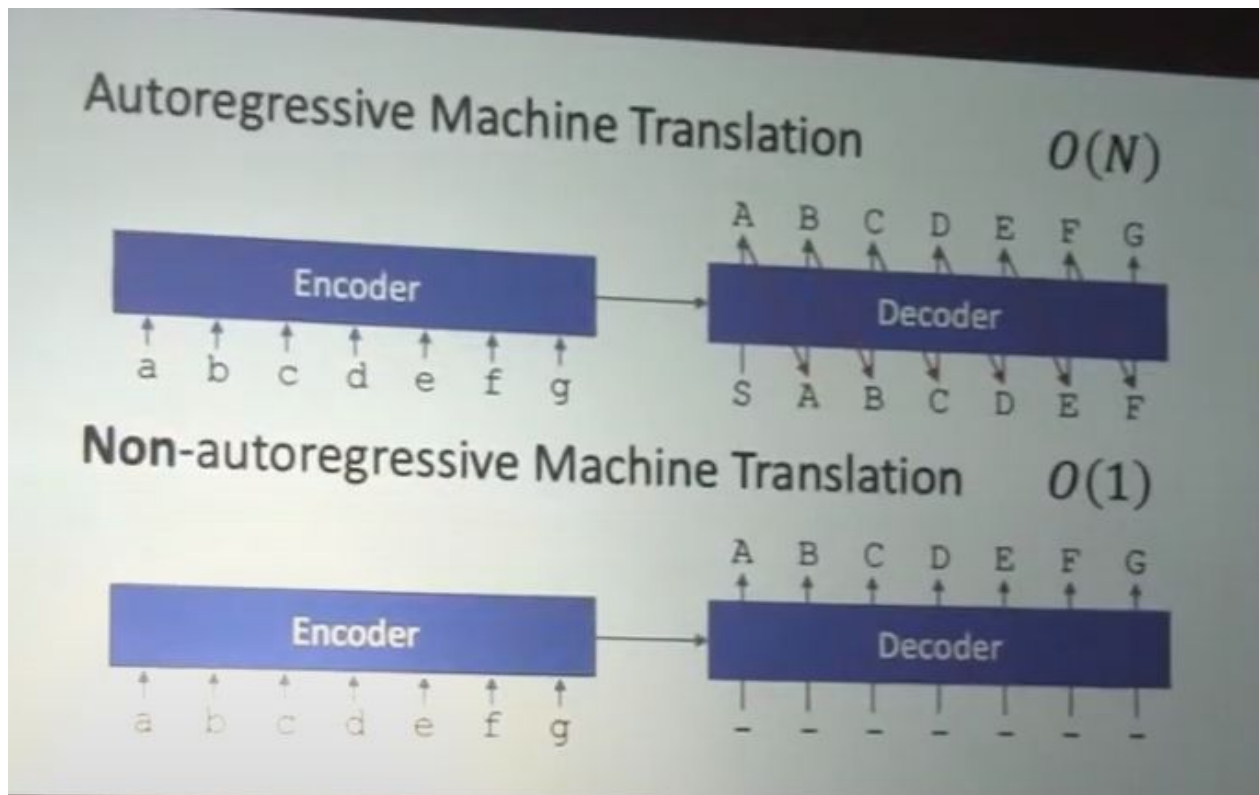


# Неавторегрессивный машинный перевод

Как описано в статье “Mask-Predict: Parallel  
Decoding of Conditional Masked Language Models”

