Обработка естественного языка

Сергей Николенко

Зимняя школа ВШЭ, 6 февраля 2016 г.

Outline

План

- Я расскажу о том, как в наше время люди обрабатывают естественный язык.
- Это надо бы разделить на три части:
  - какие задачи стоят перед нами;
  - как их люди решают;
  - чего можно ожидать в будущем.

План

- Но, конечно, у нас и времени столько нет, и не знаю я ничего о большей части этого плана. :)
- Поэтому мы сделаем так:
  - сначала действительно кратко поговорим о задачах NLP;
  - потом я расскажу о революции в методах, которая происходит во всей нашей науке, в том числе NLP, в последние десять лет;
  - а часть о будущем, скорее всего, останется на будущее. :)

Какие бывают задачи

- NLP – это про человеческие языки: русский, английский и всё такое.
- Основная цель – научить компьютер понимать естественные языки и говорить на них.
- Что это значит в реальности? Большие области:
  1. лингвистические задачи;
  2. information extraction: выделить информацию из текста;
  3. information retrieval: выдать по запросу то, что надо (это то, что делают google и yandex);
  4. работа со звуком: распознавание-синтез речи и смежные задачи;
  5. и так далее...

Какие бывают задачи

- Примеры конкретных задач:
  - категоризация текстов (классификация);
  - выделение тем текстов (topic modeling);
  - машинный перевод;
  - ответ на вопросы (question answering, не путать с IR);
  - sentiment analysis;
  - автоматическое реферирование (summarization);
  - выделение именованных сущностей (named entity recognition);
  - разбор синтаксиса (как построить синтаксическое дерево предложения);
  - разделение разных смыслов слов (word sense disambiguation);
  - разбор морфологии, анафора, части речи, ...

Какие бывают задачи

- Но мы с вами поговорим не столько о конкретных задачах, сколько о революции, связанной с глубоким обучением (deep learning).
- Это главная тема последних десяти лет в machine learning; глубокое обучение привело к мощным прорывам во многих областях (мы о них кратко поговорим).
- Например, две недели назад в Nature вышла статья о том, как основанная на глубоком обучении модель AlphaGo победила чемпиона Европы по го, 2-й профессиональный дан, со счётом 5-0; раньше ничего подобного и близко не получалось.
- Но давайте начнём немного издалека...

Почему мы лучше?

- Компьютер считает быстрее человека.
- Но гораздо хуже может:
  - понимать естественный язык,
  - узнавать людей и распознавать изображения,
  - обучаться в широком смысле этого слова,
  - ...
- Почему так?

Строение мозга

- Как человек всего этого добивается?
- В мозге много нейронов; каждый нейрон:
  - через дендриты получает сигнал от других нейронов;
  - время от времени запускает сигнал по аксону;
  - через синапсы сигнал аксона доходит до дендритов других нейронов.

Строение мозга

- Нейрон создаёт электрический импульс – spike в напряжении аксона.
- Импульсы нейрон всегда подаёт с некоторой firing rate.
- Если он неактивен, всё равно подаёт (spontaneous firing rate), но когда он видит активность на входе от других нейронов, его firing rate сильно увеличивается.
- Связь в синапсе/дендрите может быть как положительная, так и отрицательная (excitation/inhibition).

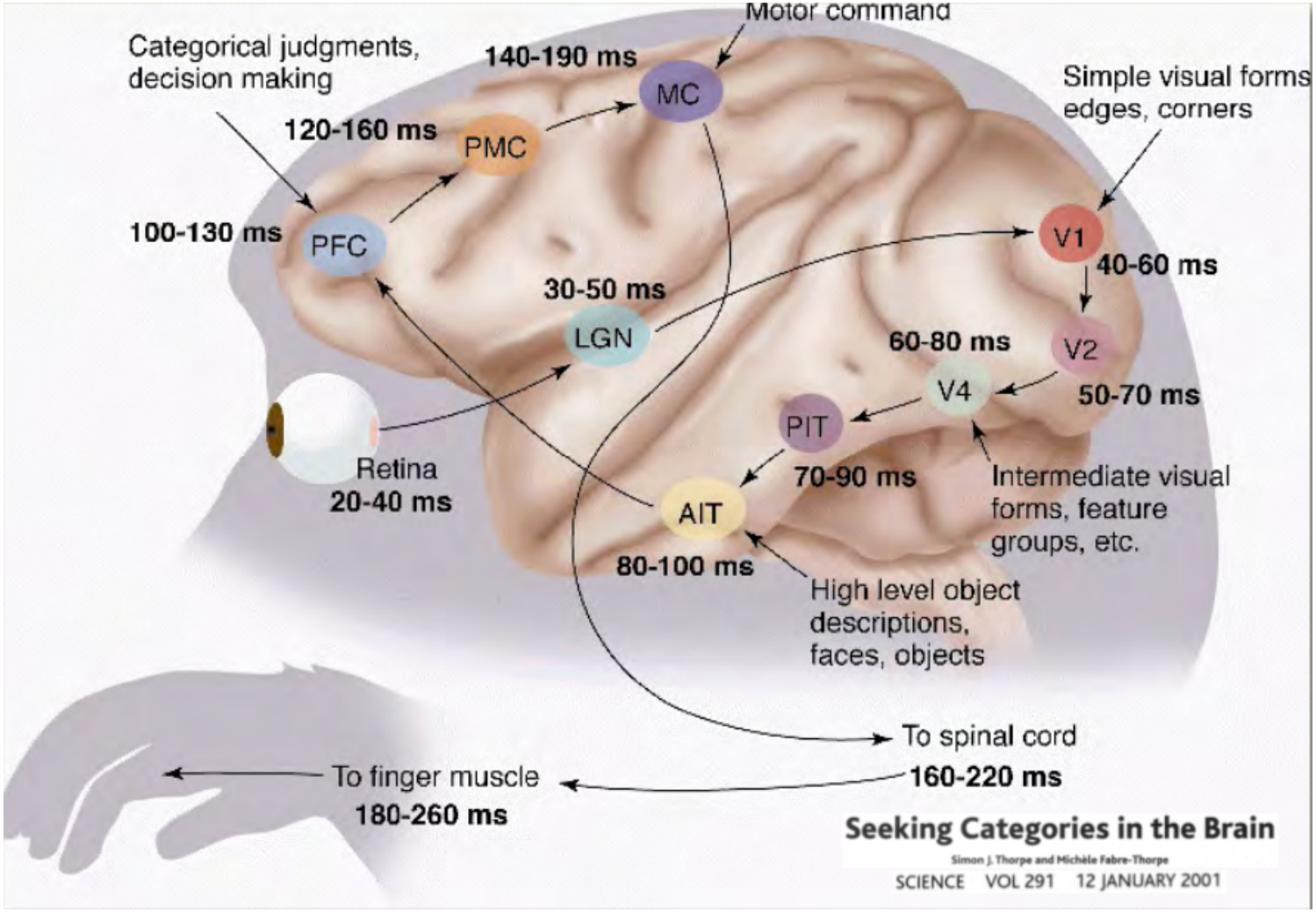
Строение мозга

- Таким образом, получается, что нейрон – это такая стохастическая штука, которая время от времени выдаёт сигналы.
- Нейрон довольно точно моделируется пуассоновским процессом, интенсивность которого зависит от входов.
- Что немножко странно: нейроны на самом деле умеют довольно точно во времени сигнал подавать, могли бы передавать гораздо больше информации.
- Мы об этом тоже поговорим.

Строение мозга

- И ещё: firing rate бывает от 10 до 200 герц примерно.
- Мы распознаём лицо за пару сотен миллисекунд.
- То есть в распознавании не могло быть цепочки длиннее нескольких десятков штук, а скорее меньше!
- Но всего нейронов очень много: 10^{11} нейронов, в среднем 7000 связей у каждого, т.е. 10^{15} синапсов.
- Значит, мозг очень хорошо структурирован в этом смысле.

Мозг

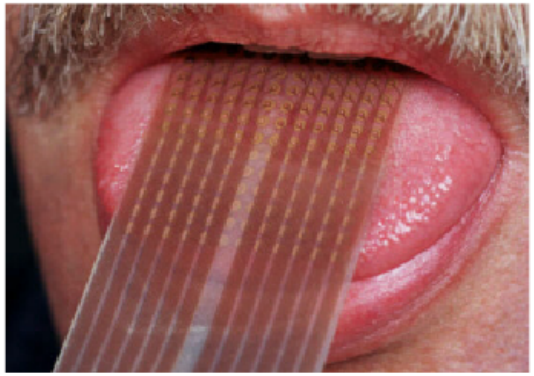

Motor command
Categorical judgments, decision making
140-190 ms
MC
Simple visual forms edges, corners
120-160 ms
PMC
100-130 ms
PFC
V1
40-60 ms
30-50 ms
LGN
60-80 ms
V2
V4
50-70 ms
PIT
Retina
20-40 ms
70-90 ms
Intermediate visual forms, feature groups, etc.
AIT
80-100 ms
High level object descriptions, faces, objects
To spinal cord
160-220 ms
To finger muscle
180-260 ms
Seeking Categories in the Brain
Simon J. Thorpe and Michèle Fabre-Thorpe
SCIENCE VOL 291 12 JANUARY 2001

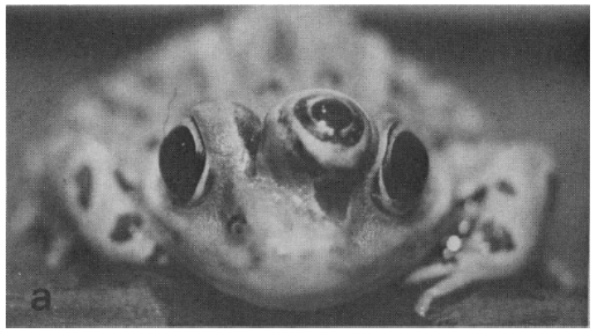
Обучение признаков

- Другая сторона вопроса – обучение признаков.
- Мозг очень хорошо умеет обучаться на очень-очень маленькой выборке данных.
- Как он это делает?
- Естественный язык – это один из самых удивительных здесь примеров, кстати; дети учат по несколько новых слов в день (в среднем!).

Пластичность

Другой аспект – пластичность мозга.





Обучение признаков

- Мозг может адаптироваться к новым источникам информации.
- Значит, не исключено, что у него есть какой-то единый алгоритм, который может обучаться на самых разных данных.
- Можем ли мы его тоже как-то промоделировать?

Обучение признаков

- Системы обработки неструктурированной информации выглядят обычно так:

$$\text{вход} \longrightarrow \text{признаки} \longrightarrow \text{классификатор}$$

- Люди много десятилетий пытались придумать хорошие признаки:
  - MFCC для распознавания речи;
  - SIFT для обработки изображений;
  - ...

Обучение признаков

- Задача feature engineering: как сделать такие признаки?
- Feature learning: может быть, можно найти признаки автоматически?
- Мозг ведь это как-то делает...

