Language Representation and Modeling

Kapil Thadani
kapil@cs.columbia.edu

YAHOO!
RESEARCH

Schedule

- Language representation and modeling ..... March 1
  - Sparse and distributional representations
  - word2vec and doc2vec
  - Task-specific representations
  - Language modeling

- Encoder-decoder frameworks ..... March 8
  - RNN units: LSTMs, GRUs
  - Sequence-to-sequence models
  - Attention mechanism
  - Copying mechanism
  - Scheduled sampling

- Applications ..... March 22
  - Parsing
  - Question-answering
  - Entailment
  - Machine reading
  - Dialog systems

Outline

- Language representation
  - Bag of words
  - Distributional hypothesis
  - word2vec
  - doc2vec
- Task-specific representations
  - Convolutional NNs
  - Recurrent NNs
  - Recursive NNs
- Language modeling
  - Probabilistic and discriminative LMs
  - NNLM
  - RNNLMs

Formal language

(i) Set of sequences over symbols from an alphabet

sentences	words	vocabulary
utterances	morphemes	
documents	MWEs	
⋮	⋮	

(ii) Rules for valid sequences

spelling	orthography, morphology
grammar	syntax
meaning	semantics, discourse, pragmatics, $\cdots$

Natural language

(i) Set of sequences over symbols from an alphabet

sentences	words	vocabulary
utterances	morphemes	
documents	MWEs	
⋮	⋮	

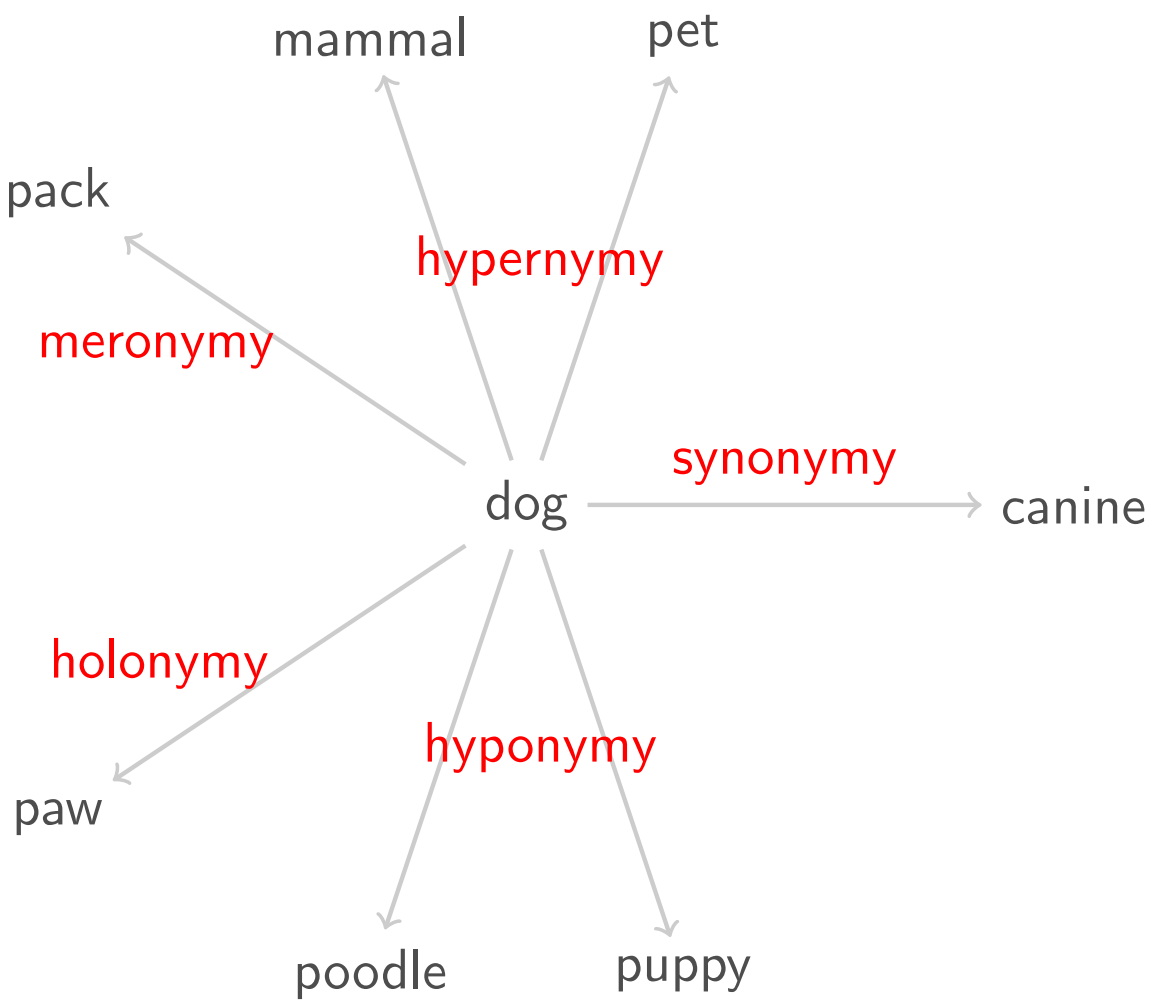
(ii) Rules for valid sequences

spelling	orthography, morphology
grammar	syntax
meaning	semantics, discourse, pragmatics, $\cdots$