Докладчик - Коля Карташев, 181

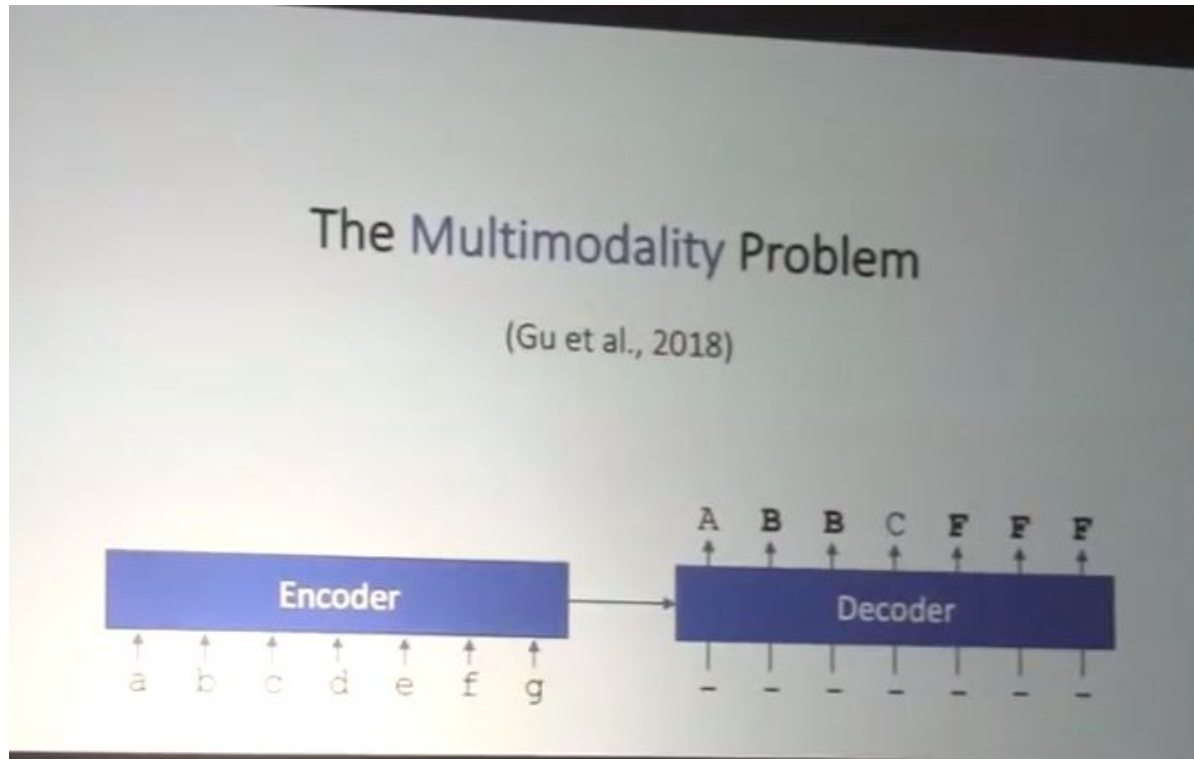
# Неавторегрессивный машинный перевод - что это



# Плюсы такого подхода

- Требуется  $O(1)$  запросов к декодеру против  $O(n)$  у классической архитектуры
- Не требуется алгоритма перебора вариантов в генерации предсказания, таких как Beam Search.

# The Multimodality Problem

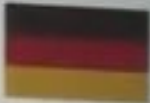


# The Multimodality Problem - пример

The Multimodality Problem (Gu et al., 2018)



Thank you very much



Danke schön (Option 1)  
Vielen Dank (Option 2)

Thank you very much → Danke Dank

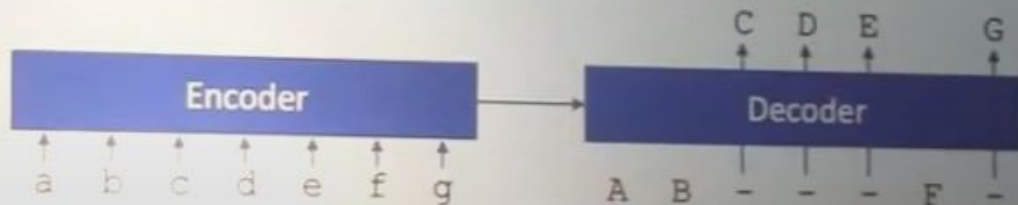
Token predictions are conditionally independent!

