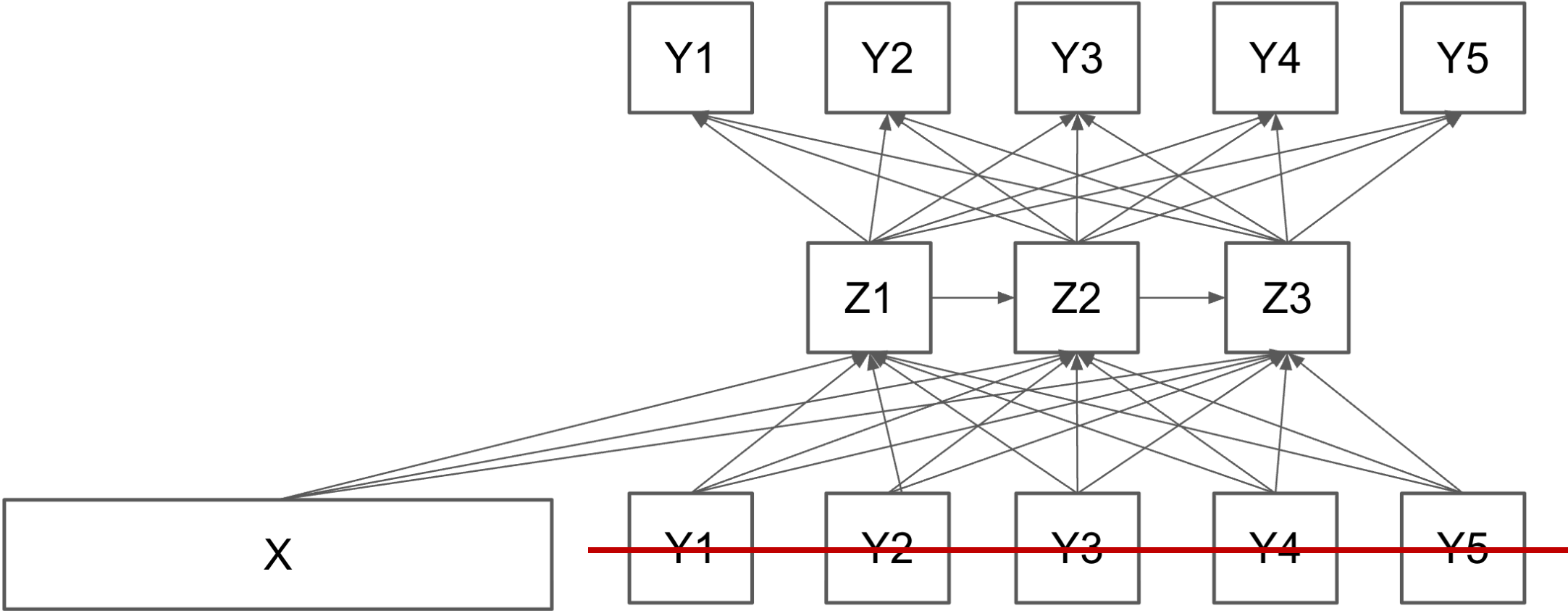


How to Decide Latent Variables?



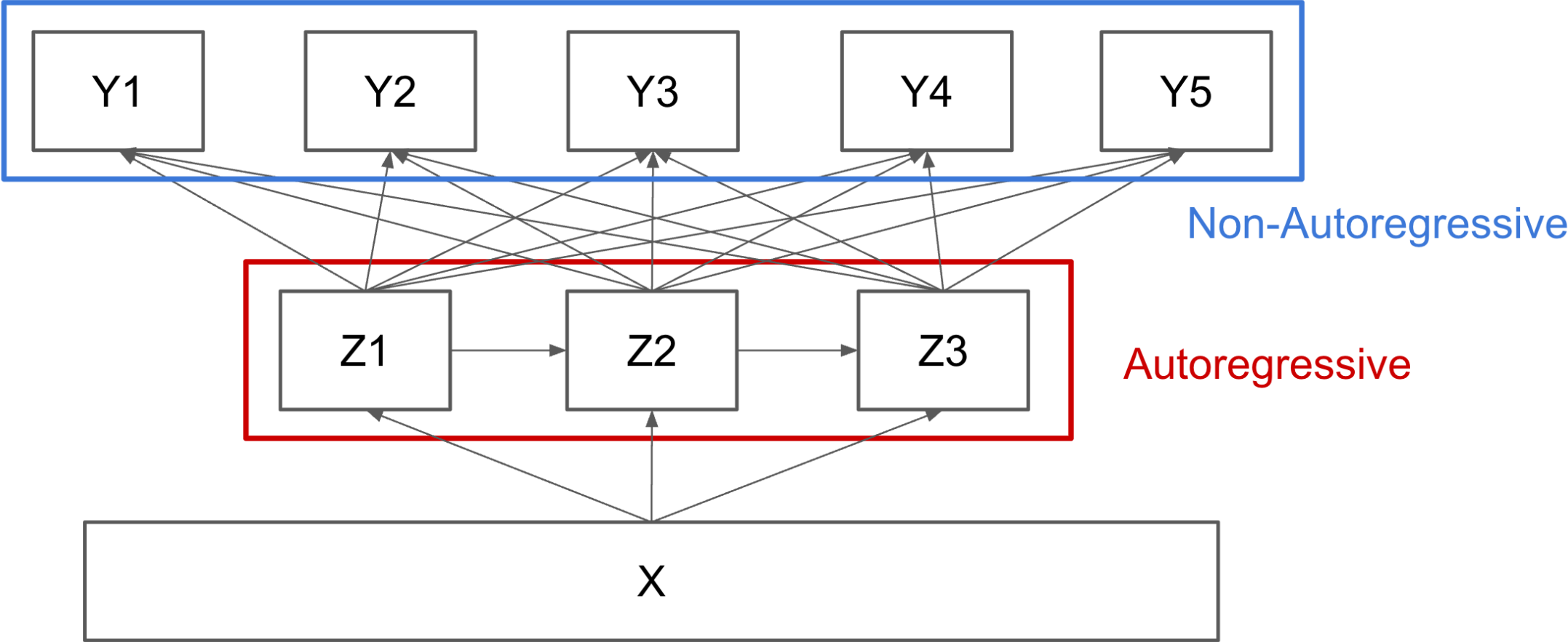
Reconstruction from latent variables

Learning Mapping Between Inputs and Latent Variables



Predict Output Length

A separate classifier to predict the output length



Results

Model	BLEU	Latency	
		$b = 1$	$b = 64$
Baseline (no beam-search)	22.7	408 ms	-
NAT	17.7	39 ms	-
NAT+NPD=10	18.7	79 ms	-
NAT+NPD=100	19.2	257 ms	-
LT, Improved Semhash	19.8	105 ms	8 ms
LT, VQ-VAE	2.78	148 ms	7 ms
LT, s-DVQ	19.7	177 ms	7 ms
LT, p-DVQ	19.8	182 ms	8 ms

