

Lecture 11: Audio and Speech

Guest lecture
by Yuriy Baburov

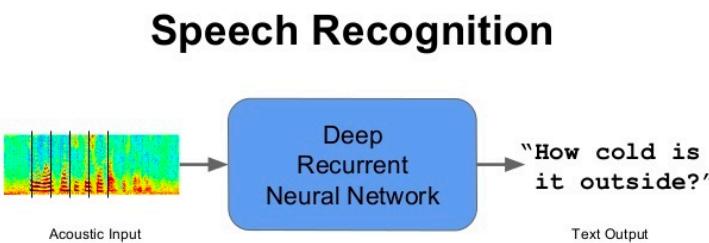
Задача: Распознавание речи

Она же ASR, TTS

- На входе: аудио, содержащее речь

- На выходе: текст

- Казалось бы, что может быть проще?



Reduced word errors by more than 30%

Research at Google

Google Research Blog - August 2012, August 2015

- Одна из самых полезных задач, связанных с машинным анализом аудио-данных.

[Вики: Speech recognition](#)

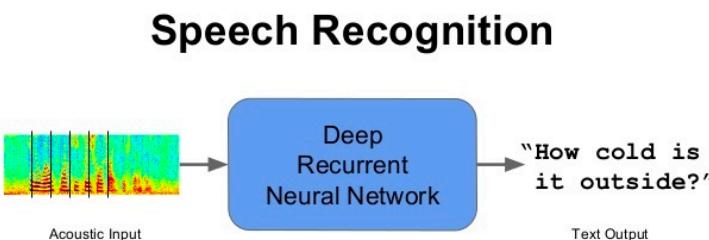
Задача: Распознавание речи

Она же ASR, TTS

- На входе: аудио, содержащее речь

- На выходе: текст

- Казалось бы, что может быть проще?



Reduced word errors by more than 30%

Google Research at Google

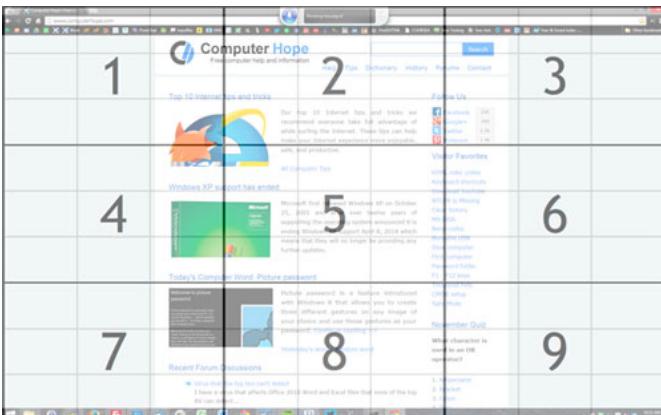
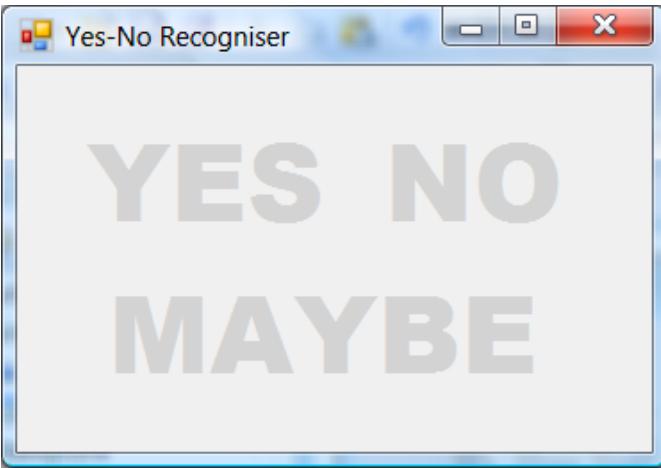


- Одна из самых полезных задач, связанных с машинным анализом аудио-данных.
- Прошла долгий путь.
- Гражданское и государственное применение

[Вики: Speech recognition](#)

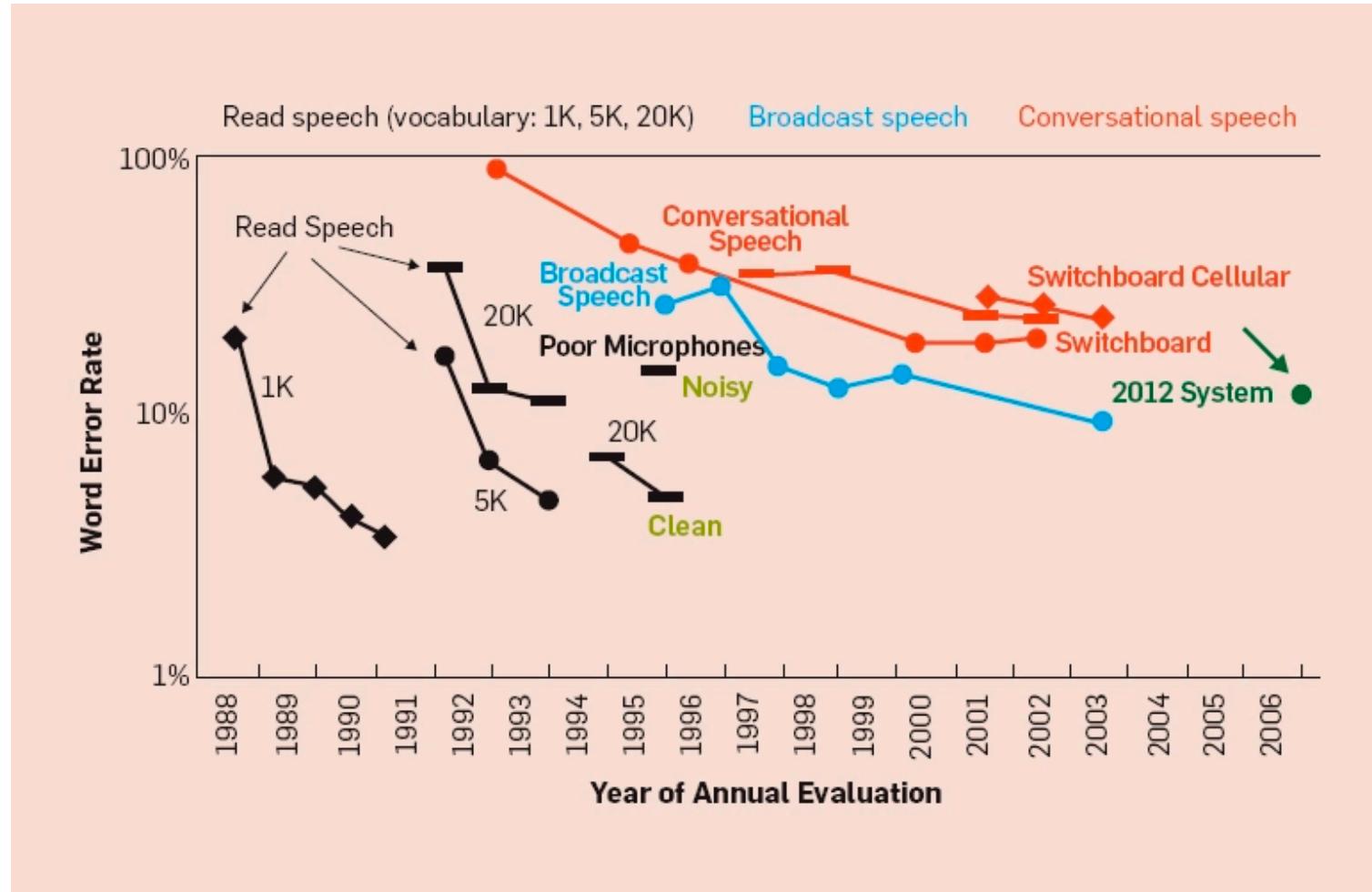
Этапы и качество 1

- Чем меньше классов -- тем лучше достигнуть хорошего качества
- Поэтому, несколько разные задачи:
 - Распознавание да/нет, цифр
 - Распознавание 100 команд
 - Распознавание раздельных слов из словаря в 1000 слов
 - Распознавание 10k, 100k слитных слов
- Прогресса в простых задачах добиться удалось...



Этапы и качество 2

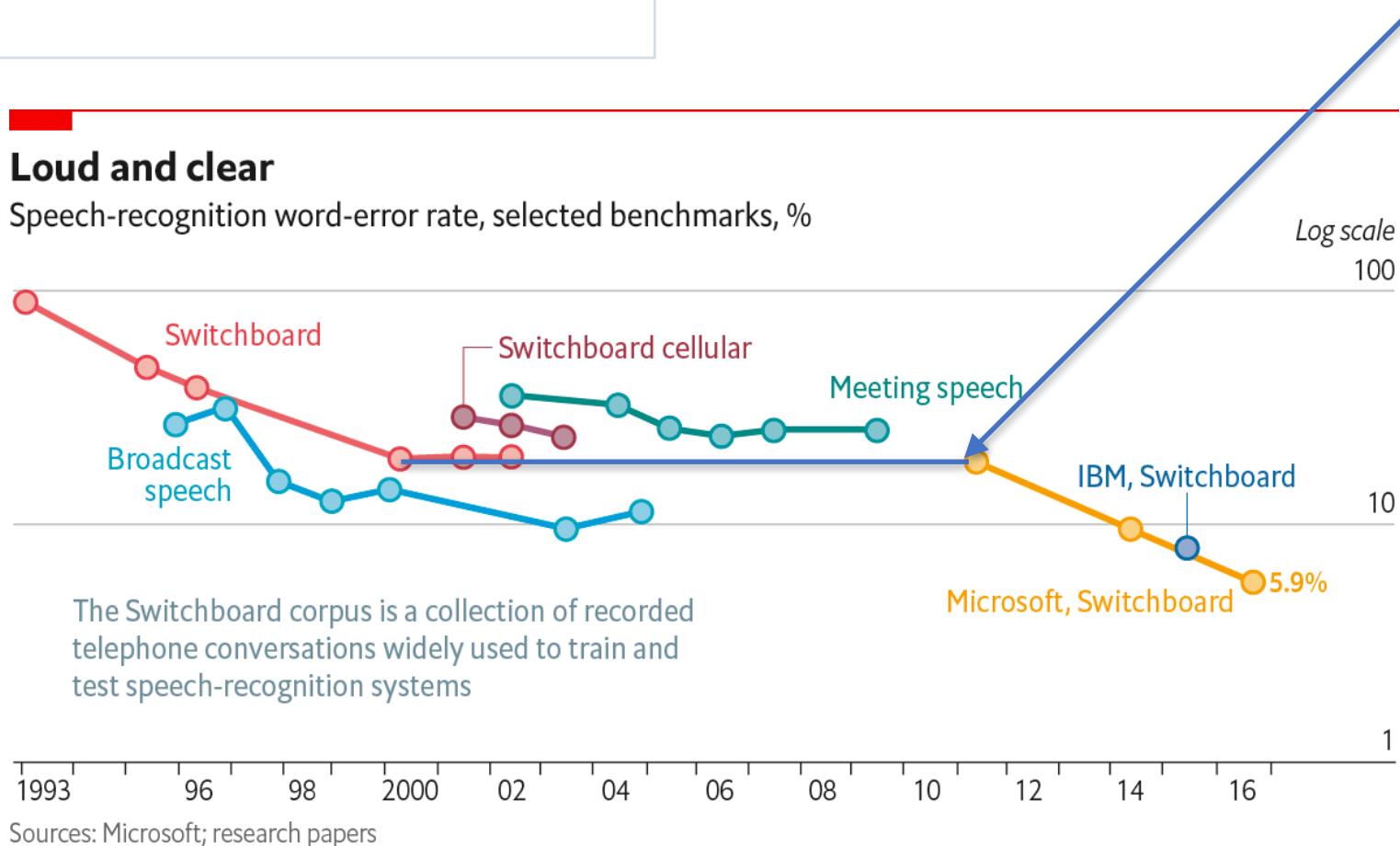
- Word Error Rate:
процент
неправильно
распознанных слов
- Влияет размер
словаря, качество
записи и наличие
шума



Этапы и качество 3

Loud and clear

Speech-recognition word-error rate, selected benchmarks, %

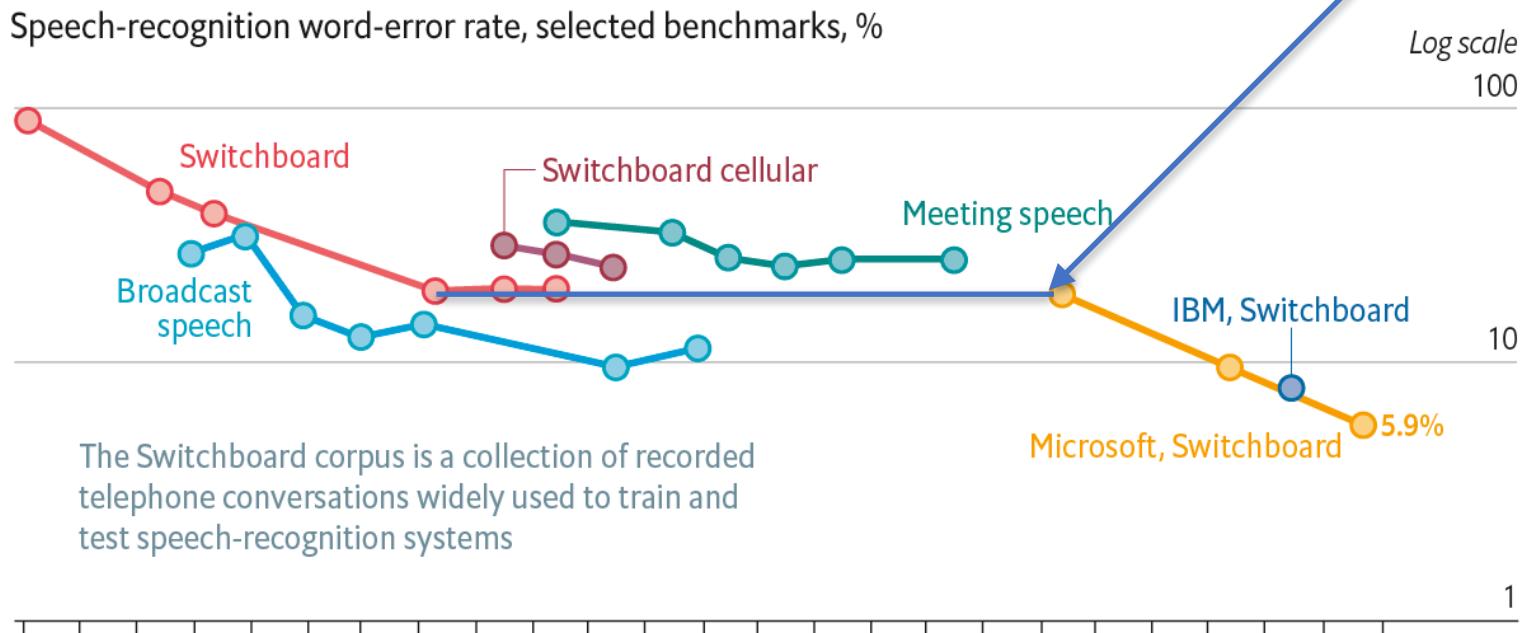


Этапы и качество 3



Loud and clear

Speech-recognition word-error rate, selected benchmarks, %



The Switchboard corpus is a collection of recorded telephone conversations widely used to train and test speech-recognition systems

Sources: Microsoft; research papers

Ничего не происходило (1999 - 2011)

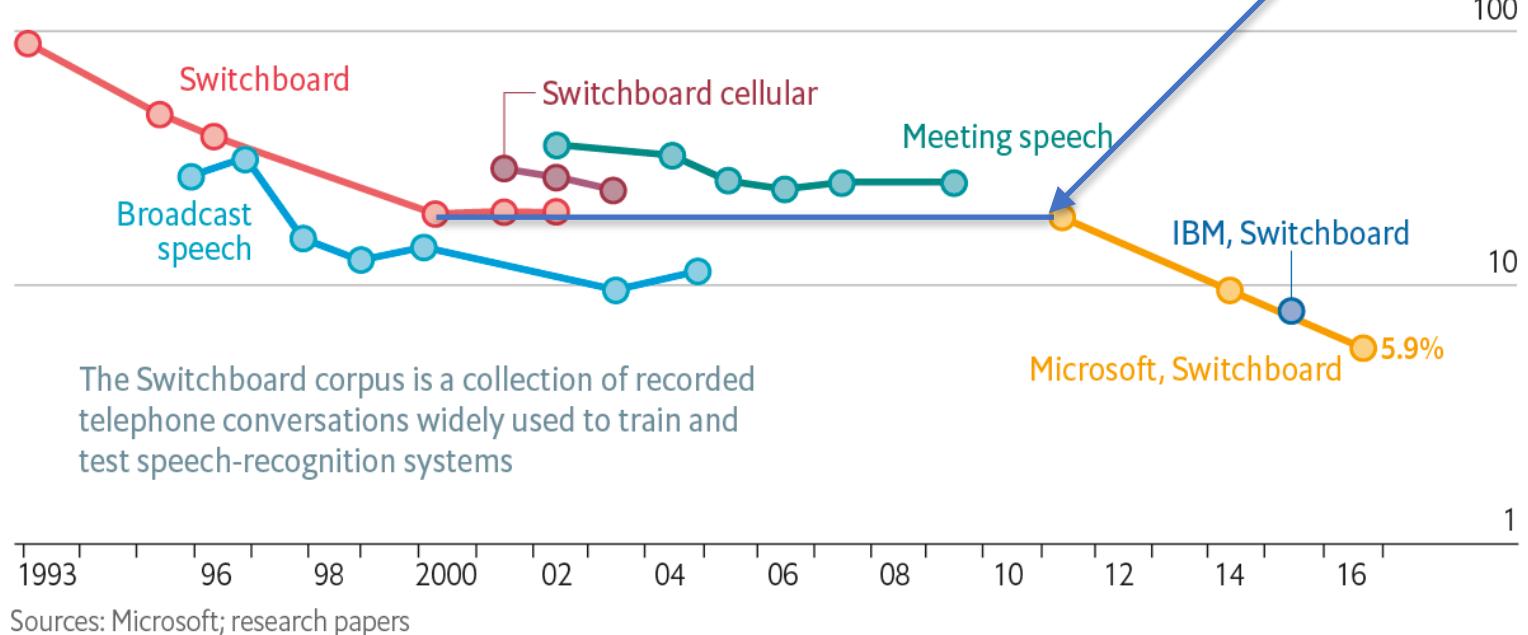
Разве что скорость вычислений за это время увеличилась в 1000 раз:
CPU 0.1x realtime в 1998, 1x в 2003г, 50x в 2011г, (500x на 6 ядрах в 2019г)

*примерные цифры для Sphinx на словаре в 20к слов, делить на 10 для 200к слов.