# Архитектура модели

## Training Conditional Masked Language Models

- Encoder-decoder transformer w/o causal self-attention
- Mask  $k$  target tokens randomly  $k \sim \text{Uniform}(1, N)$
- Predict masked tokens

$$P(D|A, B, F, a, b, c, d, e, f, g)$$



# Основные шаги Training:

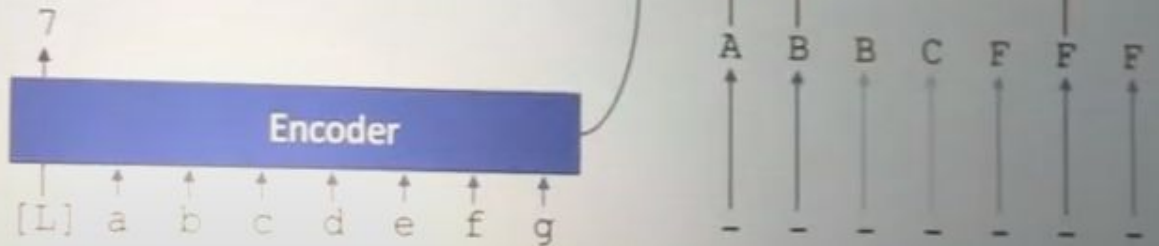
- Случайное количество токенов, выбранное равномерно, маскируются.
- Требуется предсказать замаскированные токены, на основе уже подставленных и энкодинга предложения на другом языке - BERT с подсказкой

# Prediction

## Parallel Decoding with Mask-Predict

### Iteration

- 1) **Mask** least probable tokens
- 2) Predict all target tokens in parallel via argmax





# Несколько особенностей

- Длина предложения не предсказывается этой моделью “нативно”, поэтому ее предсказывают через отдельный модуль классификации в энкодере
- Чтобы побороть мультимодальность, модель не просят размаскировать все токены сразу, а делают  $n\_steps$  шагов, на каждом добавляя  $len / n\_steps$  значений токенов.

# Пример перевода

Parallel Decoding with Mask-Predict												
Source	<i>Der Abzug der franzsischen Kampftruppen wurde am 20. November abgeschlossen.</i>											
Mask	-	-	-	-	-	-	-	-	-	-	-	-
Predict	The	departure	of	the	French	combat	completed	completed	on	20	November	-
Mask	The	-	-	-	-	-	-	-	-	20	November	-
Predict	The	departure	of	French	combat	troops	was	completed	on	20	November	-
Mask	The	-	of	French	combat	troops	was	-	on	-	-	-
Predict	The	withdrawal	of	French	combat	troops	was	completed	on	November	20th	-

# Гиперпараметры: параметры модели и тренировки

- 6 слоев, 8 attention heads
- Варианты с 512 или 2048 hidden размерностью (для лучшего сравнения с другими архитектурами)
- $\text{lr}$  с разогревом до  $5e-4$  за 10000 слоев, снижается со скоростью обратного квадратного корня

