курс «Глубокое обучение»

Анализ текстов

Александр Дьяконов

24 октября 2022 года

План

Задачи с текстами, данные, понимания языка (Language Understanding)

Свёрточные модели для текста

Модель seq2seq, обобщение

Механизм внимания, виды

Представления слов: w2v, fasttext, Glove

Контекстные представления: ELMO

Области исследований

NLP – всё, что связано с обработкой текстов

NLU – всё, что связано с пониманием текстов (текст → действие / …)

NLG – всё что связано с генерацией текста (… → текст)

Задачи с текстами

текст → метка (классификация)

определение темы / настроения / автора

определение тональности

текст → метки (тегирование)

определение тегов

разметка на части речи

текст → текст (seq2seq)

машинный перевод

аннотирование

чат-бот

продолжение текста

генерация по контенту

текст, текст → текст

ответы на вопросы

справочная / экспертная система

… → текст

описание изображения

моделирование / генерация языка

текст → …

parse tree по предложению

генерация объектов по описанию

Термины

токен – элемент последовательности (слово, несколько букв)

словарь – множество допустимых токенов (модель принимает/генерирует только их)

все перечисленные задачи с текстами (кроме последней) – классификация

классов может быть очень много = число всех токенов

Проблемы

0) текст – сигнал (к счастью, дискретный) с разделителями
(например, пробелами между словами)

1) один и тот же смысл передаётся по-разному / синонимы

- haha
- hahahahahahahaha
- haaaahaaaa
- lol
- rotflmao
- lol!!!!!!!!!!!!!
- wow that is big
- that is biiiiiig
- that. is. big.
- waaaaaaay big

2) многозначность / омонимы

«Эти типы стали есть в цехе»

Проблемы

3) динамичность языка, новые слова, сленг

е-комерция, айпадный, зафрендить и т.п.

4) устойчивые выражения, профессиональный сленг, контекст

«бабье лето», «на эпсилон старше меня»

«Петя увидел Васю, неудивительно, **он** был очень зоркий/заметный»

5) разметка, как правило, ручная

6) при попытках признаковой постановки
большие разреженные пространства

7) текст больше чем последовательность предложений

контекст, порядок изложения, расстановка акцентов

История NLP

- предобработка (регулярки, стемминг, лемматизация)
- мешок слов (нормировки), N-граммные модели
- векторные представления слов
- языковые модели
- трансферное обучение (перенос обучения)
- мультиязычность
- мультимодальность
- графы знаний

Данные – есть много открытых данных https://arxiv.org/pdf/2003.01200.pdf

Task	Dataset	Link
Machine Translation	WMT 2014 EN-DE WMT 2014 EN-FR	http://www-lium.univ-lemans.fr/~schwenk/cslm_joint_paper/
Text Summarization	CNN/DM Newsroom DUC Gigaword	https://cs.nyu.edu/~kcho/DMQA/ https://summari.es/ https://www-nlpir.nist.gov/projects/duc/data.html https://catalog.ldc.upenn.edu/LDC2012T21
Reading Comprehension Question Answering Question Generation	ARC CliCR CNN/DM NewsQA RACE SQuAD Story Cloze Test NarativeQA Quasar SearchQA	http://data.allenai.org/arc/ http://aclweb.org/anthology/N18-1140 https://cs.nyu.edu/~kcho/DMQA/ https://datasets.maluuba.com/NewsQA http://www.qizhexie.com/data/RACE_leaderboard https://rajpurkar.github.io/SQuAD-explorer/ http://aclweb.org/anthology/W17-0906.pdf https://github.com/deepmind/narrativeqa https://github.com/bdhingra/quasar https://github.com/nyu-dl/SearchQA
Semantic Parsing	AMR parsing ATIS (SQL Parsing) WikiSQL (SQL Parsing)	https://amr.isi.edu/index.html https://github.com/jkkummerfeld/text2sql-data/tree/master/data https://github.com/salesforce/WikiSQL
Sentiment Analysis	IMDB Reviews SST Yelp Reviews Subjectivity Dataset	http://ai.stanford.edu/~amaas/data/sentiment/ https://nlp.stanford.edu/sentiment/index.html https://www.yelp.com/dataset/challenge http://www.cs.cornell.edu/people/pabo/movie-review-data/
Text Classification	AG News DBpedia TREC 20 NewsGroup	http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html https://wiki.dbpedia.org/Datasets https://trec.nist.gov/data.html http://qwone.com/~jason/20Newsgroups/
Natural Language Inference	SNLI Corpus MultiNLI SciTail	https://nlp.stanford.edu/projects/snli/ https://www.nyu.edu/projects/bowman/multinli/ http://data.allenai.org/scitail/
Semantic Role Labeling	Proposition Bank OneNotes	http://propbank.github.io/ https://catalog.ldc.upenn.edu/LDC2013T19

IR-based QA

Stanford Question Answering Dataset (SQuAD) / SQuAD2.0

https://rajpurkar.github.io/SQuAD-explorer/

NewsQA

WikiQA

CuratedTREC

WebQuestions

WikiMovies

Russian: SberQUAD

IR-based QA

Dataset	Example	Article / Paragraph
SQuAD	**Q**: How many provinces did the Ottoman empire contain in the 17th century? **A**: 32	**Article**: Ottoman Empire **Paragraph**: ... At the beginning of the 17th century the empire contained 32 provinces and numerous vassal states. Some of these were later absorbed into the Ottoman Empire, while others were granted various types of autonomy during the course of centuries.
CuratedTREC	**Q**: What U.S. state's motto is "Live free or Die"? **A**: New Hampshire	**Article**: Live Free or Die **Paragraph**: "Live Free or Die" is the official motto of the U.S. state of New Hampshire, adopted by the state in 1945. It is possibly the best-known of all state mottos, partly because it conveys an assertive independence historically found in American political philosophy and partly because of its contrast to the milder sentiments found in other state mottos.
WebQuestions	**Q**: What part of the atom did Chadwick discover?† **A**: neutron	**Article**: Atom **Paragraph**: ... The atomic mass of these isotopes varied by integer amounts, called the whole number rule. The explanation for these different isotopes awaited the discovery of the neutron, an uncharged particle with a mass similar to the proton, by the physicist James Chadwick in 1932. ...
WikiMovies	**Q**: Who wrote the film Gigli? **A**: Martin Brest	**Article**: Gigli **Paragraph**: Gigli is a 2003 American romantic comedy film written and directed by Martin Brest and starring Ben Affleck, Jennifer Lopez, Justin Bartha, Al Pacino, Christopher Walken, and Lainie Kazan.

Table 1: Example training data from each QA dataset. In each case we show an associated paragraph where distant supervision (DS) correctly identified the answer within it, which is highlighted.

https://arxiv.org/pdf/1704.00051.pdf

SQuAD 1.0 → SQuAD 2.0

недостатки первой версии:
ответы на все вопросы есть в пределах параграфа
(во второй версии есть вариант «нет ответа»)

	SQuAD 1.1	SQuAD 2.0
Train		
Total examples	87,599	130,319
Negative examples	0	43,498
Total articles	442	442
Articles with negatives	0	285
Development		
Total examples	10,570	11,873
Negative examples	0	5,945
Total articles	48	35
Articles with negatives	0	35
Test		
Total examples	9,533	8,862
Negative examples	0	4,332
Total articles	46	28
Articles with negatives	0	28

Table 2: Dataset statistics of SQuAD 2.0, compared to the previous SQuAD 1.1.

https://arxiv.org/pdf/1806.03822.pdf

Данные: RACE http://www.cs.cmu.edu/~glai1/data/race/

5 типов вопросов: word matching, paraphrasing, single-sentence reasoning, multi-sentence reasoning, insucient or ambiguous questions

Passage:
In a small village in England about 150 years ago, a mail coach was standing on the street. It didn't come to that village often. People had to pay a lot to get a letter. The person who sent the letter didn't have to pay the postage, while the receiver had to.
"Here's a letter for Miss Alice Brown," said the mailman.
" I'm Alice Brown," a girl of about 18 said in a low voice.
Alice looked at the envelope for a minute, and then handed it back to the mailman.
"I'm sorry I can't take it, I don't have enough money to pay it", she said.
A gentleman standing around were very sorry for her. Then he came up and paid the postage for her.
When the gentleman gave the letter to her, she said with a smile, " Thank you very much, This letter is from Tom. I'm going to marry him. He went to London to look for work. I've waited a long time for this letter, but now I don't need it, there is nothing in it."
"Really? How do you know that?" the gentleman said in surprise.
"He told me that he would put some signs on the envelope. Look, sir, this cross in the corner means that he is well and this circle means he has found work. That's good news."
The gentleman was Sir Rowland Hill. He didn't forgot Alice and her letter.
"The postage to be paid by the receiver has to be changed," he said to himself and had a good plan.
"The postage has to be much lower, what about a penny? And the person who sends the letter pays the postage. He has to buy a stamp and put it on the envelope." he said . The government accepted his plan. Then the first stamp was put out in 1840. It was called the "Penny Black". It had a picture of the Queen on it.

Questions:

1): The first postage stamp was made _.
A. in England B. in America C. by Alice D. in 1910

2): The girl handed the letter back to the mailman because _ .
A. she didn't know whose letter it was
B. she had no money to pay the postage
C. she received the letter but she didn't want to open it
D. she had already known what was written in the letter

3): We can know from Alice's words that _ .
A. Tom had told her what the signs meant before leaving
B. Alice was clever and could guess the meaning of the signs
C. Alice had put the signs on the envelope herself
D. Tom had put the signs as Alice had told him to

4): The idea of using stamps was thought of by _ .
A. the government
B. Sir Rowland Hill
C. Alice Brown
D. Tom

5): From the passage we know the high postage made _ .
A. people never send each other letters
B. lovers almost lose every touch with each other
C. people try their best to avoid paying it
D. receivers refuse to pay the coming letters

Answer: ADABC

Table 1: Sample reading comprehension problems from our dataset.

Свёрточные модели для текста

идея как в обработке n-грамм

$$\sigma\left(W\begin{bmatrix}x_1\\x_2\end{bmatrix}+b\right)$$

проблема – как работать с последовательностями произвольной длины

- RNN
- CNN + max-pooling (max over time pooling)

Свёрточные модели для текста

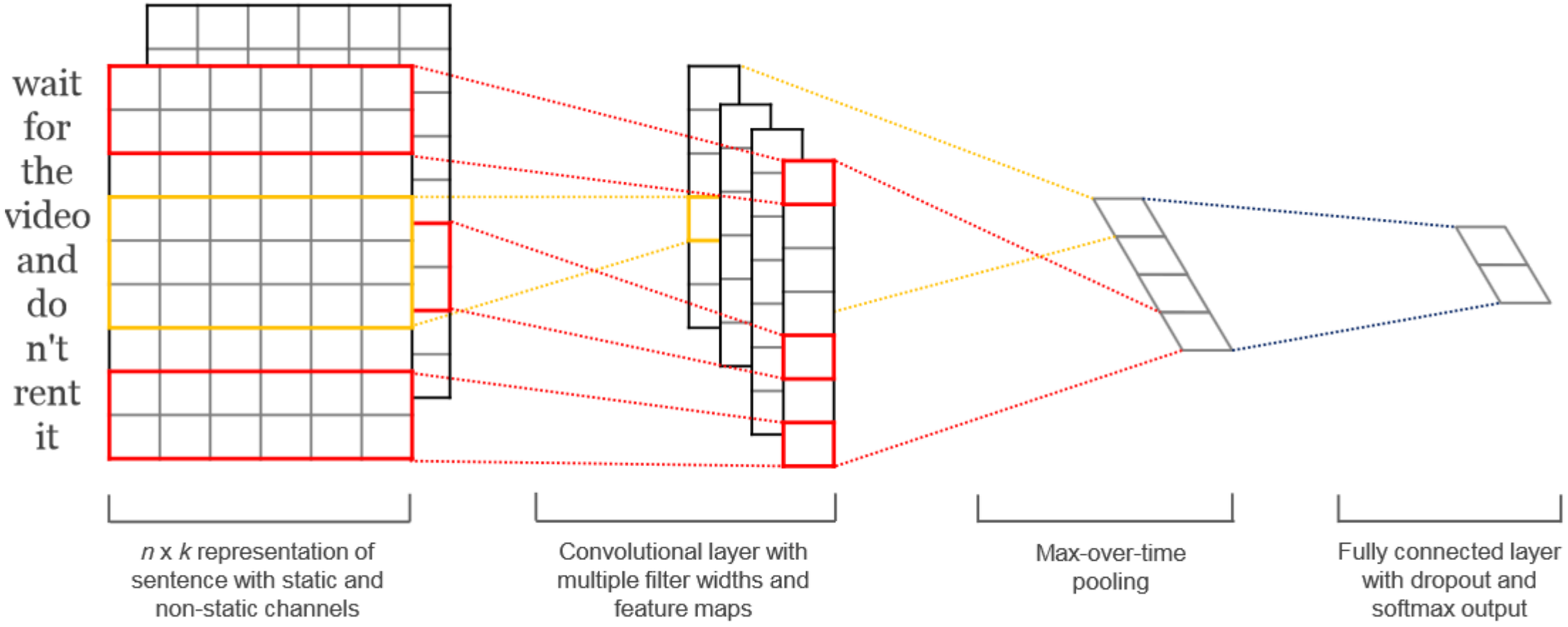


Figure 1: Model architecture with two channels for an example sentence.

два входных канала, т.к. были и эксперименты, когда один канал обучался…

Yoon Kim «Convolutional Neural Networks for Sentence Classification» // https://arxiv.org/abs/1408.5882

Свёрточные модели для текста

k – длина представления слова

n – фиксированная длина предложения

(если меньше – фиктивно набавляем)

Теперь уже наше предложение – матрица (как изображение)

подматрица векторизуется

$$c_i = \sigma\left(W_{1\times hk} \begin{bmatrix} x_i \\ \cdots \\ x_{i+h-1} \end{bmatrix}_{hk\times 1} + b \right) \in \mathbb{R}$$

на втором слое (если один фильтр):

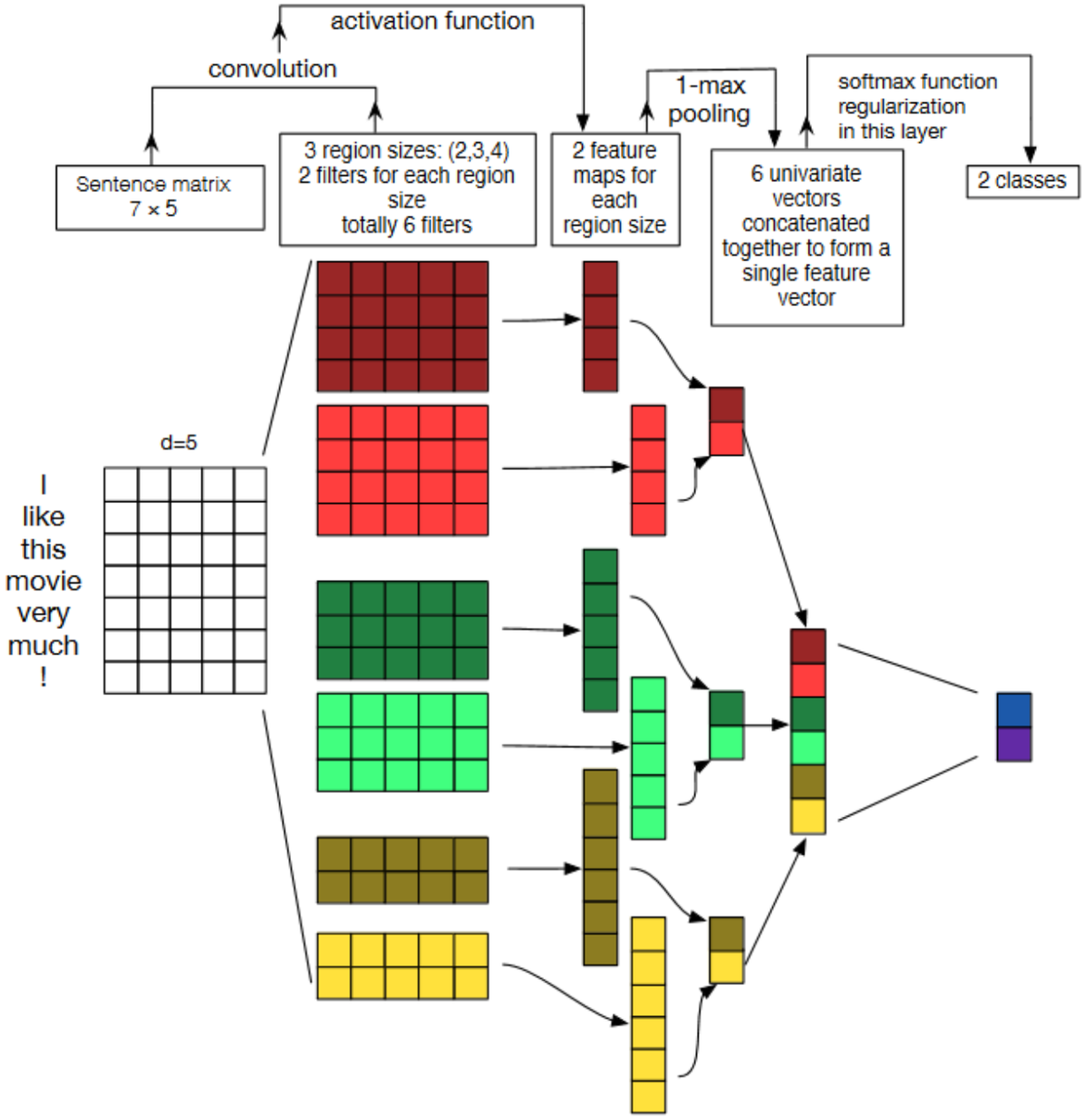
$$c = [c_1, \ldots, c_{n-h+1}] \in \mathbb{R}^{n-h+1}$$