Этапы и качество 3

Loud and clear

Speech-recognition word-error rate, selected benchmarks, %



Ничего не происходило (1999 - 2011)

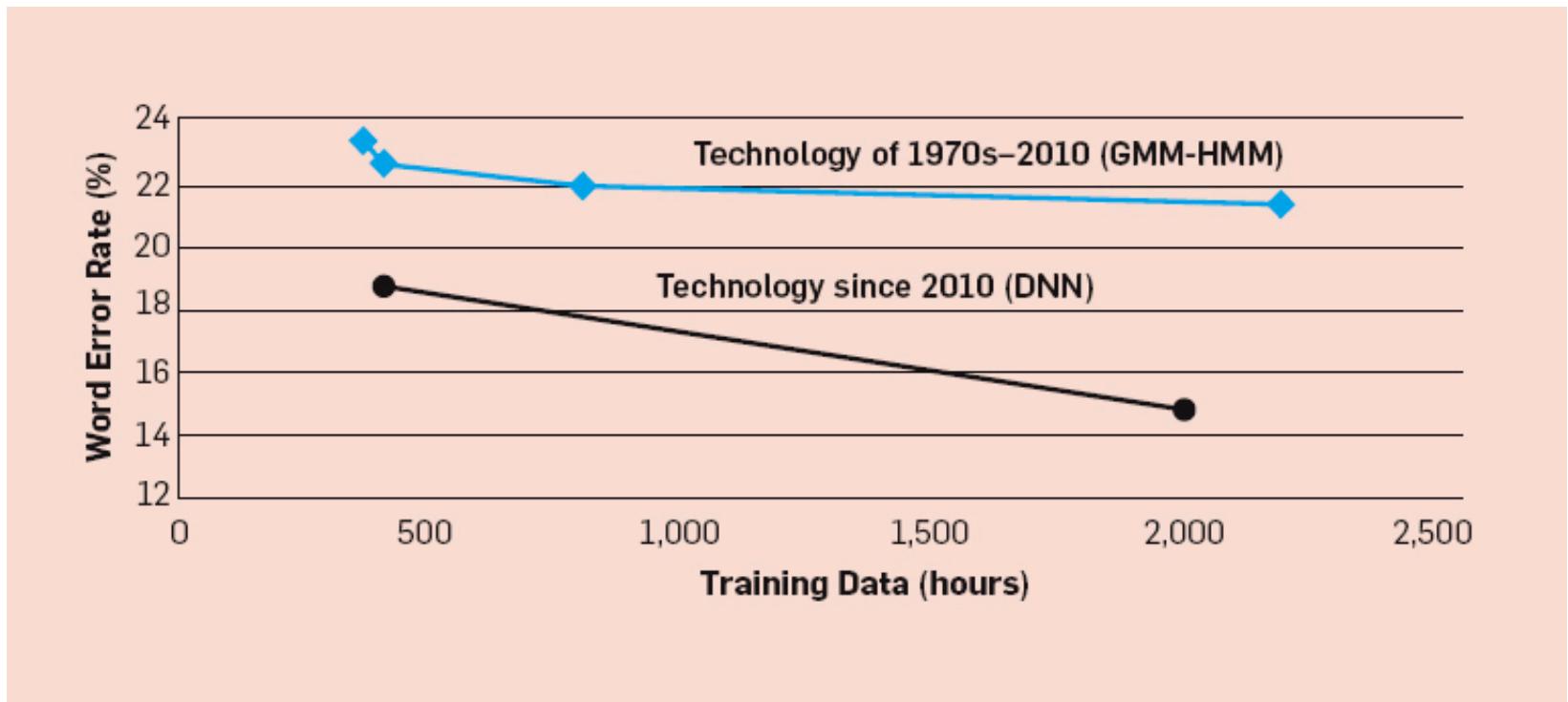
Разве что скорость вычислений за это время увеличилась в 1000 раз:
CPU 0.1x realtime в 1998, 1x в 2003г, 50x в 2011г, (500x на 6 ядрах в 2019г)

*примерные цифры для Sphinx на словаре в 20к слов, делить на 10 для 200к слов.

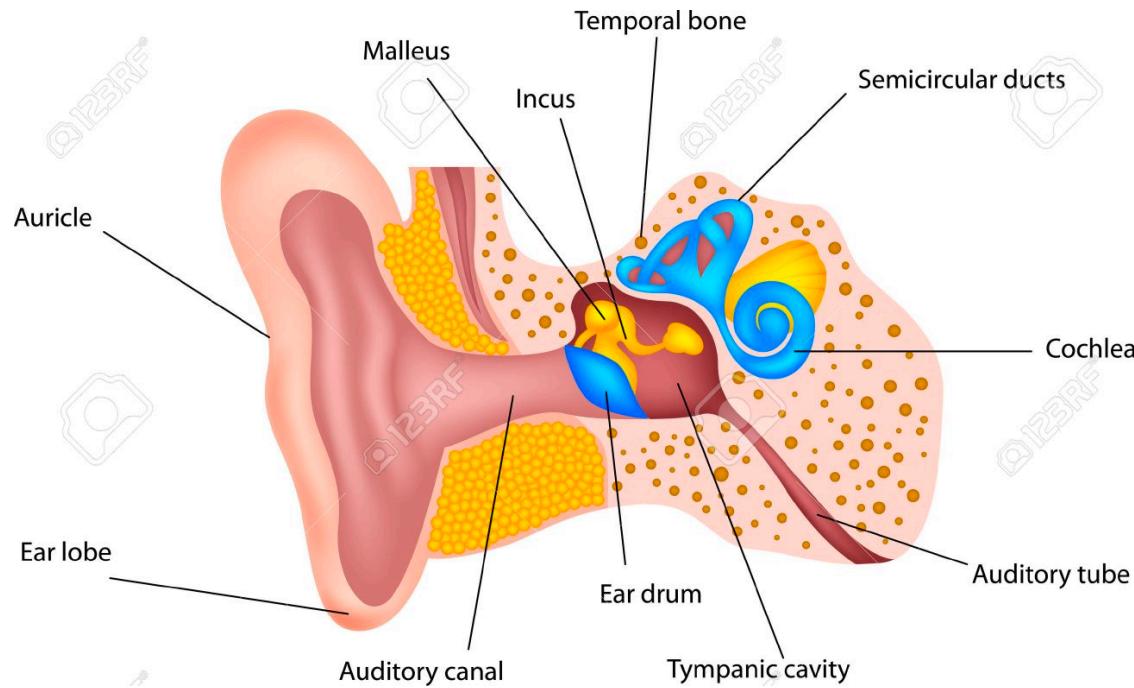
- Студент Geoffrey Hinton'a в 2011 году пришёл стажироваться в Microsoft и всё заверте...
- "ImageNet moment" в 2011 году, на год раньше AlexNet

Этапы и качество 4

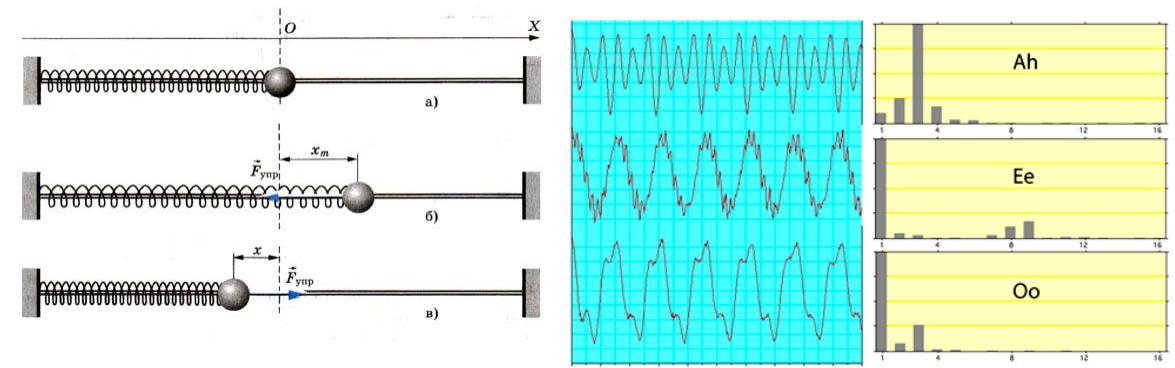
- Берём умный чёрный ящик, кормим данными, получаем хороший результат? Так просто?
- Давайте погрузимся поглубже в детали



Человеческое ухо

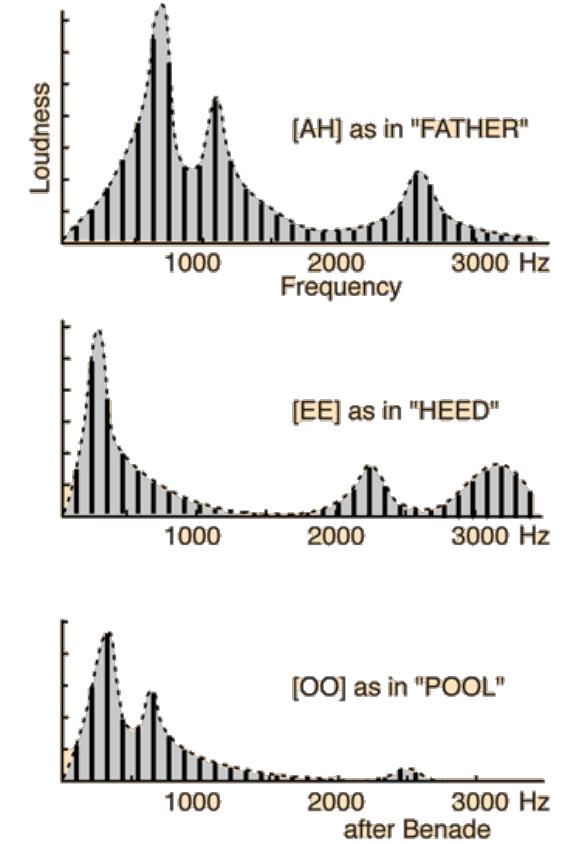
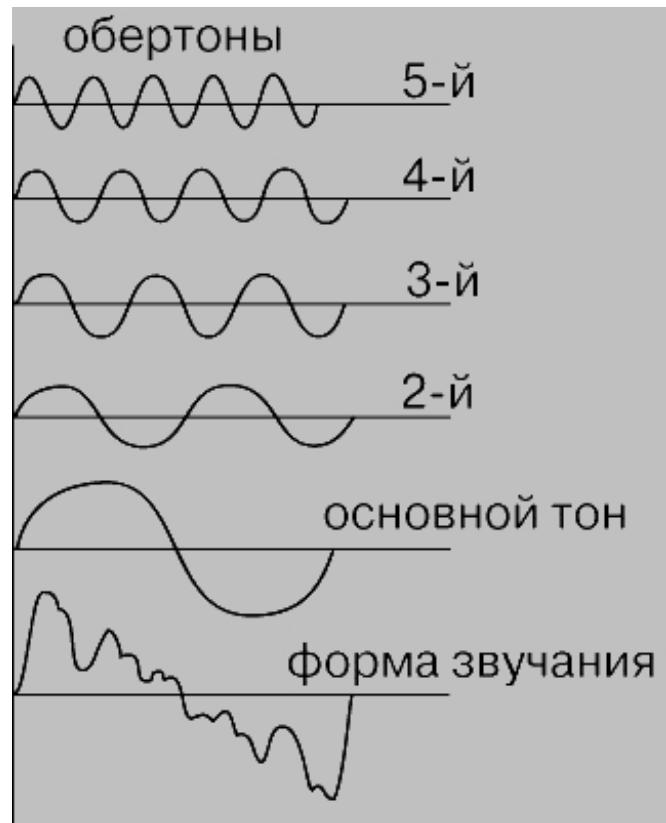


- Улитка == много резонаторов
- В улитке есть нервы, идущие к разным нейронам
- Разные места улитки колеблются с разной частотой, зависимой от входного сигнала, передавая сигнал на нервы

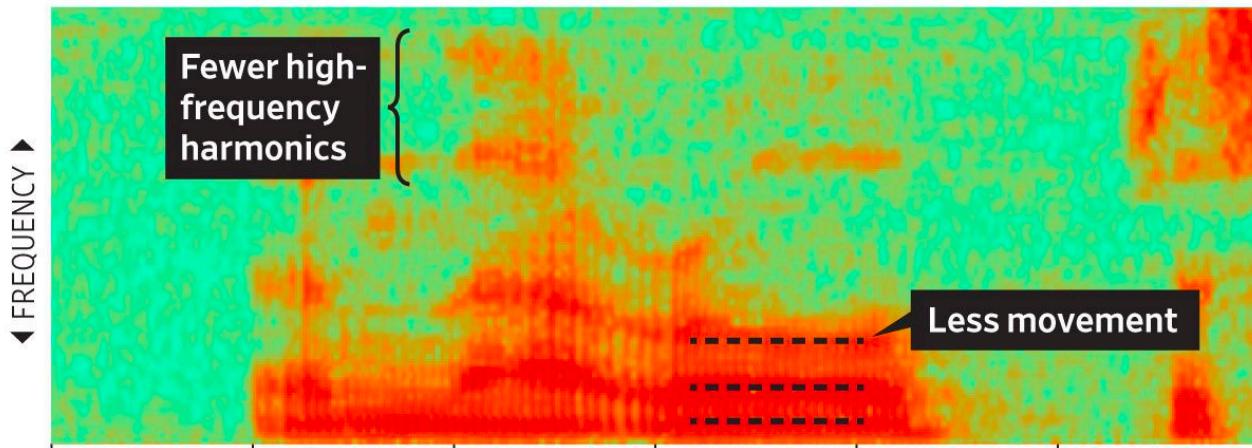
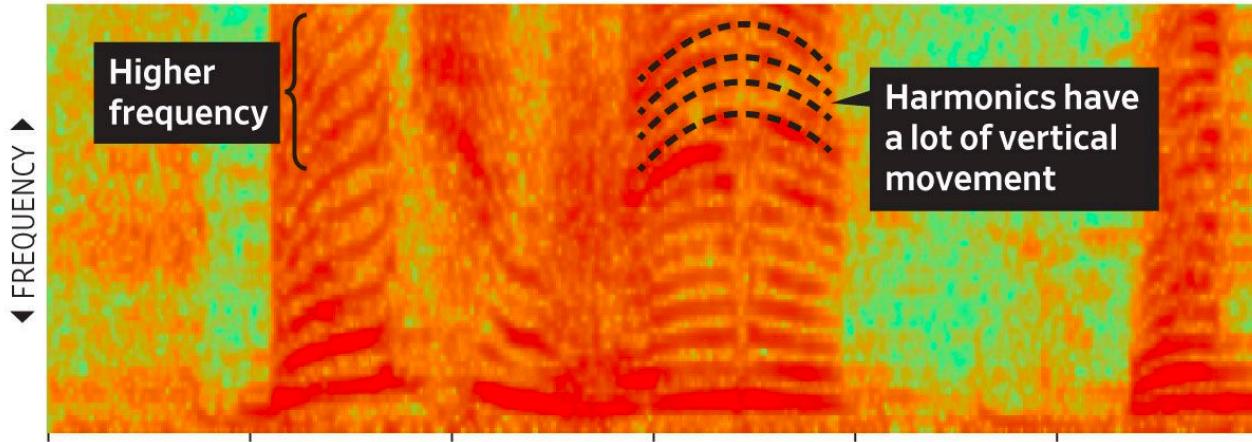
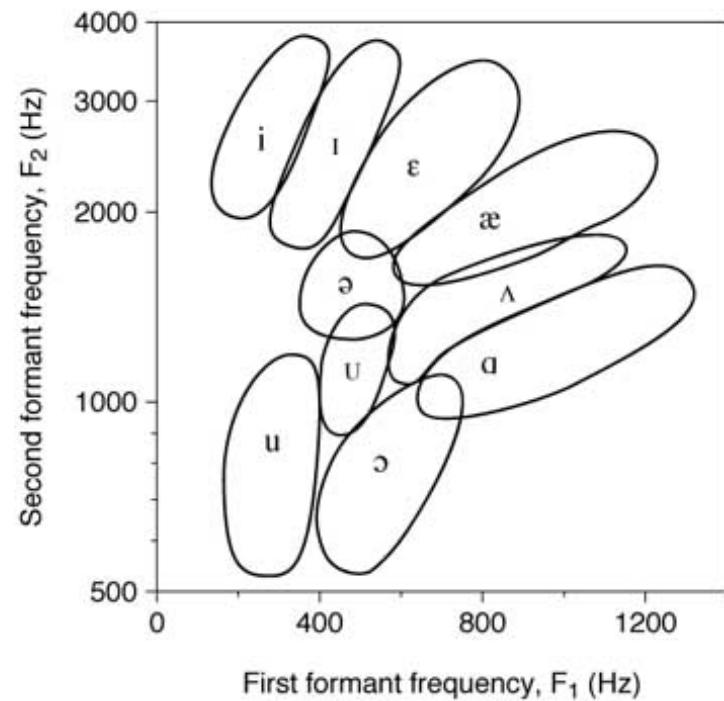


Fast Fourier Transform (FFT) оно же STFT (Short-Term FT)

