

# Lecture 9: Intro to NLP, NLP ConvNets, word embeddings

# Natural language processing: syntax level

- POS-tagging

# POS tagging

V|BZ

DT/ The JJ/ quick JJ/ brown NN/ fox NNS/ jumps IN/ over DT/ the JJ/ lazy NN/ dog

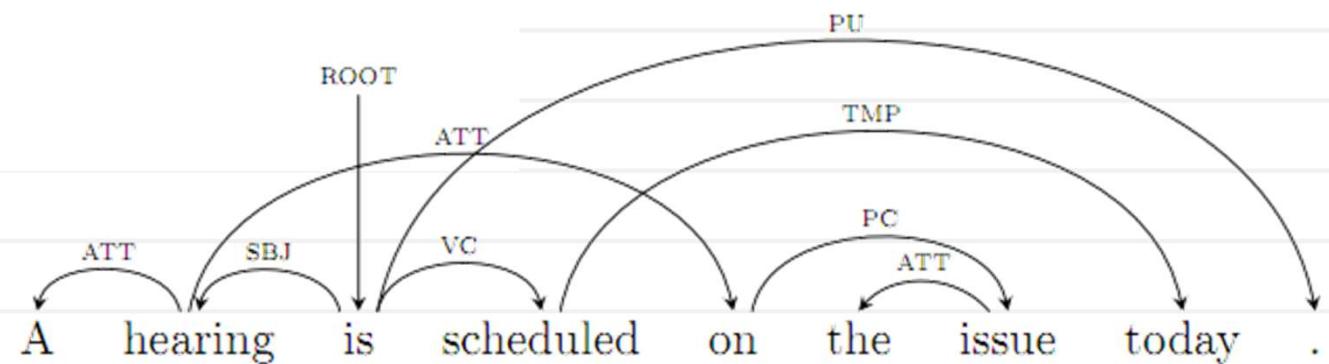
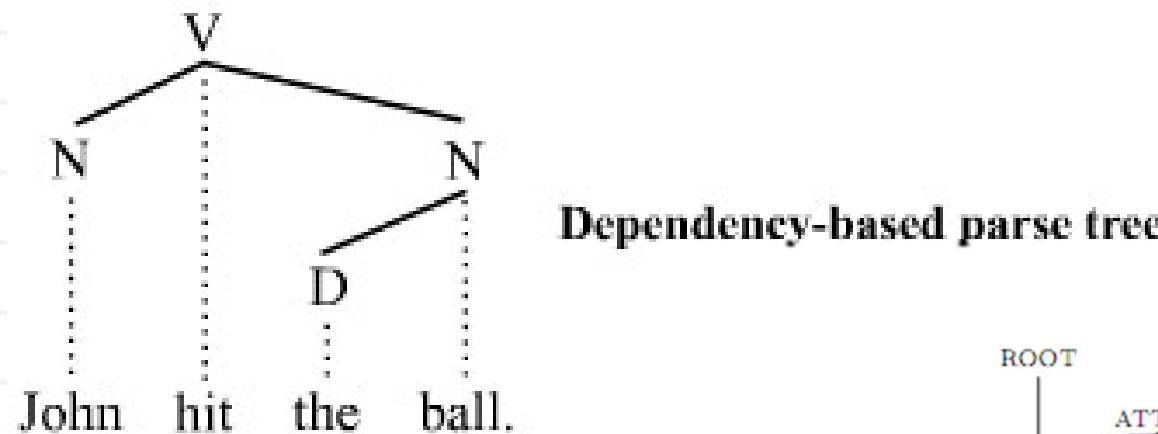
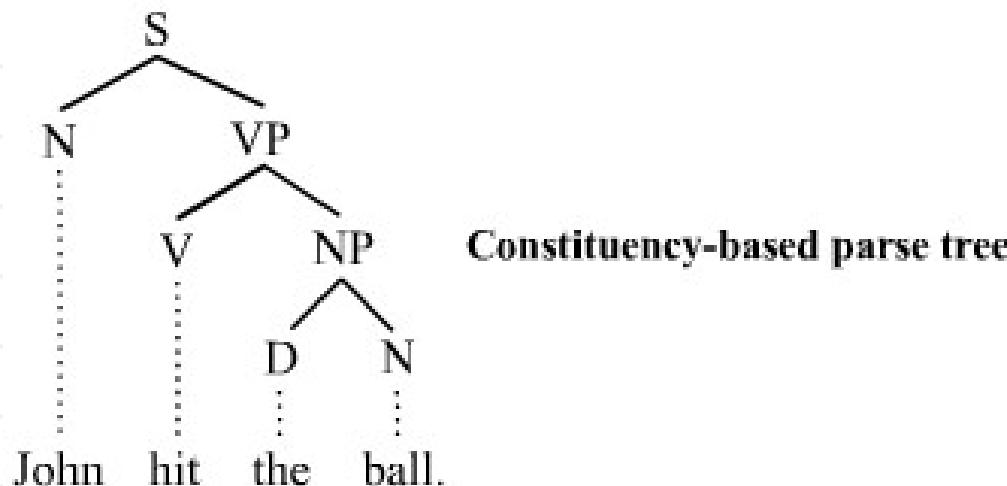
- -RRB- - Right bracket
- CD - Cardinal number
- EX - Existential there
- IN - Preposition
- JJR - Comparative adjective
- LS - List Item Marker
- NN - Singular noun
- NNP - Proper singular noun
- PDT - Predeterminer
- PRP - Personal pronoun
- RB - Adverb
- RBS - Superlative Adverb
- SYM - Symbol
- UH - Interjection
- VBD - Verb, past tense
- VBN - Verb, past participle
- VBZ - Verb, 3rd ps. sing. present
- WP - wh-pronoun
- WRB - wh-adverb
- CC - Coordinating conjunction
- DT - Determiner
- FW - Foreign word
- JJ - Adjective
- JJS - Superlative adjective
- MD - Modal
- NNS - Plural noun
- NNPS - Proper plural noun
- POS - Possessive ending
- PP\$ - Possessive pronoun
- RBR - Comparative adverb
- RP - Particle
- TO - to
- VB - Verb, base form
- VBG - Verb, gerund/present participle
- VBP - Verb, non 3rd ps. sing. present
- WDT - wh-determiner
- WP\$ - Possessive wh-pronoun

<http://cogcomp.cs.illinois.edu/demo/pos/>

# Natural language processing: syntax level

- POS-tagging
- Sentence syntax parsing

# Sentence parsing



# Natural language processing: syntax level

- POS-tagging
- Sentence syntax parsing
- Named entity recognition

# Named entity recognition

Who?

Where?



Mrs. **Green** spoke today in **New York**.  
**Green** chairs the finance committee...

[example from Koller and Friedman textbook]

# Natural language processing: syntax level

- POS-tagging
- Sentence syntax parsing
- Named entity recognition
- Coreference resolution
- Semantic role labeling (SRL)

# Semantic role labeling

*He would n't **accept** anything of value from  
those he was writing about*

**V:** verb

**Ao:** acceptor

**A1:** thing accepted

**A2:** accepted-from

**AM-MOD:** modal

**AM-NEG:** negation

Source: CoNLL-2004

# Natural language processing: semantic level

- Synonyms/paraphrases
- Question answering
- Chatbots/dialog system
- Machine translation
- Text summarization
- ....

# Word-based NLP

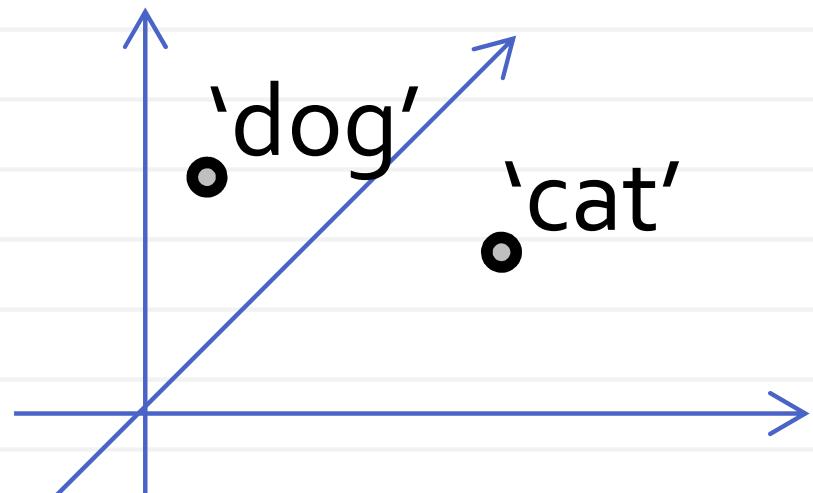
Baseline approach:

- Enumerate words (“lexicon”, “vocabulary”)
- Use *one-hot* encoding

$$x_{\text{cat}}: [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0]$$

$$x_{\text{kitten}}: [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]$$

$$x_{\text{teapot}}: [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$



**Problem:**  $\|x_{\text{cat}} - x_{\text{kitten}}\| = \|x_{\text{cat}} - x_{\text{teapot}}\|$

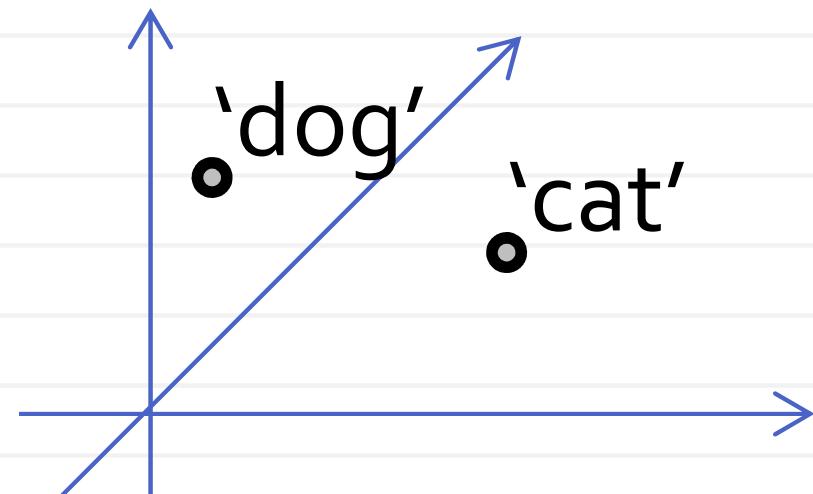
**Solution:** intermediate representations

# Word-based NLP

$x_{\text{cat}}: [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0]$

$x_{\text{kitten}}: [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]$

$x_{\text{teapot}}: [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$



**Problem:** very high dimensionality

**Partial remedy 1:** lemmatize (“*Dogs* -> *dog*”, “*I’m*” -> “*I be*”)

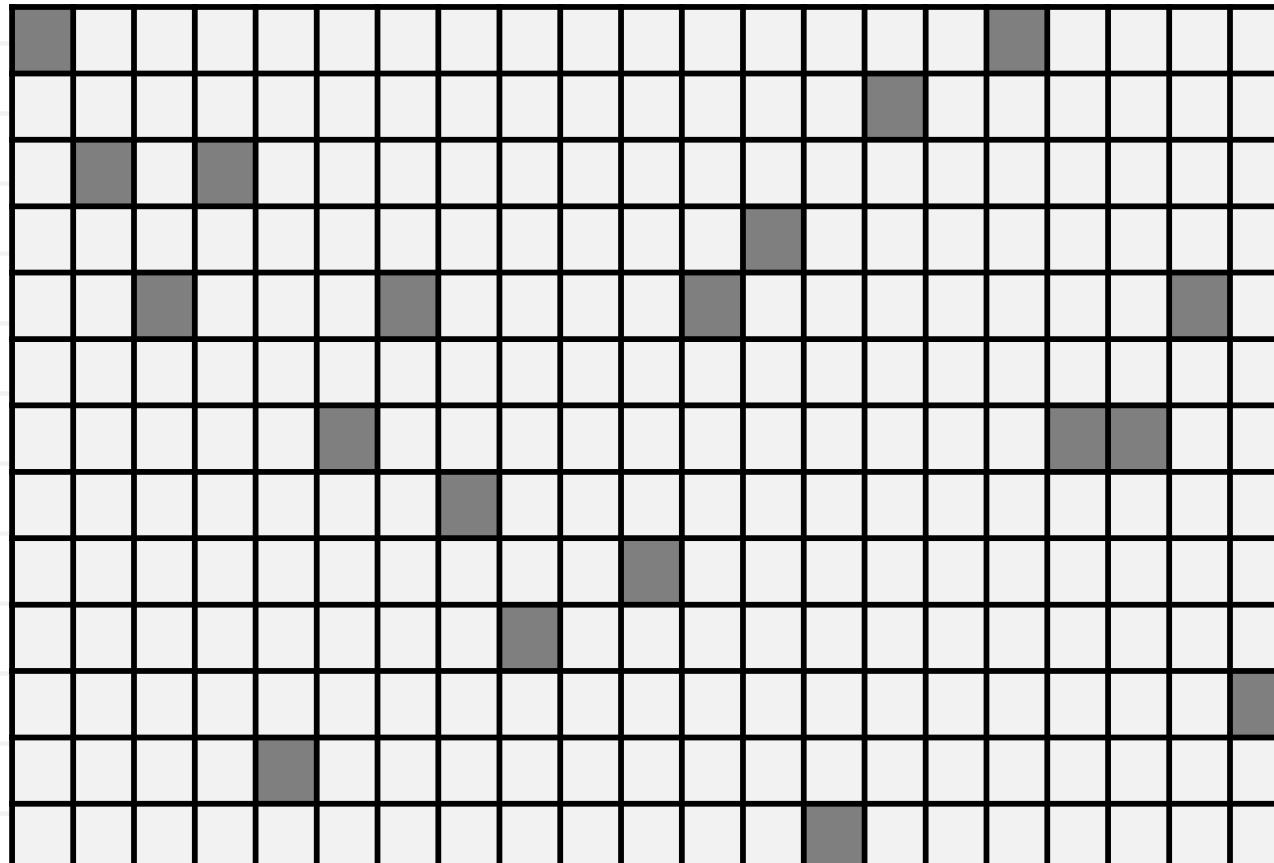
**Side-effect 1:** lemmatization loses information

**Partial remedy 2:** replace infrequent words with “OOV”

**Side-effect 2:** infrequent words are often very informative

# Character-level NLP

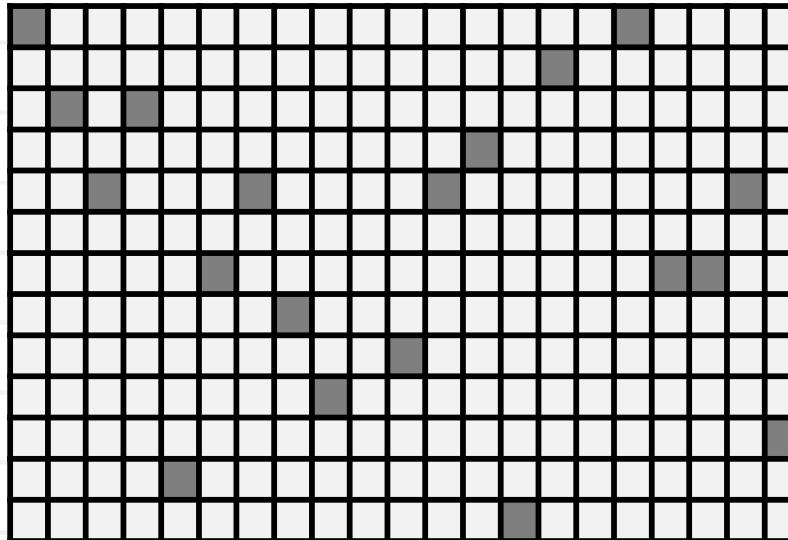
A cat sat on the mat.