# Гиперпараметры

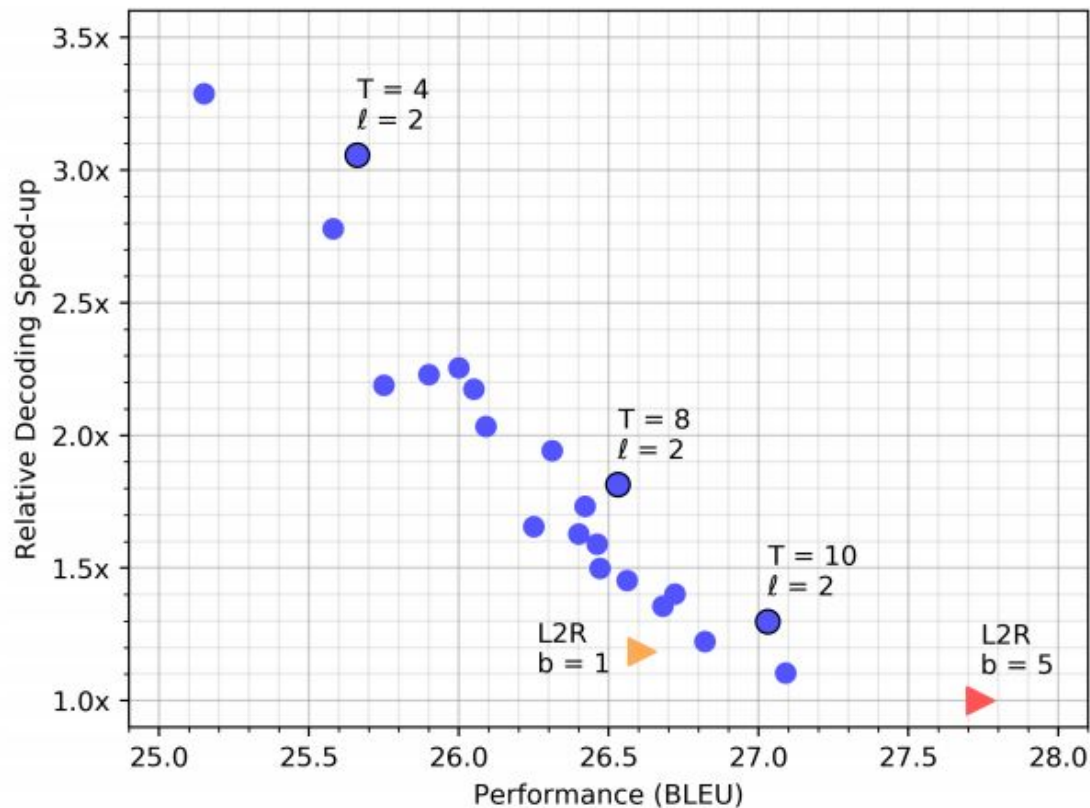
- Количество подбираемых кандидатов на длину (параметр  $l$ )
- Количество итераций декодинга (1 - 10)
- Дистиллировать ли модель?

# Альтернативные модели с быстрым декодингом

1. Обычная авторегрессивная модель: Base Transformer (Vaswani et al., 2017) [\[1\]](#)
2. Наиболее близкое к описываемой статье исследование - Iterative refinement (Lee et al., 2018) [\[2\]](#)
3. “Чисто” неавторегрессивная модель = 1 итерация декодинга (Gu & Kong, 2020) [\[3\]](#)
4. Imputer: модель, обходящая по качеству обычный трансформер. (Chan et al., 2020) [\[4\]](#)

Подробнее про каждую архитектуру позже!