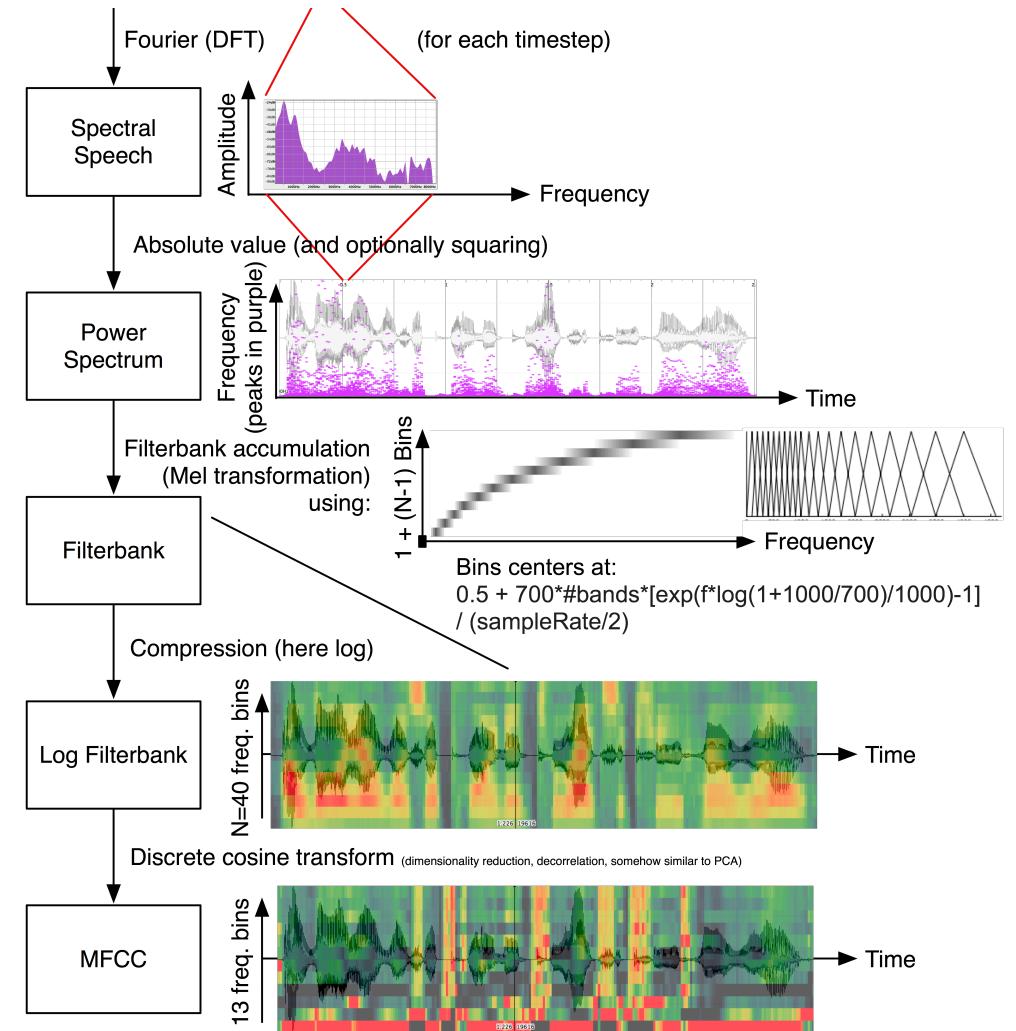
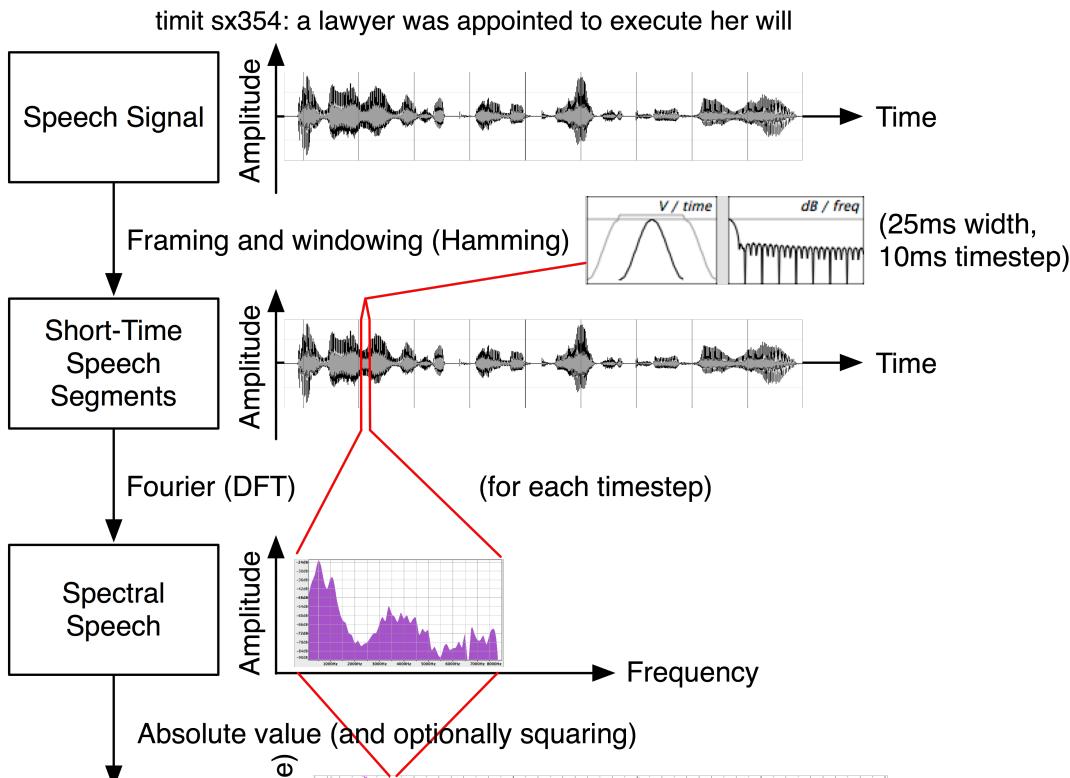
- Частота колебаний связок -- основная частота голоса, "высота тона" (60-400 Гц)
- Преобразования голосового тракта обеспечивают обертона
- STFT намного проще понять, чем исходный звуковой сигнал!



Просто о сложном: Параметры звука



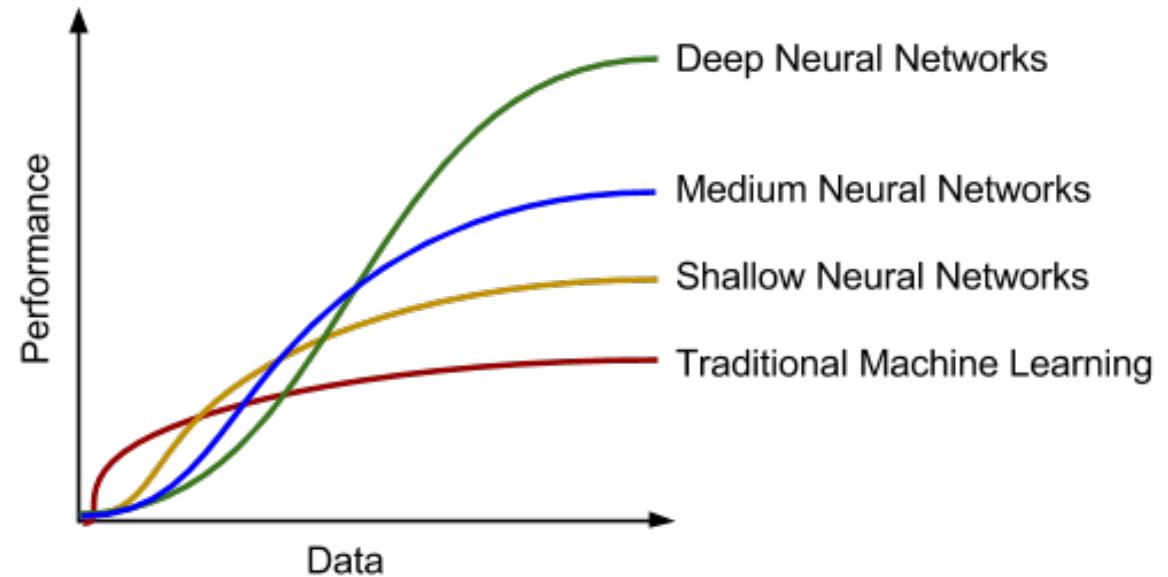
Предобработка речи



[Sample preprocessing pipeline in numpy + librosa](#)

Два пути Speech Recognition

- Речь -> ... -> Фонемы -> Словарь
 - TIMIT Database (en) - 3h, 1993
 - CMUDict (en) - 100k, 1993
- Разбили сложную задачу на две более простых... но они требуют разметки фонем, составления словаря и алгоритма поиска
- Слово "IPA"= [ai p^hi: ei]



[Open speech corpora list](#)

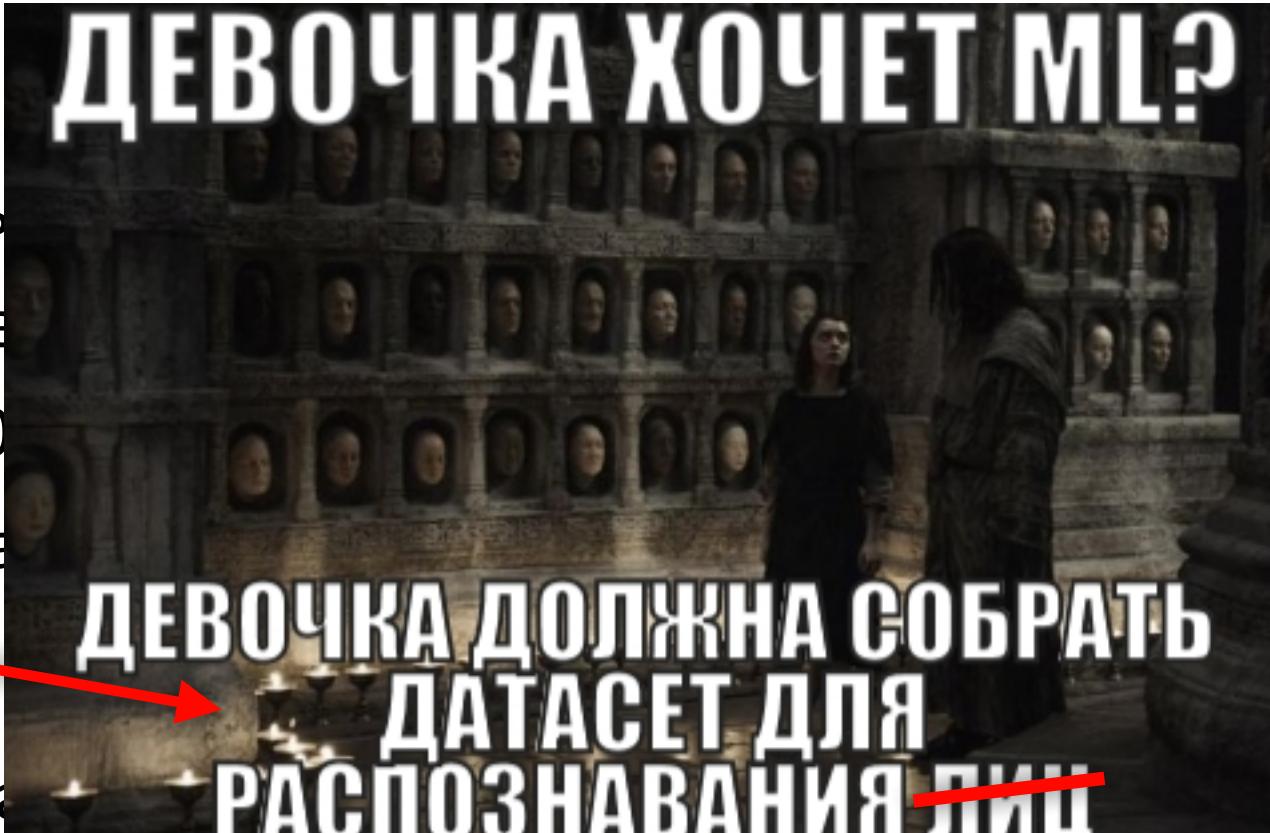
Два пути Speech Recognition

- Речь -> ... -> Фонемы -> Словарь
 - TIMIT Database (en) - 3h, 1993
 - CMUDict (en) - 100k, 1993
- Разбили сложную задачу на две более простых... но они требуют разметки фонем, составления словаря и алгоритма поиска
- Слово "IPA"= [ai p^hi: ei]
- Речь -> ... -> Буквы -> Текст
- Открытые датасеты:
 - WSJ (en) - 81h, 1993
 - Switchboard (en) - 240h, 1993
 - VoxForge (ru) - 17h, 2009

[Open speech corpora list](#)

Два пути Speech Recognition

- Речь -> ... -> Фонемы
 - TIMIT Database (en) - 1000 speakers, 100 hours, 1993
 - CMUDict (en) - 1000 speakers, 31h, 1993
- Разбили сложную задачу распознавания лиц на две части: более **огромный** объем данных для обучения алгоритма, разметки фонем, создание **датасета** для **распознавания лиц**, а также создание **словаря** и алгоритма для **реконструкции** речи.
- Слово "IPA"= [ai p^{tʃ} 1: ei]



[Open speech corpora list](#)

Два пути Speech Recognition

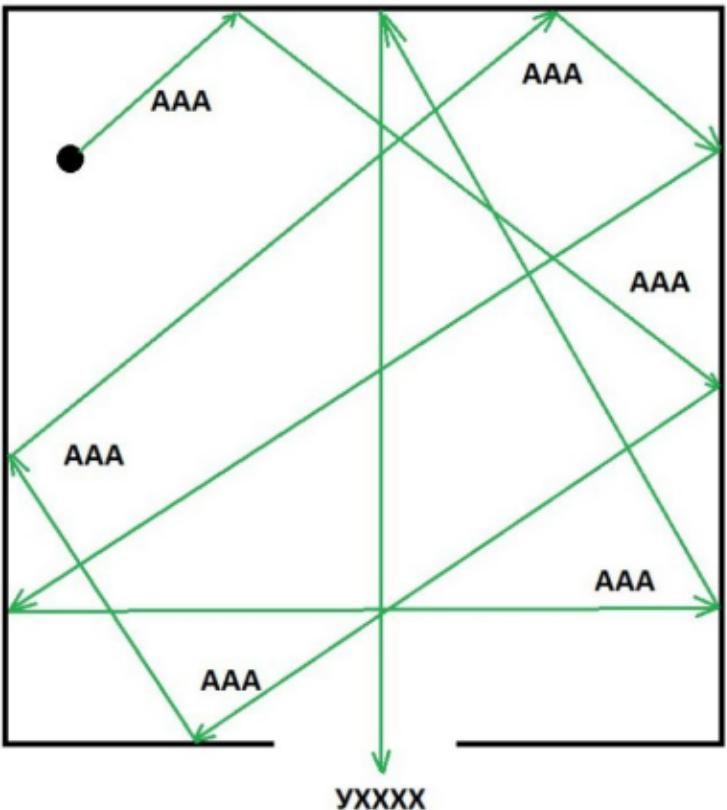
- Речь -> ... -> Фонемы -> Слова
 - TIMIT Database (en) - 3h
 - CMUDict (en) - 100k, 1993
 - Разбили сложную задачу на более простых... но они требуют разметки фонем, составление словаря и алгоритма поиска
 - Слово "IPA"= [ai p^h i:]
-

квы -> Текст
асеты:
(en) - 81h, 1993
д (en) - 240h, 1993
(ru) - 17h, 2009

[Open speech corpora list](#)

Два пути Speech Recognition

● вы находитесь здесь

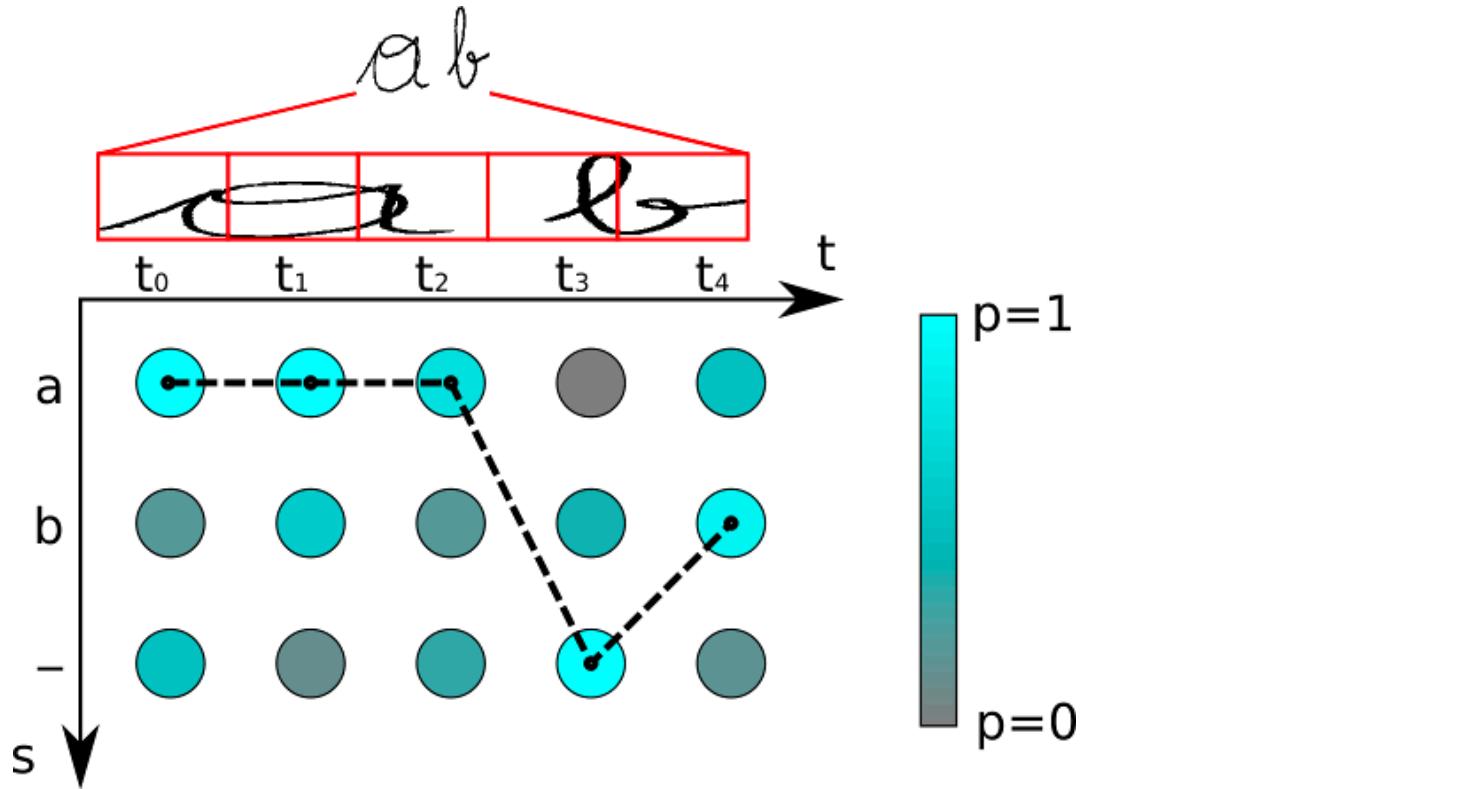
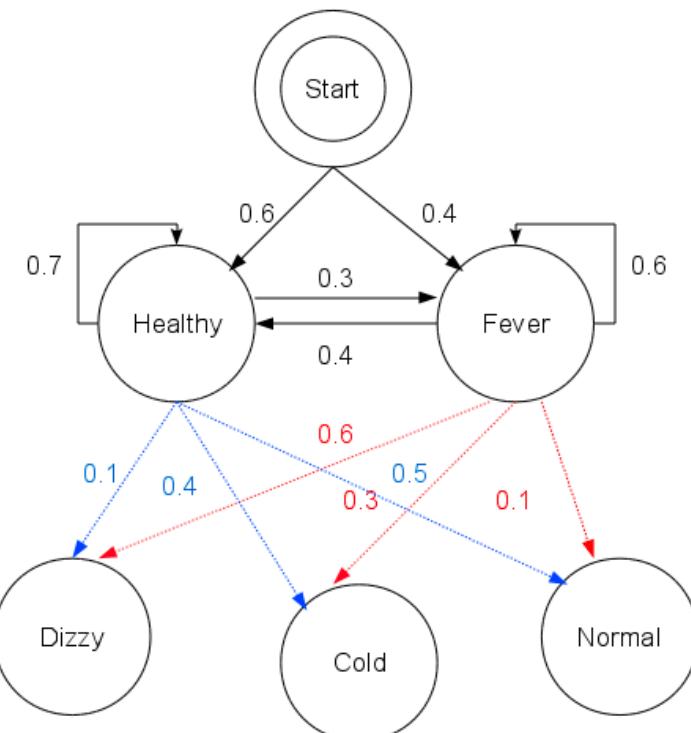