max-pooling

$$\max(c) \in \mathbb{R}$$

для нескольких слоёв аналогично → полносвязную сеть

Свёрточные модели для текста: улучшения

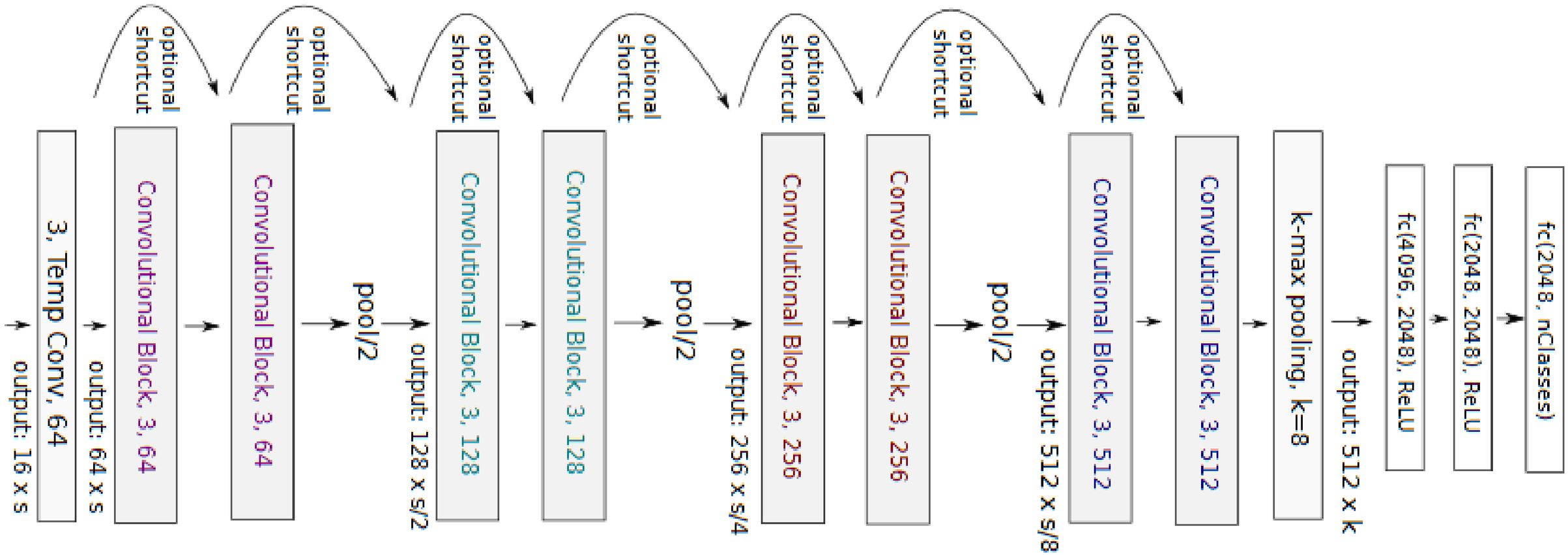


Свёртки с разной шириной

Разделить пулинг и конкатенацию

Ye Zhang, Byron Wallace A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification https://arxiv.org/abs/1510.03820

Very Deep Convolutional Networks for Text Classification: VD-CNN



хороши в посимвольном случае (character-level)

маленькие свёртки и маленькие пулинги (окно=3)

до 29 свёрточных слоёв

Very Deep Convolutional Networks for Text Classification: VD-CNN

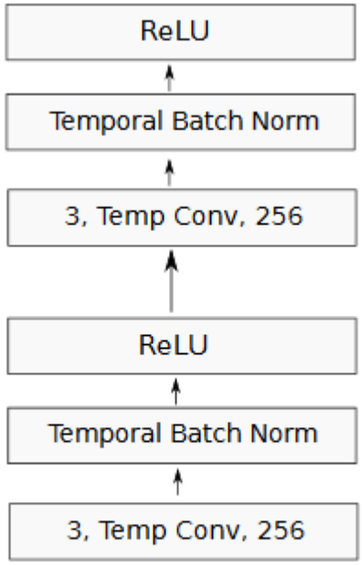


Figure 2: Convolutional block.

сделана по аналогии с VGG:

когда пространственный размер уменьшается в 2 раза

– число каналов×2

Temporal BN – для минибатча из m объектов на посл-ти длины s

статистики считаются по m·s слагаемым

Alexis Conneau, Holger Schwenk, Loïc Barrault, Yann Lecun «Very Deep Convolutional Networks for Text Classification» // https://arxiv.org/abs/1606.01781

Very Deep Convolutional Networks for Text Classification: VD-CNN

Depth	Pooling	AG	Sogou	DBP.	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. P.
9	Convolution	10.17	4.22	1.64	5.01	37.63	28.10	38.52	4.94
9	KMaxPooling	9.83	3.58	1.56	5.27	38.04	28.24	39.19	5.69
9	MaxPooling	9.17	3.70	1.35	4.88	36.73	27.60	37.95	4.70
17	Convolution	9.29	3.94	1.42	4.96	36.10	27.35	37.50	4.53
17	KMaxPooling	9.39	3.51	1.61	5.05	37.41	28.25	38.81	5.43
17	MaxPooling	8.88	3.54	1.40	4.50	36.07	27.51	37.39	4.41
29	Convolution	9.36	3.61	1.36	4.35	**35.28**	27.17	37.58	**4.28**
29	KMaxPooling	**8.67**	**3.18**	1.41	4.63	37.00	27.16	38.39	4.94
29	MaxPooling	8.73	3.36	**1.29**	**4.28**	35.74	**26.57**	**37.00**	4.31

Table 5: Testing error of our models on the 8 data sets. No data preprocessing or augmentation is used.

depth	without shortcut	with shortcut
9	37.63	40.27
17	36.10	39.18
29	35.28	36.01
49	37.41	36.15

Table 6: Test error on the Yelp Full data set for all depths, with or without residual connections.

- **глубина важна**
- **max-pool лучше всего**
- **эта модель лучше предыдущих**
- **прокидывание связей помогает на глубине**

Сравнение CNN vs RNN

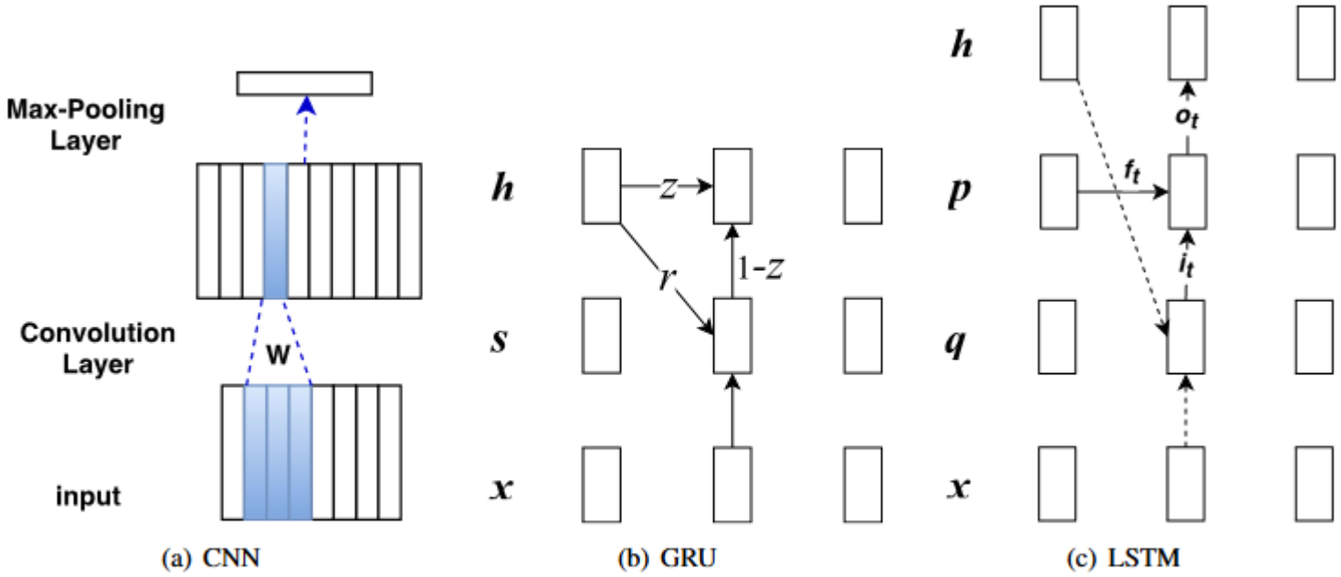


Figure 1: Three typical DNN architectures

задачи

- Sentiment Classification (SentiC)
- Relation Classification (RC)
- Textual Entailment (TE)
- Answer Selection (AS)
- Question Relation Match (QRM)
- Path Query Answering (PQA)
- Part-of-Speech Tagging

Wenpeng Yin et al.«Comparative Study of CNN and RNN for Natural Language Processing»
https://arxiv.org/pdf/1702.01923.pdf

Сравнение CNN vs RNN: нет явного победителя!

			performance	lr	hidden	batch	sentLen	filter_size	margin
TextC	SentiC (acc)	CNN	82.38	0.2	20	5	60	3	–
		GRU	**86.32**	0.1	30	50	60	–	–
		LSTM	84.51	0.2	20	40	60	–	–
	RC (F1)	CNN	68.02	0.12	70	10	20	3	–
		GRU	**68.56**	0.12	80	100	20	–	–
		LSTM	66.45	0.1	80	20	20	–	–
SemMatch	TE (acc)	CNN	77.13	0.1	70	50	50	3	–
		GRU	**78.78**	0.1	50	80	65	–	–
		LSTM	77.85	0.1	80	50	50	–	–
	AS (MAP & MRR)	CNN	**(63.69,65.01)**	0.01	30	60	40	3	0.3
		GRU	(62.58,63.59)	0.1	80	150	40	–	0.3
		LSTM	(62.00,63.26)	0.1	60	150	45	–	0.1
	QRM (acc)	CNN	**71.50**	0.125	400	50	17	5	0.01
		GRU	69.80	1.0	400	50	17	-	0.01
		LSTM	71.44	1.0	200	50	17	-	0.01
SeqOrder	PQA (hit@10)	CNN	54.42	0.01	250	50	5	3	0.4
		GRU	**55.67**	0.1	250	50	5	–	0.3
		LSTM	55.39	0.1	300	50	5	–	0.3
ContextDep	POS tagging (acc)	CNN	94.18	0.1	100	10	60	5	–
		GRU	93.15	0.1	50	50	60	–	–
		LSTM	93.18	0.1	200	70	60	–	–
		Bi-GRU	94.26	0.1	50	50	60	–	–
		Bi-LSTM	**94.35**	0.1	150	5	60	–	–

Table 1: Best results or CNN, GRU and LSTM in NLP tasks

CNN + LSTM + CRF = LSTM-CNNs-CRF

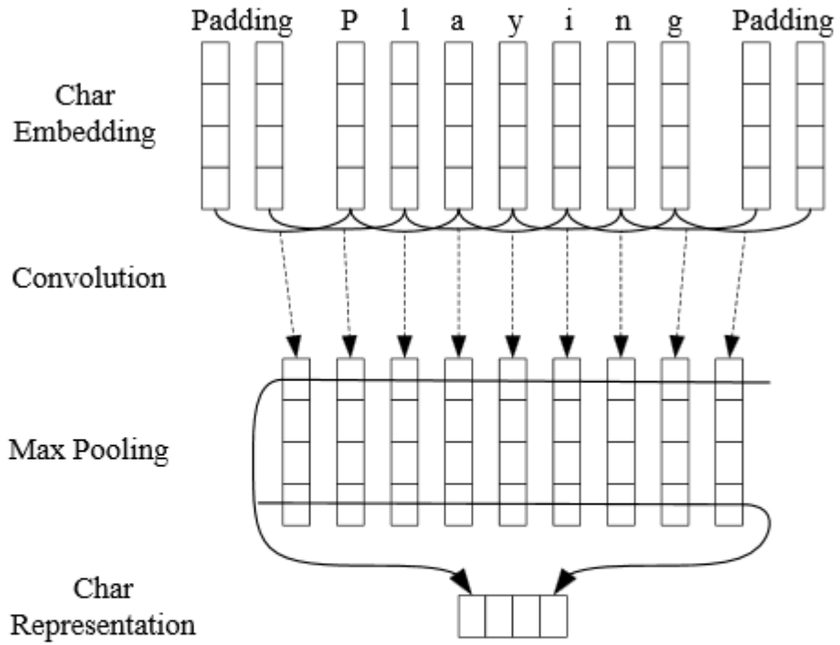
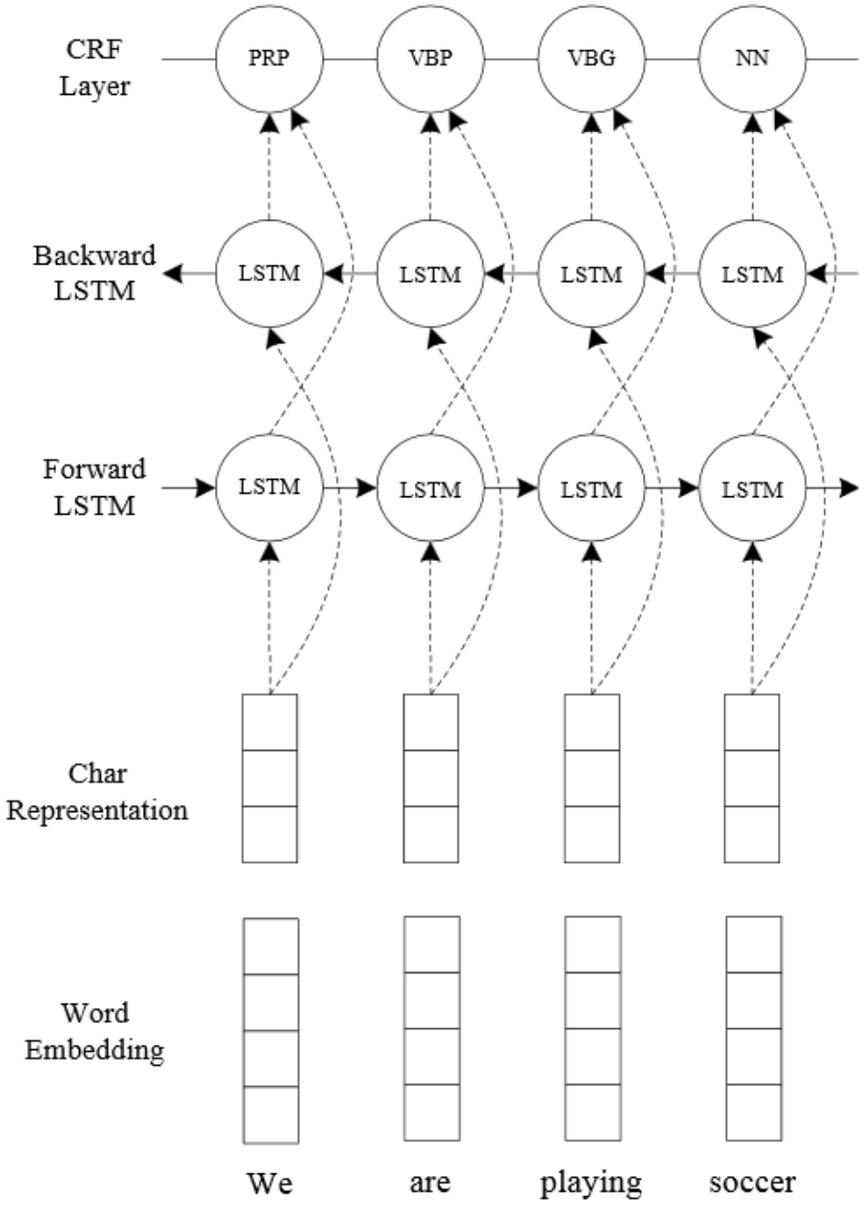


Figure 1: The convolution neural network for extracting character-level representations of words. Dashed arrows indicate a dropout layer applied before character embeddings are input to CNN.



Dashed arrows indicate dropout layers applied on both the input and output vectors of BLSTM.