Outline

Искусственные нейронные сети

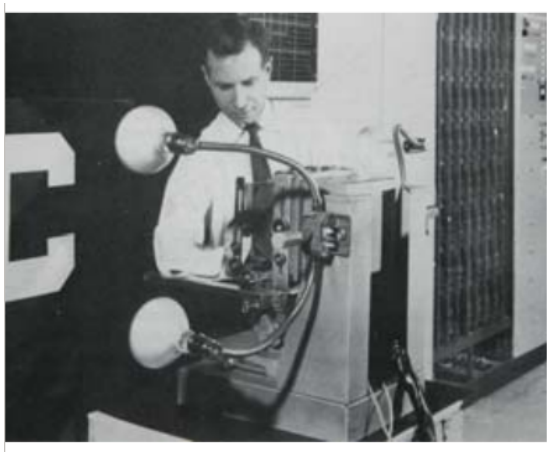
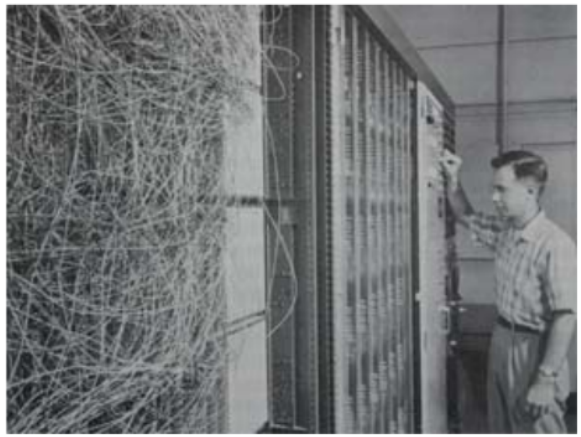
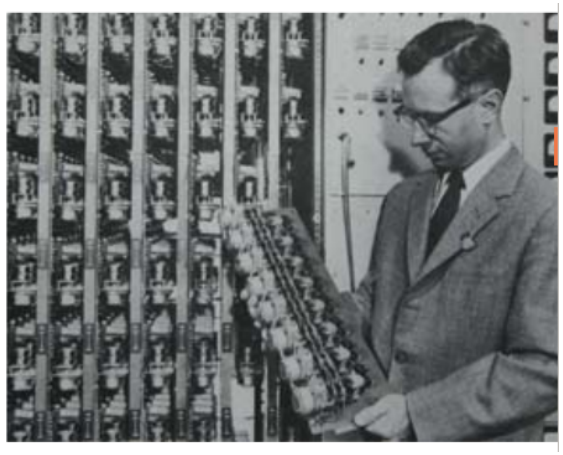
- Основная мысль нейронных сетей позаимствована у природы: есть связанные между собой нейроны, которые передают друг другу сигналы.
- Есть нейронные сети, которые стараются максимально точно моделировать головной мозг.
- В AI нейронные сети существуют давно, но лишь недавно их научились хорошо готовить (об этом мы и говорим сегодня).

История ANN

- Warren McCulloch & Walter Pitts, 1943: идея.
- Идею искусственных сетей, похожие на современные (с уровнями), предложил Алан Тьюринг (1948).
- Rosenblatt, 1958: перцептрон. Линейная разделяющая поверхность плюс сигмоид.
- Тогда же появился алгоритм обучения градиентным спуском.

История ANN

- Тёплые ламповые реализации перцептрона сразу начали применять для распознавания букв; правда, результаты были не слишком впечатляющие.

История ANN

- 1960-е годы: изучали перцептрон.
- (Minsky, Papert, 1969): XOR нельзя моделировать перцептроном.
- Это почему-то восприняли как большую проблему, которая ставит крест на нейронных сетях.

История ANN

- (Brison, Hoback, 1969): предложили алгоритм backpropagation (обратное распространение ошибки).
- (Hinton, 1974): переоткрыл backpropagation, с тех пор он стал популярным.
- Во второй половине 1970-х появились многоуровневые ANN, была разработана современная теория.
- Глубокие модели появились в первой половине 1980-х гг.! Это вообще не очень хитрая идея сама по себе.

История ANN

- (Morgan, Boulard, 1988): нейронные сети для распознавания речи.
- (Waibel et al., 1989): TDNN (time-delay neural networks).
- (Robinson et al., 1990): рекуррентные нейронные сети.

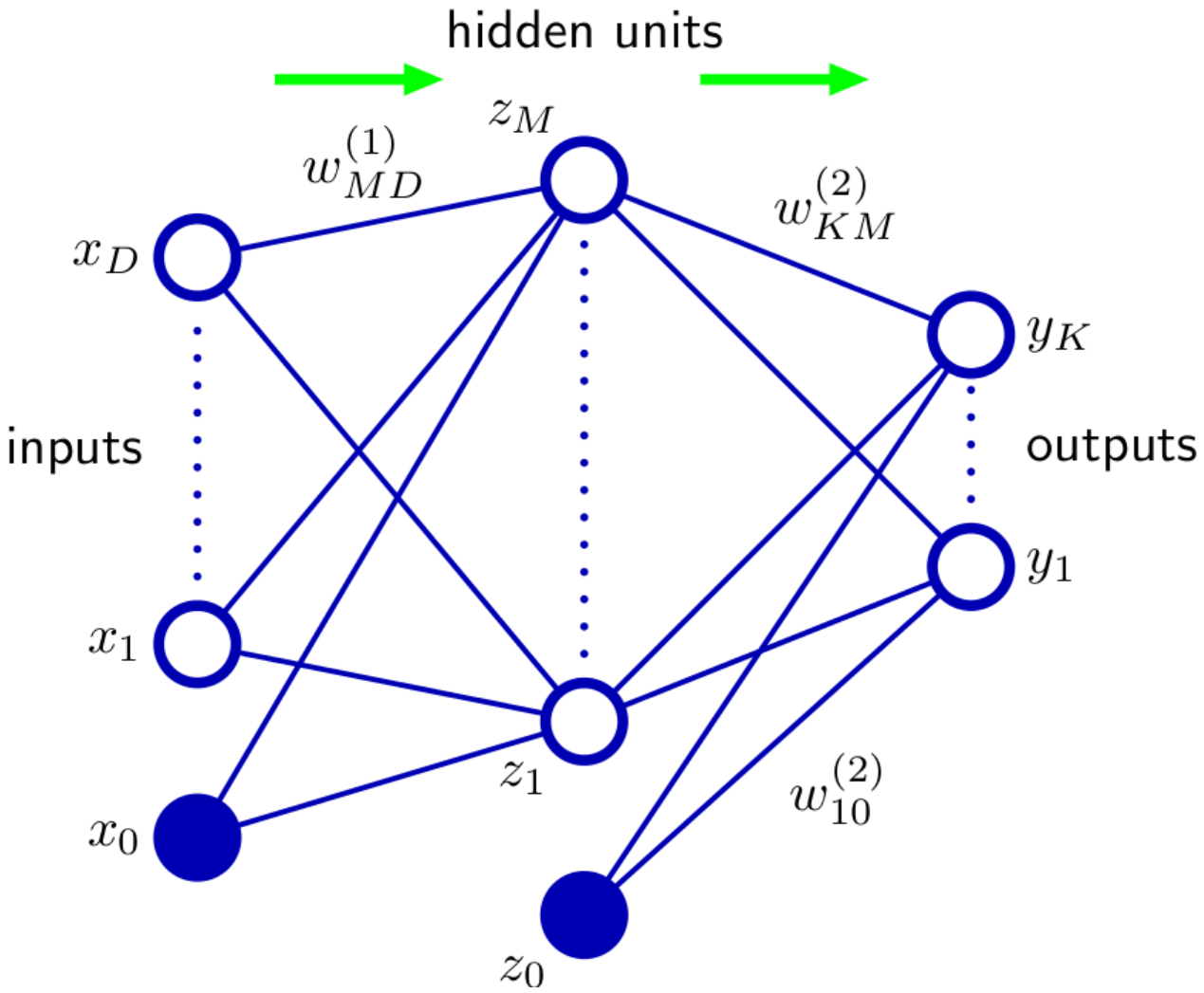
История ANN

- Но к началу 1990-х решили, что сети использовать смысла нет.
- Это было потому, что тогда их ни математически не умели, ни вычислительно не могли нормально обучить.
- Нужно было обучать сети неделями, а качество, например, на распознавании речи проигрывало HMM, в других задачах проигрывало другим методам.
- John Denker, 1994: «neural networks are the second best way of doing just about anything».

История ANN

- Deep learning начинается в 2006, когда Geoffrey Hinton изобрёл Deep Belief Networks (DBN, DNN).
- Основная идея: научились делать предобучение при помощи машин Больцмана (RBM; появились в 1983), благодаря которому сеть потом приходит в гораздо более разумный локальный максимум.
- А компьютеры стали мощнее, и вычислительно тоже всё стало доступно.

Структура сети



Зачем нужны глубокие архитектуры

- Одну и ту же функцию можно выразить более компактно и менее компактно.
- Часто это связано с *глубиной* представления:
  - схема глубины 2 выражает любую булевскую функцию, но не все эффективно;
  - есть разница и между схемами глубины k и $k+1$ (Hastad, Yao).
- Нечто в том же духе происходит и с арифметическими схемами, и со схемами с разными другими операциями...

Зачем нужны глубокие архитектуры

- Примеры из машинного обучения:
  - линейная регрессия – схема глубины 1 с линейными гейтами;
  - логистическая регрессия – схема глубины 1 с гейтами вида σ от линейных;
  - нейронная сеть со скрытым уровнем – схема глубины 2 с гейтами-нейронами;
  - если добавить вычисление ядра $K(\boldsymbol{x}, \boldsymbol{x}_n)$ в набор операций, то kernel machines (например, SVM) – это схемы глубины 2: линейная комбинация $K(\boldsymbol{x}, \boldsymbol{x}_n)$ от всех тестовых примеров;
  - бустинг добавляет один уровень к его базовым слабым классификаторам;
  - ...

Зачем нужны глубокие архитектуры

- Теорема (Hornik, 1991):
  - нейронная сеть с одним скрытым уровнем может приблизить любую непрерывную функцию с любой наперёд заданной точностью, если у неё будет достаточно много нейронов на скрытом уровне.
- Теорема применима к разным функциям активации, в том числе σ, tanh и др.
- Но, конечно, «достаточно» – это очень много, да и «может приблизить» не означает, что мы сможем придумать алгоритм обучения.

Зачем нужны глубокие архитектуры

- Можно ли как-то использовать более глубокие архитектуры, чтобы лучше обучать?
- Есть методы, которые обучают локальные (local templates); например, те же kernel machines вида $f(\boldsymbol{x}) = b + \sum_i \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i)$ с локальным ядром вроде $K(\boldsymbol{x}, \boldsymbol{x}_i) = e^{-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}$.
- Если научиться строить глубокую архитектуру из них, сила может теоретически возрасти экспоненциально: local representations vs. distributed representations.