Lexical semantics

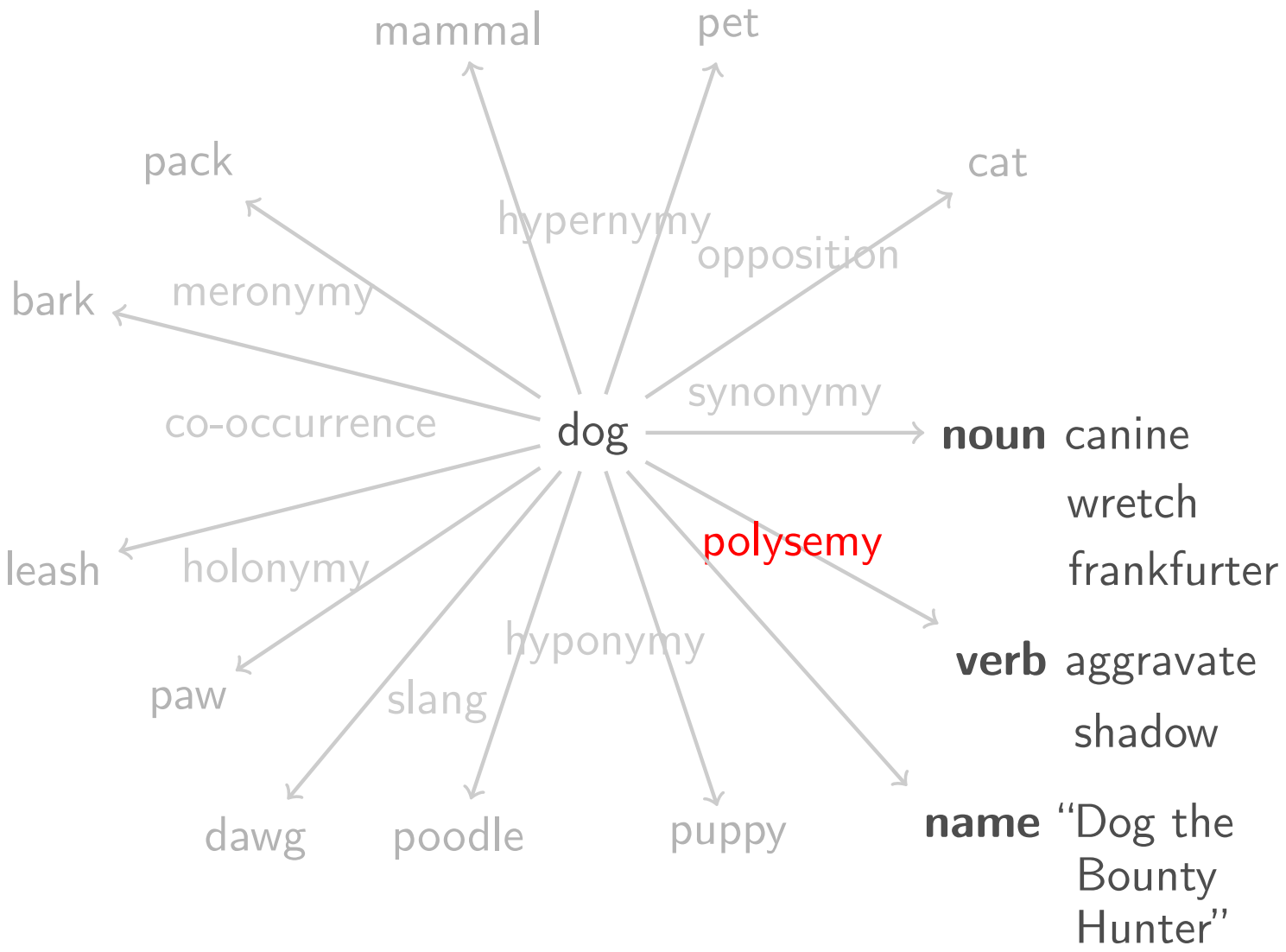
dog

Lexical semantics



Lexical semantics

Lexical semantics



Sparse representations

Word: one-hot (1-of-V) vectors

Document: "bag of words"

Emphasize rare words with inverse document frequency (IDF)

Compare documents with cosine similarity

$$\begin{array}{c} \uparrow \\ |V| \\ \downarrow \end{array} \begin{array}{c} \boxed{\begin{matrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{matrix}} \end{array} \begin{array}{l} \text{a} \\ \\ \text{dog} \\ \\ \text{zygote} \end{array}$$

+ Simple and interpretable
– No notion of word order
– No implicit semantics
– Curse of dimensionality with large $|V|$

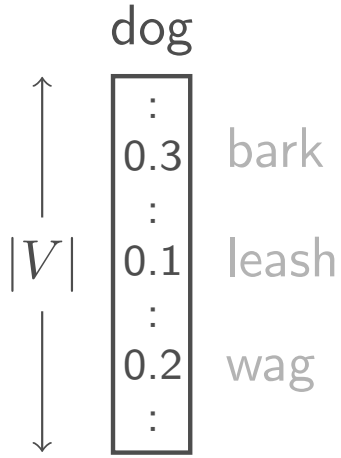
Distributional approaches

Words that occur in similar contexts have similar meanings

e.g., record word co-occurrence within a context window over a large corpus

Weight association with pointwise mutual information (PMI), etc

$$PMI(w_1, w_2) = \log_2 \frac{p(w_1, w_2)}{p(w_1)p(w_2)}$$



Latent Semantic Analysis

Deerwester et al. (1990)

Construct term-document matrix

$$M = \begin{array}{c} \longleftarrow |D| \longrightarrow \\ \begin{bmatrix} w_1^{(1)} & w_1^{(2)} & \cdots \\ w_2^{(1)} & \ddots & \\ \vdots & & \end{bmatrix} \end{array} \Big\updownarrow |V|$$

Singular value decomposition

$$M \approx \underbrace{\begin{bmatrix} u_1 & u_2 & u_3 & \cdots \end{bmatrix}}_{k} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \lambda_3 & \\ & & & \ddots \end{bmatrix} \begin{array}{c} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \end{bmatrix} \end{array} k$$