- Речь -> ... -> Буквы -> Текст
- Открытые датасеты:
 - WSJ (en) - 81h, 1993
 - Switchboard (en) - 240h, 1993
 - VoxForge (ru) - 17h, 2009
 - Fisher (en) - 2000h, 2004
 - LibriSpeech (en) - 960h, 2015
 - Open_TTS (ru) - 3000h+, 2019

[Open speech corpora list](#)

Авторазметка (Alignment)

Проблема: не хотим размечать позицию каждой буквы в тексте, хотим что-то такое:



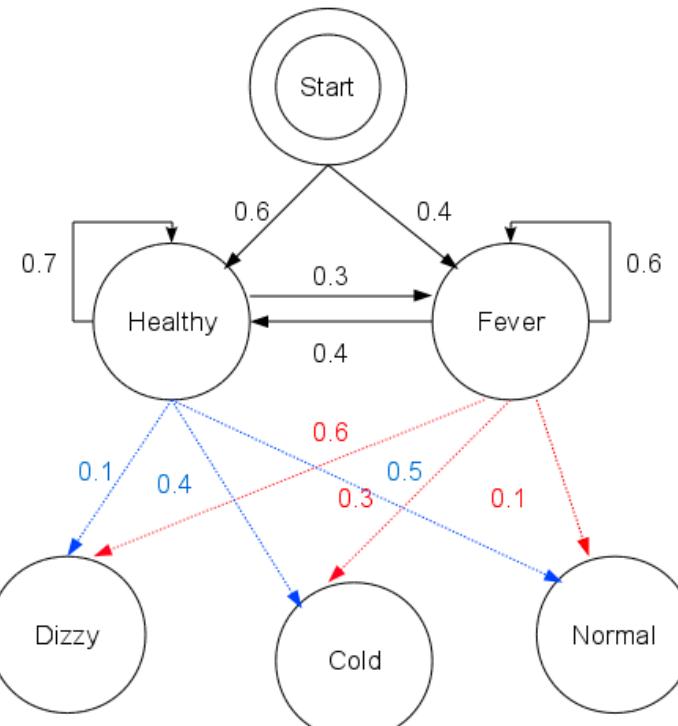
HMM -- один из способов составить таблицу переходов, позволяет считать вероятность фразы
 $P("Weather is cold" | "Audio: Wetheriscold") = ?$

Другой алгоритм -- Forward-backward algorithm

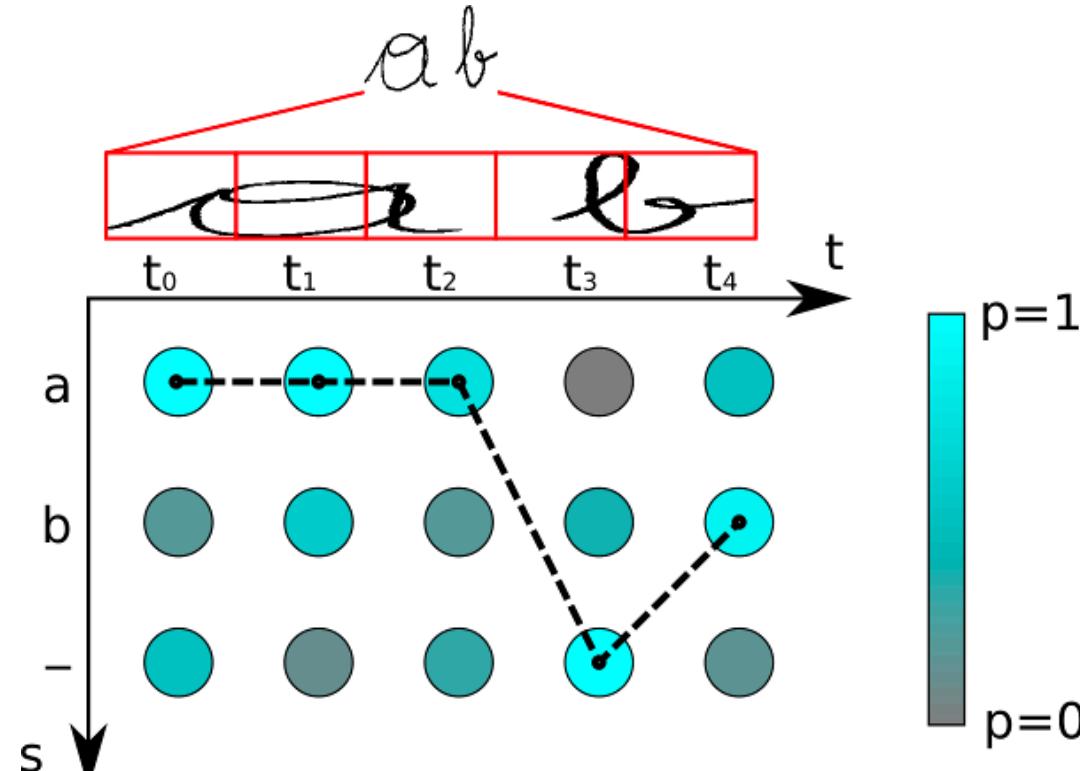
https://en.wikipedia.org/wiki/Viterbi_algorithm
https://en.wikipedia.org/wiki/Forward–backward_algorithm

Авторазметка (Alignment)

Проблема: не хотим размечать позицию каждой буквы в тексте,
хотим что-то такое:



HMM -- один из способов составить таблицу переходов,
Doesn't scale to >4-char-grams!
 $P(\text{Weather is cold} \mid \text{Audio. weatheriscold}) = ?$



Другой алгоритм -- Forward-backward algorithm

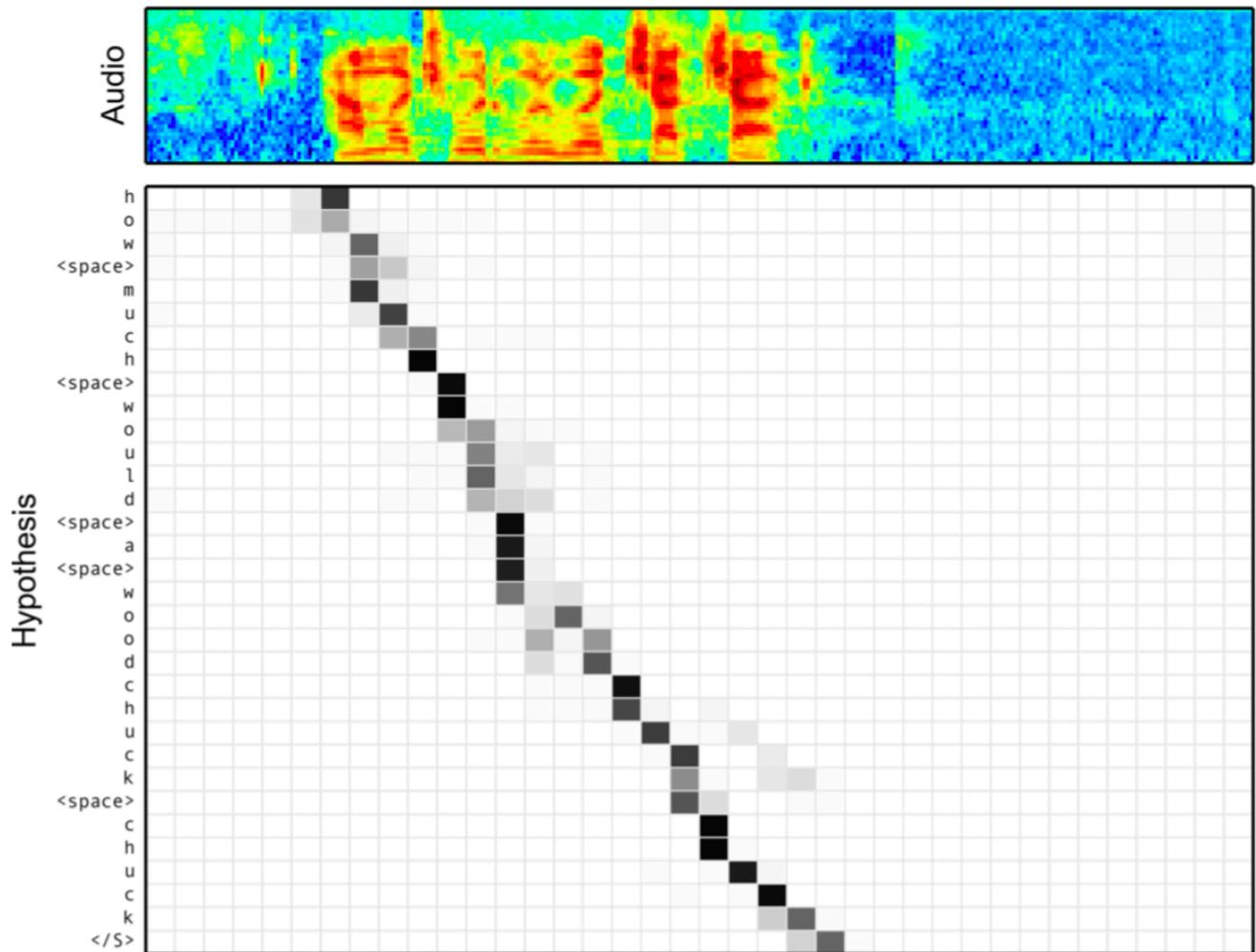
https://en.wikipedia.org/wiki/Viterbi_algorithm

https://en.wikipedia.org/wiki/Forward–backward_algorithm

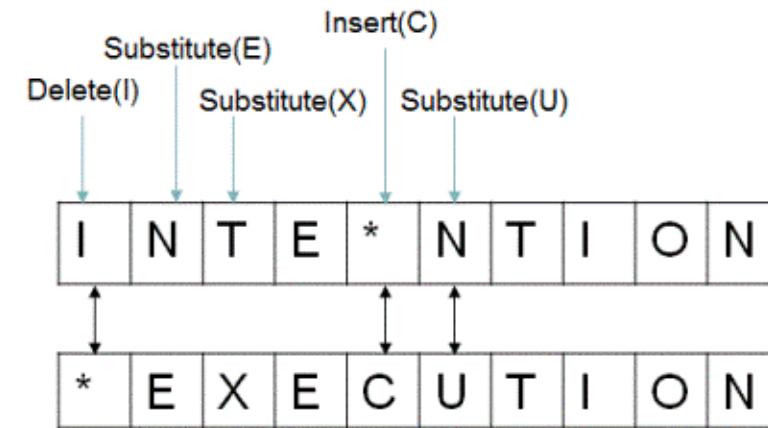
Авторазметка (Alignment)

- Каждой возможной букве или звуку сопоставляется вероятность, а потом нужно построить путь по матрице вероятностей, минимизирующий некоторую метрику...

Alignment between the Characters and Audio



CTC (Connectionists Temporal Classification)

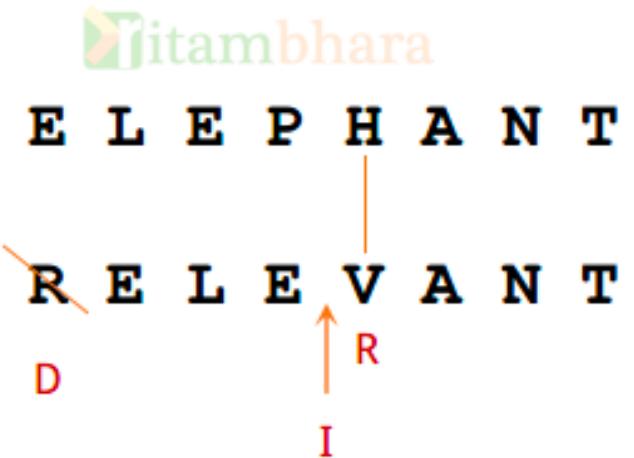


Эта метрика называется Edit Distance,
она же расстояние Левенштейна
(Если штрафы не 1, то другие названия)

Теперь будем штрафовать нашу нейросеть за
несоответствия в тех местах, где произошли ошибки!

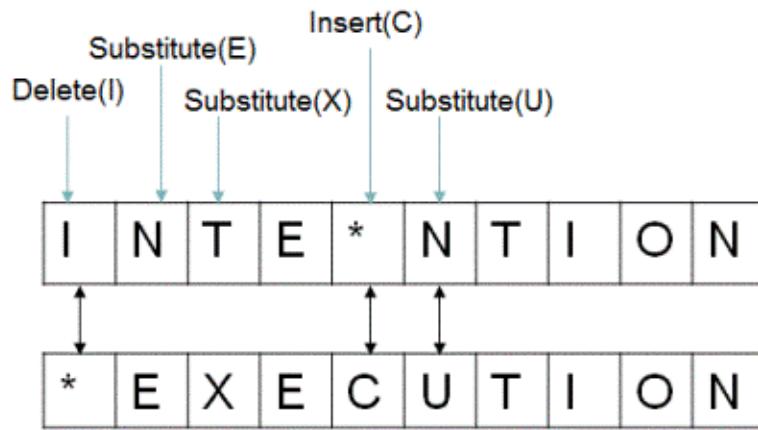
		E	L	E	P	H	A	N	T
R	0	1	2	3	4	5	6	7	8
E	1	1	2	3	4	5	6	7	8
L	2	1	2	2	3	4	5	6	7
E	3	2	1	2	3	4	5	6	7
V	4	3	2	1	2	3	4	5	6
A	5	4	3	2	2	3	4	5	6
N	6	5	4	3	3	3	4	5	6
T	7	6	5	4	4	4	4	3	4
	8	7	6	5	5	5	5	4	3

titambhara



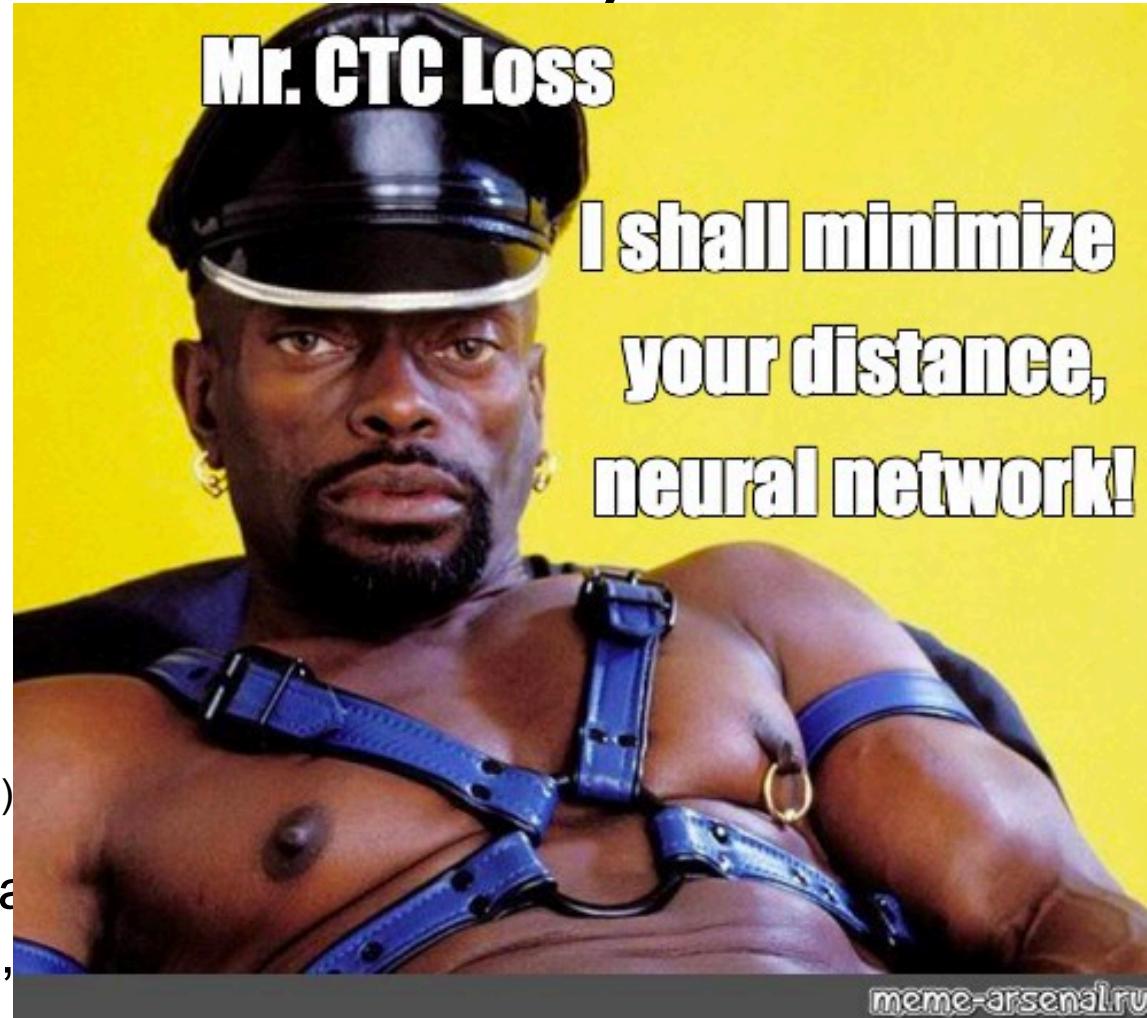
Min Edit Distance: 3

CTC (Connectionists Temporal Classification)