# Сравнение CMLM с Base Transformer



$T$  = количество итераций

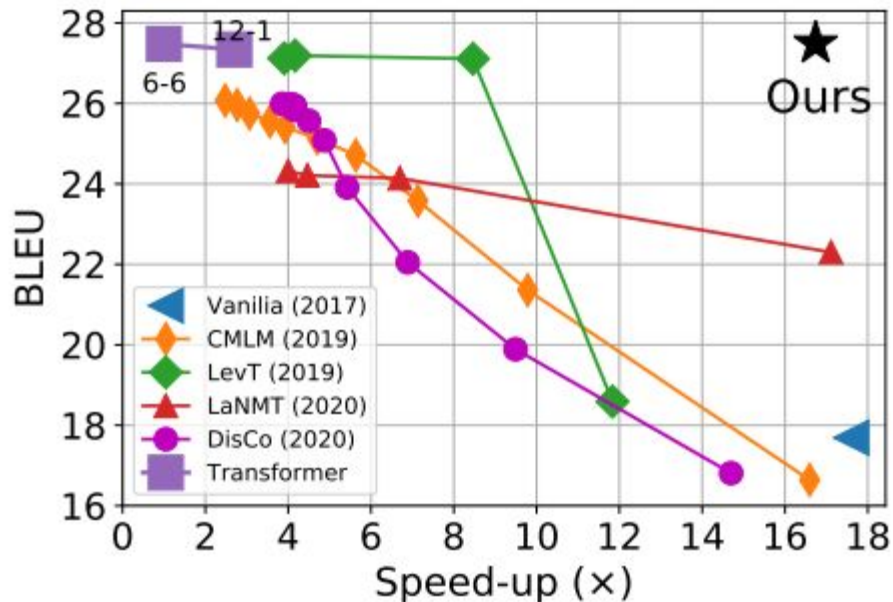
$l$  = количество кандидатов  
длины

$b$  = размер луча в Beam  
Search

# Сравнение с прошлыми моделями

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
<i>Small CMLM with Mask-Predict</i>	512/512	1	15.06	19.26	20.12	20.36
	512/512	4	<b>24.17</b>	<b>28.55</b>	<b>30.00</b>	30.43
	512/512	10	<b>25.51</b>	<b>29.47</b>	<b>31.65</b>	<b>32.27</b>
<i>Base CMLM with Mask-Predict</i>	512/2048	1	18.05	21.83	27.32	28.20
	512/2048	4	<b>25.94</b>	<b>29.90</b>	<b>32.53</b>	<b>33.23</b>
	512/2048	10	<b>27.03</b>	<b>30.53</b>	<b>33.08</b>	<b>33.31</b>
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	—	—

# Сравнение с “будущими” моделями



Под “Ours” обозначена архитектура из статьи “Fully Non-autoregressive Neural Machine Translation: Tricks of the Trade”



# Сравнение качества работы

Models		Iter.	Speed	WMT'14		WMT'16	
				EN-DE	DE-EN	EN-RO	RO-EN
AT	Transformer <i>base</i> (teacher)	N	1.0×	<b>27.48</b>	<b>31.39</b>	<b>33.70</b>	<b>34.05</b>
	Transformer <i>base</i> (12-1)	N	2.4×	26.21	30.80	33.17	33.21
	+ KD	N	2.5×	27.34	30.95	33.52	34.01
Iterative NAT	iNAT (Lee et al., 2018)	10	1.5×	21.61	25.48	29.32	30.19
	Blockwise (Stern et al., 2018)	$\approx N/5$	3.0×	27.40	-	-	-
	InsT (Stern et al., 2019)	$\approx \log N$	4.8×	27.41	-	-	-
	<u>CMLM</u> (Ghazvininejad et al., 2019)*	10	1.7×	27.03	30.53	33.08	33.31
	LevT (Gu et al., 2019)	Adv.	4.0×	27.27	-	-	33.26
	KERMIT (Chan et al., 2019)	$\approx \log N$	-	27.80	30.70	-	-
	LaNMT (Shu et al., 2020)	4	5.7×	26.30	-	-	29.10
	SMART (Ghazvininejad et al., 2020b)*	10	1.7×	27.65	31.27	-	-
	DisCO (Kasai et al., 2020a)*	Adv.	3.5×	27.34	31.31	33.22	33.25
	<u>Imputer</u> (Saharia et al., 2020)*	8	3.9×	<b>28.20</b>	<b>31.80</b>	<b>34.40</b>	<b>34.10</b>
Fully NAT	Vanilla-NAT (Gu et al., 2018a)	1	15.6×	17.69	21.47	27.29	29.06
	LT (Kaiser et al., 2018)	1	3.4×	19.80	-	-	-
	CTC (Libovický and Helcl, 2018)	1	-	16.56	18.64	19.54	24.67
	NAT-REG (Wang et al., 2019)	1	-	20.65	24.77	-	-
	Bag-of-ngrams (Shao et al., 2020)	1	10.0×	20.90	24.60	28.30	29.30
	Hint-NAT (Li et al., 2018)	1	-	21.11	25.24	-	-
	DCRF (Sun et al., 2019)	1	10.4×	23.44	27.22	-	-
	Flowseq (Ma et al., 2019)	1	1.1 ×	23.72	28.39	29.73	30.72
	ReorderNAT (Ran et al., 2019)	1	16.1×	22.79	27.28	29.30	29.50
	AXE (Ghazvininejad et al., 2020a)*	1	15.3×	23.53	27.90	30.75	31.54
	EM+ODD (Sun and Yang, 2020)	1	16.4×	24.54	27.93	-	-
	GLAT (Qian et al., 2020)	1	15.3×	25.21	29.84	31.19	32.04
	<u>Imputer</u> (Saharia et al., 2020)*	1	18.6×	25.80	28.40	32.30	31.70
	<u>Ours (Fully NAT)</u>	1	17.6×	11.40	16.47	24.52	24.79
	+ KD	1	17.6×	19.50	24.95	29.91	30.25
	+ KD + CTC	1	16.8×	26.51	30.46	33.41	34.07
	+ KD + CTC + VAE	1	16.5×	<b>27.49</b>	<b>31.10</b>	<b>33.79</b>	33.87
	+ KD + CTC + GLAT	1	16.8×	27.20	<b>31.39</b>	33.71	<b>34.16</b>