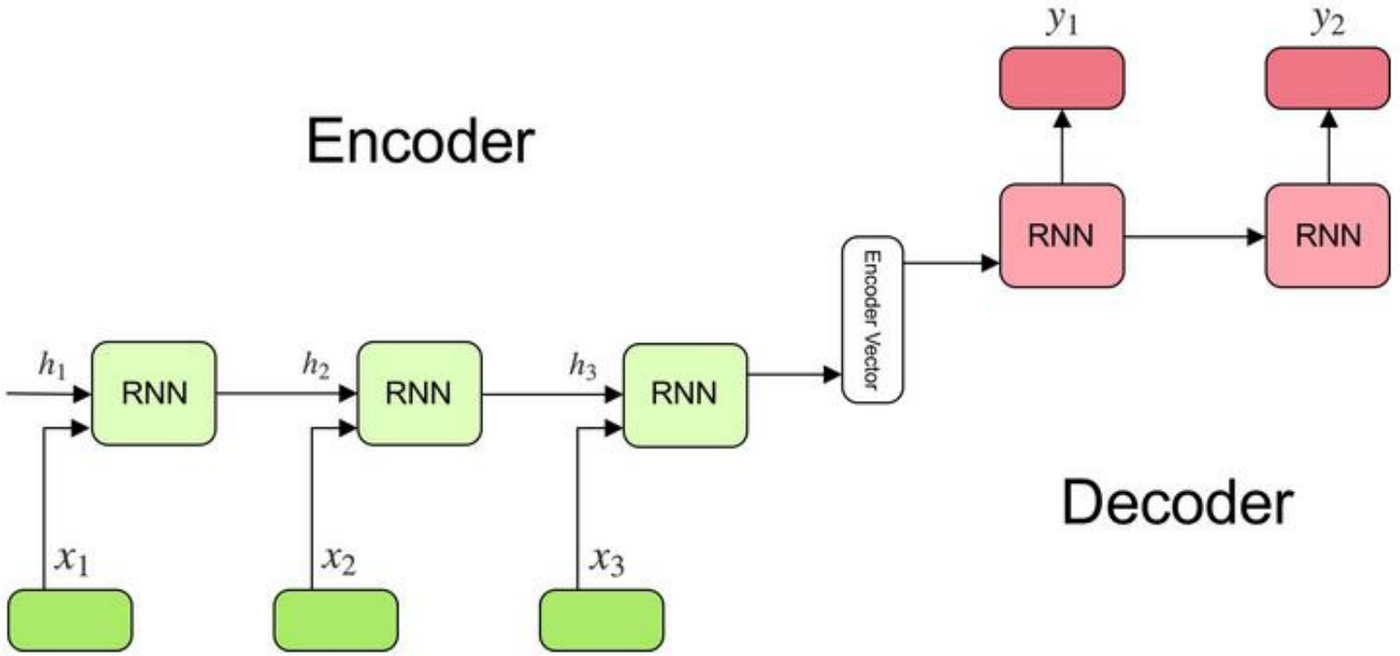
CNN + LSTM + CRF = LSTM-CNNs-CRF

	POS		NER					
	Dev	Test	Dev			Test		
Model	Acc.	Acc.	Prec.	Recall	F1	Prec.	Recall	F1
BRNN	96.56	96.76	92.04	89.13	90.56	87.05	83.88	85.44
BLSTM	96.88	96.93	92.31	90.85	91.57	87.77	86.23	87.00
BLSTM-CNN	97.34	97.33	92.52	93.64	93.07	88.53	90.21	89.36
BRNN-CNN-CRF	97.46	97.55	94.85	94.63	94.74	91.35	91.06	91.21

Table 3: Performance of our model on both the development and test sets of the two tasks, together with three baseline systems.

Xuezhe Maand, Eduard Hovy «End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF» // https://arxiv.org/pdf/1603.01354.pdf

Модель seq2seq



http://www.davidsbatista.net/blog/2020/01/25/Attention-seq2seq/

Sutskever I. «Sequence to Sequence Learning with Neural Networks», 2014 // https://arxiv.org/abs/1409.3215

Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

Kyunghyun Cho et. al. «Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation» https://www.aclweb.org/anthology/D14-1179/

Модель seq2seq: как переводить последовательность → последовательность

Многослойная (4 слоя) LSTM

размерность представления = 1000

входной словарь = 160,000

выходной словарь = 80,000

кодировщик (encoder) – декодировщик (decoder)

кодировщик: входная последовательность → вектор

декодировщик: вектор → таргетная последовательность

Это разные LSTM, у них разные параметры!

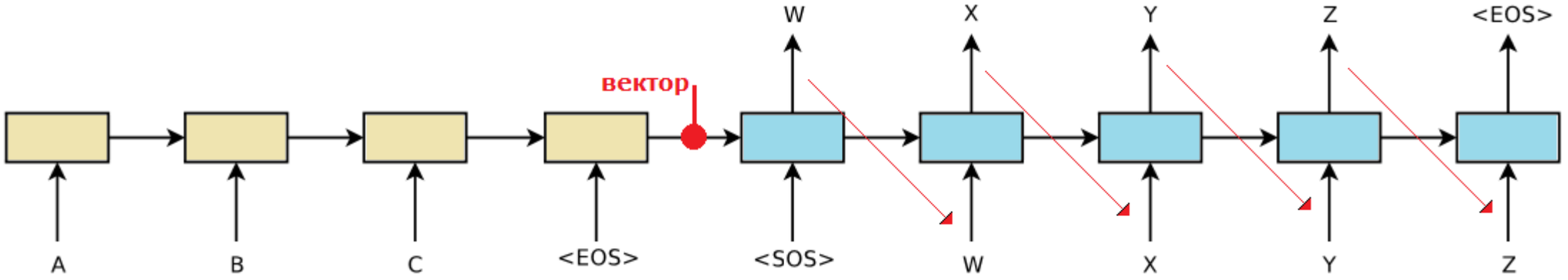
Интересно: в задаче перевода качество повышало инвертирование порядка входа!

Обучение 10 дней

Тоже хороши ансамбли

Модель seq2seq

здесь декодировщик называют также **языковой моделью**



при работе (inference) – подаём на вход сгенерированное

при обучении – среднее ошибок на всех выходах (ex negative log prob)

тонкости:

на рисунке в декодировщике передаётся только его внутреннее состояние

выход кодировщика передаётся лишь первому элементу

можно его передавать всем – чтобы информация о входе была у всех

Модель seq2seq: внутреннее представление предложений

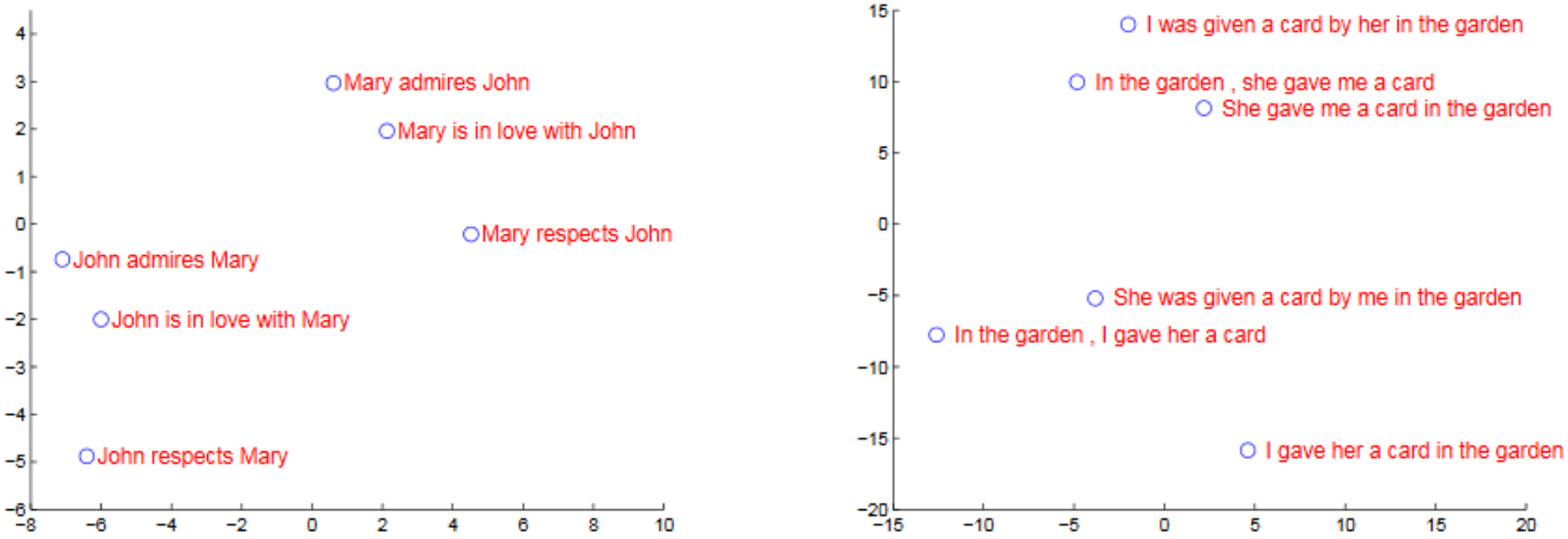


Figure 2: The figure shows a 2-dimensional PCA projection of the LSTM hidden states that are obtained after processing the phrases in the figures. The phrases are clustered by meaning, which in these examples is primarily a function of word order, which would be difficult to capture with a bag-of-words model. Notice that both clusters have similar internal structure.

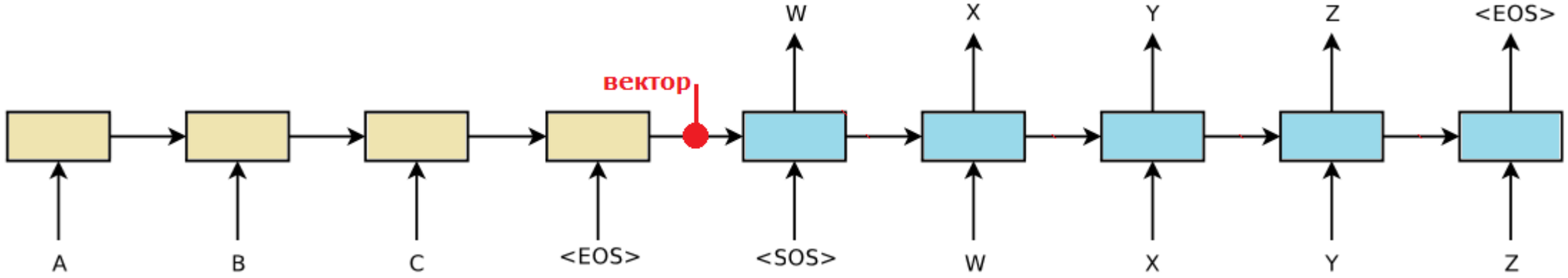
left-to-right beam-search decode потом будет

если выбираем лучшего следующего, не обязательно максимизируем качество

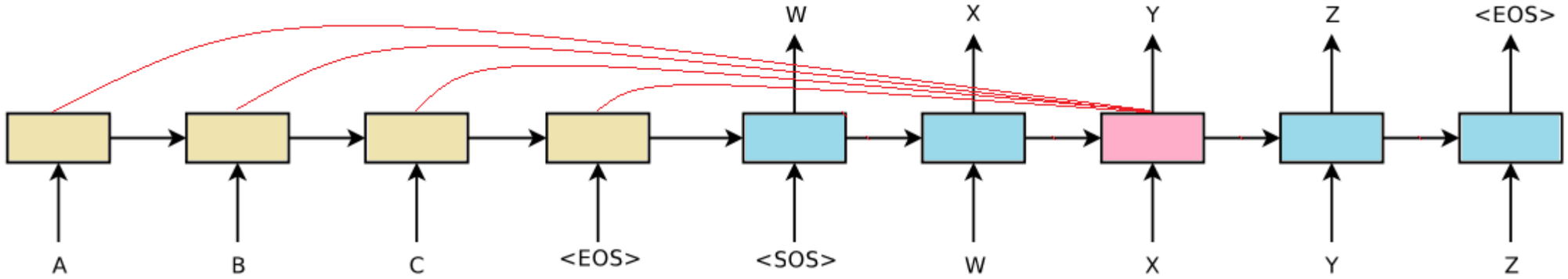
Обобщения seq2seq

На одном нейроне вся информация о тексте… плохо

(особенно для длинных последовательностей)



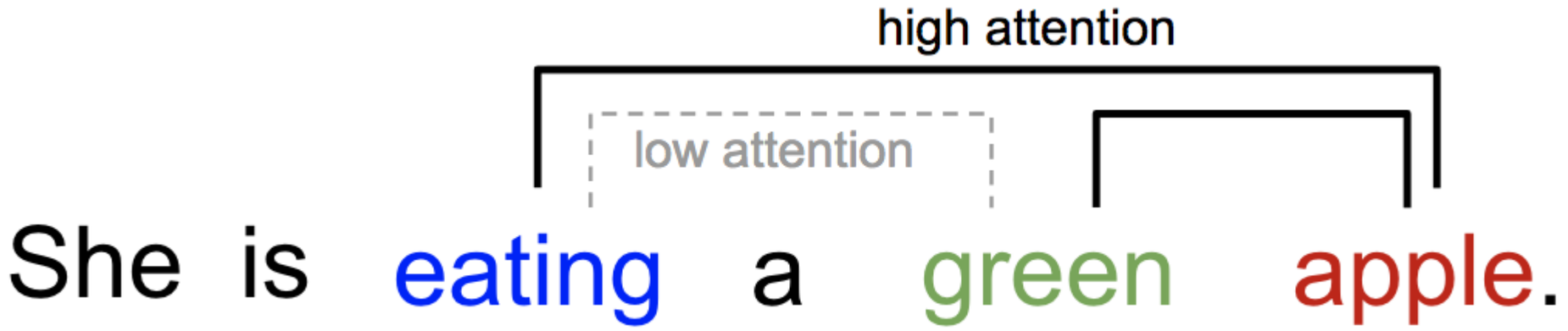
Решение – механизм внимания



Bahdanau et al. 2015 «Neural Machine Translation by Jointly Learning to Align and Translate» // ICLR 2015 https://arxiv.org/pdf/1409.0473.pdf

Механизм внимания

Концепция: есть взаимосвязи между словами



кодировщик передаёт в декодировщик не только одно состояние,
а состояния всех токенов!

– но для этого нужен механизм пулинга посл-ти состояний любой длины

выход – пулинг по схожести

https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html#born-for-translation

Механизм внимания

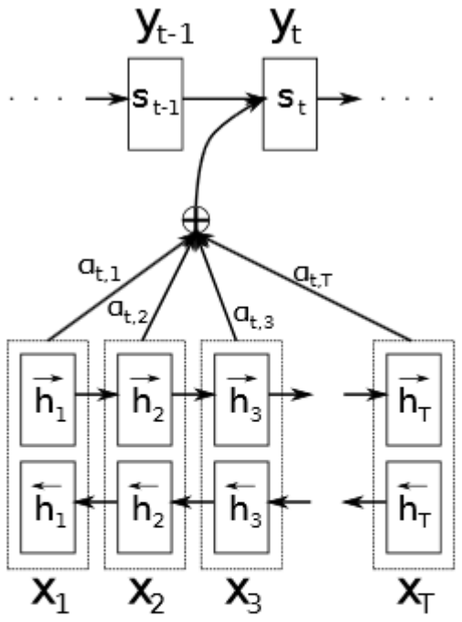


Figure 1: The graphical illustration of the proposed model trying to generate the t-th target word y_t given a source sentence $(x_1, x_2, \ldots, x_T)$.

Не будем пытаться закодировать всё предложение одним вектором!

Добавляется контекстный вектор (конкатенируется)

$$c_i = \sum_j \alpha_{ij} h_j$$

веса (softmax)

$$\alpha_{ij} = \exp(e_{ij}) / \sum_k \exp(e_{ik})$$

Насколько соответствуют состояния

$$e_{ij} = a(s_{i-1}, h_j)$$

Bidirectional RNN (BiRNN) $\Rightarrow$

учитываются не только слова ДО, но и ПОСЛЕ

Конкатенация состояния ДО и состояния ПОСЛЕ

Механизм внимания

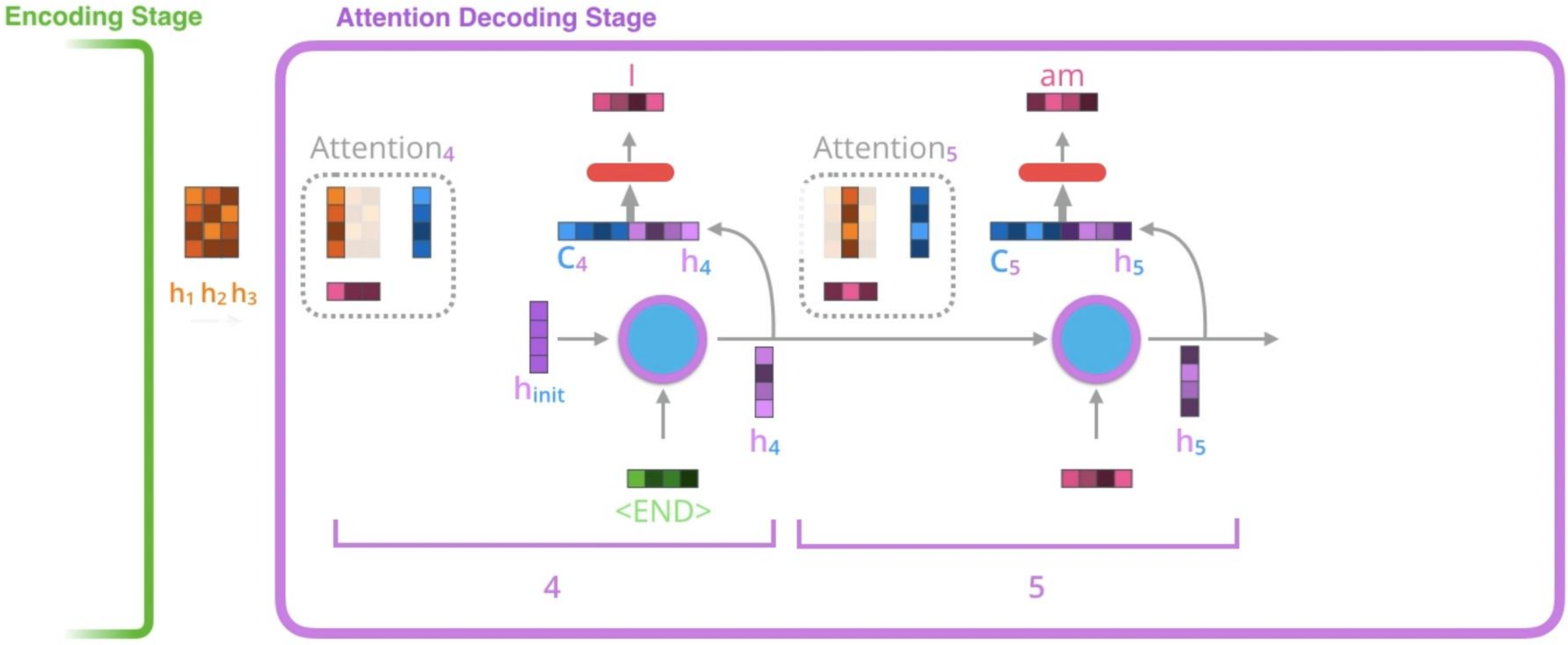
соответствие $e_{ij} = a(s_{i-1}, h_j)$ **может быть:**