Глубокие нейронные сети

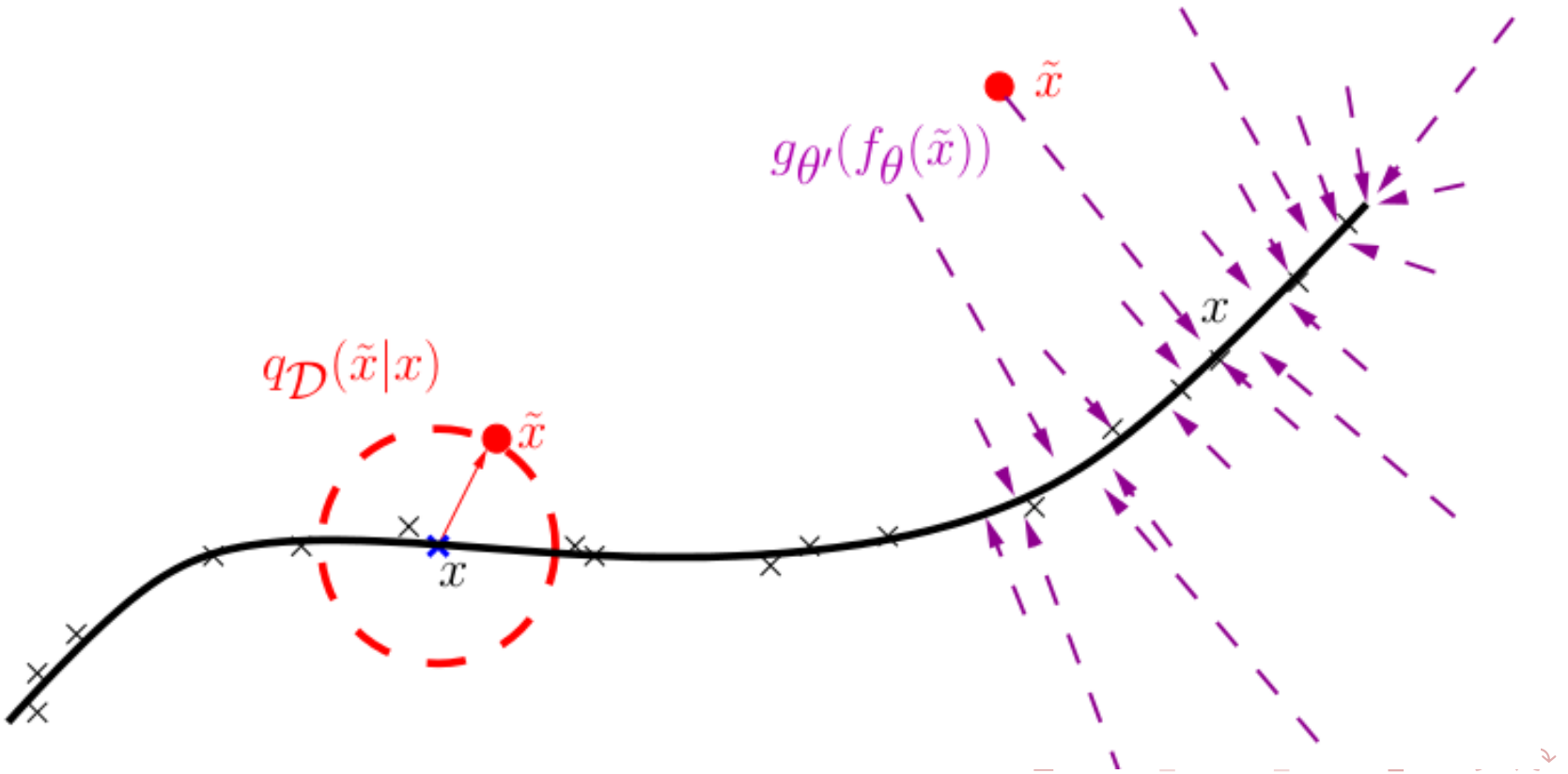
- Как построить глубокую нейронную сеть, совершенно понятно.
- Но с обучением проблемы: градиент застревает в локальных минимумах и плато.
- Vanishing gradients: при backpropagation перемножаются маленькие числа, получается почти ноль.
- Основная базовая идея (Hinton, 2006 и др.): давайте предобучать нижние уровни глубокой сети так, чтобы там появлялись признаки, описывающая особенности данных.
- Т.е. давайте делать unsupervised предобучение, а потом уже обучать последний уровень: обучать $p(\text{data})$ вместо $p(\text{label} \mid \text{data})$.

Глубокие нейронные сети

- В принципе, мозг именно это и делает.
- Теперь надо научиться делать unsupervised learning / feature extraction.
- В качестве составных частей глубоких моделей используются три основных метода:
  1. autoencoders;
  2. restricted Boltzmann machines;
  3. sparse coding.

Глубокие нейронные сети

- Например, denoising autoencoders: получается manifold learning: возвращаем точки на многообразие входов в датасете.

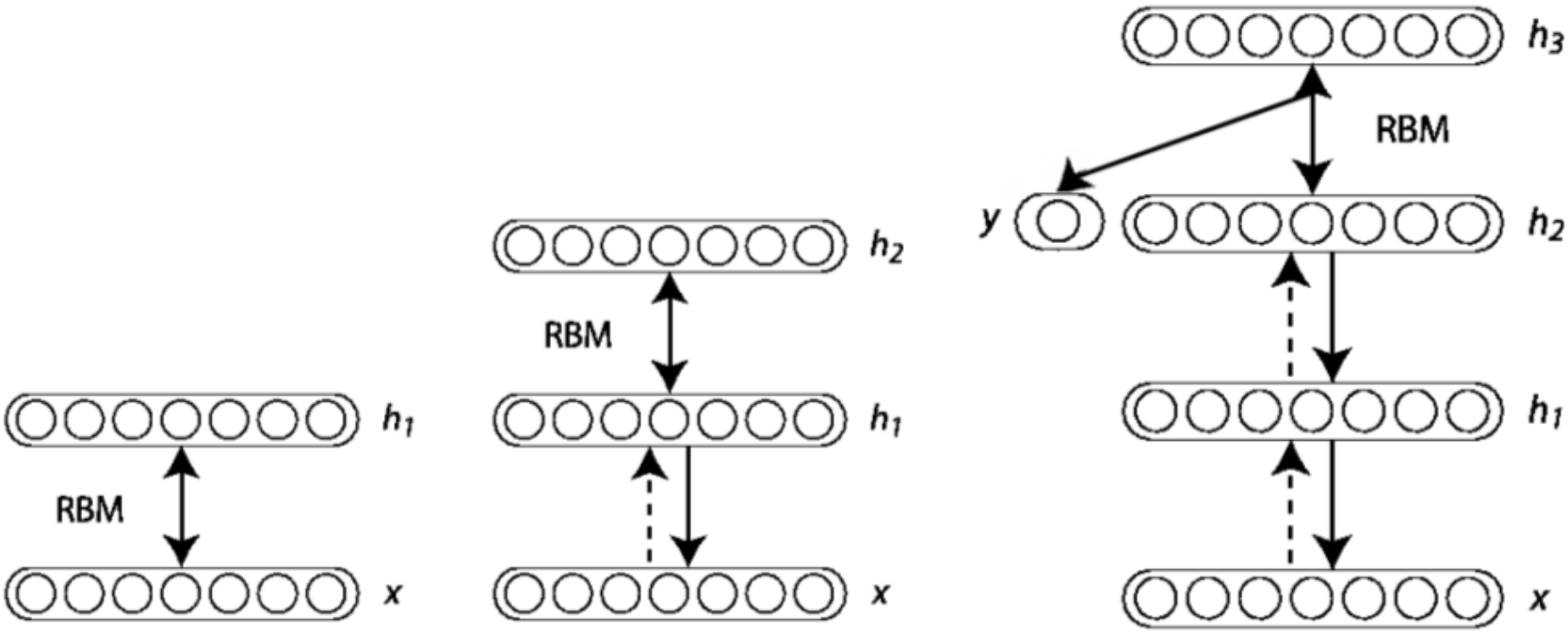


Глубокие нейронные сети

- Смысл сначала был в том, чтобы обучать каждый уровень по отдельности, снизу вверх, подавая нижние уровни на вход высшим.
- Потом появились хорошие методы регуляризации (dropout!), и сейчас часто обучают модели сразу целиком.
- Кроме того, появились вычислительные мощности для этого (GPU!), модели сейчас всё больше и больше.
- Мы подробно о математике говорить не будем, а скажем пару слов об архитектурах и перейдём к применениям, связанным с естественным языком.

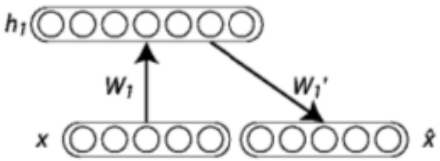
Глубокие нейронные сети

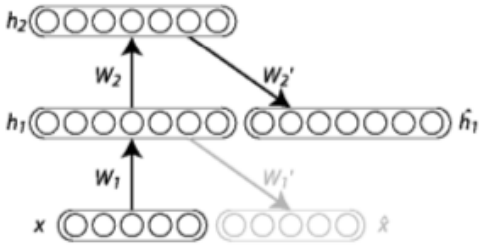
- Классические DBN (2006): обучаем через жадное обучение RBM, потом на последнем уровне можем сделать классификатор.

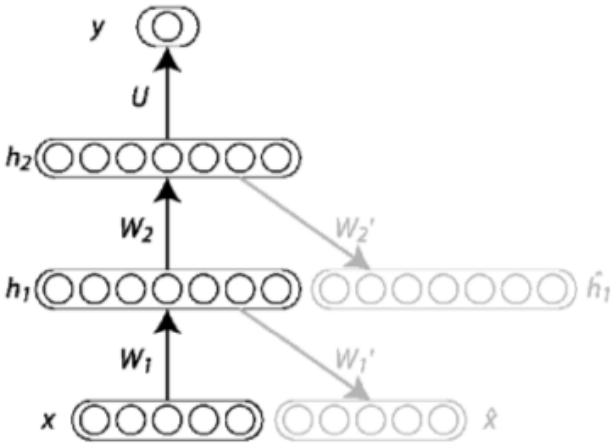


Глубокие нейронные сети

- Нейронные сети: stacked autoencoders (denoising, contractive).