symbols

70 rows  
(alphabetic,  
digits, main  
punctuation  
symbols)

# Character-level NLP: advantages



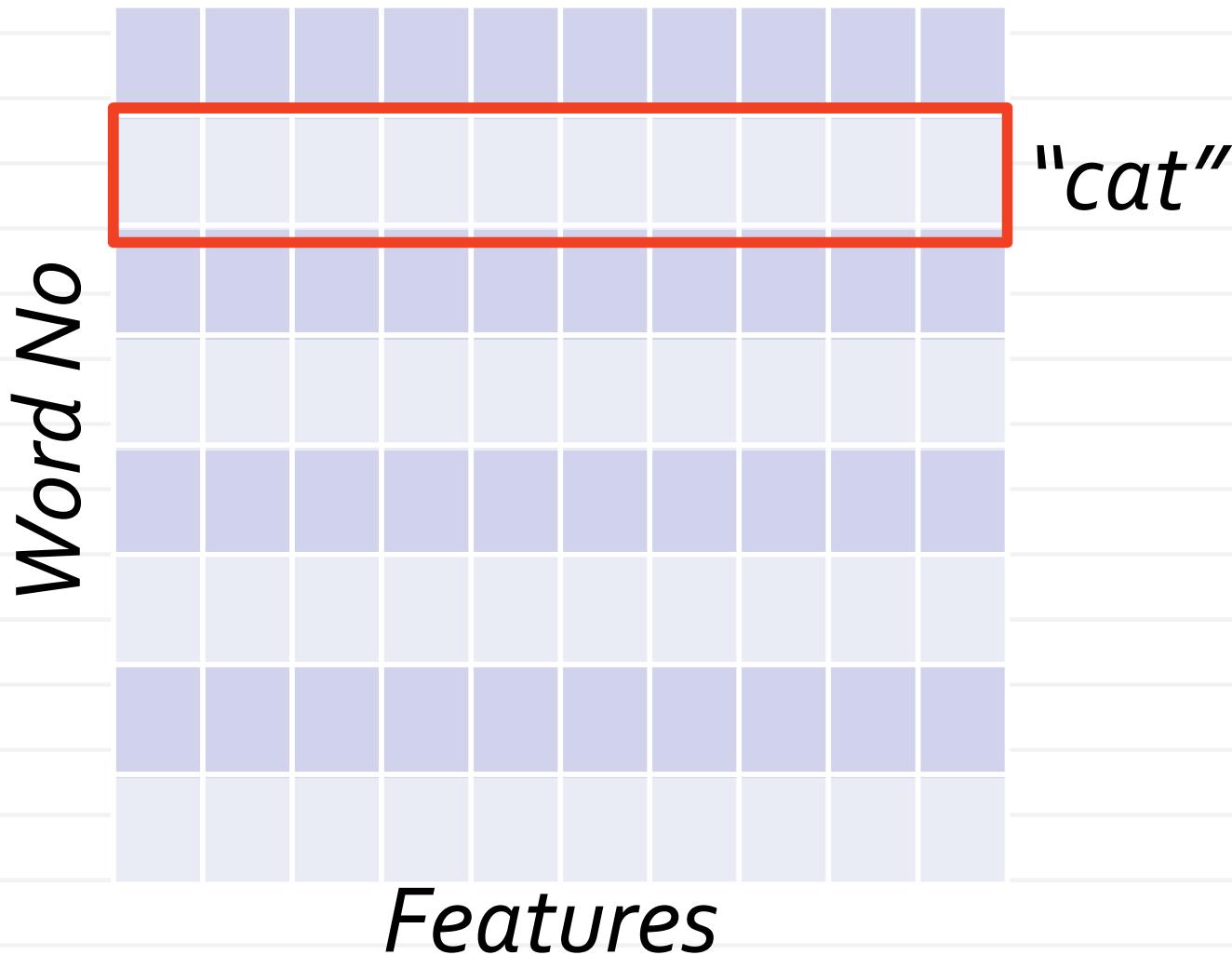
Character-level NLP is good because:

- $school \approx School$
- $colour \approx color$
- $2006 \approx 2007$
- $\|Vyatka - Satka\| < \|Vyatka - San-Francisco\|$

# Vision vs Natural Language Processing

- **Vision based on pixels:** low-level (5D vectors), noisy, unary and pairwise statistics already informative
- **NLP based on words:** unary statistics very informative (much higher-level units), very different nature (large discrete codebooks)
- **NLP based on letters:** low-level, need to consider long combinations to catch meaning (pairwise not enough)

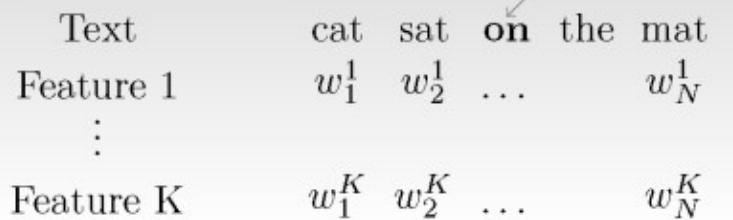
# Embedding layer: main layer for word-level



- Maps integer numbers to vectors
- Can be seen as a multiplicative layer that takes one-hot embeddings as inputs

# Supervised deep-learning for NLP

## Input Window

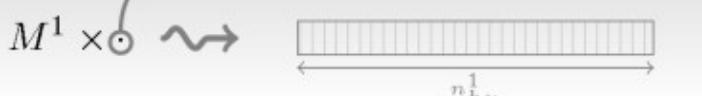


word No,  
 root No,  
 capitalization  
 , etc.

## Lookup Table



## Linear



## HardTanh



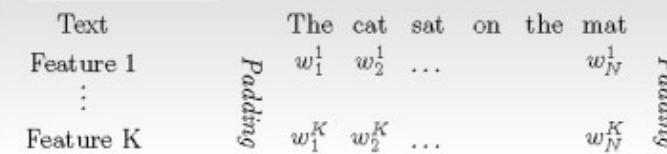
## Linear



*Loss (e.g. softmax)*

[Collobert et al. 11]

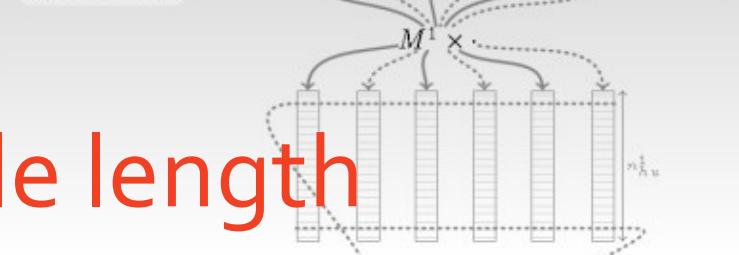
## Input Sentence



## Lookup Table



## Convolution



## Max Over Time



## Linear



## HardTanh



## Linear

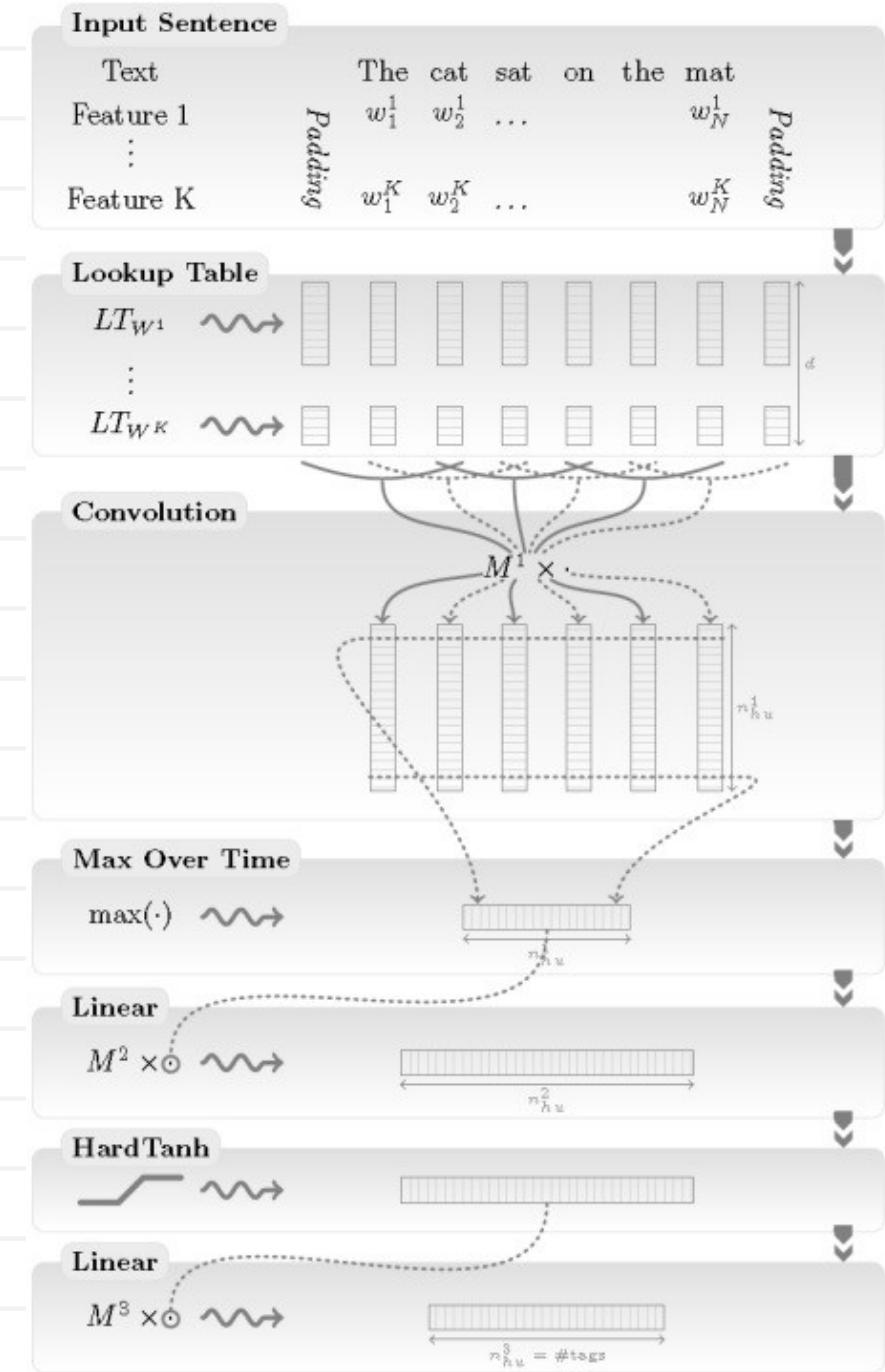


*Loss (e.g. softmax)*

# NLP (almost) from scratch: sentence-based

Remark:

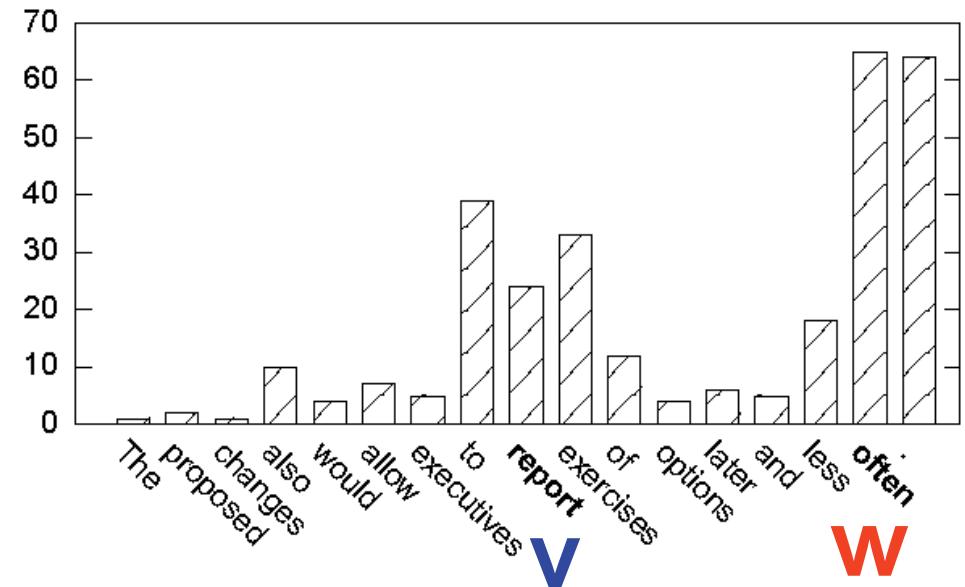
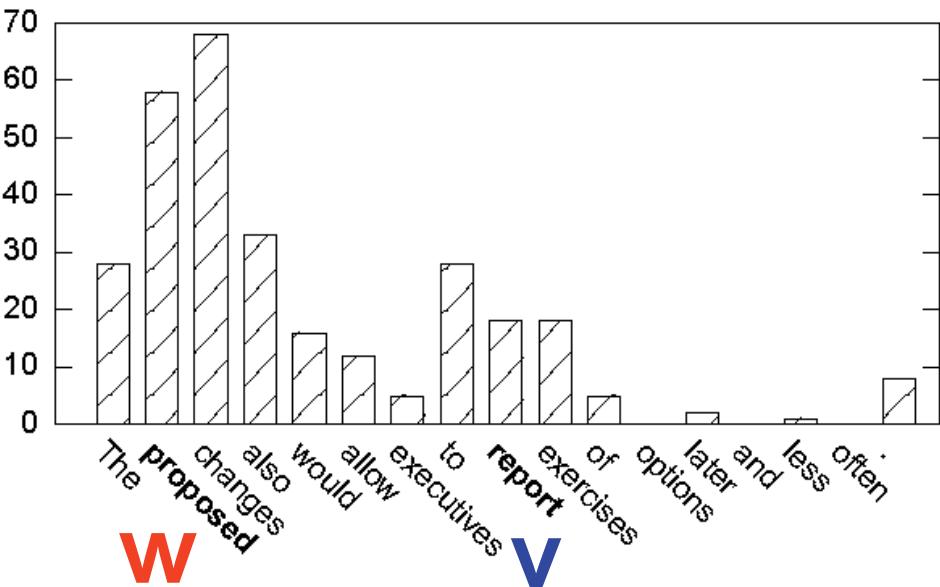
- ConvNet type architecture (weight-sharing)
- For SRL, encode which verb and which word we consider using  $i\text{-}pos_v$  and  $i\text{-}pos_w$  features



[Collobert et al. 11]

# Max-activations

- Max-layer produces 300 dim feature
- Max-layer is independent across dimensions
- How many times a certain word is selected:



# (another) ConvNet for language

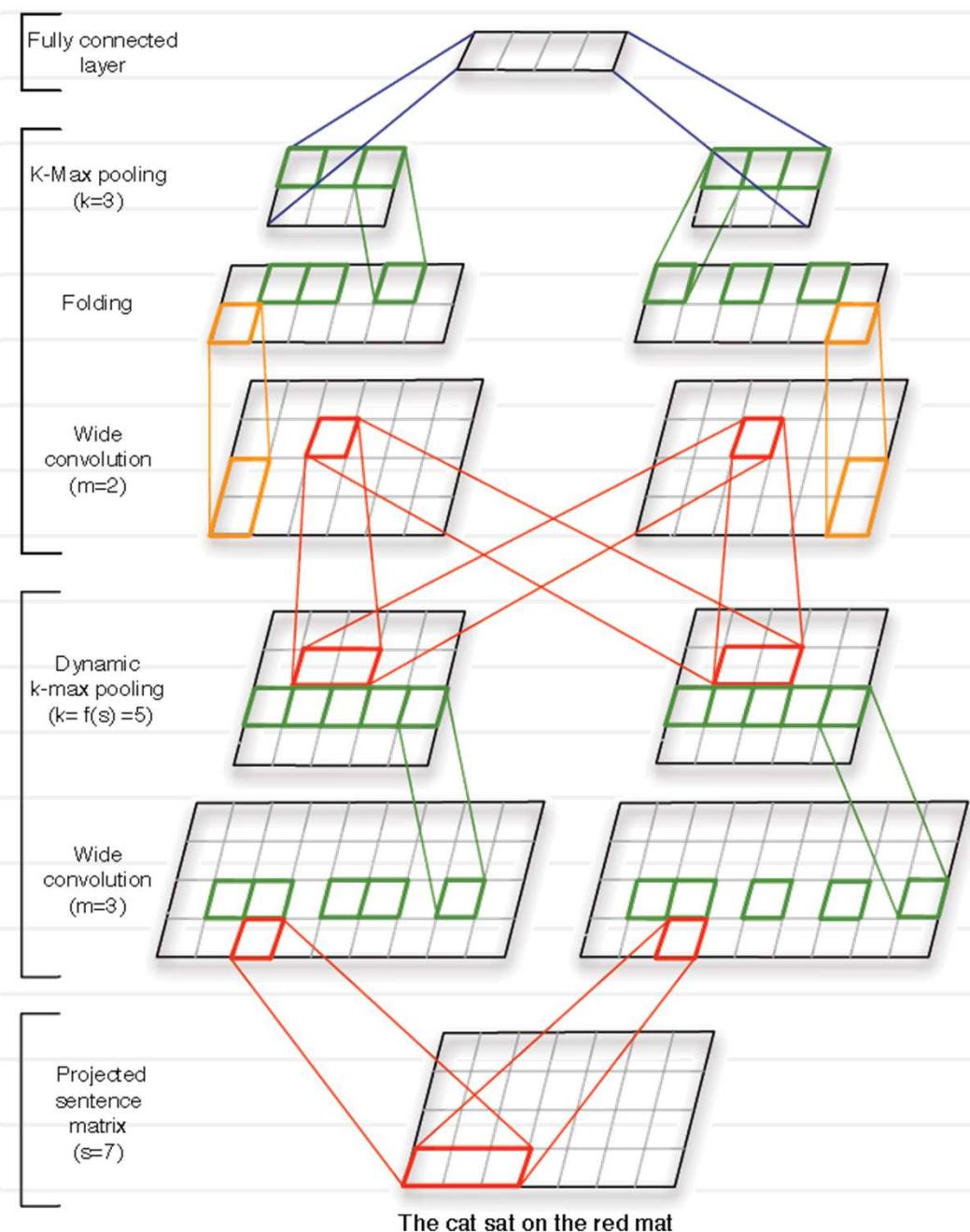
Each word encoded with a vector (e.g. “one-hot”)

Filters act along temporal dimension

“Folding” = sum pooling along embedding direction

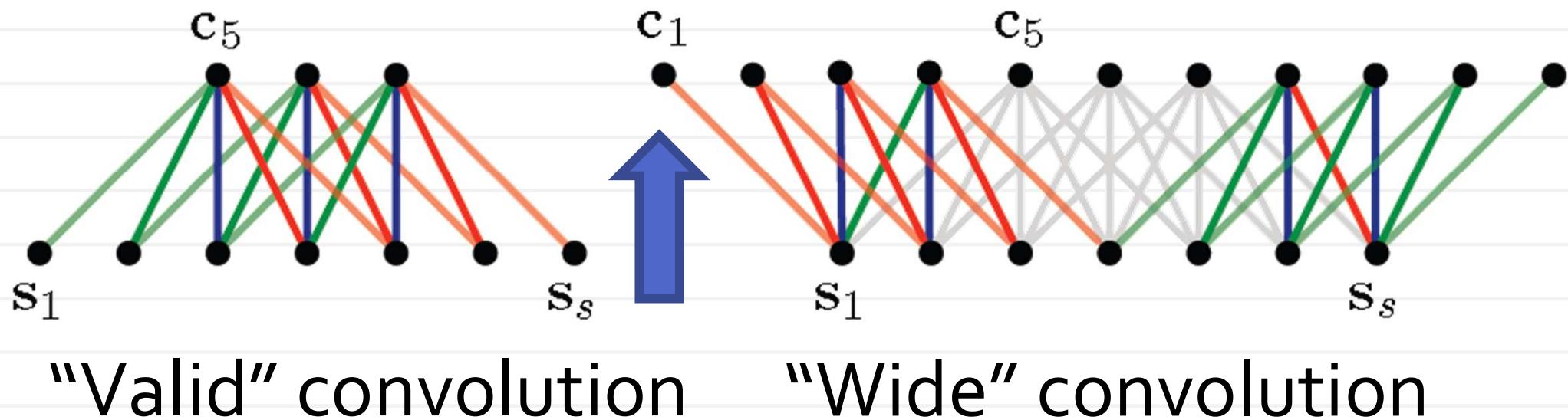
Key idea: k-max pooling

[Kalchbrenner *et al.*, ACL14]



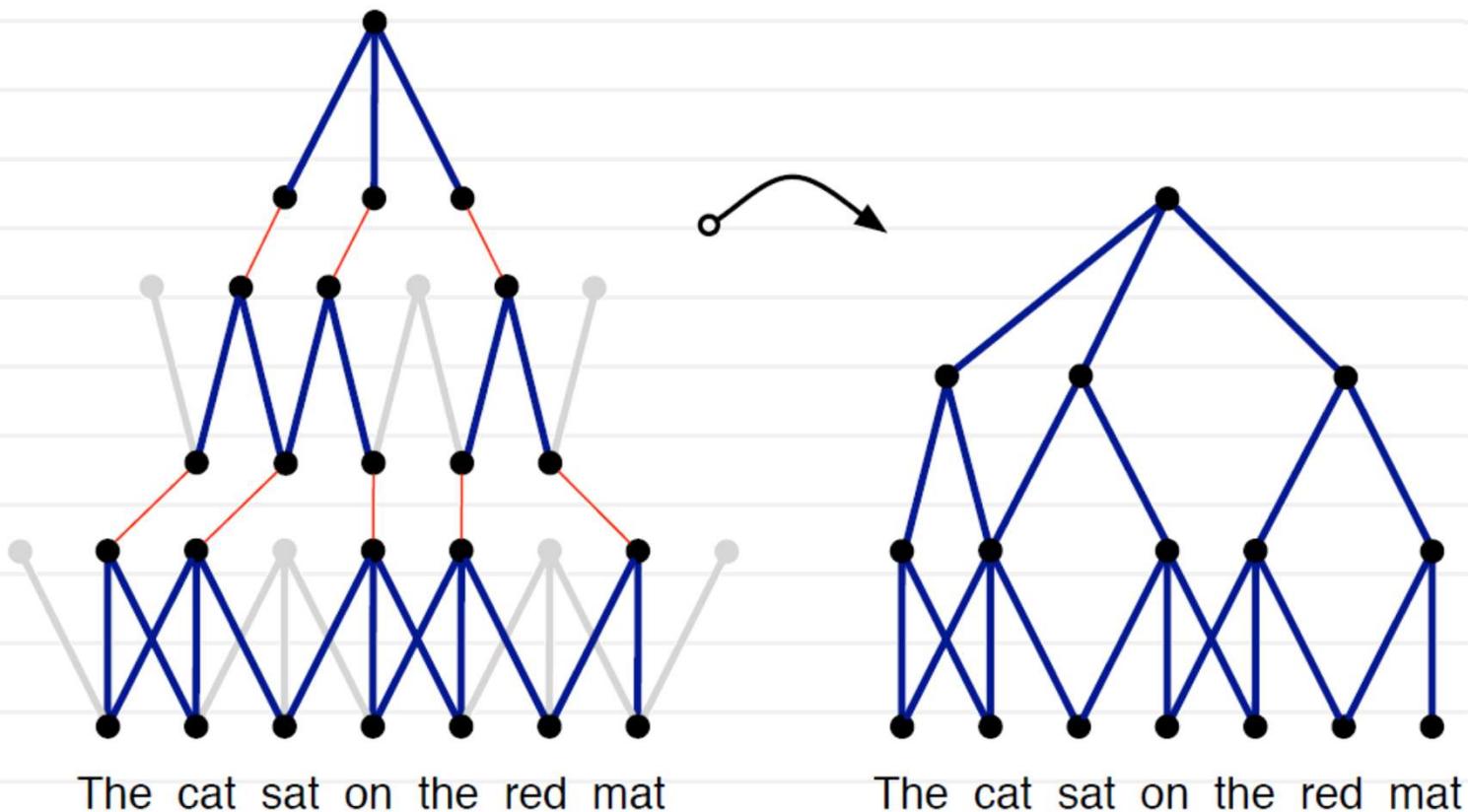
# (another) ConvNet for language

Each word encoded with “one-hot” embedding



[Kalchbrenner *et al.*, ACL14]

# k-max pooling and dynamic k-max pooling

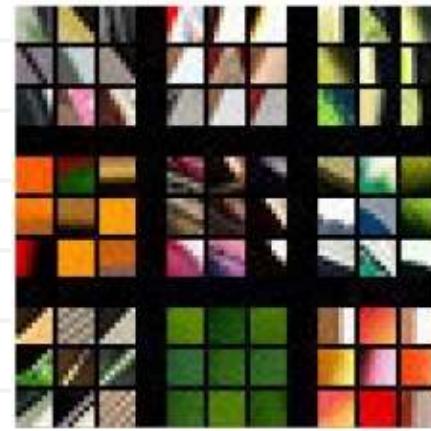


- K-max pooling picks  $k$  largest elements in each channel
- Ordering is kept
- Dynamic k-max pooling allocates  $k$  to each layer using

$$k_l = \max( k_{top}, \lceil \frac{L-l}{L} s \rceil )$$

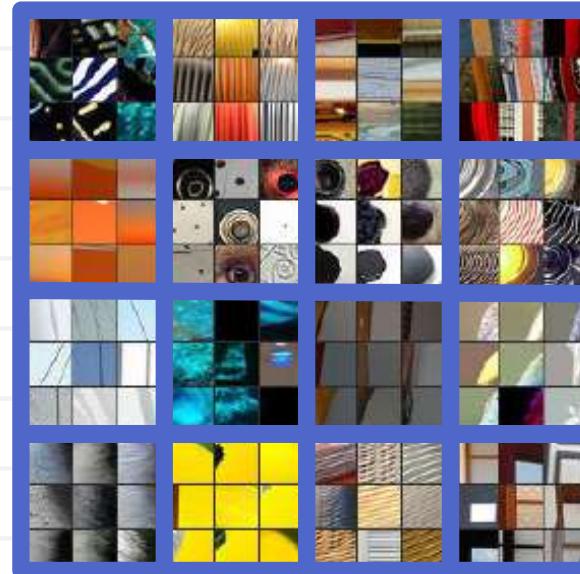
[Kalchbrenner *et al.*, ACL14]

# Visualizing filters



Layer 1

[Zeiler Fergus 14]



Layer 2

## POSITIVE

lovely      comedic      moments and several fine performances  
good      script , good dialogue , funny  
sustains      throughout is daring , inventive and  
well      written , nicely acted and beautifully  
remarkably solid and subtly satirical tour de

## NEGATIVE

, nonexistent plot and pretentious visual style  
it fails the most basic test as  
so stupid , so ill conceived ,  
, too dull and pretentious to be  
hood rats butt their ugly heads in

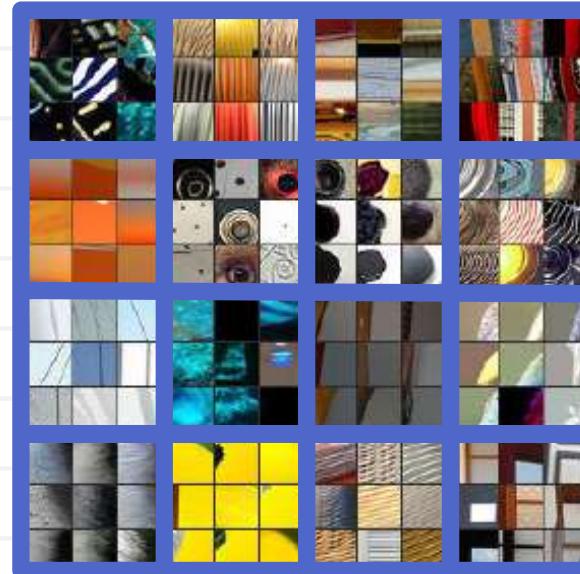
[Kalchbrenner *et al.*, ACL14]

# Visualizing filters



Layer 1

[Zeiler Fergus 14]