Basic dot-product https://arxiv.org/pdf/1508.04025.pdf	$a(s,h) = s^{\mathrm{T}} h$
Multiplicative attention https://arxiv.org/pdf/1508.04025.pdf	$a(s,h) = s^{\mathrm{T}} W h$
Additive attention https://arxiv.org/pdf/1409.0473.pdf	$a(s,h) = w^{\mathrm{T}} \tanh(W_1 s - W_2 h) = w^{\mathrm{T}} \tanh(W[s;h])$
Scaled dot-product http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf	$a(s,h) = s^{\mathrm{T}} h / \sqrt{d}$
Content-base attention https://arxiv.org/abs/1410.5401	$a(s,h) = \cos(s,h)$

+ разные нормировки по размерности

Thang Luong, Hieu Pham, Christopher D. Manning «Effective Approaches to Attention-based Neural Machine Translation» https://www.aclweb.org/anthology/D15-1166.pdf

Механизм внимания



полученный в л/к состояний кодировщика вектор конкатенируем с текущим состоянием

Механизм внимания: получаем интерпретацию и выравнивание (alignment)

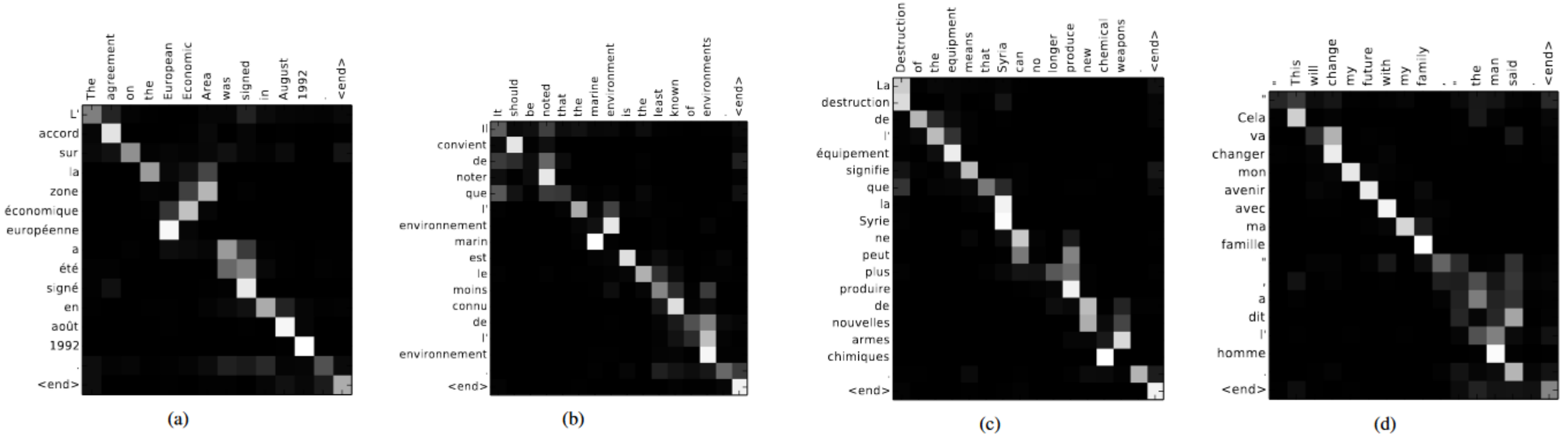


Figure 3: Four sample alignments found by RNNsearch-50. The x-axis and y-axis of each plot correspond to the words in the source sentence (English) and the generated translation (French), respectively. Each pixel shows the weight α_{ij} of the annotation of the j-th source word for the i-th target word (see Eq. (6)), in grayscale (0: black, 1: white). (a) an arbitrary sentence. (b–d) three randomly selected samples among the sentences without any unknown words and of length between 10 and 20 words from the test set.

Bahdanau D. и др. «Neural Machine Translation by Jointly Learning to Align and Translate» // https://arxiv.org/abs/1409.0473

Механизм внимания: решение проблемы «узкого горла»

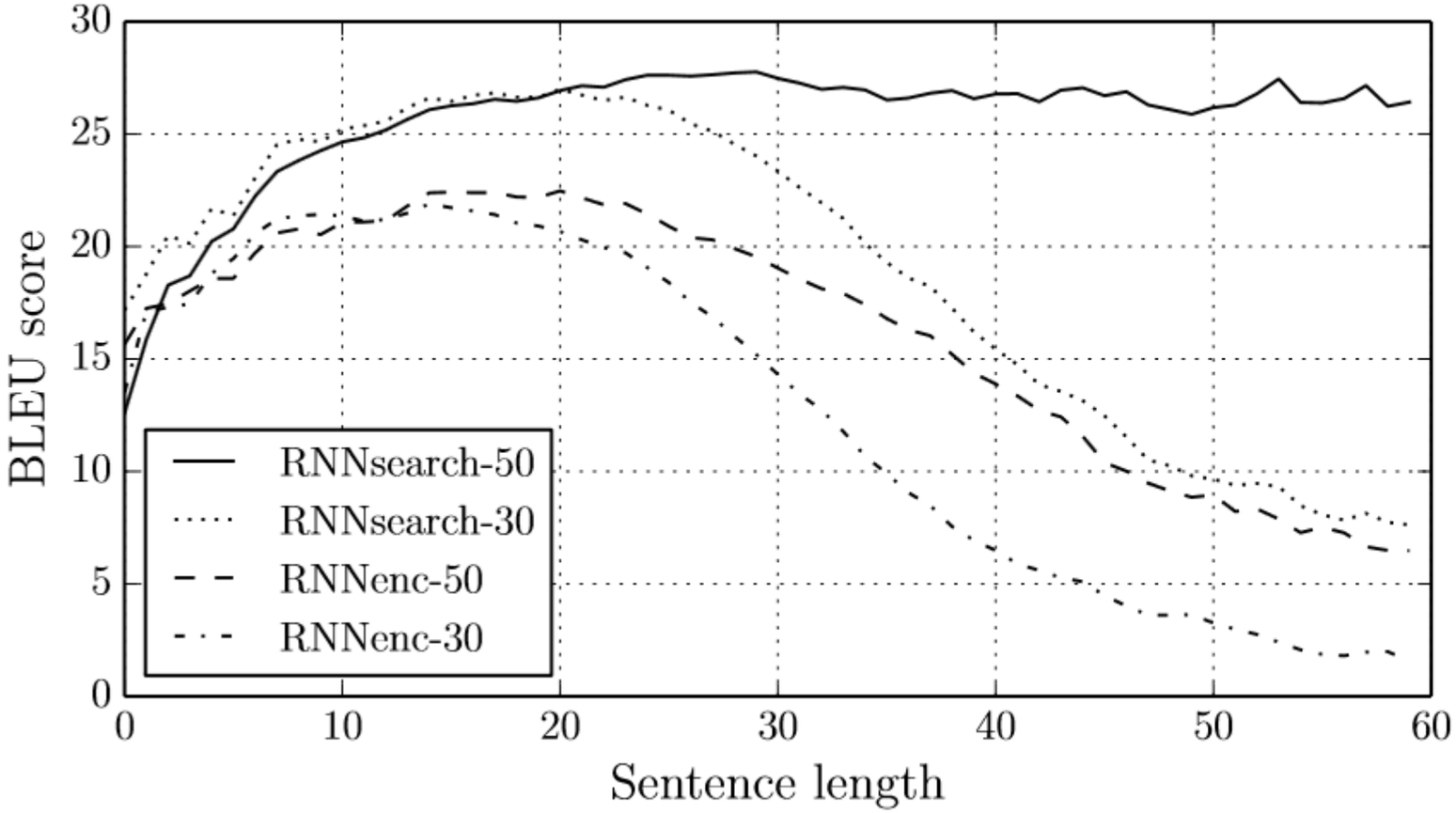
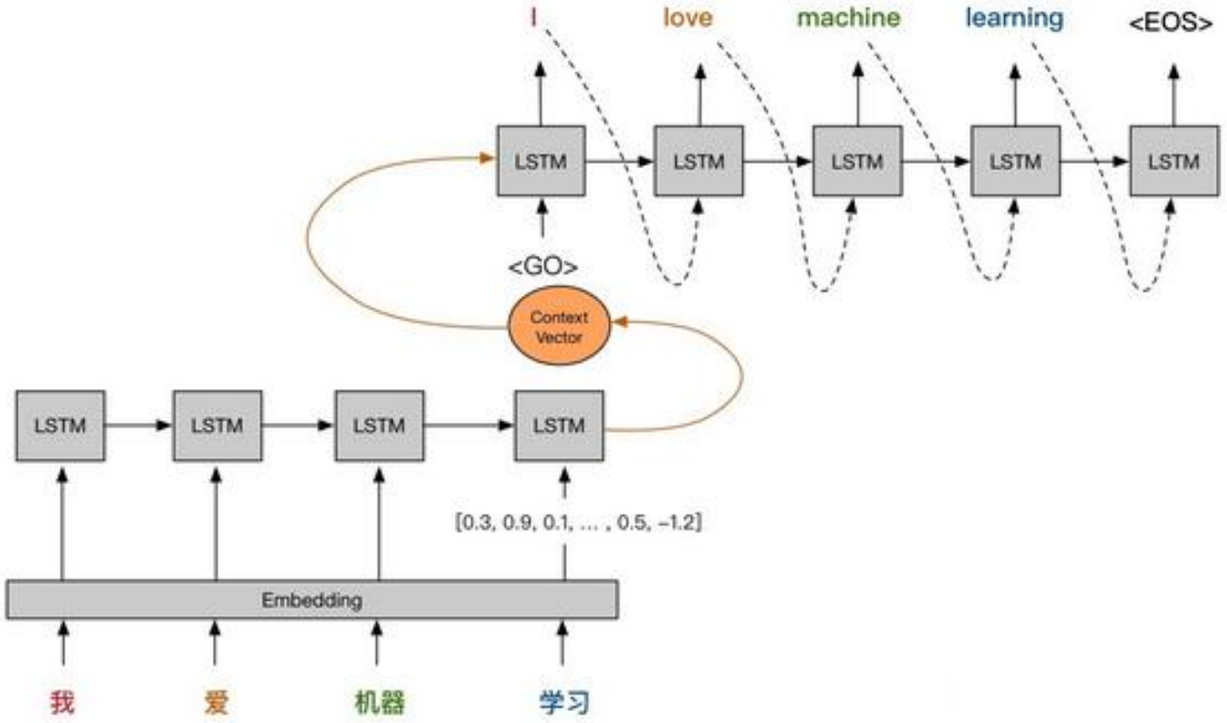
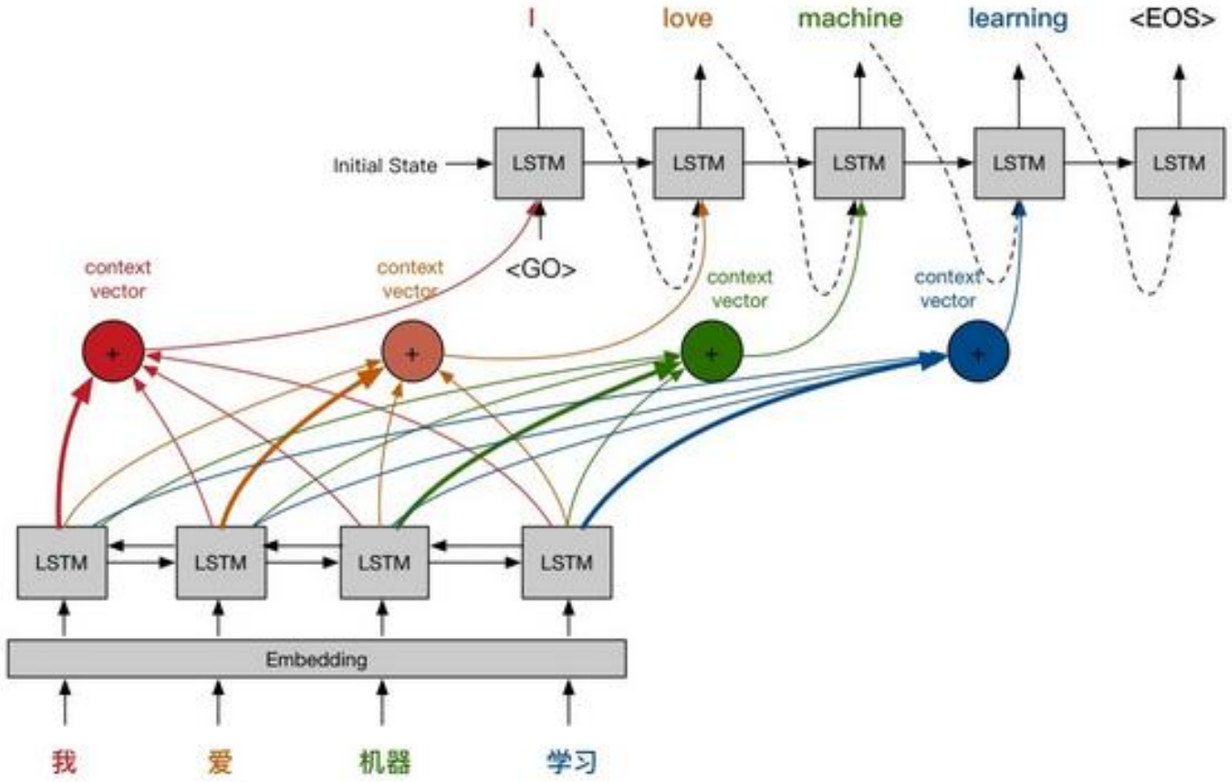


Figure 2: The BLEU scores of the generated translations on the test set with respect to the lengths of the sentences. The results are on the full test set which includes sentences having unknown words to the models.

seq2seq vs attention





Внимание – техника вычисления взвешенной суммы значений (values) по запросу (query)

~ техника получения описания (representation) фиксированного размера по запросу

https://zhuanlan.zhihu.com/p/37290775

Плюсы механизма внимания (Attention)

- улучшает качество перевода (и не только)
- решает проблему «узкого горла»
- появляется интерпретируемость
- решает проблему «затухания сигнала» / исчезающего градиента
- получаем выравнивание (alignment) «бесплатно» в переводе

Виды внимания

Self-Attention / intra-attention	к разным позициям одной и той же входной последовательности
Global / Soft	ко всему входу
Local / Hard	к части входа

http://proceedings.mlr.press/v37/xuc15.pdf

Представления слов

решают проблему «что дать на вход сети»

токен / номер $\rightarrow$ вектор

способ «засунуть» дискретные объекты в НС

представляют слова так, что похожие слова имеют похожие представления

решается проблема незнакомых слов

имеют небольшую размерность

$\Rightarrow$ сокращают число параметров сети

могут быть получены на большом неразмеченном тексте

трансферное обучение

Способы кодирования / представления слов

- **OHE**

слишком большая размерность, нет хорошей близости

- **counts (сумма OHE соседей)**

более нетривиальная оценка близости с помощью cos

- **вложение (embeddings)**

умный алгоритм задания кодировки

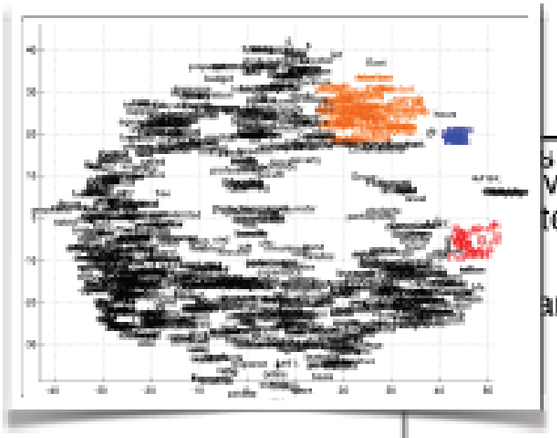
«word embeddings»