# Различные улучшения из статьи про Fully NAT

Methods	Distillation	Latent Variables	Latent Alignments	Glancing Targets
What it can do?	simplifying the training data	model any types of dependency in theory	handling token shifts in the output space	ease the difficulty of learning hard examples
What it cannot?	uncertainty exists in the teacher model	constrained by the modeling power of the used latent variables	unable to model non-monotonic dependency, e.g. reordering	training / testing phase mismatch
Potential issues	sub-optimal due to the teacher's capacity	difficult to train; posterior collapse	decoder inputs must be longer than targets	difficult to find the optimal masking ratio

# Imputer

- CTC
- Разбивает на блоки, на каждом шаге предсказывает по одному токену из блока
- Использует ДП для выравнивания предсказания с ответом

# Итоги

- Почти без потери качества можно достичь ускорения в несколько раз (x2 - x16)
- Дистилляция модели всегда полезна (возможно ниже)
- Качественно превзойти left-to-right декодеры не получается

# Нужна ли дистилляция?

Iterations	WMT'14 EN-DE		WMT'16 EN-RO	
	Raw	Dist	Raw	Dist
$T = 1$	10.64	<b>18.05</b>	21.22	<b>27.32</b>
$T = 4$	22.25	<b>25.94</b>	31.40	<b>32.53</b>
$T = 10$	24.61	<b>27.03</b>	32.86	<b>33.08</b>

- Уменьшение зашумленности данных, в теории вызывает значительное облегчение обучения модели
- Результаты значительно улучшаются с дистилляцией

# Как подбирать количество кандидатов длины?

Length Candidates	WMT'14 EN-DE BLEU	LP	WMT'16 EN-RO BLEU	LP
$\ell = 1$	26.56	16.1%	32.75	13.8%
$\ell = 2$	27.03	30.6%	33.06	26.1%
$\ell = 3$	<b>27.09</b>	43.1%	<b>33.11</b>	39.6%
$\ell = 4$	<b>27.09</b>	53.1%	32.13	49.2%
$\ell = 5$	27.03	62.2%	33.08	57.5%
$\ell = 6$	26.91	69.5%	32.91	64.3%
$\ell = 7$	26.71	75.5%	32.75	70.4%
$\ell = 8$	26.59	80.3%	32.50	74.6%
$\ell = 9$	26.42	83.8%	32.09	78.3%
Gold	27.27	—	33.20	—

- 4 кандидата  
подходят  
идеально в  
большинстве  
ситуаций

Спасибо!

# Список литературы:

- [0] Ghazvininejad et. al, 2019. [Mask-Predict: Parallel Decoding of Conditional Masked Language Models](#)
- [1] Vaswani et al., 2017. [Attention Is All You Need](#)
- [2] Lee et al., 2018. Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement
- [3]Gu & Kong, 2020. [Fully Non-autoregressive Neural Machine Translation: Tricks of the Trade](#)
- [4]Chan et al., 2020. [Imputer: Sequence Modelling via Imputation and Dynamic Programming](#)