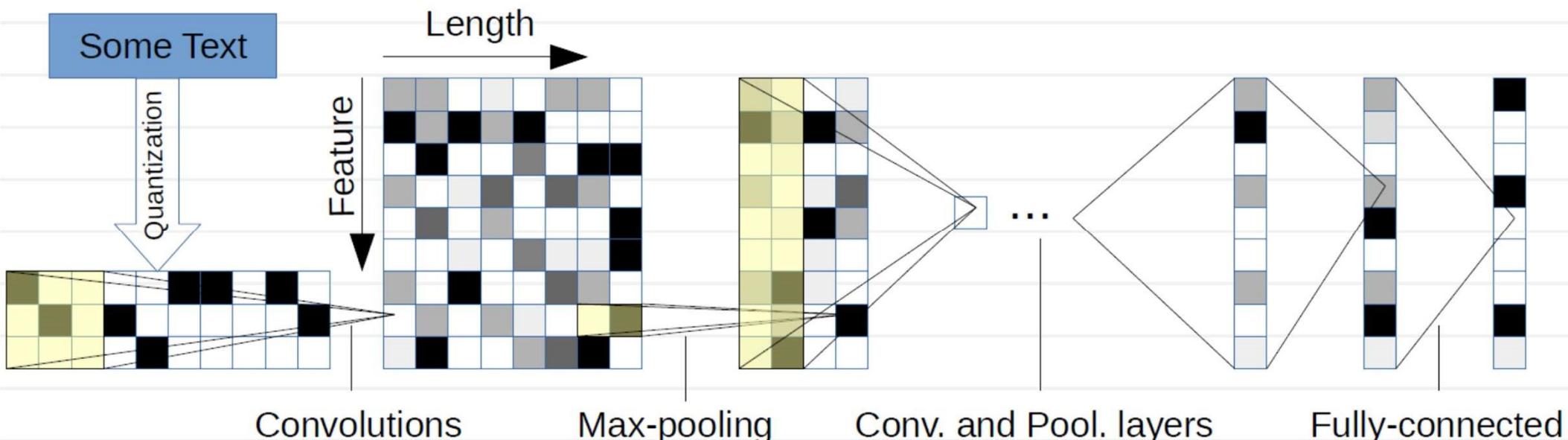


Layer 2

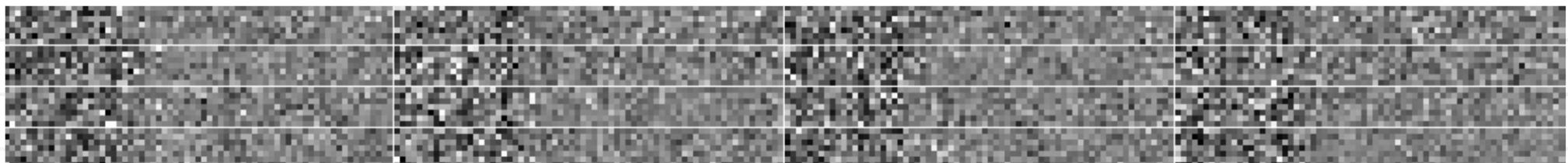
'NOT'					
n't	have	any	huge	laughs	in
no	movement	,	no	,	not
n't	stop	me	from	enjoying	much
not	that	kung	pow	is	of
not	a	moment	that	is	funny
					false
'TOO'					
,	too	dull	and	pretentious	be
either	too	serious	or	too	lighthearted
too	slow	,	too	long	,
feels	too	formulaic	and	too	and
is	too	predictable	and	too	familiar
					to
					self
					conscious

[Kalchbrenner *et al.*, ACL14]

# Character-level ConvNet



- Trained for classification end-to-end
- All reviews are clipped to 1014 characters



Learned features at the 1<sup>st</sup> layer

[Zhang et al.15]

# Character-level ConvNet: evaluation

Dataset	Classes	Train Samples	Test Samples	Epoch Size
AG's News	4	120,000	7,600	5,000
Sogou News	5	450,000	60,000	5,000
DBPedia	14	560,000	70,000	5,000
Yelp Review Polarity	2	560,000	38,000	5,000
Yelp Review Full	5	650,000	50,000	5,000
Yahoo! Answers	10	1,400,000	60,000	10,000
Amazon Review Full	5	3,000,000	650,000	30,000
Amazon Review Polarity	2	3,600,000	400,000	30,000

- Sentiment prediction and topic prediction
- Data augmentation using synonyms  
thesaurus

[Zhang et al.15]

# Character-level ConvNet: evaluation

Larger →

Model	AG	Sogou	DBP.	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. I
BoW	11.19	7.15	3.39	7.76	42.01	31.11	45.36	9.60
BoW TFIDF	10.36	6.55	2.63	6.34	40.14	28.96	44.74	9.00
ngrams	7.96	2.92	1.37	<b>4.36</b>	43.74	31.53	45.73	7.98
ngrams TFIDF	<b>7.64</b>	<b>2.81</b>	<b>1.31</b>	4.56	45.20	31.49	47.56	8.46
Bag-of-means	<b>16.91</b>	<b>10.79</b>	<b>9.55</b>	<b>12.67</b>	<b>47.46</b>	<b>39.45</b>	<b>55.87</b>	<b>18.39</b>
Word-level CNN	LSTM	13.94	4.82	1.45	5.26	41.83	29.16	40.57
	Lg. w2v Conv.	9.92	4.39	1.42	4.60	40.16	31.97	44.40
	Sm. w2v Conv.	11.35	4.54	1.71	5.56	42.13	31.50	42.59
	Lg. w2v Conv. Th.	9.91	-	1.37	4.63	39.58	31.23	43.75
	Sm. w2v Conv. Th.	10.88	-	1.53	5.36	41.09	29.86	42.50
	Lg. Lk. Conv.	8.55	4.95	1.72	4.89	40.52	29.06	45.95
	Sm. Lk. Conv.	10.87	4.93	1.85	5.54	41.41	30.02	43.66
	Lg. Lk. Conv. Th.	8.93	-	1.58	5.03	40.52	28.84	42.39
	Sm. Lk. Conv. Th.	9.12	-	1.77	5.37	41.17	28.92	43.19
	Lg. Full Conv.	9.85	8.80	1.66	5.25	38.40	29.90	40.89
	Sm. Full Conv.	11.59	8.95	1.89	5.67	38.82	30.01	40.88
	Lg. Full Conv. Th.	9.51	-	1.55	4.88	38.04	29.58	40.54
Char-level CNN	Sm. Full Conv. Th.	10.89	-	1.69	5.42	<b>37.95</b>	29.90	40.53
	Lg. Conv.	12.82	4.88	1.73	5.89	39.62	29.55	41.31
	Sm. Conv.	15.65	8.65	1.98	6.53	40.84	29.84	40.53
	Lg. Conv. Th.	13.39	-	1.60	5.82	39.30	<b>28.80</b>	40.45
	Sm. Conv. Th.	14.80	-	1.85	6.49	40.16	29.84	<b>40.43</b>
								5.67

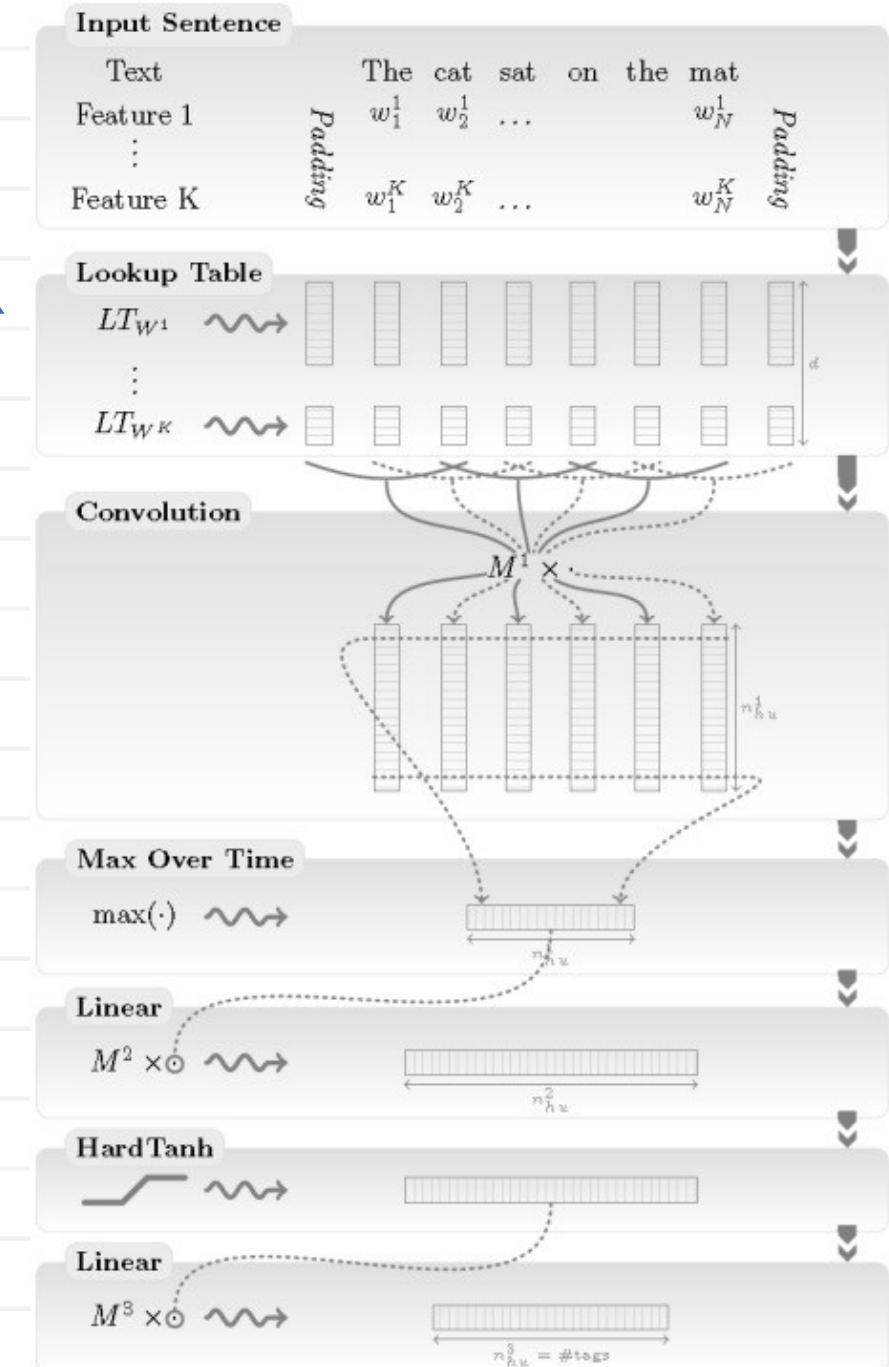
[Zhang et al.15]

# Problem with supervised word embeddings

Weak-spot!

Mapping from one-hot vectors  
to general vectors (aka  
“embedding layer”)

- $O(10 \text{ mln} - 100 \text{ mln}$  parameters)
- We now have most parameters at the deepest level (bad!)

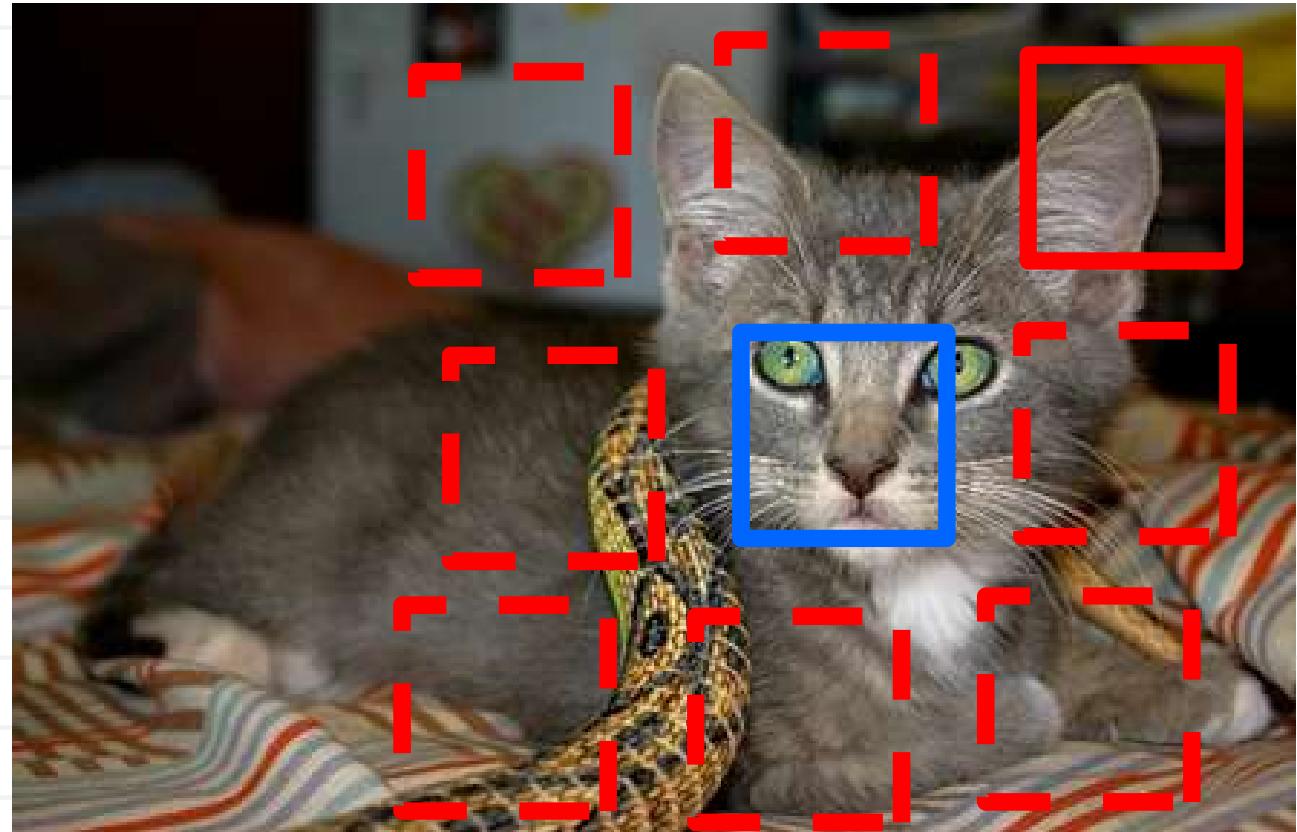
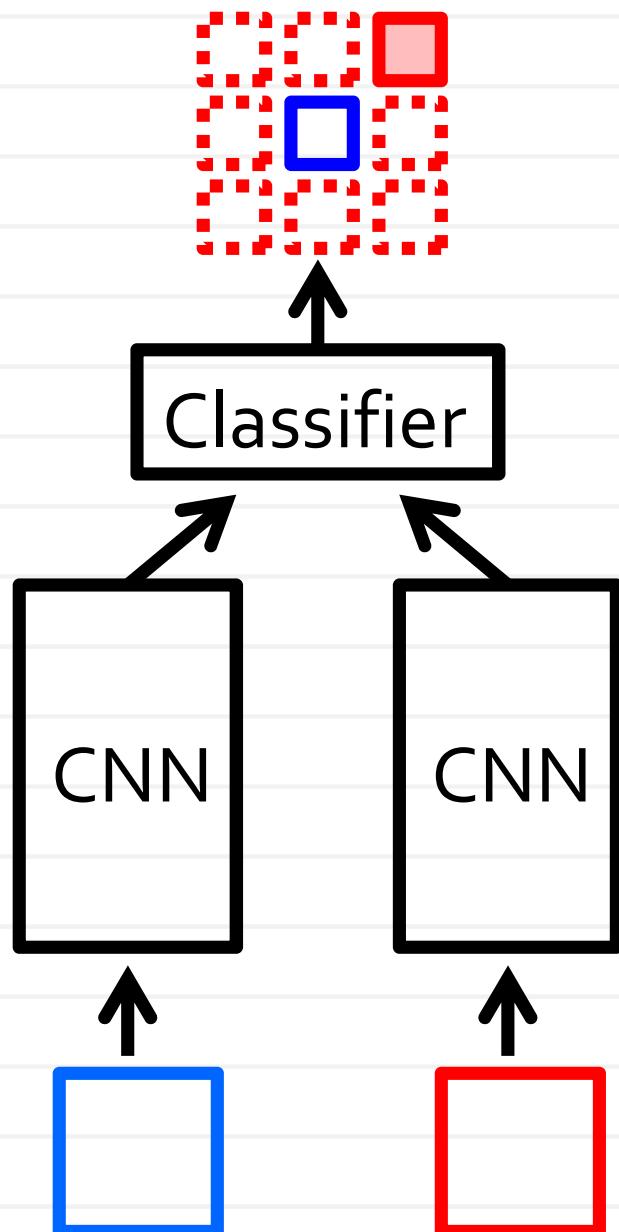