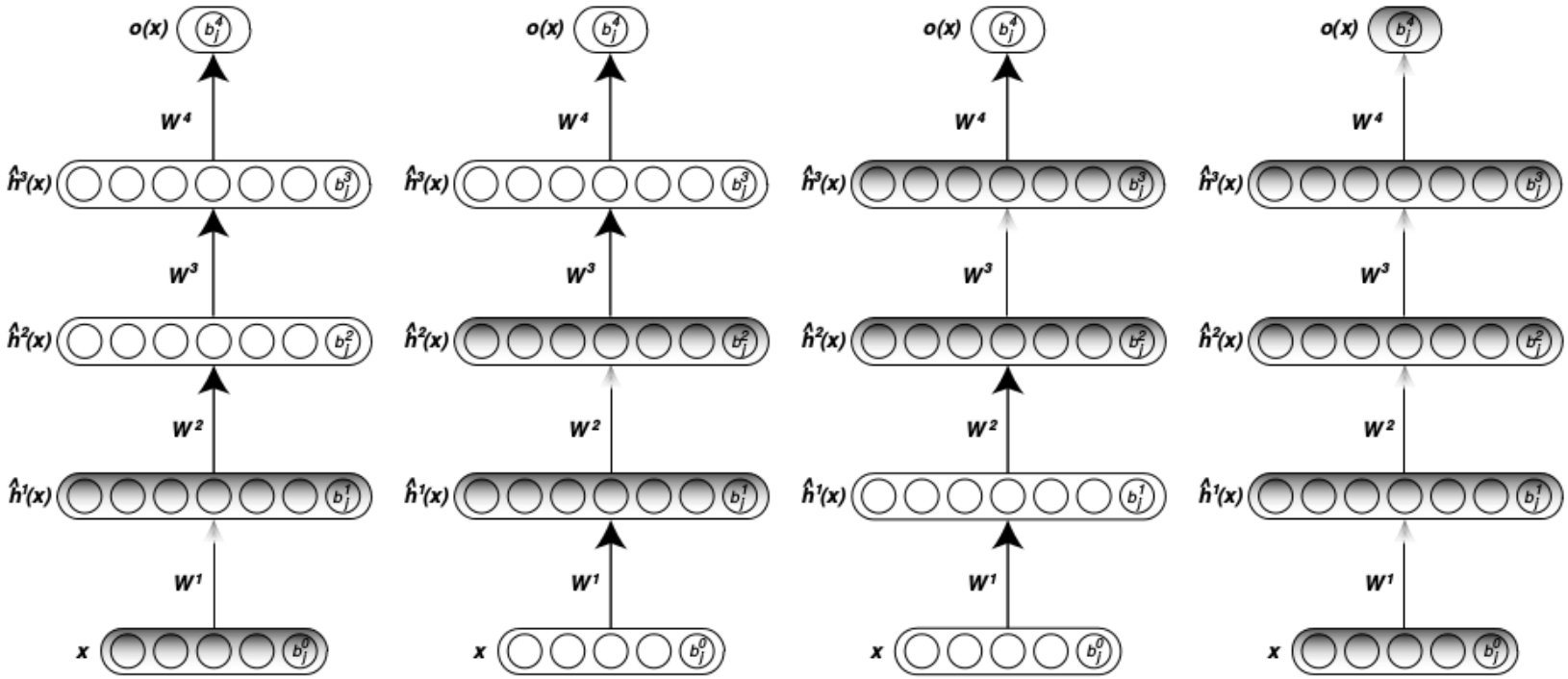






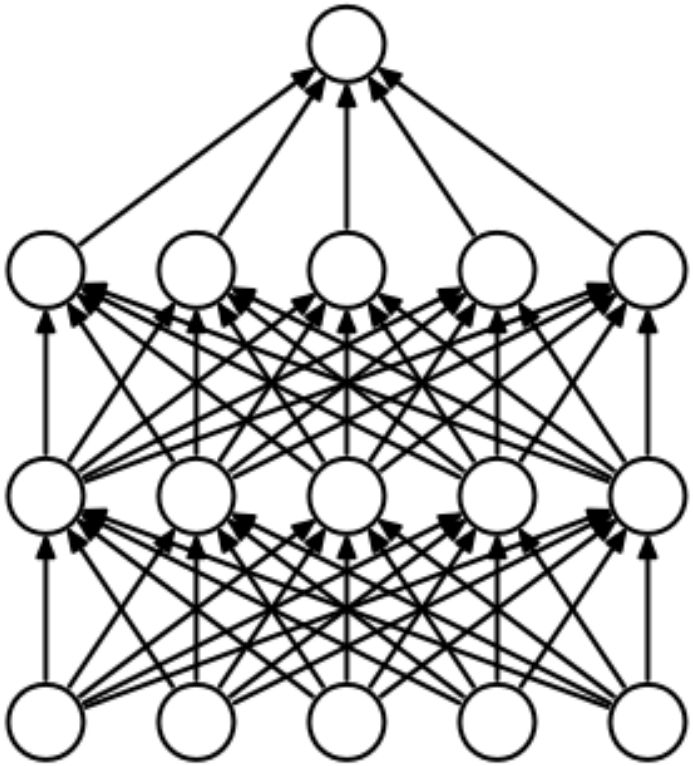
Глубокие нейронные сети

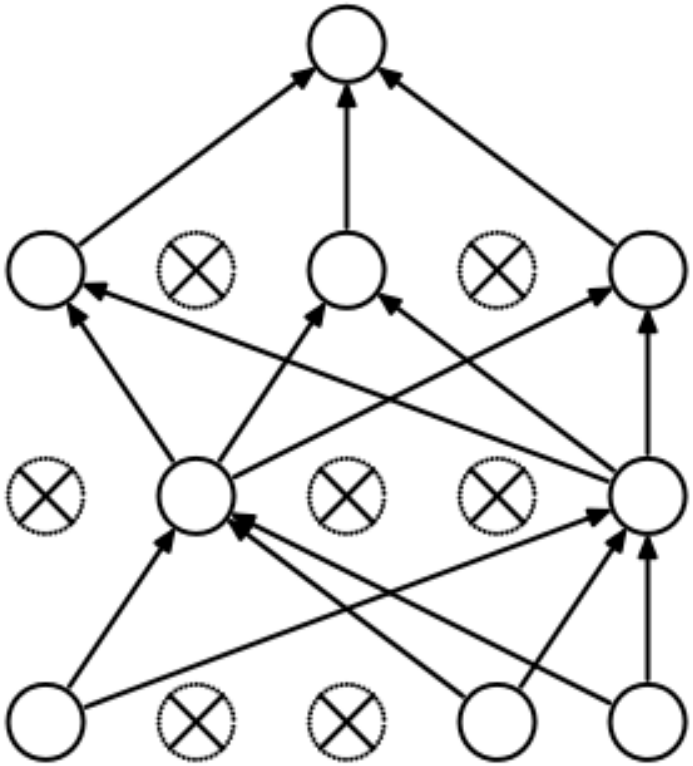
- В любом случае сначала обучают признаки, а потом делаем fine-tuning всей сети.



Глубокие нейронные сети

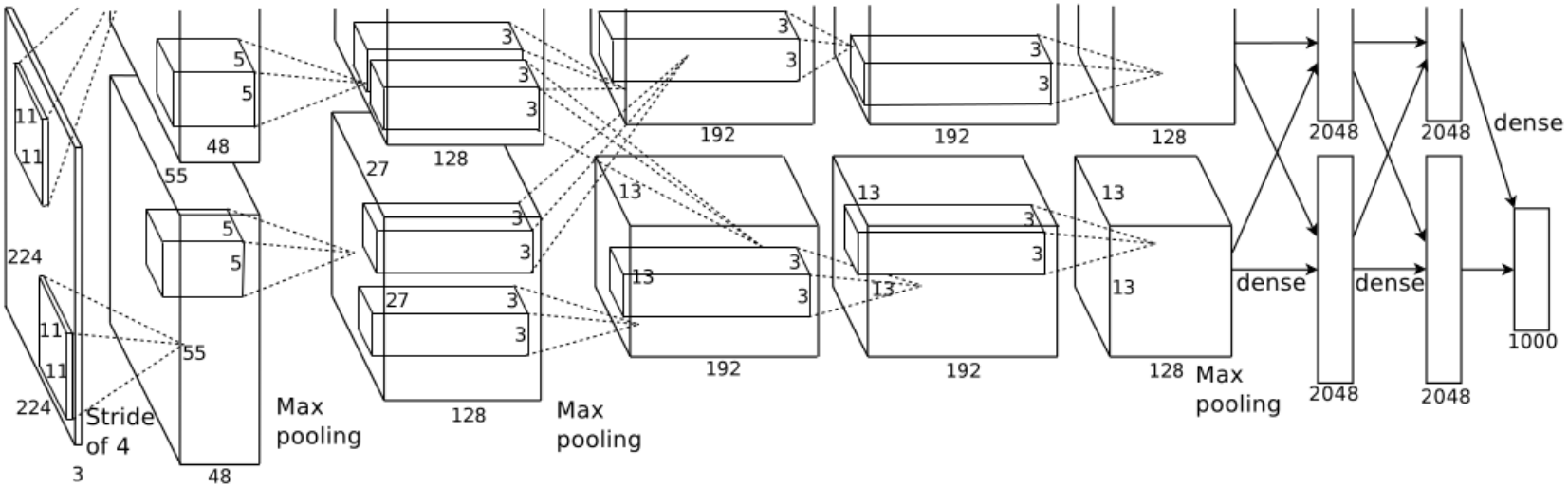
- Dropout: давайте выкидывать нейроны! Как в denoising autoencoder, но выкидываем не только вход, а ещё и скрытые уровни (Srivastava et al., 2013).





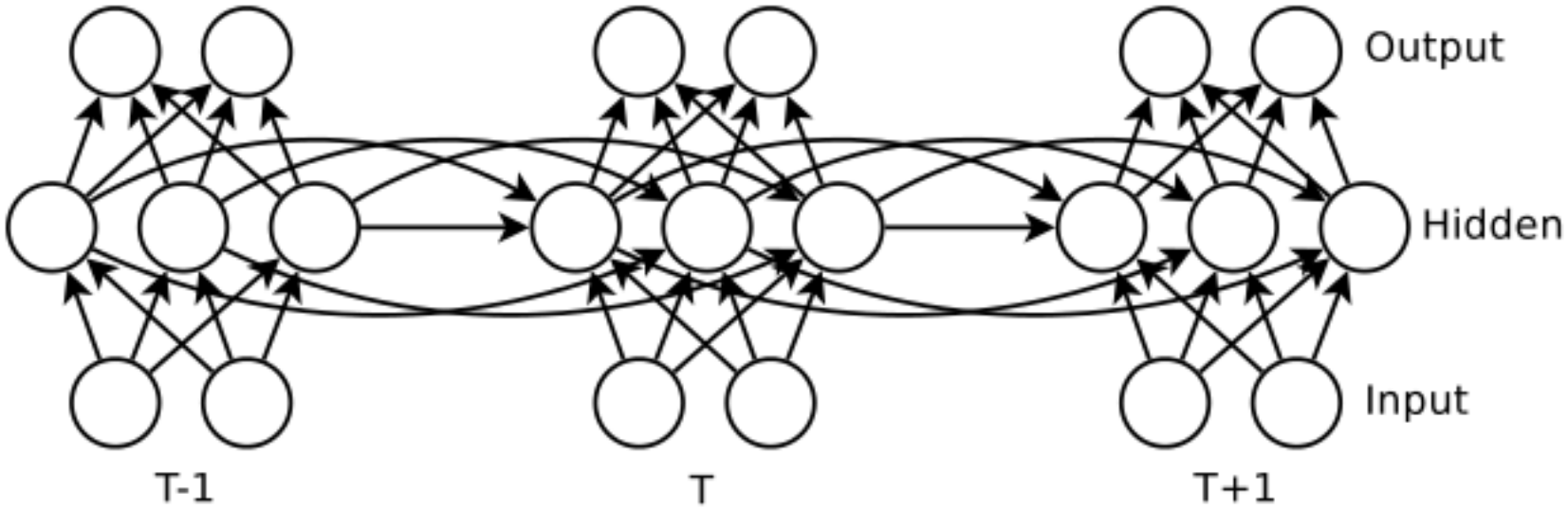
Глубокие нейронные сети

- Глубокие свёрточные сети – это state of the art в компьютерном зрении и обработке изображений.
- И они были таковым с начала 1990х. Но теперь мы будем обучать filter bank автоматически, а не подбирать руками.