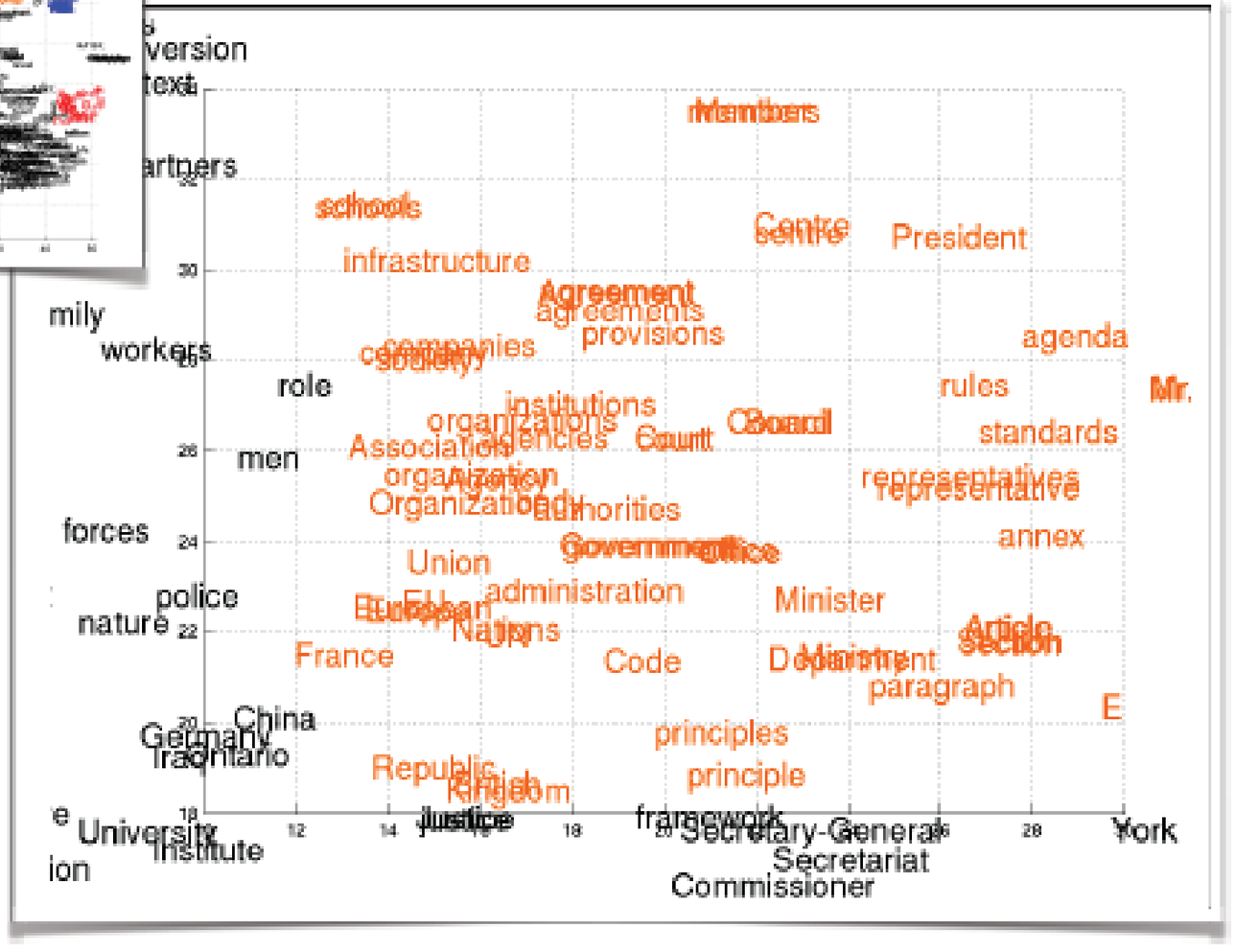
Представления слов в вещественном многомерном пространстве

$\Rightarrow$ можно использовать в матмоделях

Предобученные **Обученные для конкретной задачи**

Вложение слов в непрерывное пространство (embedding)





DL-классика: безконтекстные методы

context-free – не учитывающие контекст

учитывают контекст при обучении представления,
но при использовании это уже фиксированный вектор –
контекст не учитывается

- **word2vec** = предсказания слово $\leftrightarrow$ контекст
- **fasttext** = word2vec + ngrams
- **Glove** = разложение матрицы совместной встречаемости

word2vec – дистрибутивная семантика

Трюк: настраиваем модель, но не для использования в задаче,
которой учим (нас интересуют формируемые внутренние представления)
принцип трансферного обучения (ex: автокодировщики)

Термины «distributional semantics»

Смысл слова определяется контекстом

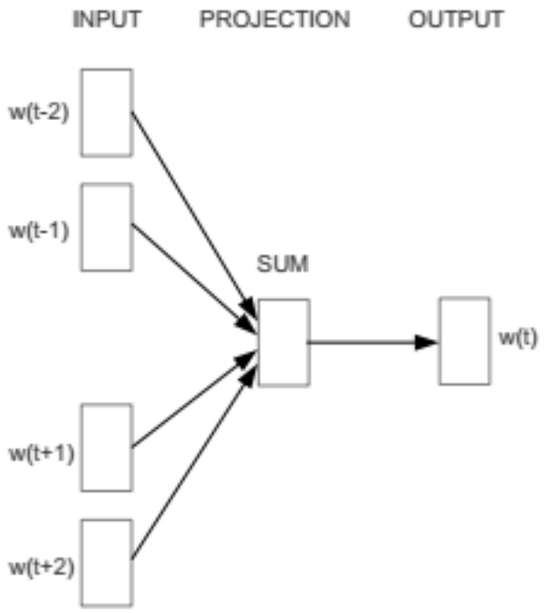
Полосатая маленькая *** мурлычит и пьёт молоко**

Весна

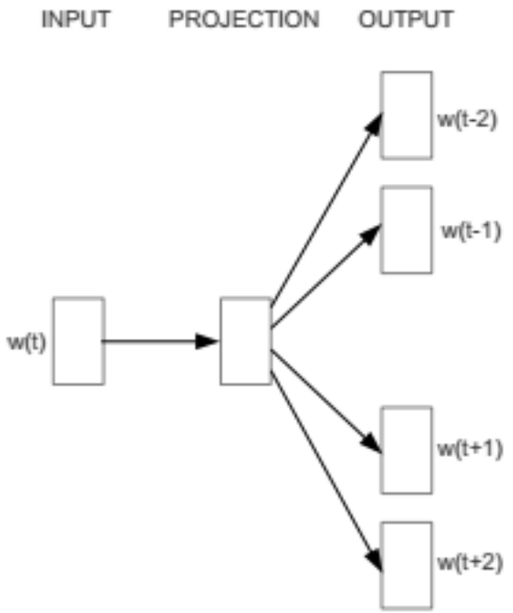
Ручьи
Тает
Цветёт
Зеленеет
Прилетают

[Mikolov et al. 2013]

word2vec: два подхода к реализации



CBOW



Skip-gram

CBOW = Continuous Bag of Words (быстрее, окно ~ 5, большие корпуса)
skipgram model (раньше считалось, что лучше, окно ~ 10, небольшие корпуса)

это пример самообучения (self-supervision) – когда разметка автоматическая
можно использовать большие корпуса внешних текстов

word2vec: CBOW

Предсказываем слово по контексту

используется реже, чем следующая реализация

$$P(x_t \mid \text{context}(x_t)) = \text{softmax}\left(V \left(W \sum_{x_i \in \text{context}(x_t)} OHE(x_i) \right) \right)$$

выделено то, что будем считать кодировкой

контекст – слово (слова), которое недалеко располагается (в окрестности)

http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

word2vec: CBOW

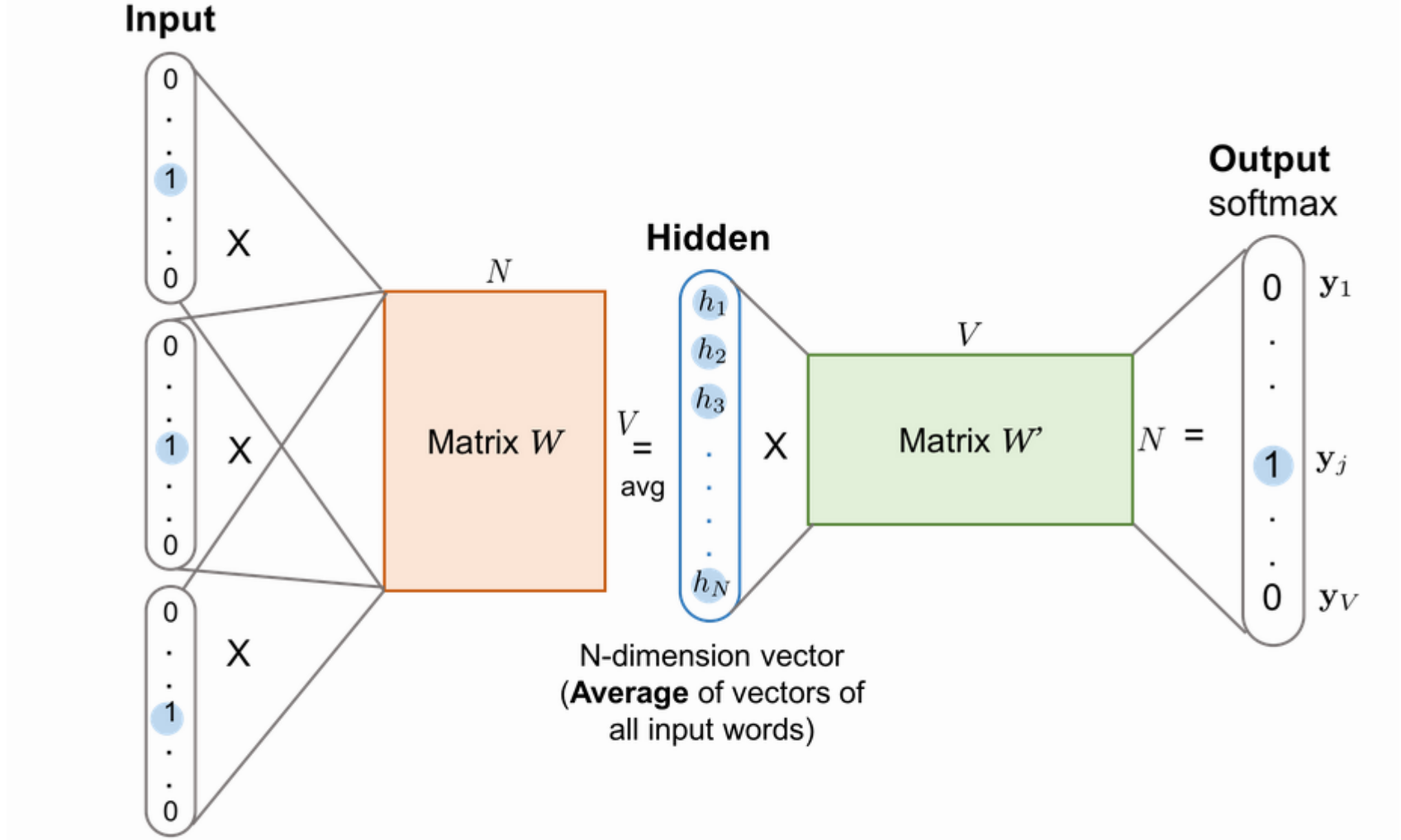
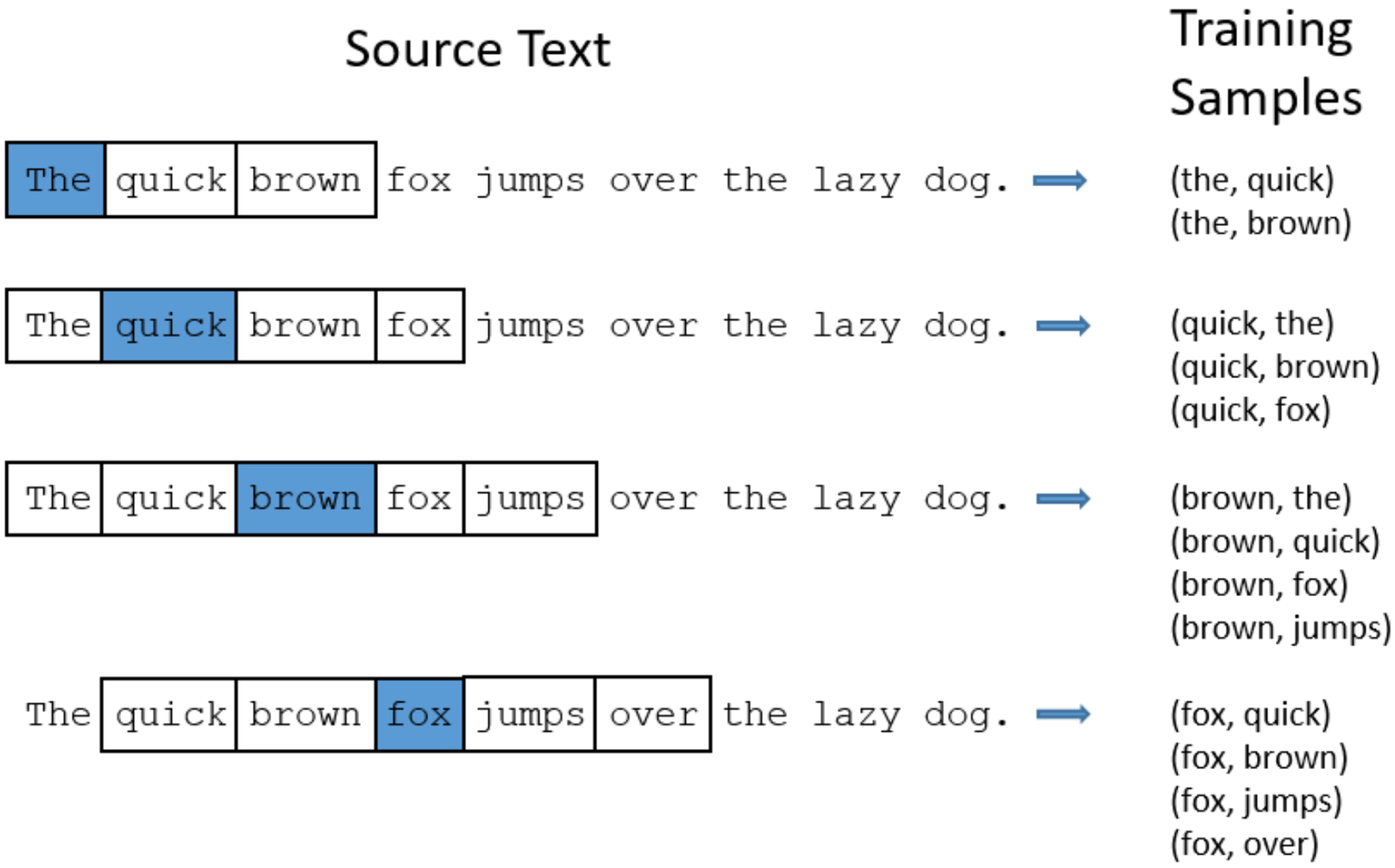


Fig. 2. The CBOW model. Word vectors of multiple context words are averaged to get a fixed-length vector as in the hidden layer. Other symbols have the same meanings as in Fig 1.

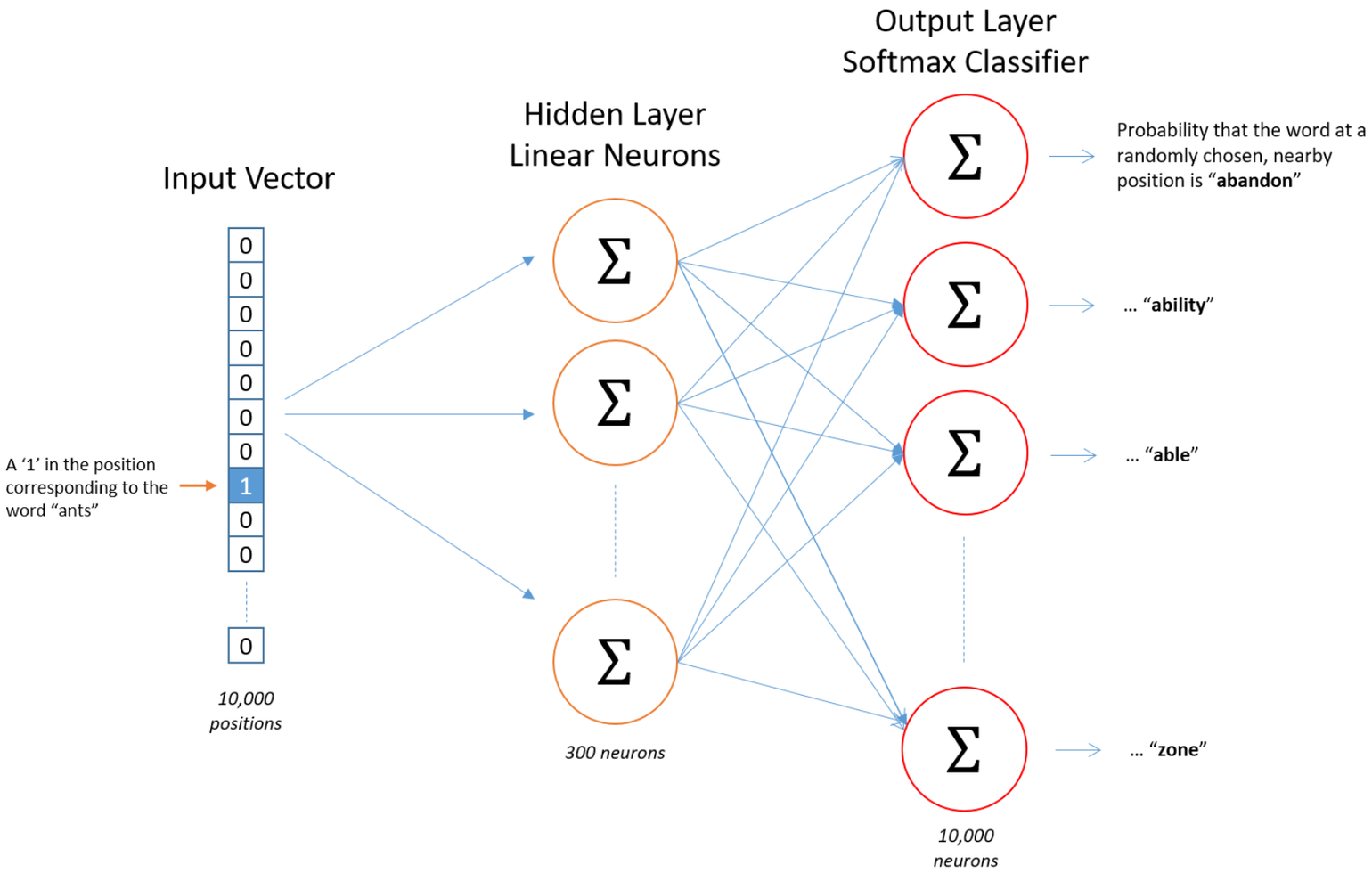
https://lilianweng.github.io/lil-log/2017/10/15/learning-word-embedding.html