Select top k singular vectors for k-dim embeddings of words/docs

word2vec

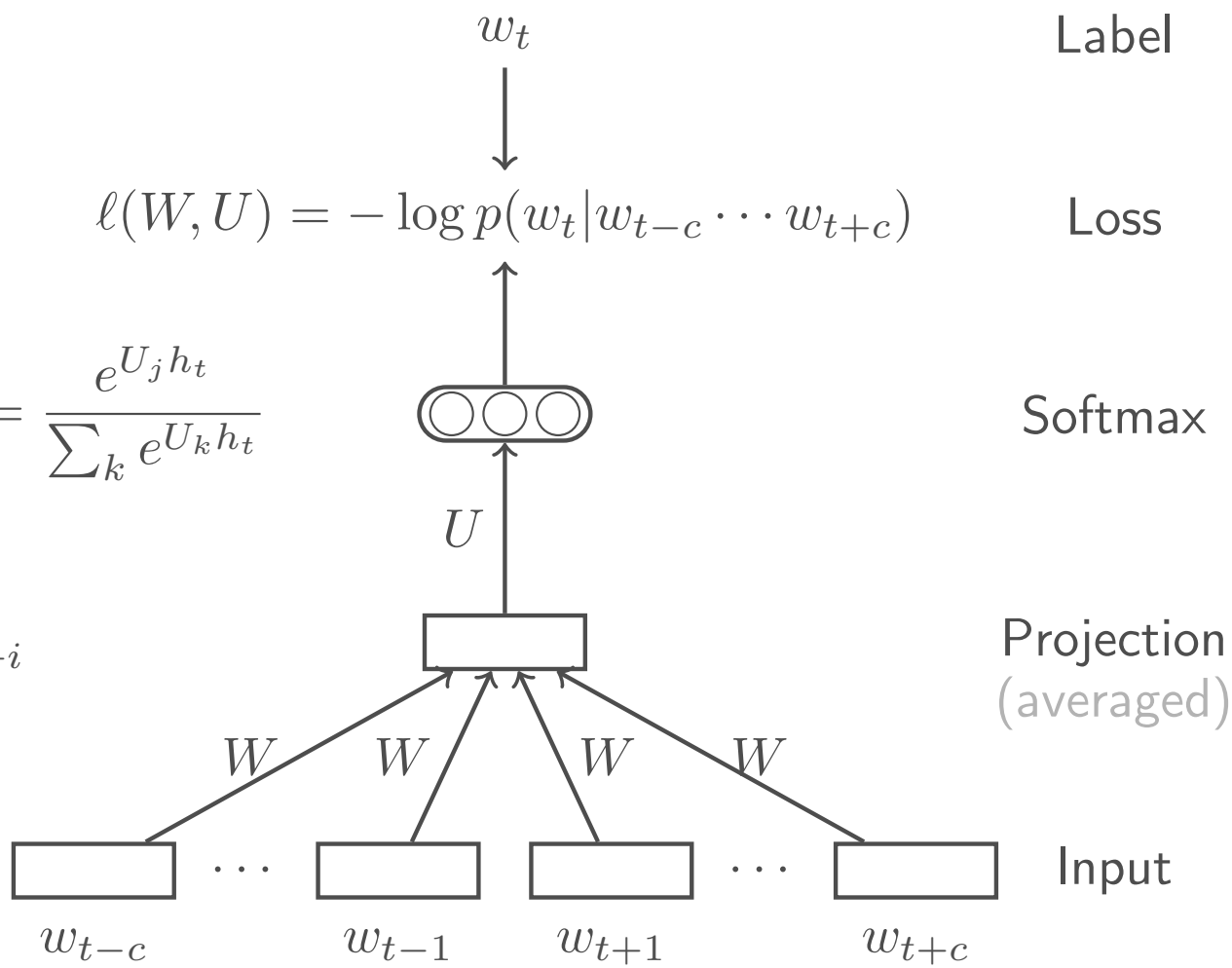
Mikolov et al. (2013)

Continuous Bag-of-Words (CBOW)