Syntactically Supervised Transformers for Faster Neural Machine Translation

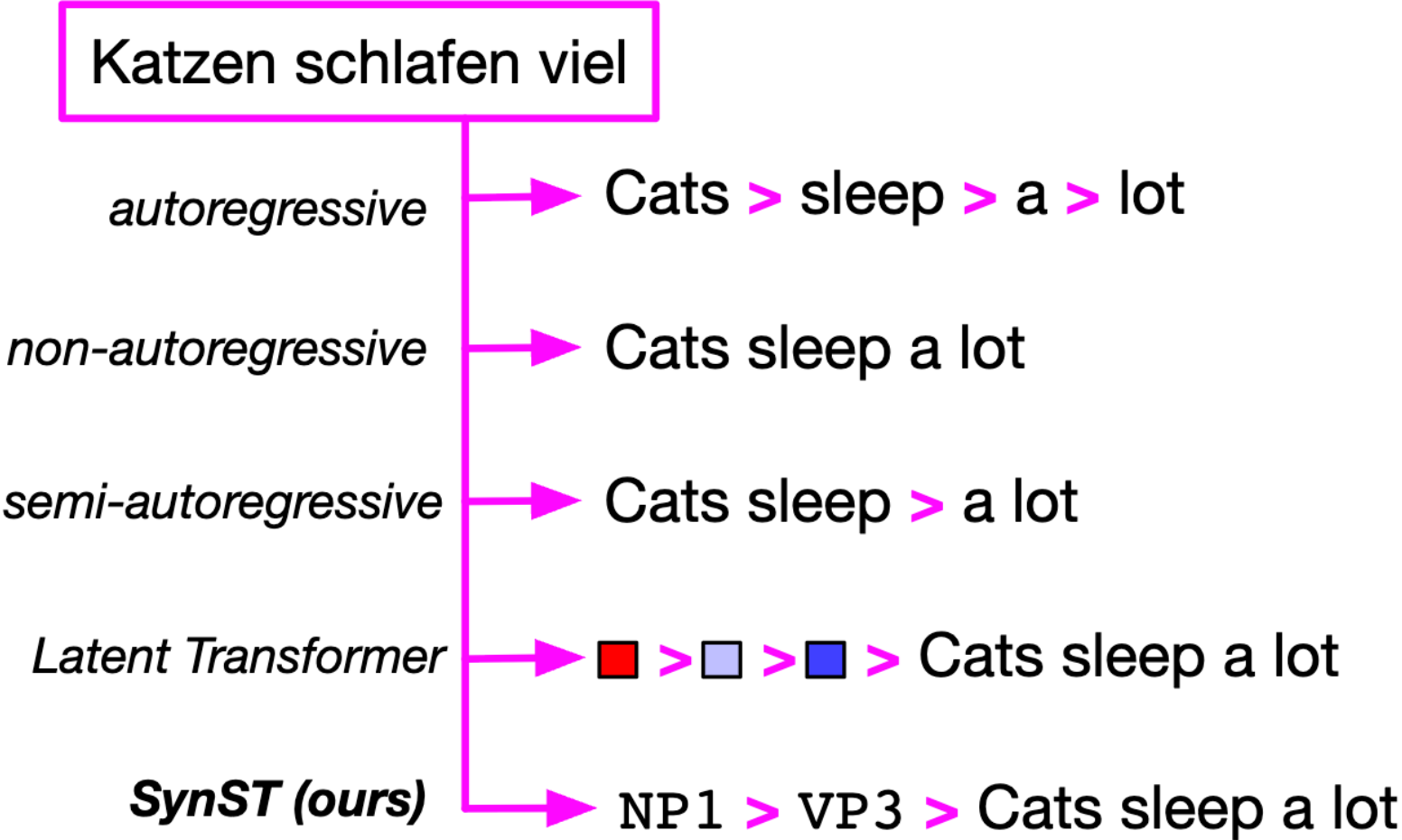
Nader Akoury, Kalpesh Krishna, Mohit Iyyer