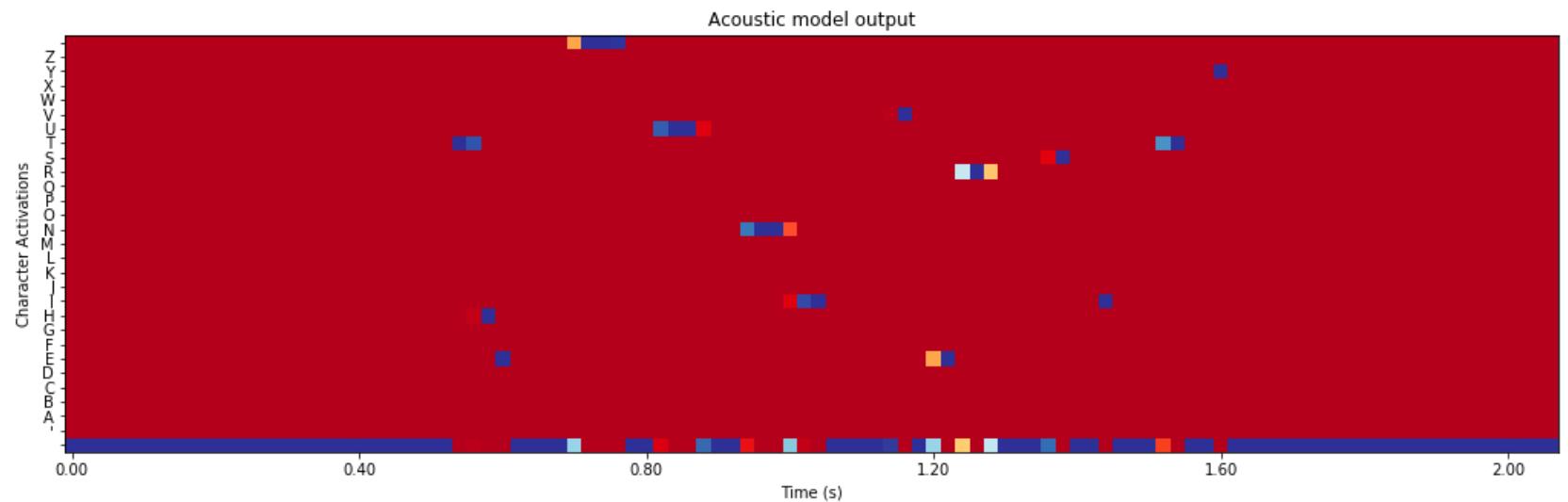
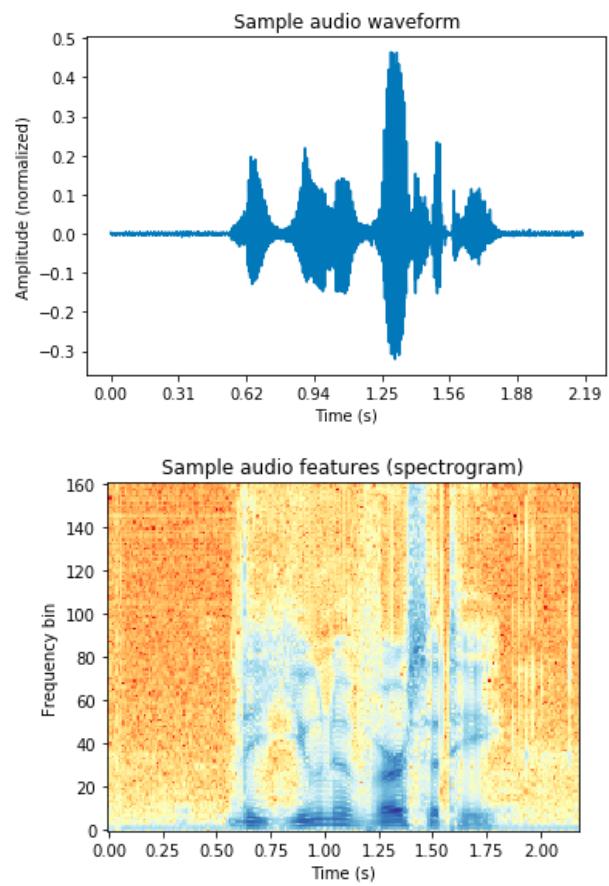


Эта метрика называется Edit Distance,
она же расстояние Левенштейна
(Если штрафы не 1, то другие названия)

Теперь будем штрафовать на
несоответствия в тех местах,



Посмотрим, как это всё работает



Error:

IRR

Decoder: THE UNIVERSITY

Correct: THEIR UNIVERSITIES

Пособие молодого бога: Ваши первые шаги для создания мира

- 1. Создать мир, животных, людей, дать людям разум
- 2. Продумать систему грехов и метрику качества жизни для попадания в рай
- 3. Запустить систему!
- 4. Отдыхать

~~Пособие молодого бøга:~~

Ваши первые шаги для создания нейросети

- 1. Создать датасет, нейросеть, выбрать гиперпараметры
- 2. Продумать штрафы (Loss) и метрики (CER, WER)
- 3. Запустить систему!
- 4. Отдыхать

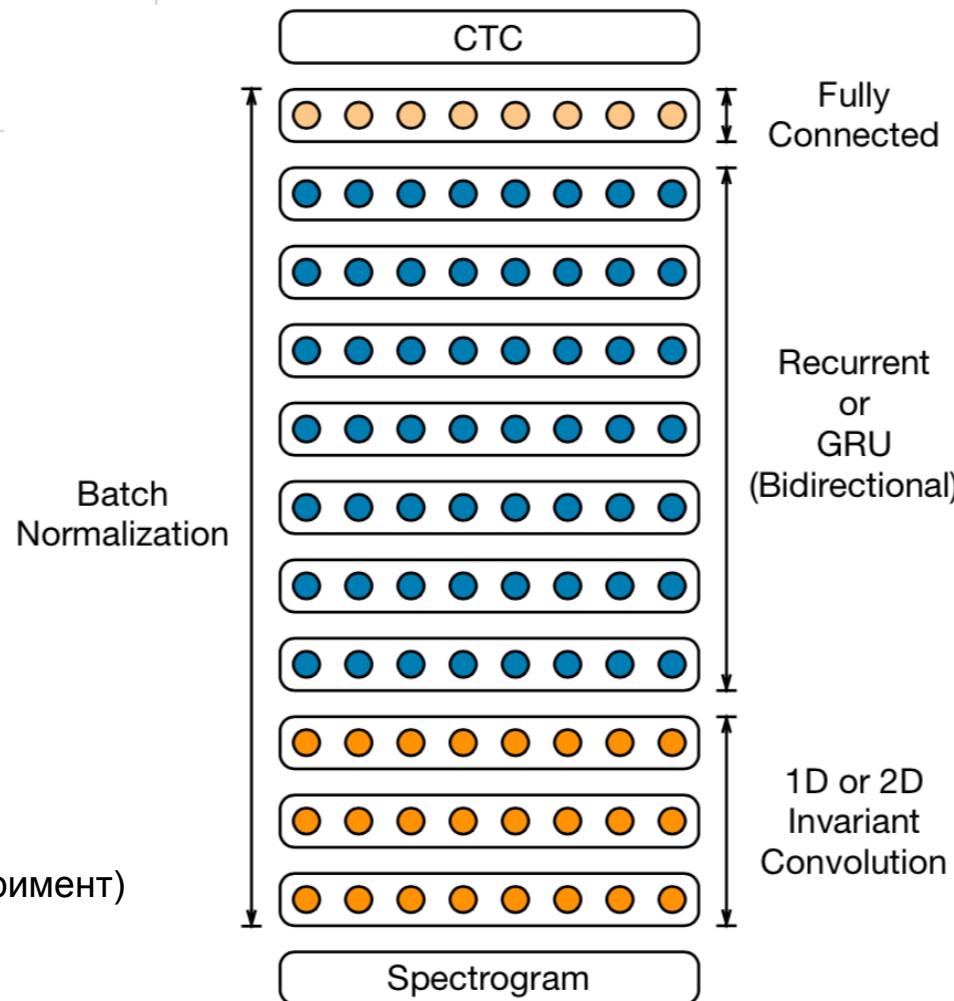
~~Пособие молодого бøга:~~

Ваши первые шаги для создания нейросети

- 1. Создать датасет, нейросеть, выбрать гиперпараметры
 - 2. Продумать штрафы (Loss) и метрики (CER, WER)
 - 3. Запустить систему!
 - 4. Отдыхать
-
-
-
-
- P.S. И хватит уже слать этот баян про бога!

Выбираем архитектуру

- Пробуют:
- 1-9 рекуррентных слоя (RNN, GRU, LSTM)
- 1-3 входных Conv слоя
- 2 дня на 8 GPU!
(На каждый эксперимент)



Deep Speech 2: End-to-End Speech Recognition in English and Mandarin

Baidu Research – Silicon Valley AI Lab*
Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro,
Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel,
Linxi Fan, Christopher Fougner, Tony Han, Awni Hannun, Billy Jun, Patrick LeGresley,
Libby Lin, Sharan Narang, Andrew Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman,
Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang,
Bo Xiao, Dani Yogatama, Jun Zhan, Zhenyao Zhu

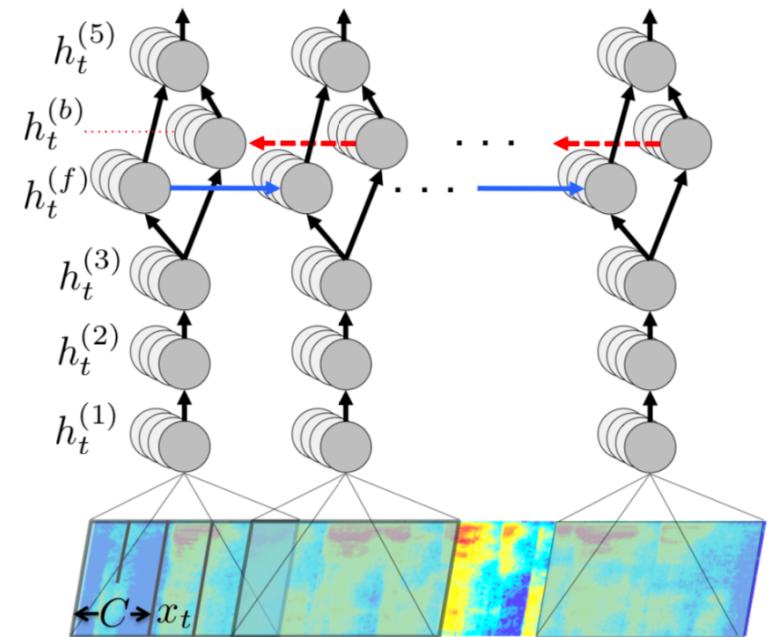


Figure 1: Structure of our RNN model and notation.

<https://arxiv.org/abs/1512.02595>

Выбираем архитектуру

Deep Speech 2: End-to-End Speech Recognition in English and Mandarin

- Лучшая сеть:
- 2x 2D-invariant convolution
- 7x recurrent layers
- (RNN-1280)
- 68M parameters

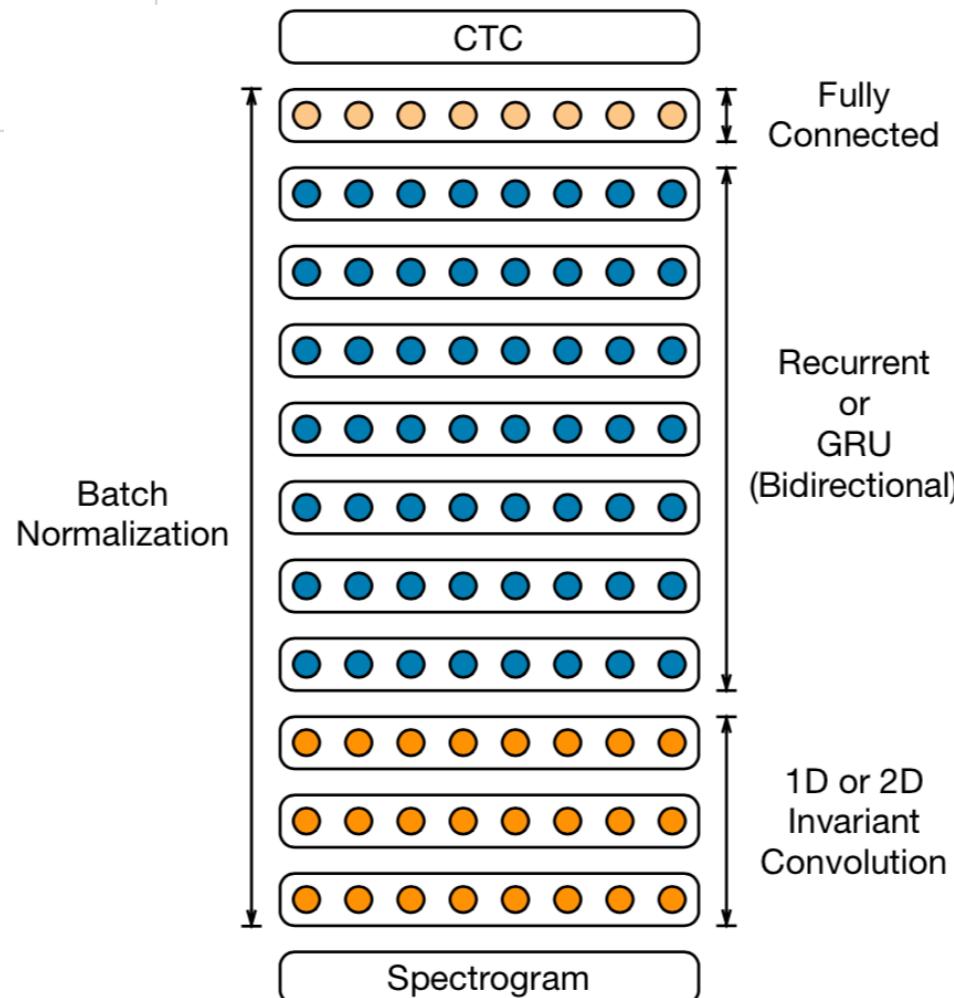
Architecture	Hidden Units		Train		Dev	
			Baseline	BatchNorm	Baseline	BatchNorm
1 RNN, 5 total	2400	10.55		11.99	13.55	14.40
3 RNN, 5 total	1880	9.55		8.29	11.61	10.56
5 RNN, 7 total	1510	8.59		7.61	10.77	9.78
7 RNN, 9 total	1280	8.76		7.68	10.83	9.52

Architecture	Simple RNN	GRU	Model size	Model type	Regular Dev	Noisy Dev
5 layers, 1 Recurrent	14.40	10.53	18×10^6	GRU	10.59	21.38
5 layers, 3 Recurrent	10.56	8.00	38×10^6	GRU	9.06	17.07
7 layers, 5 Recurrent	9.78	7.79	70×10^6	GRU	8.54	15.98
9 layers, 7 Recurrent	9.52	8.19	70×10^6	RNN	8.44	15.09
			100×10^6	GRU	7.78	14.17
			100×10^6	RNN	7.73	13.06

<https://arxiv.org/abs/1512.02595>

Выбираем архитектуру

- Лучшая сеть:
- 2x 2D-invariant convolution
- 7x recurrent layers
- (RNN-1280)
- 68M parameters
- 2 дня на 8 GPU!



Deep Speech 2: End-to-End Speech Recognition in English and Mandarin

Baidu Research – Silicon Valley AI Lab*
Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel, Linxi Fan, Christopher Fougner, Tony Han, Awni Hannun, Billy Jun, Patrick LeGresley, Libby Lin, Sharan Narang, Andrew Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang, Bo Xiao, Dani Yogatama, Jun Zhan, Zhenyao Zhu

Dataset	Speech Type	Hours
WSJ	read	80
Switchboard	conversational	300
Fisher	conversational	2000
LibriSpeech	read	960
Baidu	read	5000
Baidu	mixed	3600
Total		11940

Fraction of Data	Hours	Regular Dev	Noisy Dev
1%	120	29.23	50.97
10%	1200	13.80	22.99
20%	2400	11.65	20.41
50%	6000	9.51	15.90
100%	12000	8.46	13.59

Выбираем архитектуру

- Лучшая сеть:
- 2x 2D-invariant convolution
- 7x recurrent layers
- (RNN-1280)
- 68M parameters
- 2 дня на 8 GPU!

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

Noisy Speech			
Test set	DS1	DS2	Human
CHiME eval clean	6.30	3.34	3.46
CHiME eval real	67.94	21.79	11.84
CHiME eval sim	80.27	45.05	31.33

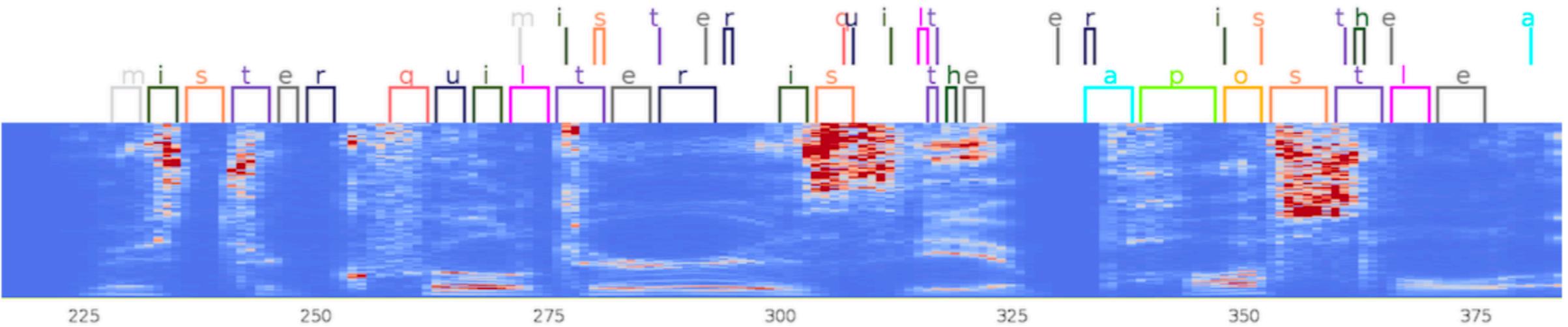
Deep Speech 2: End-to-End Speech Recognition in English and Mandarin

Baidu Research – Silicon Valley AI Lab*
Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro,
Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel,
Linxi Fan, Christopher Fougner, Tony Han, Awni Hannun, Billy Jun, Patrick LeGresley,
Libby Lin, Sharan Narang, Andrew Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman,
Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang,
Bo Xiao, Dani Yogatama, Jun Zhan, Zhenyao Zhu

Dataset	Speech Type	Hours
WSJ	read	80
Switchboard	conversational	300
Fisher	conversational	2000
LibriSpeech	read	960
Baidu	read	5000
Baidu	mixed	3600
Total		11940

Fraction of Data	Hours	Regular Dev	Noisy Dev
1%	120	29.23	50.97
10%	1200	13.80	22.99
20%	2400	11.65	20.41
50%	6000	9.51	15.90
100%	12000	8.46	13.59

СТС: предложение посложнее



- СТС плохо учится, когда несовпадений много
- Поэтому используется "Curriculum Learning":
 - нейросеть сначала учится на более простых примерах, а потом переходит на более сложные

