Automatic Speech Recognition

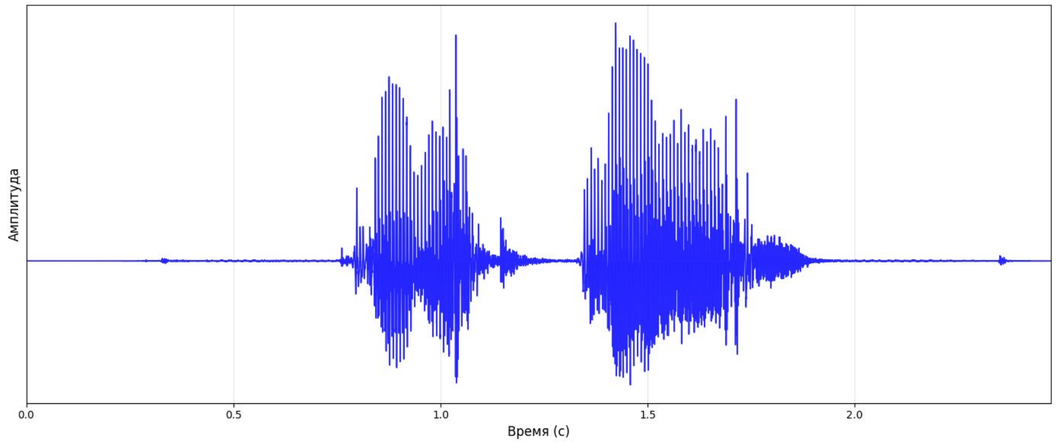
лекция 1

Апарин Георгий

- Постановка задачи и метрики оценки моделей
- Проблема выравнивания между текстом и аудио
- CTC loss
- Выбор лучшего распознавания
- Кодирование латентных представлений
- Encoder - Decoder подход
- RNNT

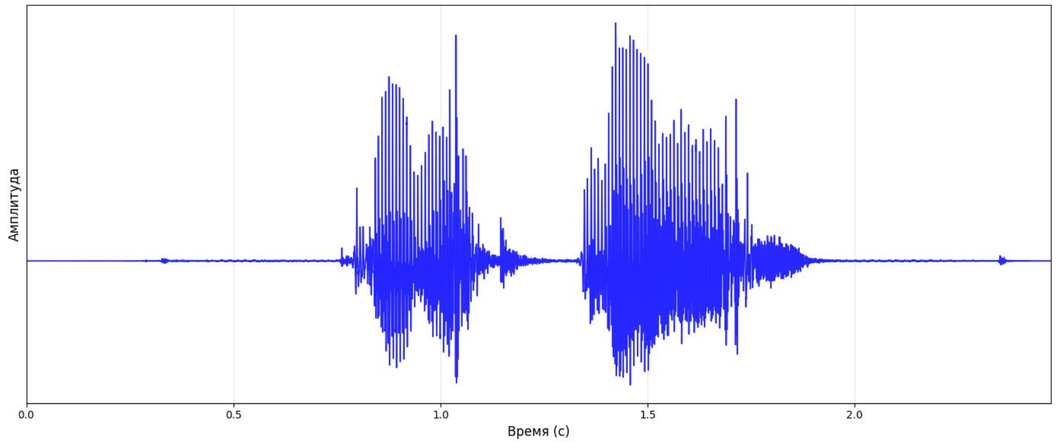
Постановка задачи и метрики оценивания

Связь аудио и текста



СКАЗАННЫЙ ТЕКСТ

Связь аудио и текста



СКАЗАННЫЙ ТЕКСТ

Типы ошибок

Референс

Замена

Удаление

Добавление

Терминология:
референс - эталонный текст
гипотеза - распознанный моделью

Типы ошибок

Референс Мы жили в маленьком домике у синего моря

Замена

Удаление

Добавление

Терминология:
референс - эталонный текст
гипотеза - распознанный моделью

Типы ошибок

Референс Мы жили в маленьком домике у синего моря

Замена Мы шили в маленьком домике у синего моря

Удаление

Добавление

Терминология:
референс - эталонный текст
гипотеза - распознанный моделью

Типы ошибок

Референс	Мы жили в маленьком домике у синего моря
Замена	Мы шили в маленьком домике у синего моря
Удаление	Мы в маленьком домике у синего моря
Добавление	

Терминология:
референс - эталонный текст
гипотеза - распознанный моделью

Типы ошибок

Референс	Мы жили в маленьком домике у синего моря
Замена	Мы шили в маленьком домике у синего моря
Удаление	Мы в маленьком домике у синего моря
Добавление	Мы и мыши жили в маленьком домике у синего моря

Терминология:
референс - эталонный текст
гипотеза - распознанный моделью

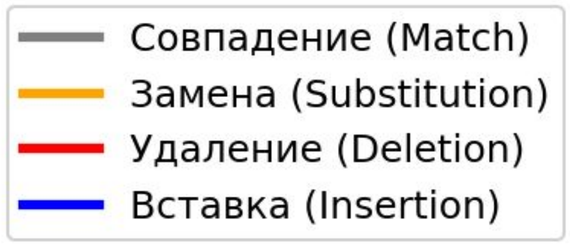
Метрики оценки

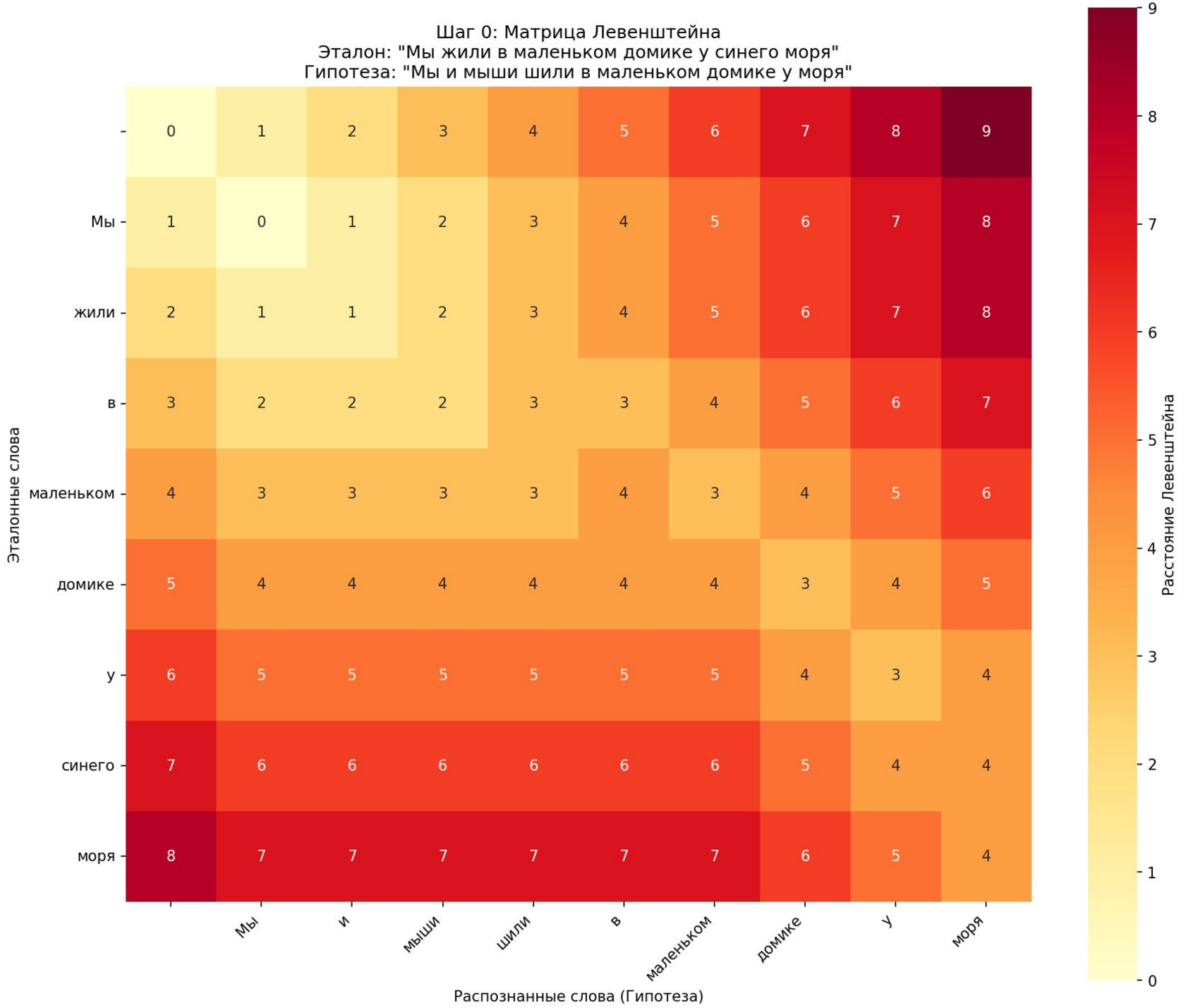
$$WER = \frac{S + D + I}{N}$$

S - количество замен (substitutions)
D - количество удалений (deletions)
I - количество добавлений (insertions)
N - количество слов всего

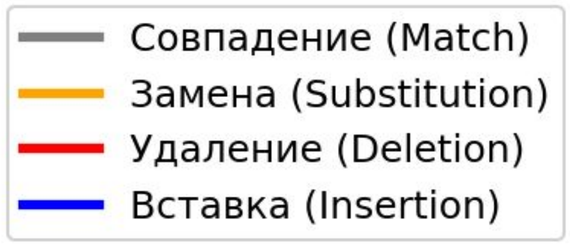
Значения S, D, I и N чаще считаются для всего датасета, а не отдельных аудио, чтобы нормализовать вклад в итоговую метрику текстов разной длины