College of Information and Computer Sciences

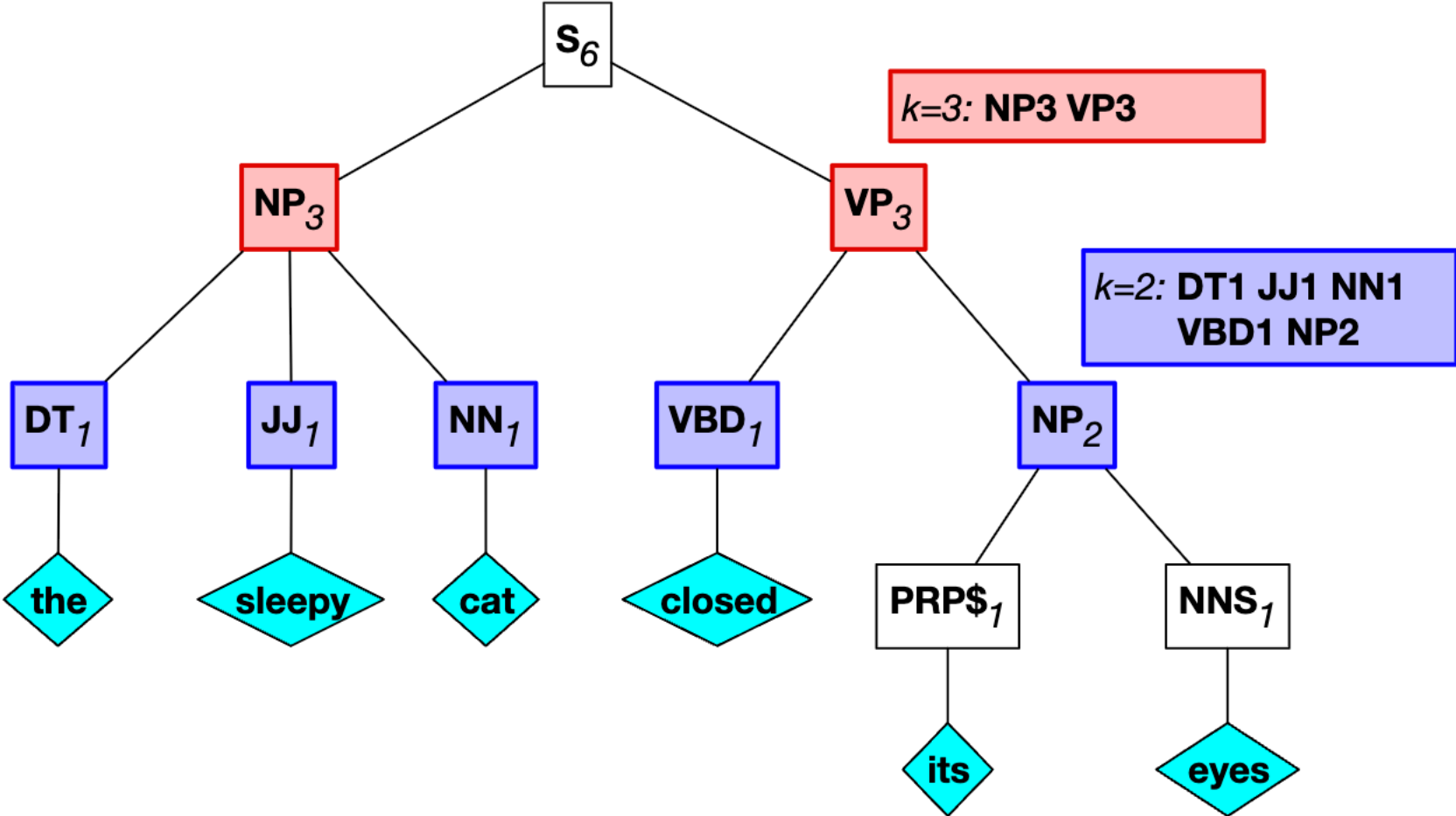
University of Massachusetts Amherst

`{nsa, kalpesh, miyyer}@cs.umass.edu`

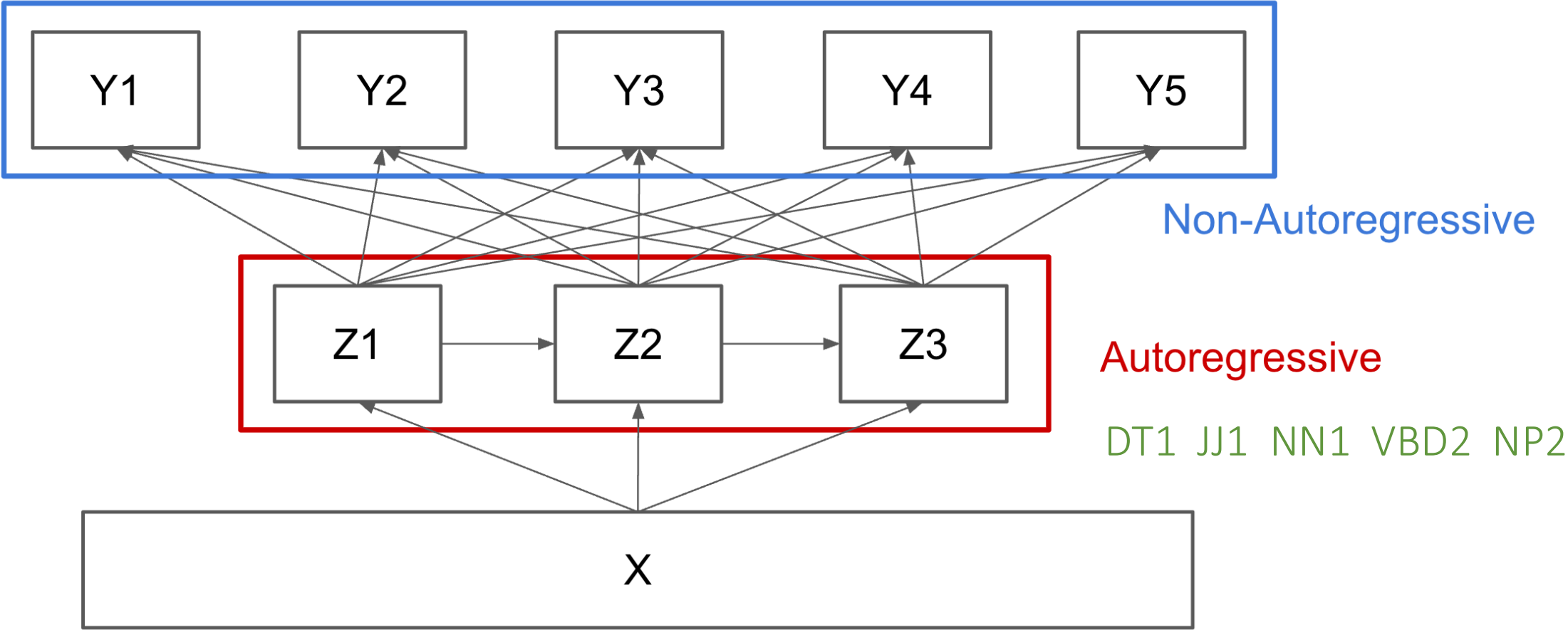
Consider Syntactic Information for Latent Variables



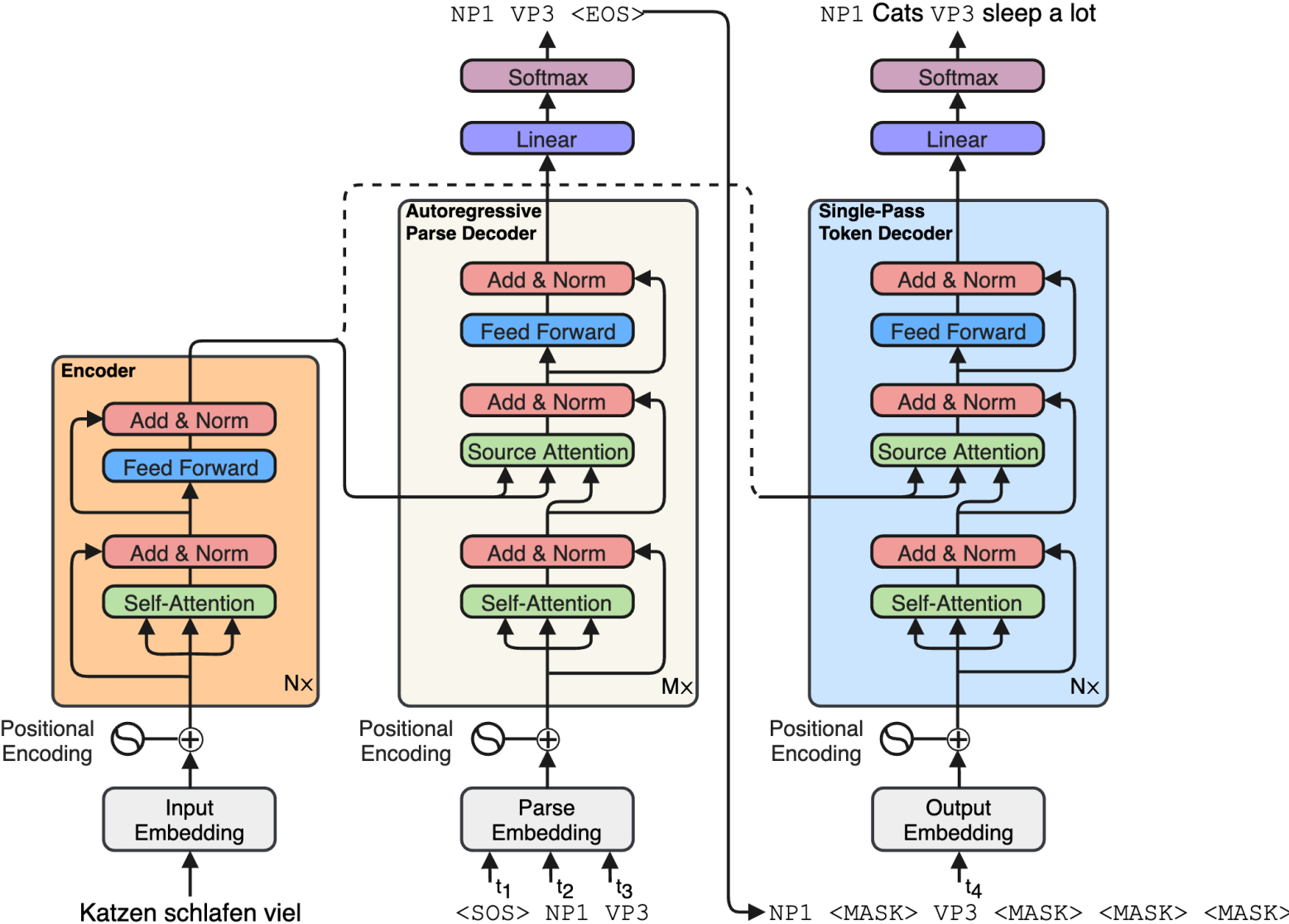
Syntactic Information: Constituency Parse Trees