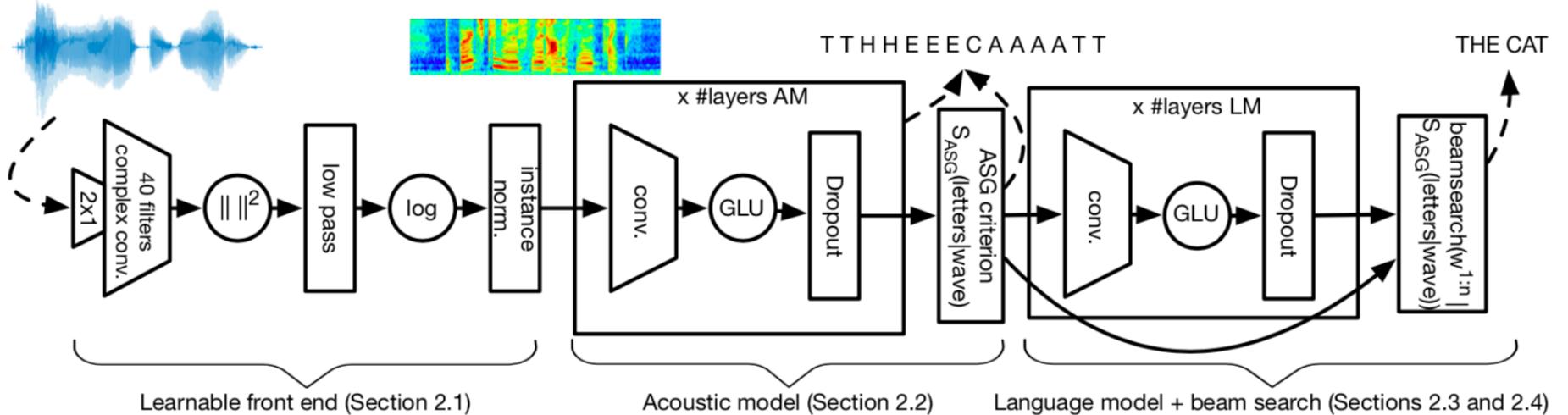
Выбираем архитектуру (2)

Fully Convolutional Speech Recognition

Neil Zeghidour^{1,2,*}, Qiantong Xu^{1,*}, Vitaliy Liptchinsky¹, Nicolas Usunier¹,
Gabriel Synnaeve¹, Ronan Collobert¹

¹ Facebook A.I. Research, Paris, France; New York & Menlo Park, USA

² CoML, ENS/CNRS/EHESS/INRIA/PSL Research University, Paris, France



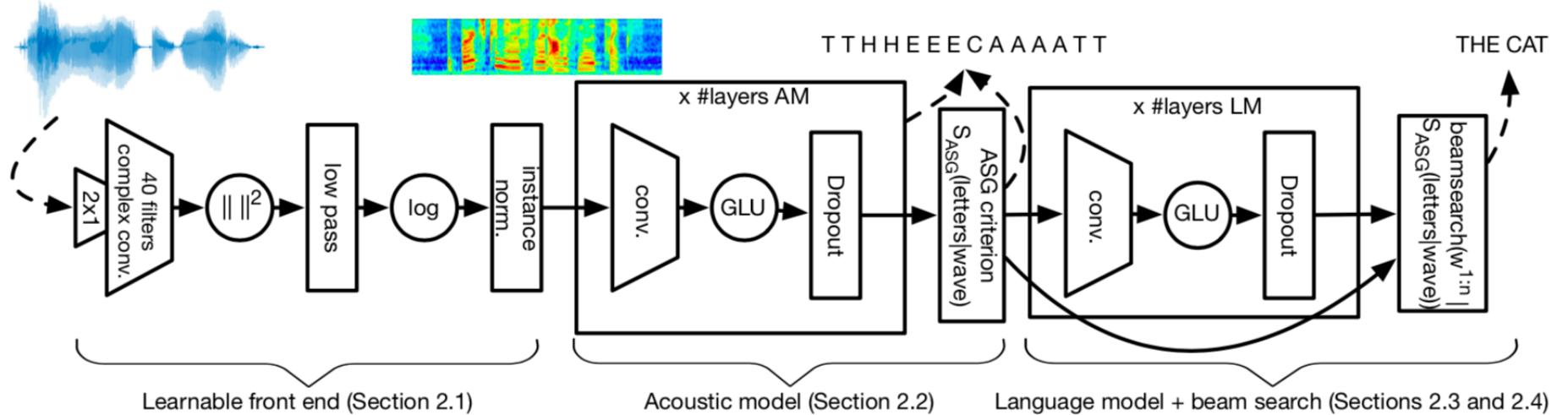
Выбираем архитектуру (2)

Fully Convolutional Speech Recognition

Neil Zeghidour^{1,2,*}, Qiantong Xu^{1,*}, Vitaliy Liptchinsky¹, Nicolas Usunier¹,
Gabriel Synnaeve¹, Ronan Collobert¹

¹ Facebook A.I. Research, Paris, France; New York & Menlo Park, USA

² CoML, ENS/CNRS/EHESS/INRIA/PSL Research University, Paris, France



Dataset	Architecture	#conv.	dropout first/last layer	#hu first/last layer	kw first/last layer	#hu full connect
WSJ	Low Dropout	17	0.25/0.25	100/375	3/21	1000
LibriSpeech	Low Dropout	17	0.25/0.25	200/750	13/27	1500
	High Dropout	19	0.20/0.60	200/1000	13/29	2000

Выбираем архитектуру (2)

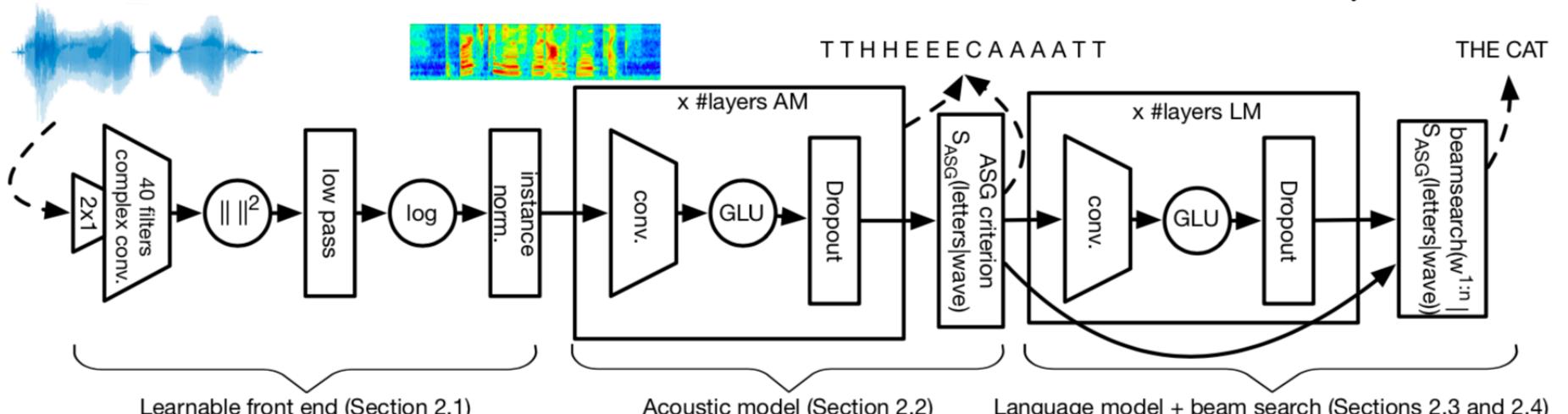
Fully Convolutional Speech Recognition

Neil Zeghidour^{1,2,*}, Qiantong Xu^{1,*}, Vitaliy Liptchinsky¹, Nicolas Usunier¹,
Gabriel Synnaeve¹, Ronan Collobert¹

¹ Facebook A.I. Research, Paris, France; New York & Menlo Park, USA

² CoML, ENS/CNRS/EHESS/INRIA/PSL Research University, Paris, France

- Что такое GLU?
- Зачем нужен Learnable front end?
- Что такое Language model?



Dataset	Architecture	#conv.	dropout first/last layer	#hu first/last layer	kw first/last layer	#hu full connect
WSJ	Low Dropout	17	0.25/0.25	100/375	3/21	1000
LibriSpeech	Low Dropout	17	0.25/0.25	200/750	13/27	1500
	High Dropout	19	0.20/0.60	200/1000	13/29	2000

Learnable frontend

- Можем учиться даже на исходном звуке, без STFT
- Входные преобразования тоже можно заменять нейросетями!

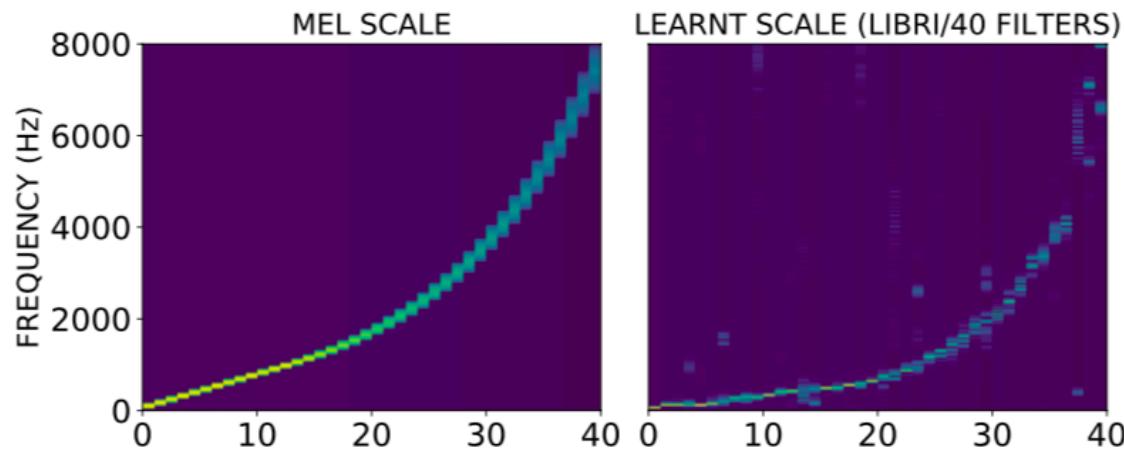


Figure 4: Power heatmap of the 40 mel-filters (left) and of the frequency response of the 40 convolutional filters learned from the raw waveform on Librispeech (right).

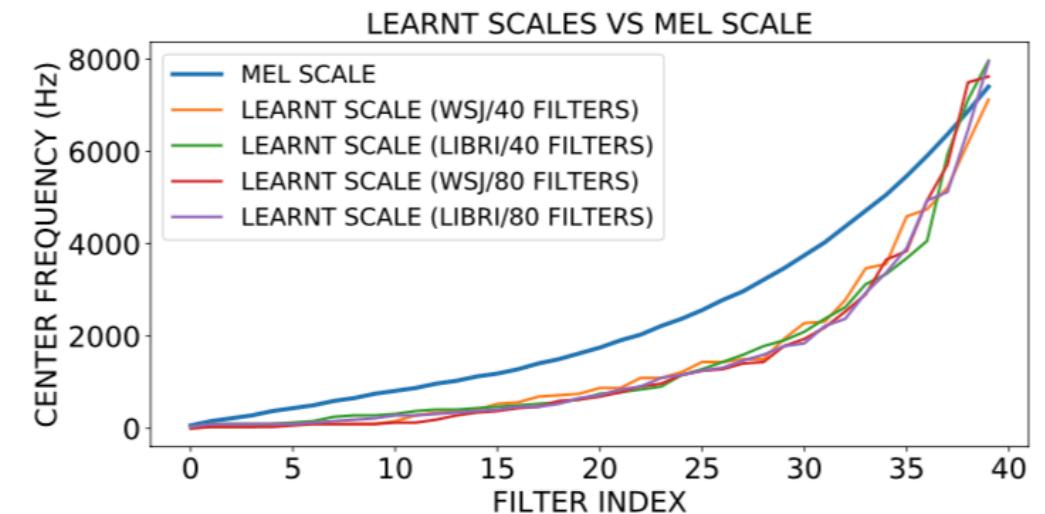
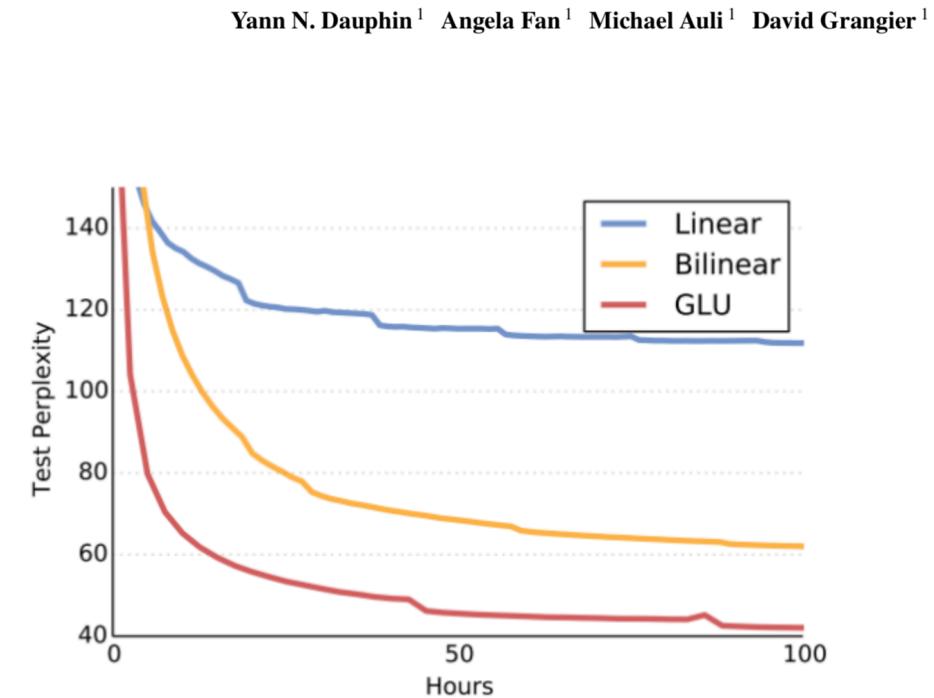
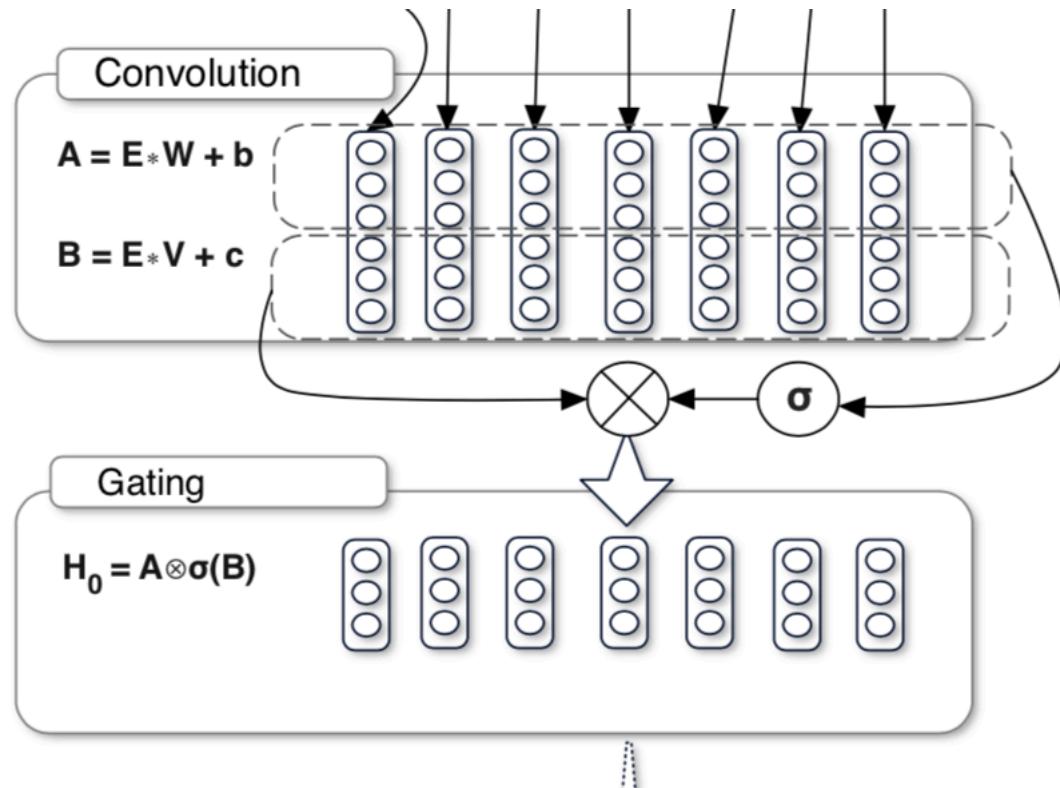


Figure 3: Center frequency of the front-end filters, for the mel-filterbank baseline and the learnable front-ends.

Conv + GLU -- на замену ResNet

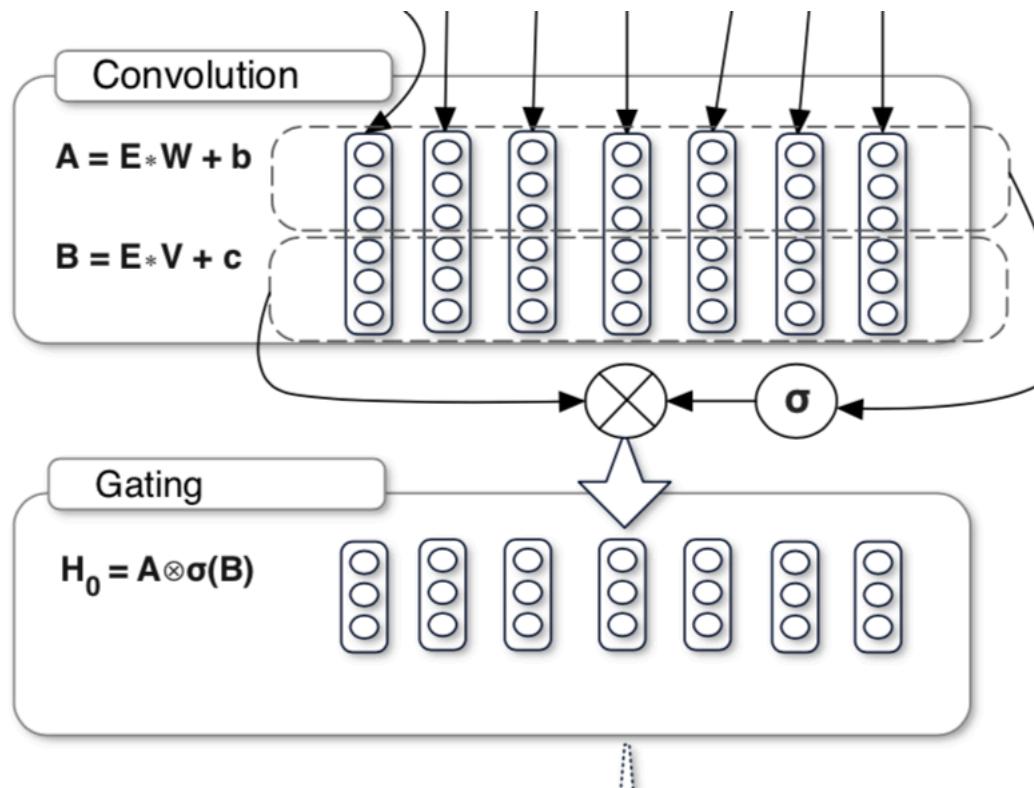
Language Modeling with Gated Convolutional Networks



<https://arxiv.org/abs/1612.08083>

<https://arxiv.org/abs/1812.06864>

Conv + GLU -- на замену ResNet



Language Modeling with Gated Convolutional Networks



<https://arxiv.org/abs/1612.08083>

<https://arxiv.org/abs/1812.06864>

Доктор, помогите! Моя нейросеть плохо работает...