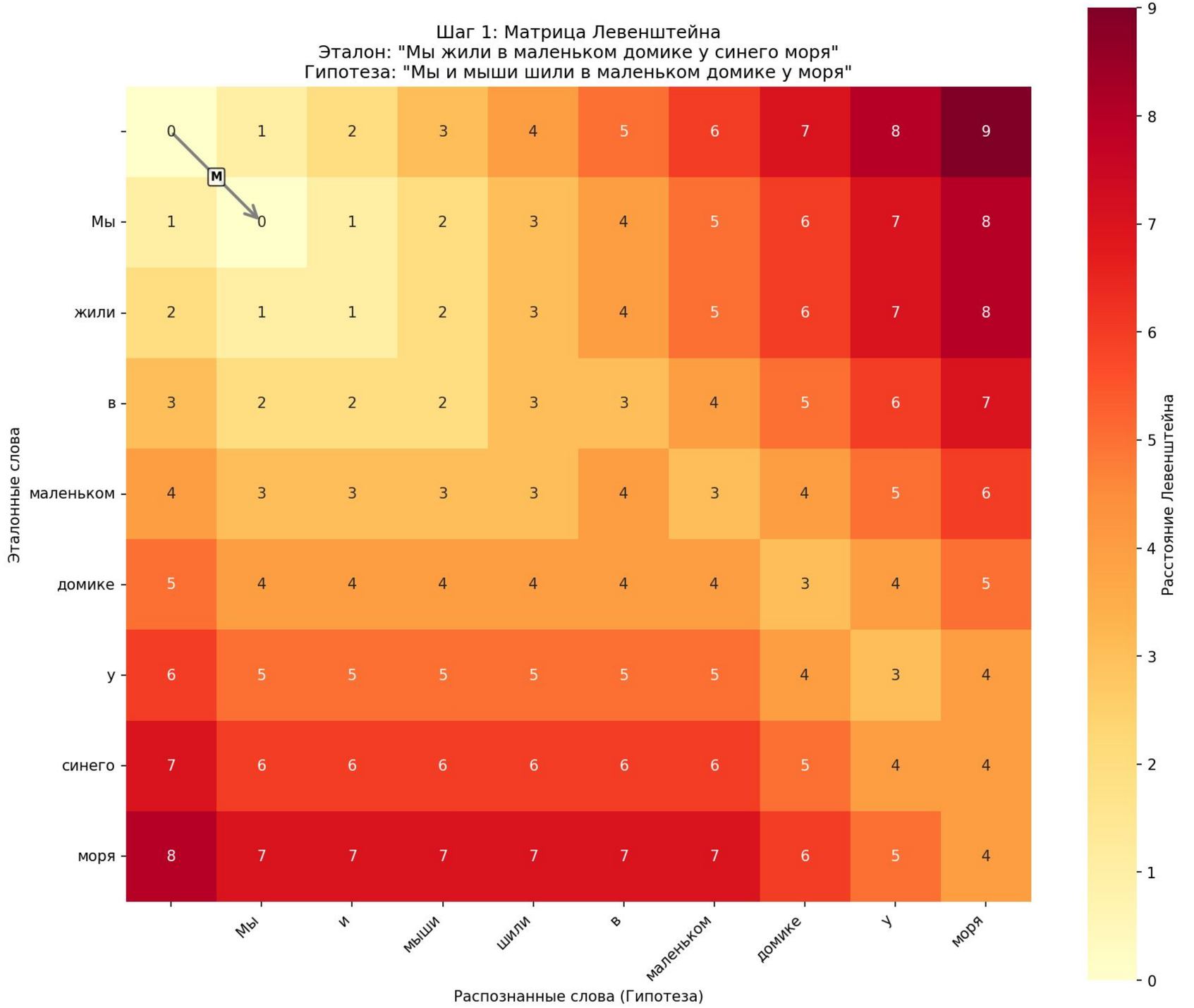
Алгоритм подсчета S, D, I



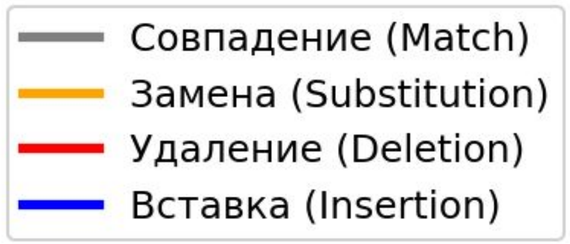


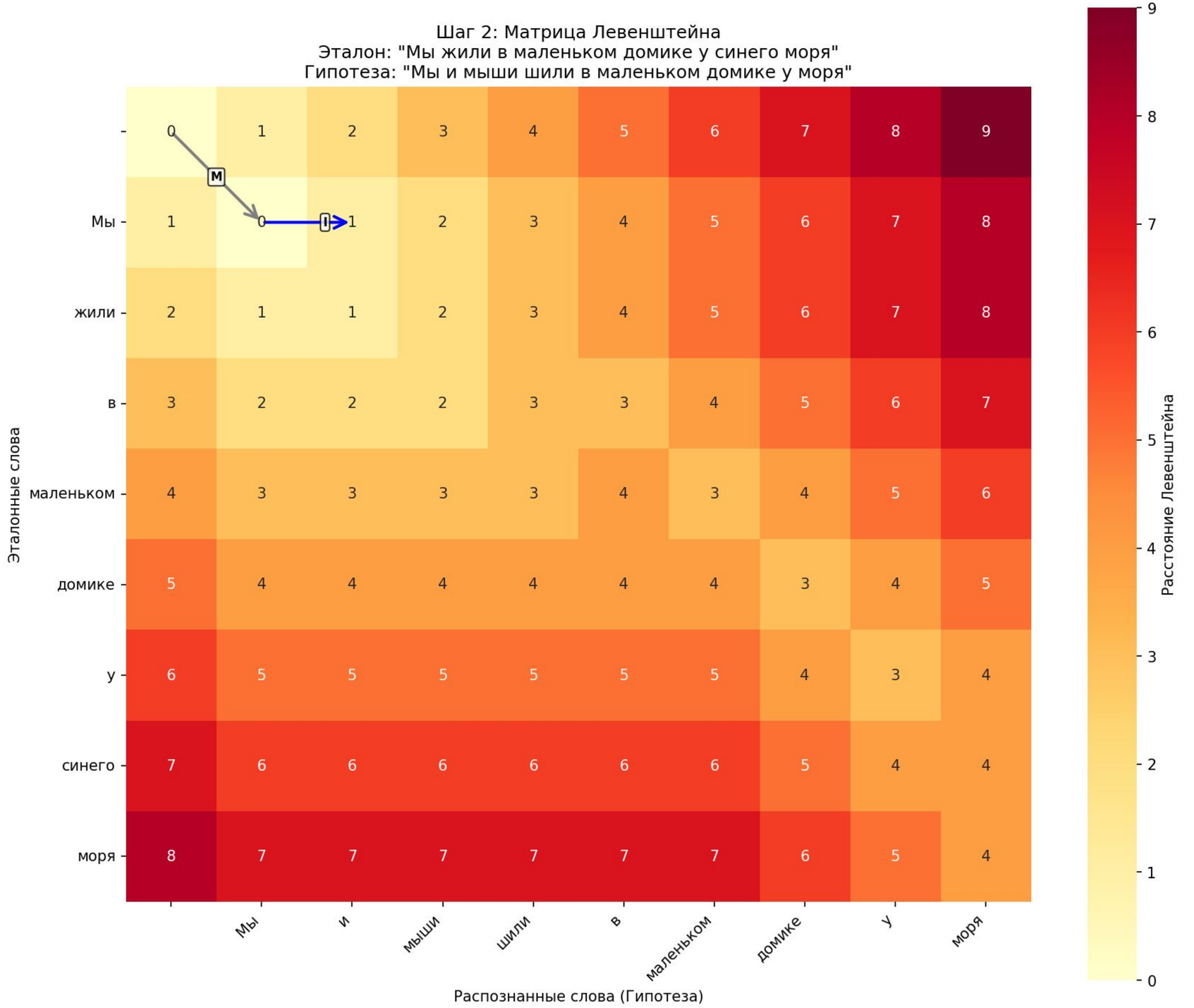
Алгоритм подсчета S, D, I



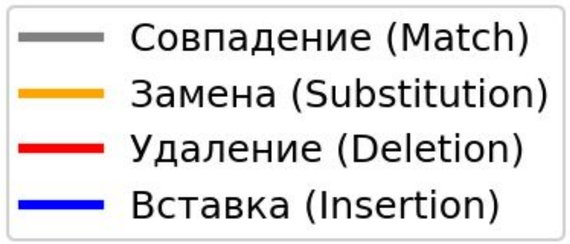


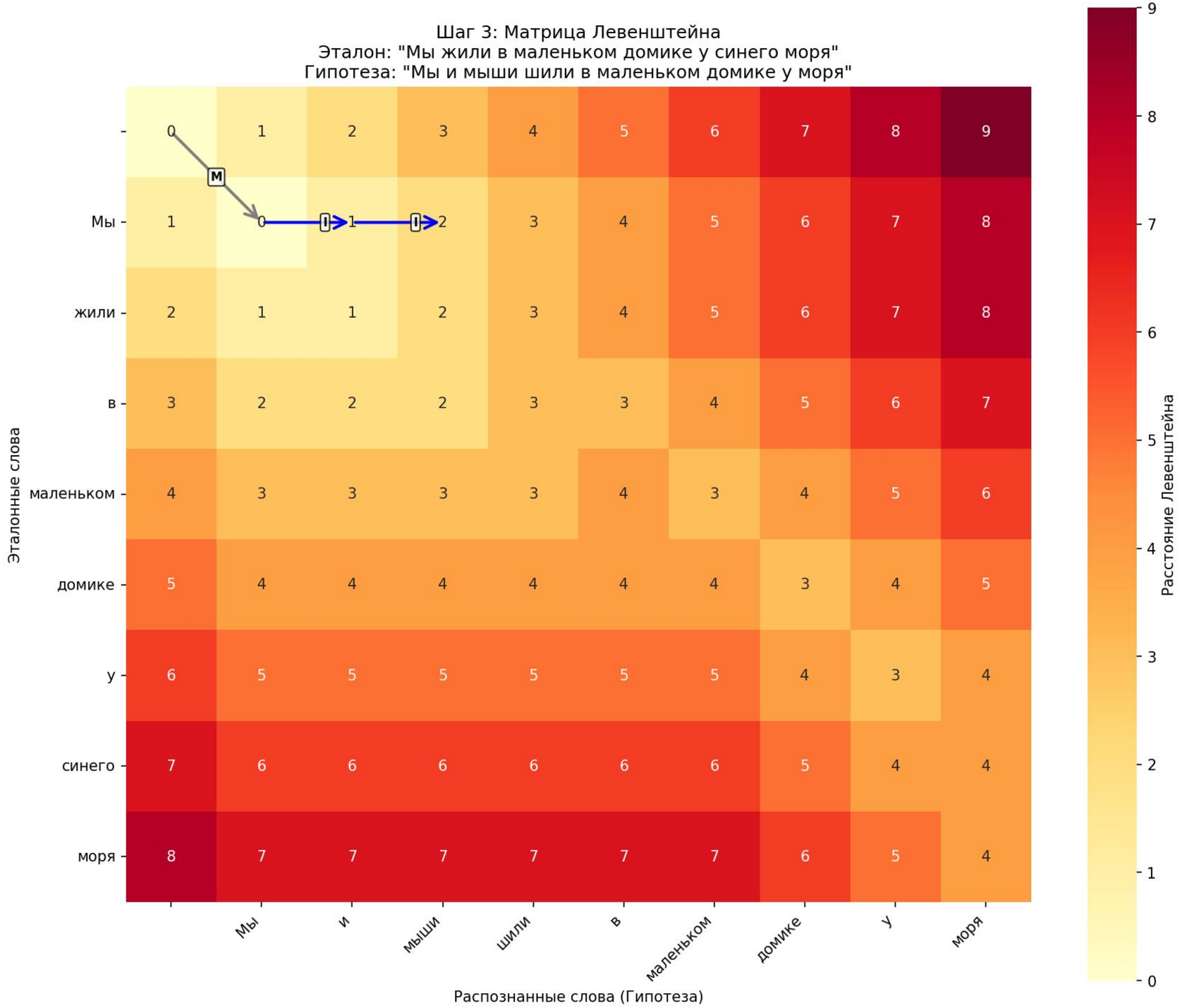
Алгоритм подсчета S, D, I



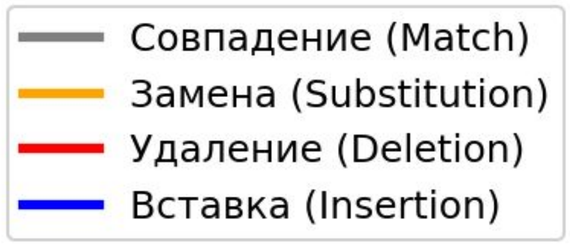


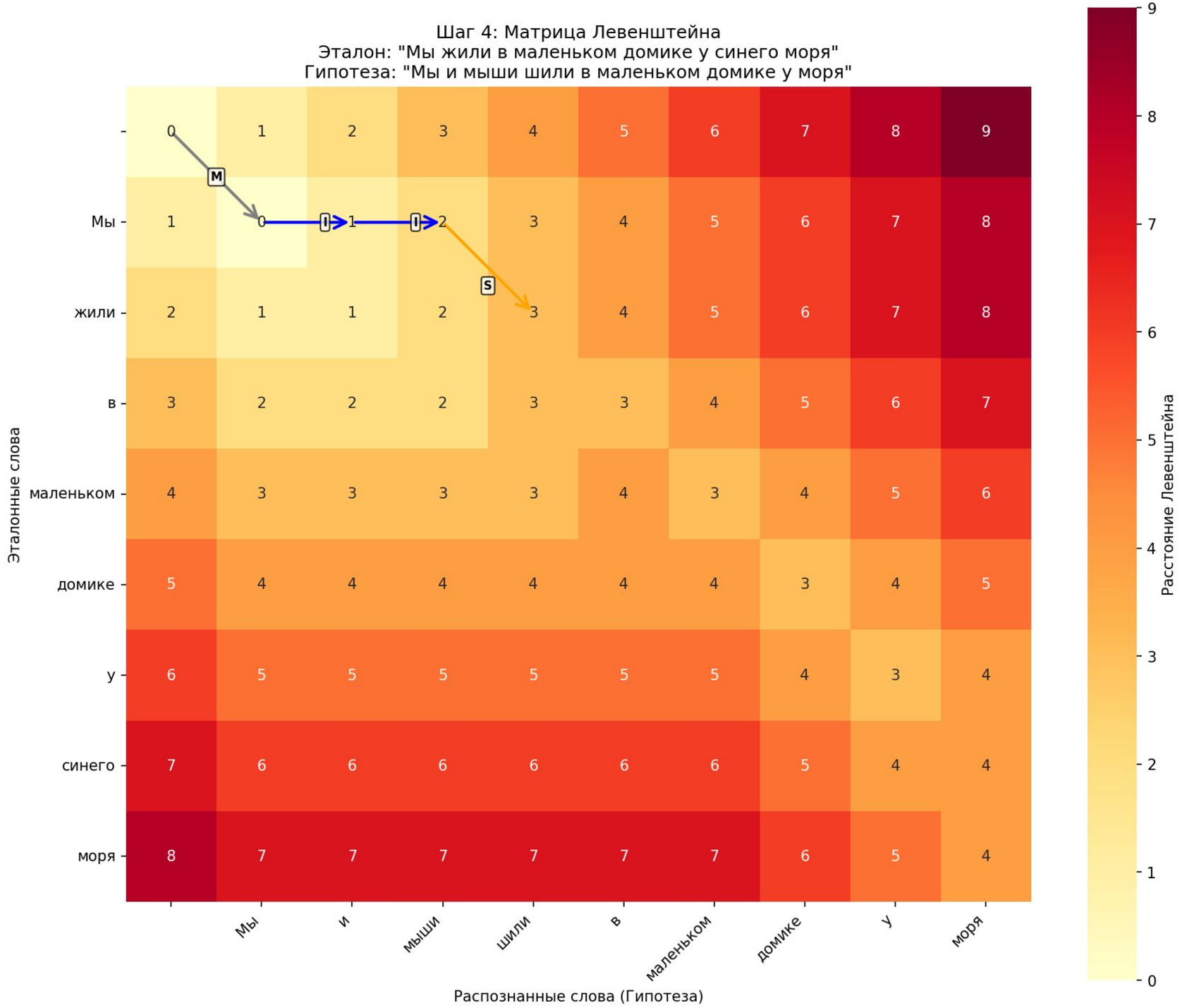
Алгоритм подсчета S, D, I



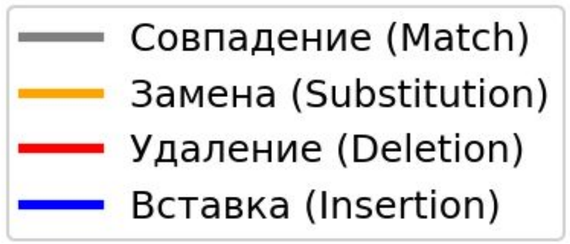


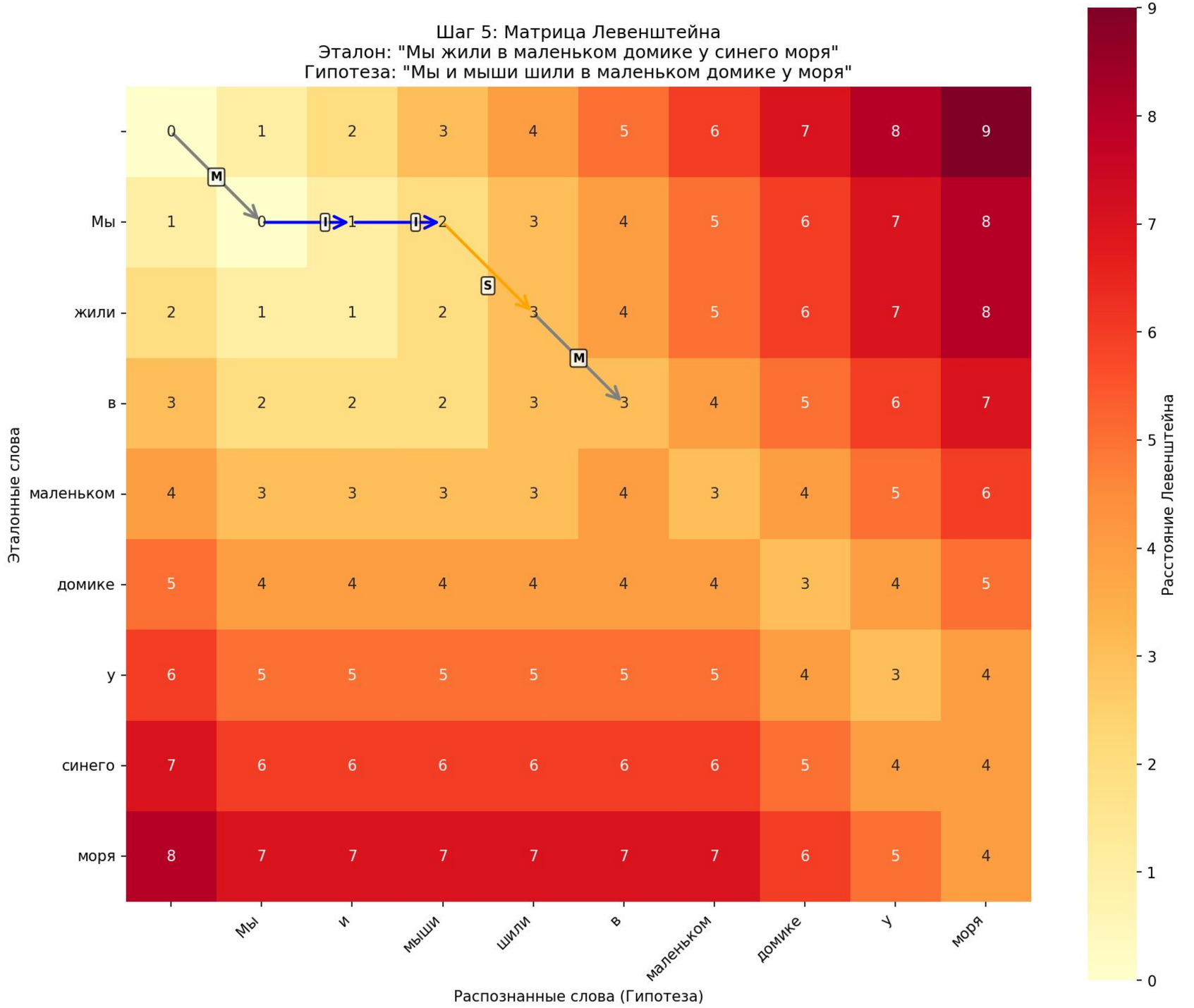
Алгоритм подсчета S, D, I



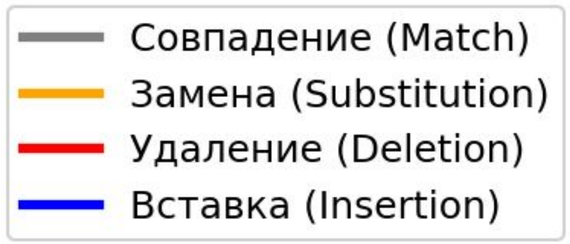


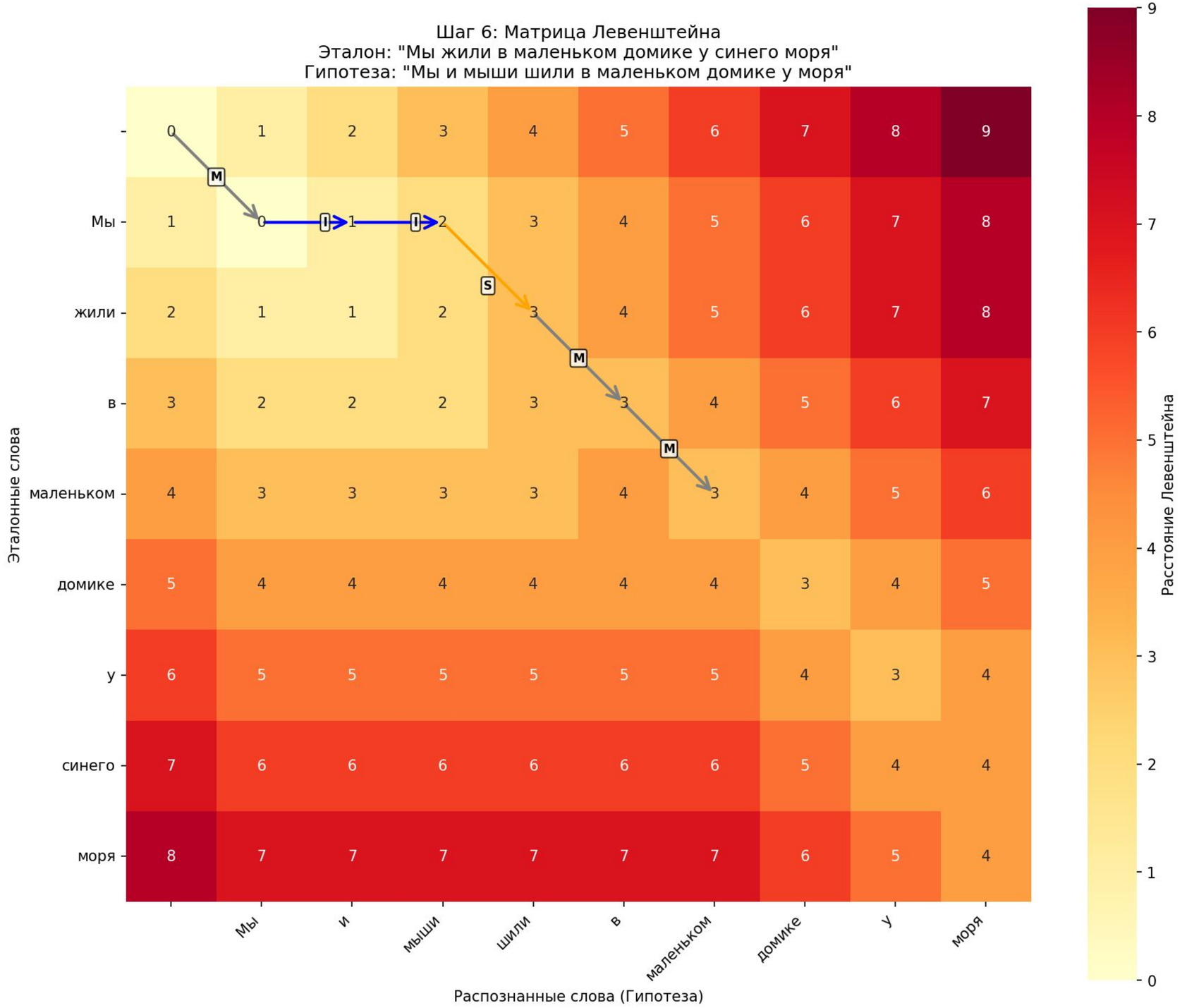
Алгоритм подсчета S, D, I



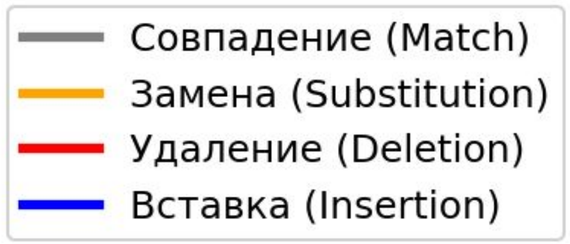


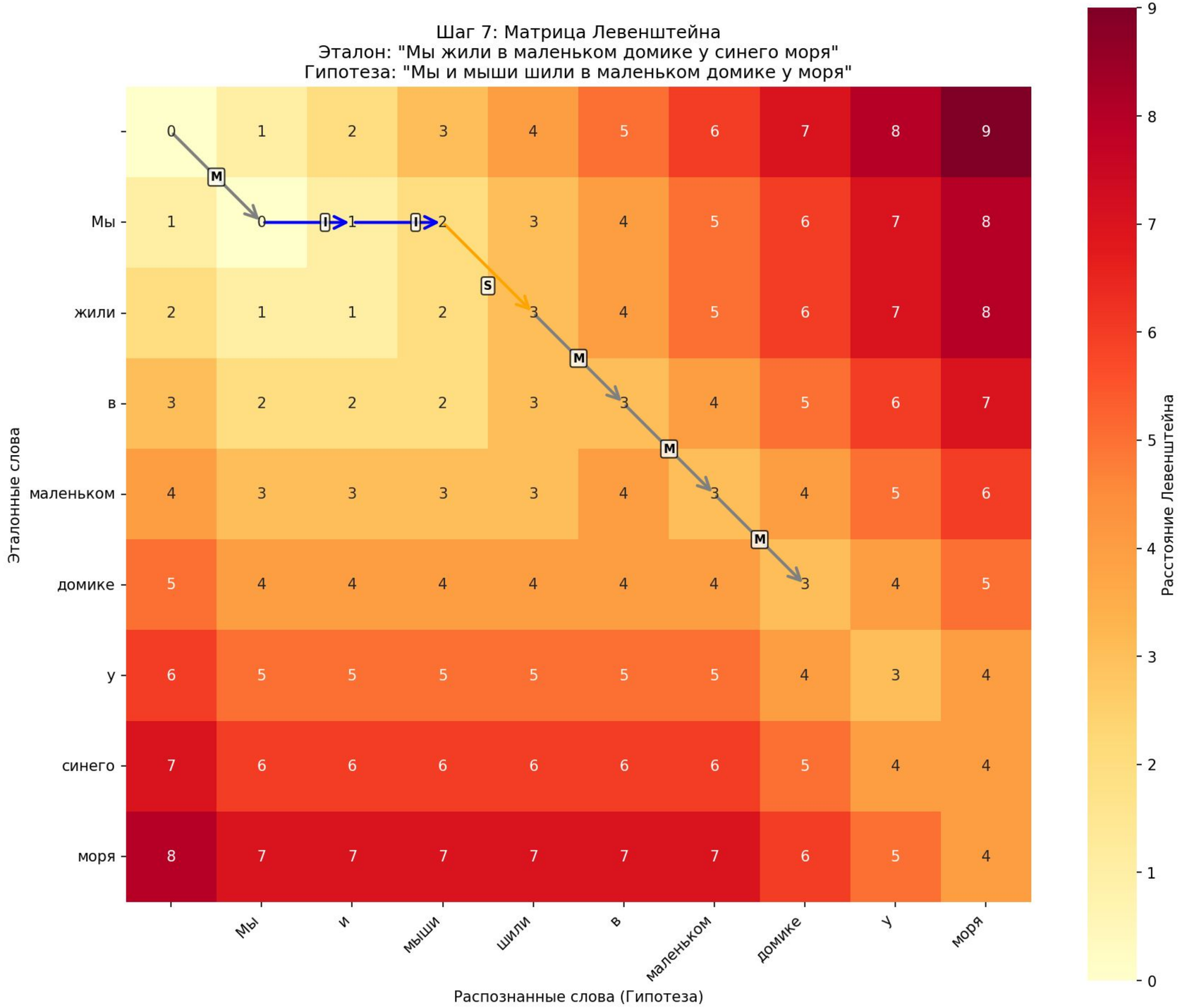
Алгоритм подсчета S, D, I



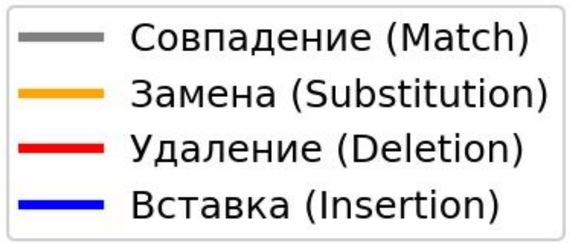


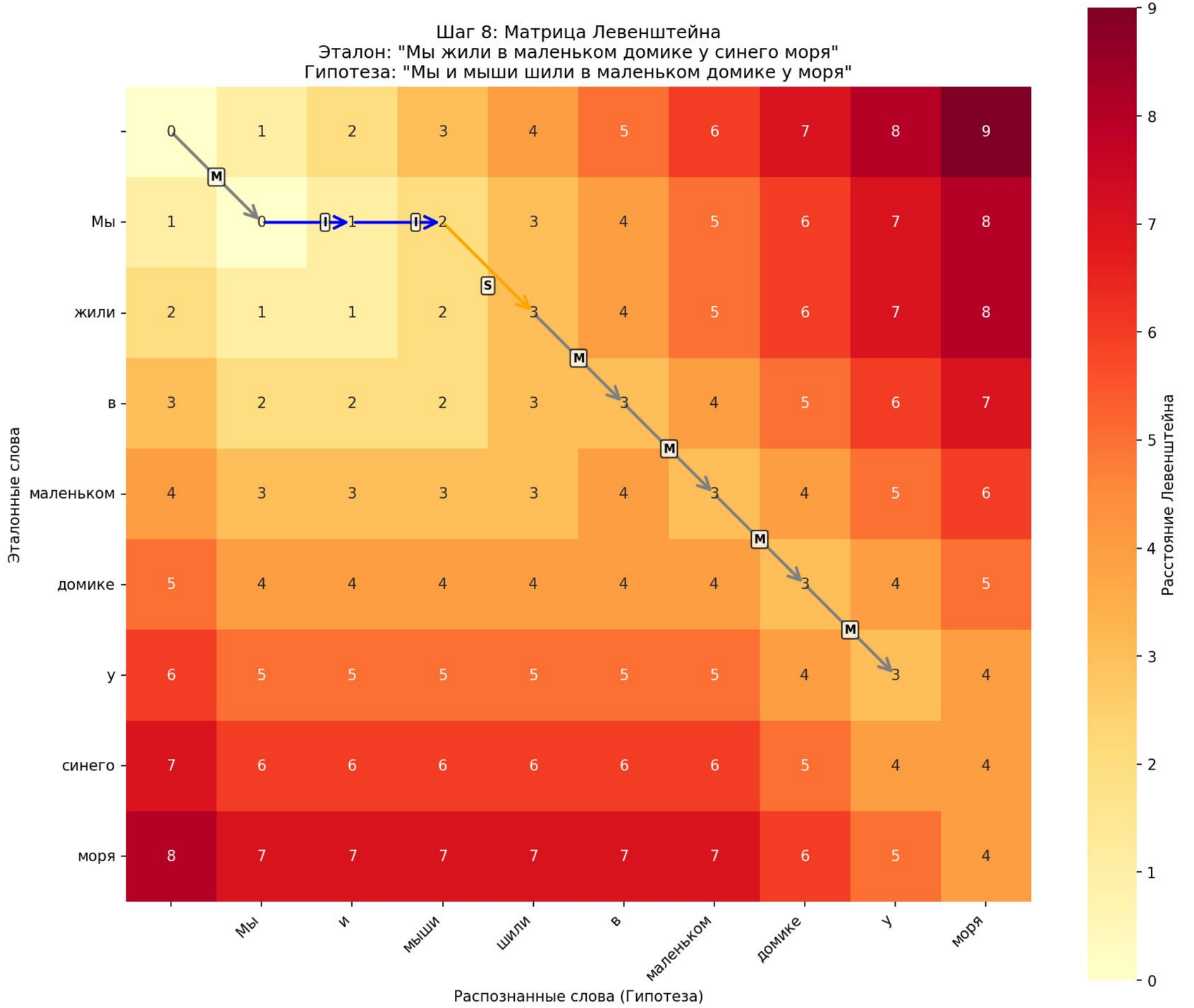
Алгоритм подсчета S, D, I



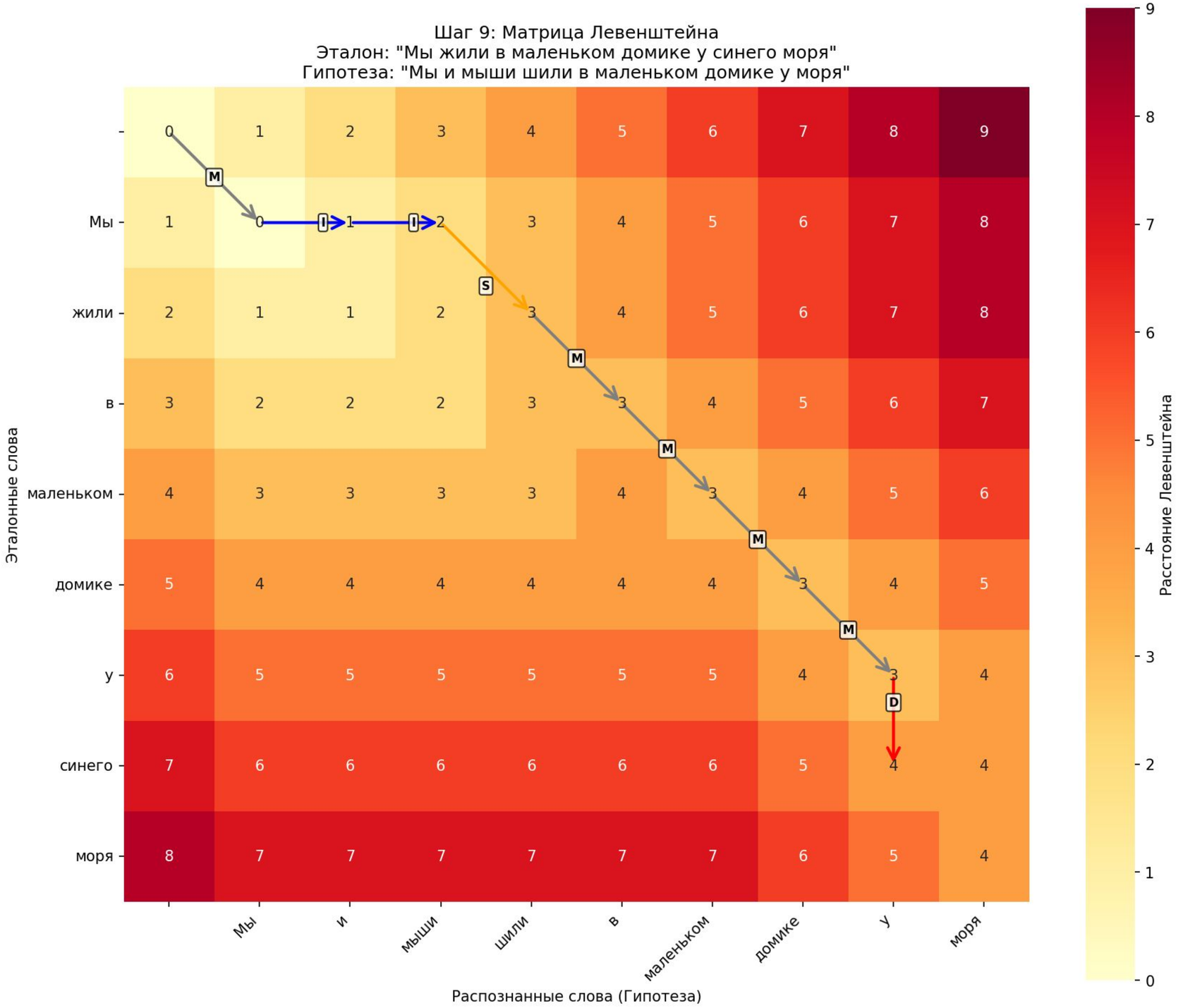


Алгоритм подсчета S, D, I

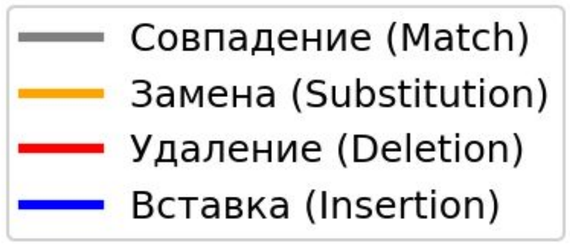


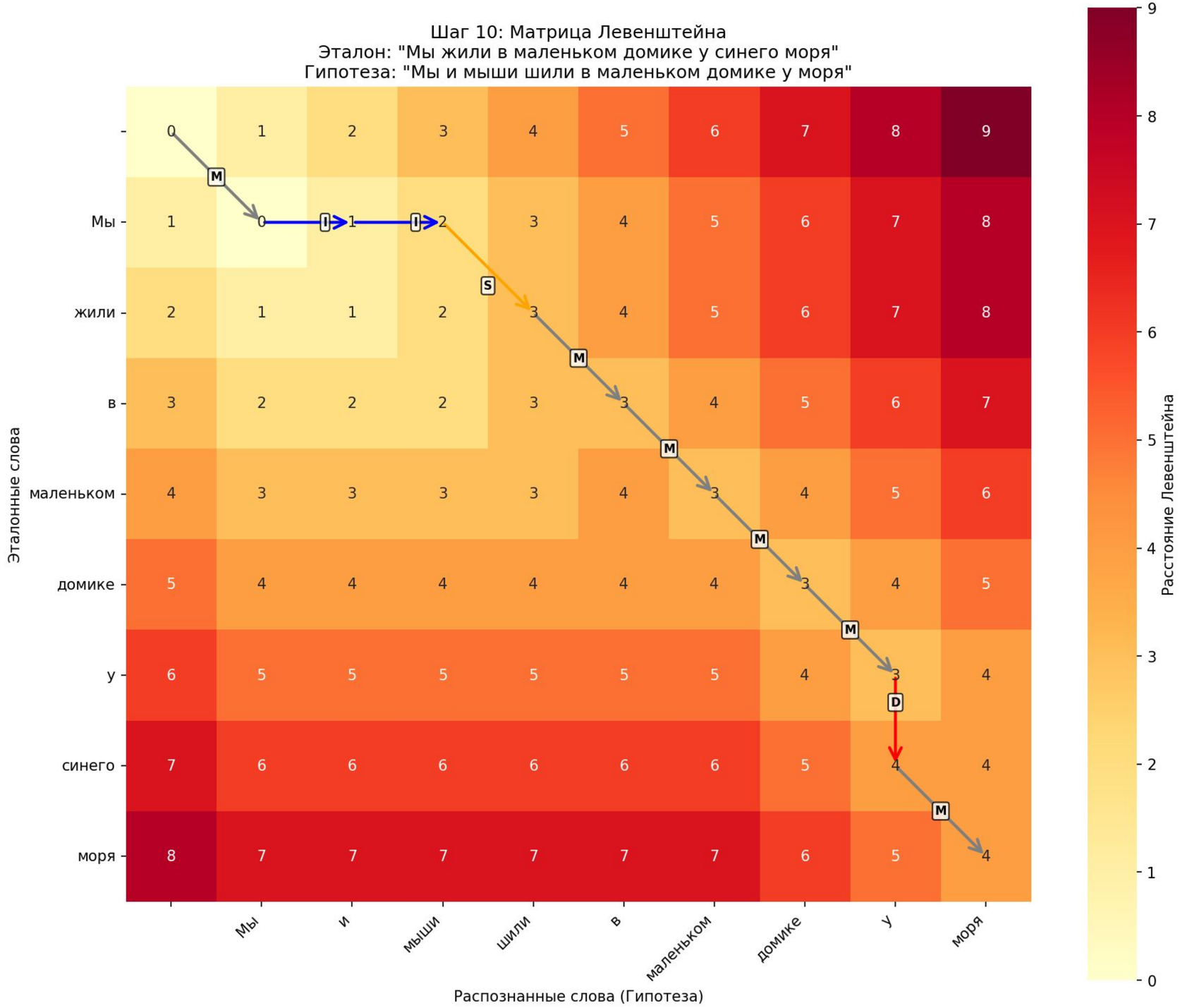


Алгоритм подсчета S, D, I



Алгоритм подсчета S, D, I



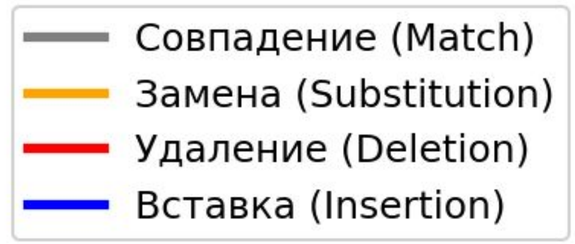


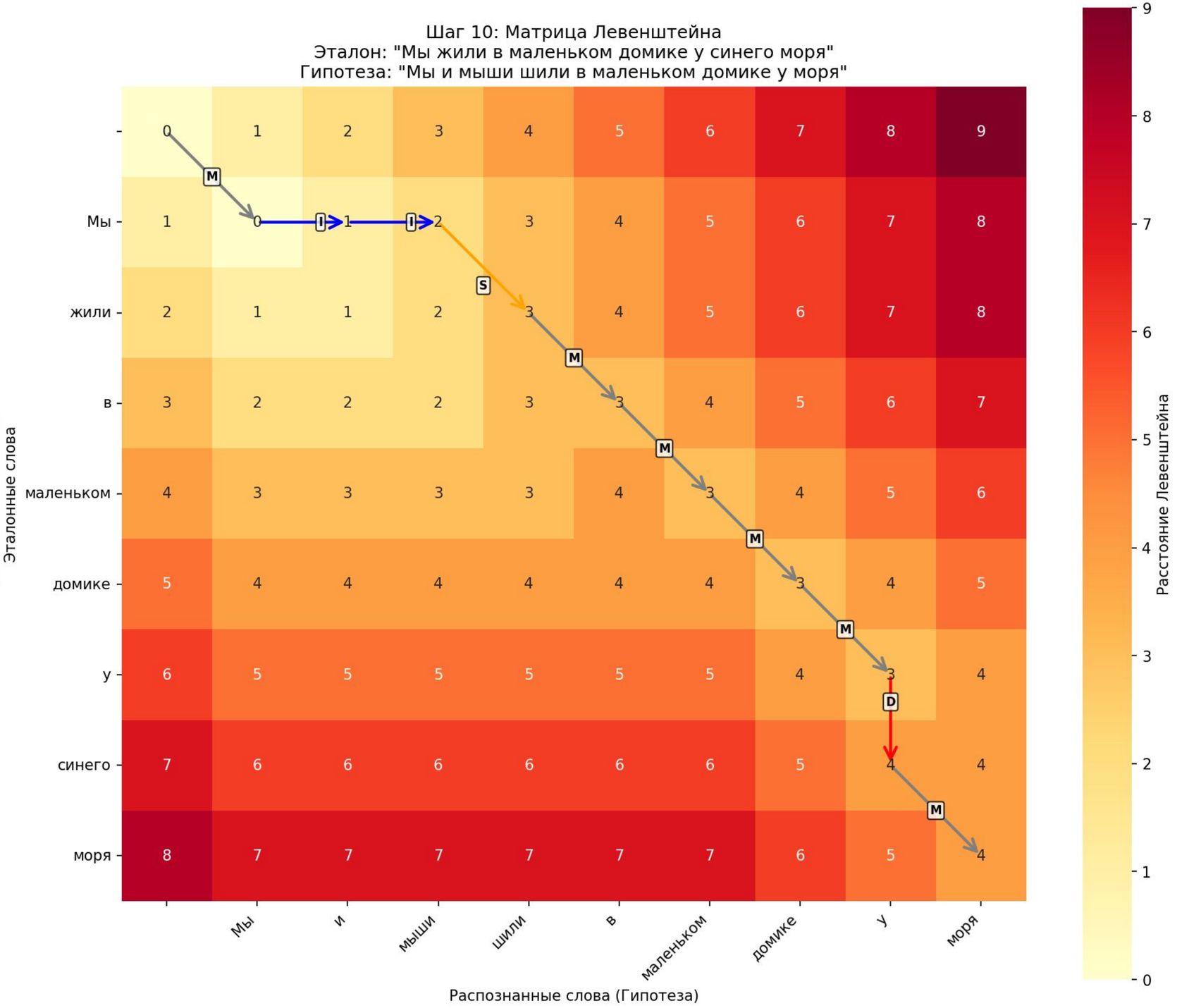
Алгоритм подсчета S, D, I

WER ИНФОРМАЦИЯ

Эталон: "Мы жили в маленьком домике у синего моря"
Гипотеза: "Мы и мыши шили в маленьком домике у моря"

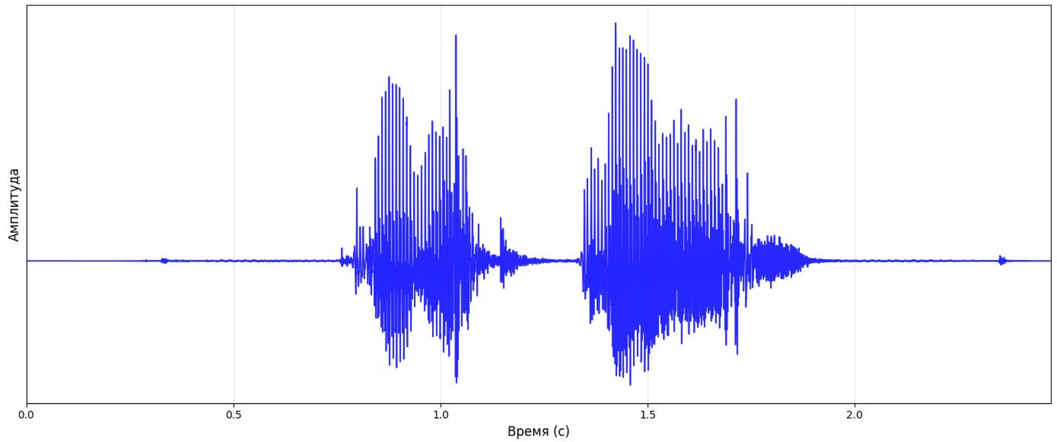
Расстояние Левенштейна: 4
Слов в эталоне: 8
WER = 4/8 = 50.0%





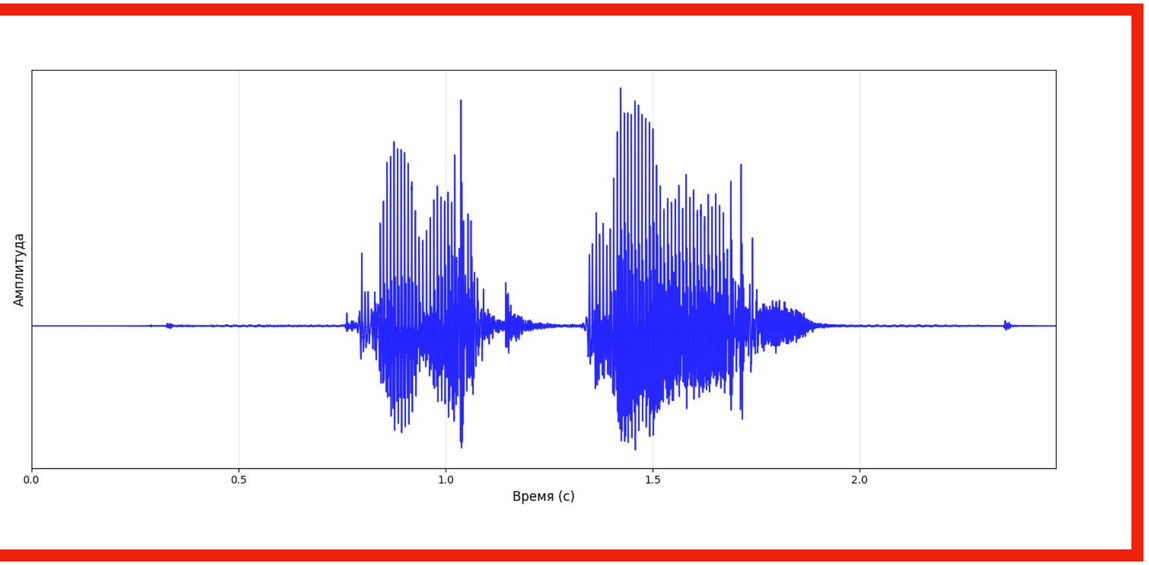
Проблема выравнивания

Связь аудио и текста



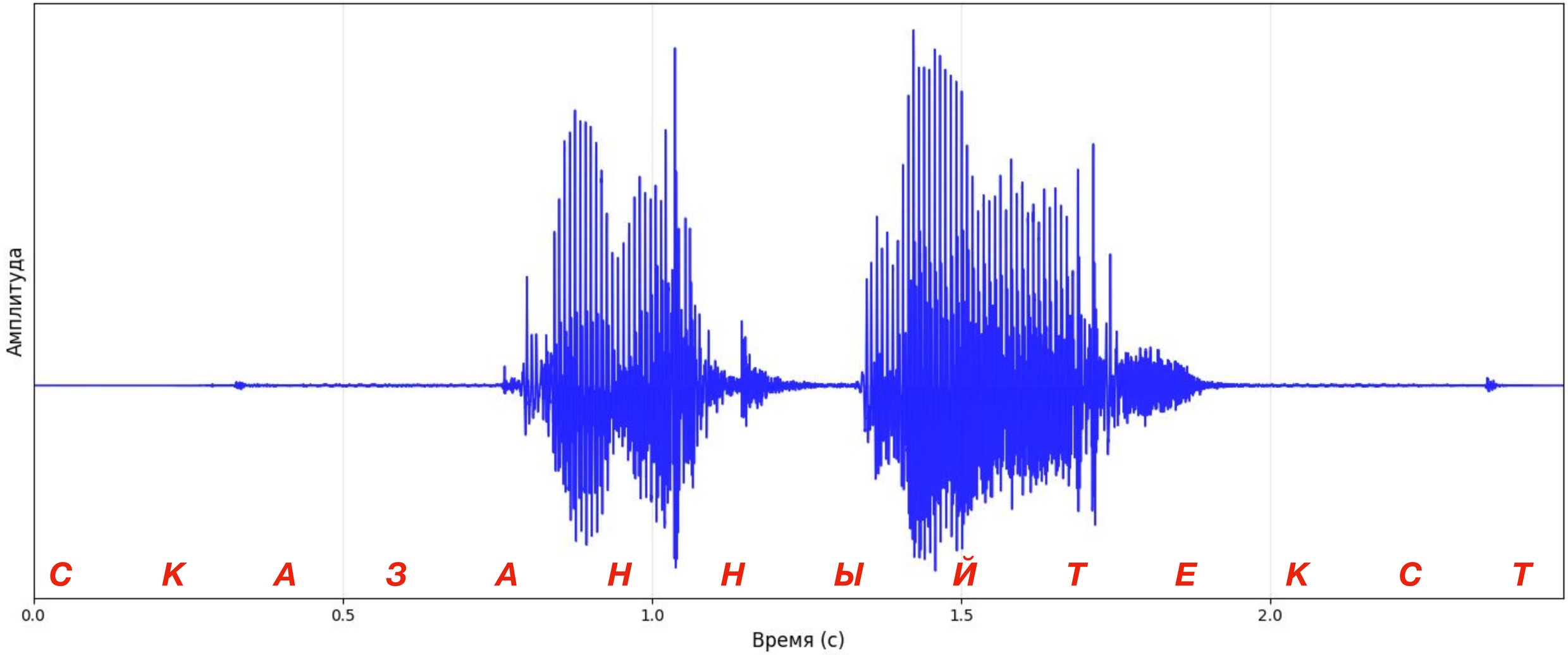
СКАЗАННЫЙ ТЕКСТ

Связь аудио и текста

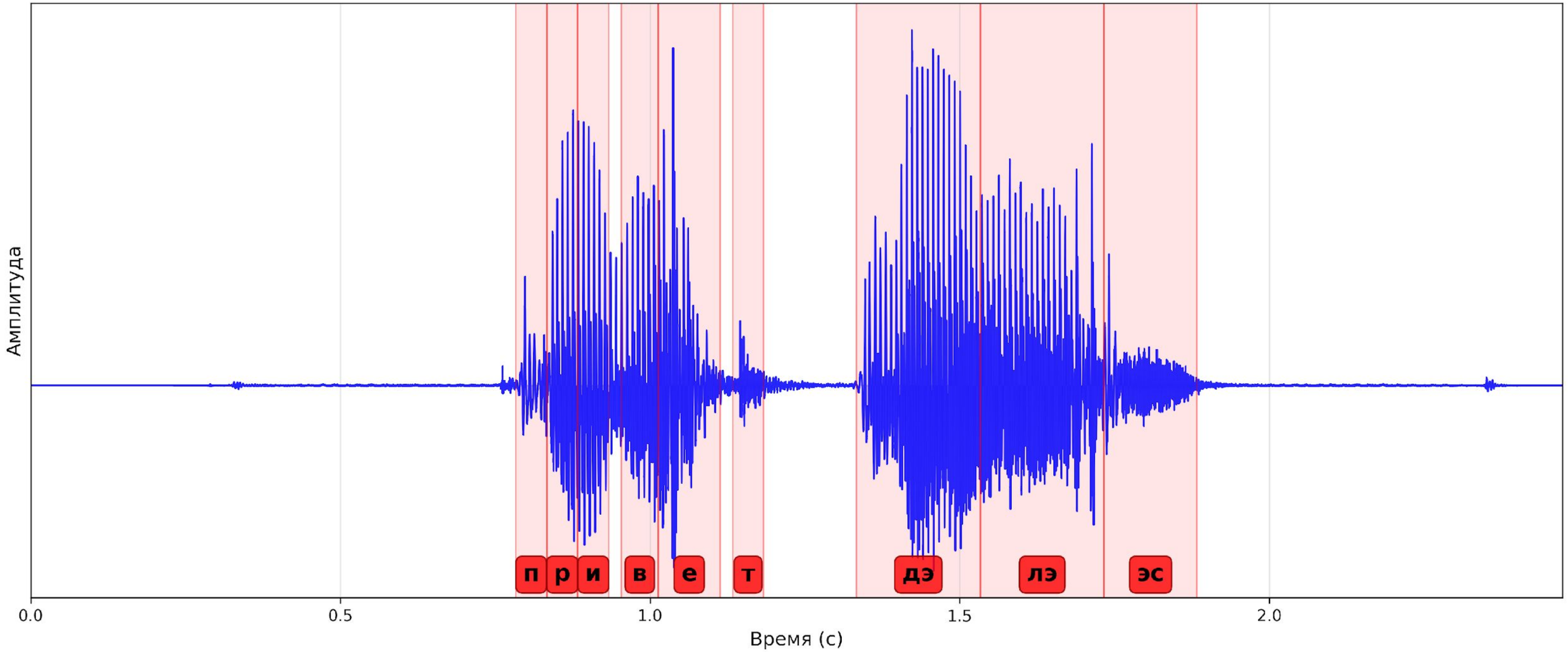


СКАЗАННЫЙ ТЕКСТ

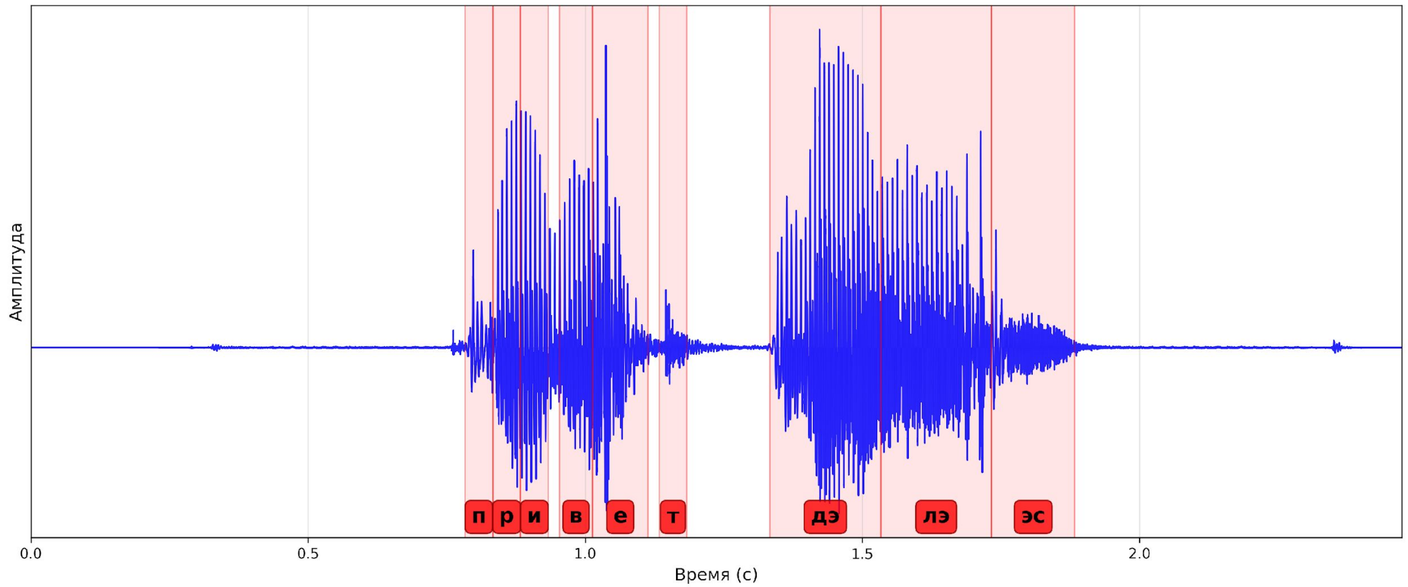
Связь аудио и текста



Связь аудио и текста



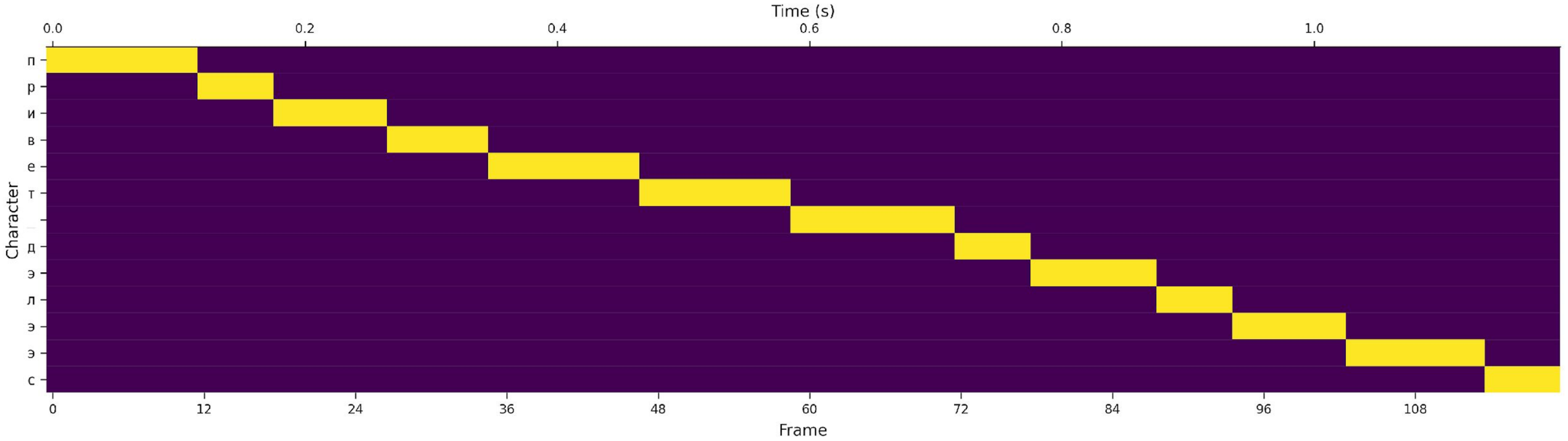
Связь аудио и текста



Проблемы:

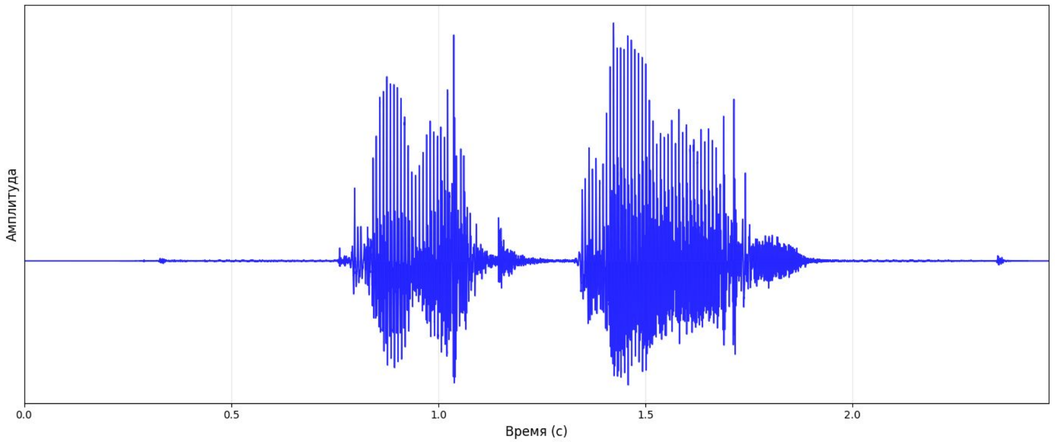
- каждая фонема имеет разную длительность
- точная разметка звуков порой невозможна
- стоимость такой разметки крайне высока

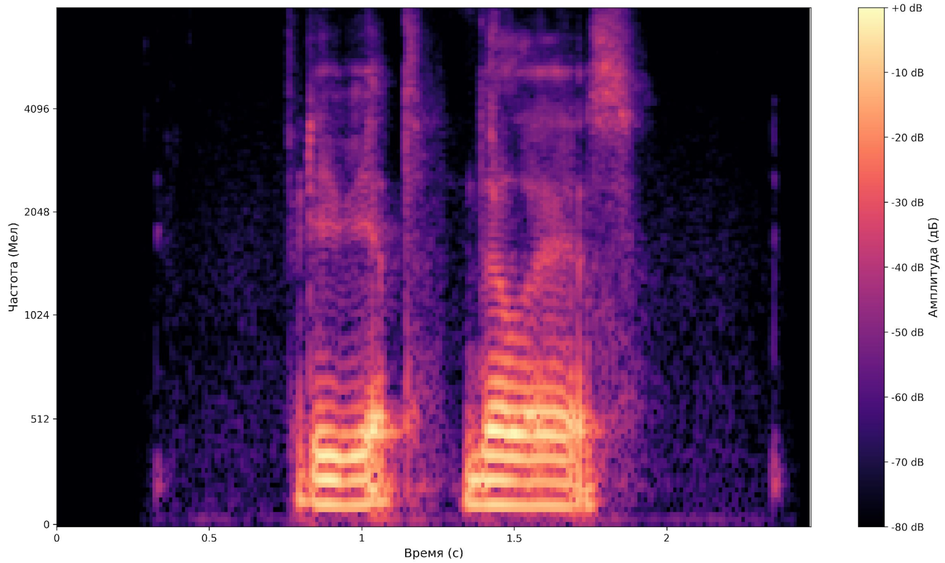
seq2seq задача



Терминология:
фрейм - сжатый фрагмент сигнала
(по аналогии с токеном в NLP)

Проблема размерности



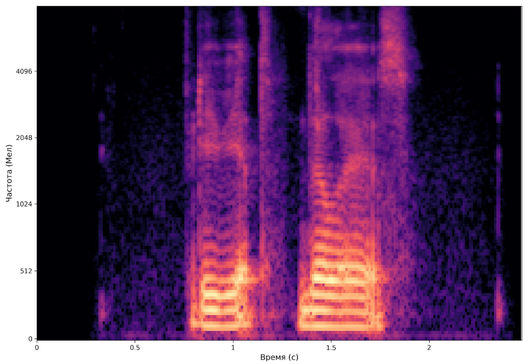


1D wavefom
16000 чисел в секунде

2D mel-spectrum
100 векторов в секунде

seq2seq задача

“привет дэлээс”