[Collobert et al. 11]

# Predictive learning

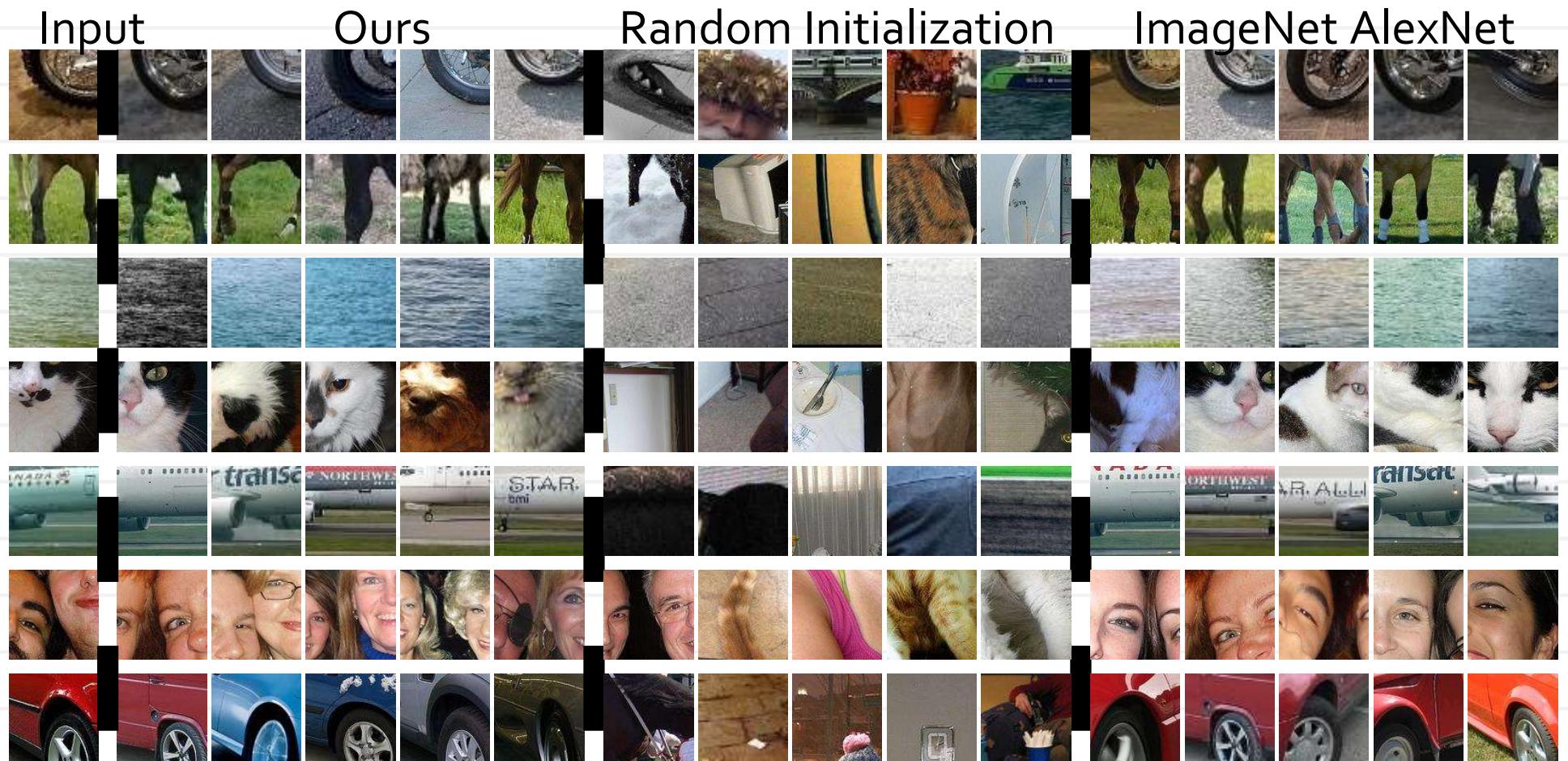
- Given an element predict *nearby* elements (e.g. next, previous, adjacent, etc.)
- Does not require annotated data (“self-supervised”)
- Usually considered as unsupervised, but often works much better than “plain” unsupervised
- Particularly prominent in NLP, but now gaining popularity in many fields

# Predictive learning for still images



[Doersch et al. ICCV15]

# Predictive learning for still images



[Doersch et al. ICCV15]

# Factorization-based word embeddings

the *distributional hypothesis* : similar context = similar meaning

co-occurrence matrix



$n$  :

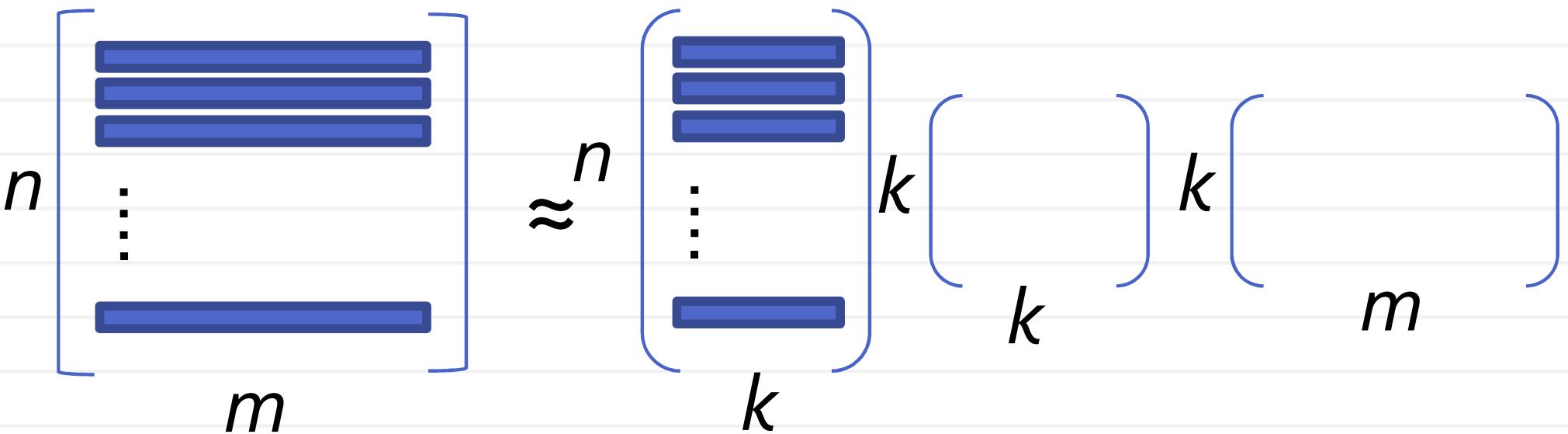


$m$

- Co-occurrences within certain range (*window*)
- Might want to distinguish left and right co-occurrences ( $m = 2n$  in this case)
- We can remove too “rare” dimensions

# Factorization-based word embeddings

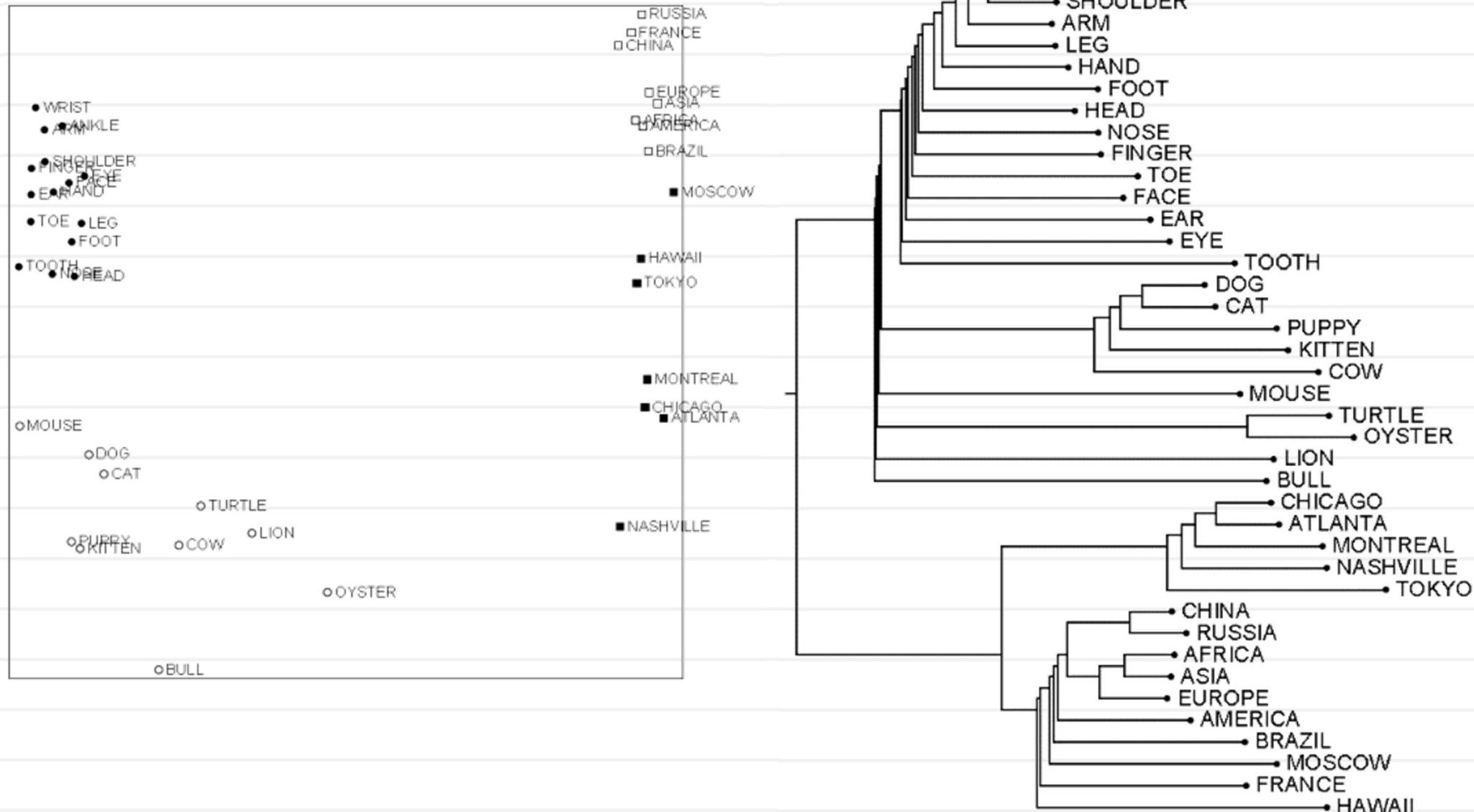
One can reduce dimension of the representation:



- SVD allows to get word vectors of e.g. 100 dimensions
- Removing negatives, renormalizing and square-rooting helps

# Visualizing word space

MDS visualization:



[Ronde et al. 2006]

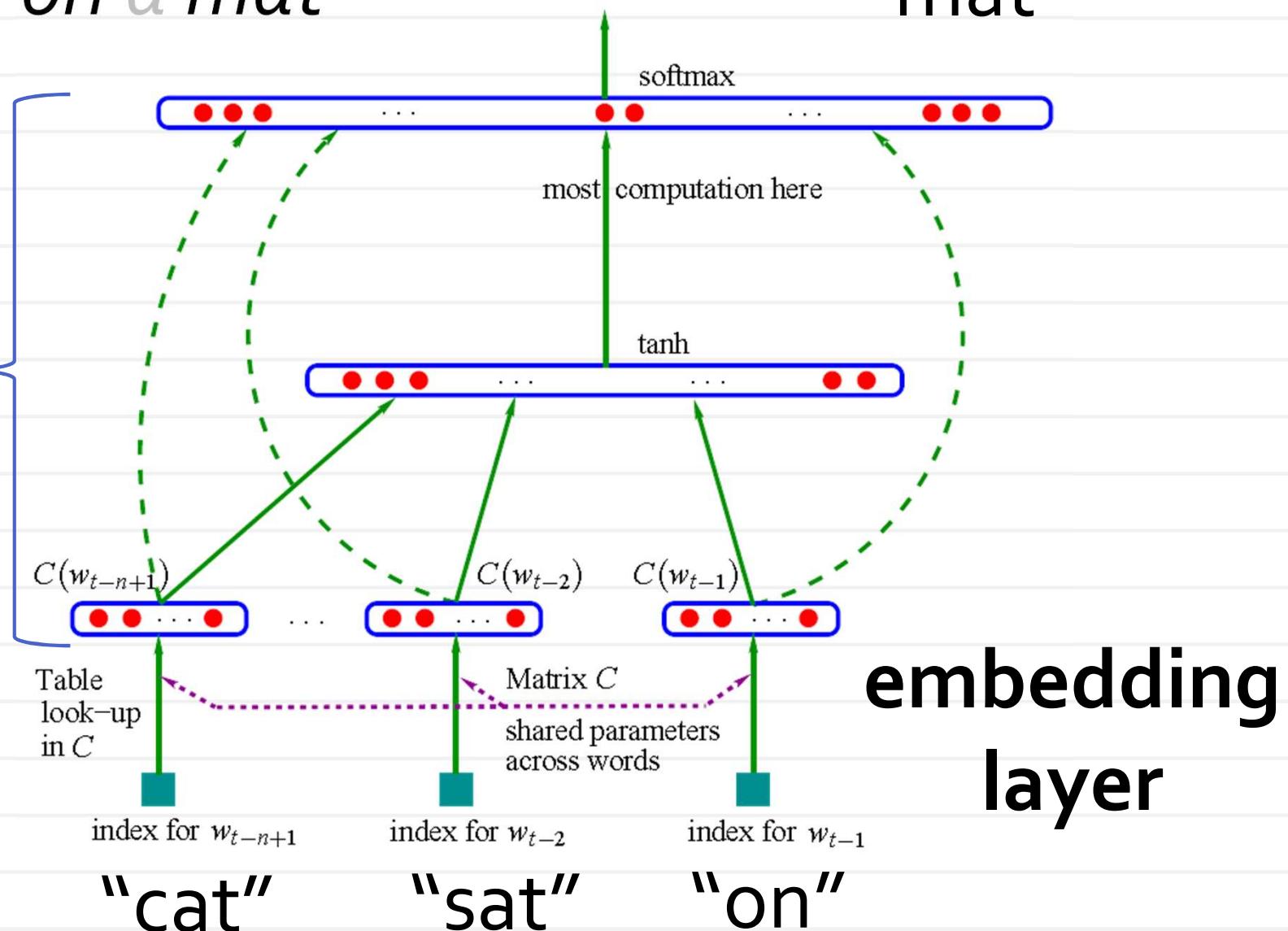
# DL for word embeddings

A cat sat on a mat

$i$ -th output =  $P(w_t = i | \text{context})$

“mat”