Consider Syntactic Information for Latent Variables



Syntactically Supervised Transformer (SynST)



Results

Model	<u>WMT En-De</u>		<u>WMT De-En</u>		<u>IWSLT En-De</u>		<u>WMT En-Fr</u>	
	BLEU	Speedup	BLEU	Speedup	BLEU	Speedup	BLEU	Speedup
Baseline ($b = 1$)	25.82	1.15×	29.83	1.14×	28.66	1.16×	39.41	1.18×
Baseline ($b = 4$)	26.87	1.00×	30.73	1.00×	30.00	1.00×	40.22	1.00×
SAT ($k = 2$)	22.81	2.05×	26.78	2.04×	25.48	2.03×	36.62	2.14×
SAT ($k = 4$)	16.44	3.61×	21.27	3.58×	20.25	3.45×	28.07	3.34×
SAT ($k = 6$)	12.55	4.86×	15.23	4.27×	14.02	4.39×	24.63	4.77×
LT*	19.8	3.89×	-	-	-	-	-	-
SynST($k = 6$)	20.74	4.86×	25.50	5.06×	23.82	3.78×	33.47	5.32×

Mask-Predict: Parallel Decoding of Conditional Masked Language Models

Marjan Ghazvininejad*

Omer Levy*

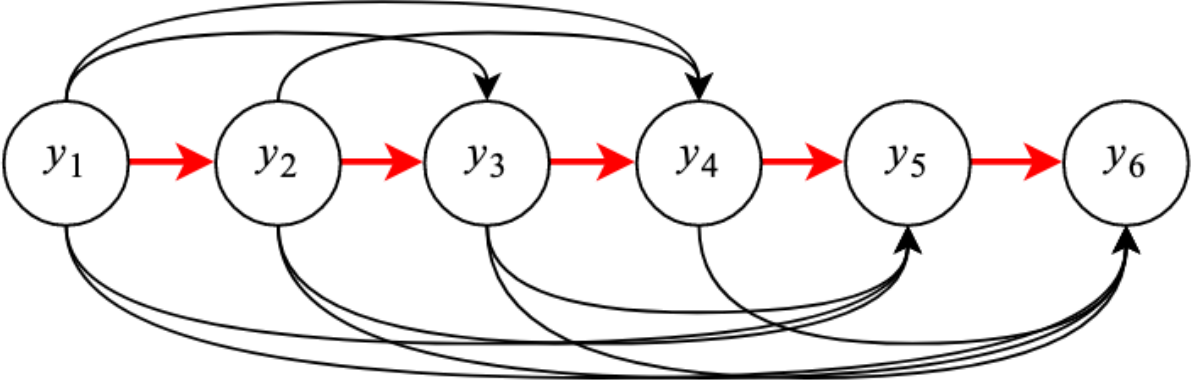
Yinhan Liu*

Luke Zettlemoyer

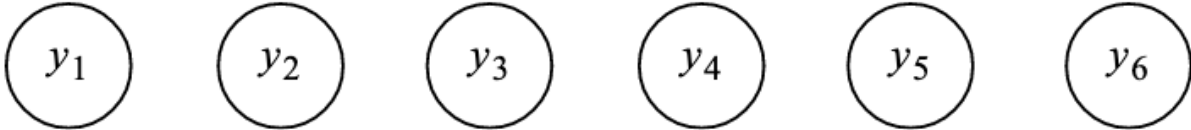
Facebook AI Research
Seattle, WA

Fully Non-Autoregressive Decoding

- Autoregressive decoding



- Fully non-autoregressive decoding



How about iterative refinement?

Mask-Predict: Iterative Refinement

- **Input:** Der Abzug der französischen Kampftruppen wurde am 20. November abgeschlossen
- **Step 1:** Predict the output length based on the input (12)

Mask-Predict: Iterative Refinement

- **Step 2:** Iterative non-autoregressive refinement

[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]
The	departure	departure	the	French	combat	completed	completed	on	20	November
0.9	0.2	0.3	0.1	0.4	0.2	0.2	0.2	0.1	0.6	0.7
The	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	[Mask]	20	November
	departure	of	French	combat	troops	troops	completed	on		
	0.2	0.9	0.8	0.7	0.8	0.1	0.2	0.9		
The	[Mask]	of	French	combat	troops	[Mask]	[Mask]	on	20	November
	withdrawal					was	completed			
	0.9					0.8	0.9			
The	withdrawal	of	French	combat	troops	was	completed	on	20	November

Results

Model	Dimensions (Model/Hidden)	Iterations	WMT'14		WMT'16	
			EN-DE	DE-EN	EN-RO	RO-EN
NAT w/ Fertility (Gu et al., 2018)	512/512	1	19.17	23.20	29.79	31.44
CTC Loss (Libovický and Helcl, 2018)	512/4096	1	17.68	19.80	19.93	24.71
Iterative Refinement (Lee et al., 2018)	512/512	1	13.91	16.77	24.45	25.73
	512/512	10	21.61	25.48	29.32	30.19
(Dynamic #Iterations)	512/512	?	21.54	25.43	29.66	30.30
<hr/>						
<i>Small CMLM with Mask-Predict</i>	512/512	1	15.06	19.26	20.12	20.36
	512/512	4	24.17	28.55	30.00	30.43
	512/512	10	25.51	29.47	31.65	32.27
<i>Base CMLM with Mask-Predict</i>	512/2048	1	18.05	21.83	27.32	28.20
	512/2048	4	25.94	29.90	32.53	33.23
	512/2048	10	27.03	30.53	33.08	33.31
<hr/>						
Base Transformer (Vaswani et al., 2017)	512/2048	<i>N</i>	27.30	— —	— —	— —
Base Transformer (Our Implementation)	512/2048	<i>N</i>	27.74	31.09	34.28	33.99
Base Transformer (+Distillation)	512/2048	<i>N</i>	27.86	31.07	— —	— —
Large Transformer (Vaswani et al., 2017)	1024/4096	<i>N</i>	28.40	— —	— —	— —
Large Transformer (Our Implementation)	1024/4096	<i>N</i>	28.60	31.71	— —	— —