- Predict target w_t given context $w_{t-c}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+c}$

$$p(w_j | w_{t-c} \cdots w_{t+c}) = \frac{e^{U_j h_t}}{\sum_k e^{U_k h_t}}$$

$$h_t = \frac{1}{2c} \sum_{\substack{i=-c \\ i \neq 0}}^{c} W^\top w_{t+i}$$

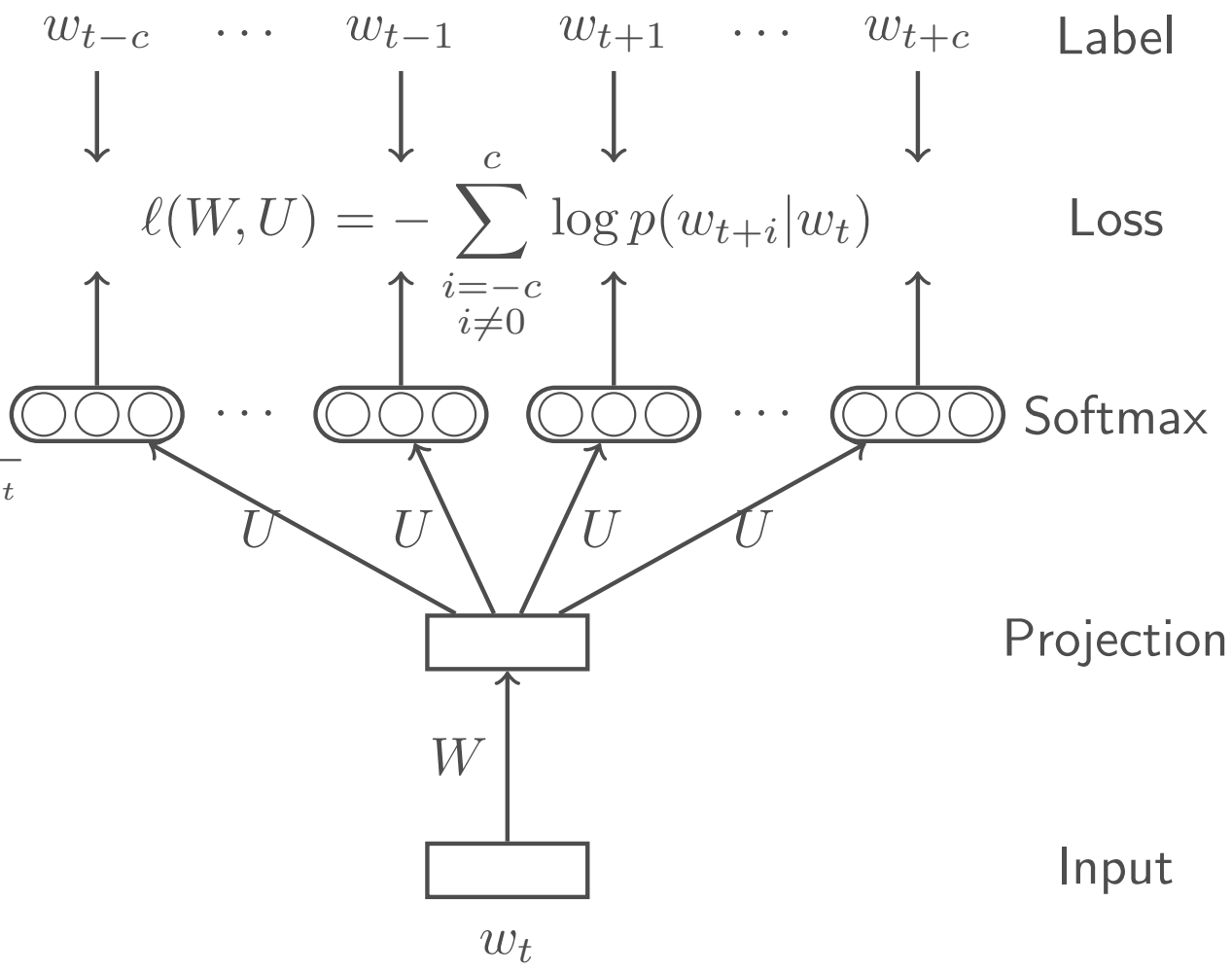


word2vec

Mikolov et al. (2013)