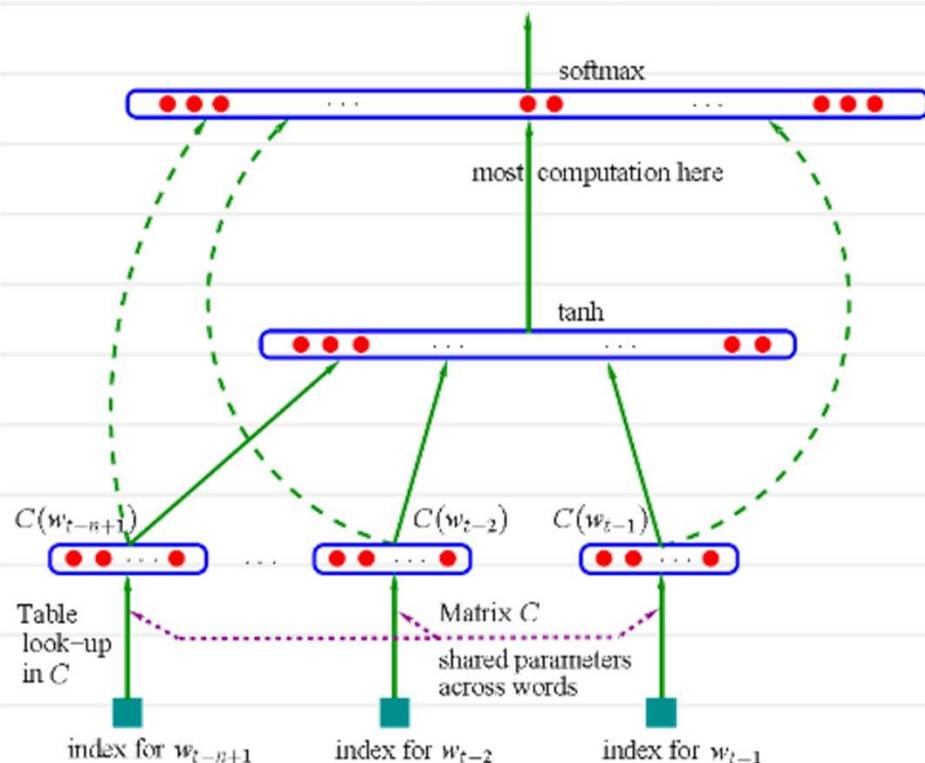
Neural network



[Bengio et al. JMLR03]

# DL for word embeddings

$i$ -th output =  $P(w_t = i \mid \text{context})$



[Bengio et al. JMLR03]

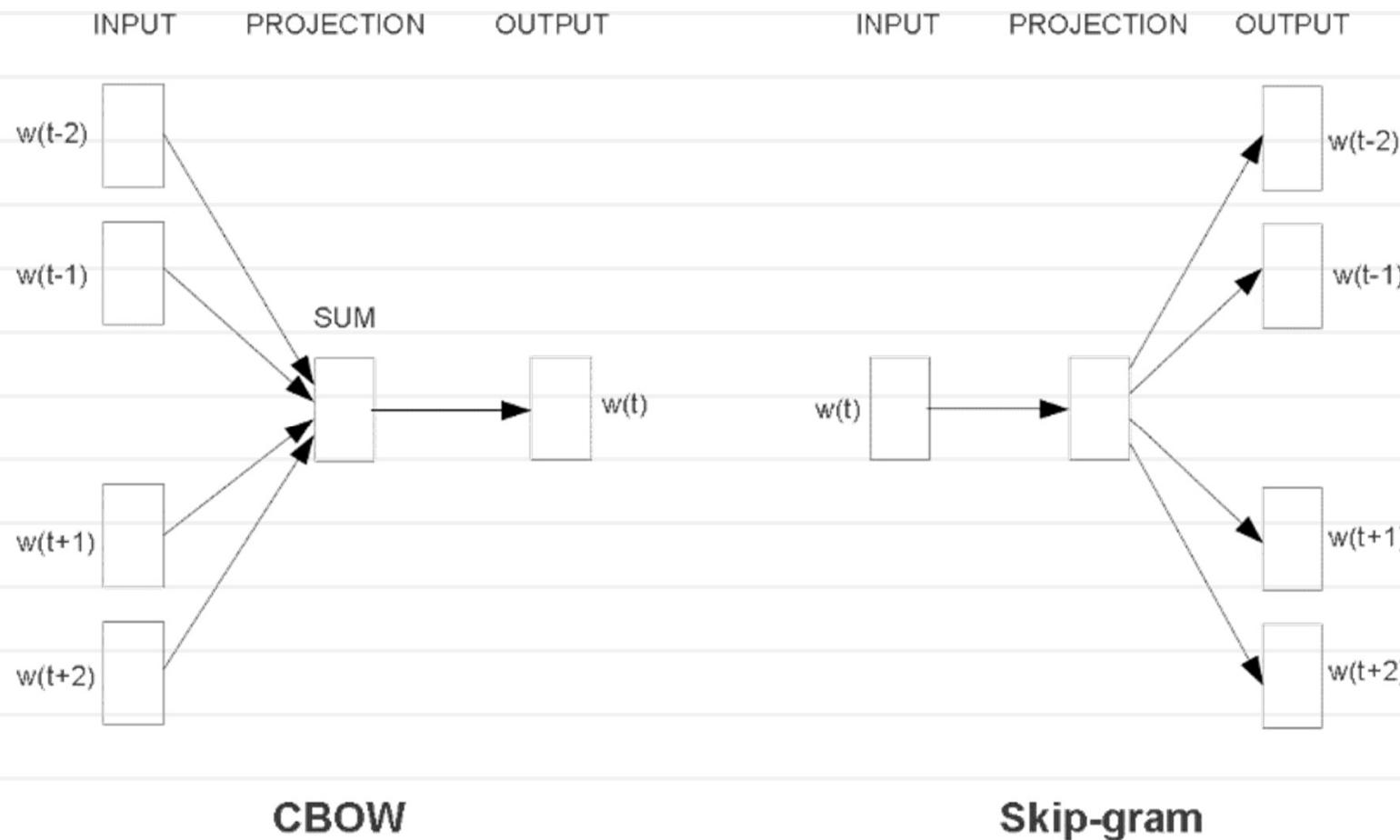
$$x = [C(w_{t-1}), C(w_{t-2}, \dots, C(w_{t-n+1})]$$

$$y = b + Wx + Utanh(d + Hx)$$

$$P(w_t | w_{t-1}, \dots, w_{t-n+1}) = \frac{\exp(y_{w_t})}{\sum_i \exp(y_i)}$$

# New architectures for word embedding

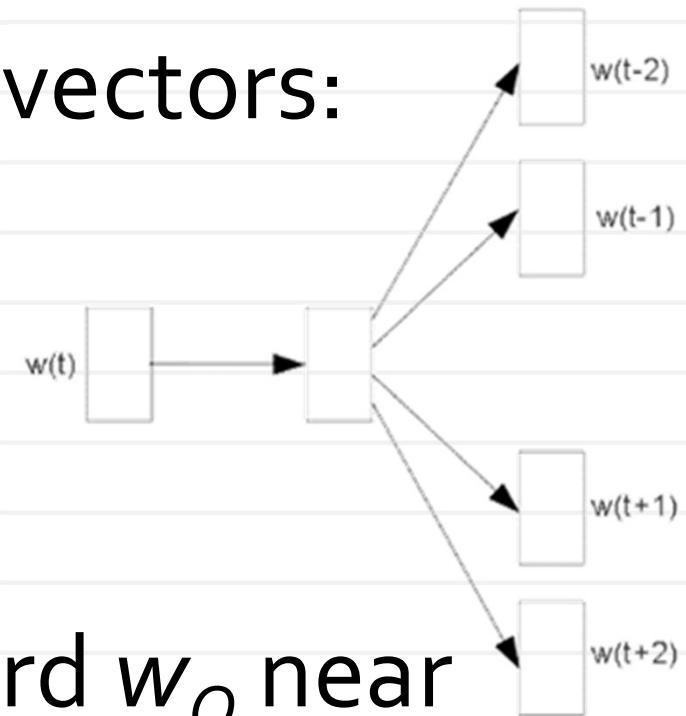
Both architectures learn an embedding of words into vector space (no hidden layer)



[Mikolov et al. 2013]

# Skip-gram model

Each word  $w$  is assigned **two** vectors:  
 $v_w$  and  $v'_w$



The probability to see the word  $w_O$  near the word  $w_I$  is then:

$$p(w_O | w_I) = \frac{\exp({v'_{w_O}}^T v_{w_I})}{\sum_{w=1}^W \exp({v'_{w}}^T v_{w_I})}$$

# Skip-gram model

Computing the probability:

$$p(w_O|w_I) = \frac{\exp({v'_{w_O}}^T v_{w_I})}{\sum_{w=1}^W \exp({v'_{w}}^T v_{w_I})}$$

Overall objective (given a center word, predict surrounding ones):

$$J(v, v') = \frac{1}{T} \sum_{t=1}^T \sum_{\substack{j=-c \\ j \neq 0}}^c \log p(w_{t+j}|w_t; v, v')$$

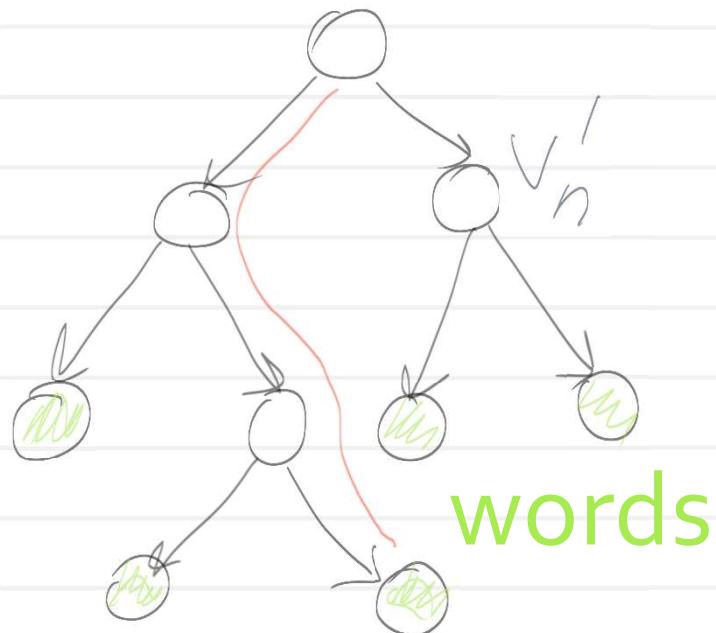
We can now learn embedding on a large corpus

# Evaluating the denominator

$$p(w_O | w_I) = \frac{\exp({v'_{w_O}}^T v_{w_I})}{\sum_{w=1}^W \exp({v'_{w}}^T v_{w_I})}$$

Impractical!

**Idea 1:** *Hierarchical softmax*



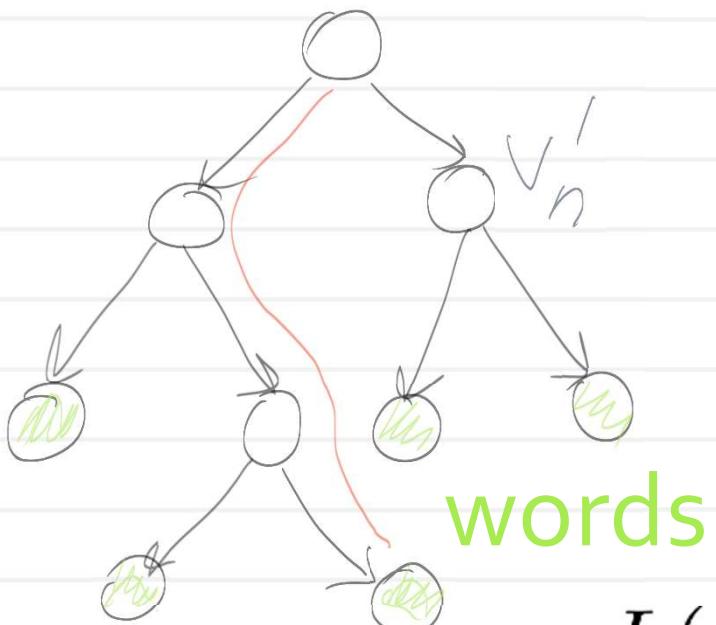
Probability of a word =  
probability of a path

[Mnih & Hinton 2009]

# Hierarchical softmax

[Mnih & Hinton 2009]

Probability of a word =  
probability of a path



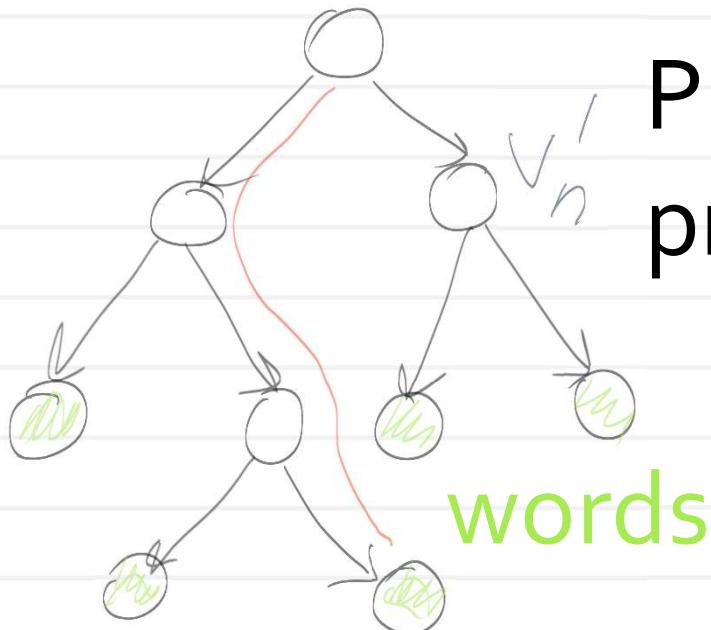
$$P(w|w_I) = \prod_{j=1}^{L(w)-1} \sigma(\pm v'_{n(w,j)} {}^T v_{w_I})$$

+1 for right turn,  
-1 for left turn

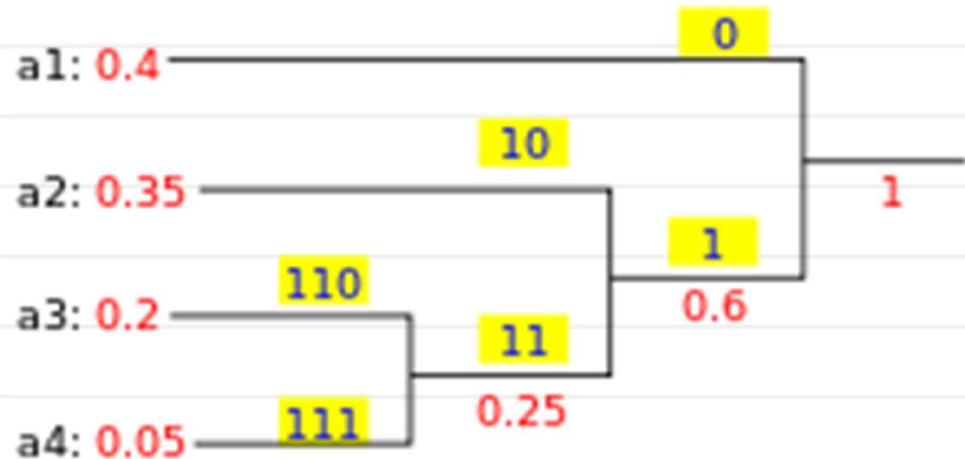
Resulting conditional distribution is normalized!