Diffusion-LM Improves Controllable Text Generation

Xiang Lisa Li
Stanford University
xlisali@stanford.edu

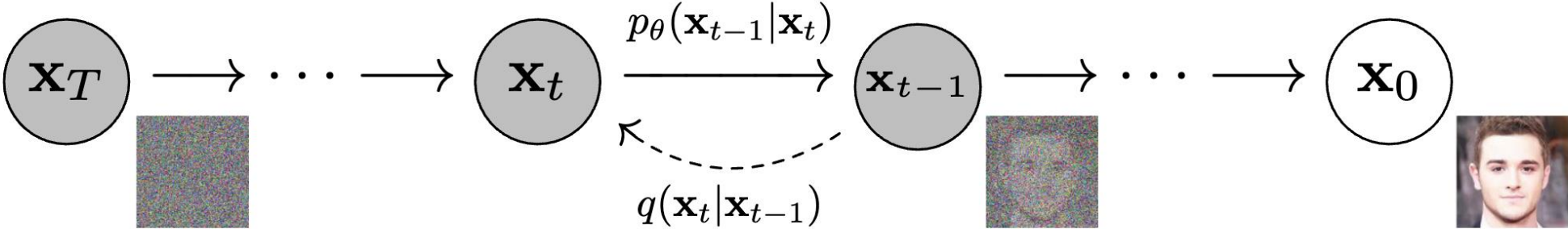
John Thickstun
Stanford University
jthickst@stanford.edu

Ishaan Gulrajani
Stanford University
igul@stanford.edu

Percy Liang
Stanford University
pliang@cs.stanford.edu

Tatsunori B. Hashimoto
Stanford University
thashim@stanford.edu

Image Diffusion



$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

Gradually add noise to image

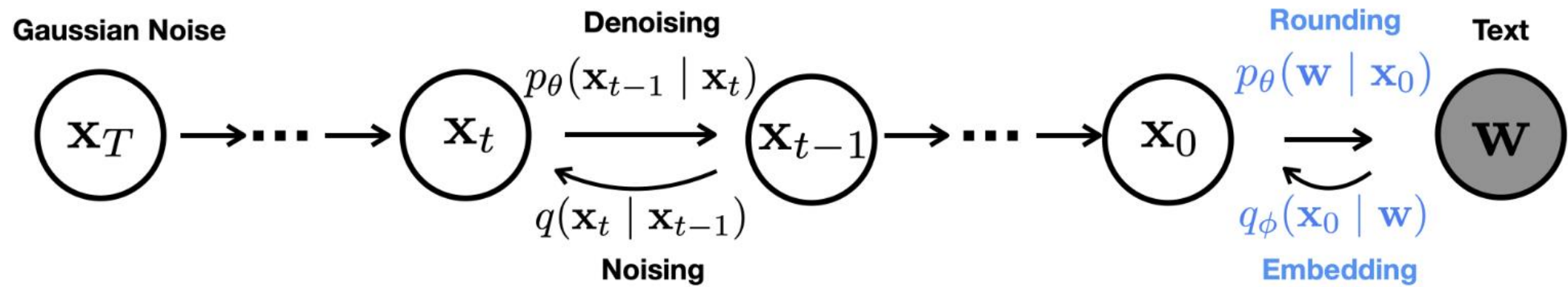
$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$

Learn to denoise

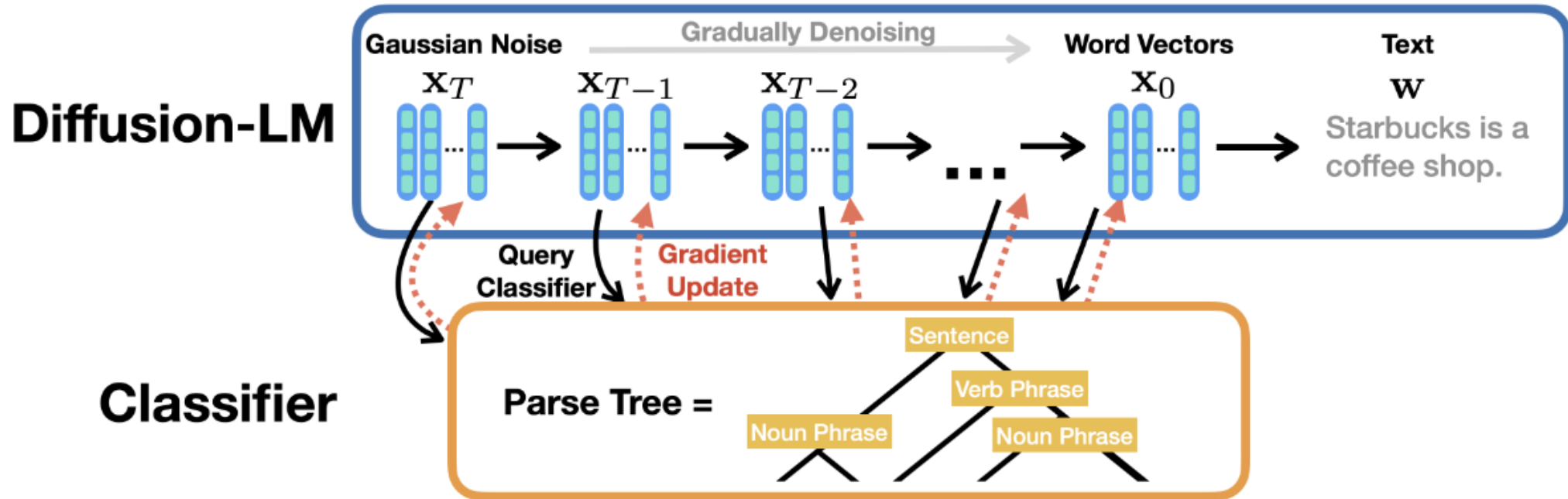
Image Diffusion Examples



Text Diffusion



Adding Control Capability



Adding Control Capability

input (Semantic Content)	food : Japanese
output text	Browns Cambridge is good for Japanese food and also children friendly near The Sorrento .
input (Parts-of-speech)	PROPN AUX DET ADJ NOUN NOUN VERB ADP DET NOUN ADP DET NOUN PUNCT
output text	Zizzi is a local coffee shop located on the outskirts of the city .
input (Syntax Tree)	(TOP (S (NP (*) (*) (*)) (VP (*) (NP (NP (*) (*))))))
output text	The Twenty Two has great food
input (Syntax Spans)	(7, 10, VP)
output text	Wildwood pub serves multicultural dishes and is ranked 3 stars
input (Length)	14
output text	Browns Cambridge offers Japanese food located near The Sorrento in the city centre .
input (left context)	My dog loved tennis balls.
input (right context)	My dog had stolen every one and put it under there.
output text	One day, I found all of my lost tennis balls underneath the bed.