Skip-gram

- Predict context $w_{t-c}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+c}$ given target w_t



$$p(w_j|w_t) = \frac{e^{U_j h_t}}{\sum_k e^{U_k h_t}}$$

$$h_t = W^\top w_t$$

word2vec

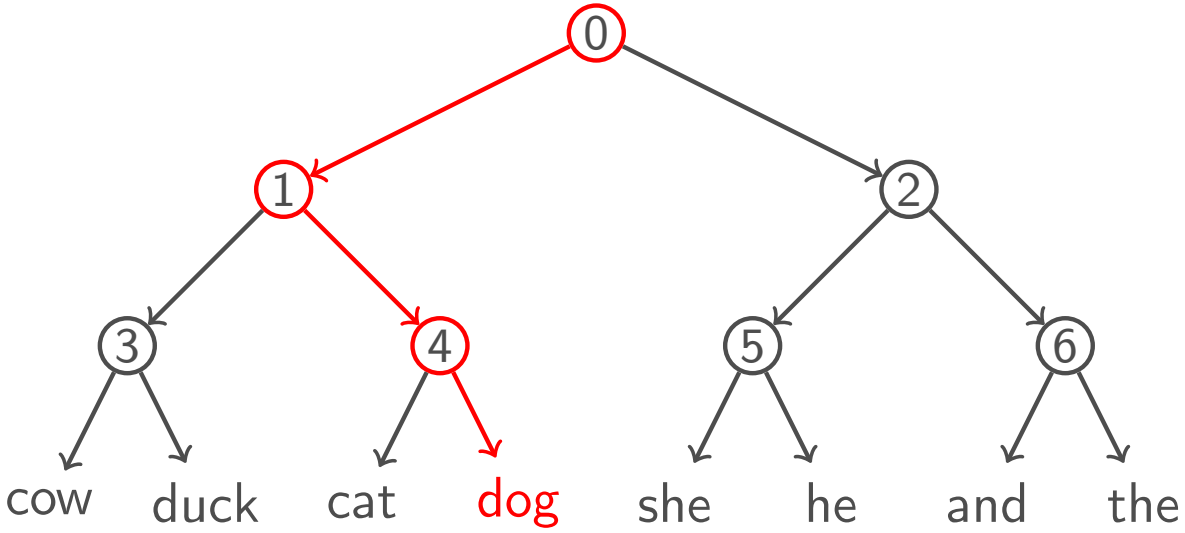
Mikolov et al. (2013)

Cost of computing $\nabla p(w_j | \cdots)$ is proportional to V!

Alternative 1: Hierarchical softmax

- Predict path in binary tree representation of output layer
- Reduces to $\log_2(V)$ binary decisions

$$p(w_t = \text{“dog”} | \cdots) = (1 - \sigma(U_0 h_t)) \times \sigma(U_1 h_t) \times \sigma(U_4 h_t)$$



word2vec

Mikolov et al. (2013)

Cost of computing $\nabla p(w_j | \cdots)$ is proportional to V!

Alternative 2: Negative sampling

- Change objective to differentiate target vector from noisy samples with logistic regression

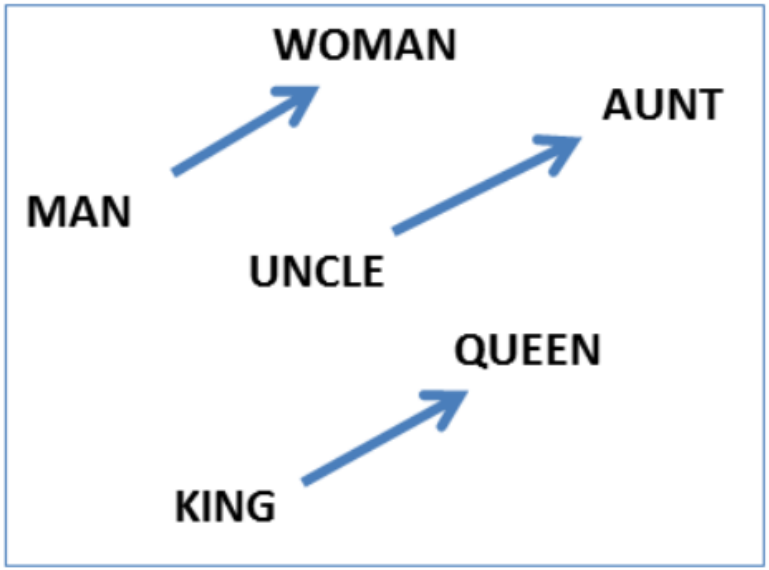
$$\max \ \log \sigma(u_j^\top h_t) + \sum_{k=1}^{K} \mathbb{E}_{w_m \sim \Psi} \log \sigma(-u_m{}^\top h_t)$$