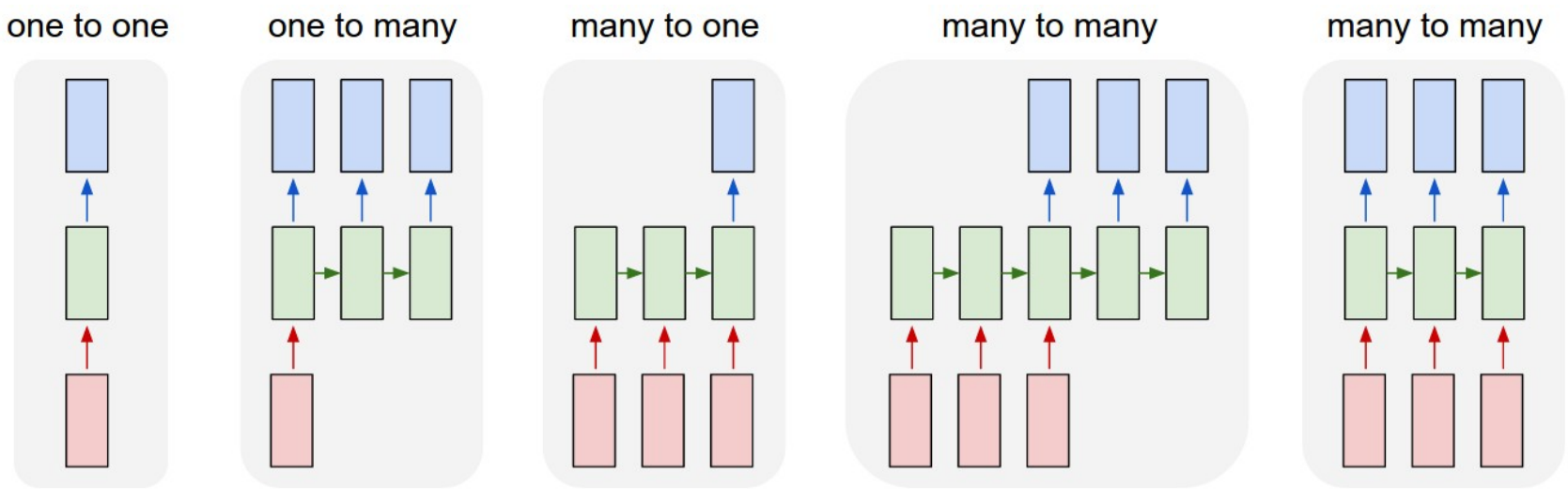


Глубокие нейронные сети

- Рекуррентная нейронная сеть (RNN) – это «очень глубокая» сеть, которая последовательно раскручивает цепочку входов.



От обычной классификации к RNN



Слева направо: фиксированный вход в фиксированный выход (классификация), выход в виде последовательности (image captioning), вход в виде последовательности (sentiment analysis), и то и другое (машинный перевод).

Deep learning в распознавании речи

- Первым большим успехом глубоких сетей было распознавание речи.
- В распознавании речи обычно использовали HMM (скрытые марковские модели) с GMM (смеси гауссианов) в качестве моделей наблюдаемых.
- Но в последнее время перешли практически полностью на DNN.
- Apple Siri, Google Now, Microsoft, IBM – все сейчас на DNN.

Deep learning в распознавании речи

- Для речи используют Gaussian–Bernoulli RBM (GRBM):

$$E(\boldsymbol{v}, \boldsymbol{h}) = \sum_i \frac{1}{2\sigma_i^2}(v_i - a_i)^2 - \sum_j b_j h_j - \sum_{i,j} \frac{v_i}{\sigma_i} h_j w_{ij},$$

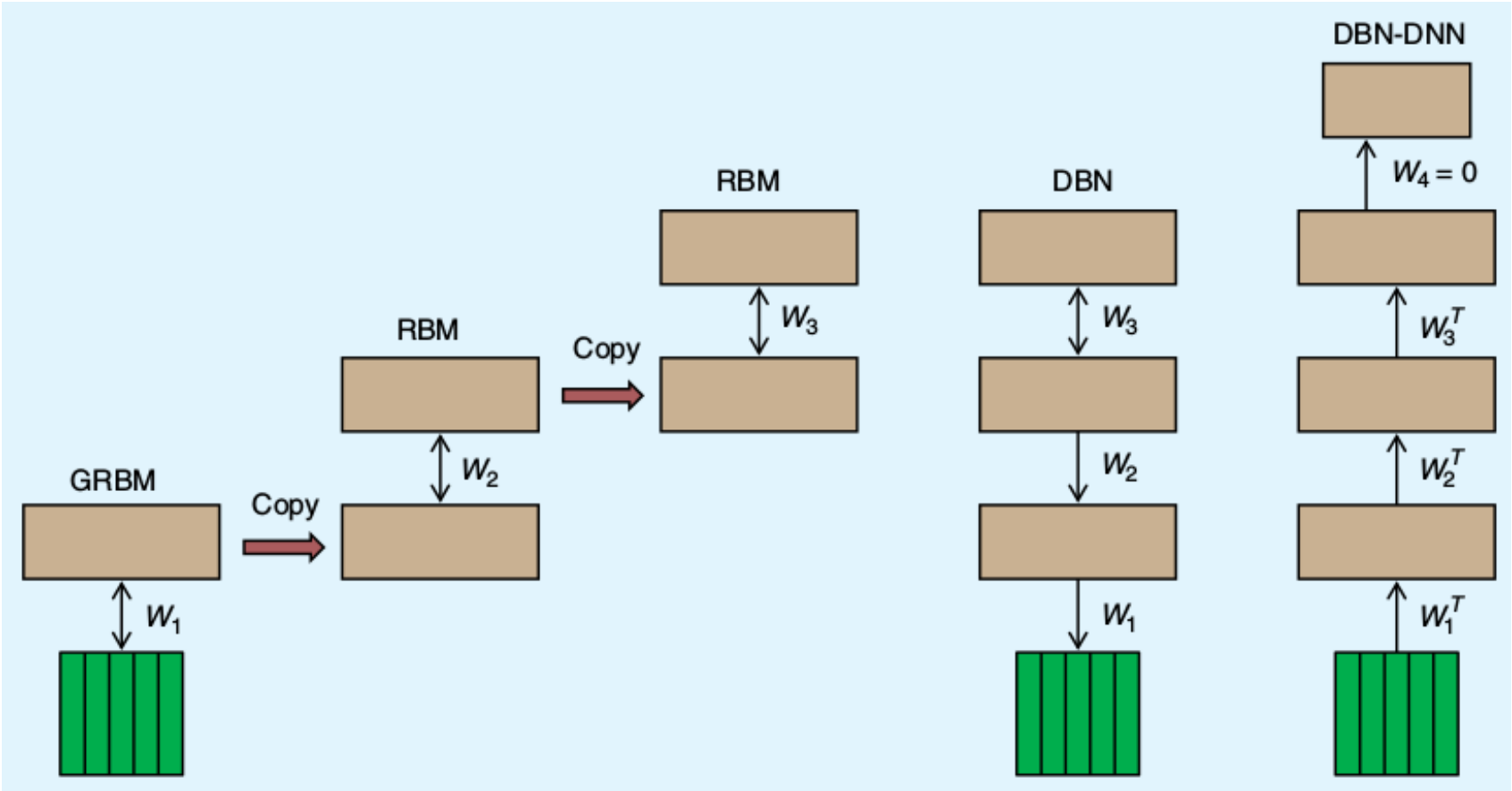
$$p(h_j \mid \boldsymbol{v}) = \sigma(b_j + \sum_i \frac{v_i}{\sigma_i} h_j w_{ij}),$$

$$p(v_j \mid \boldsymbol{h}) = \mathcal{N}(a_i + \sigma_i \sum_j h_j w_{ij}, \sigma_i^2).$$

- И дальше на ней строят глубокую сеть (DBN) из обычных бинарных RBM на следующих уровнях.

Deep learning в распознавании речи

- Обучение DBN для речи:



Deep learning в распознавании речи

- И это всё использовалось для распознавания фонем.

METHOD	PER
CD-HMM [26]	27.3%
AUGMENTED CONDITIONAL RANDOM FIELDS [26]	26.6%
RANDOMLY INITIALIZED RECURRENT NEURAL NETS [27]	26.1%
BAYESIAN TRIPHONE GMM-HMM [28]	25.6%
MONOPHONE HTMS [29]	24.8%
HETEROGENEOUS CLASSIFIERS [30]	24.4%
MONOPHONE RANDOMLY INITIALIZED DNNs (SIX LAYERS) [13]	23.4%
MONOPHONE DBN-DNNs (SIX LAYERS) [13]	22.4%
MONOPHONE DBN-DNNs WITH MMI TRAINING [31]	22.1%
TRIPHONE GMM-HMMs DT W/ BMMI [32]	21.7%
MONOPHONE DBN-DNNs ON FBANK (EIGHT LAYERS) [13]	20.7%
MONOPHONE MCRBM-DBN-DNNs ON FBANK (FIVE LAYERS) [33]	20.5%

Deep learning в распознавании речи

- Чтобы делать распознавание речи (слов, предложений), надо добавить ещё уровень с контекстом. Можно добавлять просто HMM.

		WER	
MODELING TECHNIQUE	**#PARAMS [10^6]**	**HUB5'00-SWB**	**RT03S-FSH**
GMM, 40 MIX DT 309H SI	29.4	23.6	27.4
NN 1 HIDDEN-LAYER × 4,634 UNITS	43.6	26.0	29.4
+ 2 × 5 NEIGHBORING FRAMES	45.1	22.4	25.7
DBN-DNN 7 HIDDEN LAYERS × 2,048 UNITS	45.1	17.1	19.6
+ UPDATED STATE ALIGNMENT	45.1	16.4	18.6
+ SPARSIFICATION	15.2 NZ	16.1	18.5
GMM 72 MIX DT 2000H SA	102.4	17.1	18.6

Deep learning в распознавании речи

- И если добавить ещё данных, то становится ещё лучше.

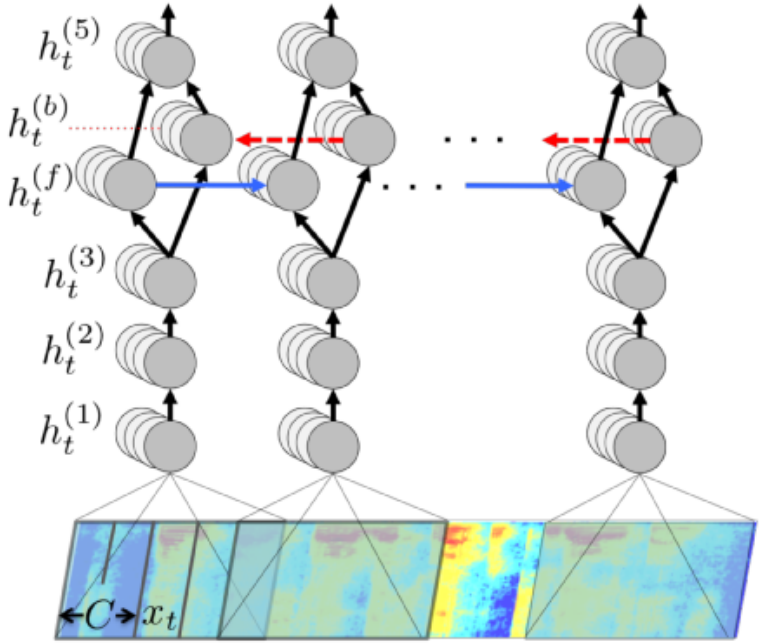
TASK	HOURS OF TRAINING DATA	DNN-HMM	GMM-HMM WITH SAME DATA	GMM-HMM WITH MORE DATA
SWITCHBOARD (TEST SET 1)	309	18.5	27.4	18.6 (2,000 H)
SWITCHBOARD (TEST SET 2)	309	16.1	23.6	17.1 (2,000 H)
ENGLISH BROADCAST NEWS	50	17.5	18.8	
BING VOICE SEARCH (SENTENCE ERROR RATES)	24	30.4	36.2	
GOOGLE VOICE INPUT	5,870	12.3		16.0 (>> 5,870 H)
YOUTUBE	1,400	47.6	52.3	

Deep learning в распознавании речи

- А можно добавить рекуррентные нейронные сети (RNN).
- Теперь строим несколько первых уровней из autoencoders (не рекуррентных), а потом на последнем уровне уже добавляем RNN, которая хранит текущее состояние и историю.
- Для регуляризации используют dropout.