Proof idea:  $\sigma(x) + \sigma(-x) = 1$

# Hierarchical softmax



Probability of a word =  
probability of a path



- Tree choice affect quality and speed
- Mikolov et al. use *Huffman tree* (maximizing speed): frequent words have short paths

# Negative sampling

Idea 2: Negative sampling

Maximizing:

$$J(v, v') = \sum_{w_I} \left( \frac{1}{T} \sum_{w \in C^+(w_I)} \log \sigma({v'_w}^T v_{w_I}) + \right.$$

$$\left. + \mathbb{E}_{w \sim P(w)} [\log \sigma(-{v'_w}^T v_{w_I})] \right)$$

approximating  
by a few  
samples (2 – 20)

$$\frac{1}{Z} P_{\text{empirical}}^{3/4}(w)$$

# Evaluating using analogies

Semantic

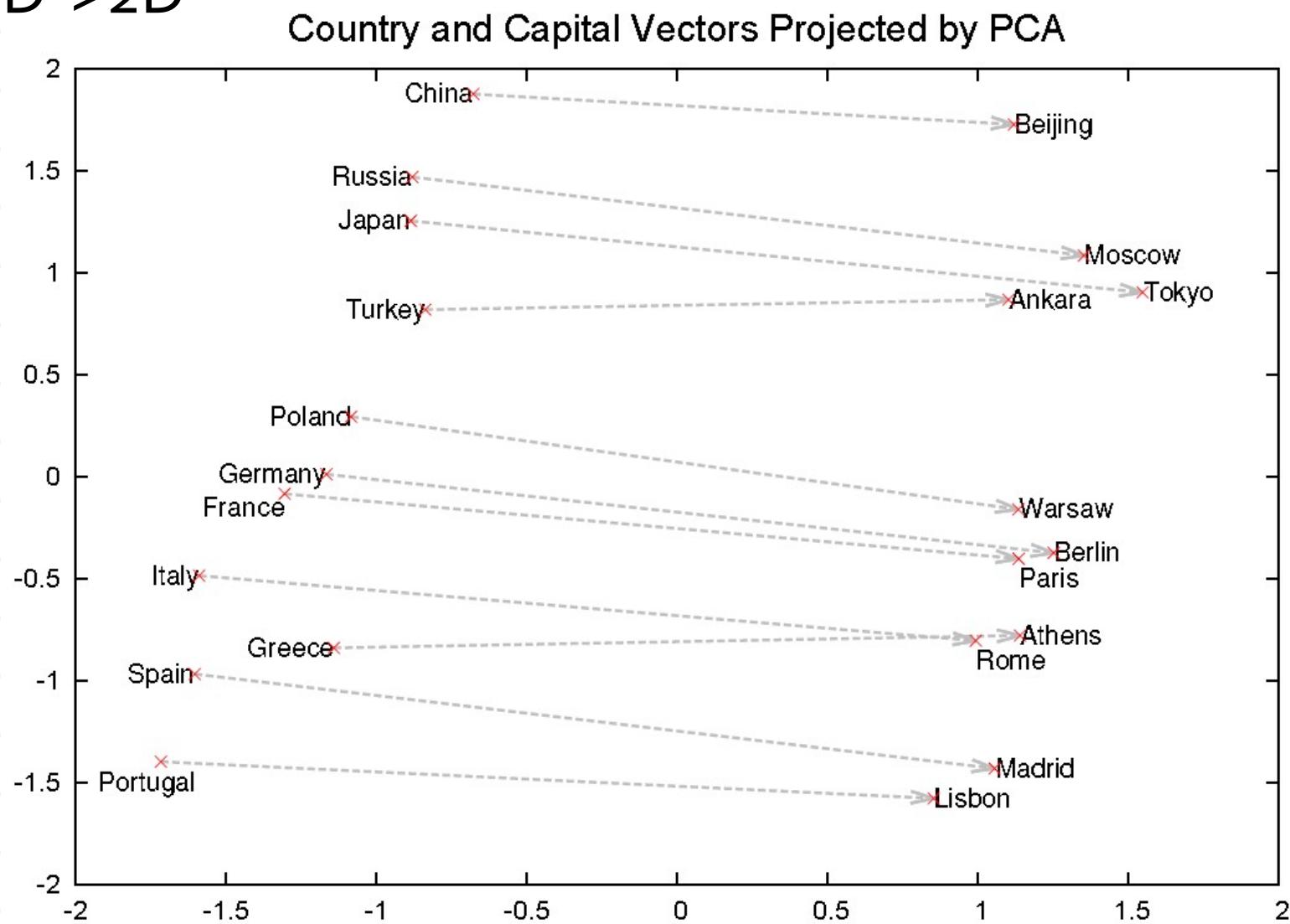
Syntactic

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

$$q = v('Greece') - v('Athens') + v('Oslo')$$

# Word2vec analogies

PCA: 1000D->2D



[Mikolov et al. 2013]

# Ground truth analogies

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

[Mikolov et al. 2013]

# Evaluating using analogies

Model	Vector Dimensionality	Training words	Accuracy [%]			Training time [days]
			Semantic	Syntactic	Total	
3 epoch CBOW	300	783M	15.5	53.1	36.1	1
3 epoch Skip-gram	300	783M	50.0	55.9	53.3	3
1 epoch CBOW	300	783M	13.8	49.9	33.6	0.3
1 epoch CBOW	300	1.6B	16.1	52.6	36.1	0.6
1 epoch CBOW	600	783M	15.4	53.3	36.2	0.7
1 epoch Skip-gram	300	783M	45.6	52.2	49.2	1
1 epoch Skip-gram	300	1.6B	52.2	55.1	53.8	2
1 epoch Skip-gram	600	783M	56.7	54.5	55.5	2.5

[Mikolov et al. 2013]

# Evaluating using analogies

Model	Vector Dimensionality	Training words	Accuracy [%]		
			Semantic	Syntactic	Total
Collobert-Weston NNLM	50	660M	9.3	12.3	11.0
Turian NNLM	50	37M	1.4	2.6	2.1
Turian NNLM	200	37M	1.4	2.2	1.8
Mnih NNLM	50	37M	1.8	9.1	5.8
Mnih NNLM	100	37M	3.3	13.2	8.8
Mikolov RNNLM	80	320M	4.9	18.4	12.7
Mikolov RNNLM	640	320M	8.6	36.5	24.6
Huang NNLM	50	990M	13.3	11.6	12.3
Our NNLM	20	6B	12.9	26.4	20.3
Our NNLM	50	6B	27.9	55.8	43.2
Our NNLM	100	6B	34.2	<b>64.5</b>	50.8
CBOW	300	783M	15.5	53.1	36.1
Skip-gram	300	783M	<b>50.0</b>	55.9	<b>53.3</b>

[Mikolov et al. 2013]

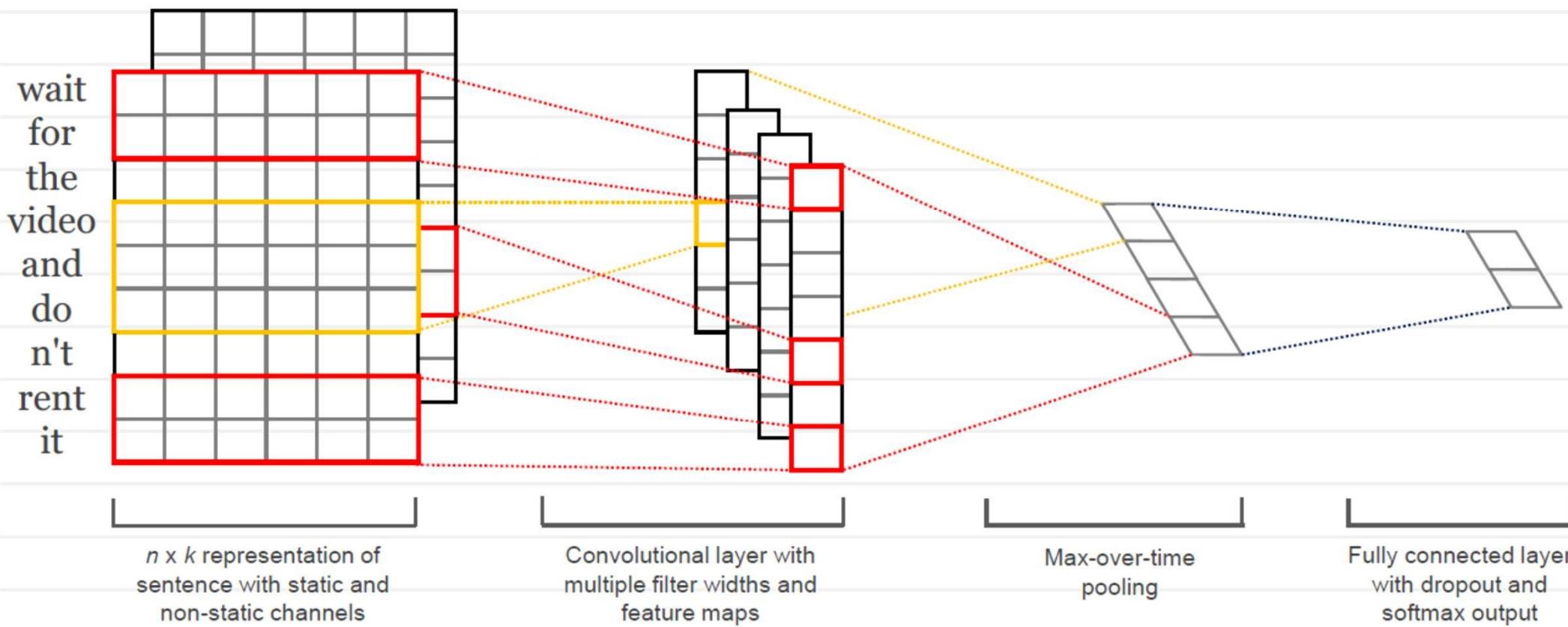
# Evaluating using analogies

Trained on 783M words, 300 dim

Relationship	Example 1	Example 2	Example 3
France - Paris big - bigger	Italy: Rome small: larger	Japan: Tokyo cold: colder	Florida: Tallahassee quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France copper - Cu	Berlusconi: Italy zinc: Zn	Merkel: Germany gold: Au	Koizumi: Japan uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

[Mikolov et al. 2013]

# Word2vec reuse



- Simple ConvNet initialized with word2vec

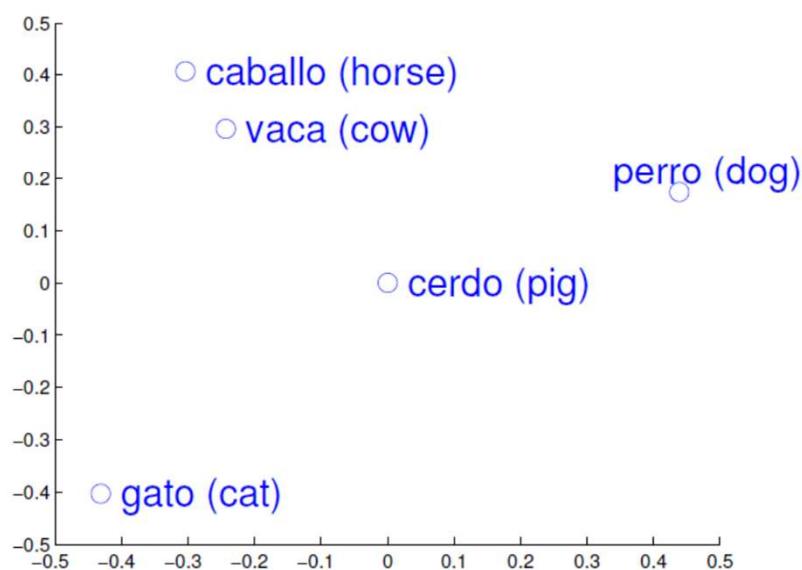
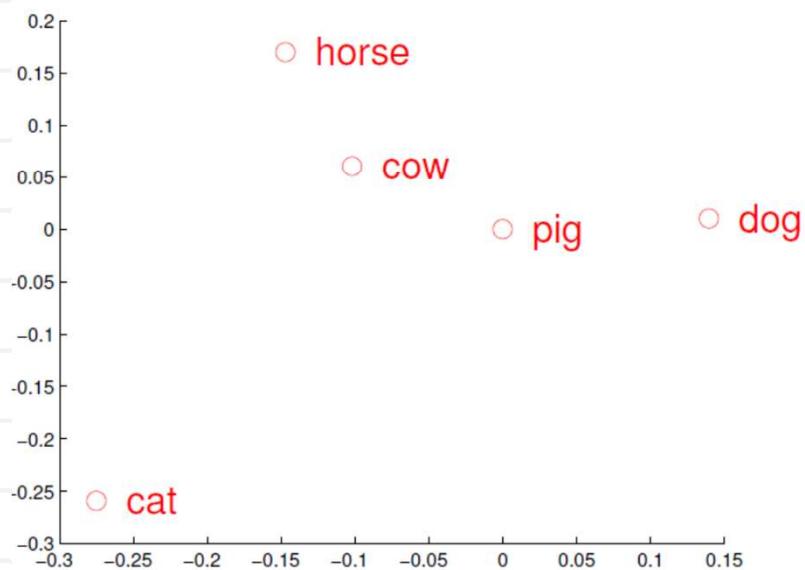
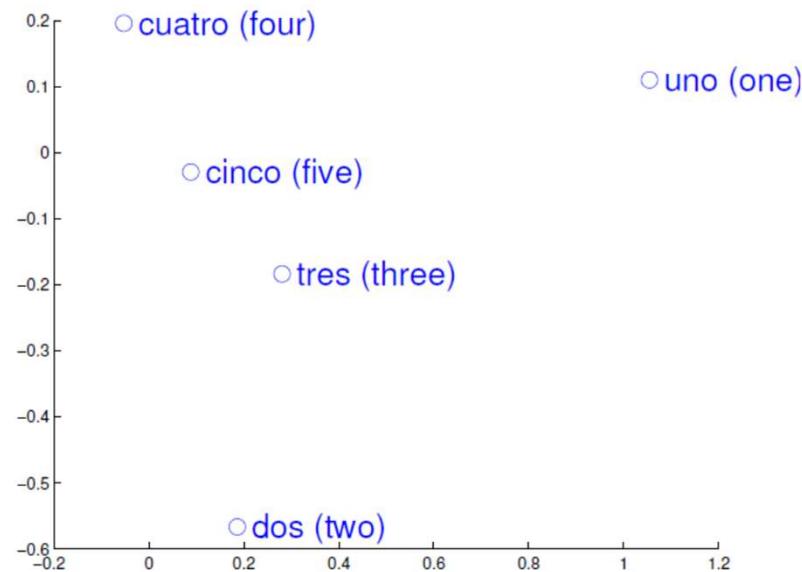
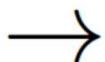
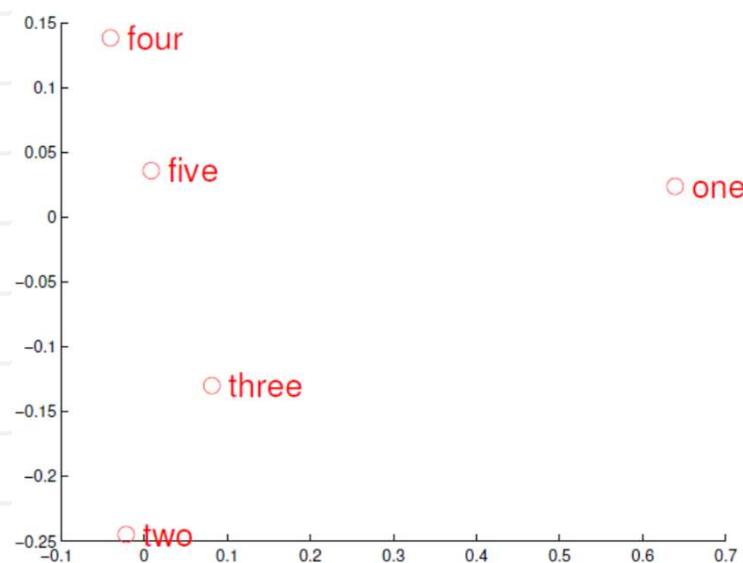
[Kim EMNLP14]