Insertion Transformer: Flexible Sequence Generation via Insertion Operations

Mitchell Stern^{1,2} William Chan¹ Jamie Kiros¹ Jakob Uszkoreit¹

Insertion Operation

- Generate text by inserting words

This is a book

<slot 0> This <slot 1> is <slot 2> a <slot 3> book <slot 4>

Insert “boring” at <slot 3>

This is a boring book

<slot 0> This <slot 1> is <slot 2> a <slot 3> boring <slot 4> book <slot 5>

Insert “very” at <slot 3>

This is a very boring book

Example

t	Canvas	Insertion
0	[]	(ate, 0)
1	[<u>ate</u>]	(together, 1)
2	[ate, <u>together</u>]	(friends, 0)
3	[<u>friends</u> , ate, together]	(three, 0)
4	[<u>three</u> , friends, ate, together]	(lunch, 3)
5	[three, friends, ate, <u>lunch</u> , together]	(⟨EOS⟩, 5)

Example (Parallel Version)

t	Canvas	Insertions
0	[]	(ate, 0)
1	[<u>ate</u>]	(friends, 0), (together, 1)
2	[<u>friends</u> , ate, <u>together</u>]	(three, 0), (lunch, 2)
3	[<u>three</u> , friends, ate, <u>lunch</u> , together]	(⟨EOS⟩, 5)

Advantages

- More similar to human writing
- Don't need to predict the output length in advance
 - Dynamically decide when to stop

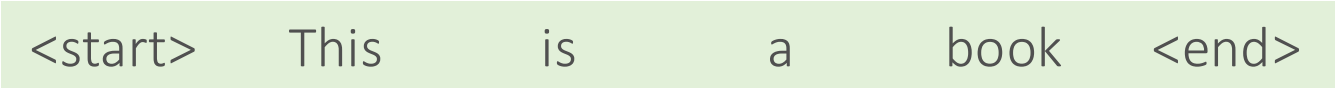
There is a bird on the tree.

There is a red bird on the green tree.

There is a red bird with a big beak on the green tree.

Insertion Transformer

This is a book



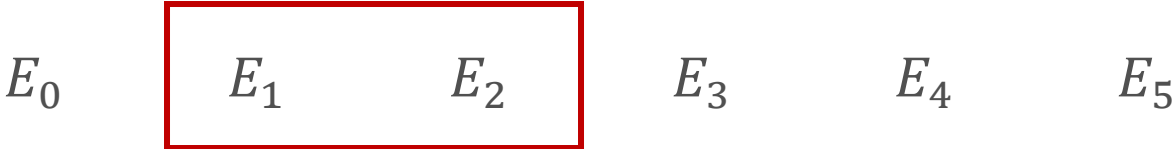
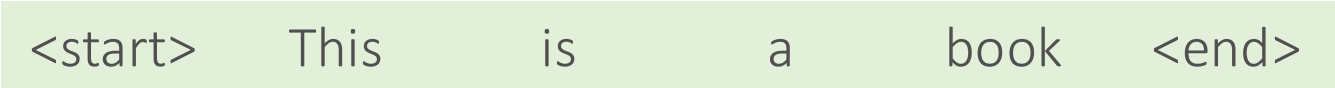
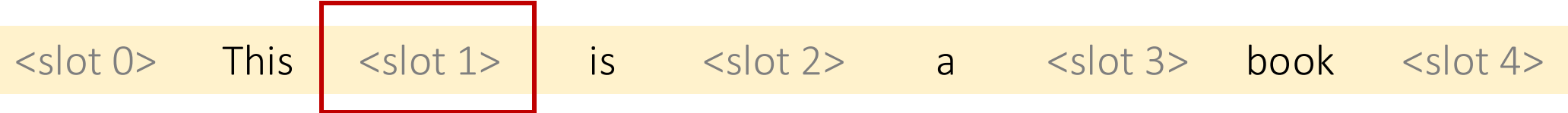
E_2 E_3 E_4 E_5

Transformer
Output

Slot Representation

Insertion Transformer

This is a book

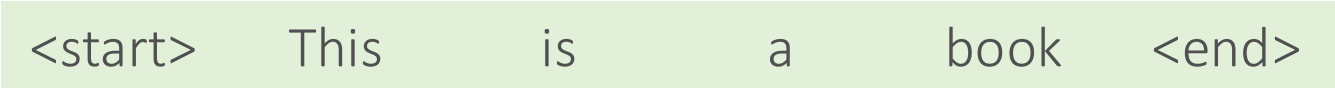


Transformer
Output

Slot Representation

Insertion Transformer

This is a book



E_0 E_1 E_2 E_3 E_4 E_5

Transformer
Output

Slot Representation

Predict words based on slot
representations

$$P = W^T \text{concat}(E_i, E_{i+1})$$

Training Loss

A B C D E F G H I J K L M N O

<S0> A <S1> C <S2> D <S3> I <S4> M <S5>

$$Loss^{(0)} = -\log p(EOS)$$

$$Loss^{(1)} = -\log p(B)$$

$$Loss^{(2)} = -\log p(EOS)$$

$$Loss^{(3)} = \text{Avg}(-\log p(E), -\log p(F), -\log p(G), -\log p(H))$$

$$Loss^{(4)} = \text{Avg}(-\log p(J), -\log p(K), -\log p(L))$$

$$Loss^{(5)} = \text{Avg}(-\log p(N), -\log p(O))$$

Training Loss (Balanced Binary Tree)

A B C D E F G H I J K L M N O

<S0> A <S1> C <S2> D <S3> I <S4> M <S5>

$$Loss^{(0)} = -\log p(EOS)$$

$$Loss^{(1)} = -\log p(B)$$

$$Loss^{(2)} = -\log p(EOS)$$

$$Loss^{(3)} = \text{Avg}(-\log p(E), -\log p(F), -\log p(G), -\log p(H))$$

$$Loss^{(4)} = \text{Avg}(-\log p(J), -\log p(K), -\log p(L))$$

$$Loss^{(5)} = \text{Avg}(-\log p(N), -\log p(O))$$

Training Loss (Balanced Binary Tree)



Results

Loss	Termination	BLEU (+EOS)	BLEU (+EOS)	BLEU (+EOS)
			+Distillation	+Distillation, +Parallel
Left-to-Right	Sequence	20.92 (20.92)	23.29 (23.36)	-
Binary Tree ($\tau = 0.5$)	Slot	20.35 (21.39)	24.49 (25.55)	25.33 (25.70)
Binary Tree ($\tau = 1.0$)	Slot	21.02 (22.37)	24.36 (25.43)	25.43 (25.76)
Binary Tree ($\tau = 2.0$)	Slot	20.52 (21.95)	24.59 (25.80)	25.33 (25.80)
Uniform	Sequence	19.34 (22.64)	22.75 (25.45)	-
Uniform	Slot	18.26 (22.16)	22.39 (25.58)	24.31 (24.91)

Examples

Input: Everyone has the Internet, an iPad and eBooks.