2D mel-spectrum
100 векторов в секунде

Encoder

CNN +
LSTM/Transformer

Linear projection

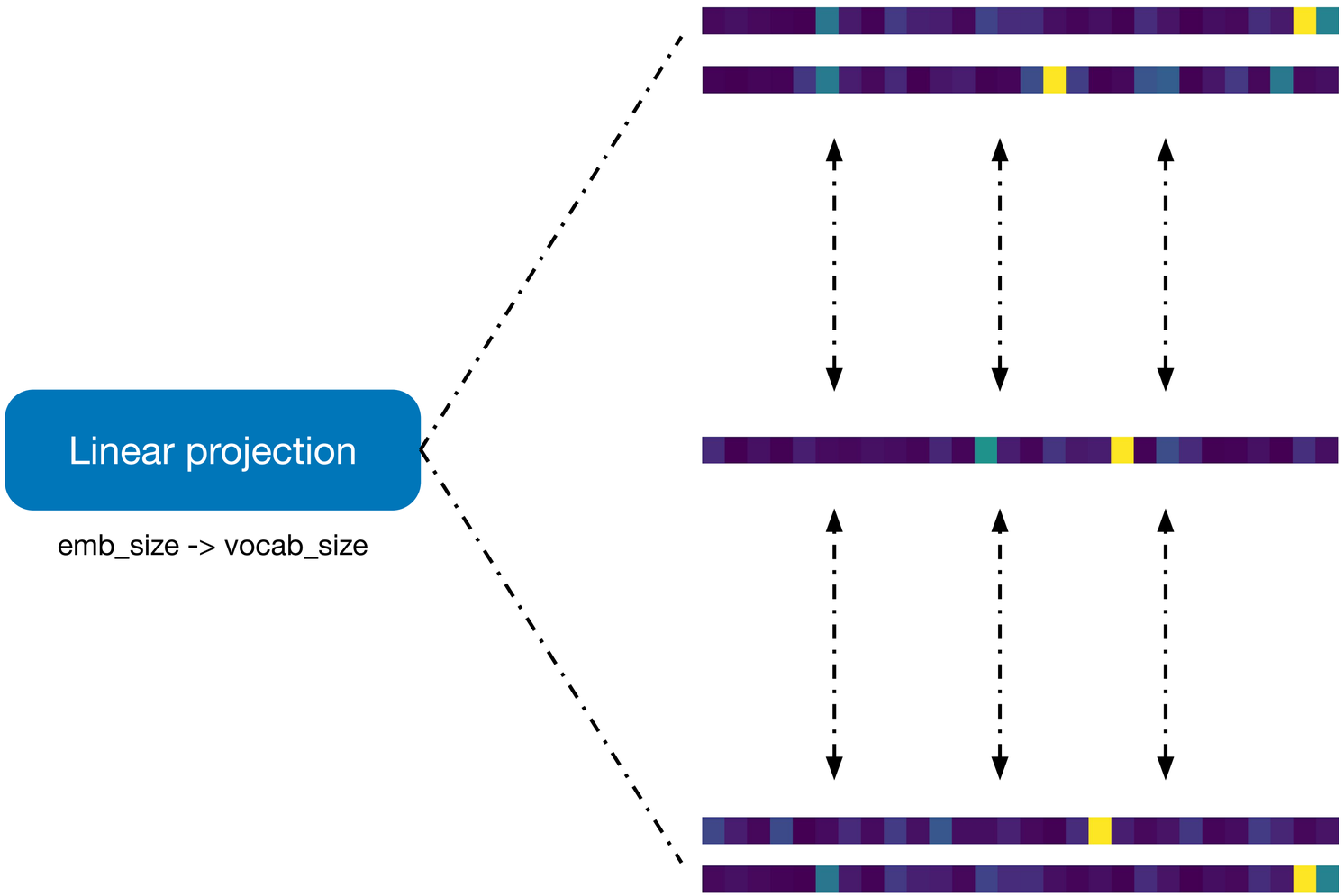
emb_size -> vocab_size

seq2seq задача

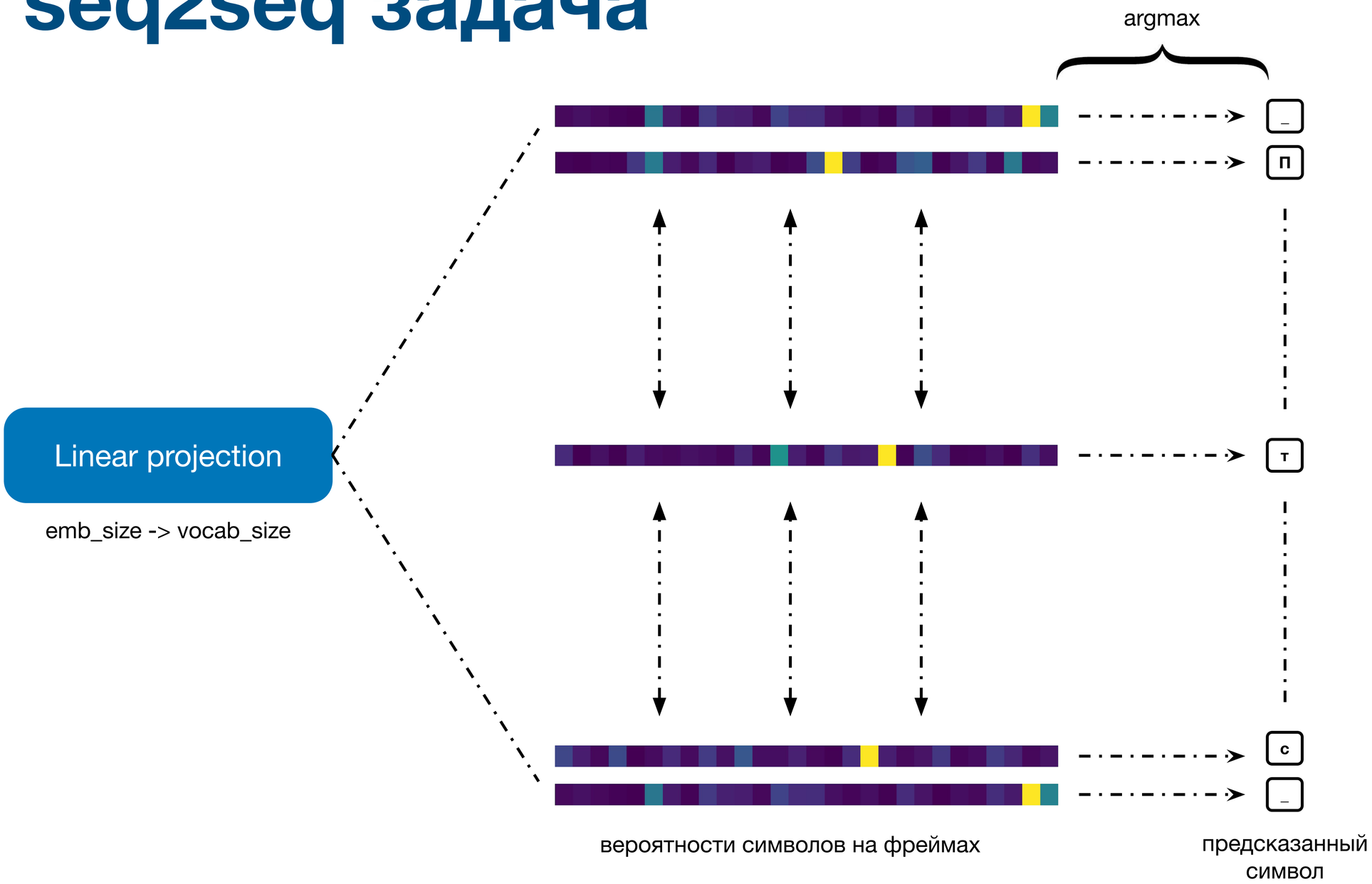
Linear projection

emb_size -> vocab_size

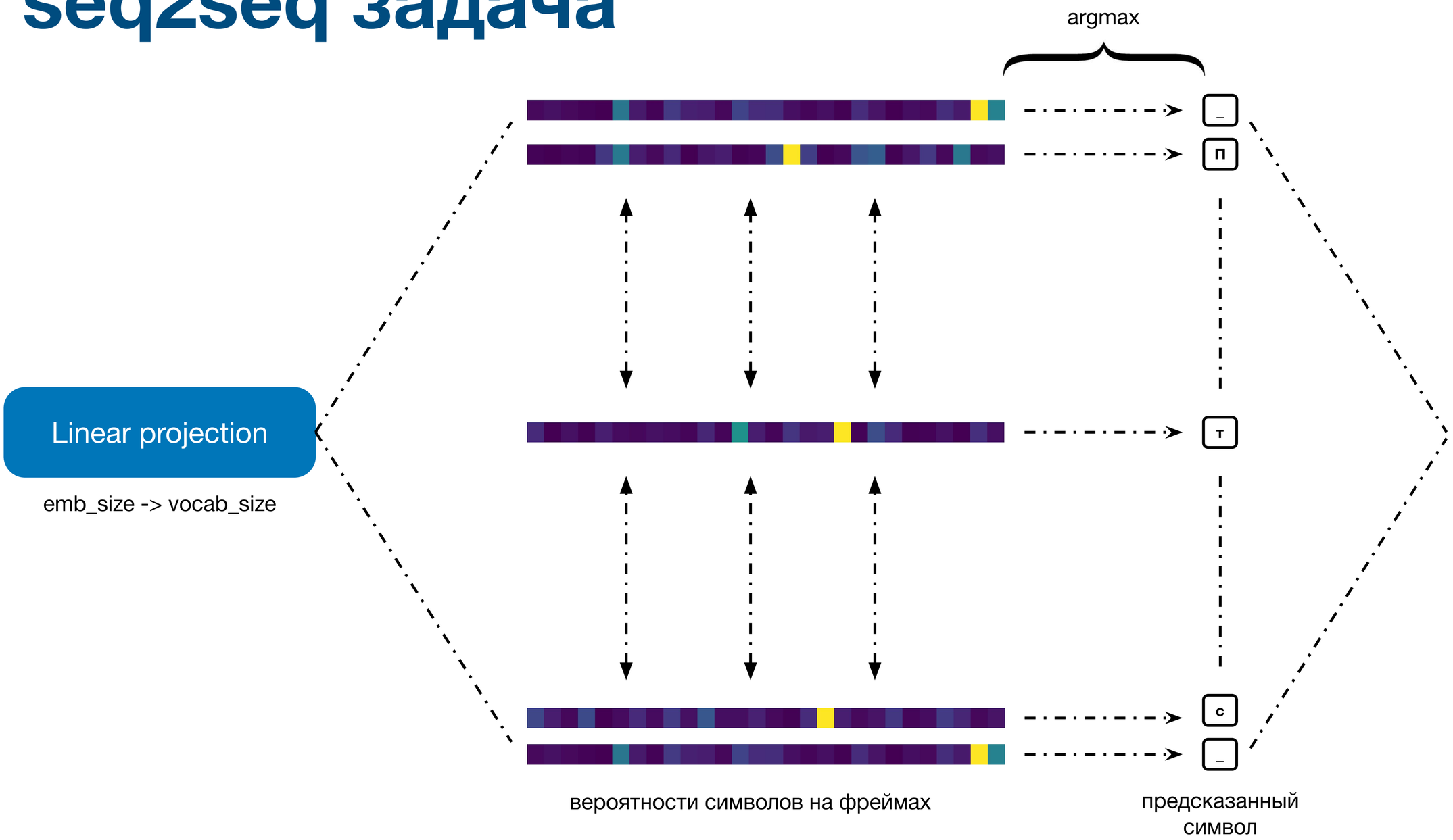
seq2seq задача



seq2seq задача



seq2seq задача



_ _ _ п п п р р и и в в е е т т т _ _ _ д э э э л э э с с с _ _

Пост-обработка текста

___пппрррриииивввееееттт___дээээлээээссс__

Пост-обработка текста

___пппрррриииивввееееттт___дээээлээээссс__

привет_дэлэс

Пост-обработка текста

___пппрррриииивввееееттт___дээээлээээссс__

привет_дэлэс

Пост-обработка текста

___пппрррриииивввееееттт___дээээлээээссс__

привет_дэлэс

привет_дэлэээс

Blank токен

___пппрррриииивввееееттт___дээээлээээссс__

___пппрррриииивввееееттт___дээээлэ|ээссс__

привет_дэлэээс

Blank токен

|||ппп|ррр|иии|ввв|еее|ттт|||_||дэээ|лээ|эссс|||

Blank токен

|||||п|||р|||и|вв||||е|||т|||||_||||||дэээ|||лээ|||||эс|||

Blank токен

|||||п|||р|||и|вв||||е|||т|||||_||||||дээээ|||лээ|||||эс|||

|п|р|и|в|е|т|_|д|э|л|э|э|с|

Blank токен

|||||п|||р|||и|вв||||е|||т|||||_||||||дээээ|||лээ|||||эс|||

|п|р|и|в|е|т|_|д|э|л|э|э|с|

привет_дэлээс

Проблема разных предсказаний

|||ппп|ррр|иии|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппп**|**рр|иии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппп**|**р|иии|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппппп|р**р|**ии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппппп|р**рр|**и|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппппппппппппппппппп|р|и|в|е|т|_|дэ|лэ|эс|
|

Проблема разных предсказаний

|||ппп|ррр|иии|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппп|рр|иии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппп|р|иии|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппппп|рр|ии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппппп|ррр|и|ввв|еее|ттт||_|дэээ|лэ|эээсссс||

|||пппппппппппппппппппп|р|и|в|е|т|_|дэ|лэ|эс|
|

привет дэлээс

Проблема разных предсказаний

|||ппп|ррр|иии|ввв|еее|ттт||_|дэээ|лэ|эээсс||

|||пппп|рр|иии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппп|р|иии|ввв|еее|ттт||_|дэээ|лэ|эээсс||

|||пппппп|рр|ии|ввв|еее|ттт|_||дэээ|лээ|эссс||

|||пппппп|ррр|и|ввв|еее|ттт||_|дэээ|лэ|эээсс||

|||пппппппппппппппппп|р|и|в|е|т|_|дэ|лэ|эс|
|

привет дэлээс

CTC лосс

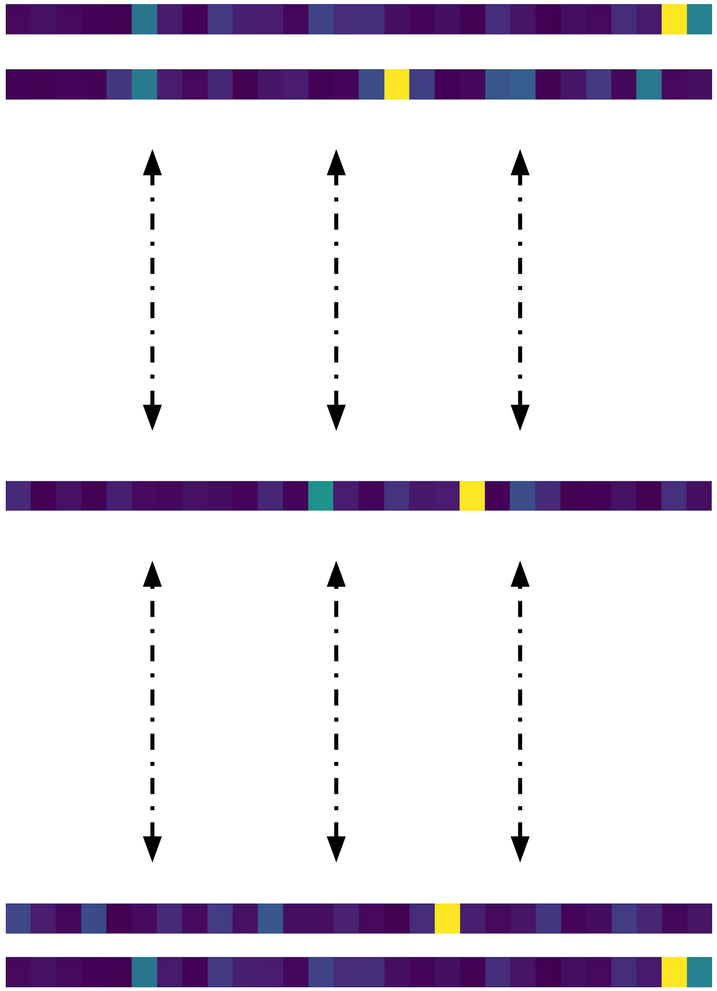
CTC лосс

$$L(Y, \hat{Y}) = -\log \left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y}) \right)$$

CTC лосс

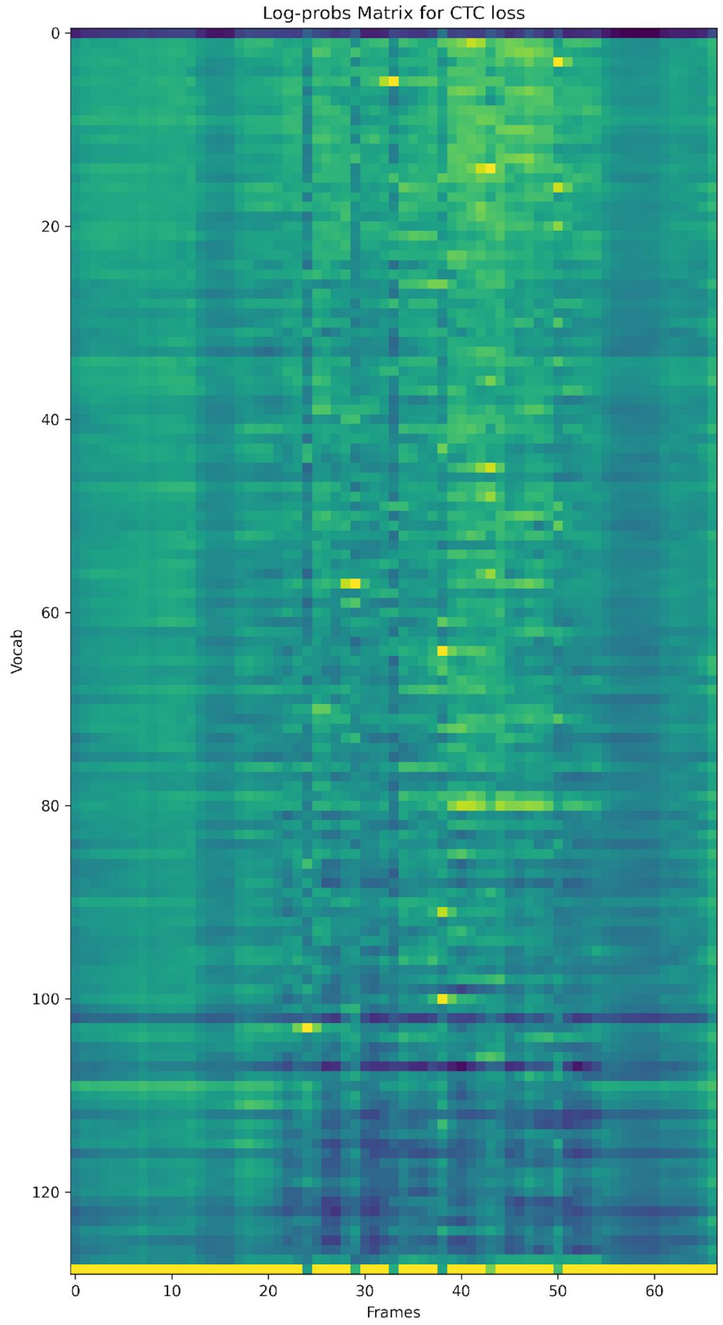
$$L(Y, \hat{Y}) = -\log\left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y})\right)$$

CTC лосс

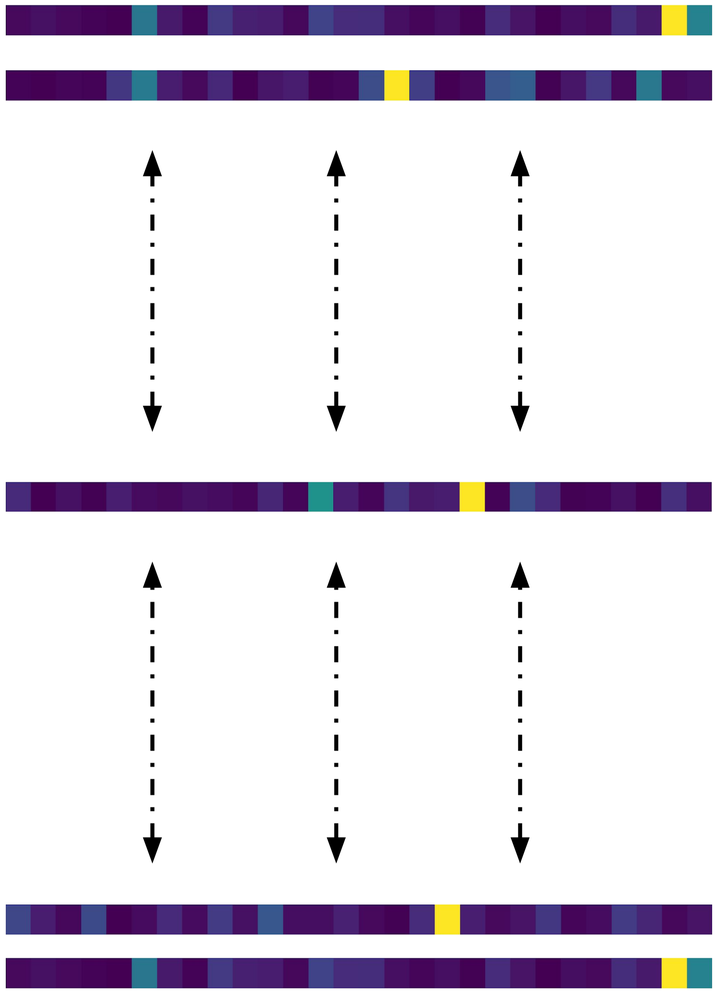


~~вероятности~~ символов на фреймах

Модель:
stt_ru_conformer_ctc_large

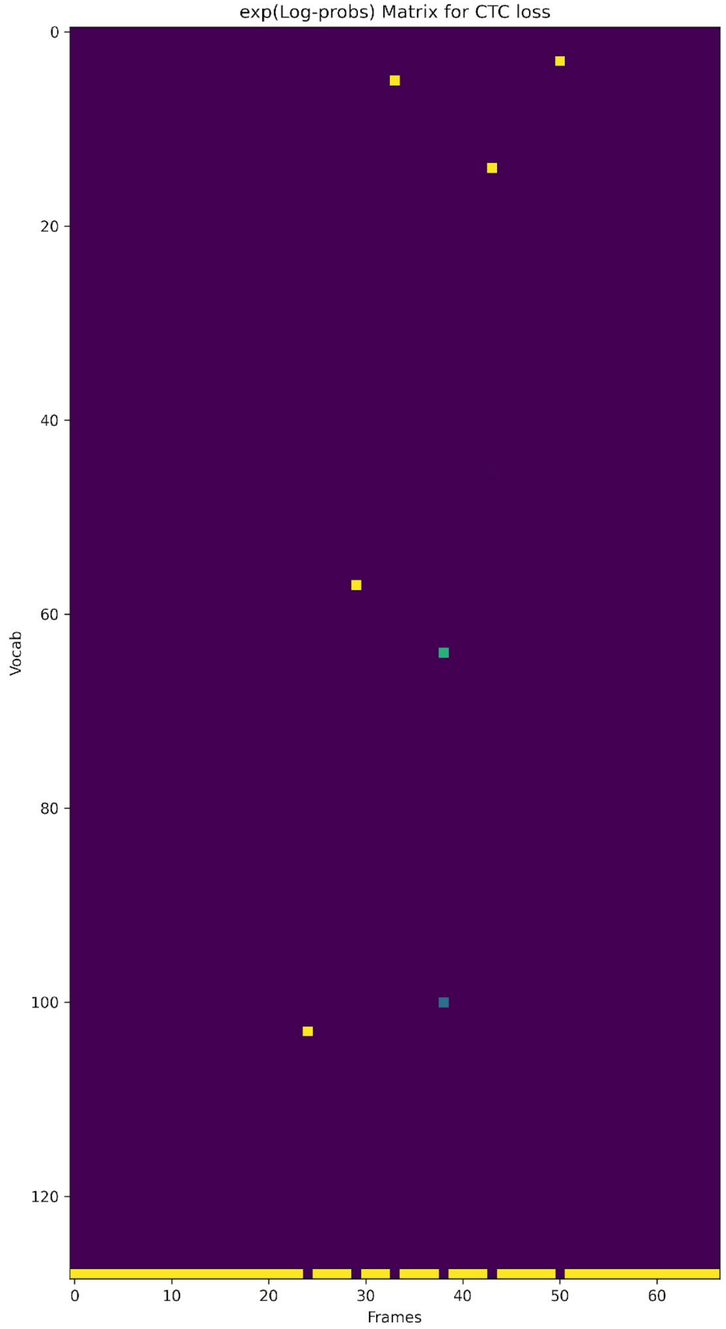


CTC лосс



вероятности символов на фреймах

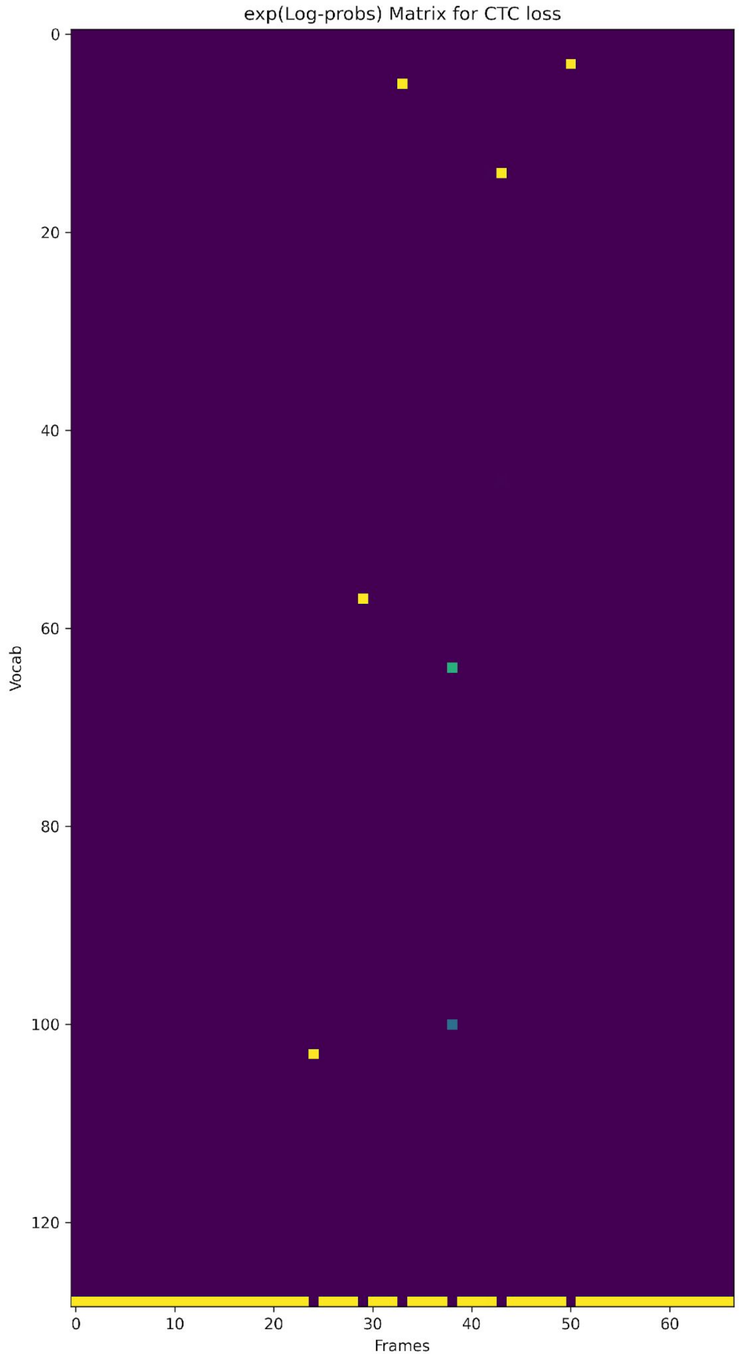
Модель:
stt_ru_conformer_ctc_large



CTC лосс

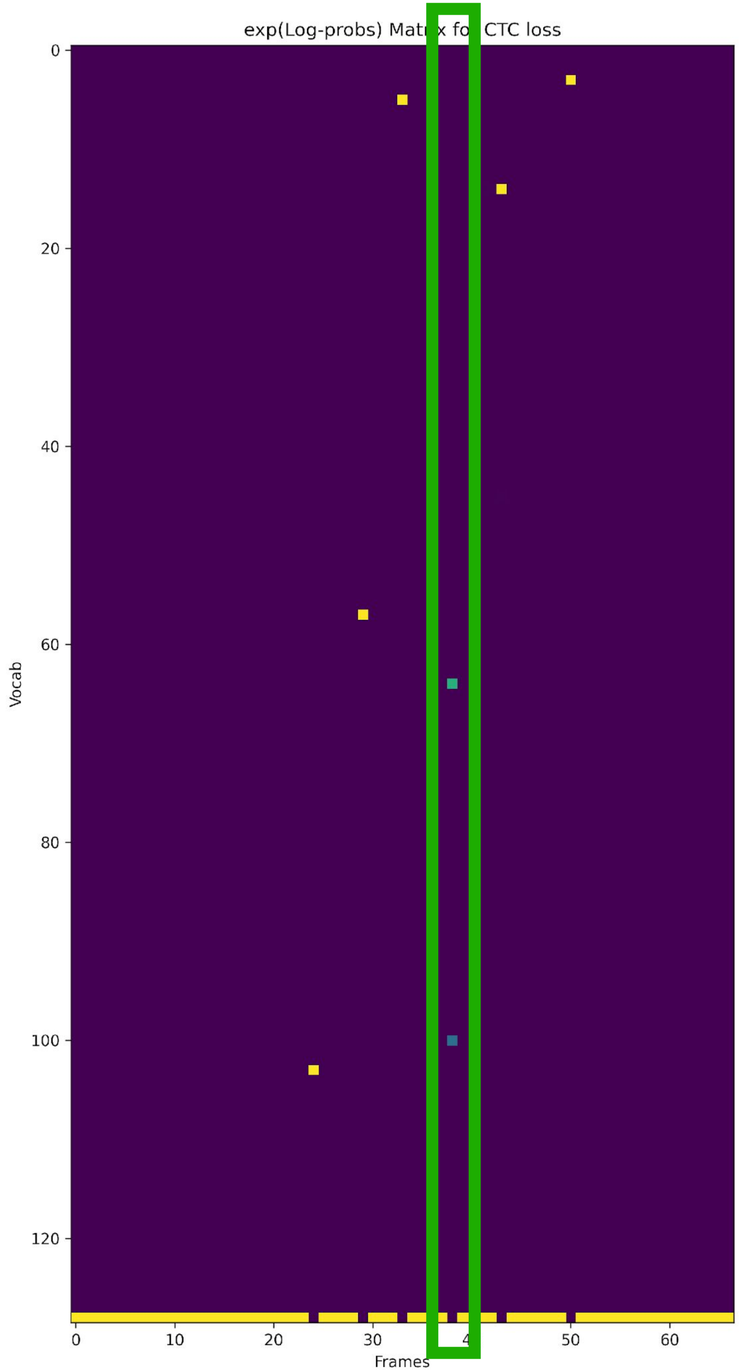
$$L(Y, \hat{Y}) = -\log \left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y}) \right)$$

CTC лосс



Модель: stt_ru_conformer_ctc_large

CTC лосс

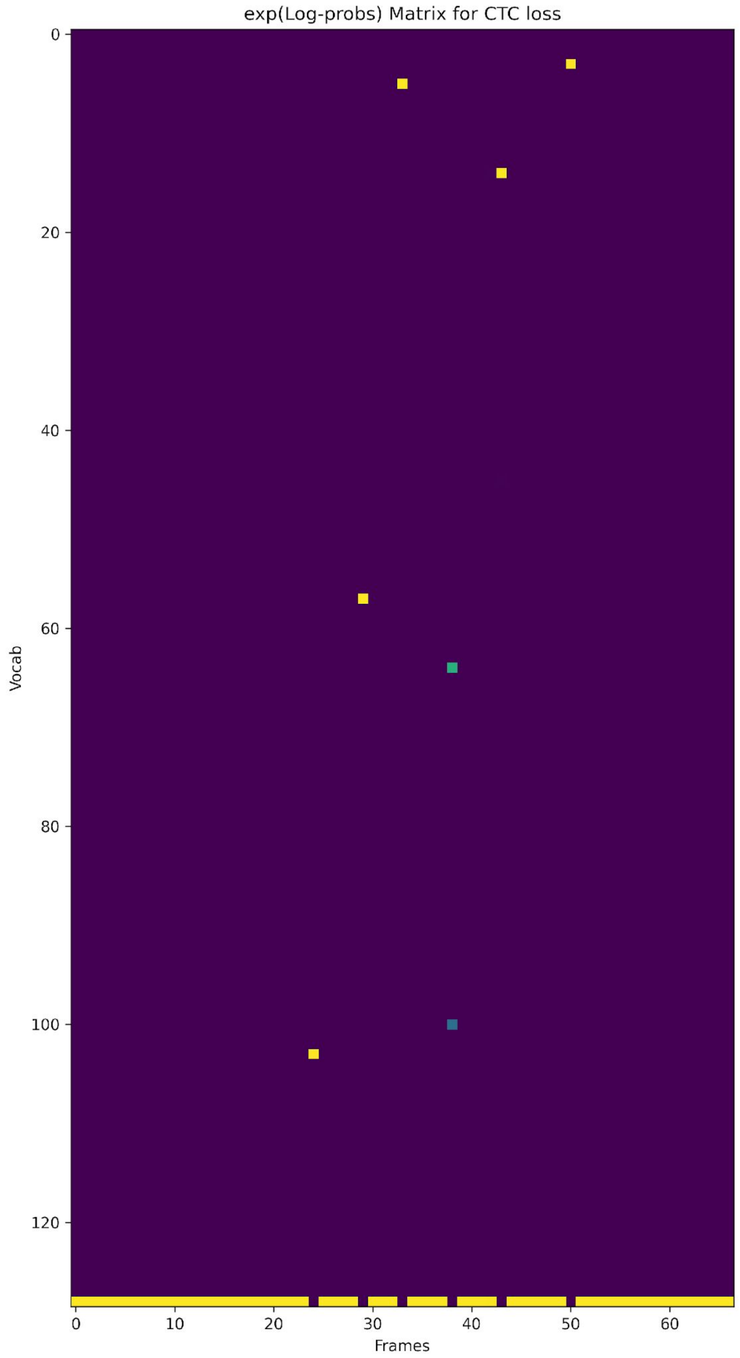


Модель: stt_ru_conformer_ctc_large

CTC лосс

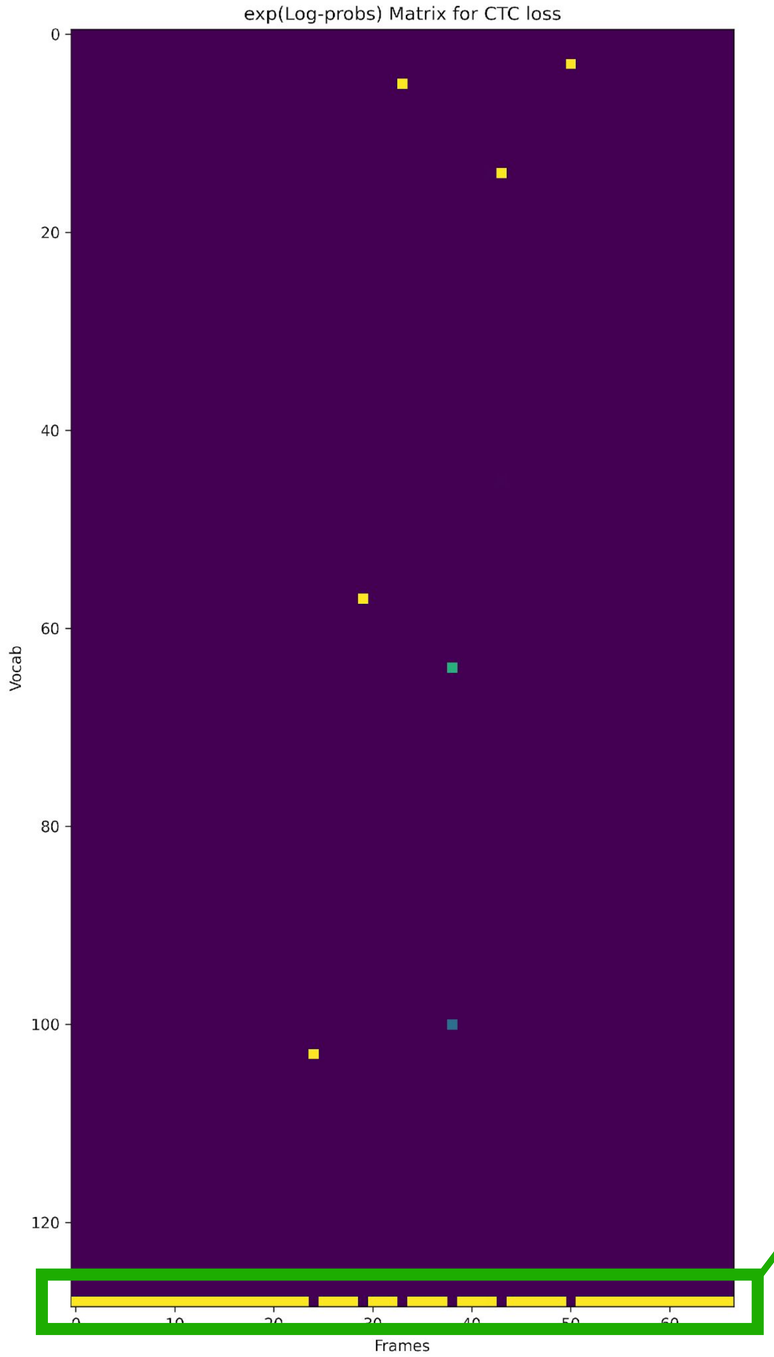
$$L(Y, \hat{Y}) = -\log\left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y})\right)$$

CTC лосс



Модель: stt_ru_conformer_ctc_large

CTC лосс

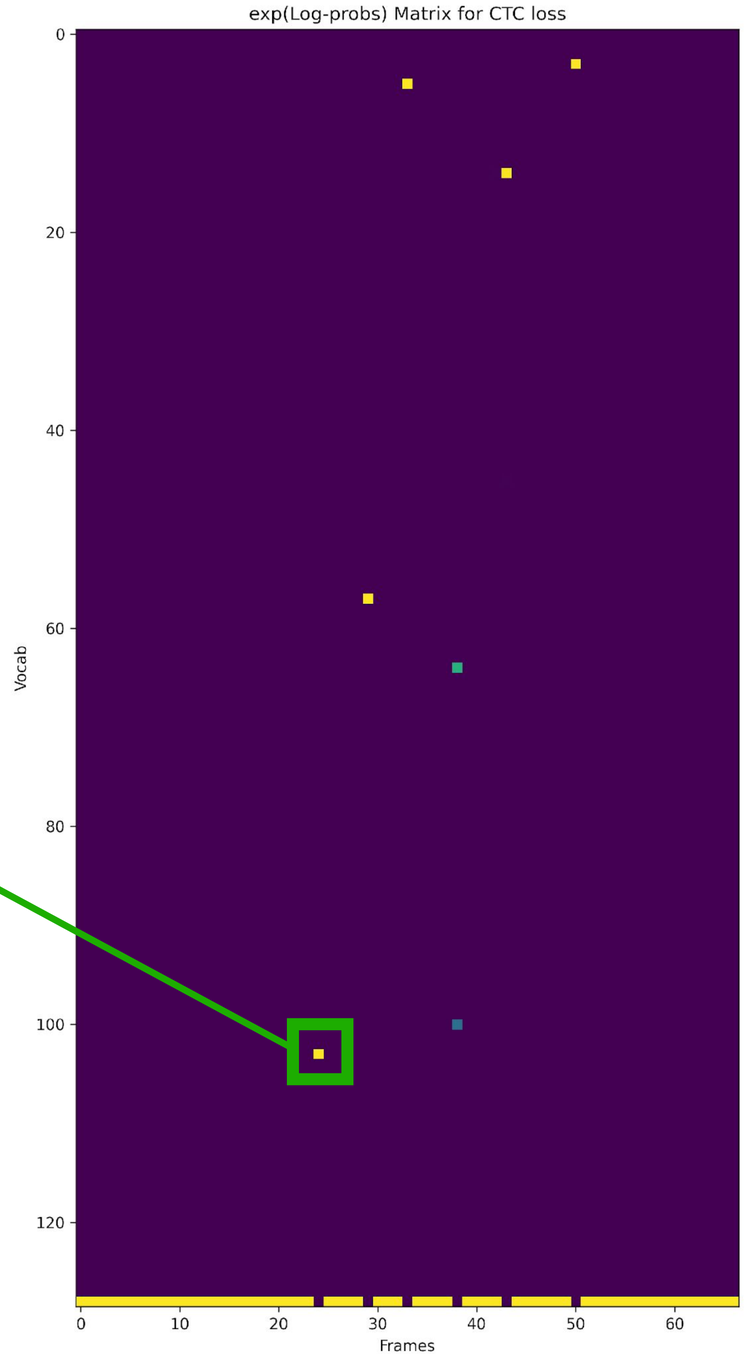


p(<blank token>)