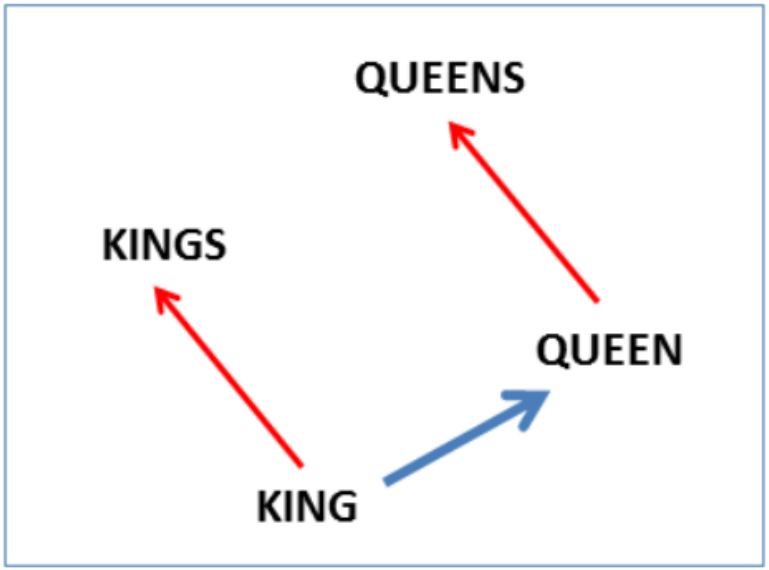
  where $u_j = U_j = j$'th column of U

  and $w_j \in \text{context}(w_t)$
- Noise distribution Ψ typically unigram, uniform or in between
- Number of samples K typically 5–20

word2vec

Mikolov et al. (2013)