Output: Jeder hat das Internet, ein iPad und eBooks.

Greedy decode (uniform loss):

Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..

(continued)

Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..
Jeder_ hat_ das_ Internet_ , _ ein_ i Pad _ und_ eB oo ks_ ..

Examples

Input: But on the other side of the state, that is not the impression many people have of their former governor.

Output: Aber auf der anderen Seite des Staates ist das nicht der Eindruck, den viele von ihrem ehemaligen Gouverneur haben.

Parallel decode (binary tree loss):

Aber_ auf_ der_ anderen_ Seite_ des_ Staates_ ist_ das_ nicht_ der_ Eindruck , _ den_ viele_ von_ ihrem_ ehemaligen_ Gouverneur_ haben_ ..
Aber_ auf_ der_ anderen_ Seite_ des_ Staates_ ist_ das_ nicht_ der_ Eindruck_ , _ den_ viele_ von_ ihrem_ ehemaligen_ Gouverneur_ haben_ ..
Aber_ auf_ der_ anderen_ Seite_ des_ Staates_ ist_ das_ nicht_ der_ Eindruck_ , _ den_ viele_ von_ ihrem_ ehemaligen_ Gouverneur_ haben_ ..
Aber_ auf_ der_ anderen_ Seite_ des_ Staates_ ist_ das_ nicht_ der_ Eindruck_ , _ den_ viele_ von_ ihrem_ ehemaligen_ Gouverneur_ haben_ ..
Aber_ auf_ der_ anderen_ Seite_ des_ Staates_ ist_ das_ nicht_ der_ Eindruck_ , _ den_ viele_ von_ ihrem_ ehemaligen_ Gouverneur_ haben_ ..

Levenshtein Transformer

Jiatao Gu[†], Changhan Wang[†], and Jake Zhao (Junbo)[‡][◇]

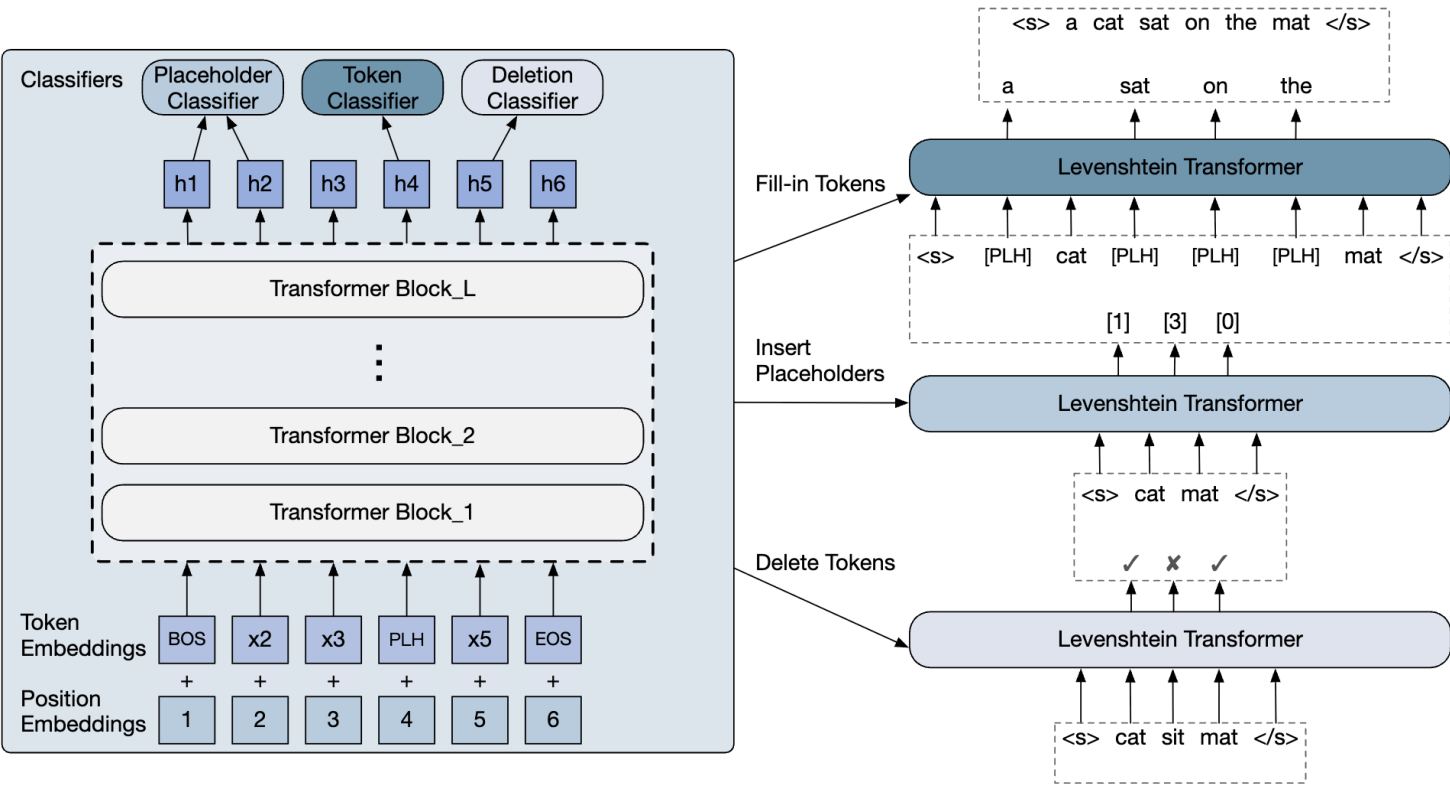
[†]Facebook AI Research

[‡]New York University [◇]Tigerobo Inc.