Модель: stt_ru_conformer_ctc_large

CTC лосс

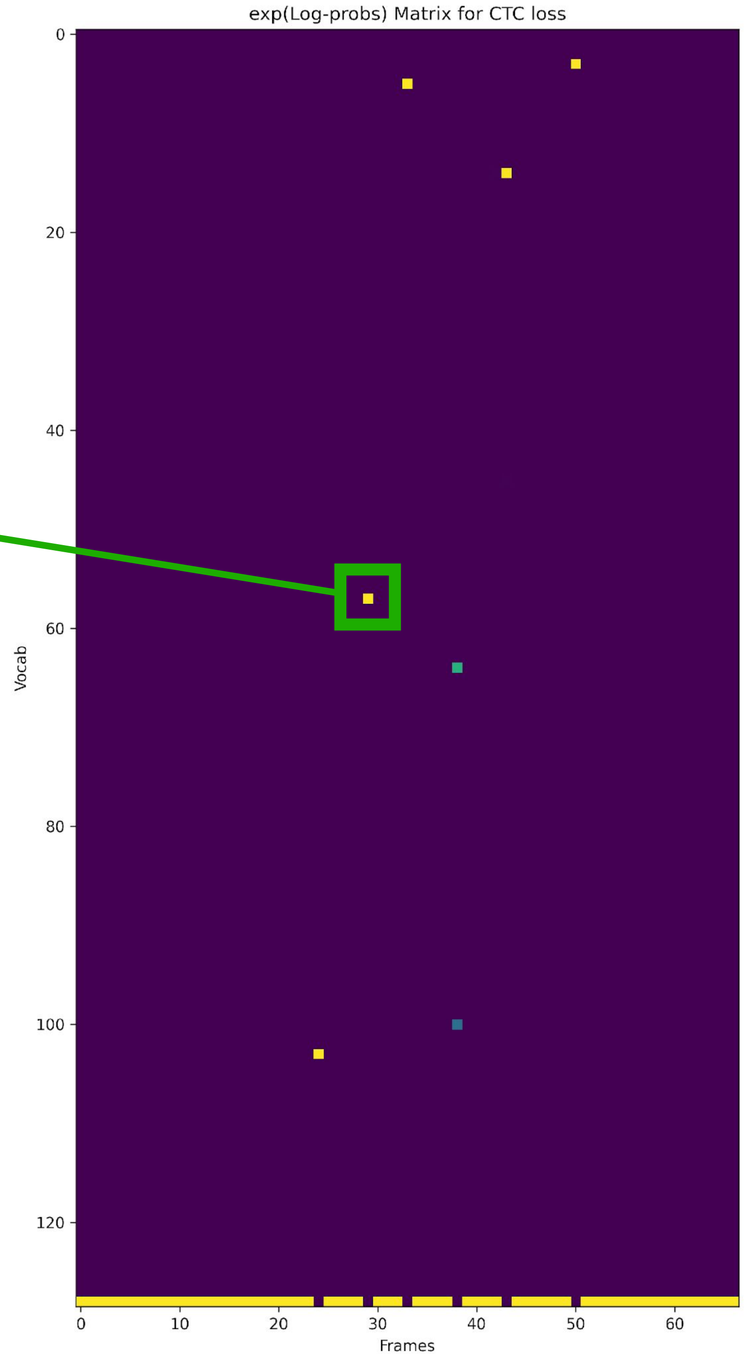
p(_при)



Модель: stt_ru_conformer_ctc_large

CTC лосс

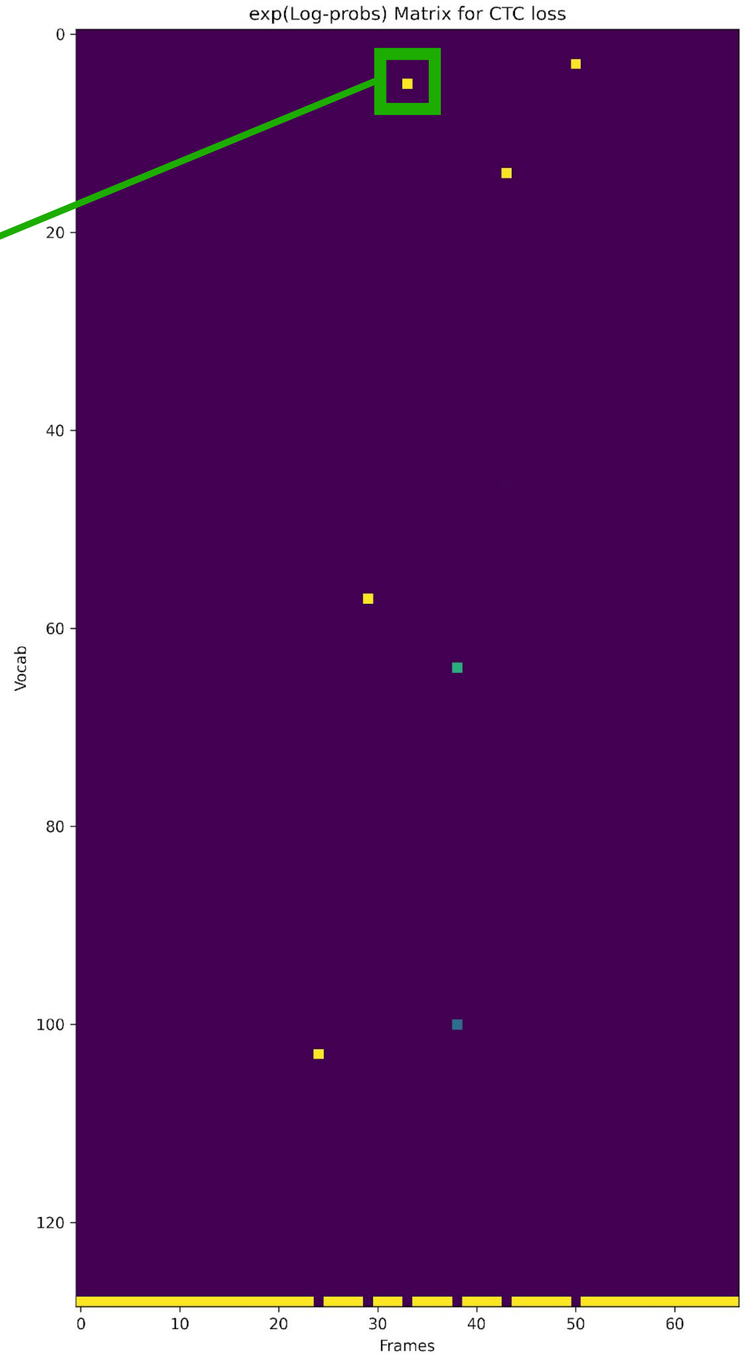
p(_при) p(ве)



Модель: stt_ru_conformer_ctc_large

CTC лосс

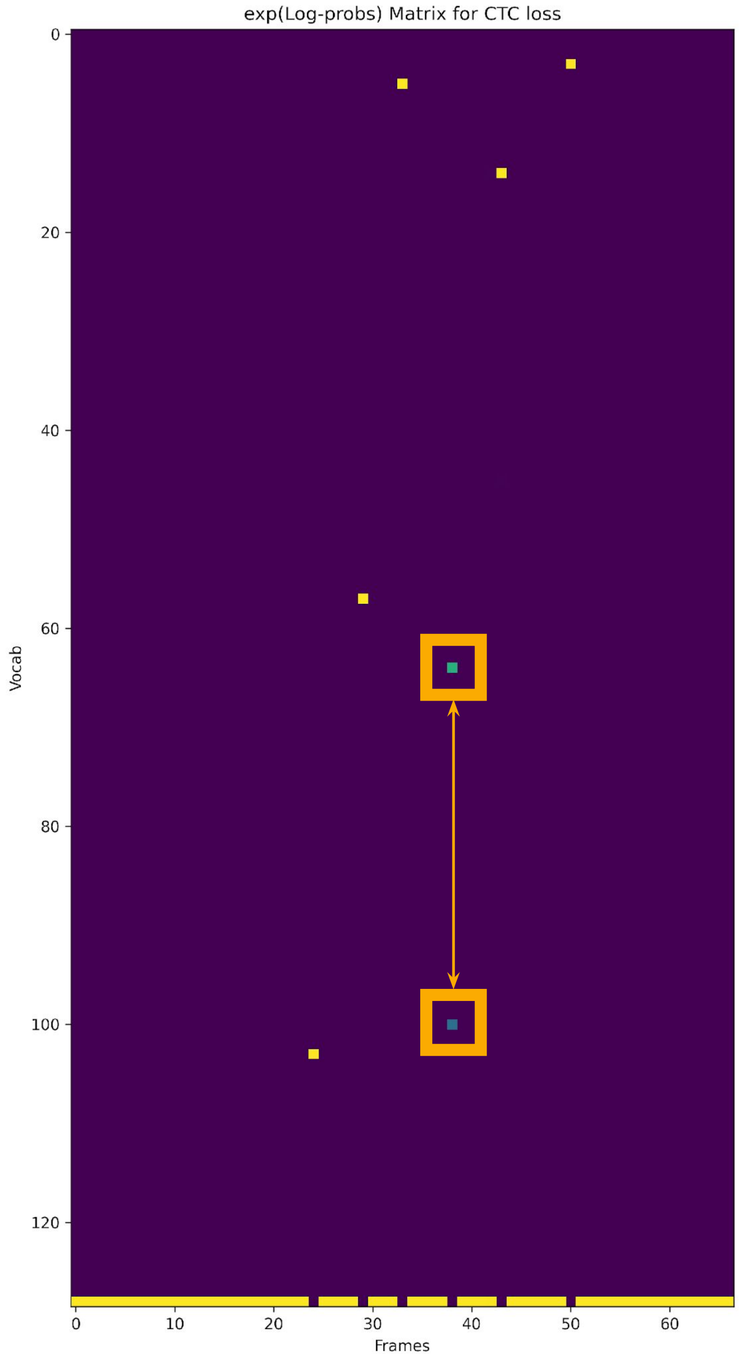
p(_при) p(ве) p(т)



Модель: stt_ru_conformer_ctc_large

CTC лосс

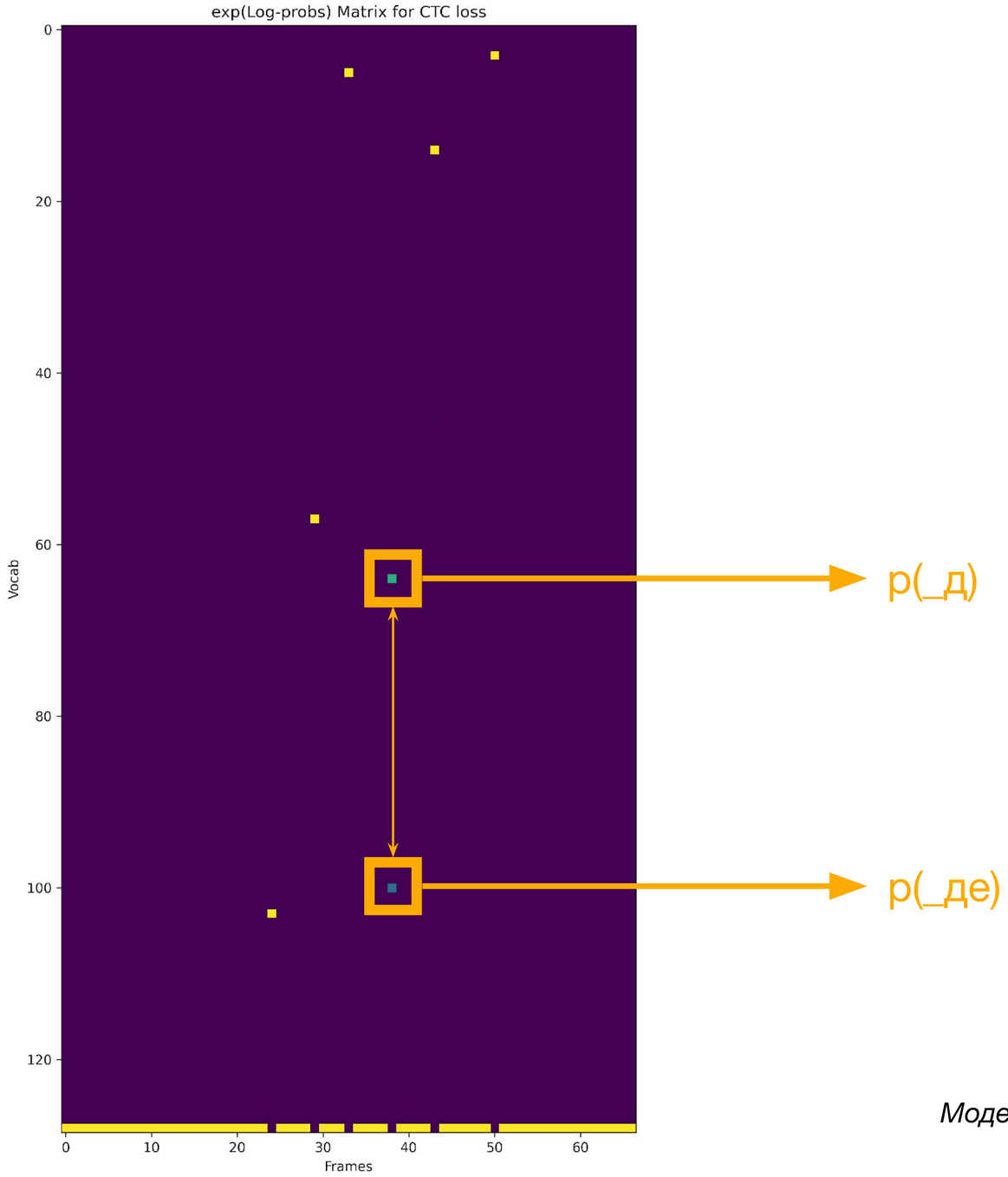
p(_при) p(ве) p(т)



Модель: stt_ru_conformer_ctc_large

CTC лосс

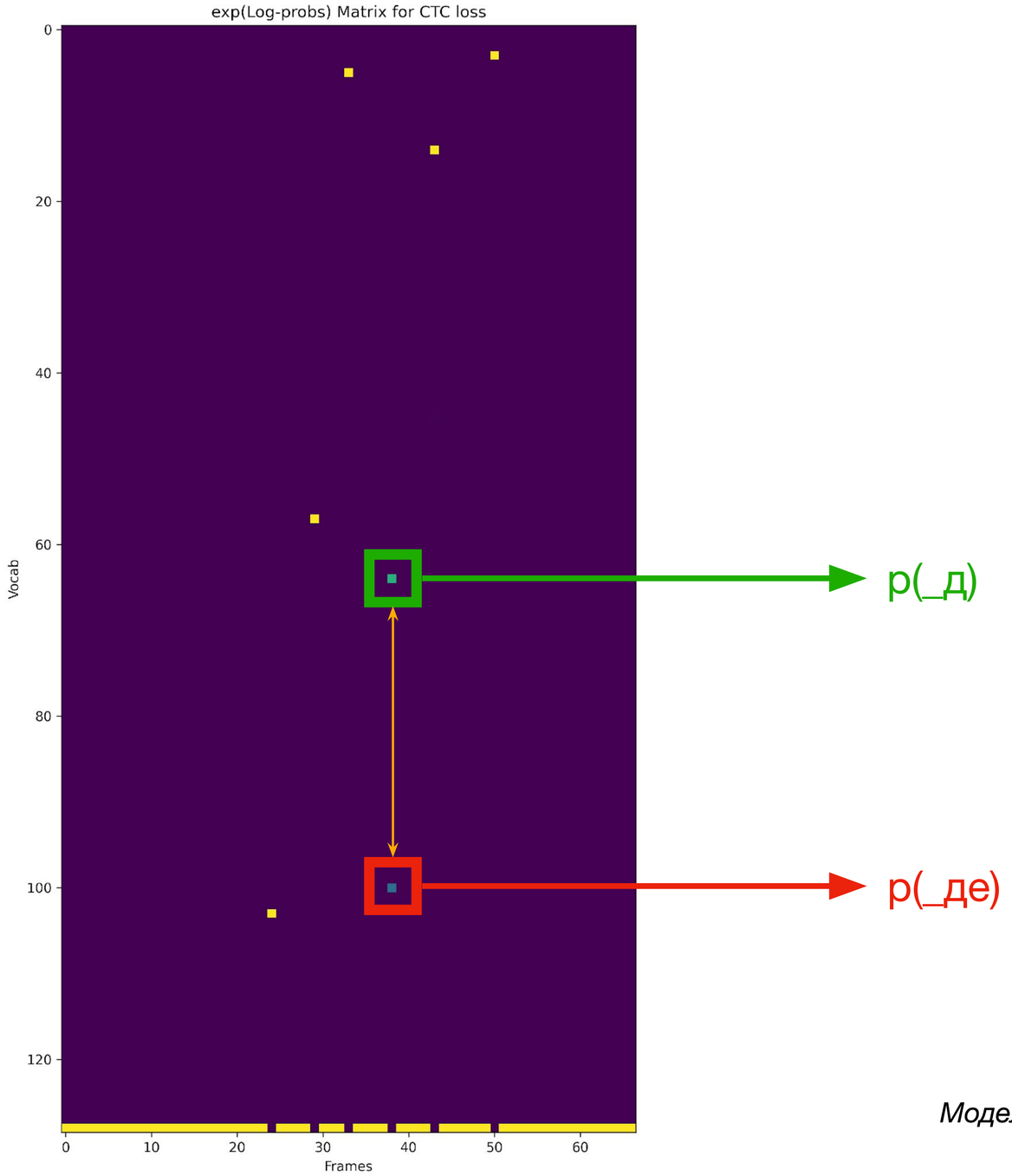
p(_при) p(ве) p(т)



Модель: stt_ru_conformer_ctc_large

CTC лосс

p(_при) p(ве) p(т)

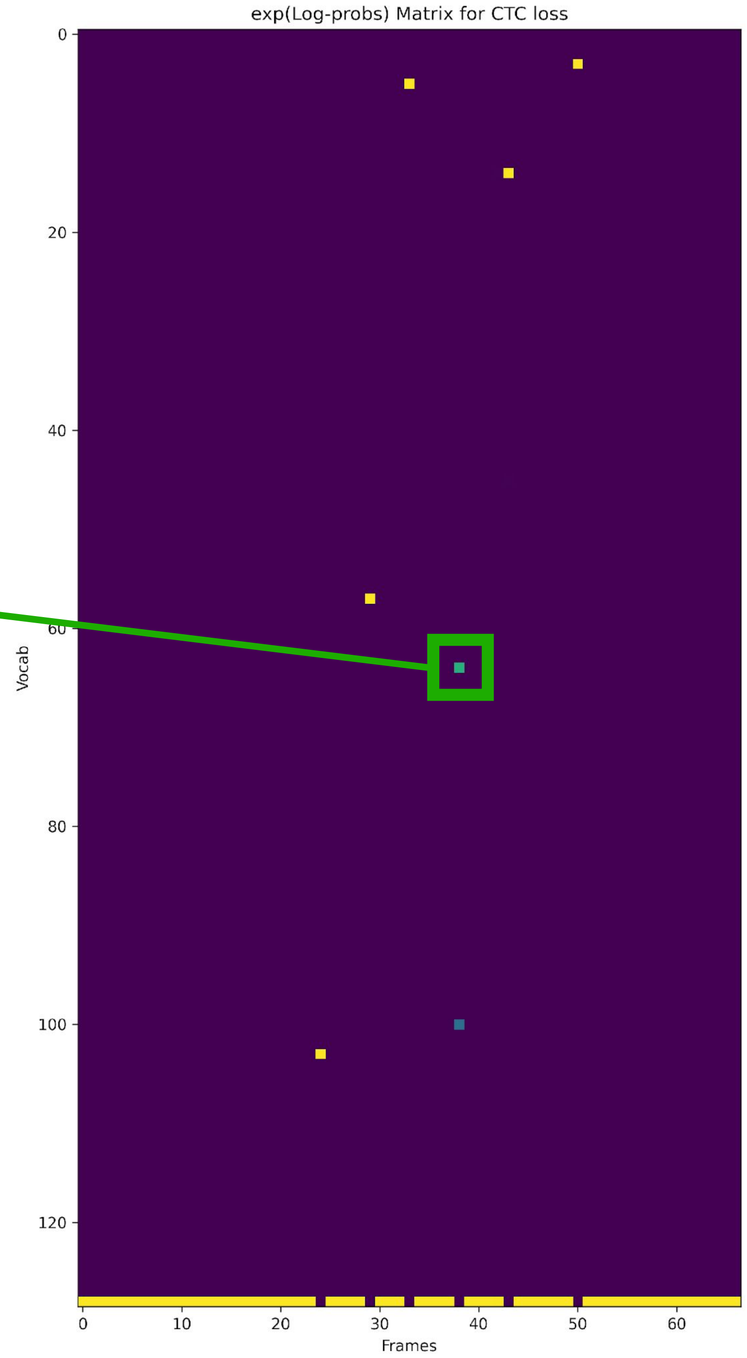


Модель: stt_ru_conformer_ctc_large

CTC лосс

p(_при) p(ве) p(т)

p(_д)

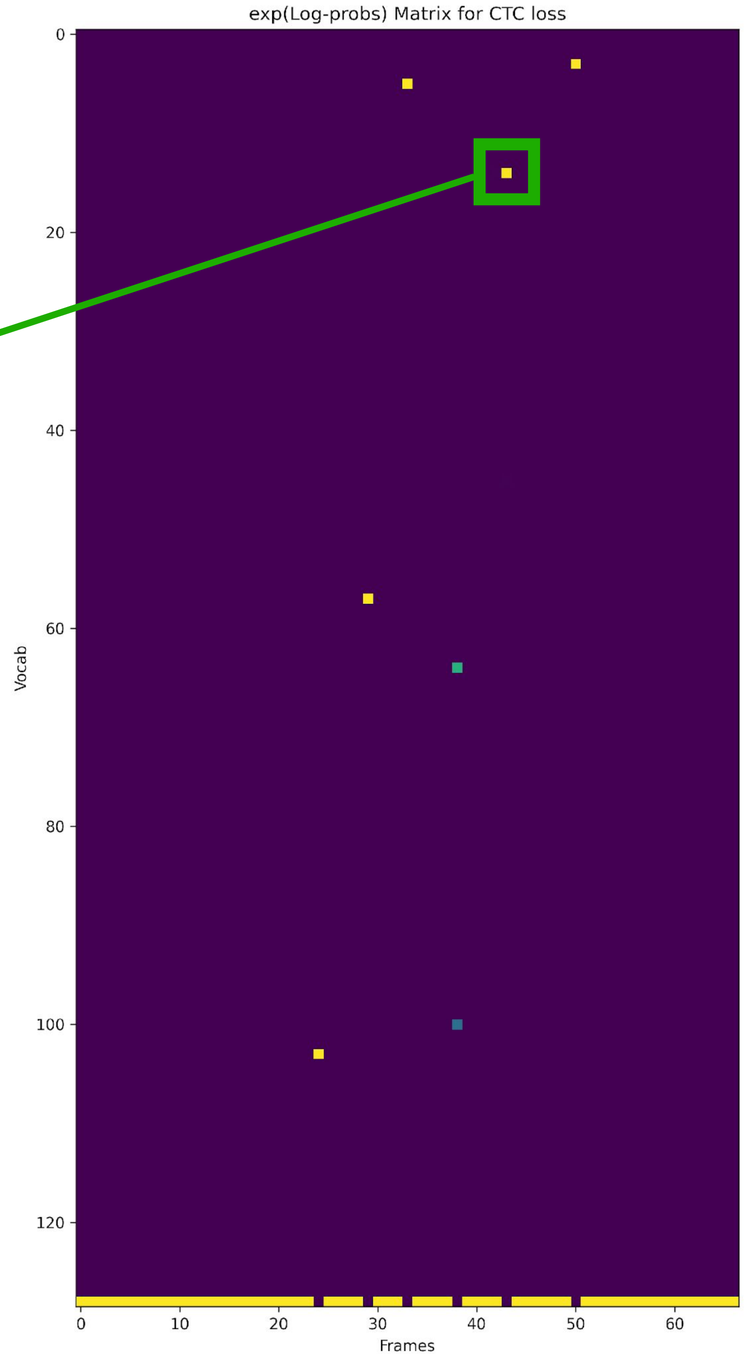


Модель: stt_ru_conformer_ctc_large

CTC лосс

p(_при) p(ве) p(т)

p(_д) p(л)

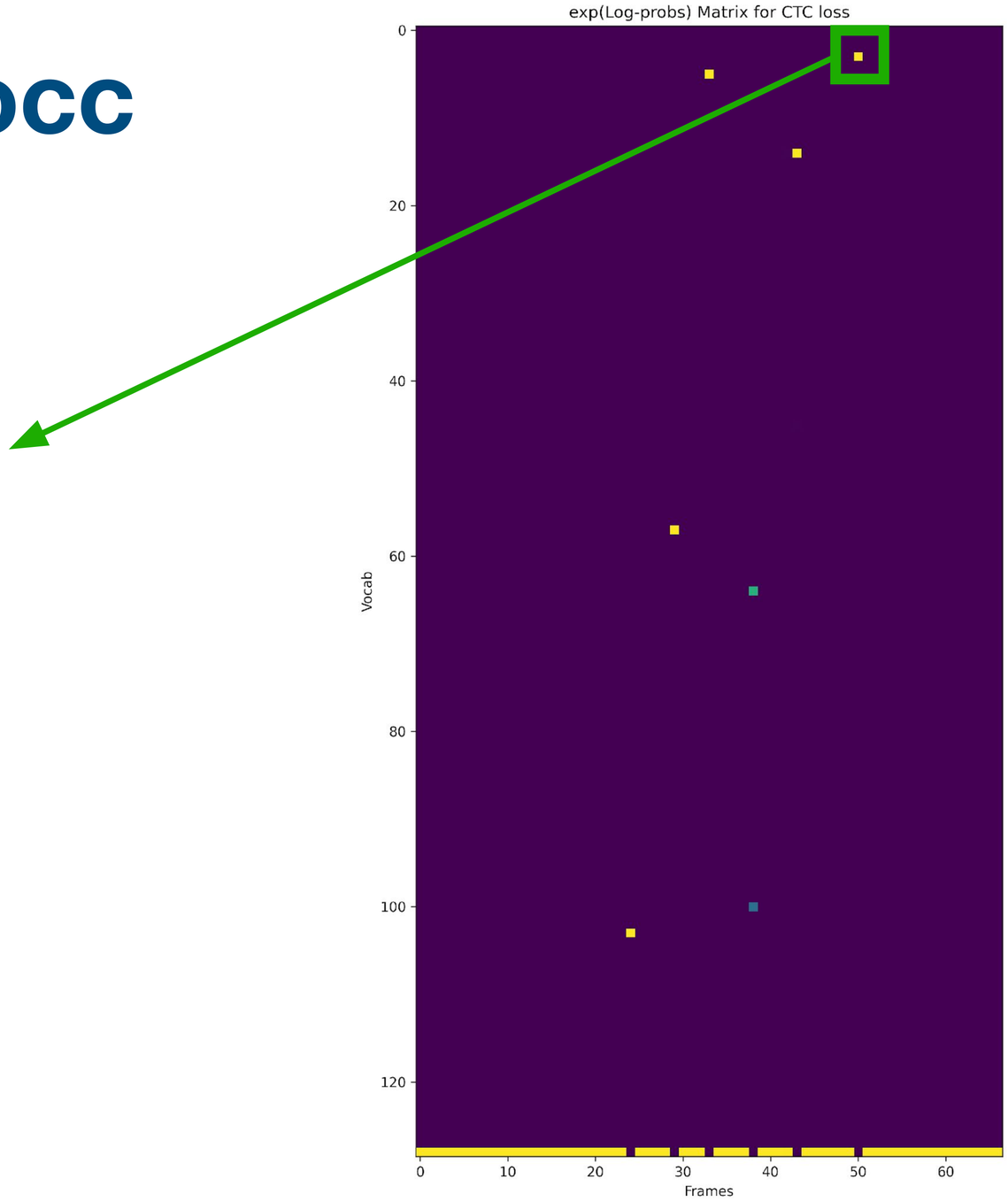


Модель: stt_ru_conformer_ctc_large

CTC лосс

p(_при) p(ве) p(т)

p(_д) p(л) p(с)



Модель: stt_ru_conformer_ctc_large

CTC лосс

$$\prod_{t=1}^{T} p(c_t \mid \hat{Y}) \quad = \quad \text{p(\_при)} \times \text{p(ве)} \times \text{p(т)} \times \text{p(\_д)} \times \text{p(л)} \times \text{p(с)}$$

Модель: stt_ru_conformer_ctc_large

CTC лосс

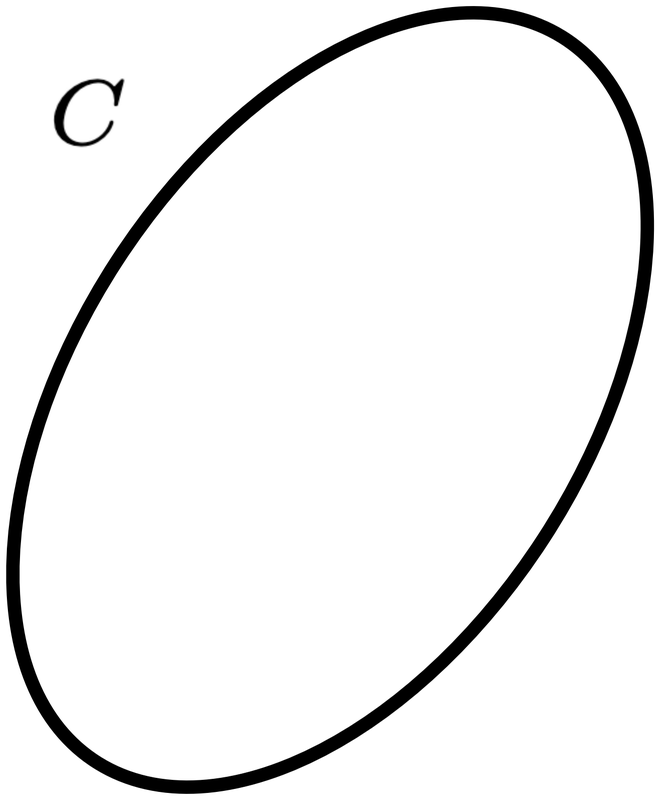
$$\prod_{t=1}^{T} p(c_t \mid \hat{Y}) \quad = \quad \text{p(<blank>)} \times \text{p(<blank>)} \times \ldots \times \text{p(\_при)} \times \ldots$$

Модель: stt_ru_conformer_ctc_large

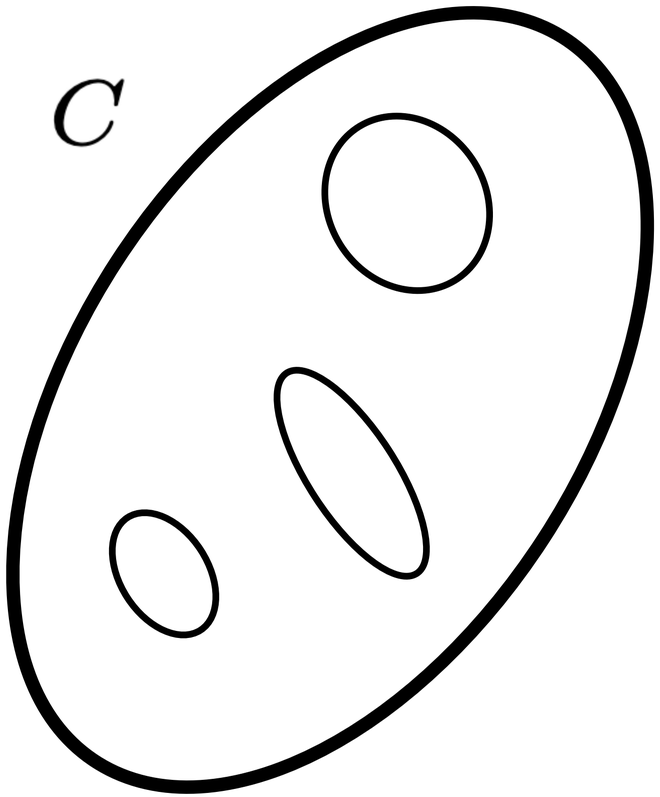
CTC лосс

$$L(Y, \hat{Y}) = -\log \left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y}) \right)$$

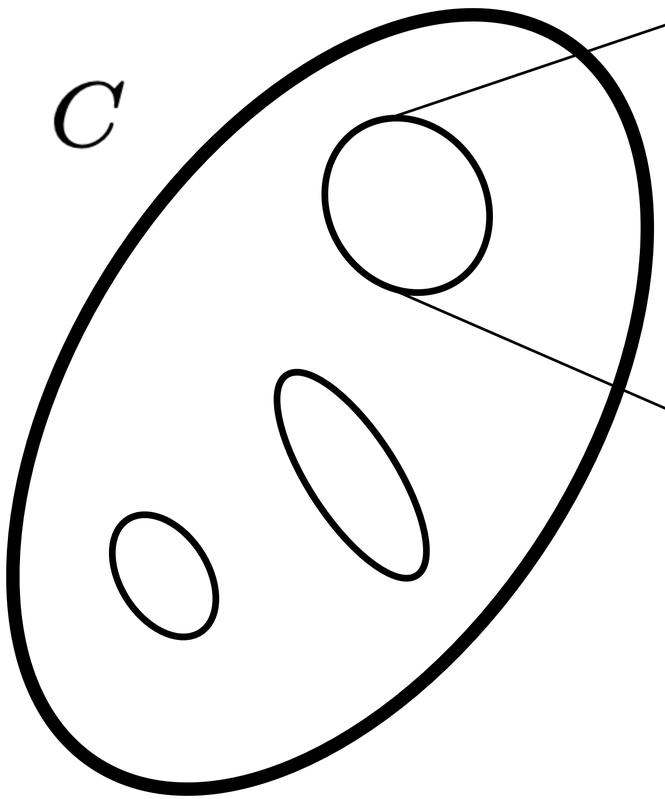
CTC лосс



CTC лосс



CTC лосс



|||ппп|ррр|иии|ввв|еее|ттт||_|дэээ|лэ|ээссс||

|||ппп**п**|рр|иии|ввв|еее|ттт|_||дэээ|лээ|эссс||

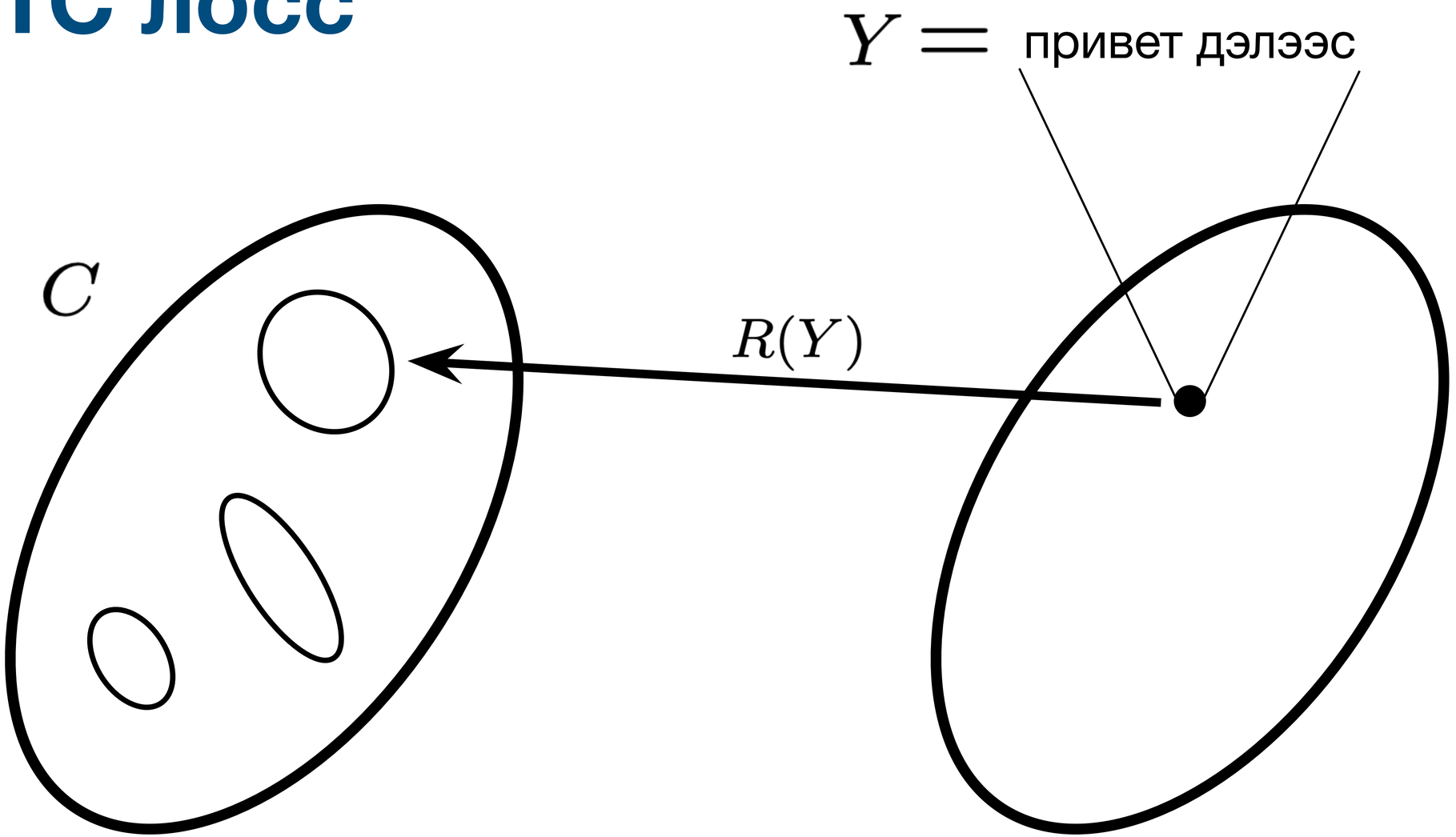
|||ппп**пп**|р|иии|ввв|еее|ттт||_|дэээ|лэ|ээссс||

|||ппппп|р**р**|ии|ввв|еее|ттт|_||дэээ|лээ|эссс||

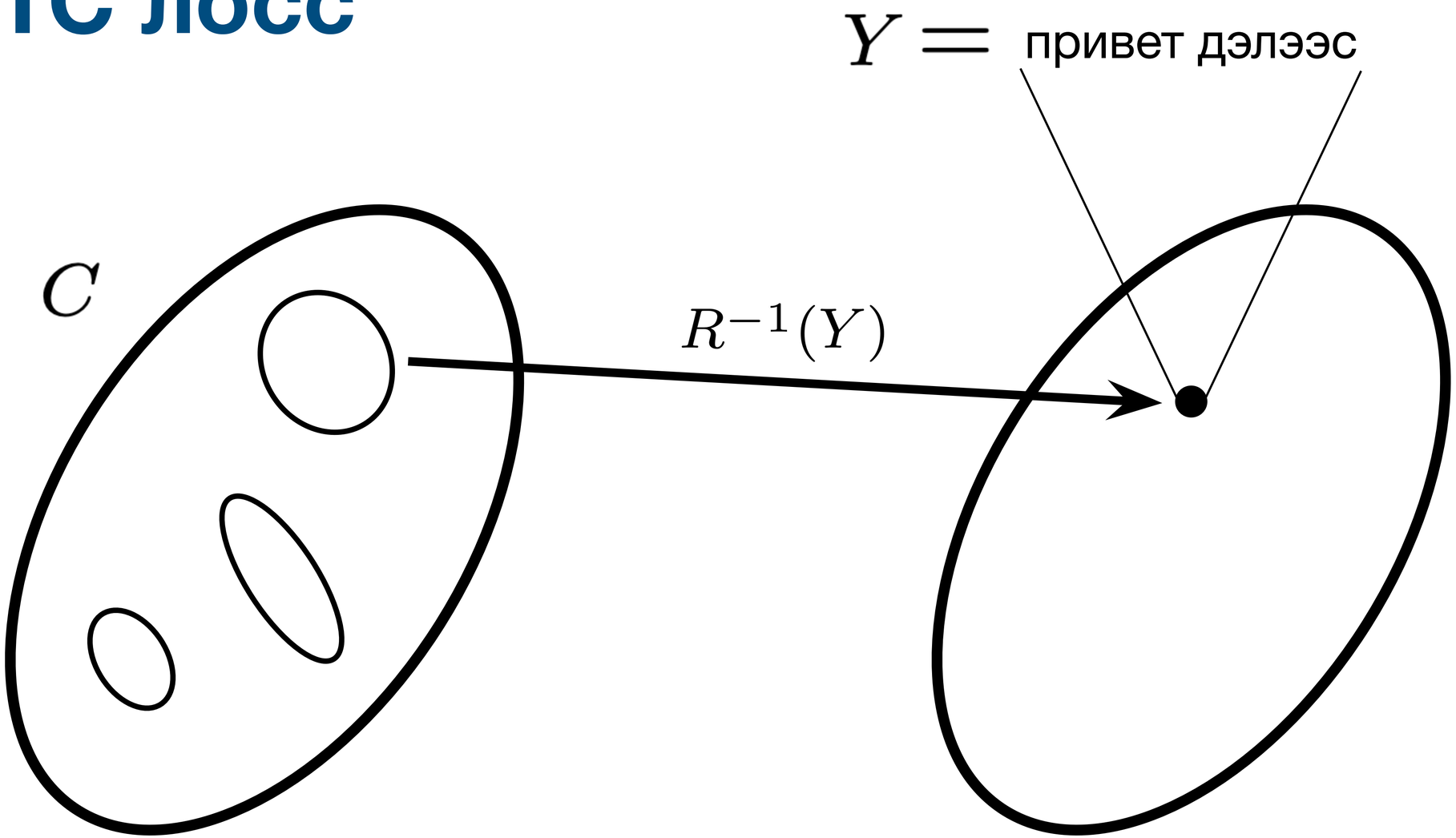
|||ппппп|р**рр**|и|ввв|еее|ттт||_|дэээ|лэ|ээссс||

|||пппппппппппппппппп|р|и|в|е|т|_|дэ|лэ|эс||

CTC лосс



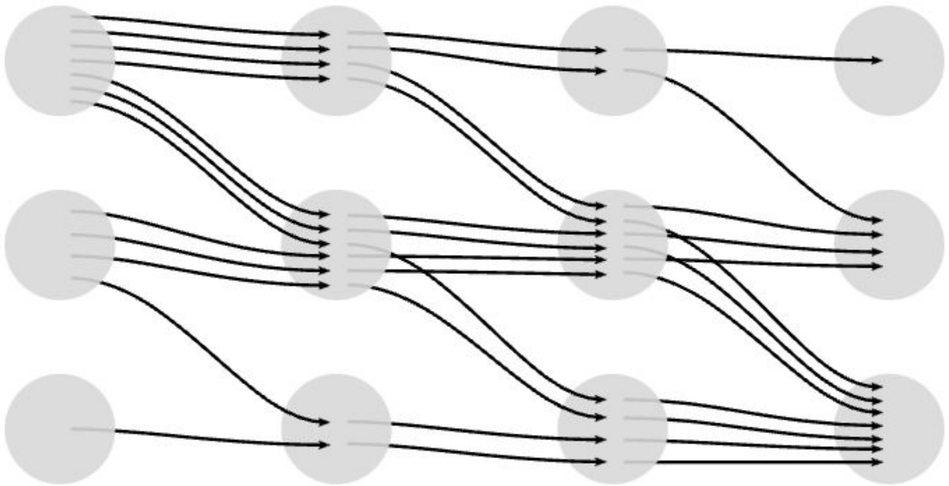
CTC лосс



CTC лосс

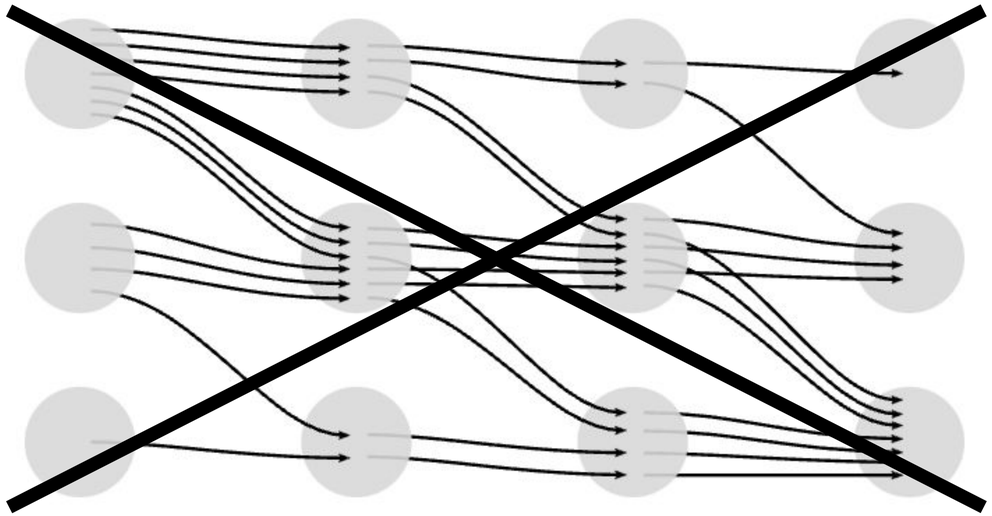
$$L(Y, \hat{Y}) = -\log \left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y}) \right)$$