Deep learning в распознавании речи

- (Hannun et al., 2014): система Deep Speech.



Deep learning в распознавании речи

- Результаты превосходят всё, что было до этого.

Model	SWB	CH	Full
Vesely et al. (GMM-HMM BMMI) [44]	18.6	33.0	25.8
Vesely et al. (DNN-HMM sMBR) [44]	12.6	24.1	18.4
Maas et al. (DNN-HMM SWB) [28]	14.6	26.3	20.5
Maas et al. (DNN-HMM FSH) [28]	16.0	23.7	19.9
Seide et al. (CD-DNN) [39]	16.1	n/a	n/a
Kingsbury et al. (DNN-HMM sMBR HF) [22]	13.3	n/a	n/a
Sainath et al. (CNN-HMM) [36]	11.5	n/a	n/a
Soltau et al. (MLP/CNN+I-Vector) [40]	**10.4**	n/a	n/a
Deep Speech SWB	20.0	31.8	25.9
Deep Speech SWB + FSH	12.6	**19.3**	**16.0**

System	Clean (94)	Noisy (82)	Combined (176)
Apple Dictation	14.24	43.76	26.73
Bing Speech	11.73	36.12	22.05
Google API	6.64	30.47	16.72
wit.ai	7.94	35.06	19.41
Deep Speech	**6.56**	**19.06**	**11.85**

Deep learning и музыка

- Или музыка: (Boulanger-Lewandowski et al., 2012) моделируют полифоническую музыку, предсказывая ноты.

Model	Piano-midi.de		Nottingham		MuseData		JSB chorales	
	LL	ACC %	LL	ACC %	LL	ACC %	LL	ACC %
Random	-61.00	3.35	-61.00	4.53	-61.00	3.74	-61.00	4.42
1-Gram (Add-p)	-27.64	4.85	-5.94	22.76	-19.03	6.67	-12.22	16.80
1-Gram (Gaussian)	-10.79	6.04	-5.30	21.31	-10.15	7.87	-7.56	17.41
Note 1-Gram	-11.05	5.80	-10.25	19.87	-11.51	7.72	-11.06	15.25
Note 1-Gram (IID)	-12.90	2.51	-16.24	3.56	-14.06	2.82	-15.93	3.51
GMM	-15.84	5.08	-7.87	22.62	-12.20	7.37	-11.90	15.84
RBM	-10.17	5.63	-5.25	5.81	-9.56	8.19	-7.43	4.47
NADE	-10.28	5.82	-5.48	22.67	-10.06	7.65	-7.19	17.88
Previous + Gaussian	-12.48	25.50	-8.41	55.69	-12.90	25.93	-19.00	18.36
N-Gram (Add-p)	-46.04	7.42	-6.50	63.45	-35.22	10.47	-29.98	24.20
N-Gram (Gaussian)	-12.22	10.01	-3.16	65.97	-10.59	16.15	-9.74	28.79
Note N-Gram	-7.50	26.80	-4.54	62.49	-7.91	26.35	-10.26	20.34
GMM + HMM	-15.30	7.91	-6.17	59.27	-11.17	13.93	-11.89	19.24
(Allan & Williams, 2005)	–	–	–	–	–	–	-9.24	16.32
(Lavrenko & Pickens, 2003)	-9.05	18.37	-5.44	55.34	-9.87	18.39	-8.78	22.93
MLP	-8.13	20.29	-4.38	63.46	-7.94	25.68	-8.70	30.41
RNN	-8.37	19.33	-4.46	62.93	-8.13	23.25	-8.71	28.46
RNN (HF)	-7.66	23.34	-3.89	66.64	-7.19	30.49	-8.58	29.41
RTRBM	-7.36	22.99	-2.62	75.01	-6.35	30.85	-6.35	30.17
RNN-RBM	**-7.09**	**28.92**	**-2.39**	**75.40**	-6.01	**34.02**	-6.27	**33.12**
RNN-NADE	-7.48	20.69	-2.91	64.95	-6.74	24.91	-5.83	32.11
RNN-NADE (HF)	**-7.05**	23.42	**-2.31**	71.50	**-5.60**	32.60	**-5.56**	32.50

Deep learning для NLP

- Deep learning применяется для обработки естественного языка: обучаем распределённое представление (distributed representation) для каждого слова.
- (Collobert et al., 2011): в результате фактически без учителя обучаются представления, из которых выводятся части речи, распознавание именованных сущностей, обучение семантики и т.п.
- Здесь тоже в каком-то смысле свёрточные сети (сворачиваем некоторое окно в тексте), но не такие глубокие.

DL в NLP

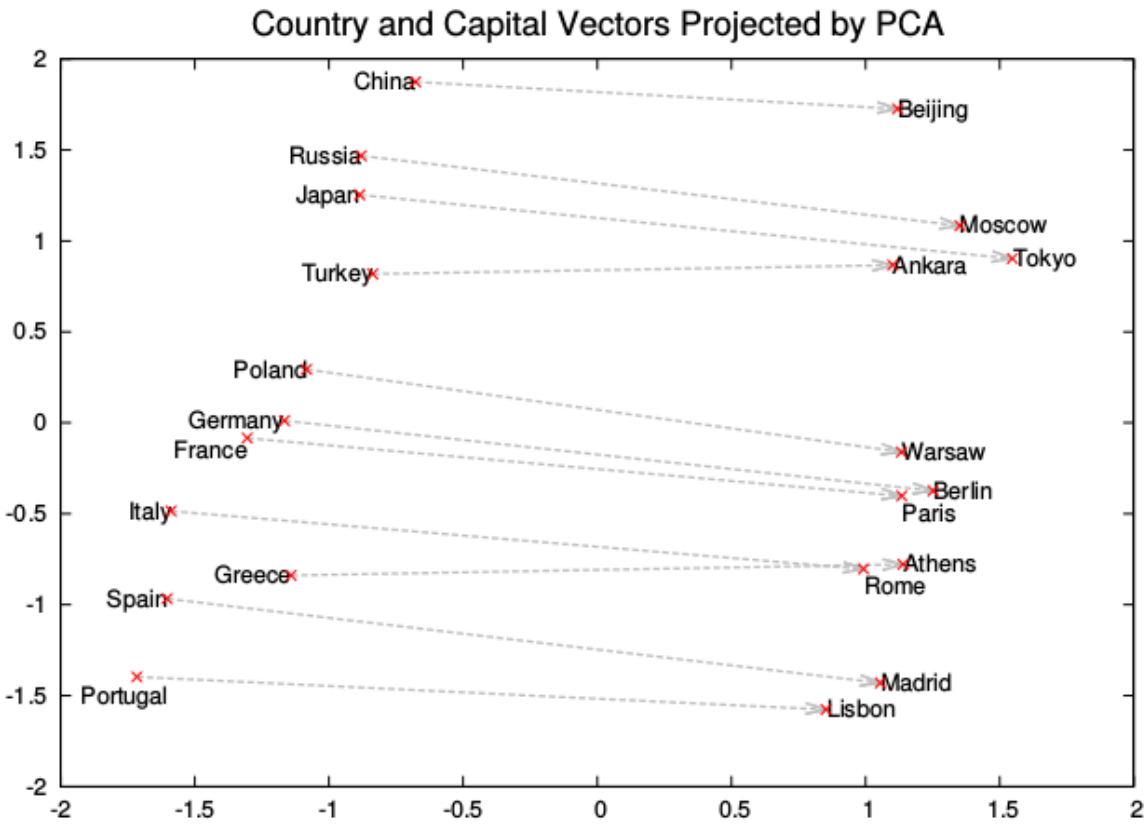
- Как применять глубокое обучение для обработки естественного языка?
- Это тоже неструктурированные данные, из которых нужно что-то извлечь; и задачи те же:
  - автоматическое выделение признаков;
  - задачи, грубо говоря, классификации (части речи, грамматический парсинг и т.п.);
  - конечная цель – распознавание семантики происходящего.

DL в NLP

- (Mikolov et al., 2013):
  - обучаем вложение слов в n-мерное пространство признаков, предсказывая слово из контекста (cbow) или контекст из слова (skip-gram);
  - модель skip-gram: по слову пытаемся предсказать его контекст;
  - модель cbow (continuous bag of words): по контексту предсказываем слово;
  - положительные примеры – из данных; отрицательные – берём n-граммы из данных и вставляем случайные слова.

DL в NLP

- На выходе обучаются линейные зависимости вида $\text{king} - \text{man} + \text{woman} \approx \text{queen}$.



DL в NLP

- Примеры ближайших соседей (в сравнении с другой моделью):

	NEG-15 with 10^{-5} subsampling	HS with 10^{-5} subsampling
Vasco de Gama	Lingsugur	Italian explorer
Lake Baikal	Great Rift Valley	Aral Sea
Alan Bean	Rebbeca Naomi	moonwalker
Ionian Sea	Ruegen	Ionian Islands
chess master	chess grandmaster	Garry Kasparov

- Примеры ближайших соседей к запросам-словосочетаниям:

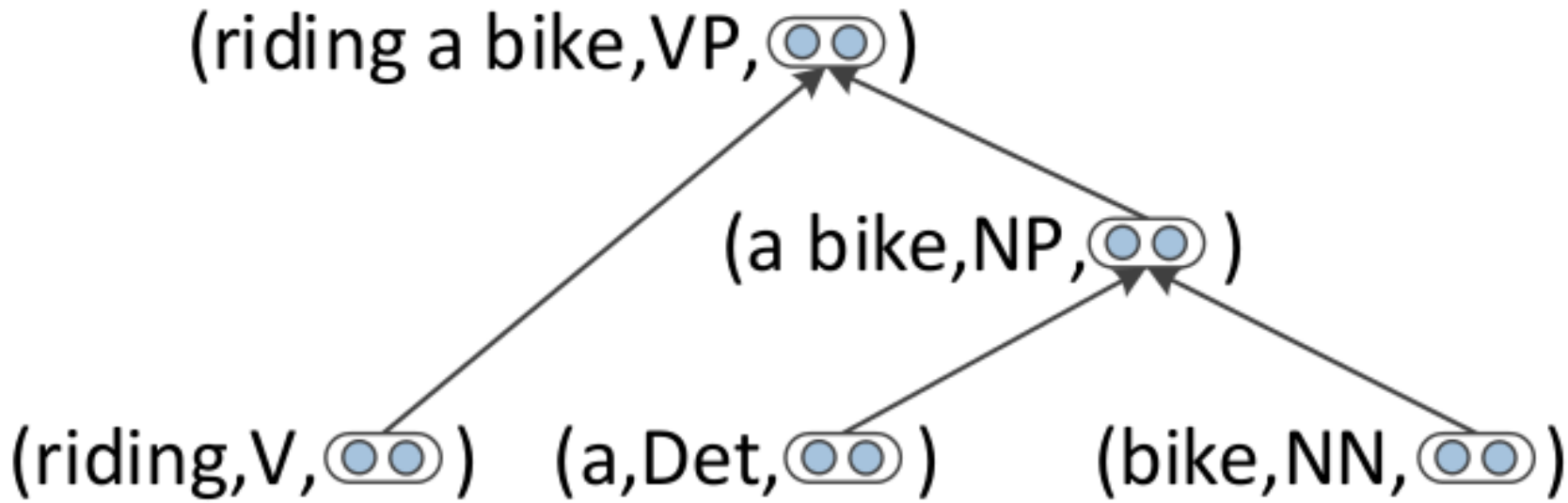
Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

DL в NLP

- Word embeddings – это важно:
  - (Wang, Manning, 2013): сравнивают обычный CRF и SLNN (sentence-level likelihood neural nets) для named entity recognition на стандартных датасетах – получается, что для дискретного представления CRF ничем не хуже, но на word embeddings по методу Collobert et al. SLNN сильно выигрывают.