[†]{jgu, changhan}@fb.com [‡]jakezhao@cs.nyu.edu

Levenshtein Transformer

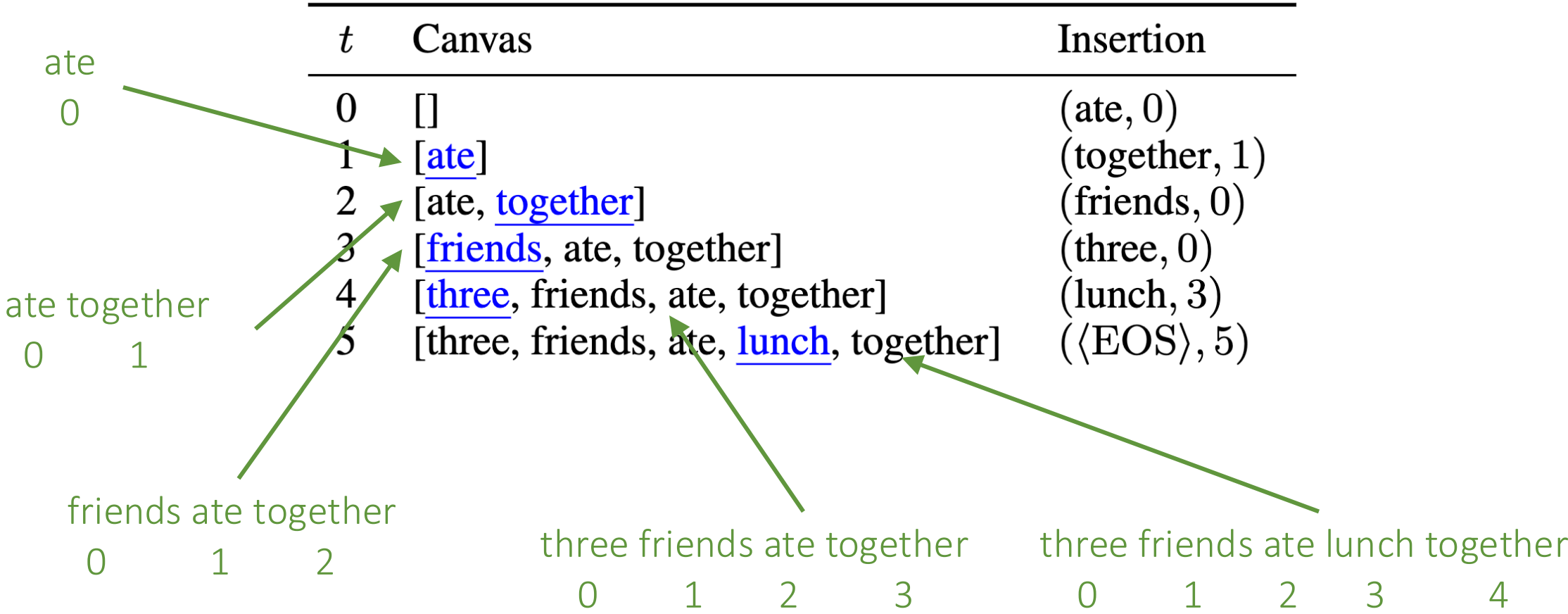
- Extension of Insertion Transformer
- Three operations
 - Insert placeholders
 - Fill-in tokens
 - Delete tokens



INSNET: An Efficient, Flexible, and Performant Insertion-based Text Generation Model

Sidi Lu, Tao Meng, Nanyun Peng
University of California, Los Angeles
`{sidilu, tmeng, violetpeng}@cs.ucla.edu`

Inefficient Position Encoding for Insertion Operation



Modified Relative Positional Encoding

	<s>	</s>	have	pen	I	a	.
Step 0:	0	-	-	-	-	-	-
Step 1:	0	1	-	-	-	-	-
Step 2:	0	1 2	1	-	-	-	-
Step 3:	0	2 3	1	2	-	-	-
Step 4:	0	3 4	1 2	2 3	1	-	-
Step 5:	0	4 5	2	3 4	1	3	-
Step 6:	0	5 6	2	4	1	3	5

(a) Absolute Positions

	<s>	</s>	have	pen	I	a	.
<s>	0	6	2	4	1	3	5
</s>	-6	0	-4	-2	-5	-3	-1
have	-2	4	0	2	-1	1	3
pen	-4	2	-2	0	-3	-1	1
I	-1	5	1	3	0	2	4
a	-3	3	-1	1	-2	0	2
.	-5	1	-3	-1	-4	-2	0

(b) Vanilla Relative Positions

	<s>	</s>	have	pen	I	a	.
<s>	0	-	-	-	-	-	-
</s>	-1	0	-	-	-	-	-
have	-1	1	0	-	-	-	-
pen	-2	1	-1	0	-	-	-
I	-1	3	1	2	0	-	-
a	-3	2	-1	1	-2	0	-
.	-5	1	-3	-1	-4	-2	0

(c) Our Relative Positions

	<s>	I	have	a	pen	.	</s>
<s>	0	-	-	-	-	-	-
I	-1	0	1	-	2	-	3
have	-1	-	0	-	-	-	1
a	-3	-2	-1	0	1	-	2
pen	-2	-	-1	-	0	-	1
.	-5	-4	-3	-2	-1	0	1
</s>	-1	-	-	-	-	-	0

(d) Ours, Order-Restored

Model	#Re-Enc. w/ ParaDec	#Re-Enc. w/ SeqDec	PosInfo
Ins. Trans.	$\Theta(\log n)$	$O(n)$	Absolute
Lev. Trans.	$\Theta(\log n)$	$\Omega(n)$	Absolute
NMSTG	$O(n)$	$O(n)$	Markovian/Absolute
InDIGO	N/A	$O(1)$	Direction-only
INSNET (Ours)	$O(1)$	$O(1)$	Relative

Results

Model	Yelp Review			News		
	BLEU-2/4	NIST-2/4	# Dec. Steps	BLEU-2/4	NIST-2/4	# Dec. Steps
Auto-regressive Transformer (Plan-And-Write-static, Yao et al. (2019))	16.68/5.46	2.79/2.86	39.24	8.79/2.40	1.65/1.67	36.74
NMSTG (Welleck et al., 2019)	10.06/1.92	1.11/1.12	27.92	10.67/1.58	2.70/2.70	27.85
InDIGO* (Gu et al., 2019a) (w/ Searched Adaptive Order)	16.14/4.63	3.08/3.10	45.63	13.89/3.62	3.08/3.10	26.78
Levenshtein Transformer (Parallel Decoding, Gu et al., 2019b)	14.84/3.96	2.84/2.89	14.28	11.76/1.89	2.64/2.71	16.13
InsT-POINTER-Base (BERT init)	11.48/2.16	2.15/2.15	6.00	12.13/1.63	2.90/2.80	6.00
InsT-POINTER-Base (BERT init+Wiki)	15.63/3.32	3.27/3.30	6.00	13.01/2.51	3.04/3.06	6.00
InsT-POINTER-Large (BERT init+Wiki)	16.78/3.79	3.49/3.53	6.00	14.04/3.04	3.28/3.30	6.00
INSNET (Ours, Fully-Sequential)	19.36/5.78	3.51/3.54	48.73	16.31/4.96	3.10/3.13	32.69
INSNET-uniform (Ours)	12.31/2.30	2.19/2.17	7.00	12.89/2.01	2.99/2.90	7.00
INSNET-Dinic (Ours, $\tau = 10.0$)	16.73/4.35	3.19/3.20	11.83	14.13/3.75	2.97/3.00	8.13