word2vec: skip-gram

Предсказываем контекст по слову: слово → слово



word2vec: skip-gram



вход: ONE-кодировка слова

выход: распределение вероятностей

средний слой – для нашего кодирования

word2vec: skip-gram

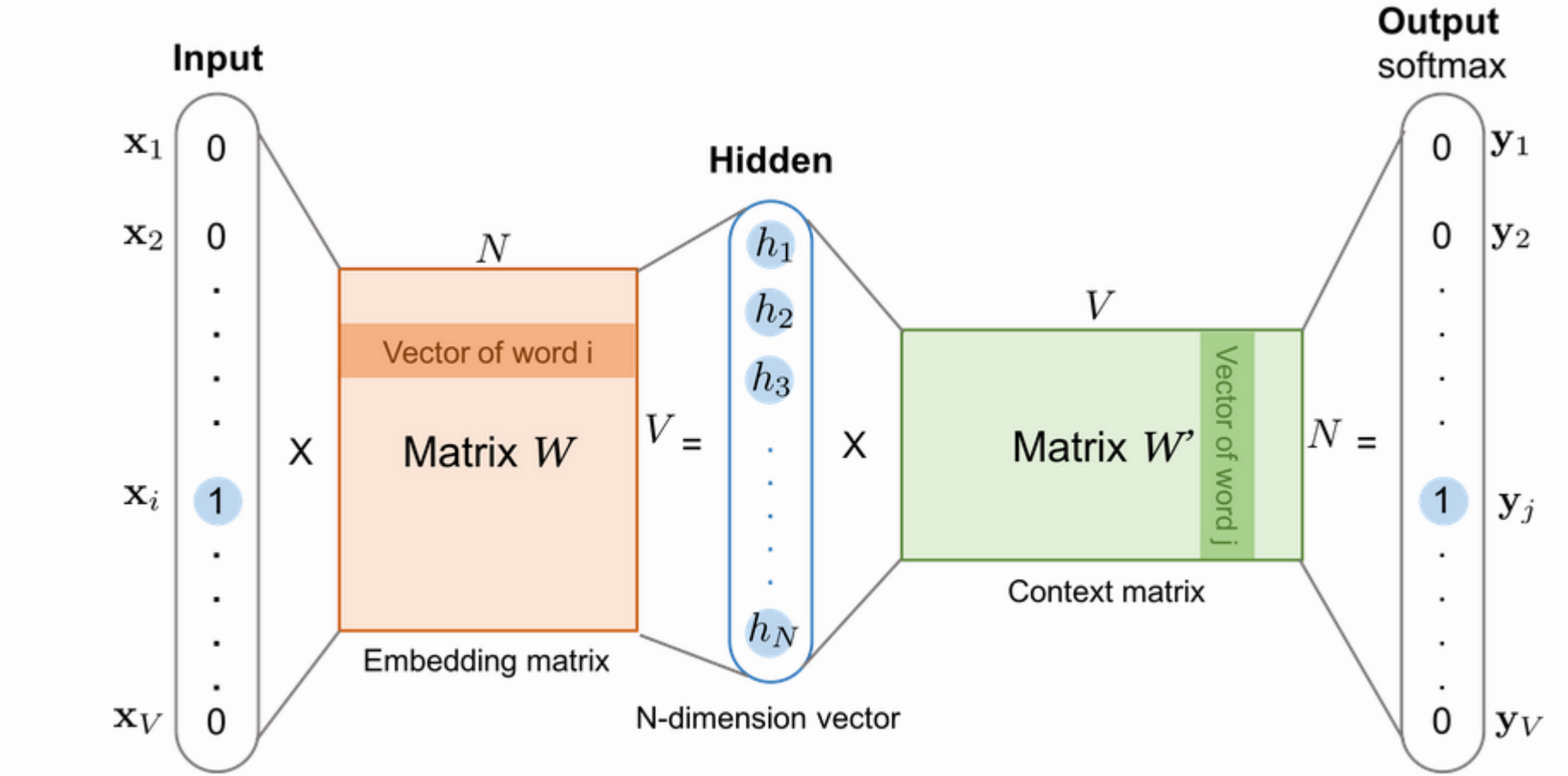


Fig. 1. The skip-gram model. Both the input vector $\mathbf{x}$ *and the output* $\mathbf{y}$ *are one-hot encoded word representations. The hidden layer is the word embedding of size* N.

https://lilianweng.github.io/lil-log/2017/10/15/learning-word-embedding.html

word2vec

Огромная НС

Первый слой – #слов × размерность представления (~300)

Как обучать????

здесь предложены модификации обучения:

Mikolov T. «Distributed Representations of Words and Phrases and their Compositionality» // https://arxiv.org/pdf/1310.4546.pdf

не только уменьшают время обучения, но и улучшают качество представлений

/ код слова = строка первой матрицы + столбец второй

или строка первой матрицы

Следующие слайды по

http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/

Есть отличия между реализацией и статьёй!

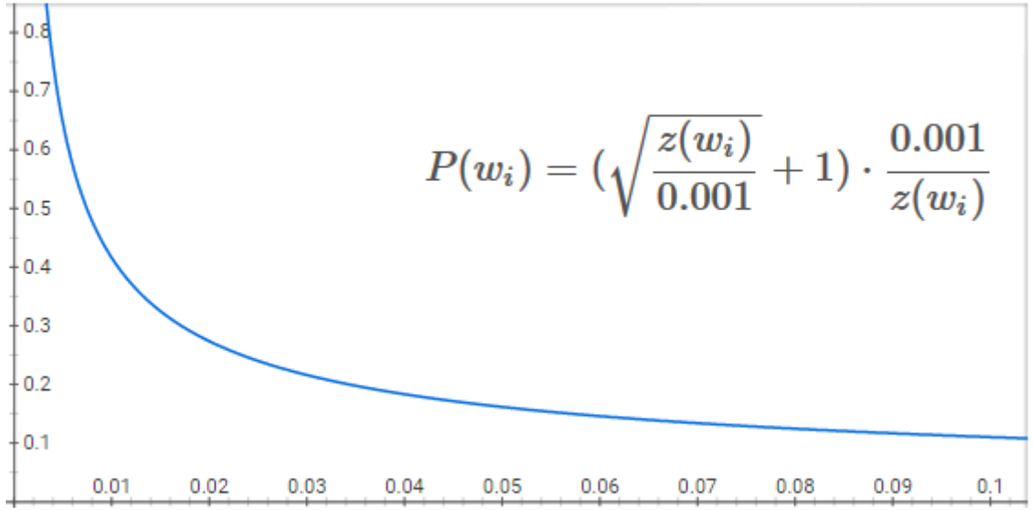
word2vec

Распространённые фразы – одно слово

White_Spunner_Construction
Bad_Habits
Toxics_Alliance

Частые слова – реже выбираются при обучении

вероятность быть выбранным от частоты:



(quick, the)
– про эти слова речь
используются не все пары
– идёт сэмплирование

«Negative Sampling»

у («открыл») = OHE(«дверь»)
чтобы не править много выходов, соответствующим нулям, выбираем несколько случайных (5–20)

word2vec – немного математики

Последовательность слов $x_1, \ldots, x_T$

Правдоподобие

$$\prod_{t=1}^{T} \prod_{c \in C_t} p(x_c \mid x_t) \sim \sum_{t=1}^{T} \sum_{c \in C_t} \log p(x_c \mid x_t) \to \max$$

(второе произведение по окрестности – индексы соседних слов)

Можно: $p(x_c \mid x_t) = \dfrac{\exp(s(x_t, x_c))}{\sum_x \exp(s(x_t, x))}$

Такая модель подходила бы,
если бы для каждого слова один правильный ответ
хотя тоже используется

word2vec: Negative Sampling

Как делаем… «skipgram model with negative sampling» [Mikolov]

Используем «negative log-likehood»

$$\log\left(1+\exp(-s(x_t,x_c))\right)+\sum_{x\in N_{t,c}}\log\left(1+\exp(s(x_t,x))\right)$$

$N_{t,c}$ **– выборка негативных примеров**

Если logloss $l(z)=\log\left(1+\exp(-z)\right)$**, то**

$$\sum_{t=1}^{T}\left[\sum_{c\in C_t}l(s(x_t,x_c))+\sum_{x\in N_{t,c}}l(-s(x_t,x))\right]\to\min$$

Скоринговая функция: $s(x_t,x_c)=\mathrm{vec}(x_t)^{\mathrm{T}}\cdot\mathrm{vec}(x_c)$

– тут представление входа и выхода

тут нужны будут негативные примеры

word2vec: Negative Sampling

Выбор негативных слов производится не равномерно

тогда вероятность выбора слова

$$\frac{\#i}{\sum_j \#j}$$

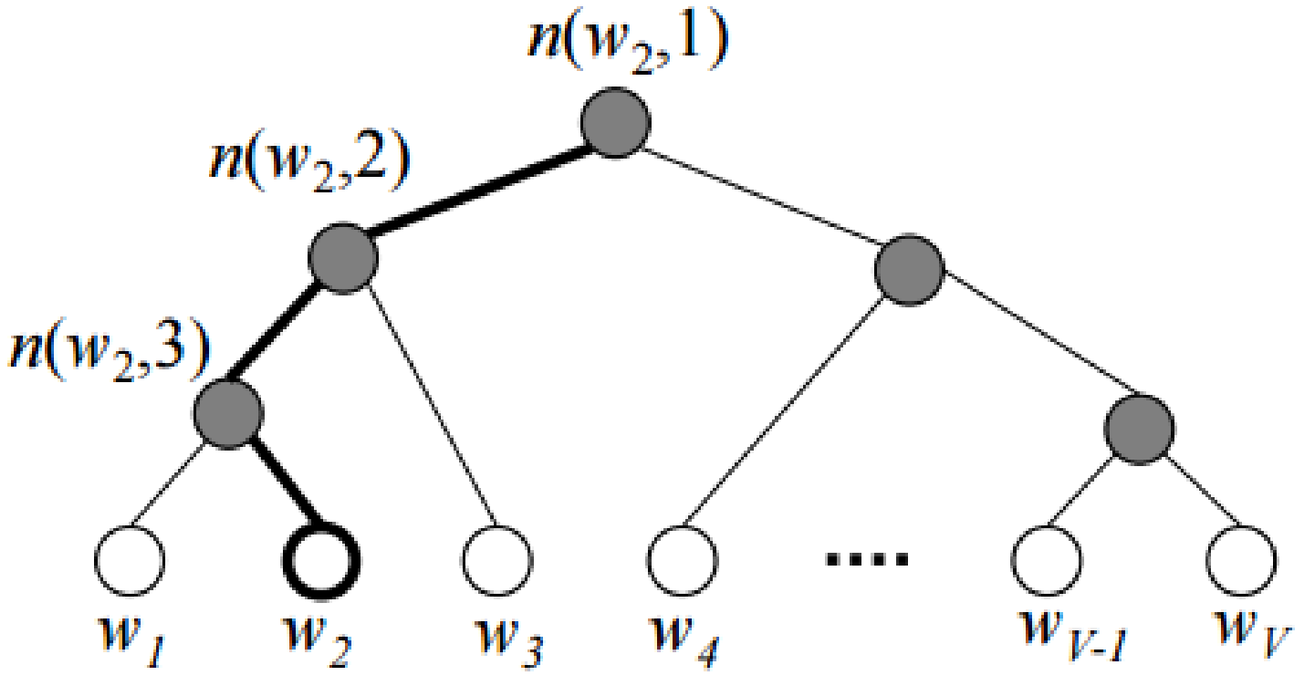
а с вероятностью

$$\frac{(\#i)^{3/4}}{\sum_j (\#j)^{3/4}}$$

в результате экспериментов – так лучше

Hierarchical Softmax

softmax-слой представляется так (специальная кодировка Хаффмана)



листья – слова

вероятность = произведение вероятностей в рёбрах пути

Ближайшие соседи

Peace	Path	Stop
Peaceful	Paths	Quit
Friendship	Approach	Stopped
Nonviolence	Titled	Avoid
	Pathway	Resist
	Way	

http://bionlp-www.utu.fi/wv_demo/

+ осмысленное соседство

+ осмысленные арифметические операции

Операции над представлениями слов

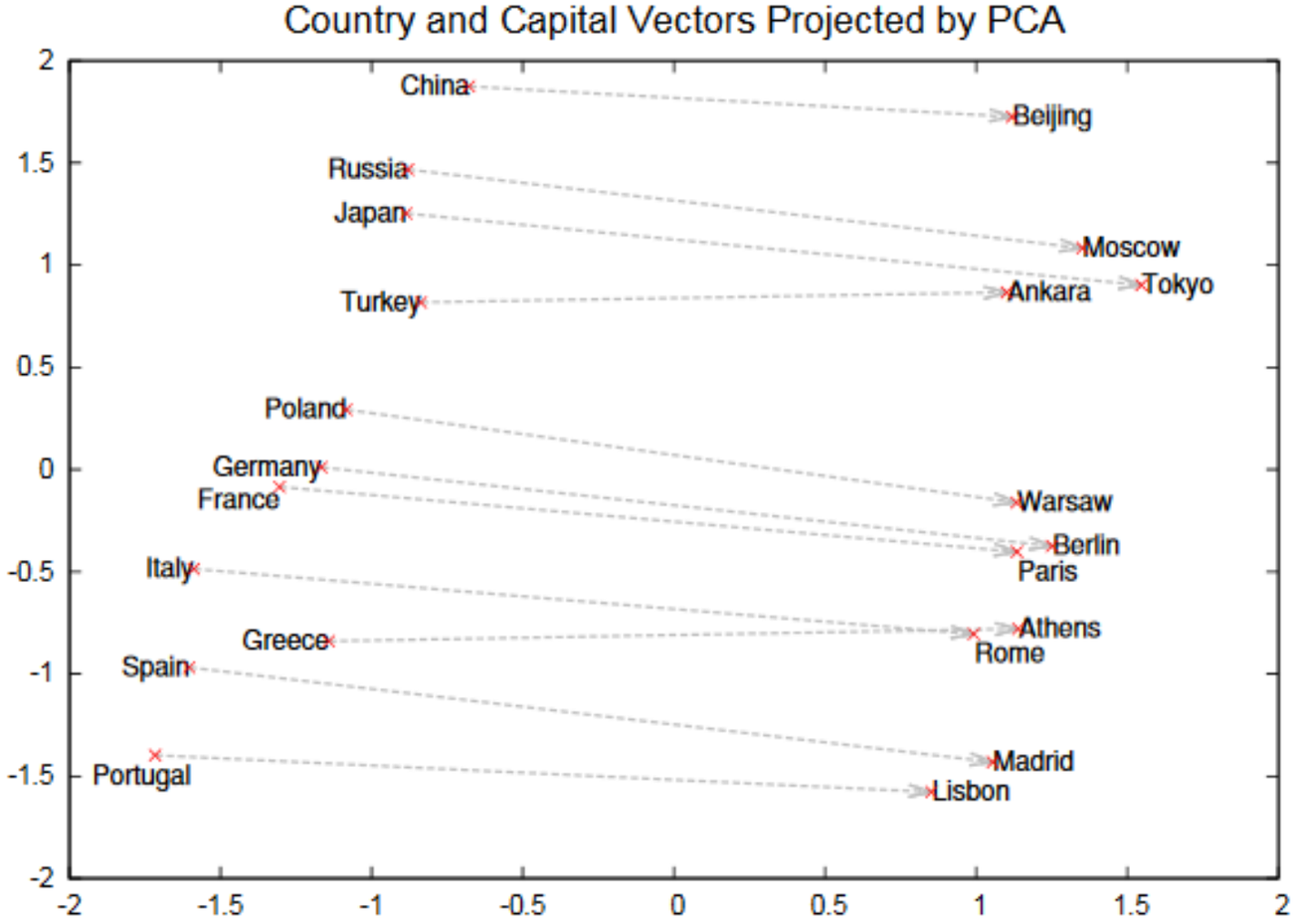


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

[Mikolov et al., 2013] https://arxiv.org/pdf/1310.4546.pdf

word2vec: ошибка в исходном коде

Считалось, что CBOW хуже Skip-gram

Оказалось, что была ошибка в negative sampling в популярных реализациях

$$\frac{\partial \mathcal{L}}{\partial v_{w_j}} = \boxed{\frac{1}{C}} \left[(\sigma({v'_{w_O}}^{\top} v_c) - 1) v'_{w_O} + \sum_{i=1}^{k} \sigma({v'_{n_i}}^{\top} v_c) v'_{n_i} \right]$$

Figure 2: Average negative sampling loss per token for every batch of 5 million tokens for a single epoch of CBOW training on Wikipedia. The shaded region corresponds to the 95% bootstrapped confidence interval over average token loss on 100K token batches.