Рекурсивные нейронные сети

- Пример: задача синтаксического парсинга предложений.
- Т.е. мы хотим из предложения получить что-то в духе

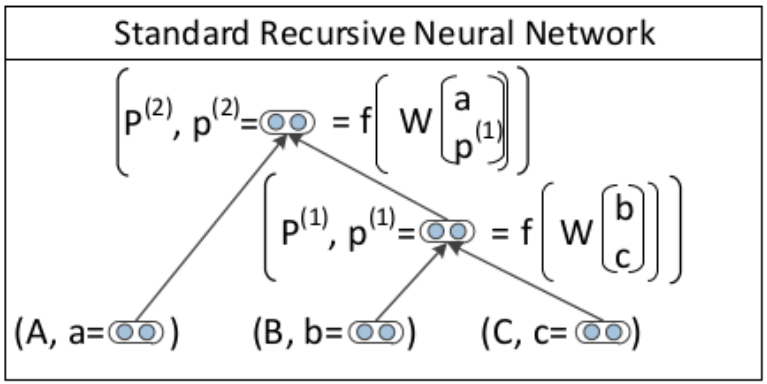


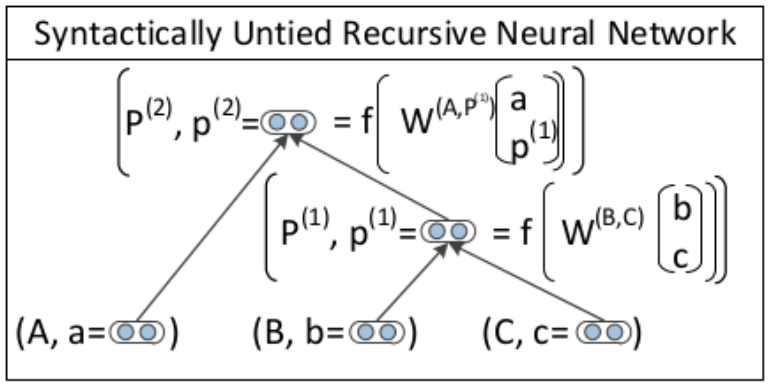
Рекурсивные нейронные сети

- Воспользуемся word embeddings, это понятно.
- Но теперь проблема: предложения разной длины, т.е. длина входа неизвестна. Это можно решить таким образом:
  - будем строить дерево шаг за шагом;
  - на каждом шаге будем нейронной сетью предсказывать, насколько хорошо ли эти листья сейчас объединить;
  - а веса нейронной сети будем использовать всё время одни и те же, т.е. дерево представляет функцию $f(Wf(Wf(\dots f(Wx)\dots)))$.
- Это и называется рекурсивными нейронными сетями.

Рекурсивные нейронные сети

- (Socher, Bauer, Manning, Ng, 2013): синтаксический парсинг: как построить дерево парсинга (функция ошибки смотрит, насколько дерево отличается от правильного); мы на самом деле обучаем одну или несколько матриц на все узлы.





Рекурсивные нейронные сети

- (Bordes, Weston, Collobert, Bengio, AAAI 2011): structured embeddings:
  - рассмотрим базу знаний как граф отношений;
  - обучим word embeddings – представления слов в $\mathbb{R}^d$;
  - в пространстве $\mathbb{R}^d$ отношение – это некая метрика похожести;
  - мы моделируем её двумя матрицами преобразований:

$$\text{sim}_k(E_i, E_j) = \left\| R_k^{lhs} E_i - R_k^{rhs} E_j \right\|_p$$

  и обучаем матрицы.

Рекурсивные нейронные сети

- Получаются интересные результаты – ниже примеры, из которых исключили всё, что было в обучающем наборе:

e^l	_everest_1	_brain_1
r	_part_of	_has_part
e^r	*_north_vietnam_1*	*_subthalamic_nucleus_1*
	_hindu_kush_1	*_cladode_1*
	_karakoram_1	*_subthalamus_1*
	_federal_2	*_fluid_ounce_1*
	_burma_1	*_sympathetic_nervous_system_1*
e^l	*_judgement_3*	*_thing_13*
	_delayed_action_1	*_transfer_5*
	_experience_5	*_situation_1*
	_bawl_out_1	*_illness_1*
	_carry_over_1	*_cognition_1*
r	_type_of	_has_instance
e^r	_deciding_1	_language_1

Рекурсивные нейронные сети

- Сами embeddings – ближайшие соседи:

_lawn_tennis_1	_artist_1	_field_1	_field_2	_pablo_picasso	_audrey
_badminton_1	*_critic_1*	*_yard_9*	*_universal_set_1*	*_lin_liang*	*_wil_va*
_squash_4	*_part_7*	*_picnic_area_1*	*_diagonal_3*	*_zhou_fang*	*_signe*
_baseball_1	*_singer_1*	*_center_stage_1*	*_analysis_situs_1*	*_wu_guanzhong*	*_joyce_*
_cricket_2	*_prospector_1*	*_range_11*	*_positive_10*	*_paul_cezanne*	*_greta_*
_hockey_2	*_condition_3*	*_eden_1*	*_oblique_3*	*_yves_klein*	*_ingrid_*

WordNet data

- Извлечение знаний:

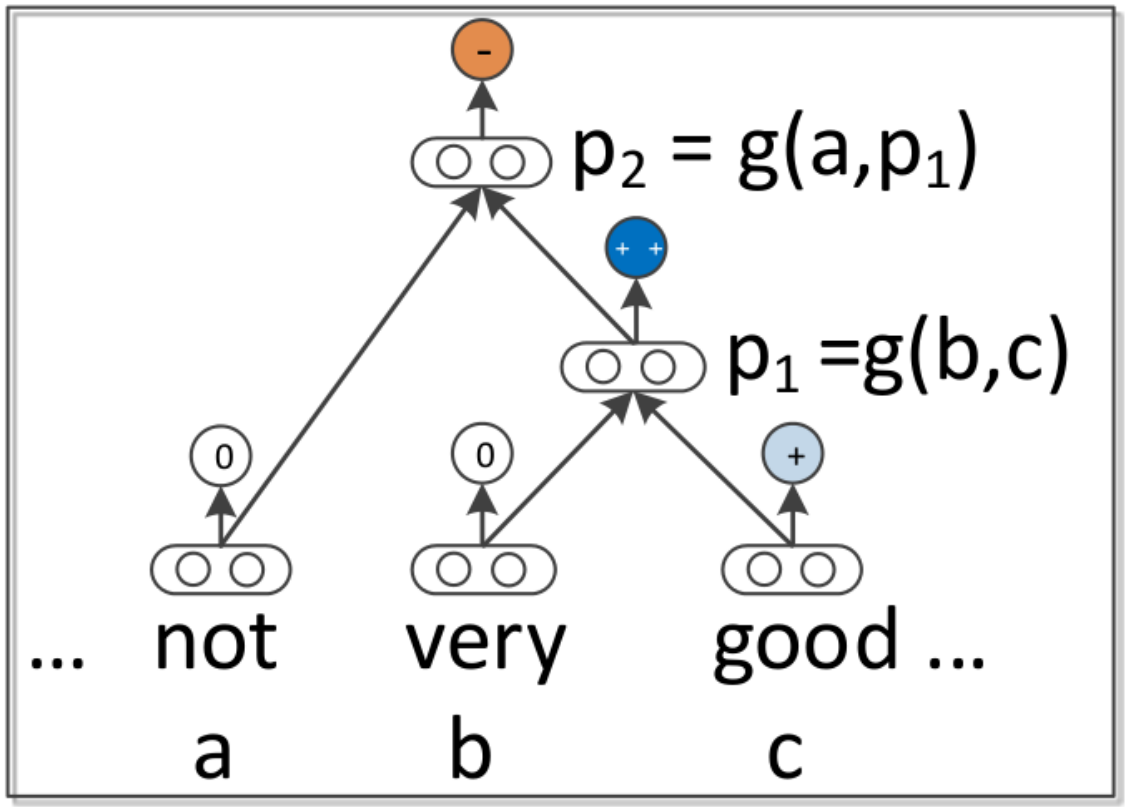
e^l	people				
r	build	destroy	won	suffer	control
e^r	*livelihoods*	*icons*	*emmy*	*sores*	*rocket*
	homes	*virtue*	*award*	*agitation*	*stores*
	altars	*donkeys*	*everything*	*treatise*	*emotions*
	houses	*cowboy*	*standings*	*eczema*	*spending*
	ramps	*chimpanzees*	*pounds*	*copd*	*fertility*

Рекурсивные нейронные сети

- (Bordes, Glorot, Weston, Bengio, AISTATS 2012) – open-text semantic parsing:
  - модель (lhs, relation, rhs);
  - концепт – вектор, отношение – две матрицы; матрица работает как оператор;
  - ранжируем по энергии, низкая на тренировочных примерах, высокая вне.

Sentiment analysis

- Пример применения: sentiment analysis (там ещё LSTM).



Глубокое обучение и NLP

- Но важно заметить:
  - люди занимаются NLP давно, было много методов;
  - когда глубокие сети пришли в распознавание речи, они сразу победили всех и намного;
  - то же самое было с распознаванием изображений;
  - в NLP такого (пока) нет! пока скорее «мы делаем примерно так же или чуть лучше, чем вы, но ничего не зная о лингвистике и с нуля»;
  - это область, которая продолжает очень бурно развиваться, и я всех приглашаю попробовать ей заняться.