# Word2vec reuse

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	<b>89.6</b>
CNN-non-static	<b>81.5</b>	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	<b>88.1</b>	93.2	92.2	<b>85.0</b>	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	—	—	—	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	—	—	—	—
RNTN (Socher et al., 2013)	—	45.7	85.4	—	—	—	—
DCNN (Kalchbrenner et al., 2014)	—	48.5	86.8	—	93.0	—	—
Paragraph-Vec (Le and Mikolov, 2014)	—	<b>48.7</b>	87.8	—	—	—	—
CCAE (Hermann and Blunsom, 2013)	77.8	—	—	—	—	—	87.2
Sent-Parse (Dong et al., 2014)	79.5	—	—	—	—	—	86.3
NBSVM (Wang and Manning, 2012)	79.4	—	—	93.2	—	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	—	—	<b>93.6</b>	—	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	—	—	93.4	—	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	—	—	<b>93.6</b>	—	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	—	—	—	—	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	—	—	—	—	—	82.7	—
SVM <sub>S</sub> (Silva et al., 2011)	—	—	—	—	<b>95.0</b>	—	—

[Kim EMNLP14]

# Going cross-lingual

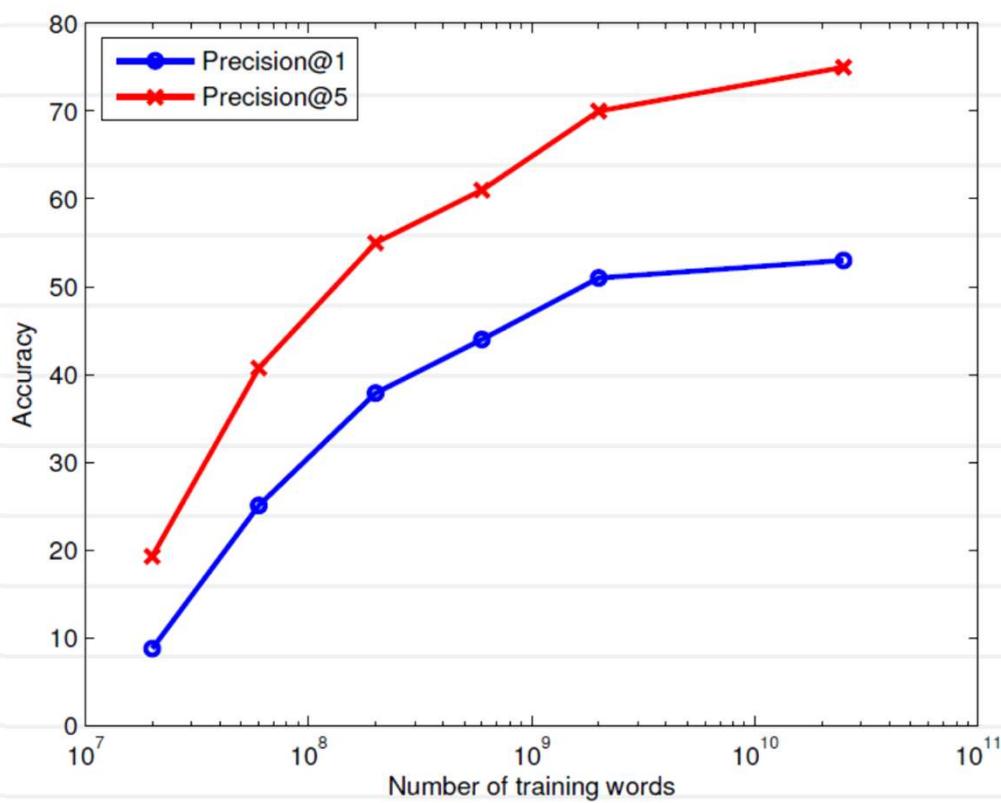


[Mikolov et al. 2013]

# Going cross-lingual

$$\min_W \sum_{i=1}^n \|Wx_i - z_i\|^2$$

Train on 5K most frequent



Influence of word2vec training set

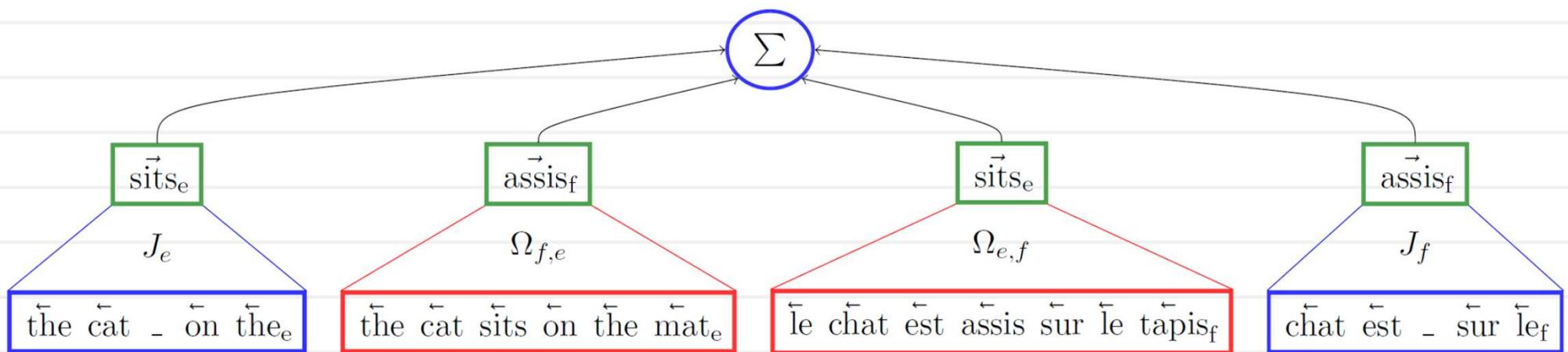
Spanish word	Computed English Translations	Dictionary Entry
emociones	emotions emotion feelings	emotions
protegida	wetland undevlopable protected	protected
imperio	dictatorship imperialism tyranny	empire
determinante	crucial key important	determinant
preparada	prepared ready prepare	prepared
millas	kilometers kilometres miles	miles
hablamos	talking talked talk	talk
destacaron	highlighted emphasized emphasised	highlighted

[Mikolov et al. 2013]

# Trans-gram model

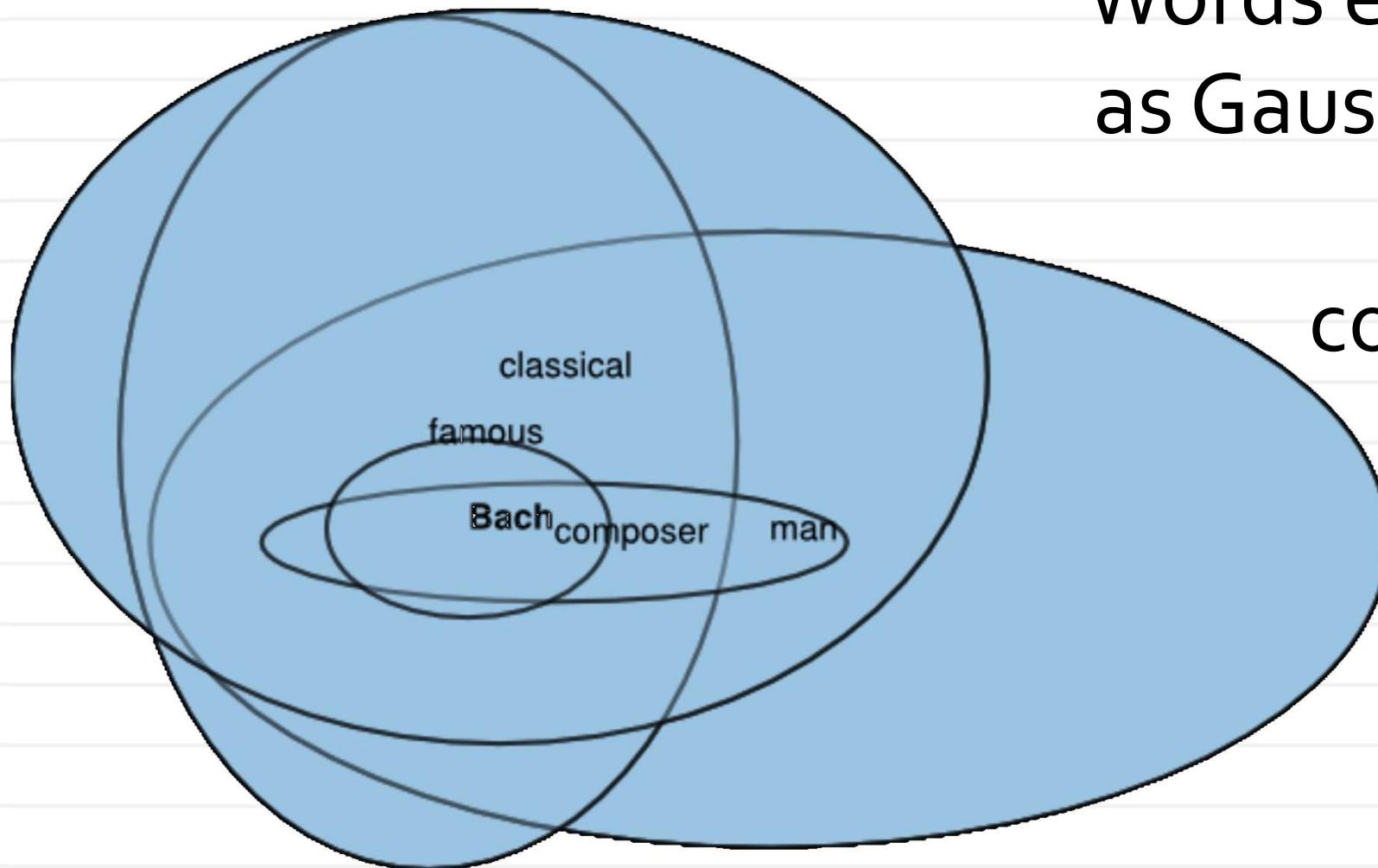
Trains from:

- Mono-lingual French corpus
- Mono-lingual English corpus
- Sentence-level aligned parallel corpus



[Coulmance et al. EMNLP15]

# Gaussian embeddings



Words embedded  
as Gaussians with  
diagonal  
covariances

[Vilnis & McCallum ICLR15]

# Mixture embeddings: sense disambiguation



Word	Co.	Nearest Neighbors
rock	0	basalt:1, boulder:1, boulders:0, stalagmites:0, stalactites:0, rocks:1, sand:0, quartzite:1, bedrock:0
rock	1	rock:/1, ska:0, funk:1, pop-rock:1, punk:1, indie-rock:0, band:0, indie:0, pop:1
bank	0	banks:1, mouth:1, river:1, River:0, confluence:0, waterway:1, downstream:1, upstream:0, dammed:0
bank	1	banks:0, banking:1, banker:0, Banks:1, bankas:1, Citibank:1, Interbank:1, Bankers:0, transactions:1
Apple	0	Strawberry:0, Tomato:1, Raspberry:1, Blackberry:1, Apples:0, Pineapple:1, Grape:1, Lemon:0
Apple	1	Macintosh:1, Mac:1, OS:1, Amiga:0, Compaq:0, Atari:1, PC:1, Windows:0, iMac:0
star	0	stars:0, Quaid:0, starlet:0, Dafoe:0, Stallone:0, Geena:0, Niro:0, Zeta-Jones:1, superstar:0
star	1	stars:1, brightest:0, Milky:0, constellation:1, stellar:0, nebula:1, galactic:1, supernova:1, Ophiuchus:1

[Athiwaratkun & Wilson ACL17]

# Recap

- NLP tasks are amenable to deep learning methods
- Char-level NLP is promising
- *Peculiarity 1:* mapping words to vectors (embedding layers)
- *Peculiarity 2:* variable sentence length (more next time)
- Lots of success with predictive learning
- Today's lecture: primer (word-level predictive learning), next lectures going to sentences and beyond

# Bibliography

Douglas L. T. Rohde, Laura M. Gonnerman, and David C. Plaut. 2006. An improved model of semantic similarity based on lexical co-occurrence. Communications of the ACM, 49(8):627–633.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, Christian Janvin: A Neural Probabilistic Language Model. Journal of Machine Learning Research 3: 1137-1155 (2003)

Karlin, Nurit. The Fat Cat Sat on the Mat. New York: HarperCollins, 1996

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, Jeffrey Dean:  
Distributed Representations of Words and Phrases and their Compositionality. NIPS 2013: 3111-3119

Carl Doersch, Abhinav Gupta, Alexei A. Efros:  
Unsupervised Visual Representation Learning by Context Prediction. ICCV 2015: 1422-1430

# Bibliography

Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean:  
Efficient Estimation of Word Representations in Vector Space. CoRR  
abs/1301.3781 (2013)

Mnih, A. and Hinton, G.-E. A Scalable Hierarchical Distributed Language Model. NIPS 2009

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, Pavel P. Kuksa:  
Natural Language Processing (Almost) from Scratch. Journal of Machine Learning Research 12: 2493-2537 (2011)

Nal Kalchbrenner, Edward Grefenstette, Phil Blunsom:  
A Convolutional Neural Network for Modelling Sentences. ACL (1) 2014: 655-665

Jocelyn Coulmance, Jean-Marc Marty, Guillaume Wenzek, Amine Benhalloum: Trans-gram, Fast Cross-lingual Word-embeddings. EMNLP 2015: 1109-1113

# Bibliography

---

Vilnis, L., & McCallum, A. Word Representations via Gaussian Embedding.  
ICLR 2015

Athiwaratkun, B., & Wilson, A. G. Multimodal Word Distributions. In  
Proceedings of the 55th Annual Meeting of the Association for Computational  
Linguistics (ACL 2017)