CTC лосс

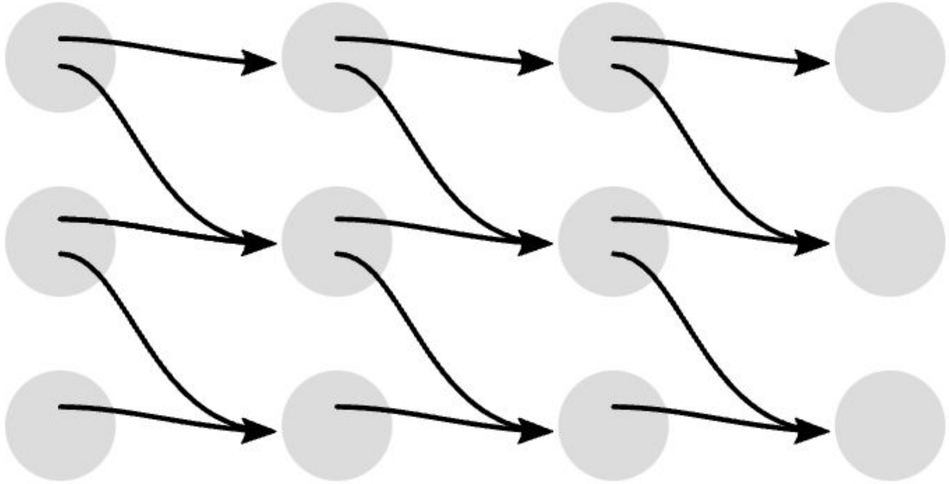


Подсчёт всех возможных (vocab_size^num_frames) путей не оптимален

CTC лосс

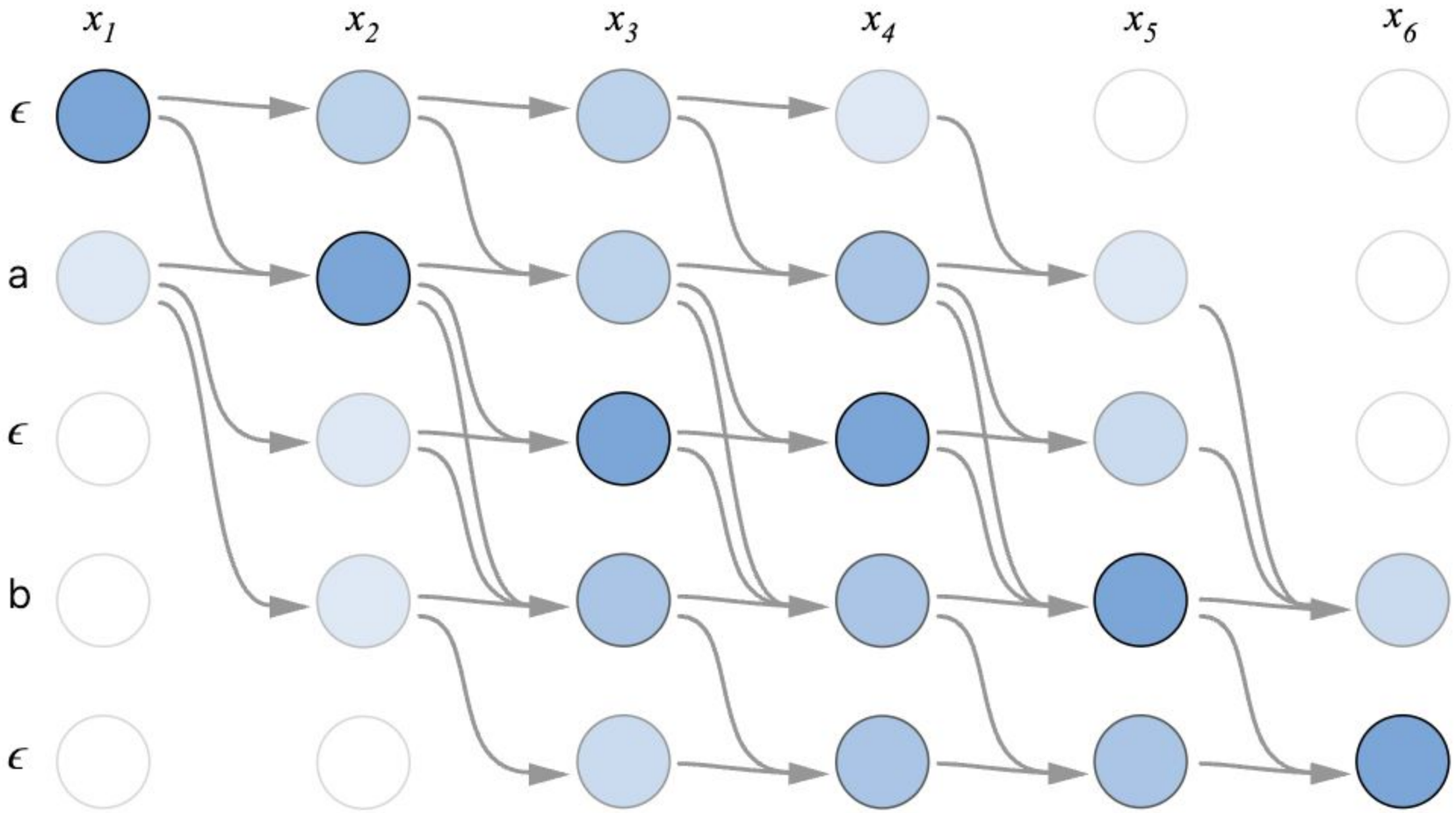


Подсчёт всех возможных (vocab_size^num_frames) путей не оптимален



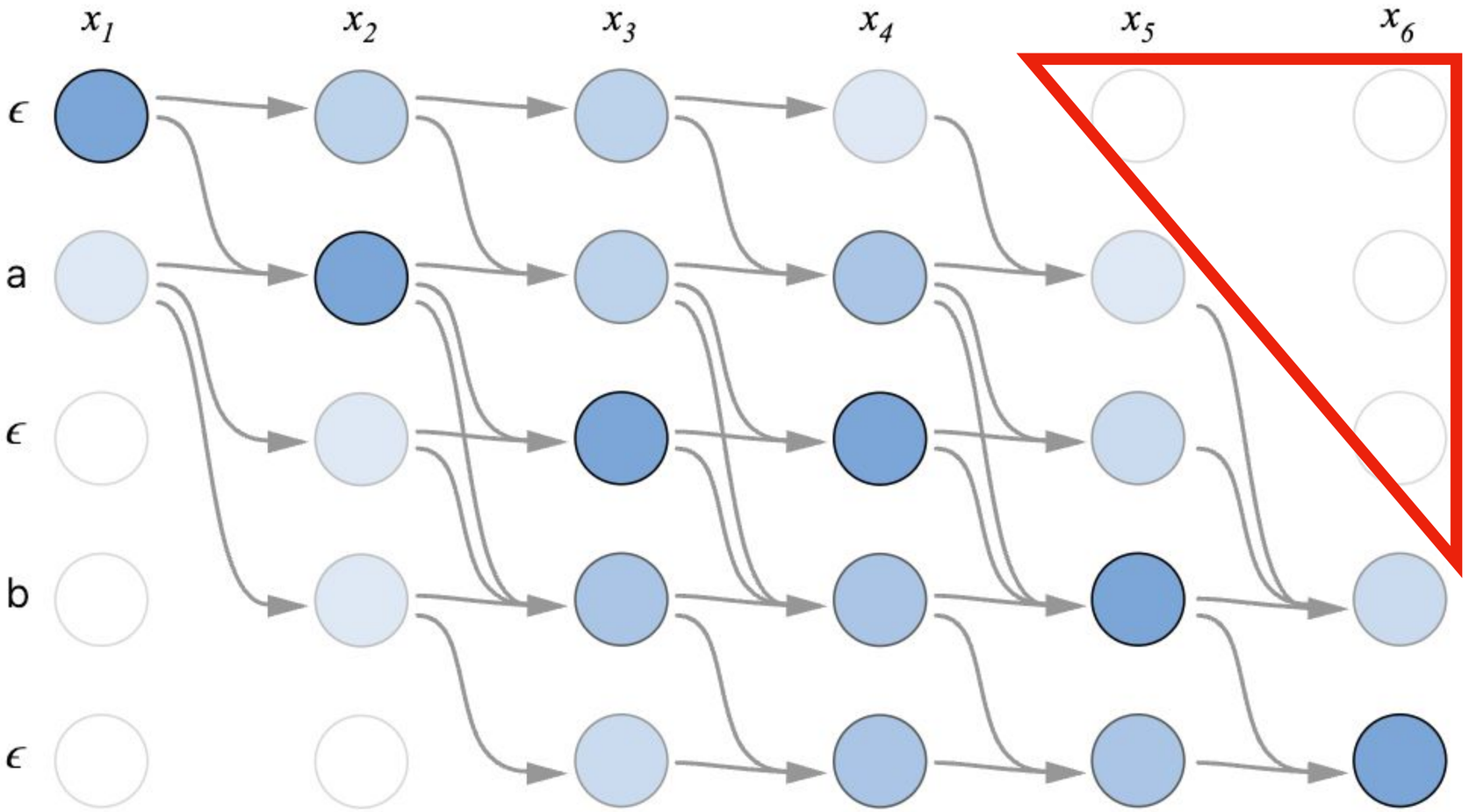
С помощью динамического программирования считаются только возможные пути

CTC лосс

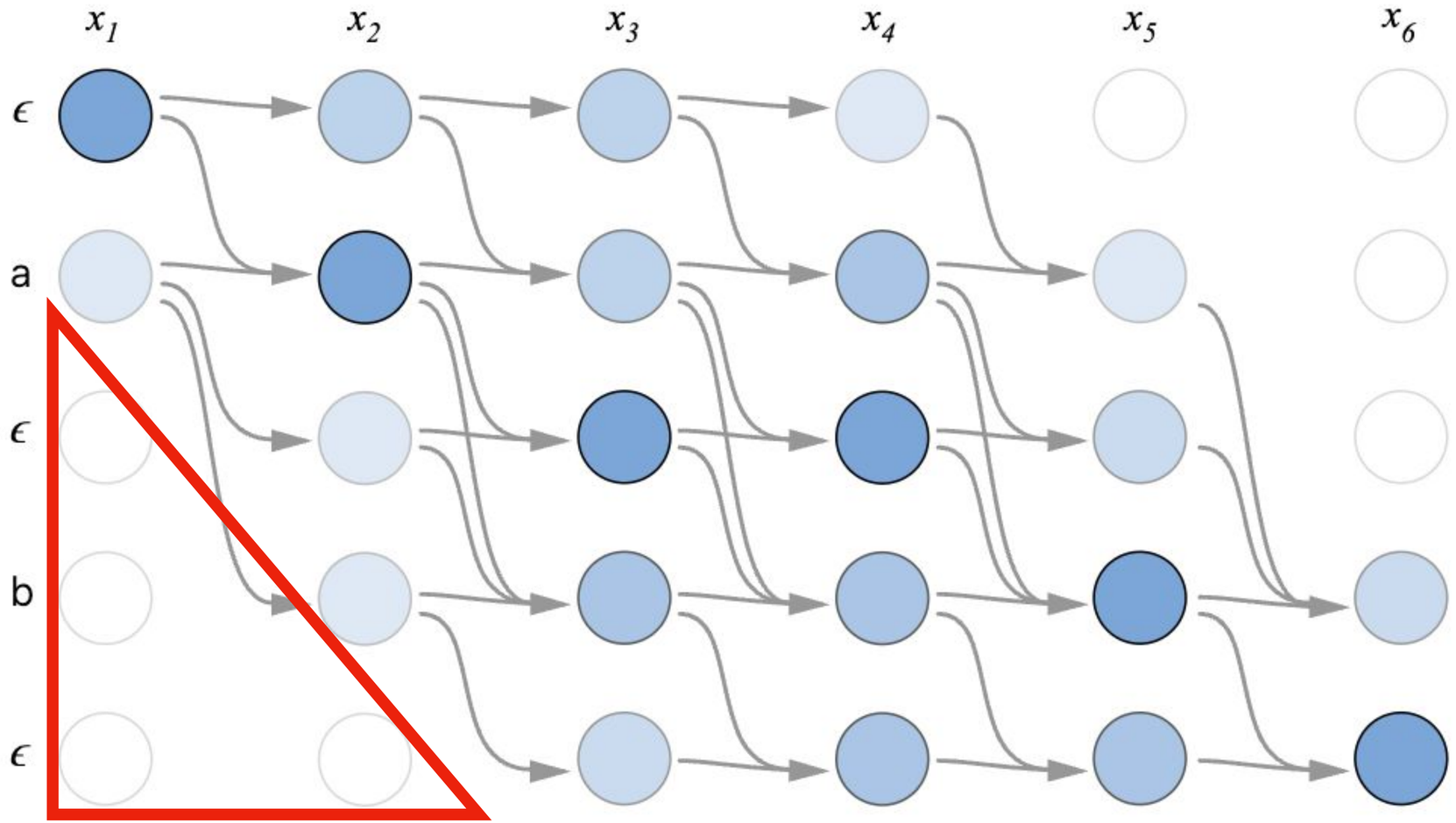


CTC лосс

заведомо не хватит фреймов для предсказания таргета



CTC лосс

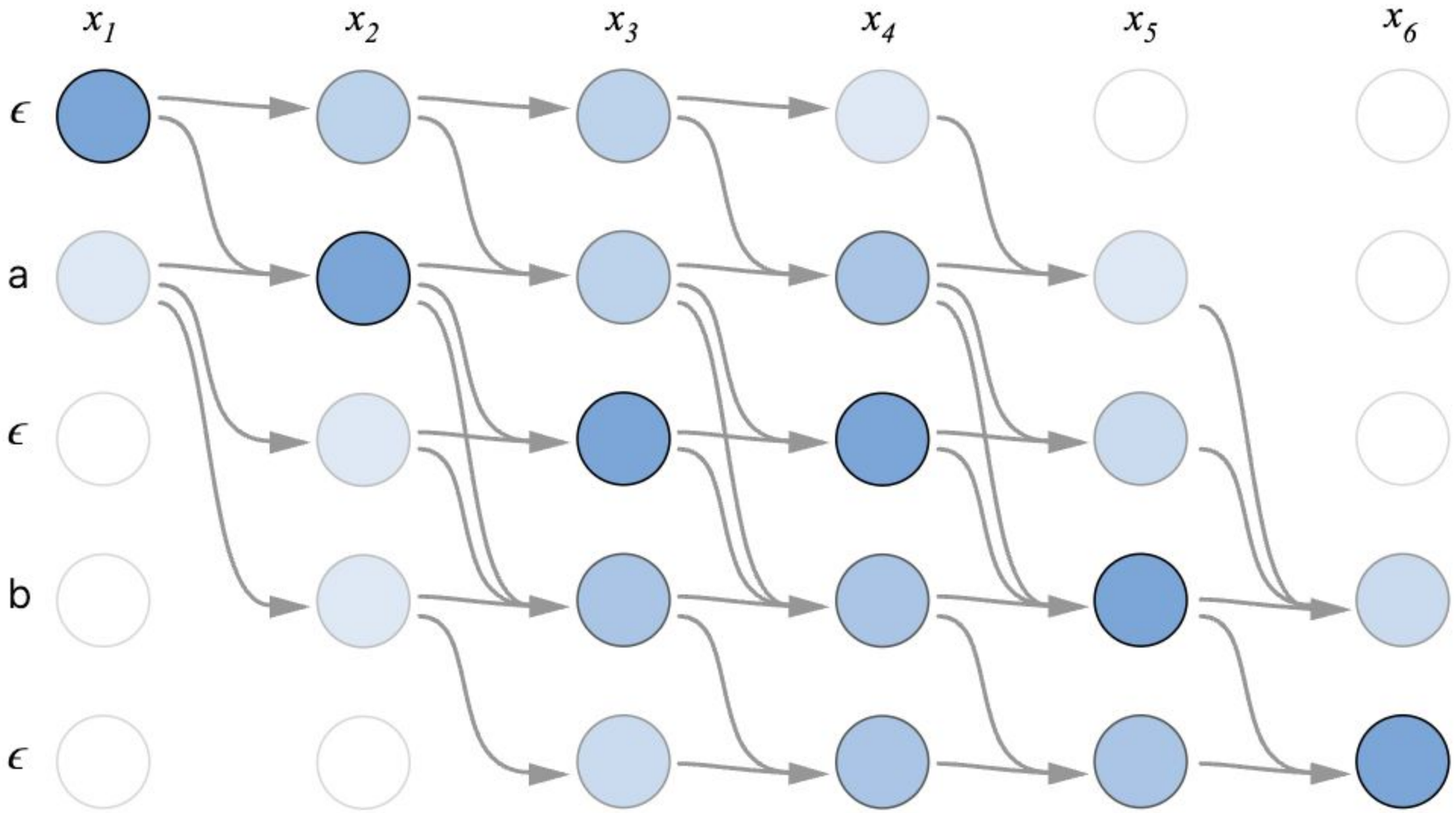


заведомо не хватит токенов для предсказания таргета

CTC лосс

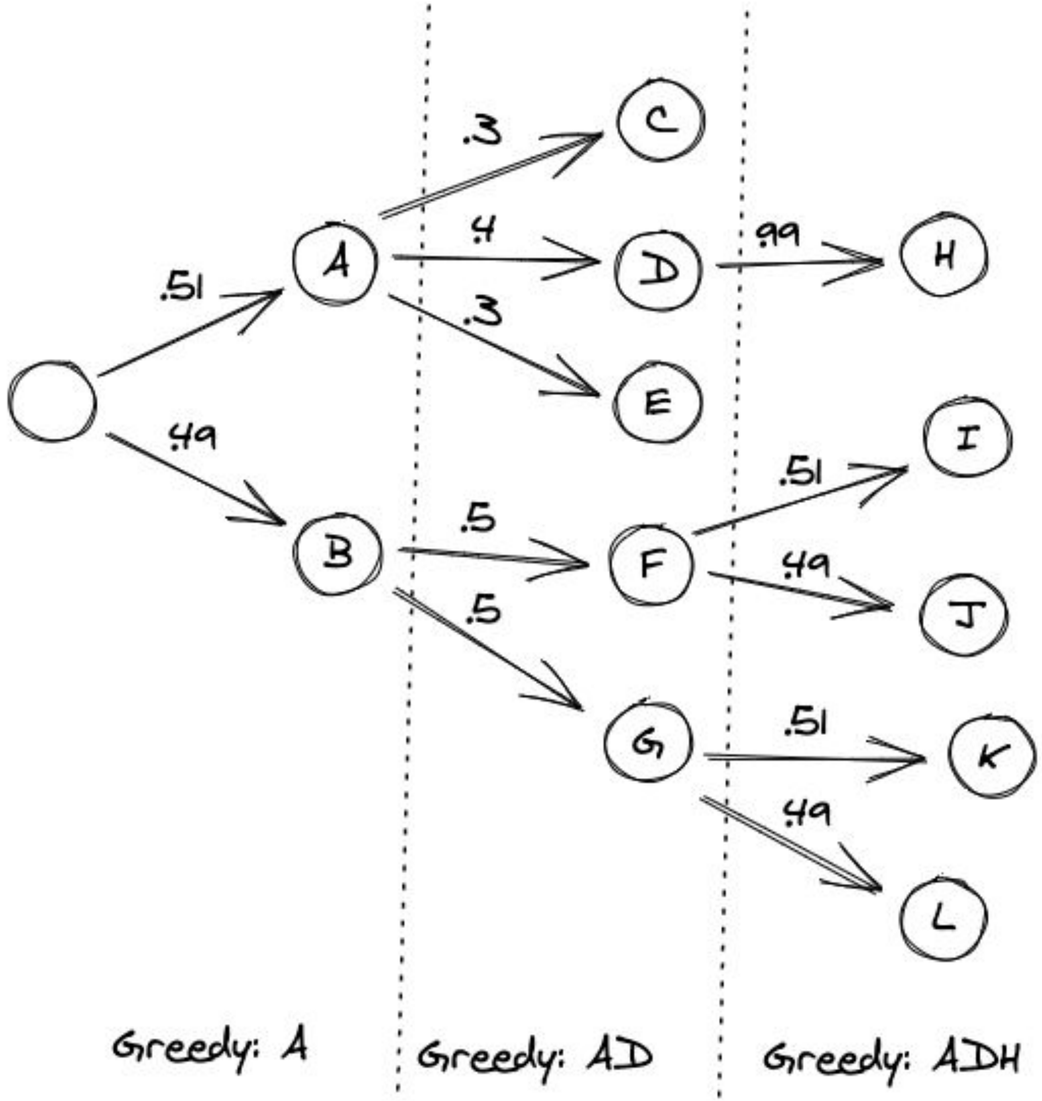
$$L(Y, \hat{Y}) = -\log\left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y})\right)$$

CTC лосс

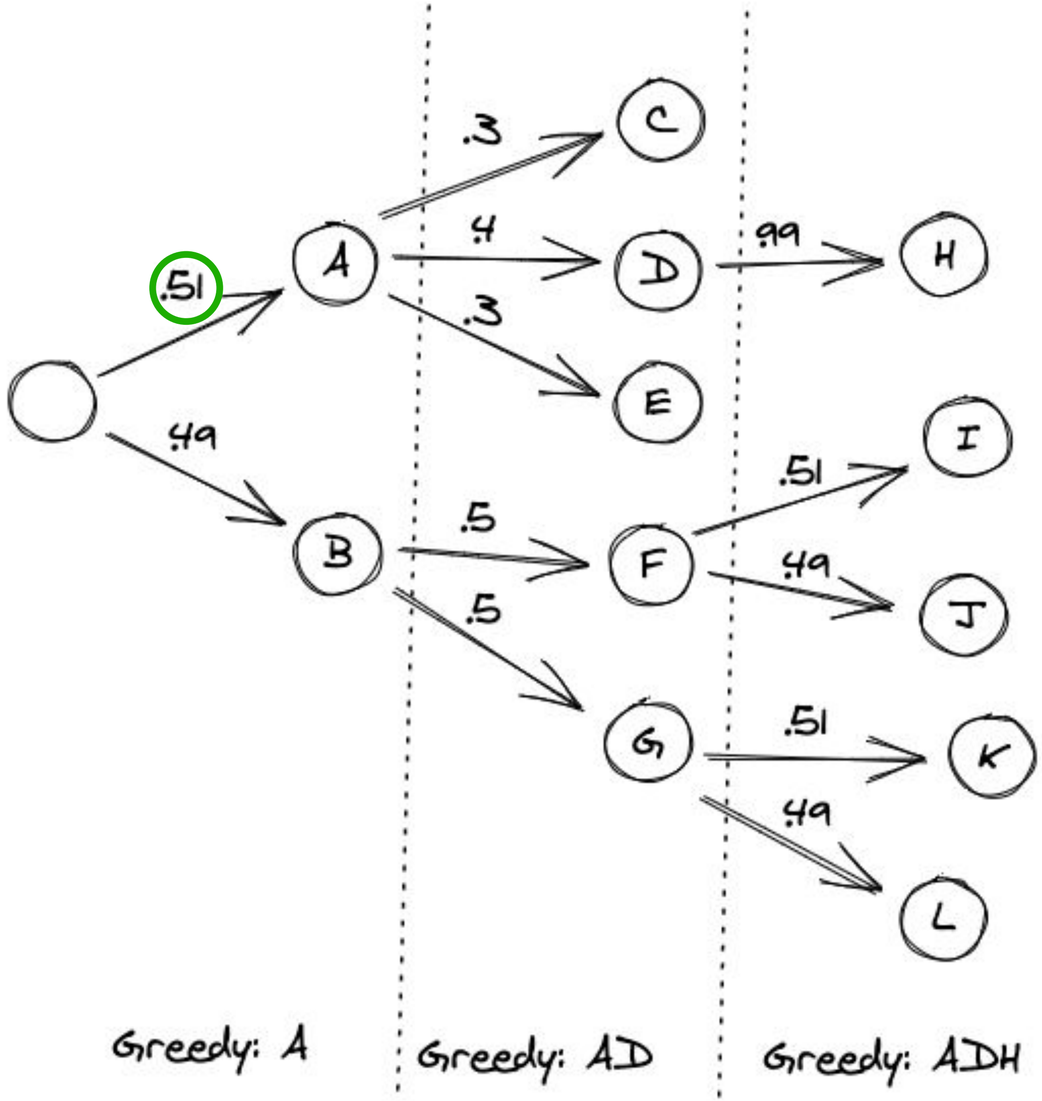


Выбор лучшей гипотезы

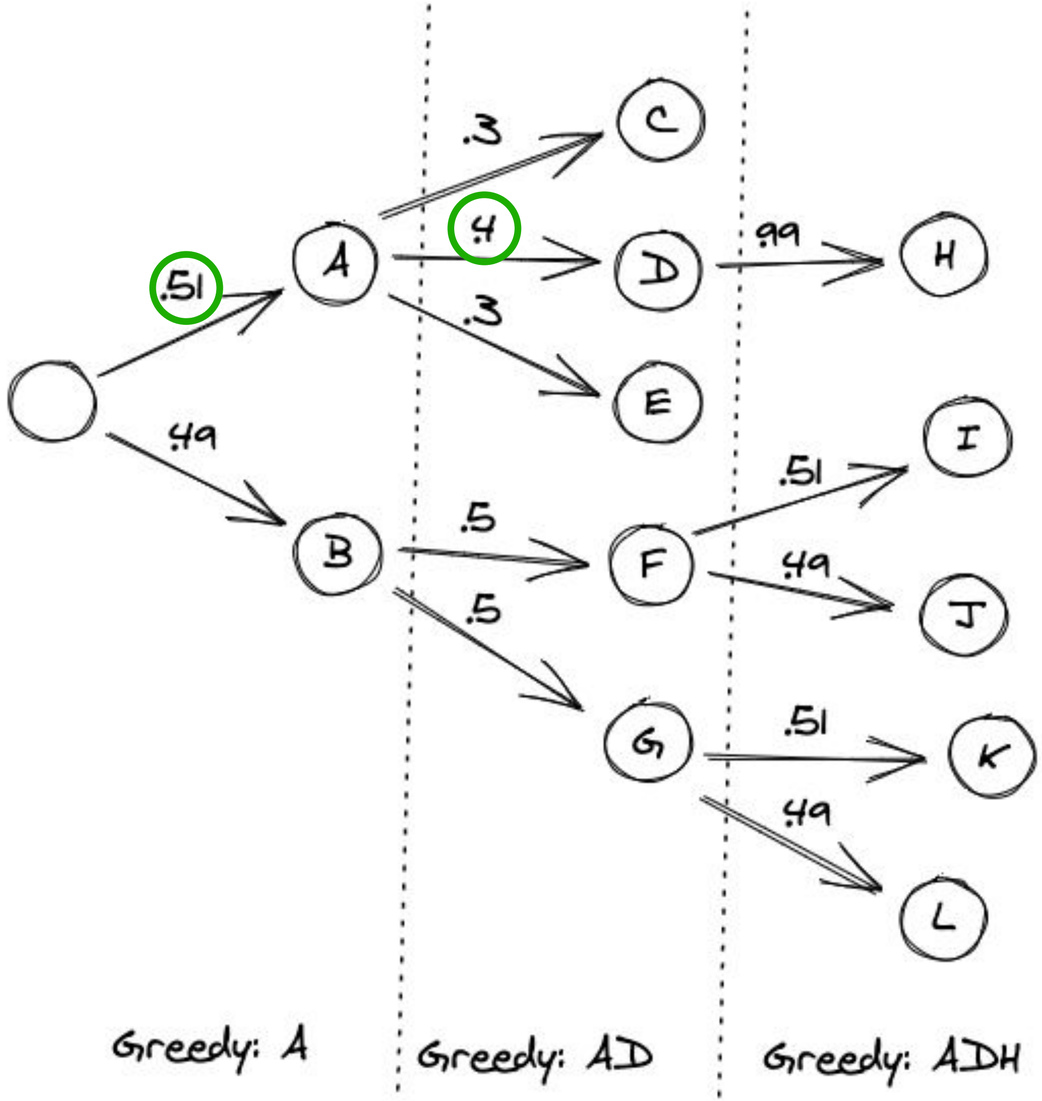
Жадный поиск (Greedy search)



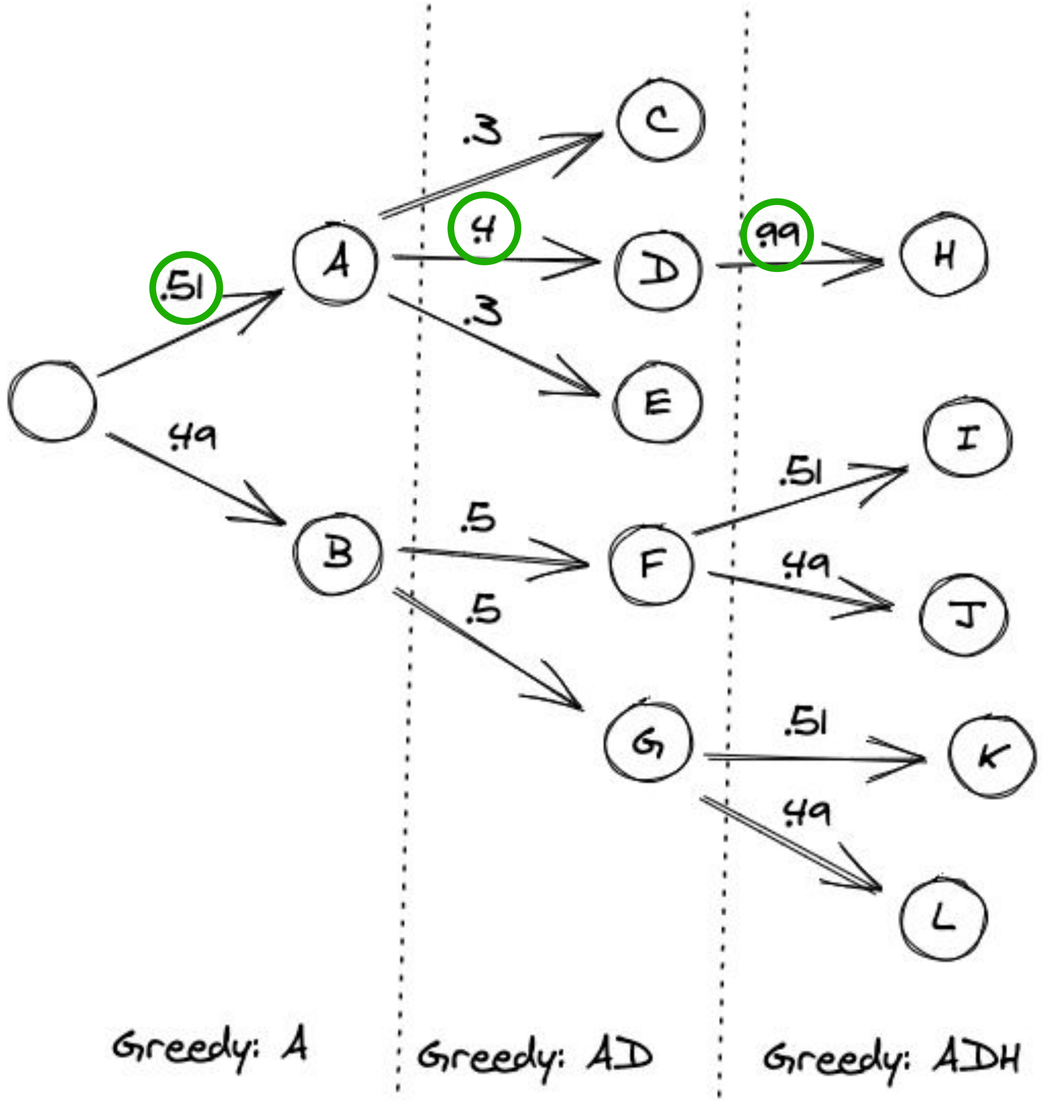
Жадный поиск (Greedy search)



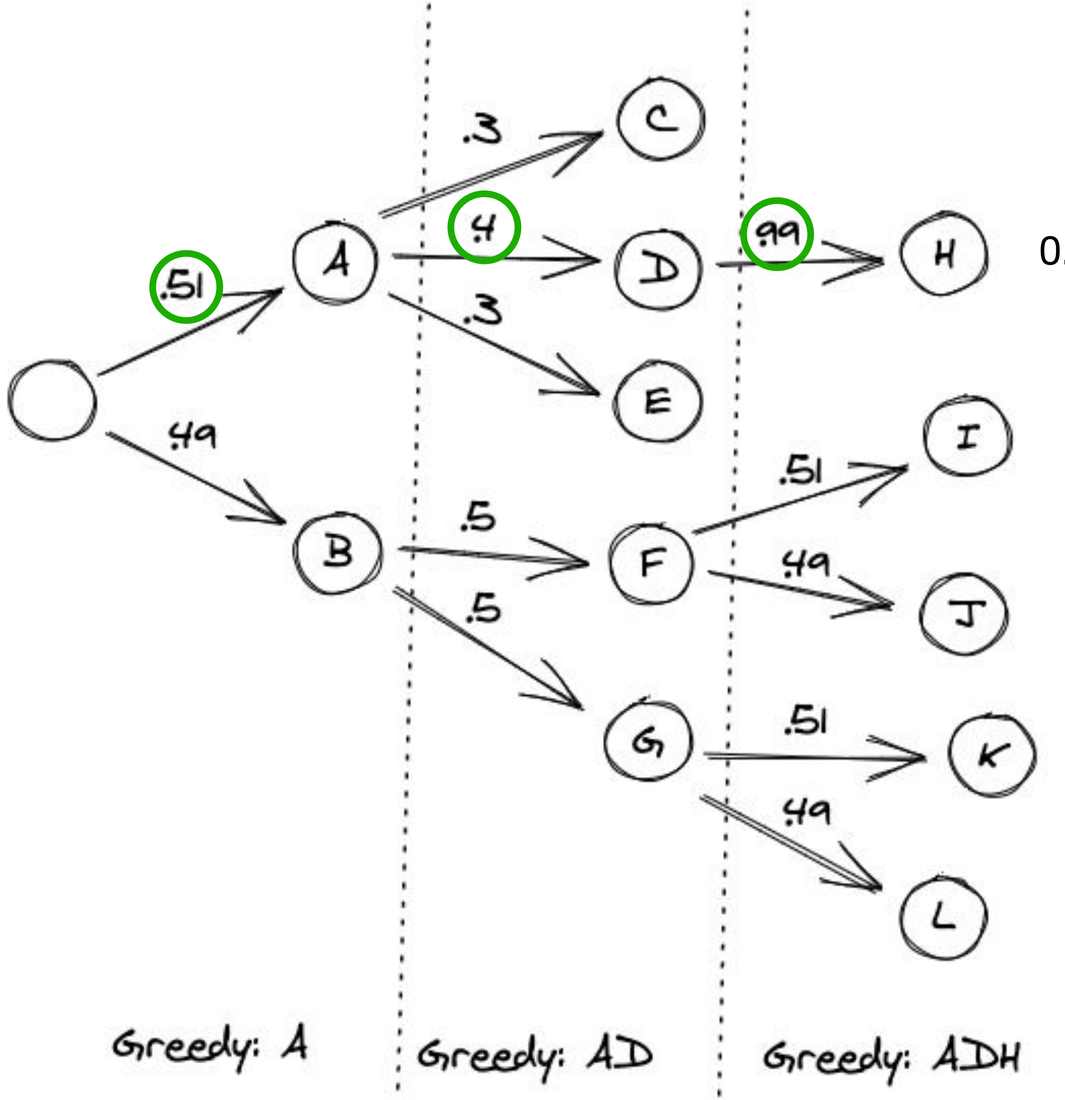
Жадный поиск (Greedy search)