Пример задачи

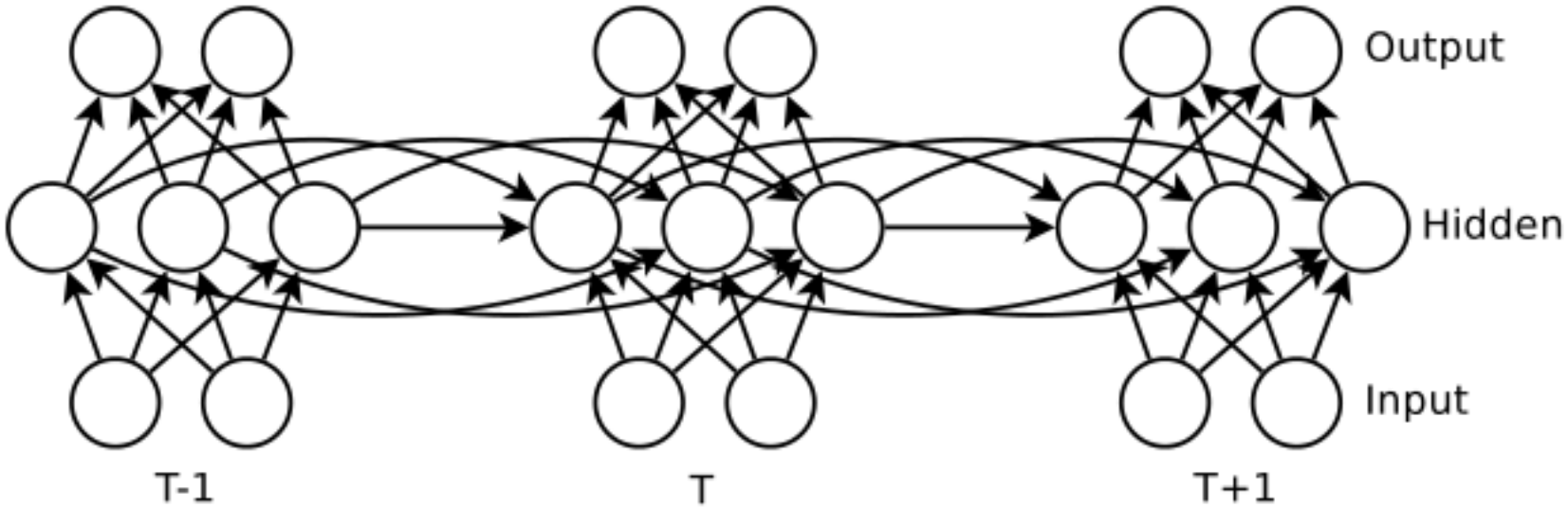
- Пример сложной задачи: question answering. IBM Watson был уже давно, но на самом деле в плане понимания языка всё пока не очень хорошо.
- На вот таких задачах лучший результат порядка 20-25% ошибки:

```
Sam walks into the kitchen.
Sam picks up an apple.
Sam walks into the bedroom.
Sam drops the apple.
Q: Where is the apple?
A. Bedroom
```

```
Brian is a lion.
Julius is a lion.
Julius is white.
Bernhard is green.
Q: What color is Brian?
A. White
```

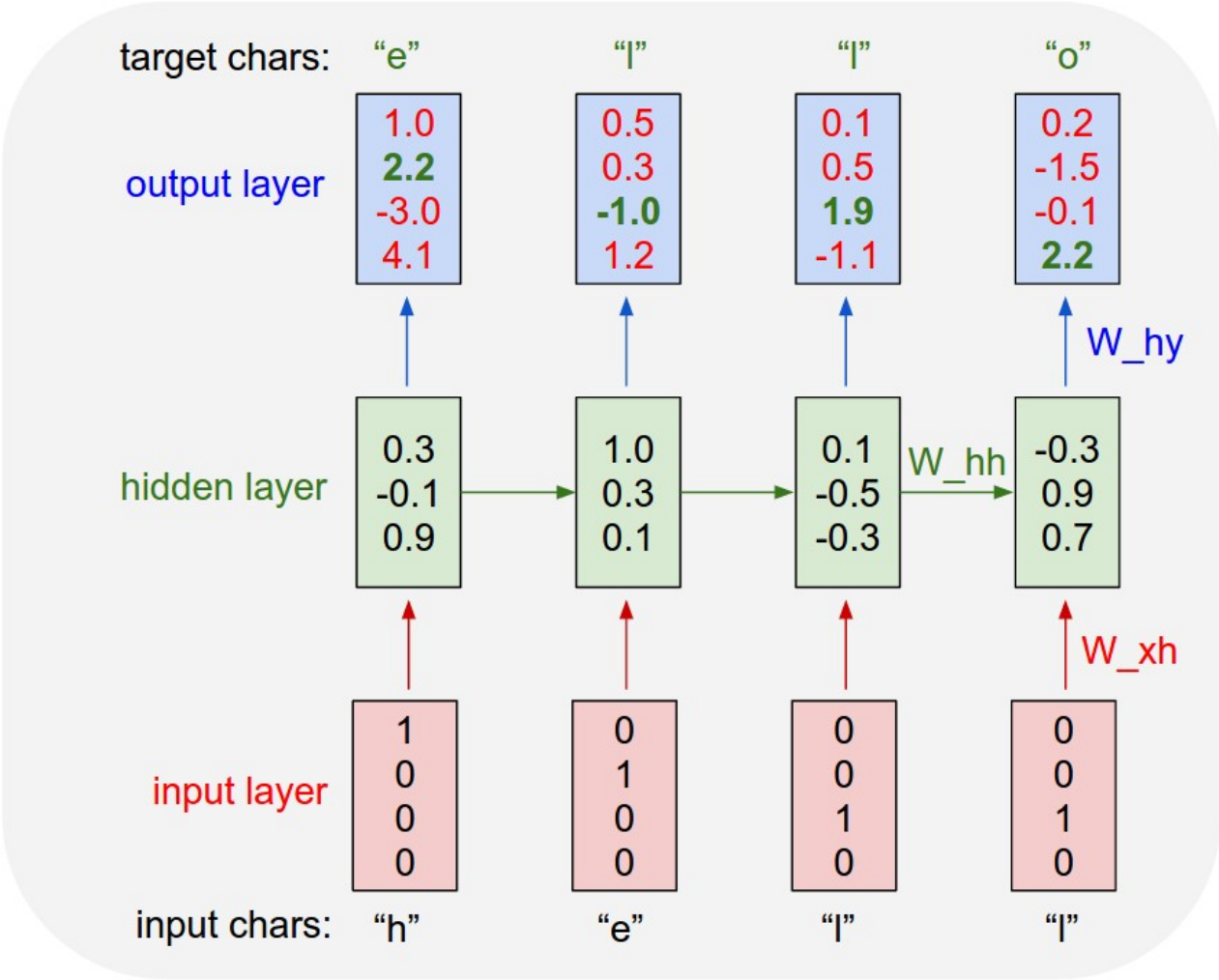
Генерируем язык символ за символом

- (Sutskever, Martens, Hinton, 2011): рекуррентные нейронные сети генерируют текст *символ за символом*.
- RNN – это «очень глубокая» сеть, которая последовательно раскручивает цепочку входов.



Генерируем язык символ за символом

- RNN:



Генерируем язык символ за символом

- RNN: по $(x_1, \ldots, x_T)$ вычисляем

$$h_t = \tanh(W_{hx} x_t + W_{hh} h_{t-1} + b_h),$$
$$o_t = W_{oh} h_t + b_o,$$

  где все веса не зависят от времени.
- Градиенты легко подсчитать (backpropagation through time), но обучить сложно: очень нестабильно, saturation мешает, vanishing/exploding gradients.

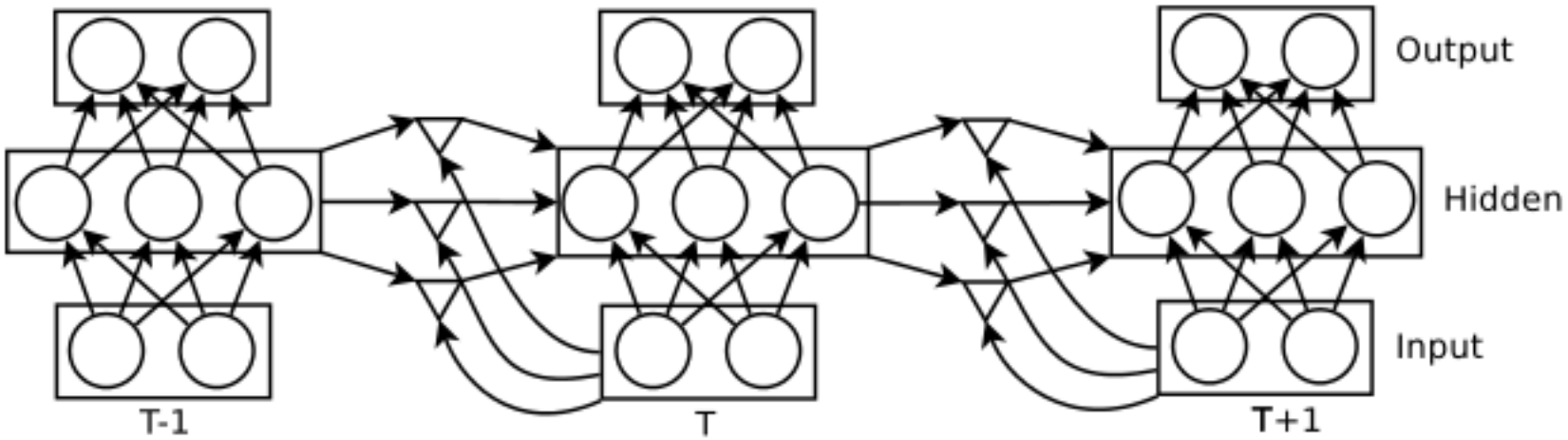
Генерируем язык символ за символом

- (Martens, 2010; Martens, Sutskever, 2011): новый алгоритм второго порядка для RNN, hessian-free optimization.
- Он позволяет обучать относительно большие RNN.
- Но на самом деле в языке есть более сложные зависимости: нужно, чтобы следующее состояние определялось одновременно скрытым состоянием и текущим символом (основа глагола + i предсказывают n).

Генерируем язык символ за символом

- Получается Multiplicative RNN (MRNN):

$$
\begin{aligned}
f_t &= \text{diag}(W_{fx} x_t) W_{fh} h_{t-1}, \\
h_t &= \tanh(W_{hf} f_t + W_{hh} h_{t-1} + b_h), \\
o_t &= W_{oh} h_t + b_o.
\end{aligned}
$$



Генерируем язык символ за символом

- И получается модель, которая обучает символ за символом и при этом может, например, закрывать скобки.
- Вот пример сгенерированного текста (просто сэмплирование из модели):

Recurrent network with the Stiefel information for logistic regression methods. Along with either of the algorithms previously (two or more skewprecision) is more similar to the model with the same average mismatched graph.

Генерируем язык символ за символом

- Вот что получится, если обучиться на Шекспире:

PANDARUS:

Alas, I think he shall be come approached and the day

When little srain would be attain'd into being never fed,

And who is but a chain and subjects of his death,

I should not sleep.

Second Senator:

They are away this miseries, produced upon my soul,

Breaking and strongly should be buried, when I perish

The earth and thoughts of many states.

DUKE VINCENTIO:

Well, your wit is in the care of side and that.

Пример

- А можно попросить модель закончить фразу: генерировать фразу символ за символом (!) после заданного начала, учитывая его как контекст.

The meaning of life is...

Пример

- А можно попросить модель закончить фразу: генерировать фразу символ за символом (!) после заданного начала, учитывая его как контекст.

The meaning of life is the tradition of the ancient human reproduction: it is less favorable to the good boy for when to remove her bigger.

Thank you!

Спасибо за внимание!