- **а в реализациях длина контекста C тоже сэмплируется!**
- **и ещё в производной по другому параметру C нет,**

так что у нас получается смещённый вектор

Ozan İrsoy et al. «Corrected CBOW Performs as well as Skip-gram» // https://arxiv.org/pdf/2012.15332.pdf

Другие представления: fasttext

тоже «слово → контекст»

попытка учесть морфологию слов

раньше «сеть», «сетевой», «сетью» разные векторы…

+ использовать n-граммные представления слова

«where» ~ <wh, whe, her, ere, re>

n-граммы хэшируются;)

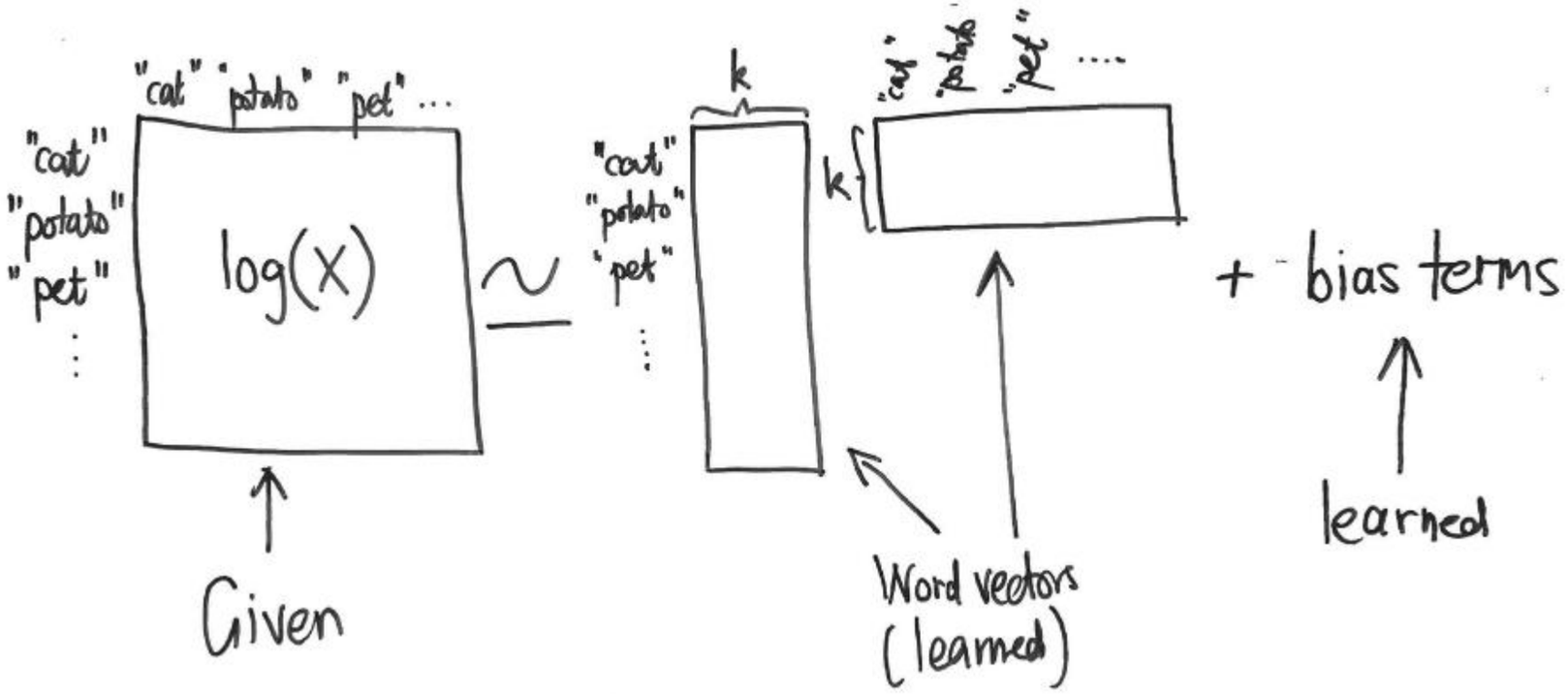
код = сумма кодов для n-грамм

решается проблема новых слов

Bojanowski P. et al. «Enriching Word Vectors with Subword Information» //
https://arxiv.org/pdf/1607.04606.pdf

https://fasttext.cc – тут есть все ссылки!!!

Glove: Global Vectors for Word Representation



идея в разложении матрицы

http://building-babylon.net/2015/07/29/glove-global-vectors-for-word-representations/

https://nlp.stanford.edu/projects/glove/

Glove: Global Vectors for Word Representation

$\#ij$ – **сколько раз слово j в контексте слова i**

(на расстоянии ≤k слов) **есть и другие варианты**

$$\sum_{i,j} f(\#ij)(w_i^{\mathrm{T}}\tilde{w}_j + b_i + \tilde{b}_j - \log(\#ij))^2 \to \min$$

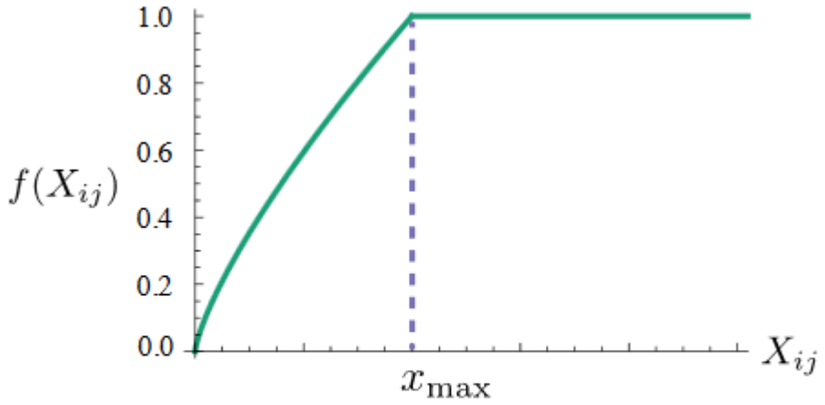


Figure 1: Weighting function f with $\alpha = 3/4$.

$$f(x) = \begin{cases} \left(\dfrac{x}{x_{\max}}\right)^{\alpha}, & x < x_{\max}, \\ 1, & x \geq x_{\max}. \end{cases}$$

Glove: ближайшие соседи

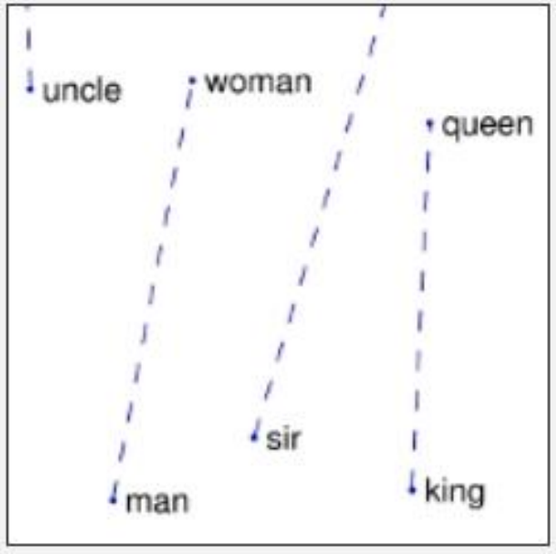
frog
frogs
toad
litoria
leptodactylidae
rana
lizard
leutherodactylus


3. litoria

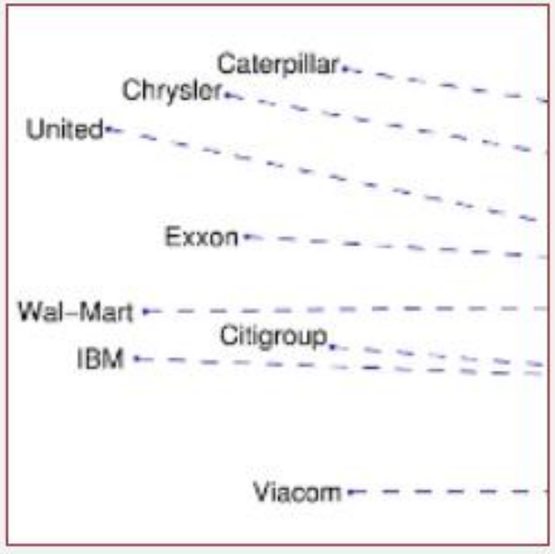

4. leptodactylidae


5. rana

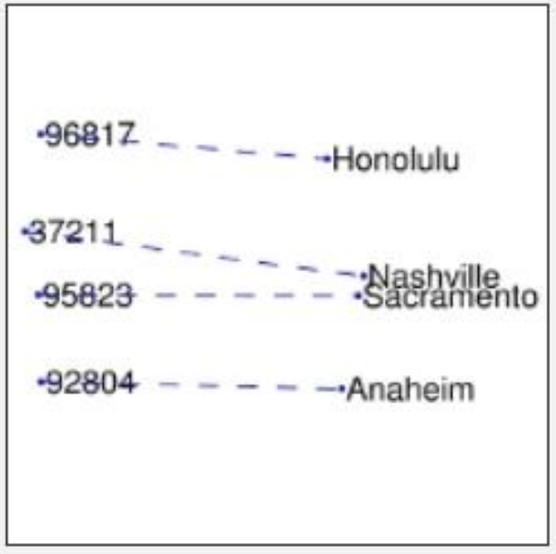

7. eleutherodactylus



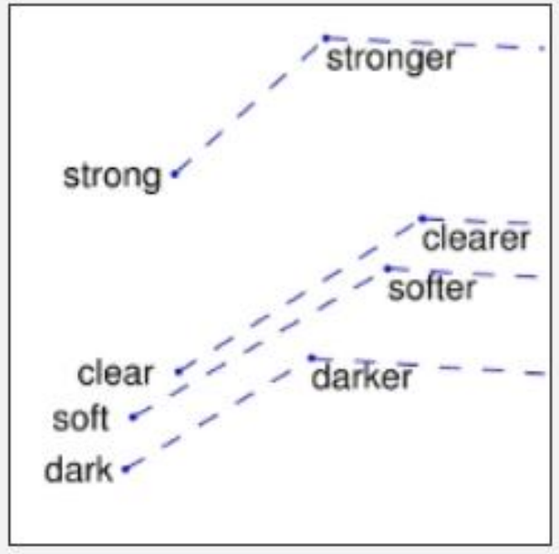
man - woman



company - ceo



city - zip code



comparative - superlative

Проверка представлений

1) визуально смотрим проекцию в пространстве, кластеры слов и т.п.

2) ближайшие соседи

есть бенчмарки схожих слов

https://nlp.stanford.edu/~lmthang/data/papers/conll13_morpho.pdf

задача «найди лишнее слово»

3) проверка арифметики

«король – мужчина + женщина = королева»

есть бенчмарки аналогий https://arxiv.org/pdf/1301.3781.pdf

кстати, в разных языках одинаковые линейные отношения

это позволяет наложить представление одного языка на другое!

4) качество при решении задач ML (downstream tasks)

ex: классификация тональности текстов фиксированным классификатором

Проверка представлений

Словарь аналогий

Table 1: *Examples of five types of semantic and nine types of syntactic questions in the Semantic-Syntactic Word Relationship test set.*

Type of relationship	Word Pair 1		Word Pair 2	
Common capital city	Athens	Greece	Oslo	Norway
All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
Currency	Angola	kwanza	Iran	rial
City-in-state	Chicago	Illinois	Stockton	California
Man-Woman	brother	sister	grandson	granddaughter
Adjective to adverb	apparent	apparently	rapid	rapidly
Opposite	possibly	impossibly	ethical	unethical
Comparative	great	greater	tough	tougher
Superlative	easy	easiest	lucky	luckiest
Present Participle	think	thinking	read	reading
Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
Past tense	walking	walked	swimming	swam
Plural nouns	mouse	mice	dollar	dollars
Plural verbs	work	works	speak	speaks

Семантический сдвиг

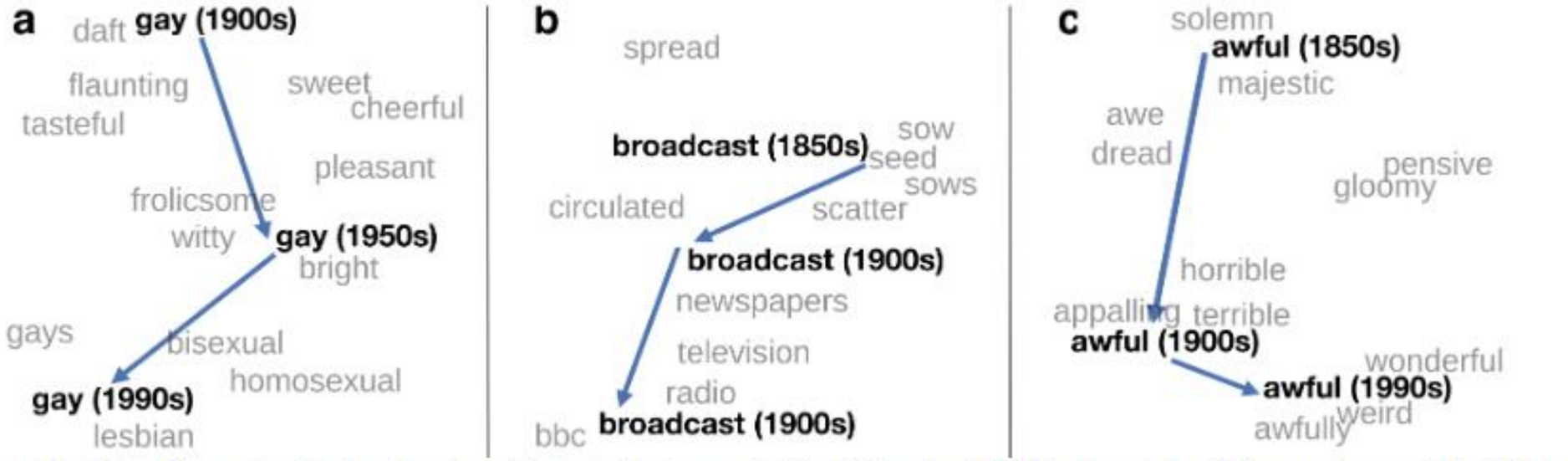


Figure 1: Two-dimensional visualization of semantic change in English using SGNS vectors.[2] **a**, The word *gay* shifted from meaning "cheerful" or "frolicsome" to referring to homosexuality. **b**, In the early 20th century *broadcast* referred to "casting out seeds"; with the rise of television and radio its meaning shifted to "transmitting signals". **c**, *Awful* underwent a process of pejoration, as it shifted from meaning "full of awe" to meaning "terrible or appalling" (Simpson et al., 1989).

обучаем модель на текстах разных периодов

«выравниваем» пространства линейным/ортогональным преобразованием

логично применять растяжение и поворот

William L. Hamilton, et al. «Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change» // https://www.aclweb.org/anthology/P16-1141.pdf

«Наложение языков»

Ingredients:

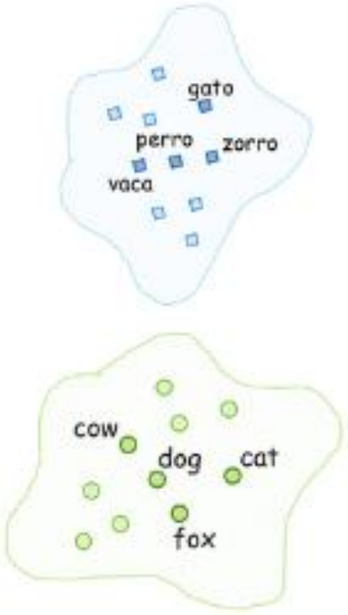
- corpus in one language (e.g., English)
- corpus in another language (e.g., Spanish)

- very small dictionary

cat ↔ gato
cow ↔ vaca
dog ↔ perro
fox ↔ zorro
...