Жадный поиск (Greedy search)

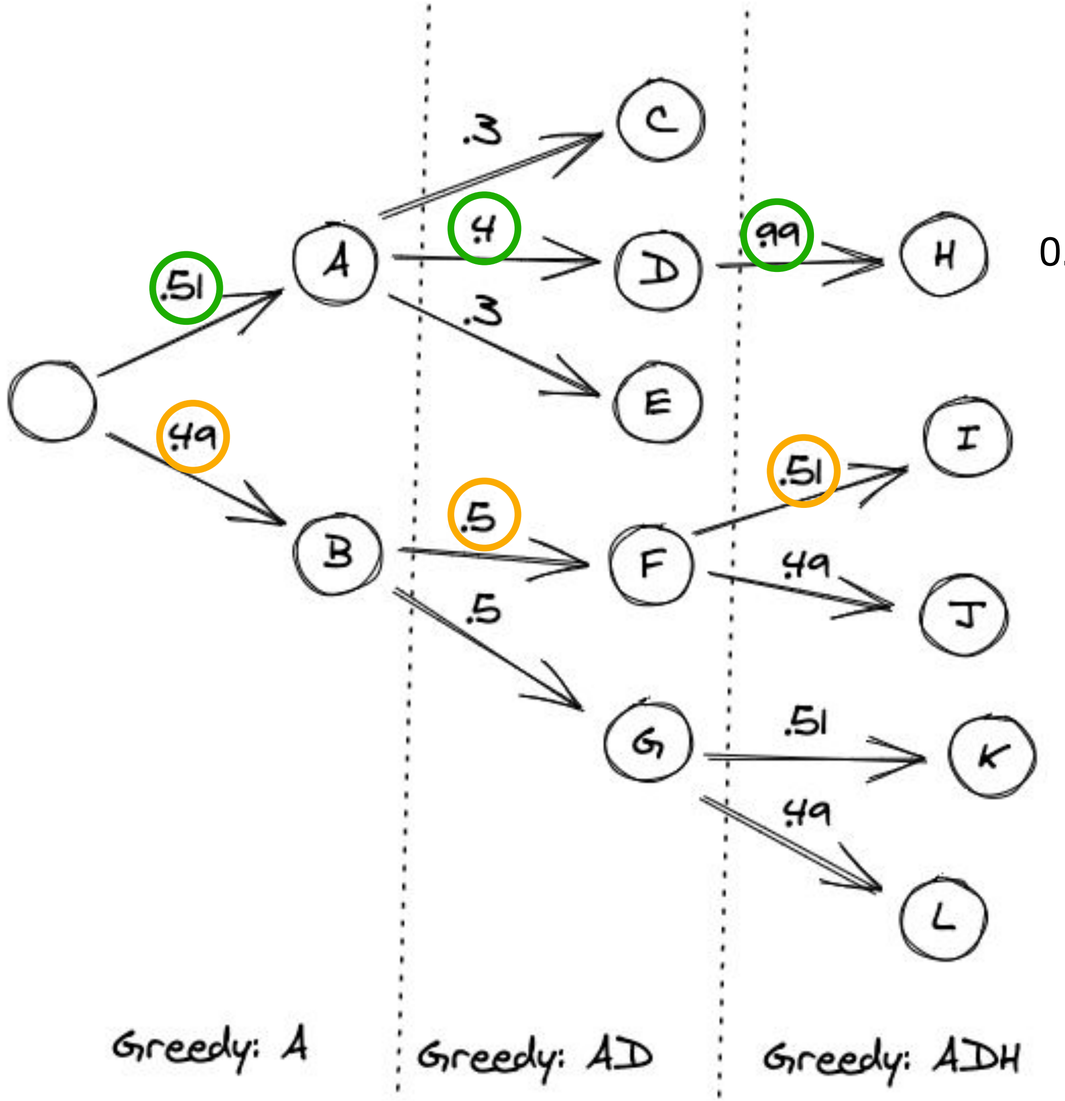


Жадный поиск (Greedy search)



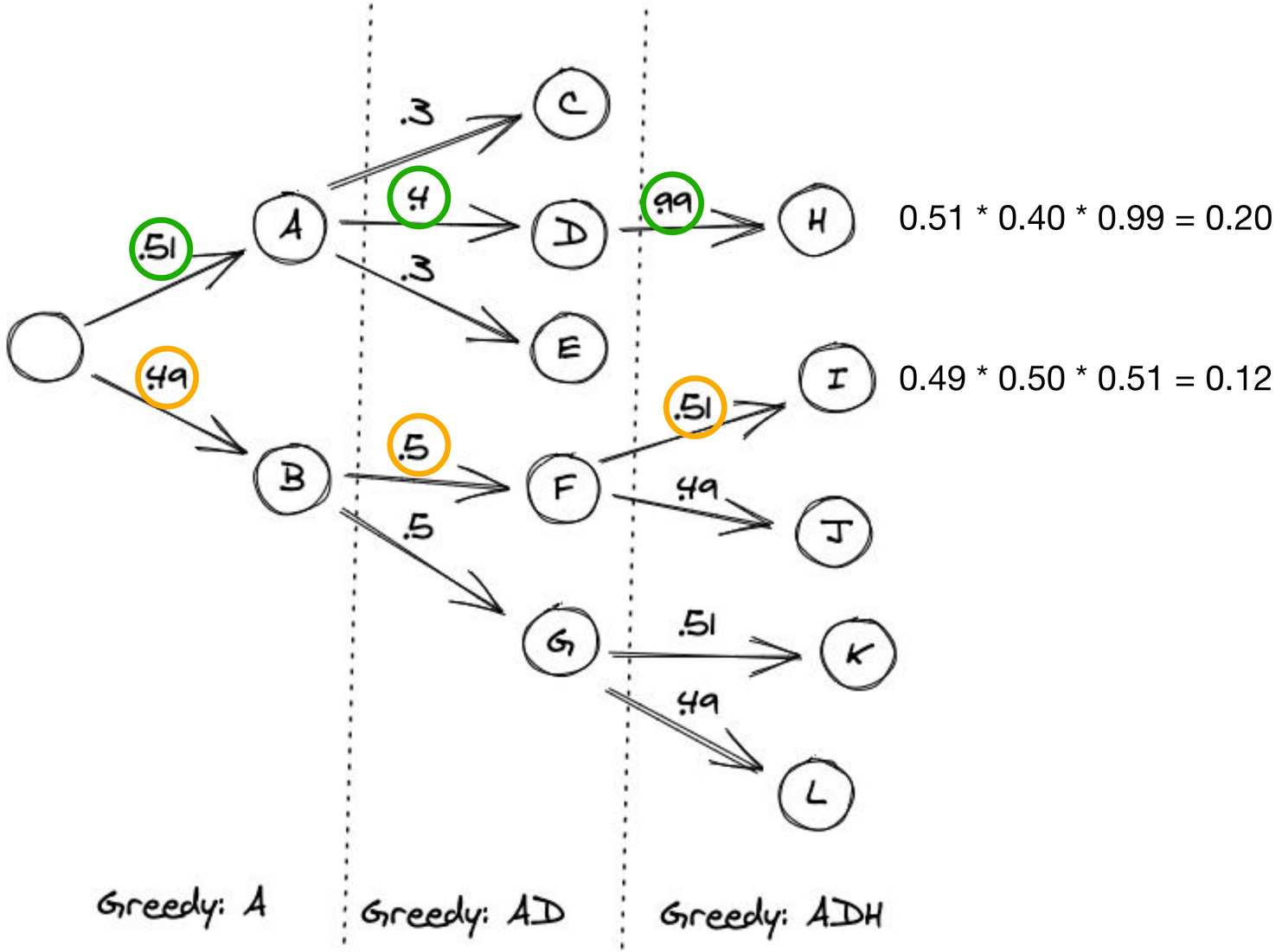
0.51 * 0.40 * 0.99 = 0.20

Жадный поиск (Greedy search)



0.51 * 0.40 * 0.99 = 0.20

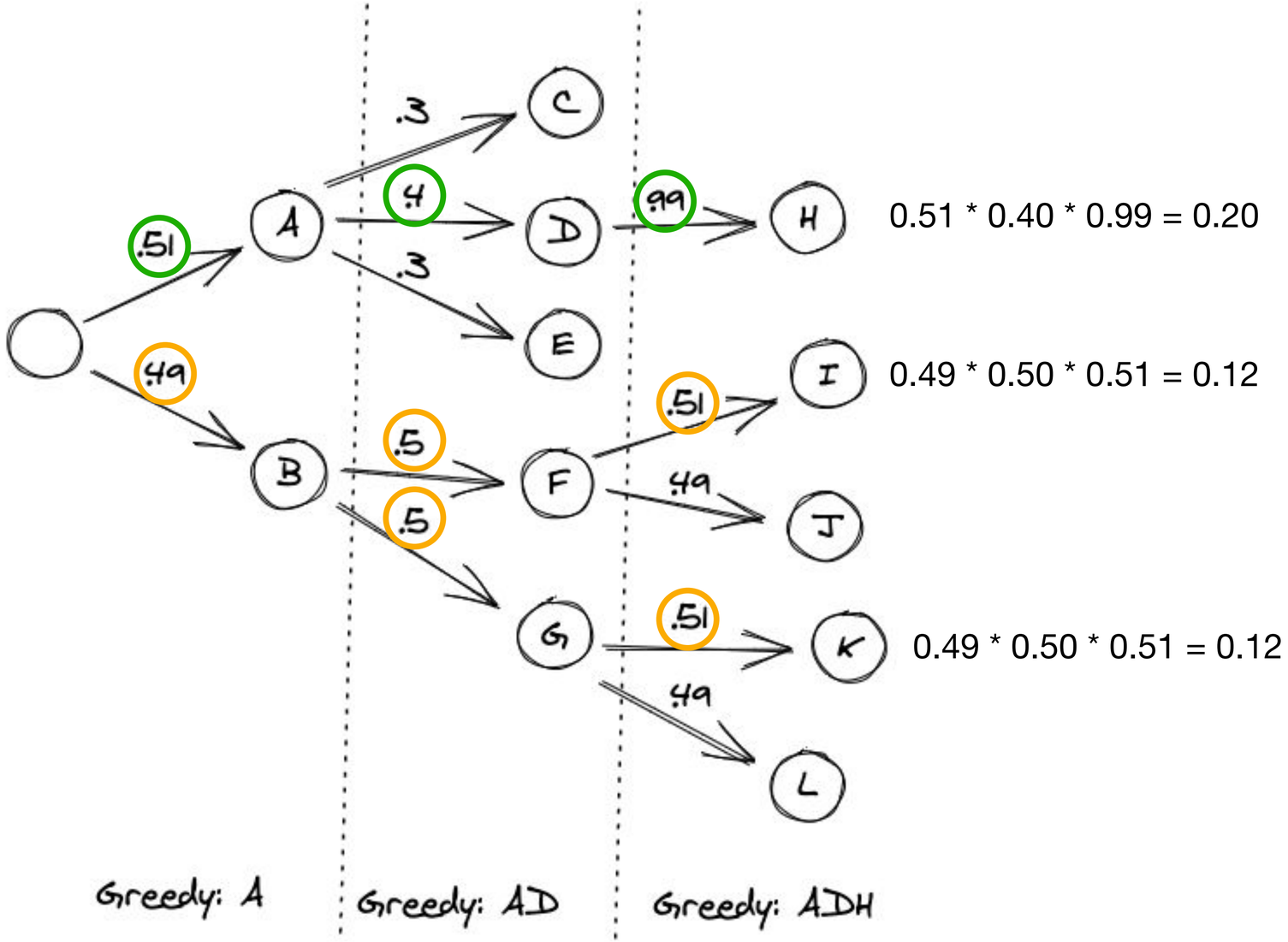
Жадный поиск (Greedy search)



0.51 * 0.40 * 0.99 = 0.20

0.49 * 0.50 * 0.51 = 0.12

Жадный поиск (Greedy search)



0.51 * 0.40 * 0.99 = 0.20

0.49 * 0.50 * 0.51 = 0.12

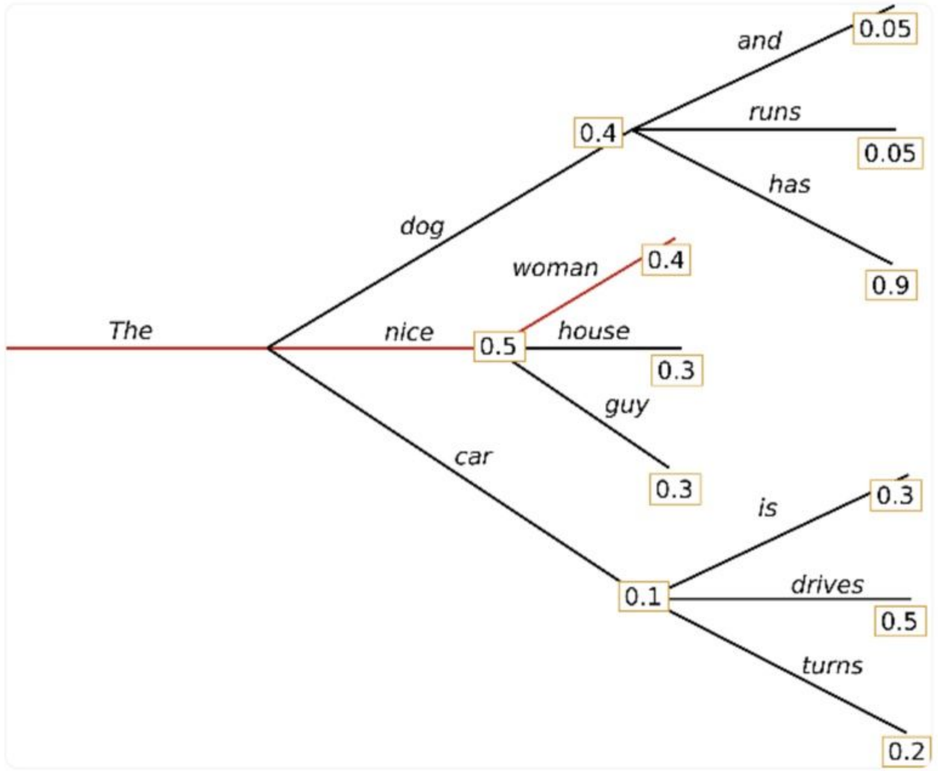
0.49 * 0.50 * 0.51 = 0.12

CTC лосс

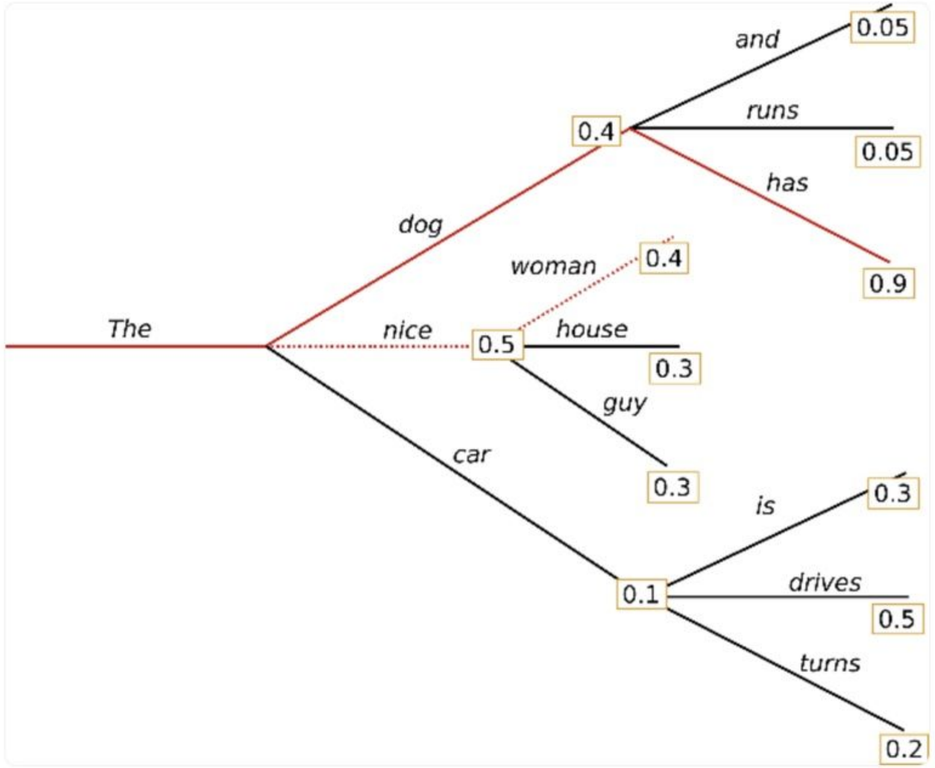
$$L(Y, \hat{Y}) = -\log \left(\sum_{C \in R(Y)} \prod_{t=1}^{T} p(c_t \mid \hat{Y}) \right)$$

Лучевой поиск (Beam search)

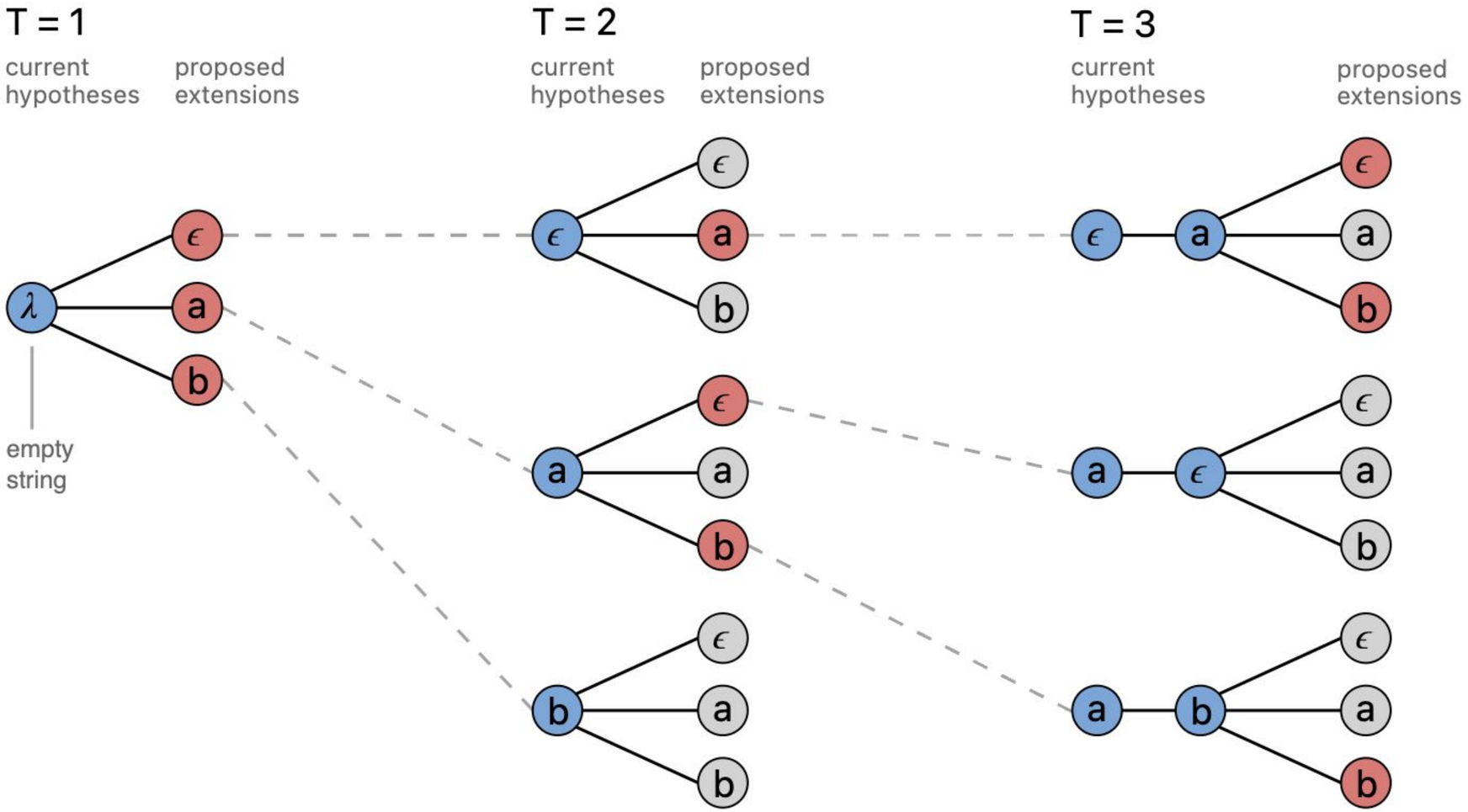
Greedy



Beam

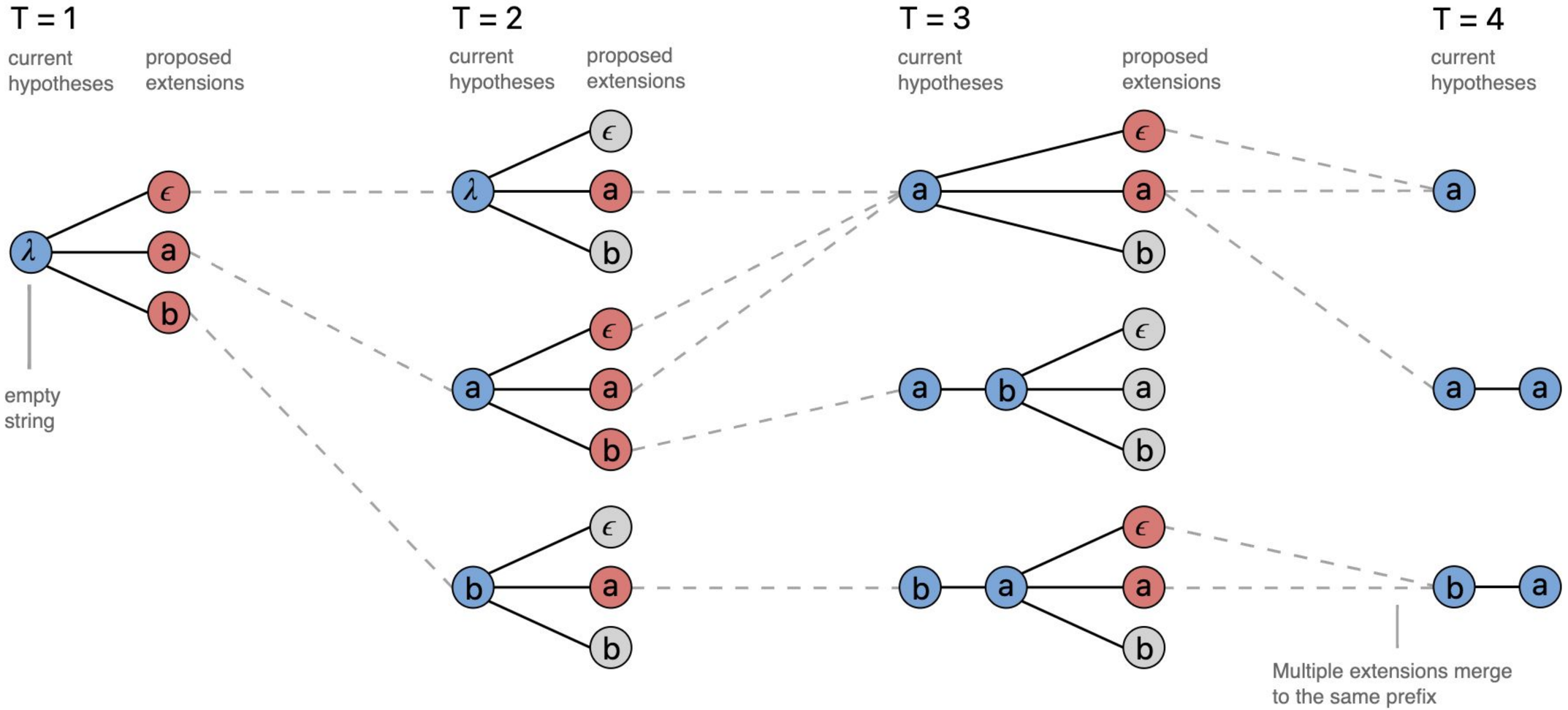


Лучевой поиск (Beam search)



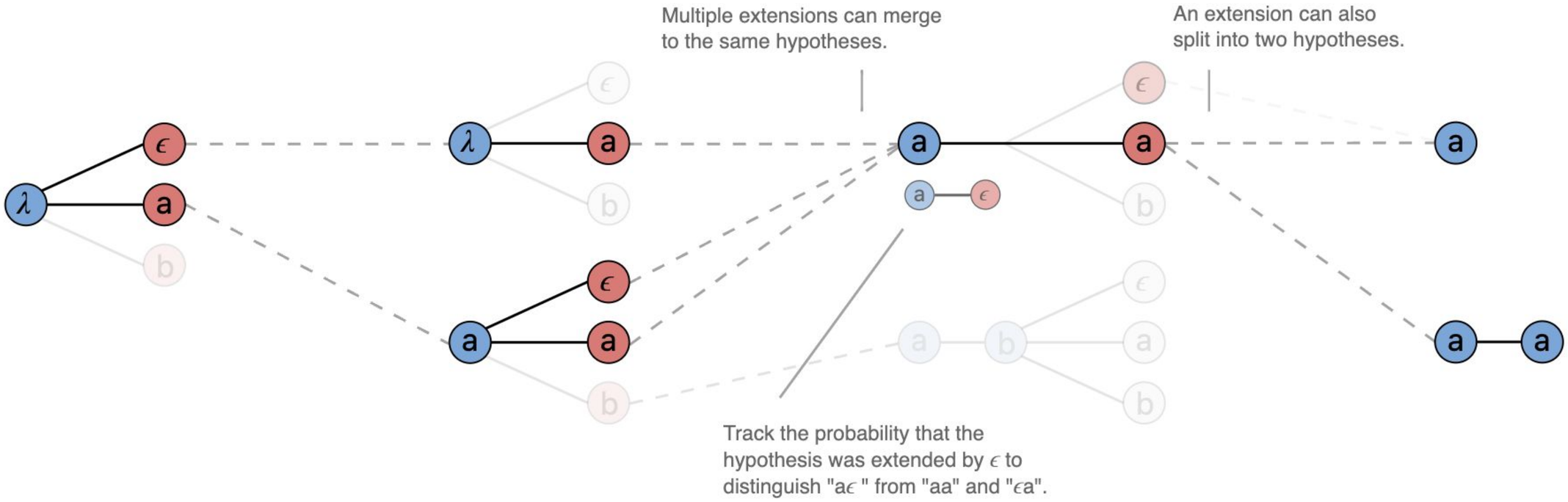
A standard beam search algorithm with an alphabet of $\{\epsilon, a, b\}$ and a beam size of three.

Лучевой поиск (Beam search)



The CTC beam search algorithm with an output alphabet $\{\epsilon, a, b\}$ and a beam size of three.

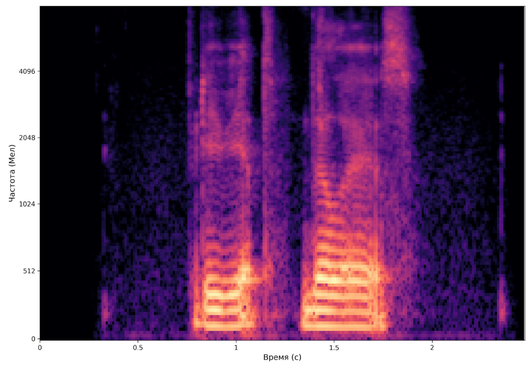
Лучевой поиск (Beam search)



Кодирование эмбеддингов

Данные

“привет дэлээс”



2D mel-spectrum
100 векторов в секунде

Encoder

CNN +
LSTM/Transformer

Linear projection

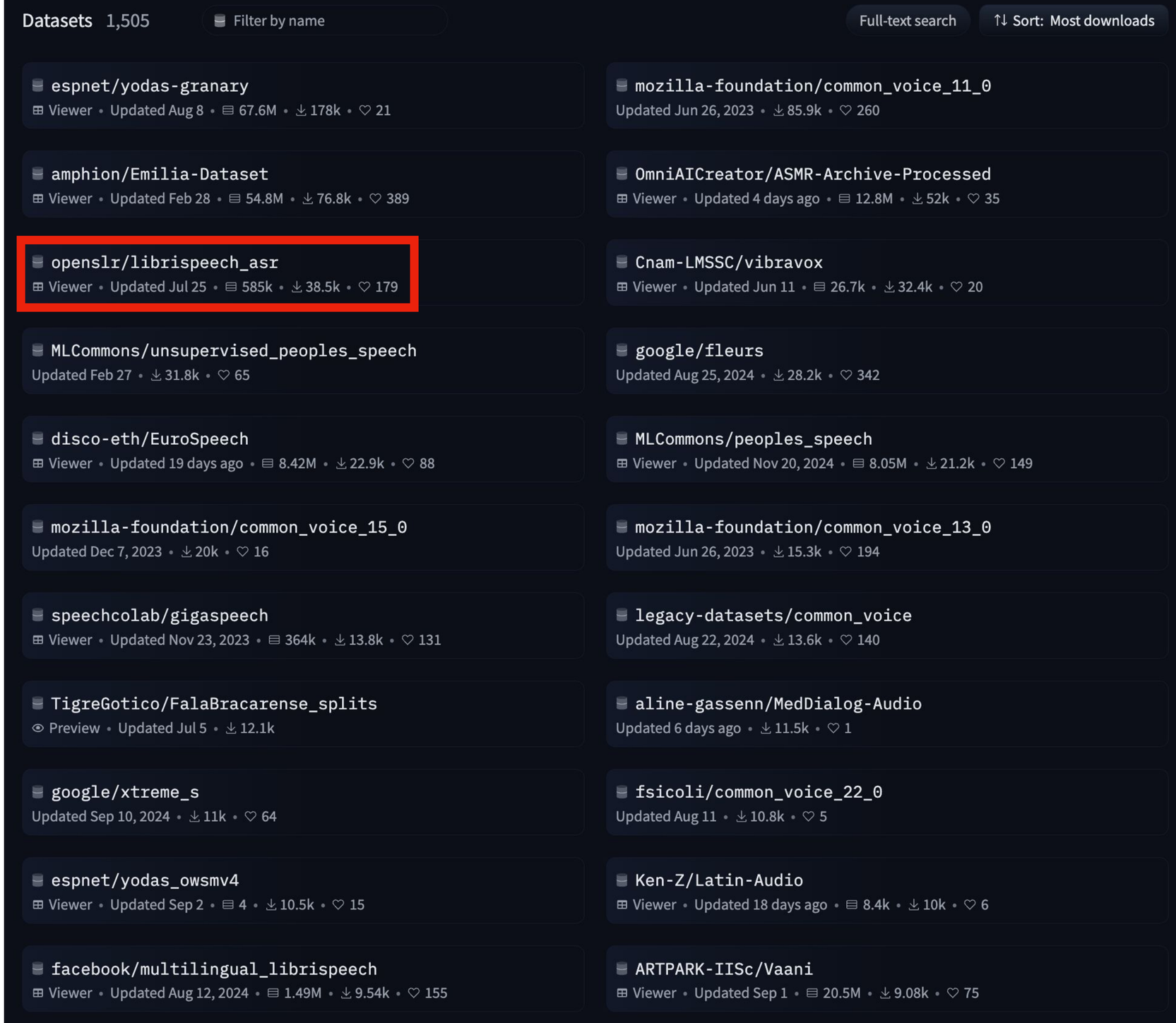
emb_size -> vocab_size

Датасеты

Datasets 1,505 | Filter by name | Full-text search | Sort: Most downloads

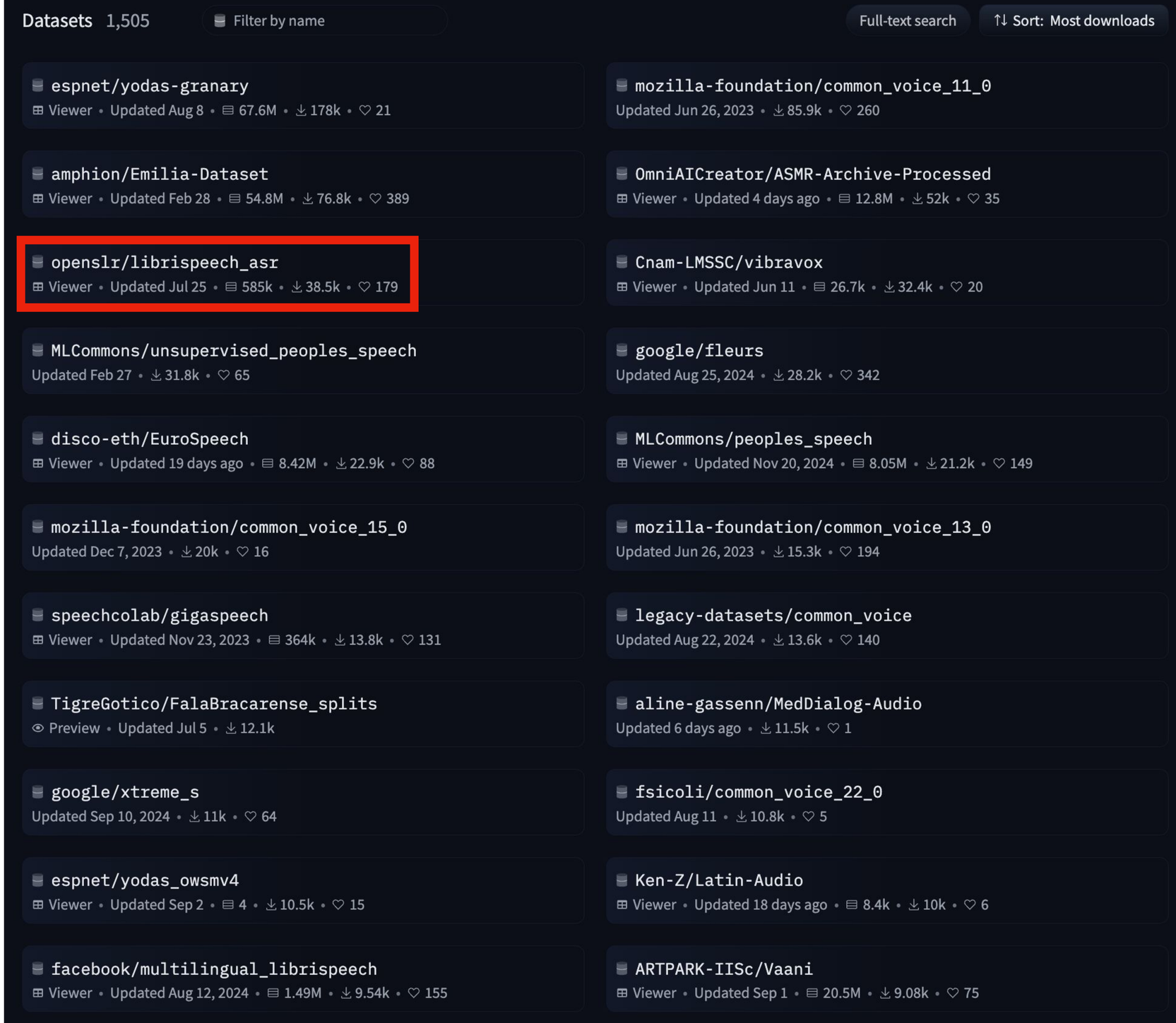
- espnet/yodas-granary
  Viewer • Updated Aug 8 • 67.6M • 178k • ♡ 21
- mozilla-foundation/common_voice_11_0
  Updated Jun 26, 2023 • 85.9k • ♡ 260
- amphion/Emilia-Dataset
  Viewer • Updated Feb 28 • 54.8M • 76.8k • ♡ 389
- OmniAICreator/ASMR-Archive-Processed
  Viewer • Updated 4 days ago • 12.8M • 52k • ♡ 35
- openslr/librispeech_asr
  Viewer • Updated Jul 25 • 585k • 38.5k • ♡ 179
- Cnam-LMSSC/vibravox
  Viewer • Updated Jun 11 • 26.7k • 32.4k • ♡ 20
- MLCommons/unsupervised_peoples_speech
  Updated Feb 27 • 31.8k • ♡ 65
- google/fleurs
  Updated Aug 25, 2024 • 28.2k • ♡ 342
- disco-eth/EuroSpeech
  Viewer • Updated 19 days ago • 8.42M • 22.9k • ♡ 88
- MLCommons/peoples_speech
  Viewer • Updated Nov 20, 2024 • 8.05M • 21.2k • ♡ 149
- mozilla-foundation/common_voice_15_0
  Updated Dec 7, 2023 • 20k • ♡ 16
- mozilla-foundation/common_voice_13_0
  Updated Jun 26, 2023 • 15.3k • ♡ 194
- speechcolab/gigaspeech
  Viewer • Updated Nov 23, 2023 • 364k • 13.8k • ♡ 131
- legacy-datasets/common_voice
  Updated Aug 22, 2024 • 13.6k • ♡ 140
- TigreGotico/FalaBracarense_splits
  Preview • Updated Jul 5 • 12.1k
- aline-gassenn/MedDialog-Audio
  Updated 6 days ago • 11.5k • ♡ 1
- google/xtreme_s
  Updated Sep 10, 2024 • 11k • ♡ 64
- fsicoli/common_voice_22_0
  Updated Aug 11 • 10.8k • ♡ 5
- espnet/yodas_owsmv4
  Viewer • Updated Sep 2 • 4 • 10.5k • ♡ 15
- Ken-Z/Latin-Audio
  Viewer • Updated 18 days ago • 8.4k • 10k • ♡ 6
- facebook/multilingual_librispeech
  Viewer • Updated Aug 12, 2024 • 1.49M • 9.54k • ♡ 155
- ARTPARK-IISc/Vaani
  Viewer • Updated Sep 1 • 20.5M • 9.08k • ♡ 75