RNN output	Decoded Transcription
what is the weather like in bostin right now prime miniter nerenr modi arther n tickets for the game	what is the weather like in boston right now prime minister narendra modi are there any tickets for the game

АНО НСТДРУКЦИЮ ПО ПОИСКУ НЕСПРАВНОСТЬИ
ГЛЮБОМОХЛОЖНОМ ТИХНУЬ И ХКУМАСПРООСТВЕНАМБИЛЬ

ТЕЛЕИЗРИТОД

ИПРВЕДЕН ПЫРИЧИН НЕЖНХ ПЗНАКУ НЕГИТ АМПОЧКУ НЕ БУДИТ
МТОРЧИК И ПГДАРОЛЕ ПО КОТОРЫМ ВА ОЖЕ ГЕОТЫГАТЬ
НУДРАНИЗ ПО ВРЫЖЕНЕ

ЖА НА ИНСТРУКЦИЮ ПО ПОИСКУ НЕИСПРАВНОСТЕЙ В
ЛЮБОМ СЛОЖНОМ ТЕХНИЧЕСКОМ УСТРОЙСТВЕ
АВТОМОБИЛЕ

Е ТЕЛЕВИЗОРЕ И Т Д

Д ТАМ ПРИВЕДЕН ПЕРЕЧЕНЬ ВНЕШНИХ ПРИЗНАКОВ НЕ
ГОРИТ ЛАМПОЧКА НЕ ГУДИТ МОТОРЧИК И Т Д ПО
КОТОРЫМ ВЫ МО

Языковая модель 1

A roindan nunibr: 1234.



"A roindan nunibr: 1234."

Fig. 1: Why does the NN make such dumb mistakes?

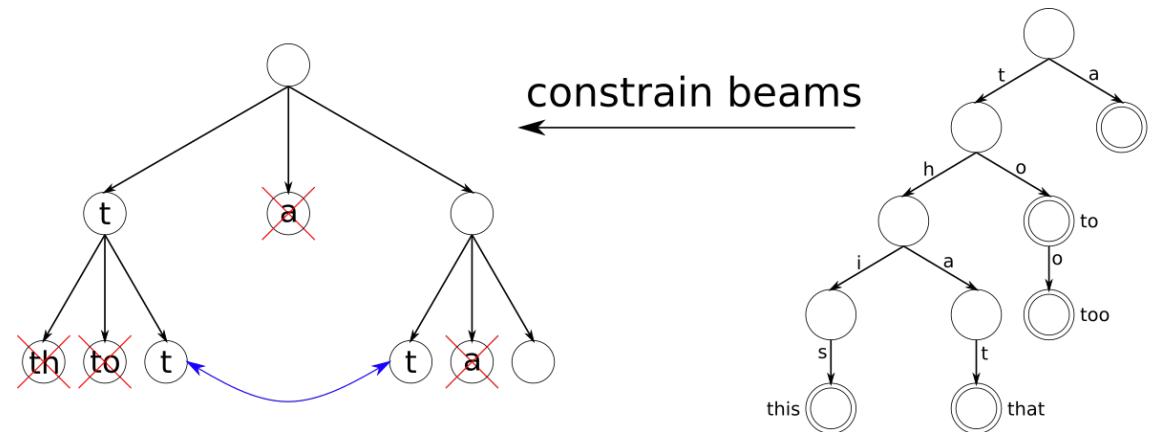
Языковая модель 2

A roindan number: 1234.



"A roindan nunibr: 1234."

Fig. 1: Why does the NN make such dumb mistakes?



Добавим Beam Search:
это поиск по графу возможных
вариантов,
и отсечение кандидатов
по дополнительным критериям

(И удаляем дубликаты букв для СТС,
они у нас неотличимы)

blank: 'aa-' + $\begin{cases} '-' = 'aa--' \rightarrow "a" \text{ (copy)} \\ 'a' = 'aa-a' \rightarrow "aa" \text{ (extend)} \\ 'b' = 'aa-b' \rightarrow "ab" \text{ (extend)} \end{cases}$

non-blank: 'aaa' + $\begin{cases} '-' = 'aaa-' \rightarrow "a" \text{ (copy)} \\ 'a' = 'aaaa' \rightarrow "a" \text{ (copy)} \\ 'b' = 'aaab' \rightarrow "ab" \text{ (extend)} \end{cases}$

Языковая модель 3

- Например, метрикой может быть вероятность букв составлять слово или быть популярным сочетанием букв.
- Нейросеть и сама это выучивает, но медленно, и у неё уж точно не хватит памяти, чтобы запомнить все слова языка.

```
Ground truth:      "the fake friend of the family, like the"  
Best path decoding: "the fak friend of the fomly hae tC"  
Beam search:       "the fak friend of the fomcly hae tC"  
Beam search with LM: "the fake friend of the family, lie th"
```

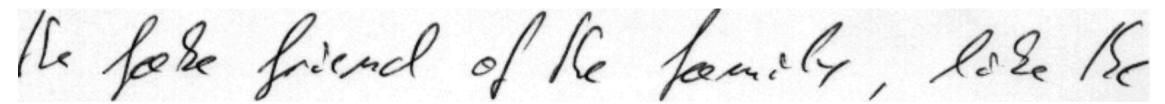
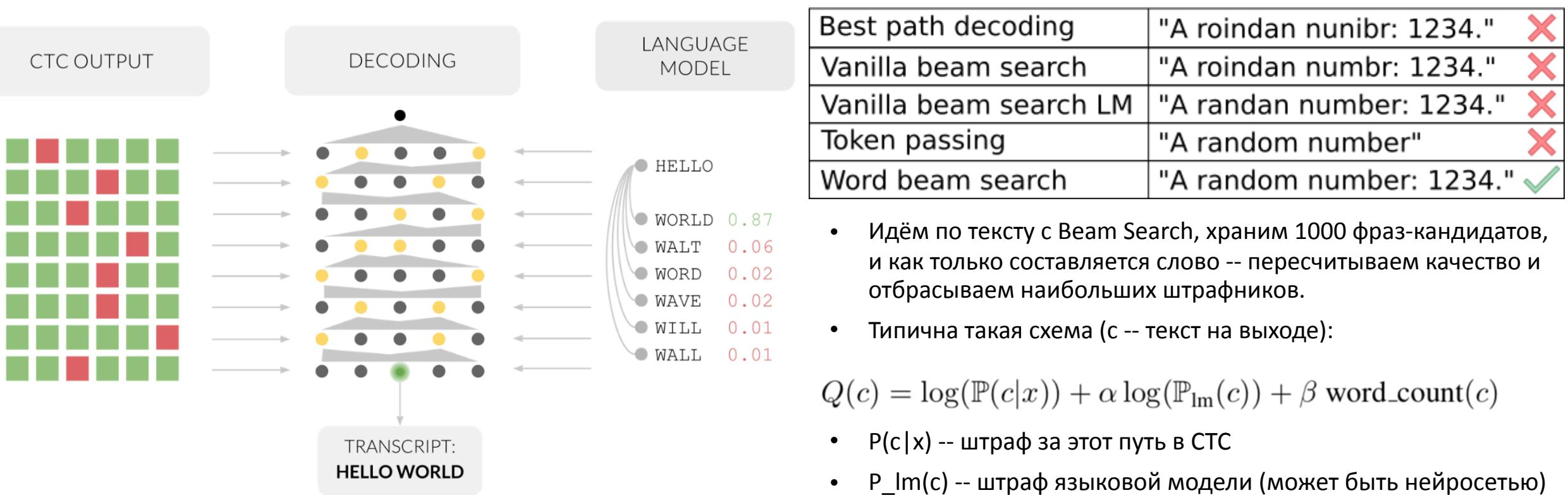


Fig. 8: Sample from IAM dataset.

- Очевидные недостатки:
 - не знает слова не из словаря
 - может исправить редкое слово на более частое

Языковая модель 4

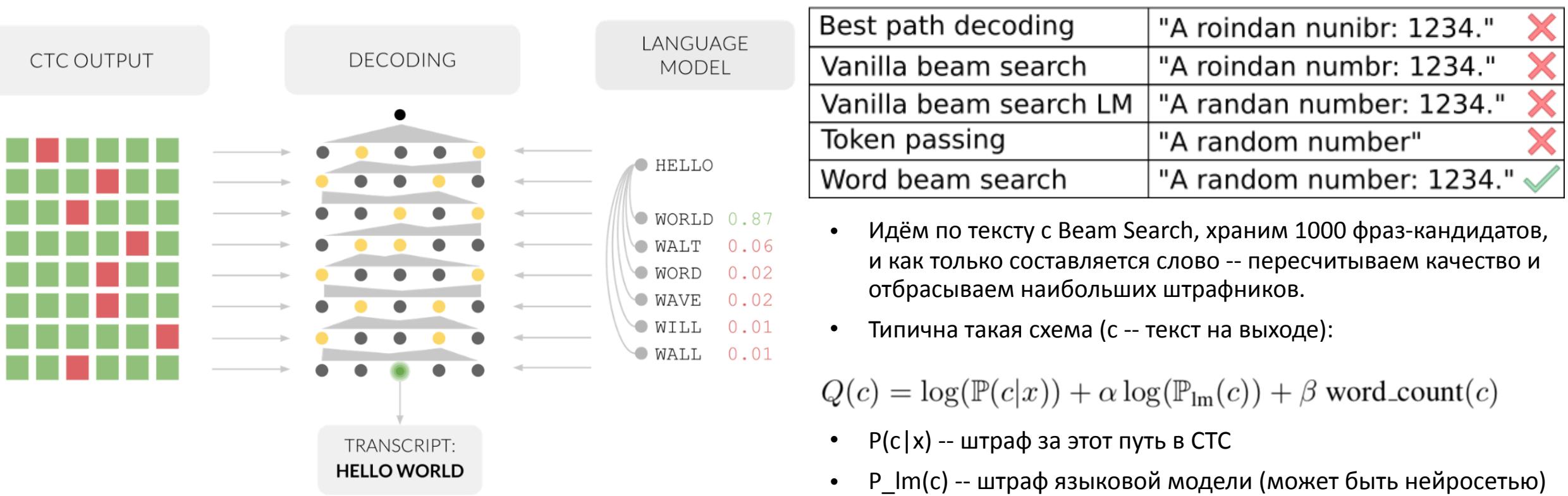


<https://towardsdatascience.com/word-beam-search-a-ctc-decoding-algorithm-b051d28f3d2e>

arxiv.org/abs/1412.5567

<https://medium.com/corti-ai/ctc-networks-and-language-models-prefix-beam-search-explained-c11d1ee23306>

Языковая модель 4



$$Q(c) = \log(\mathbb{P}(c|x)) + \alpha \log(\mathbb{P}_{\text{lm}}(c)) + \beta \text{word_count}(c)$$

- $\mathbb{P}(c|x)$ -- штраф за этот путь в CTC
- $\mathbb{P}_{\text{lm}}(c)$ -- штраф языковой модели (может быть нейросетью)
- $\text{word_count}(c)$ -- количество слов ("защита кур совой")

<https://towardsdatascience.com/word-beam-search-a-ctc-decoding-algorithm-b051d28f3d2e>

arxiv.org/abs/1412.5567

<https://medium.com/corti-ai/ctc-networks-and-language-models-prefix-beam-search-explained-c11d1ee23306>

Забавные факты

- "Эффект уличного шума" (Lombard effect)
- ADAM не даёт прироста скорости
- Чем плоха bidirectional model?
- CNN, 60 миллионов параметров и LSTM, 60 миллионов параметров : кто быстрее?
- Аугментации для речи
- Точность для разных языков (финский, китайский, английский, русский)

Что дальше?

- Насколько вообще возможно повышать качество распознавания?
- Как удалять шум?
- Можно ли эту нейросеть запихнуть в телефон?
- Как распознавать музыку?
- Как распознавать говорящего?
- А что насчёт синтеза речи?

Синтез речи

- Задачу тоже приходится делить на части, вот только части другие:
 - Генерация спектрограммы (predictor)
 - Генерация звука (vocoder)
- Датасеты:
 - LJ Speech (24 часа)

Сначала версия попроще

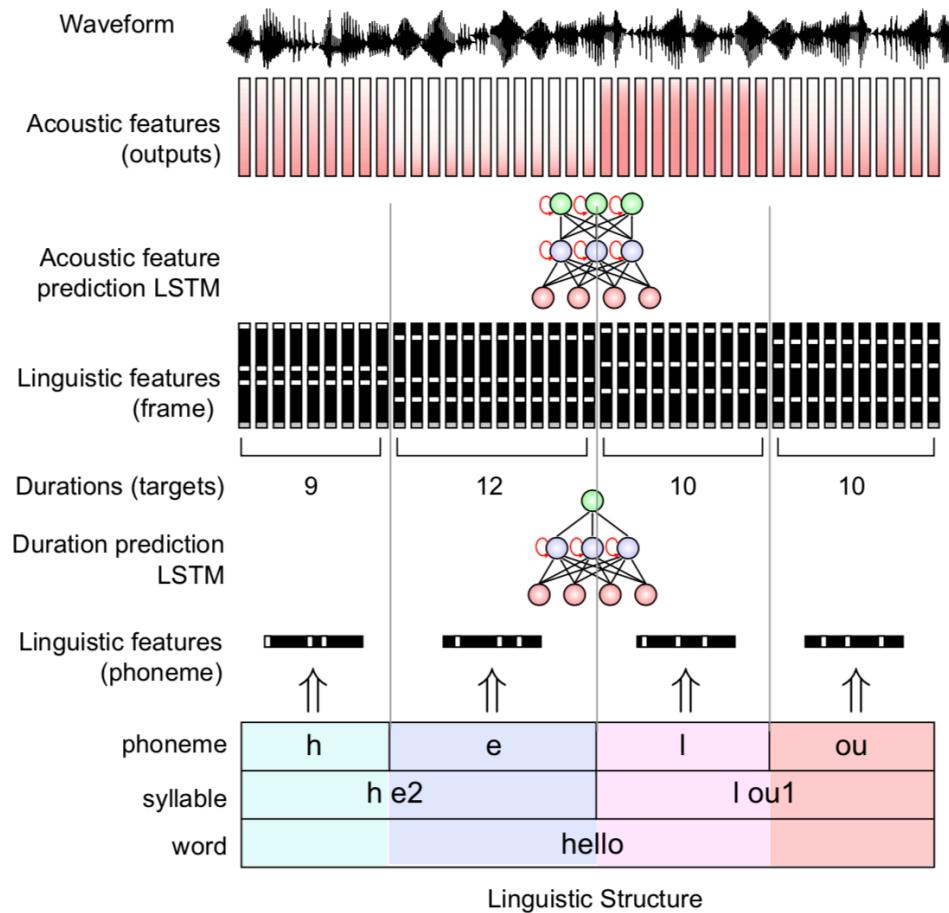
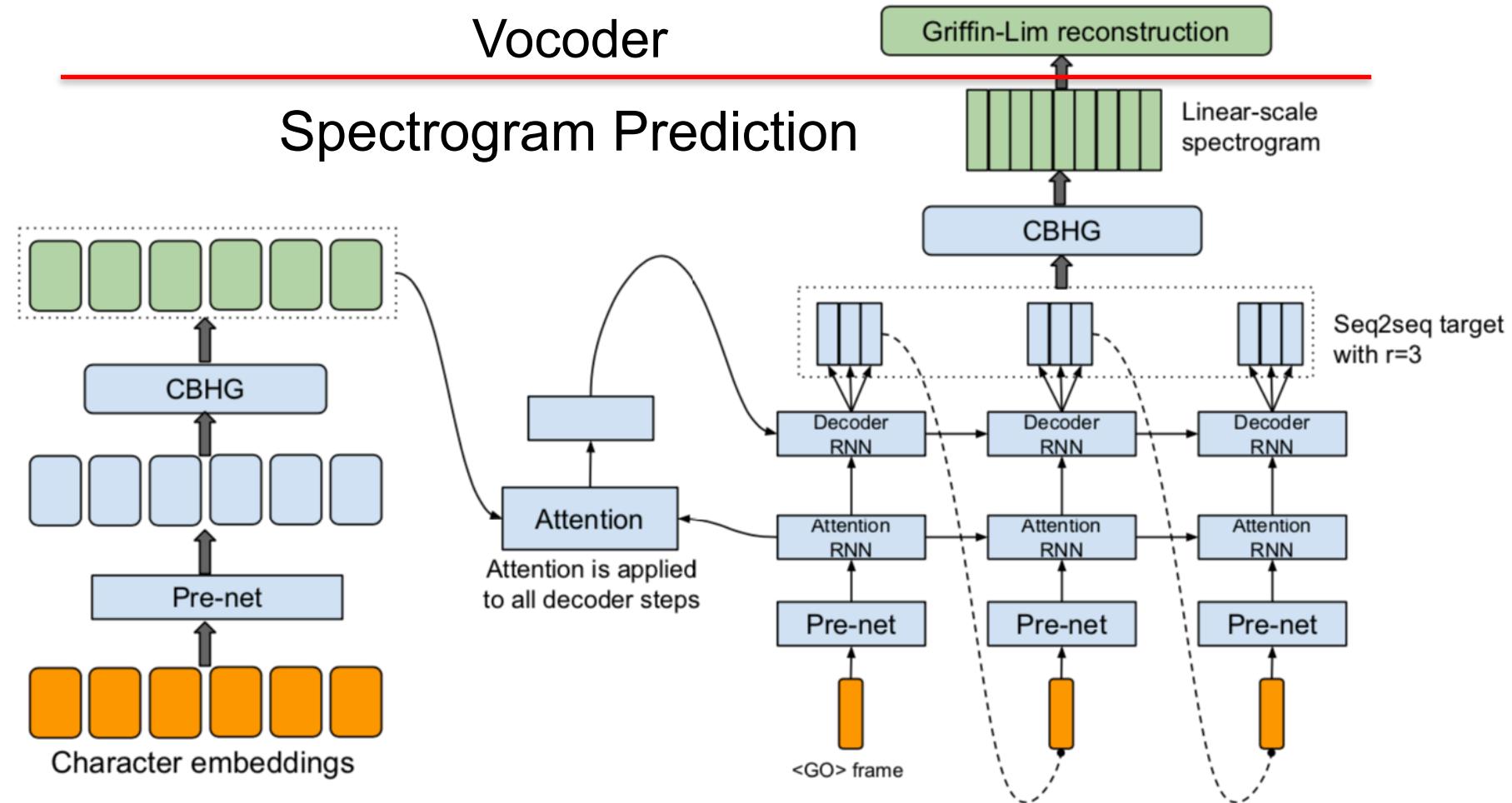
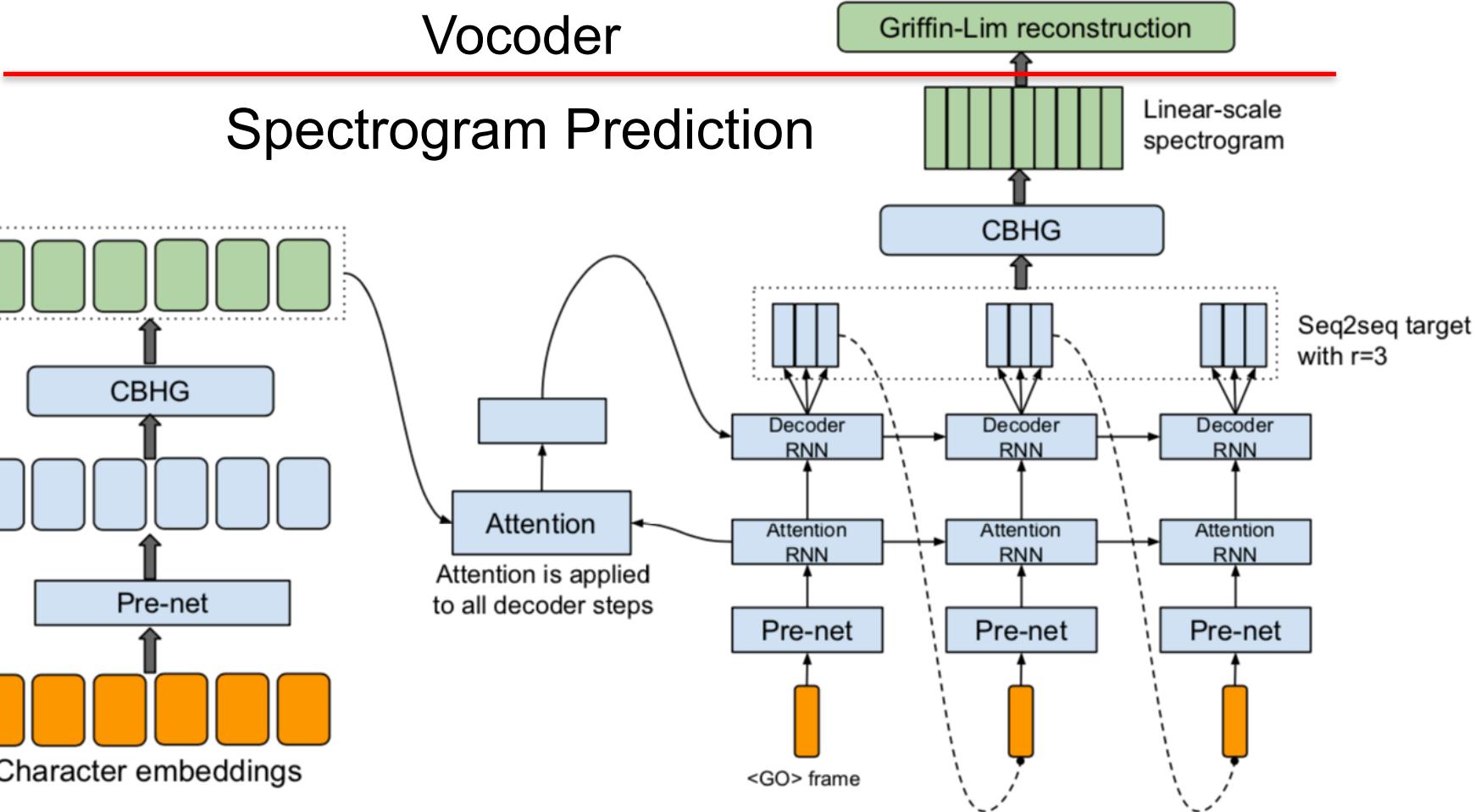