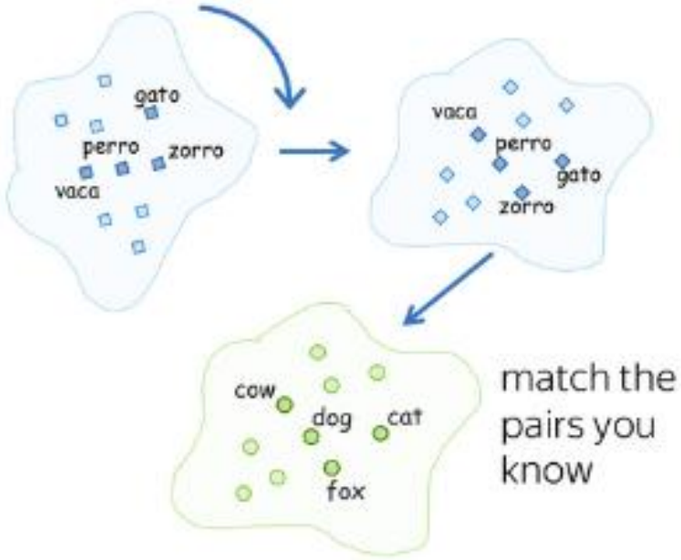
Step 1:

- train embeddings for each language



Step 2:

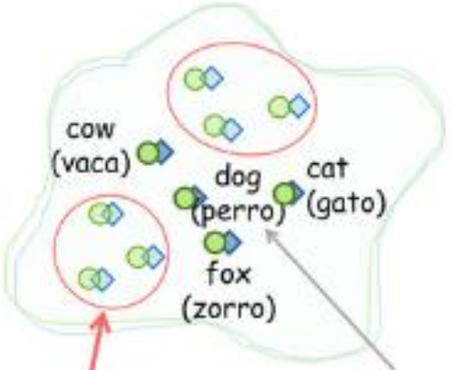
- linearly map one embeddings to the other to match words from the dictionary



match the pairs you know

Step 3:

- after matching the two spaces, get new pairs from the new matches



old translations – the ones you knew

new translations – you learned them!

https://lena-voita.github.io/nlp_course/word_embeddings.html
https://arxiv.org/pdf/1309.4168.pdf

Контекстные представления слов – Contextualized Word Embeddings

недостатки предыдущих вложений – не учитывают контекст

«Рискую всем банком»
«В банке не работал кондиционер»
«Хранить деньги в банках не стоит»
«На банке сидела муха»

«The bank will not be accepting cash on Saturdays»
«The river overflowed the bank»

Выход:
языковые модели

- embeddings in Tag LM
- CoVe
- ELMo
- Flair

ELMo: Embeddings from Language Models

представление с помощью предтренировки без учителя
biLM обучена на большом корпусе текстов

новое предложение в нашей задаче пропускается через biLM
представление слоя = лк состояний слова

$\Rightarrow$

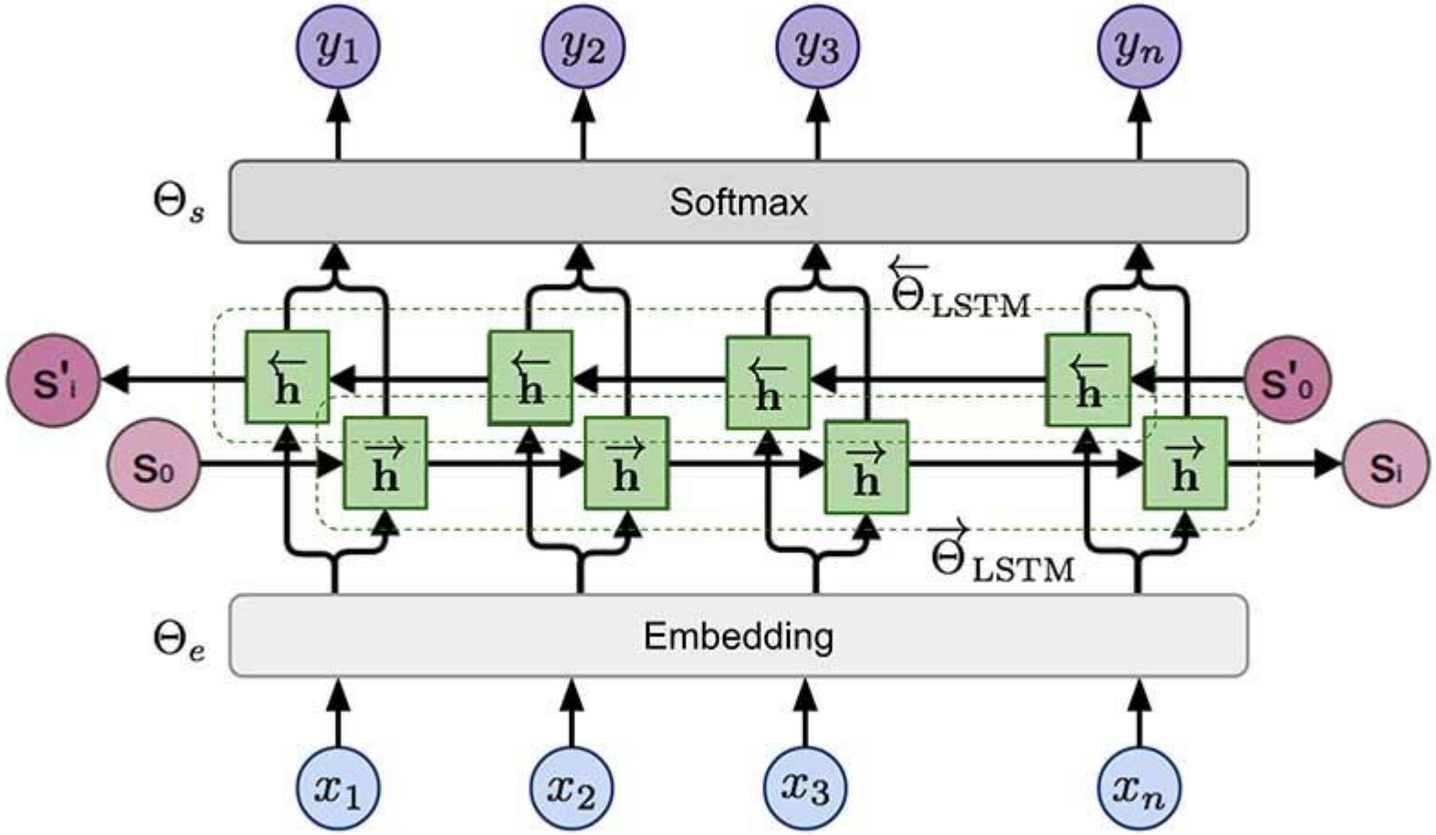
- **зависит от всего предложения**
- **глубокое (зависит от всех слоёв)**
- **есть возможность его обучать (т.к. лк)**

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer «Deep contextualized word representations» // https://arxiv.org/abs/1802.05365

ELMo: Embeddings from Language Models

строим biLM (Bidirectional language model):

$$\sum_k \log p(t_k \mid t_1,\ldots,t_{k-1};\Theta_x,\overrightarrow{\theta}_{\text{LSTM}},\Theta_s)) + \log p(t_k \mid t_{k+1},\ldots,t_n;\Theta_x,\overleftarrow{\theta}_{\text{LSTM}},\Theta_s))$$



Θ_x – **представление токенов**

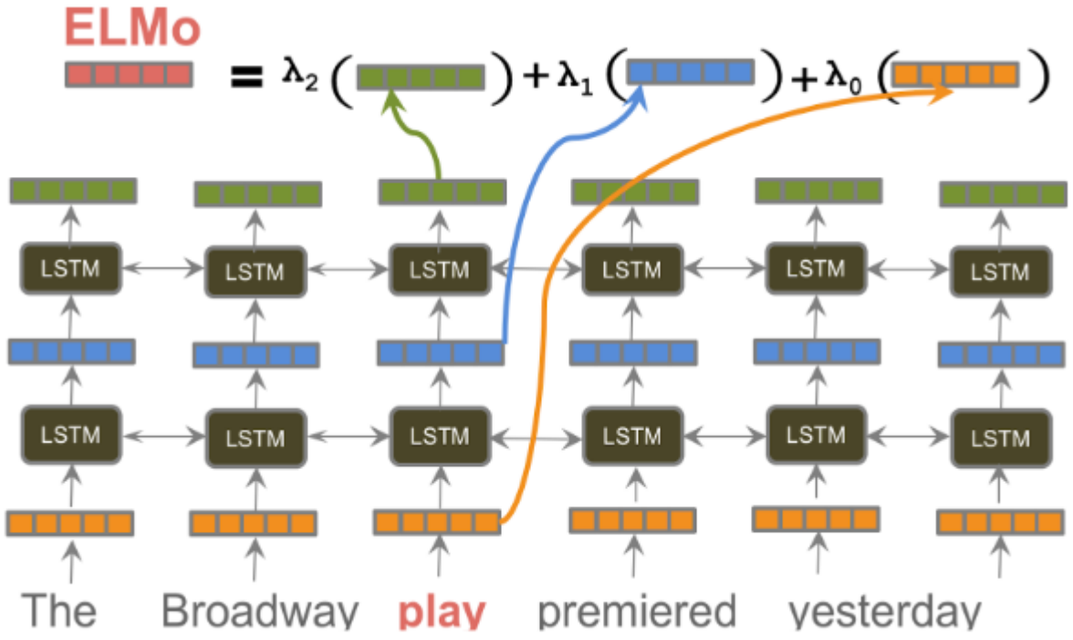
Θ_s – **softmax-слой**

https://www.topbots.com/generalized-language-models-cove-elmo/

ELMo: Embeddings from Language Models

$$\sum_k \log p(t_k \mid t_1,\ldots,t_{k-1};\Theta_x,\overrightarrow{\theta}_{\text{LSTM}},\Theta_s)) + \log p(t_k \mid t_{k+1},\ldots,t_n;\Theta_x,\overleftarrow{\theta}_{\text{LSTM}},\Theta_s))$$

можно затачивать представление под конкретную задачу –
– такую л/к скрытых состояний



$$\text{ELMO}_k = \gamma^{\text{task}} \sum_{l\in\text{layers}} s_j^{\text{task}} [\vec{h}_{k,j}^{\text{LM}}, \overleftarrow{h}_{k,j}^{\text{LM}}]$$

сюда ещё добавляют и выход embedding-слоя

разные слои – разный уровень абстракции
низкие ~ части речи
высокие ~ ответы на вопросы

ELMo: Embeddings from Language Models

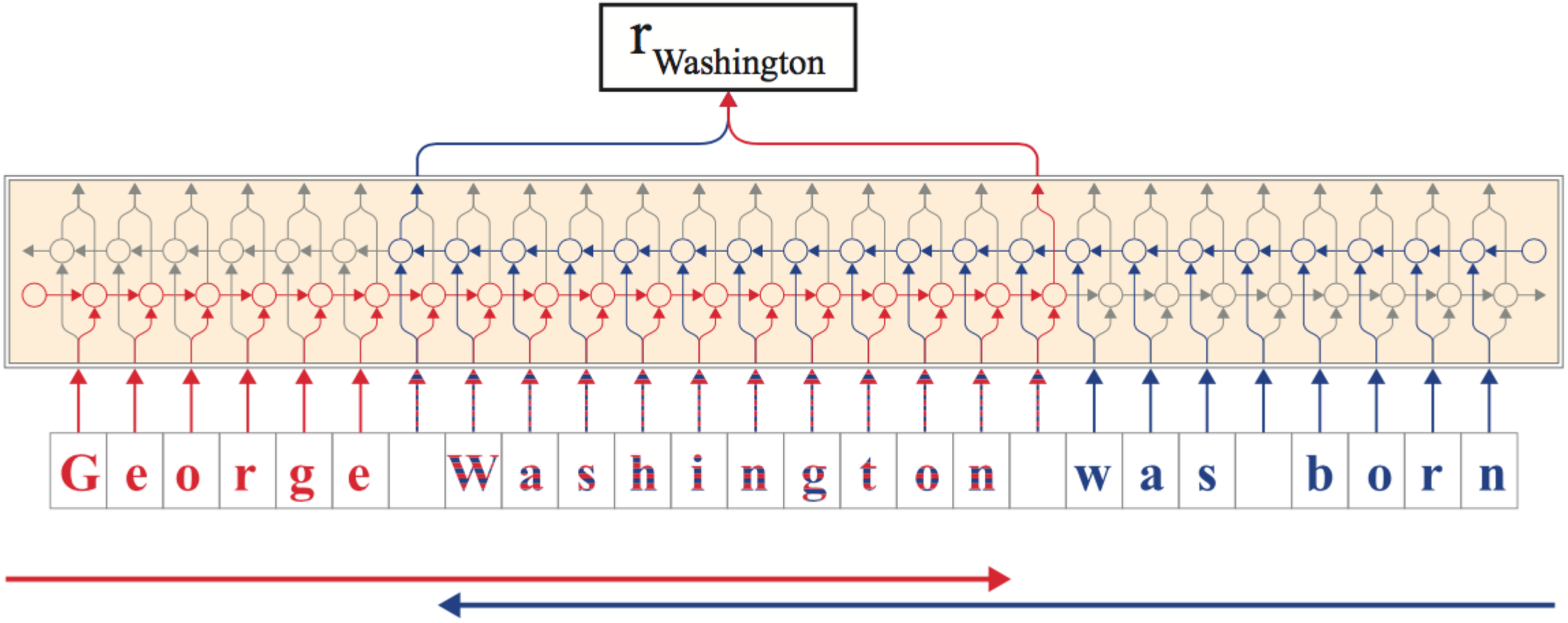
	Source	Nearest Neighbors
GloVe	play	playing, game, games, played, players, plays, player, Play, football, multiplayer
biLM	Chico Ruiz made a spectacular play on Alusik 's grounder {...}	Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play .
	Olivia De Havilland signed to do a Broadway play for Garson {...}	{...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement .

Table 4: Nearest neighbors to "play" using GloVe and the context embeddings from a biLM.

FLAIR: Contextual String Embeddings for Sequence Labelling

учим посимвольную двунаправленную LM (Character-level Language Model)

конкатенируем скрытое состояние последней буквы LM→, первой LM←



Alan Akbik, Duncan Blythe, Roland Vollgraf «Contextual String Embeddings for Sequence Labeling» https://www.aclweb.org/anthology/C18-1139/

FLAIR: Contextual String Embeddings for Sequence Labelling

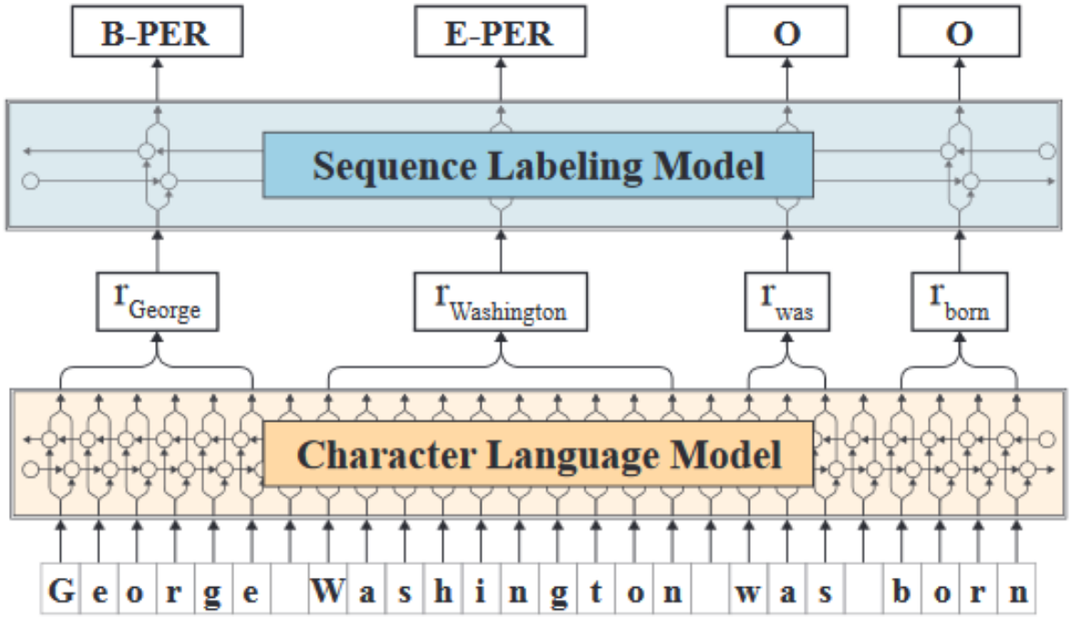


Figure 1: High level overview of proposed approach. A sentence is input as a character sequence into a pre-trained bidirectional character language model. From this LM, we retrieve for each word a contextual embedding that we pass into a vanilla BiLSTM-CRF sequence labeler, achieving robust state-of-the-art results on downstream tasks (NER in Figure).

FLAIR: Contextual String Embeddings for Sequence Labelling

word	context	selected nearest neighbors
Washington	(a) ***Washington*** *to curb support for [..]*	(1) ***Washington*** *would also take [..] action [..]* (2) ***Russia*** *to clamp down on barter deals [..]* (3) ***Brazil*** *to use hovercrafts for [..]*
Washington	(b) *[..] Anthony* ***Washington*** *(U.S.) [..]*	(1) *[..] Carla* ***Sacramento*** *(Portugal) [..]* (2) *[..] Charles* ***Austin*** *(U.S.) [..]* (3) *[..] Steve* ***Backley*** *(Britain) [..]*
Washington	(c) *[..] flown to* ***Washington*** *for [..]*	(1) *[..] while visiting* ***Washington*** *to [..]* (2) *[..] journey to New York City and* ***Washington*** *[..]* (14) *[..] lives in* ***Chicago*** *[..]*
Washington	(d) *[..] when* ***Washington*** *came charging back [..]*	(1) *[..] point for victory when* ***Washington*** *found [..]* (4) *[..] before* ***England*** *struck back with [..]* (6) *[..] before* ***Ethiopia*** *won the spot kick decider [..]*
Washington	(e) *[..] said* ***Washington*** *[..]*	(1) *[..] subdue the never-say-die* ***Washington*** *[..]* (4) *[..] a private school in* ***Washington*** *[..]* (9) *[..] said* ***Florida*** *manager John Boles [..]*

Table 4: Examples of the word "Washington" in different contexts in the CoNLL03 data set, and nearest neighbors using cosine distance over our proposed embeddings. Since our approach produces different embeddings based on context, we retrieve different nearest neighbors for each mention of the same word.

Совместное использование представлений

можно конкатенировать разные представления

использовать одни как инициализации для вычисления других

Итог

свёрточные сети – не только для изображений
можно CNN + RNN

seq2seq – простая и понятная архитектура

внимание – на что «смотрим»
коэффициенты специально считаются

Есть разные виды внимания

Есть классические испытанные способы
Они используются и для получения более продвинутых представлений

Есть способы учёта контекста
дальше будем ещё с этим работать

Ссылки

Обзор про внимание

https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html

хорошо тонкости методов расписаны

https://lilianweng.github.io/lil-log/2017/10/15/learning-word-embedding.html

хороший обзор по этой теме

https://lena-voita.github.io/nlp_course/

хороший туториал по w2v

http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/