Датасеты


Datasets 1,505
Filter by name
Full-text search
Sort: Most downloads
espnet/yodas-granary
Viewer • Updated Aug 8 • 67.6M • 178k • 21
mozilla-foundation/common_voice_11_0
Updated Jun 26, 2023 • 85.9k • 260
amphion/Emilia-Dataset
Viewer • Updated Feb 28 • 54.8M • 76.8k • 389
OmniAICreator/ASMR-Archive-Processed
Viewer • Updated 4 days ago • 12.8M • 52k • 35
openslr/librispeech_asr
Viewer • Updated Jul 25 • 585k • 38.5k • 179
Cnam-LMSSC/vibravox
Viewer • Updated Jun 11 • 26.7k • 32.4k • 20
MLCommons/unsupervised_peoples_speech
Updated Feb 27 • 31.8k • 65
google/fleurs
Updated Aug 25, 2024 • 28.2k • 342
disco-eth/EuroSpeech
Viewer • Updated 19 days ago • 8.42M • 22.9k • 88
MLCommons/peoples_speech
Viewer • Updated Nov 20, 2024 • 8.05M • 21.2k • 149
mozilla-foundation/common_voice_15_0
Updated Dec 7, 2023 • 20k • 16
mozilla-foundation/common_voice_13_0
Updated Jun 26, 2023 • 15.3k • 194
speechcolab/gigaspeech
Viewer • Updated Nov 23, 2023 • 364k • 13.8k • 131
legacy-datasets/common_voice
Updated Aug 22, 2024 • 13.6k • 140
TigreGotico/FalaBracarense_splits
Preview • Updated Jul 5 • 12.1k
aline-gassenn/MedDialog-Audio
Updated 6 days ago • 11.5k • 1
google/xtreme_s
Updated Sep 10, 2024 • 11k • 64
fsicoli/common_voice_22_0
Updated Aug 11 • 10.8k • 5
espnet/yodas_owsmv4
Viewer • Updated Sep 2 • 4 • 10.5k • 15
Ken-Z/Latin-Audio
Viewer • Updated 18 days ago • 8.4k • 10k • 6
facebook/multilingual_librispeech
Viewer • Updated Aug 12, 2024 • 1.49M • 9.54k • 155
ARTPARK-IISc/Vaani
Viewer • Updated Sep 1 • 20.5M • 9.08k • 75

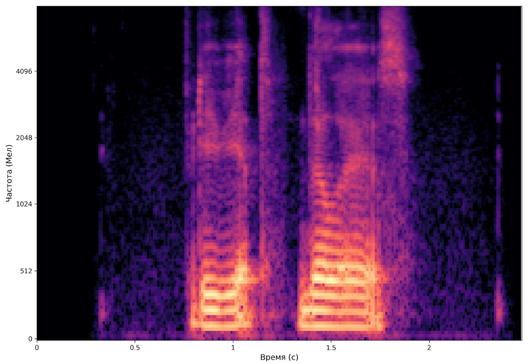
Датасеты

Subset Name	Hours	Cleanliness (Quality Tier)	Number of Speakers
train-clean-100	100.6	Clean	251
train-clean-360	363.6	Clean	921
train-other-500	496.7	Other (noisy)	1.166
dev-clean	5.4	Clean	40
dev-other	5.3	Other	33
test-clean	5.4	Clean	40
test-other	5.1	Other	33

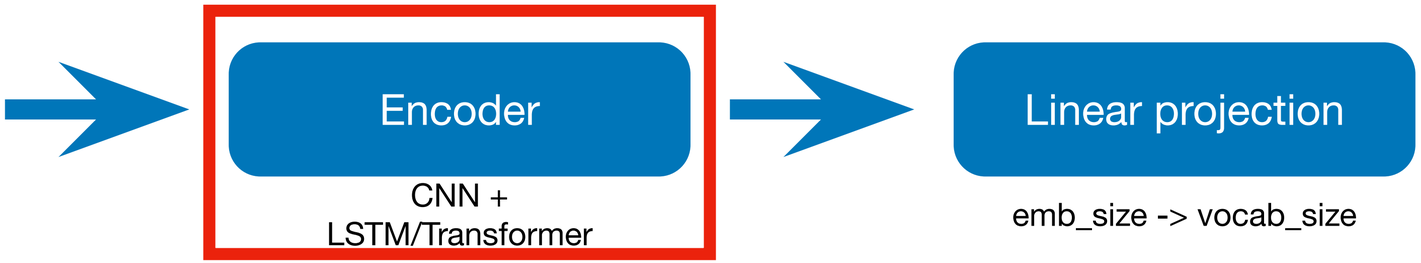


Архитектуры

“привет дэлээс”



2D mel-spectrum
100 векторов в секунде



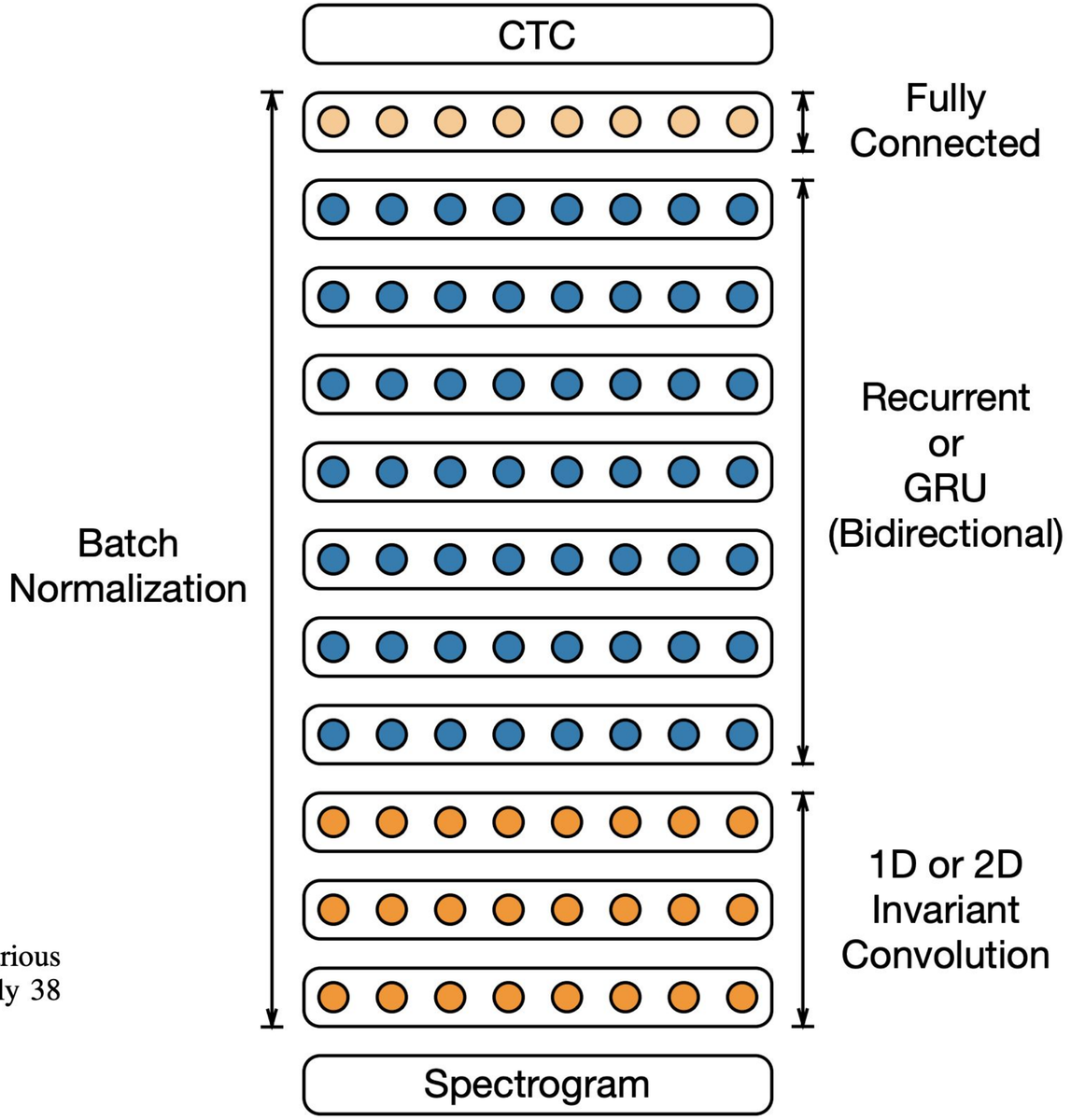
Deep Speech 2 (2015)

Read Speech			
Test set	DS1	DS2	Human
WSJ eval'92	4.94	3.60	5.03
WSJ eval'93	6.94	4.98	8.08
LibriSpeech test-clean	7.89	5.33	5.83
LibriSpeech test-other	21.74	13.25	12.69

Table 13: Comparison of WER for two speech systems and human level performance on read speech.

Architecture	Simple RNN	GRU
5 layers, 1 Recurrent	14.40	10.53
5 layers, 3 Recurrent	10.56	8.00
7 layers, 5 Recurrent	9.78	7.79
9 layers, 7 Recurrent	9.52	8.19

Table 3: Comparison of development set WER for networks with either simple RNN or GRU, for various depths. All models have batch normalization, one layer of 1D-invariant convolution, and approximately 38 million parameters.



Jasper (2019)

Table 2: *Normalization and Activation: Greedy WER, LibriSpeech after 50 epochs*

Model	Normalization	Activation	Dev	
			Clean	Other
Jasper 5x3	Batch Norm	ReLU	8.82	23.26
		cReLU	8.89	23.02
		lReLU	11.31	26.90
		GLU	9.46	24.30
		GAU	9.41	24.65
	Layer Norm (masked)	ReLU	8.82	**22.83**
		cReLU	9.14	23.26
		lReLU	11.29	26.35
		GLU	12.62	29.22
		GAU	**8.35**	23.07
	Weight Norm	ReLU	9.98	24.87
		cReLU	11.25	26.87
		lReLU	11.87	27.54
		GLU	11.05	27.10
		GAU	11.25	27.70
Jasper 10x4	Batch Norm	ReLU	**6.15**	**17.58**
	Layer Norm (Masked)	ReLU	6.56	18.48
		GAU	7.14	19.19

Table 3: *Sequence Masking: Greedy WER, LibriSpeech for Jasper 10x4 after 50 epochs*

Model	Masking	Dev	
		Clean	Other
Jasper DR 10x4	None	5.88	17.62
Jasper DR 10x4	BN Mask	5.92	17.63
Jasper DR 10x4	Conv Mask	**5.66**	**16.77**
Jasper DR 10x4	Conv+BN Mask	5.80	16.97

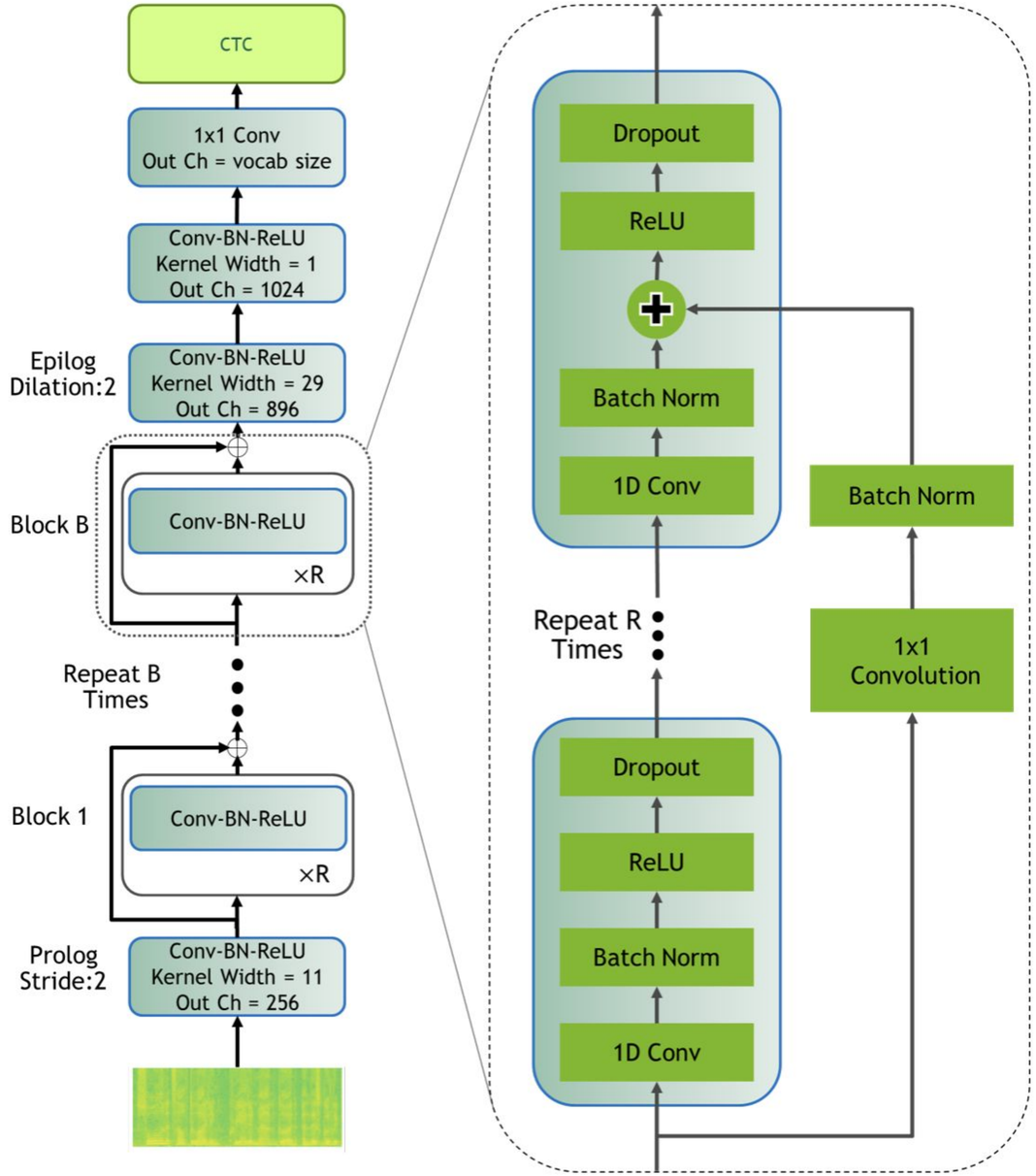


Figure 1: *Jasper BxR model: B - number of blocks, R - number of sub-blocks.*

Jasper (2019)

Table 5: *LibriSpeech, WER (%)*

Model	E2E	LM	dev-clean	dev-other	test-clean	test-other
CAPIO (single) [23]	N	RNN	3.02	8.28	3.56	8.58
pFSMN-Chain [25]	N	RNN	2.56	7.47	2.97	**7.5**
DeepSpeech2 [26]	Y	5-gram	-	-	5.33	13.25
Deep bLSTM w/ attention [21]	Y	LSTM	3.54	11.52	3.82	12.76
wav2letter++ [27]	Y	ConvLM	3.16	10.05	3.44	11.24
LAS + SpecAugment [4] [28]	Y	RNN	-	-	*2.5*	*5.8*
Jasper DR 10x5	Y	-	3.64	11.89	3.86	11.95
Jasper DR 10x5	Y	6-gram	2.89	9.53	3.34	9.62
Jasper DR 10x5	Y	Transformer-XL	2.68	8.62	**2.95**	8.79
Jasper DR 10x5 + Time/Freq Masks [4]	Y	Transformer-XL	2.62	7.61	2.84	7.84

Jasper (2019)

Table 5: *LibriSpeech, WER (%)*

Model	E2E	LM	dev-clean	dev-other	test-clean	test-other
CAPIO (single) [23]	N	RNN	3.02	8.28	3.56	8.58
pFSMN-Chain [25]	N	RNN	2.56	7.47	2.97	**7.5**
DeepSpeech2 [26]	Y	5-gram	-	-	5.33	13.25
Deep bLSTM w/ attention [21]	Y	LSTM	3.54	11.52	3.82	12.76
wav2letter++ [27]	Y	ConvLM	3.16	10.05	3.44	11.24
LAS + SpecAugment [4] [28]	Y	RNN	-	-	*2.5*	*5.8*
Jasper DR 10x5	Y	-	3.64	11.89	3.86	11.95
Jasper DR 10x5	Y	6-gram	2.89	9.53	3.34	9.62
Jasper DR 10x5	Y	Transformer-XL	2.68	8.62	**2.95**	8.79
Jasper DR 10x5 + Time/Freq Masks [4]	Y	Transformer-XL	2.62	7.61	2.84	7.84

QuartzNet (2020)

3.2. Pointwise convolutions with groups

The total number of weights for a time-channel separable convolution block is $K \times c_{in} + c_{in} \times c_{out}$ weights. Since K is generally several times smaller than c_{out}, most weights are concentrated in the pointwise convolution part. In order to further reduce the number of parameters, we explore using group convolutions for this layer. We also added a group shuffle layer to increase cross-group interchange [5].

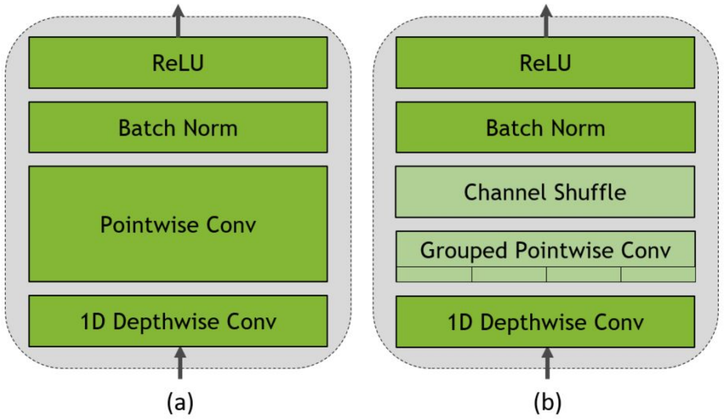


Fig. 2. (a) Time-channel separable 1D convolutional module (b) Time-channel separable 1D convolutional module with groups and shuffle

Using groups allows us to significantly reduce the number of weights at the cost of some accuracy. Table 3 shows the trade-off between accuracy and number of parameters for group sizes one, two, and four, evaluated on LibriSpeech.

Table 3. QuartzNet-15x5 with grouped convolutions trained on LibriSpeech for 300 epochs, greedy WER (%)

# Groups	dev-clean	dev-other	Params, M
1	3.98	11.58	18.9
2	4.29	12.52	12.1
4	4.51	13.48	8.70

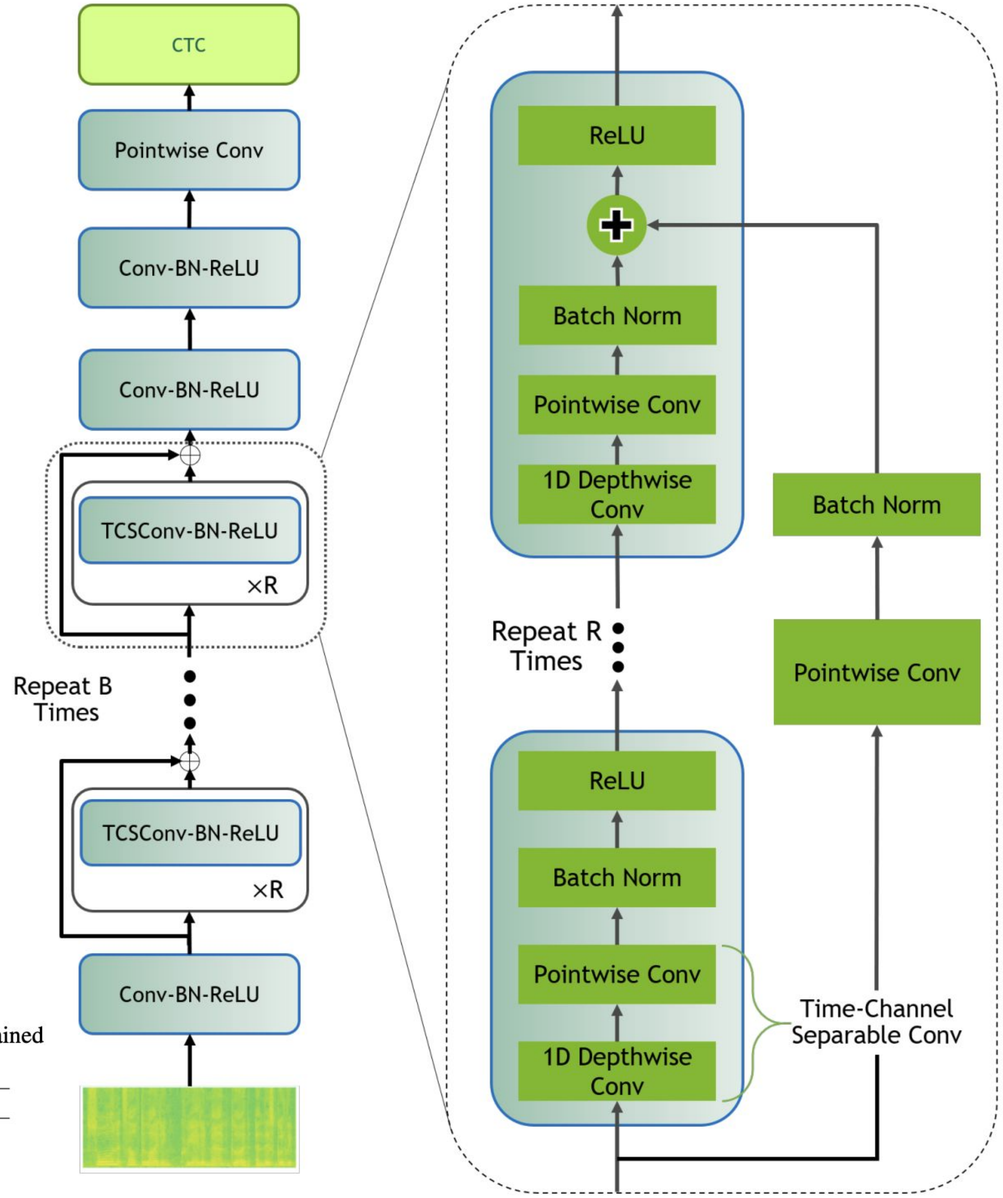


Fig. 1. QuartzNet BxR architecture

QuartzNet (2020)

Table 4. LibriSpeech results, WER (%)

Model	Augment	LM	Test clean	Test other	Params,M
wav2letter++ [27]	speed perturb	ConvLM	3.26	10.47	208
LAS [23]	SpecAugment	RNN	2.5	5.8	360
TDS Conv [12]	dropout+	-	5.36	15.64	37
	label smooth	4-gram	4.21	11.87	
		ConvLM	3.28	9.84	
MSSA[13]	speed perturb	4-gram	2.93	8.32	23
		4-LSTM	2.20	5.82	
JasperDR-10x5[14]	SpecAugment+	-	4.32	11.82	333
	speed perturb	6-gram	3.24	8.76	
		T-XL	2.84	7.84	
QuartzNet 15x5	SpecCutout+	-	3.90	11.28	19
	speed perturb	6-gram	2.96	8.07	
		T-XL	2.69	7.25	

QuartzNet (2020)

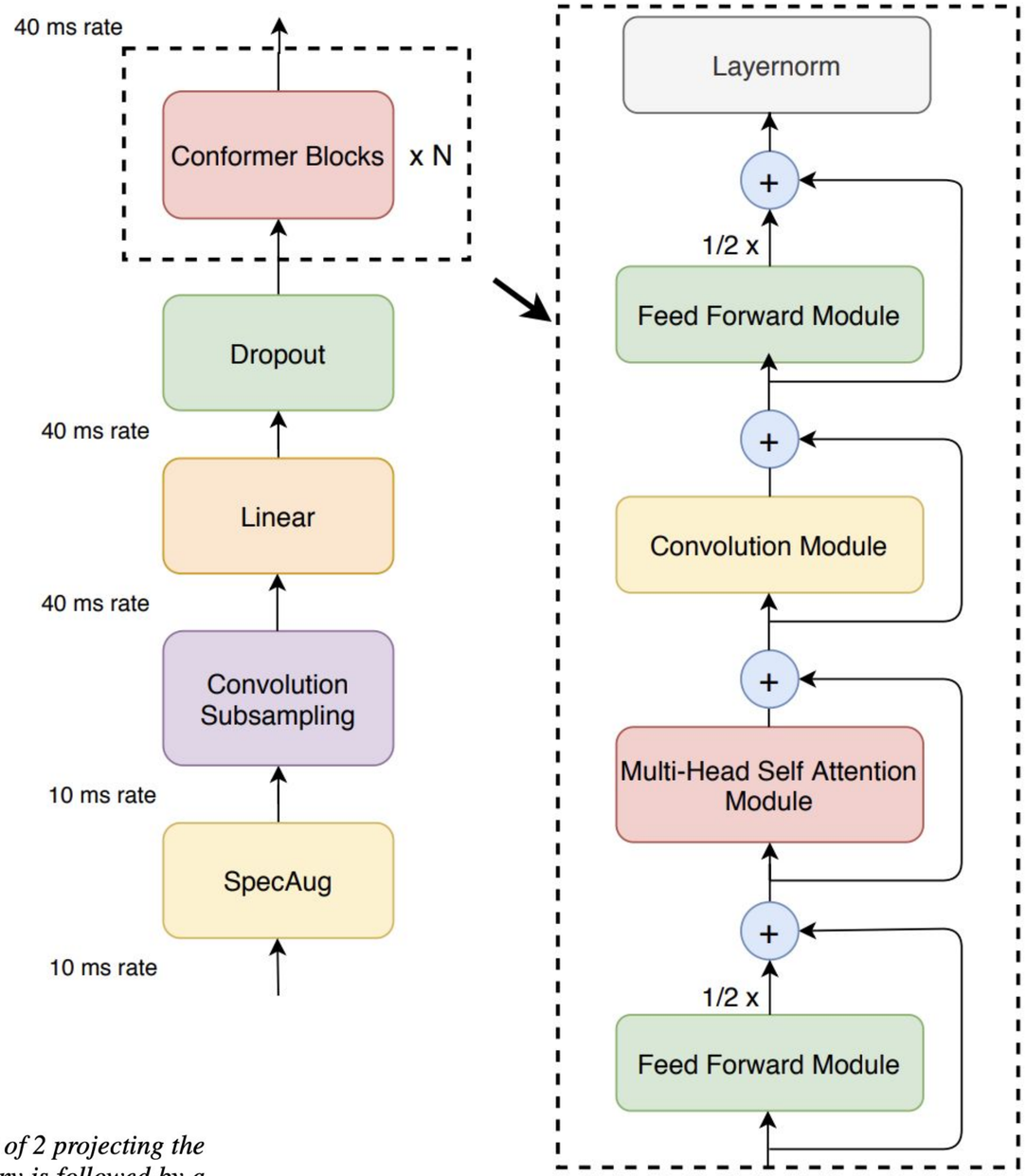
Table 4. LibriSpeech results, WER (%)

Model	Augment	LM	Test clean	Test other	Params,M
wav2letter++ [27]	speed perturb	ConvLM	3.26	10.47	208
LAS [23]	SpecAugment	RNN	2.5	5.8	360
TDS Conv [12]	dropout+	-	5.36	15.64	37
	label smooth	4-gram	4.21	11.87	
		ConvLM	3.28	9.84	
MSSA[13]	speed perturb	4-gram	2.93	8.32	23
		4-LSTM	2.20	5.82	
JasperDR-10x5[14]	SpecAugment+	-	4.32	11.82	333
	speed perturb	6-gram	3.24	8.76	
		T-XL	2.84	7.84	
QuartzNet 15x5	SpecCutout+	-	3.90	11.28	19
	speed perturb	6-gram	2.96	8.07	
		T-XL	2.69	7.25	

Conformer (2020)

4. Conclusion

In this work, we introduced Conformer, an architecture that integrates components from CNNs and Transformers for end-to-end speech recognition. We studied the importance of each component, and demonstrated that the inclusion of convolution modules is critical to the performance of the Conformer model. The model exhibits better accuracy with fewer parameters than previous work on the LibriSpeech dataset, and achieves a new state-of-the-art performance at 1.9%/3.9% for test/testother.



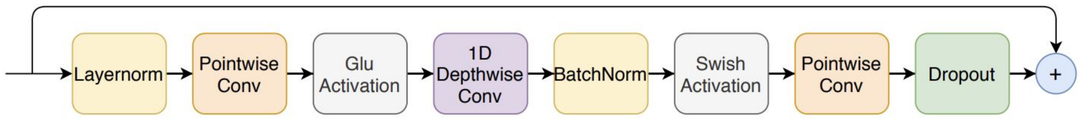


Figure 2: ***Convolution module.*** *The convolution module contains a pointwise convolution with an expansion factor of 2 projecting the number of channels with a GLU activation layer, followed by a 1-D Depthwise convolution. The 1-D depthwise conv is followed by a Batchnorm and then a swish activation layer.*

Zipformer (2023)

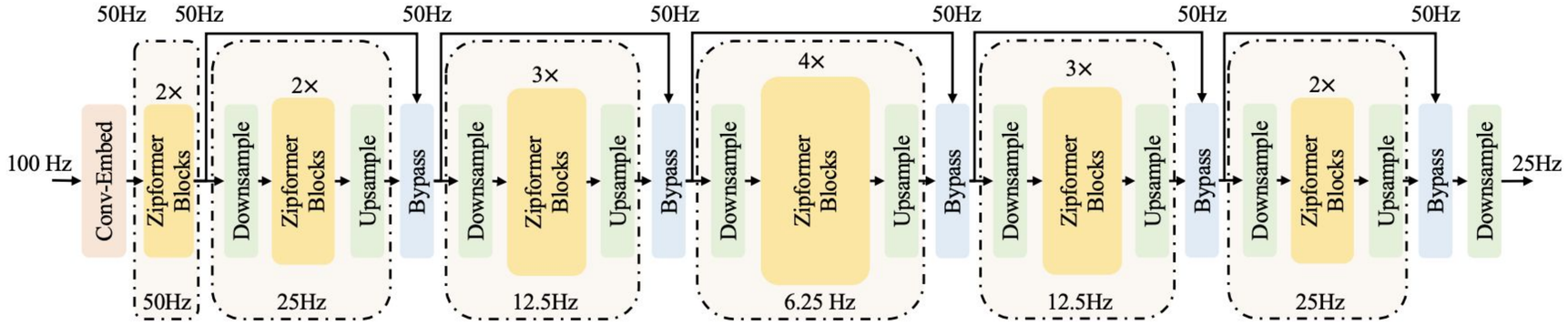


Figure 1: Overall architecture of Zipformer.

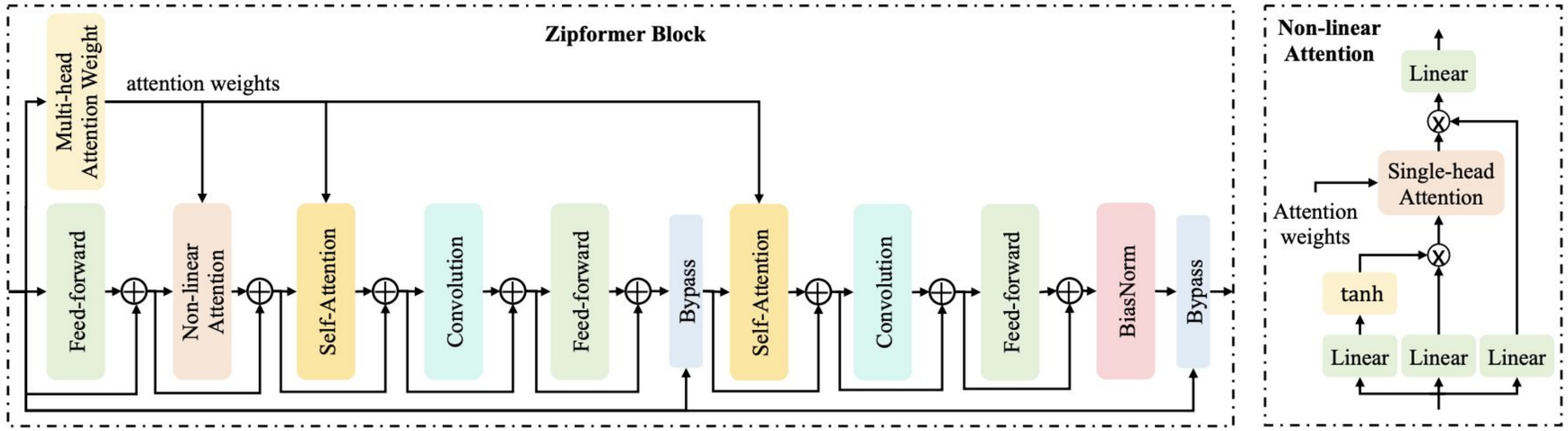
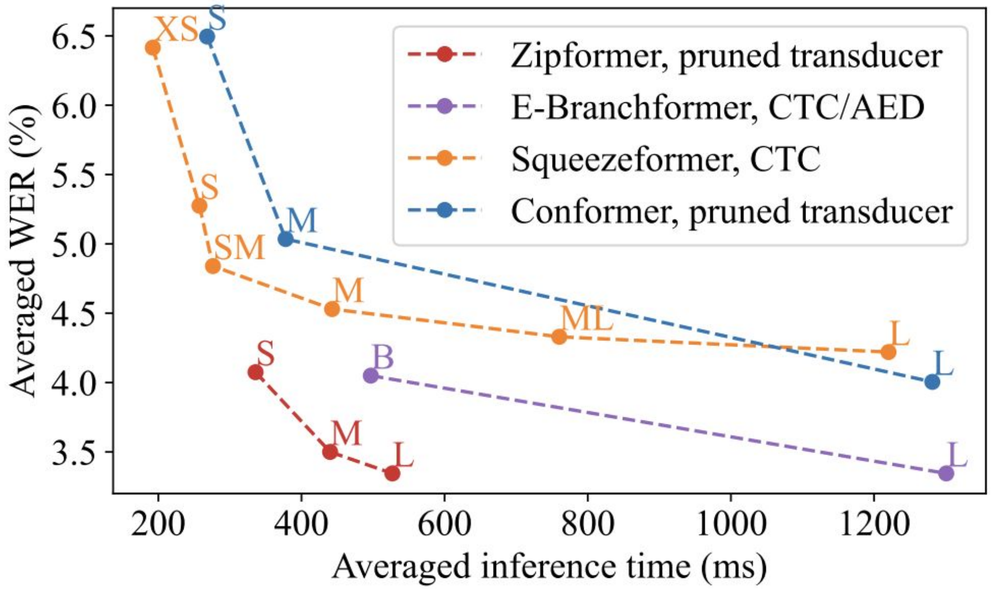
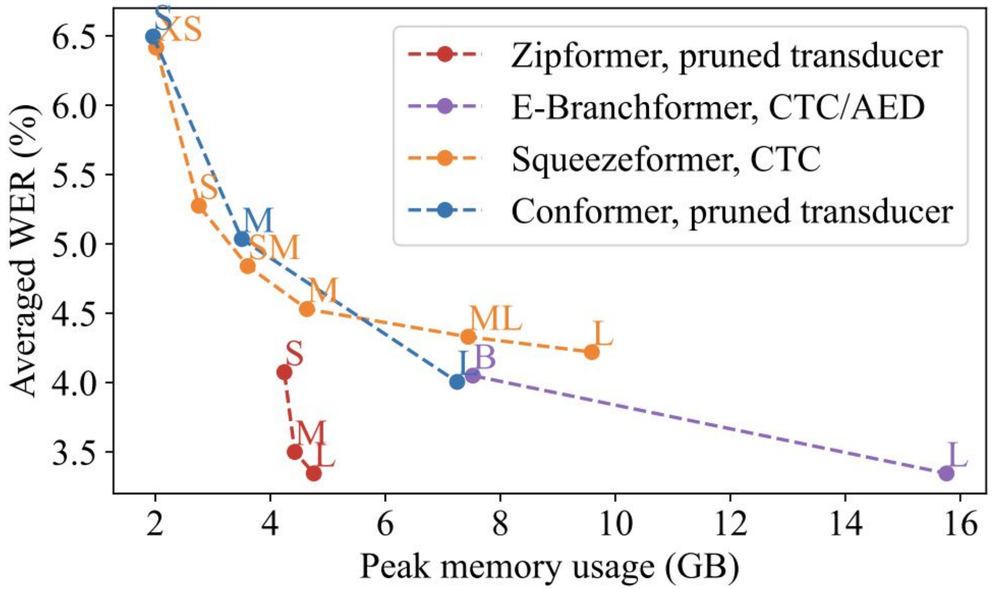


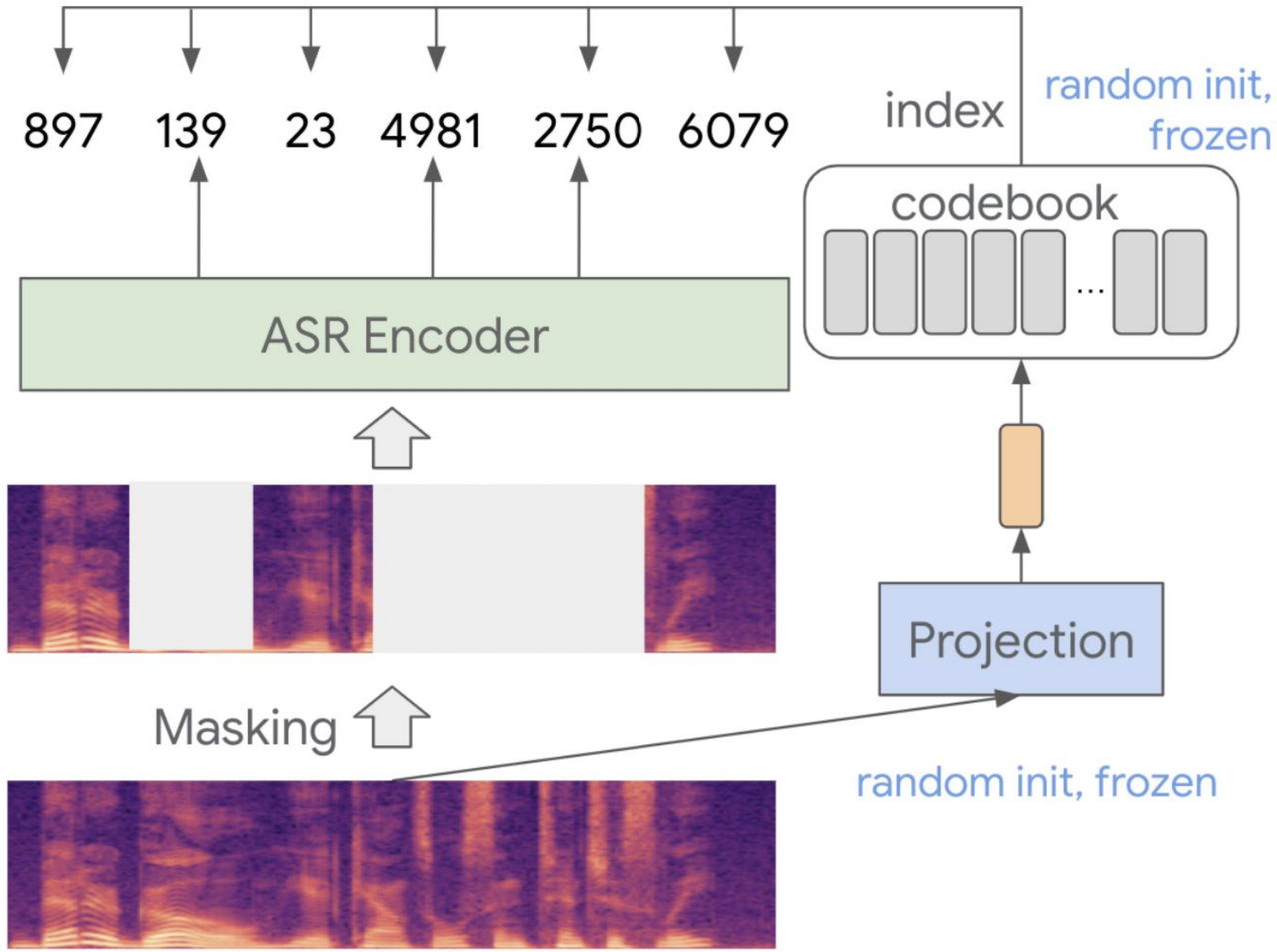
Figure 2: (Left): Zipformer block structure. (Right): Non-Linear Attention module structure.