Figure 1: Overview of the streaming SPSS architecture using LSTM-RNN-based acoustic and duration models [9].

Tacotron

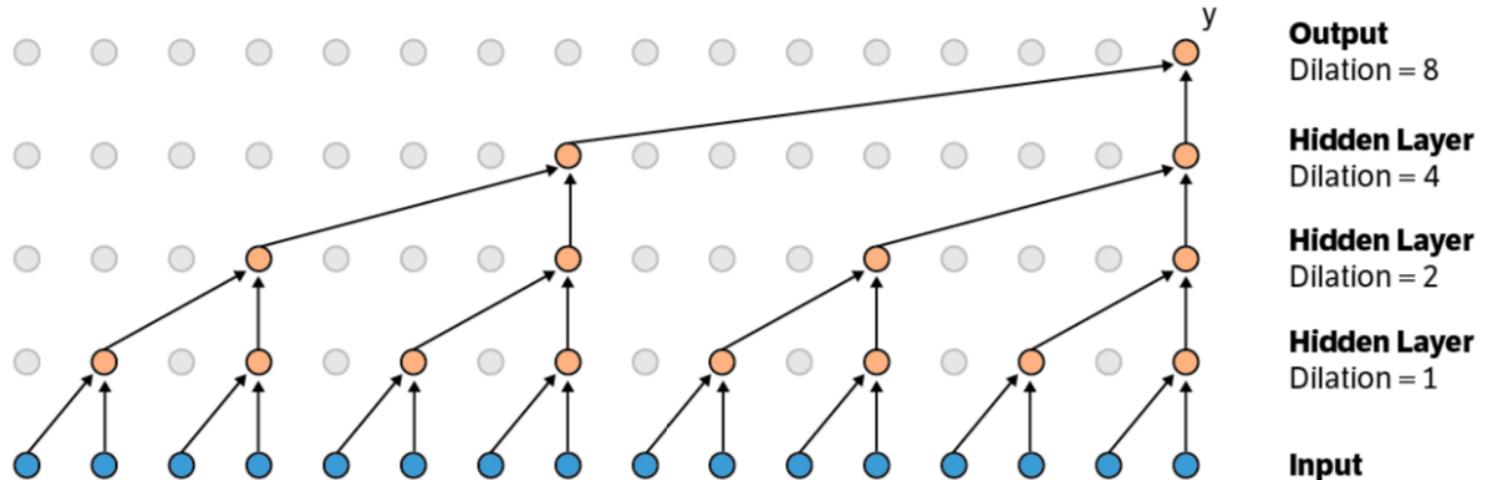


Tacotron

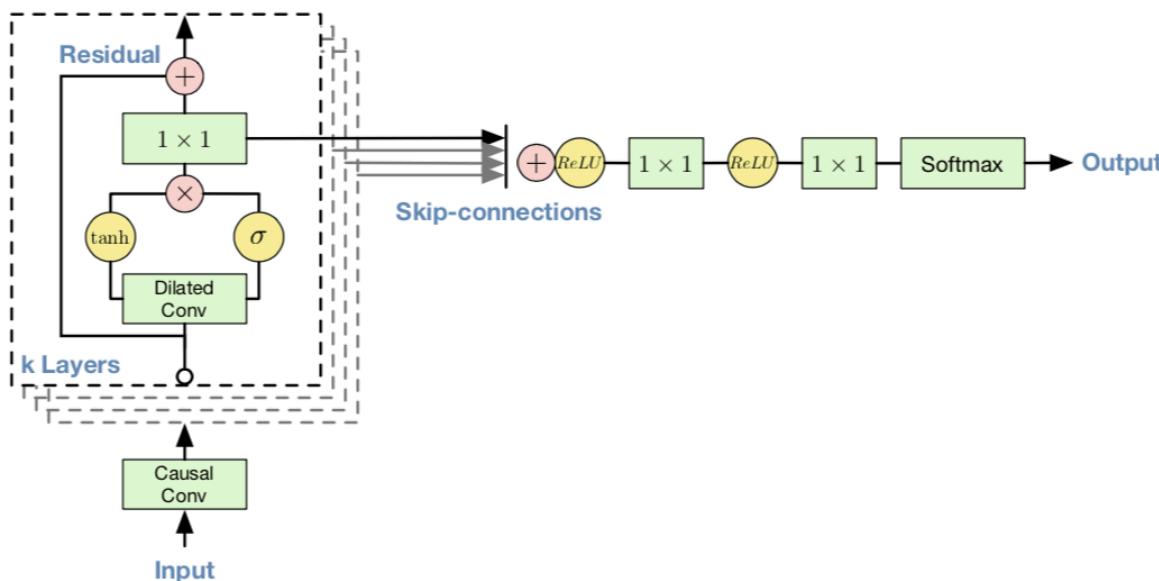


WaveNet

Causal convolution block (9 layers)

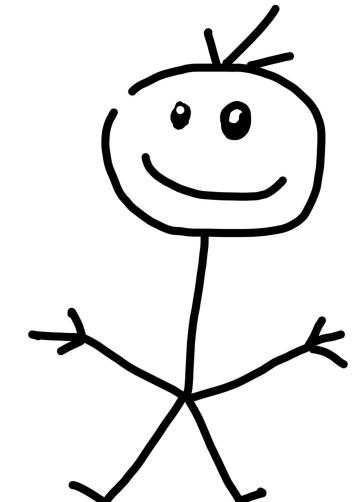


- Dilations:
 $\{1, 2, 4, 8, \dots, 512\} * N \rightarrow N$ блоков
- Каждый Dilation block заменяет Conv с ядром свёртки kernel=1024
- Похоже на ResNet?
- Causal Conv -- это Dilation=1 и без Residual

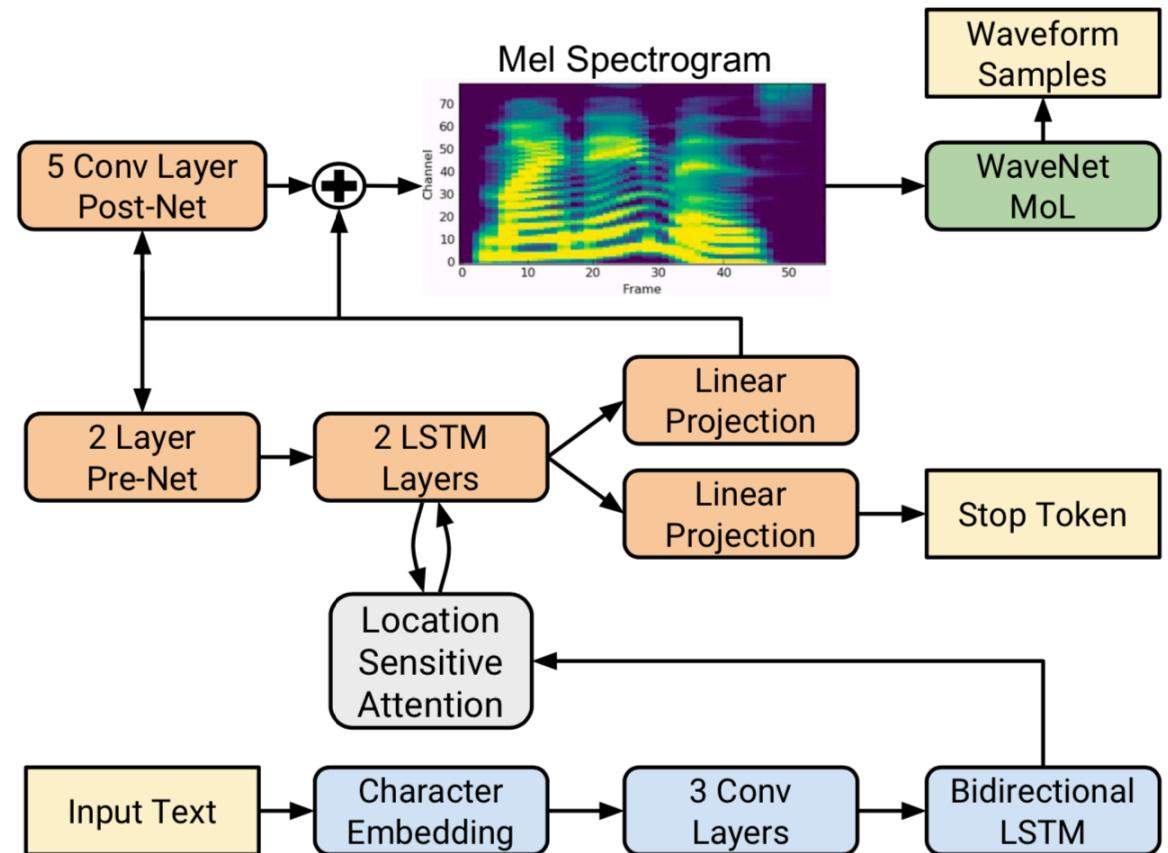


WaveNet

- А займет это 1 месяц на GPU...
- Не параллелится!
- Впрочем, уже есть альтернативы, дающие качество ненамного хуже

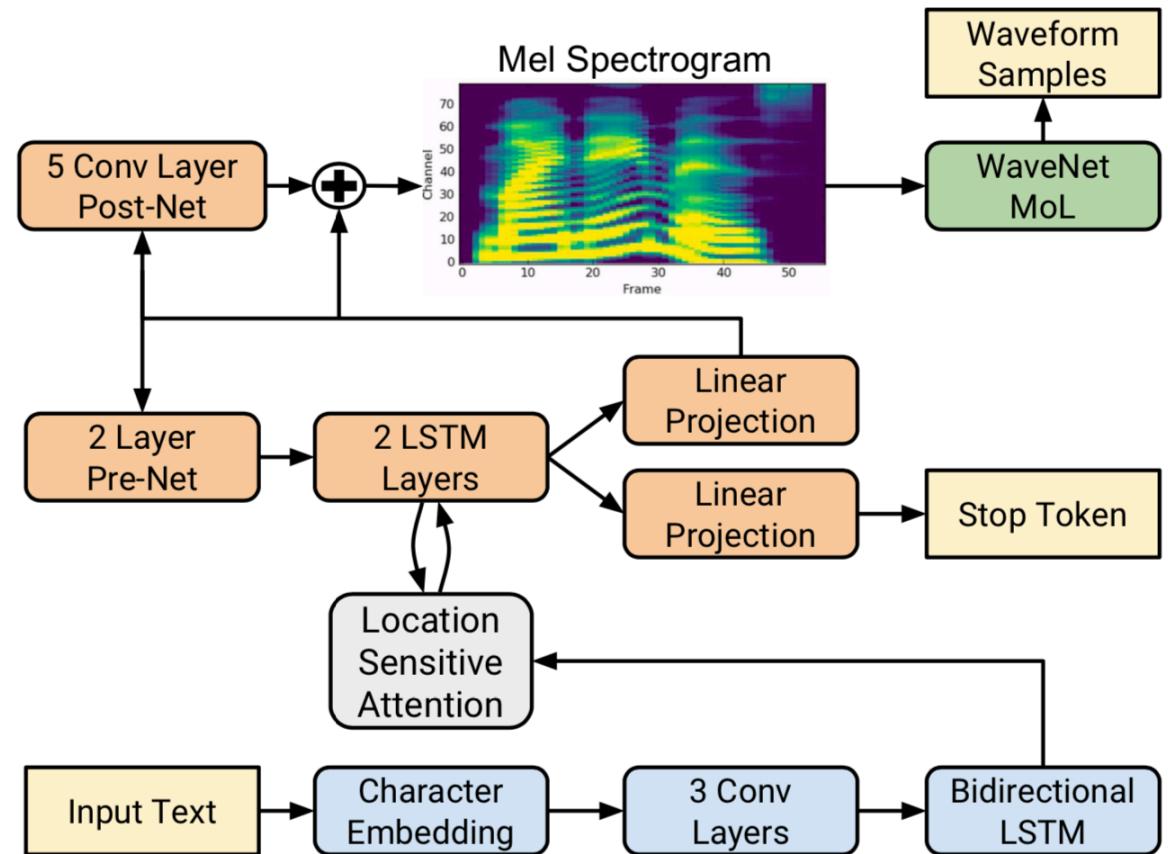
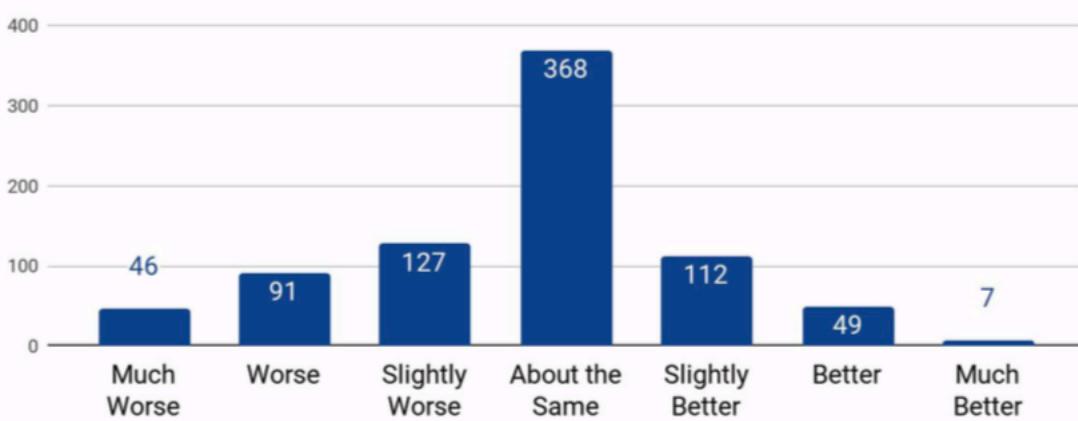


Tacotron 2



Tacotron 2

System	MOS
Parametric	3.492 ± 0.096
Tacotron (Griffin-Lim)	4.001 ± 0.087
Concatenative	4.166 ± 0.091
WaveNet (Linguistic)	4.341 ± 0.051
Ground truth	4.582 ± 0.053
Tacotron 2 (this paper)	4.526 ± 0.066



[Примеры Tacotron2 и тест: сможете ли вы различить робота и человека](#)

That's all, folks!