Zipformer (2023)





SSL предобучение



SSL предобучение

Table 1. LibriSpeech results with non-streaming models. The LM used in our experiment is a Transfomer LM with model size 0.1B.

Method	Size (B)	**No LM**				**With LM**			
		dev	**dev-other**	**test**	**test-other**	**dev**	**dev-other**	**test**	**test-other**
wav2vec 2.0 (Baevski et al., 2020b)	0.3	2.1	4.5	2.2	4.5	1.6	3.0	1.8	3.3
HuBERT Large (Hsu et al., 2021)	0.3	–	–	–	–	1.5	3.0	1.9	3.3
HuBERT X-Large (Hsu et al., 2021)	1.0	–	–	–	–	1.5	**2.5**	1.8	2.9
w2v-Conformer XL (Zhang et al., 2020)	0.6	1.7	3.5	1.7	3.5	1.6	3.2	**1.5**	3.2
w2v-BERT XL (Chung et al., 2021)	0.6	**1.5**	2.9	**1.5**	**2.9**	**1.4**	2.8	**1.5**	2.8
BEST-RQ (Ours)	0.6	**1.5**	**2.8**	1.6	**2.9**	**1.4**	2.6	**1.5**	**2.7**

Encoder - Decoder архитектура

Whisper (2022)

Multitask training data (680k hours)

English transcription

"Ask not what your country can do for ⋯"

Ask not what your country can do for ⋯

Any-to-English speech translation

"El rápido zorro marrón salta sobre ⋯"

The quick brown fox jumps over ⋯

Non-English transcription

"언덕 위에 올라 내려다보면 너무나 넓고 넓은 ⋯"

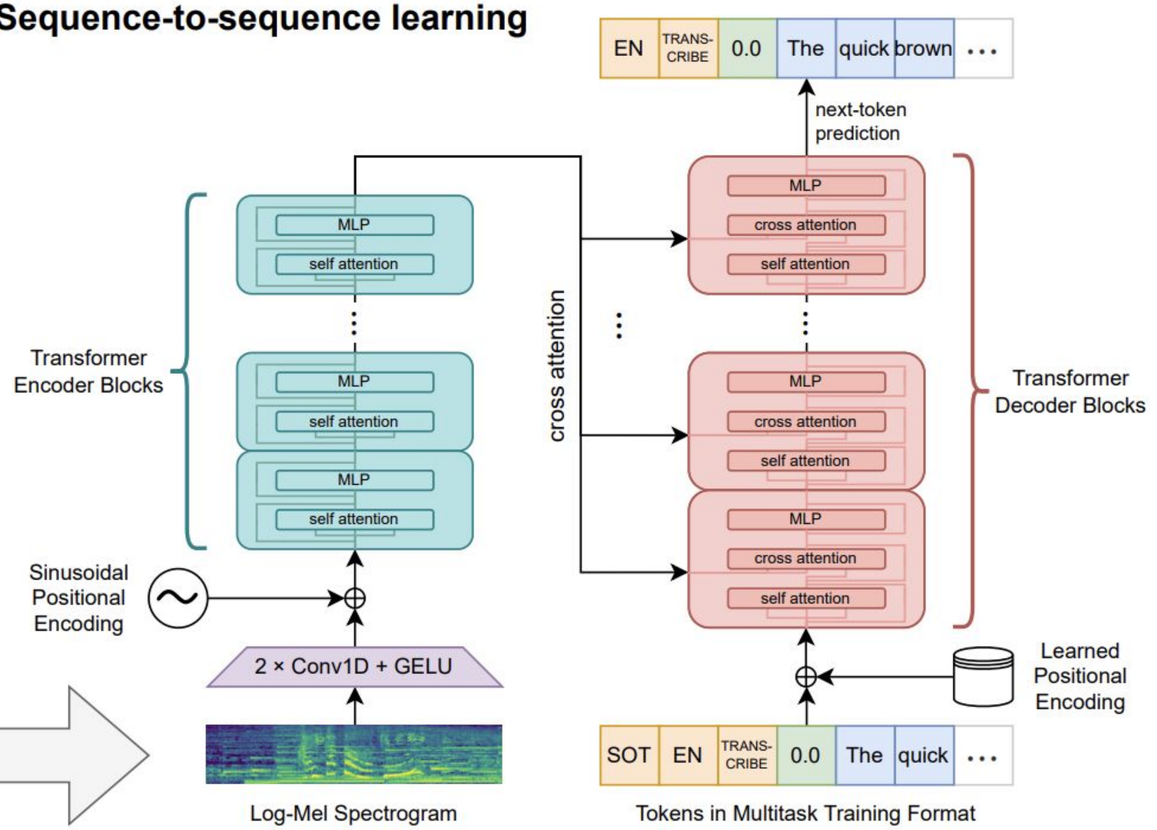
언덕 위에 올라 내려다보면 너무나 넓고 넓은 ⋯

No speech

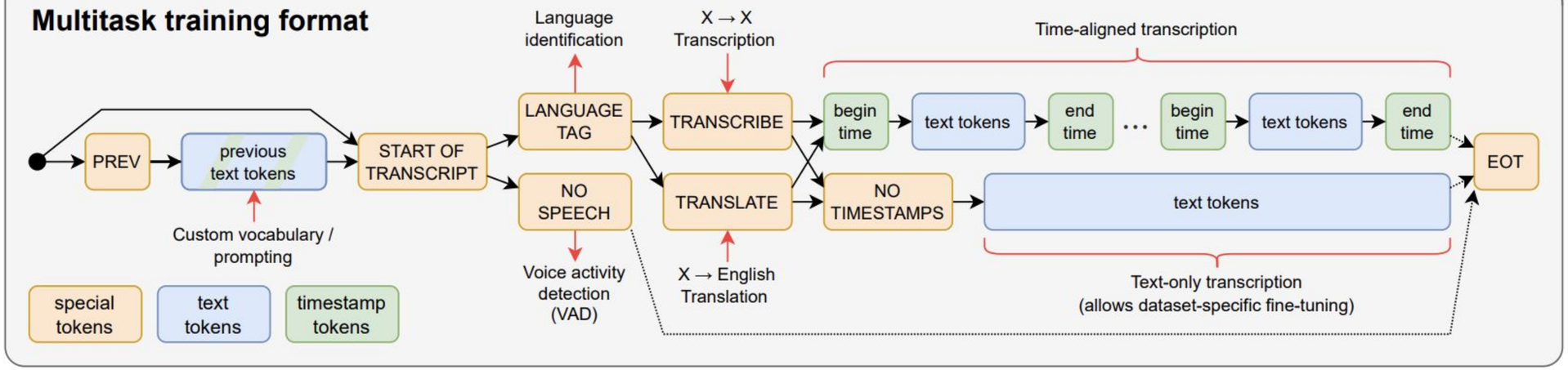
(background music playing)

∅

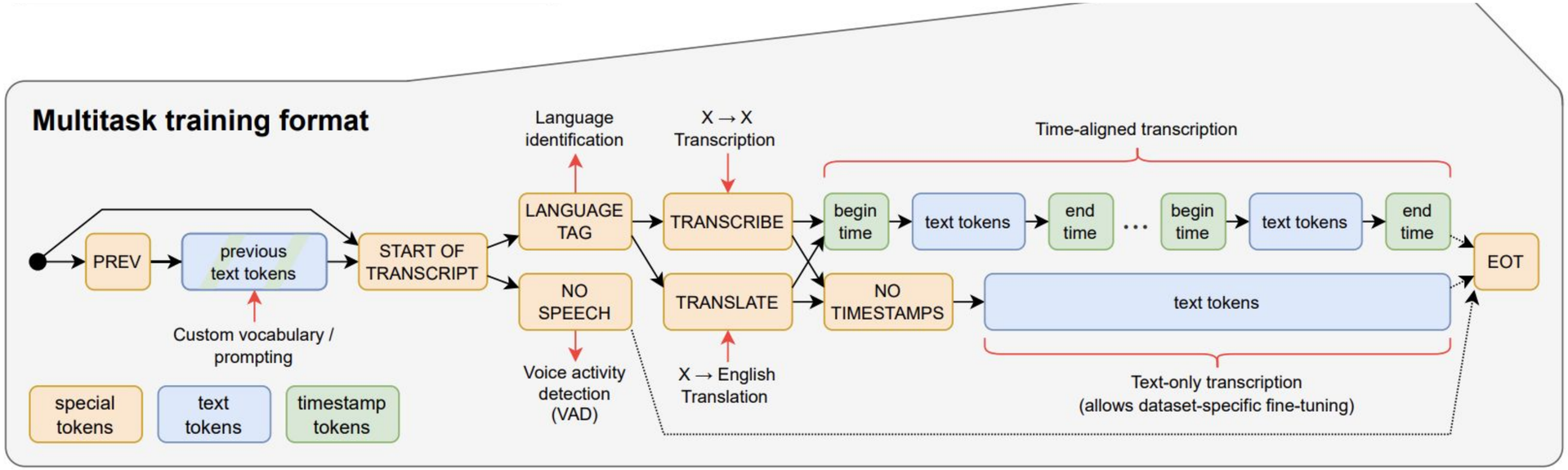
Sequence-to-sequence learning



Multitask training format



Whisper (2022)

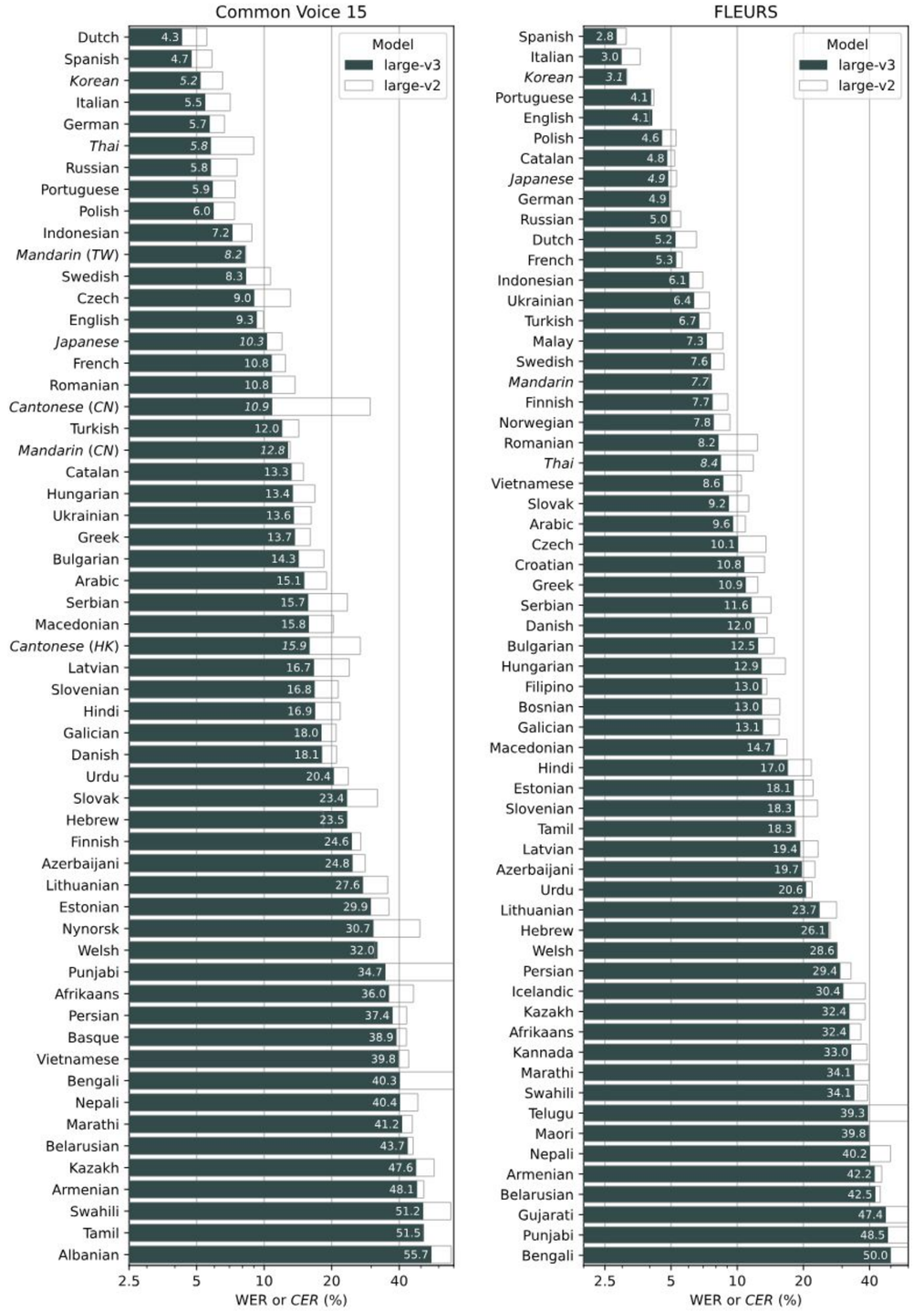


Whisper (2022)

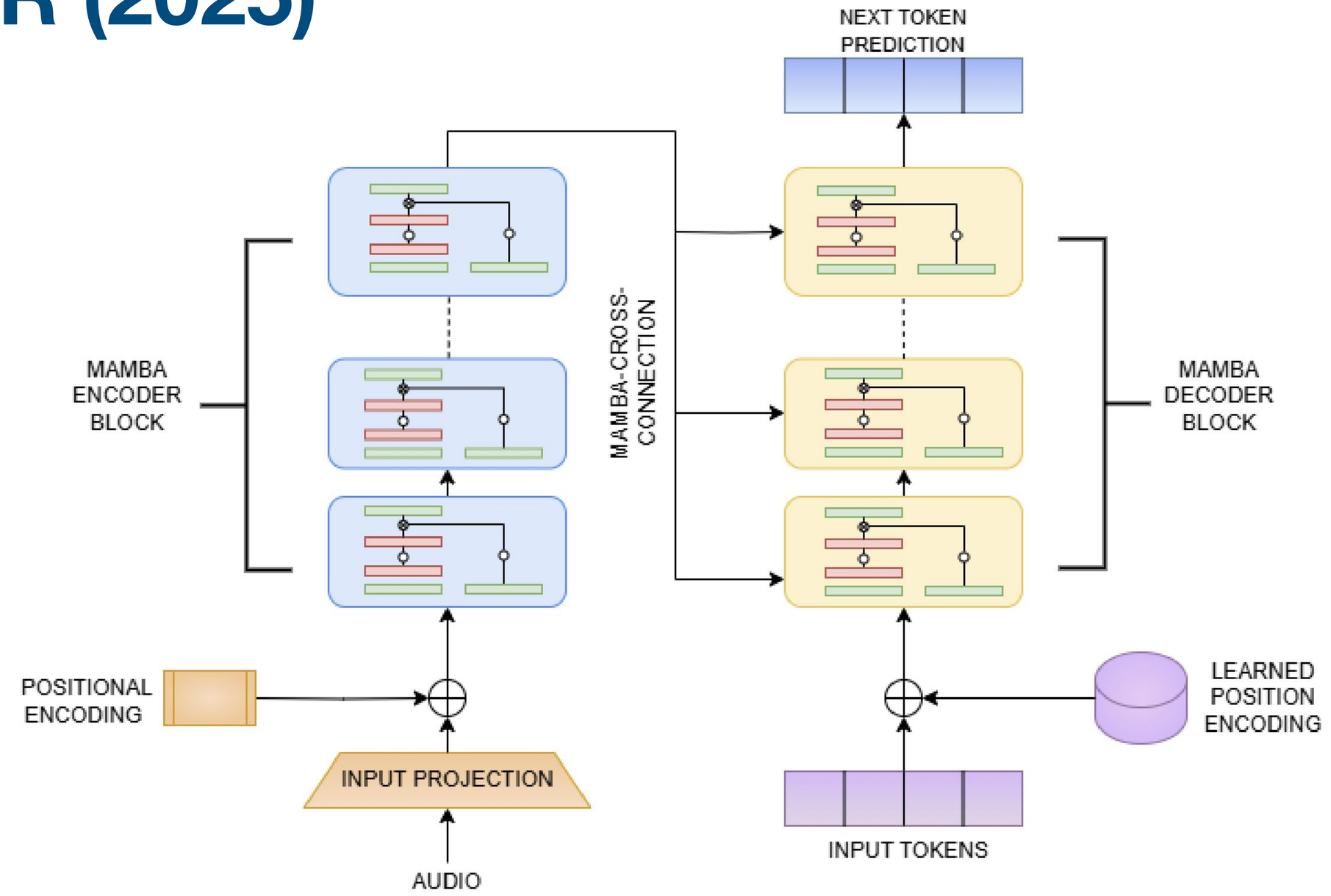
Dataset	wav2vec 2.0 Large (no LM)	Whisper Large V2	RER (%)
LibriSpeech Clean	**2.7**	**2.7**	0.0
Artie	24.5	**6.2**	74.7
Common Voice	29.9	**9.0**	69.9
Fleurs En	14.6	**4.4**	69.9
Tedlium	10.5	**4.0**	61.9
CHiME6	65.8	**25.5**	61.2
VoxPopuli En	17.9	**7.3**	59.2
CORAAL	35.6	**16.2**	54.5
AMI IHM	37.0	**16.9**	54.3
Switchboard	28.3	**13.8**	51.2
CallHome	34.8	**17.6**	49.4
WSJ	7.7	**3.9**	49.4
AMI SDM1	67.6	**36.4**	46.2
LibriSpeech Other	6.2	**5.2**	16.1
Average	29.3	**12.8**	55.2

Whisper (2022)

Size	Parameters	English-only model	Multilingual model	Required VRAM	Relative speed
tiny	39 M	`tiny.en`	`tiny`	~1 GB	~10x
base	74 M	`base.en`	`base`	~1 GB	~7x
small	244 M	`small.en`	`small`	~2 GB	~4x
medium	769 M	`medium.en`	`medium`	~5 GB	~2x
large	1550 M	N/A	`large`	~10 GB	1x
turbo	809 M	N/A	`turbo`	~6 GB	~8x



Samba-ASR (2025)



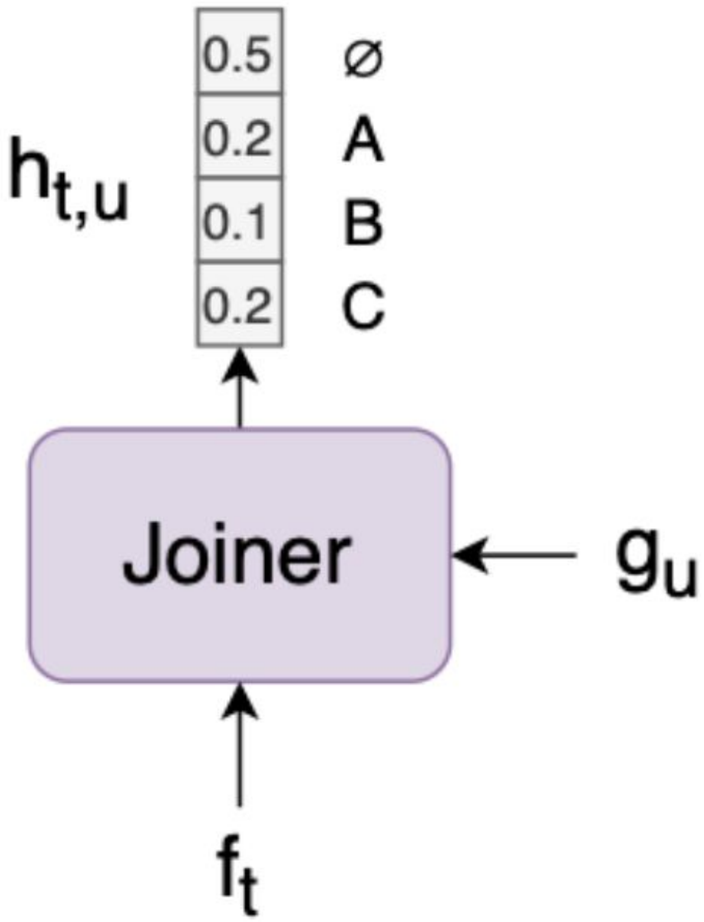
Samba-ASR (2025)

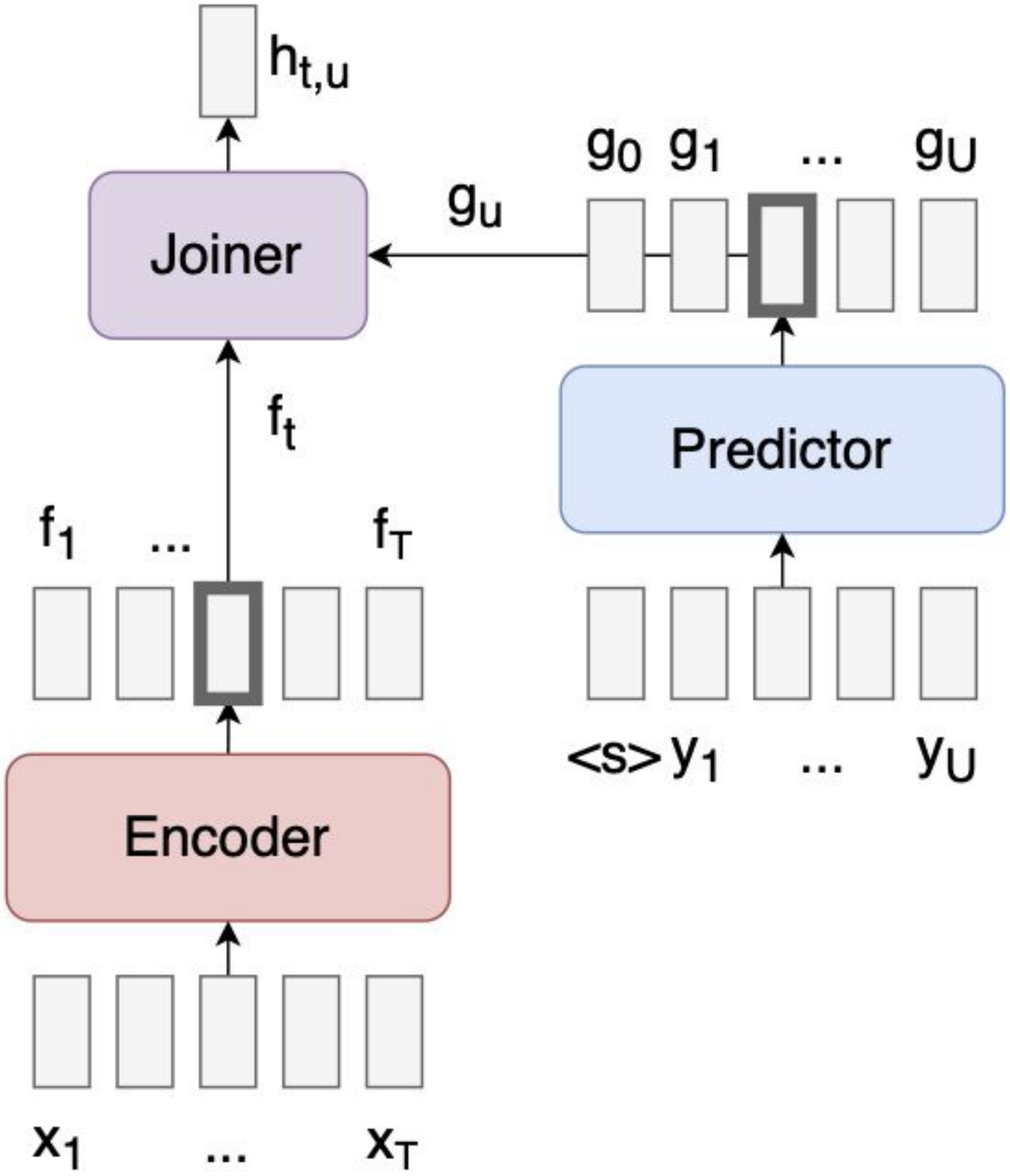
Model	Average WER	Gigaspeech	LS Clean	LS Other	SPGISpeech
Samba-ASR (SandLogic)	**3.65**	**9.12**	**1.17**	**2.48**	**1.84**
nvidia/canary-1b	4.15	10.12	1.48	2.93	2.06
nyrahealth/CrisperWhisper	4.69	10.24	1.82	4.00	2.7
nvidia/parakeet-tdt-1.1b	7.01	9.52	1.40	2.60	3.16
openai/whisper-large-v3	7.44	10.02	2.01	3.91	2.94

Table 2: Model Performance Comparison Across Various Datasets

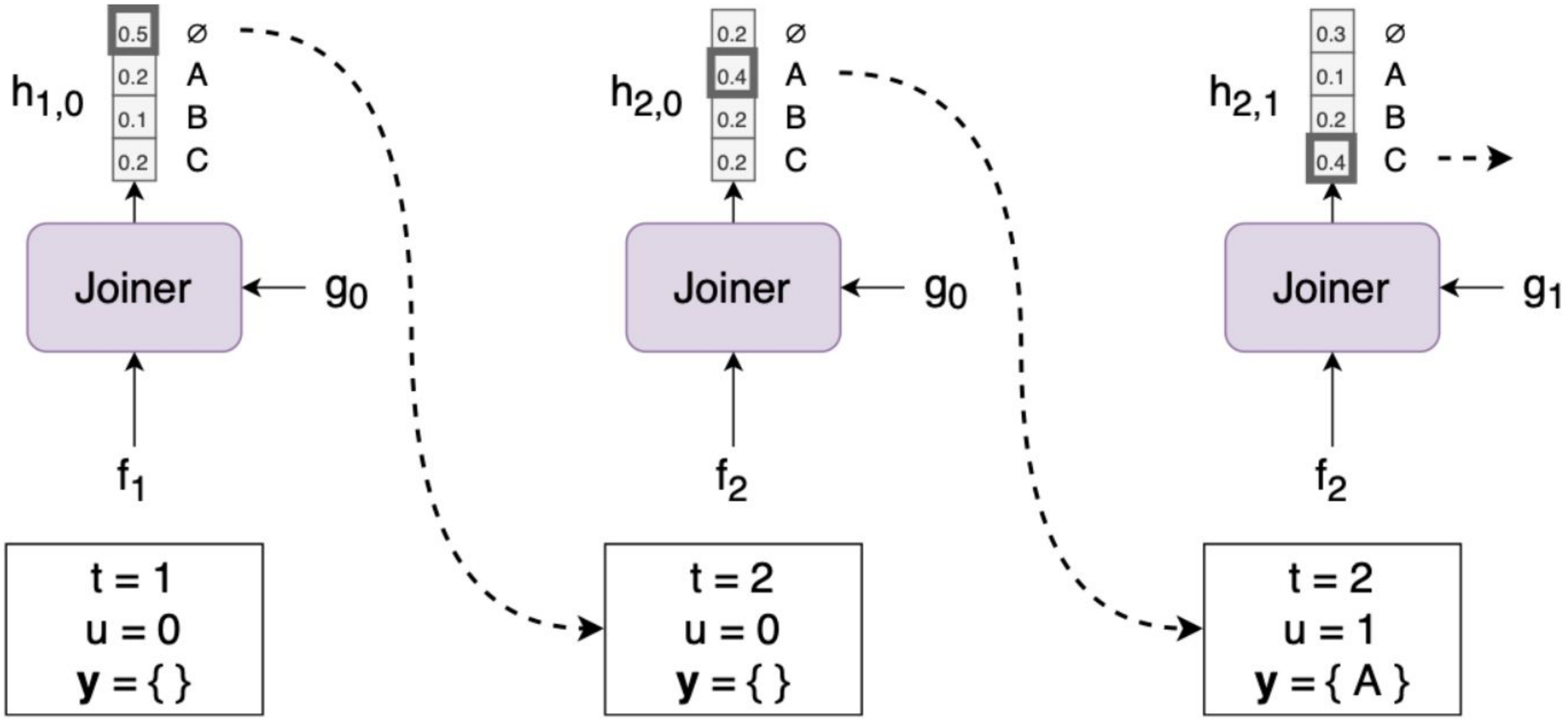
RNN-Transducer архитектура

RNN-Transducer

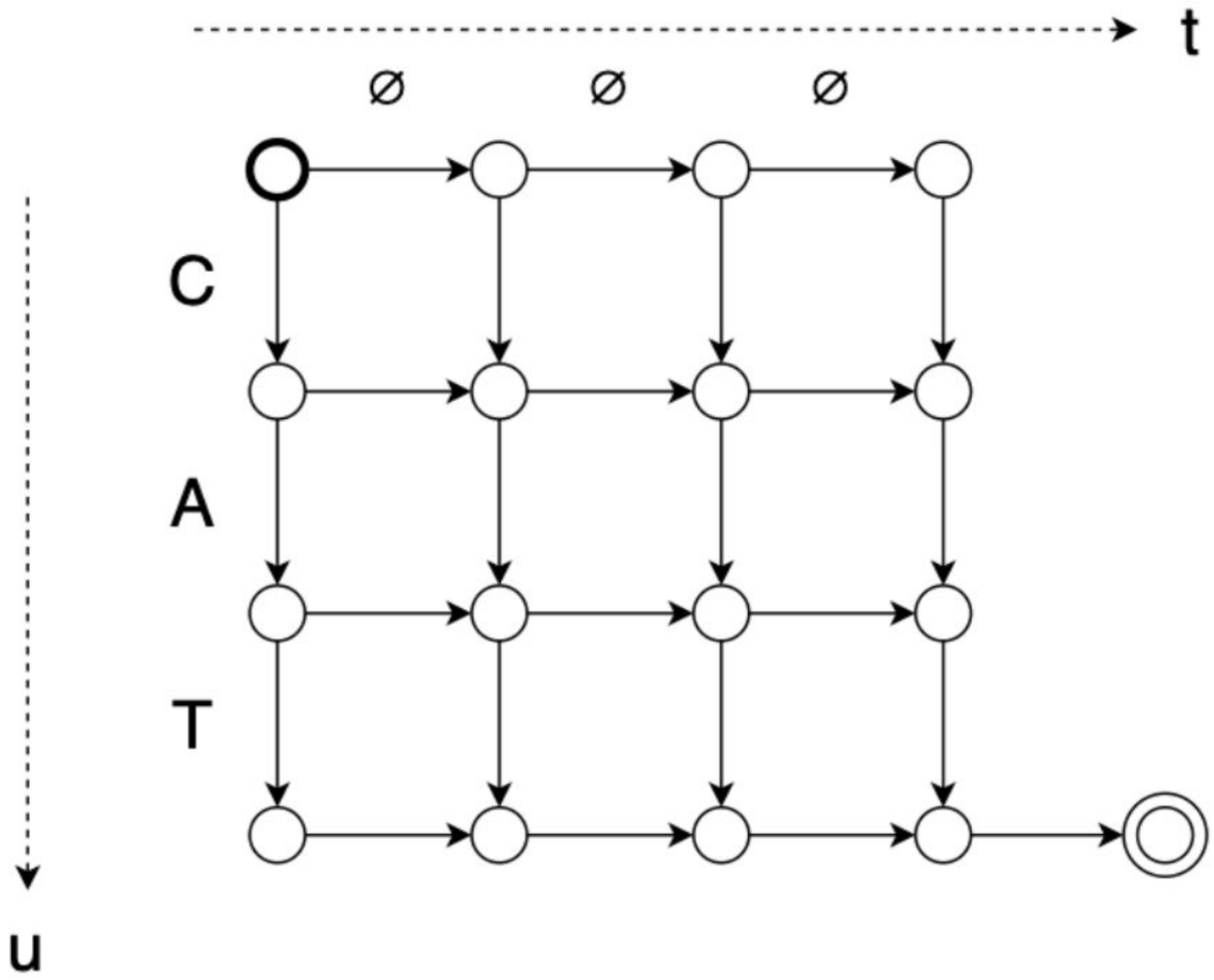




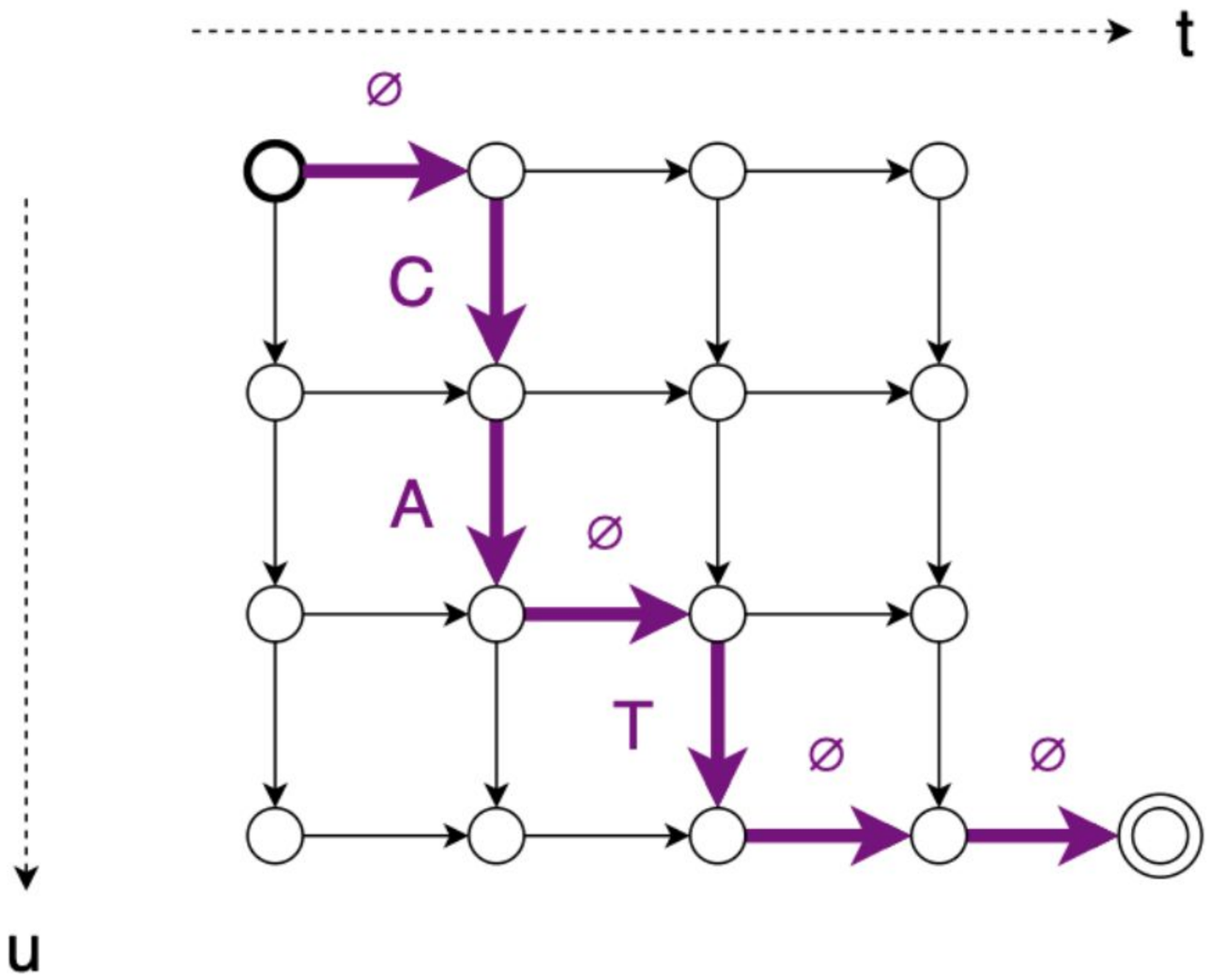
RNN-Transducer



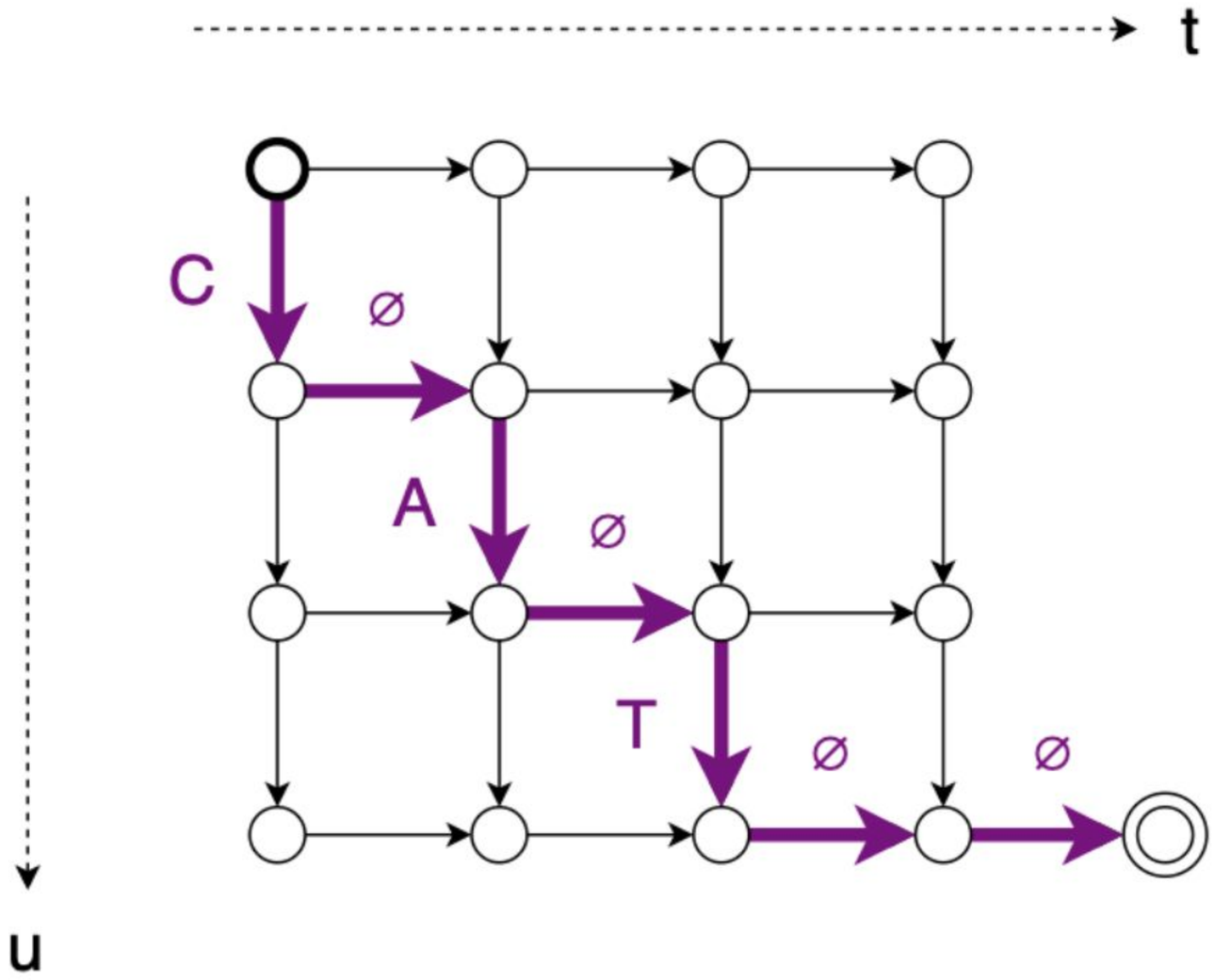
RNN-Transducer



RNN-Transducer



RNN-Transducer



RNN-Transducer