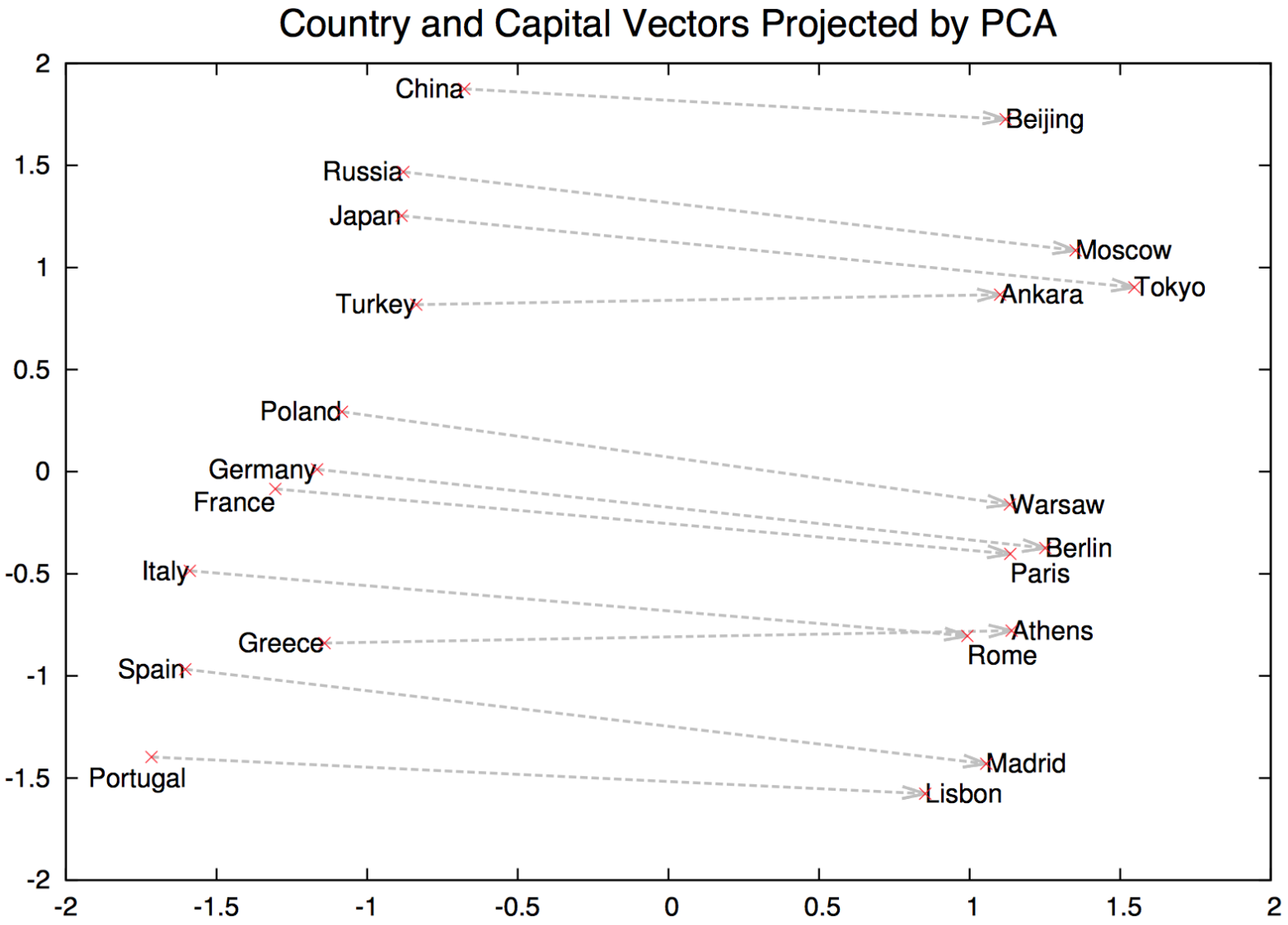




Linear relationships between related words

word2vec

Mikolov et al. (2013)



Visualizing lexical relationships

word2vec

Mikolov et al. (2013)

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	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

Analogical reasoning

word2vec

Mikolov et al. (2013)

Newspapers			
New York	New York Times	Baltimore	Baltimore Sun
San Jose	San Jose Mercury News	Cincinnati	Cincinnati Enquirer
NHL Teams			
Boston	Boston Bruins	Montreal	Montreal Canadiens
Phoenix	Phoenix Coyotes	Nashville	Nashville Predators
NBA Teams			
Detroit	Detroit Pistons	Toronto	Toronto Raptors
Oakland	Golden State Warriors	Memphis	Memphis Grizzlies
Airlines			
Austria	Austrian Airlines	Spain	Spainair
Belgium	Brussels Airlines	Greece	Aegean Airlines
Company executives			
Steve Ballmer	Microsoft	Larry Page	Google
Samuel J. Palmisano	IBM	Werner Vogels	Amazon

Phrase analogies

word2vec

Mikolov et al. (2013)

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

Additive compositionality

word2vec

Levy & Goldberg (2014)

Skip-gram with negative sampling increases $u_j^\top h_t$ for real word-context pairs $\langle w_t, w_j \rangle$ and decreases it for noise pairs

Given:

- a matrix of d-dim word vectors W ($|V_w| \times d$)
- a matrix of d-dim context vectors U ($|V_u| \times d$)

Skip-gram is implicitly factorizing the matrix $M = WU^\top$

What is M?

- Word-context matrix where each cell (i, j) contains $PMI(w_i, w_j)$
- If number of negative samples $K > 1$, this is shifted by a constant $-\log K$
- (Assuming large enough d and iterations)

Beyond words

Can we add word vectors to make sentence/paragraph/doc vectors?

doc $A = a_1 + a_2 + a_3$

doc $B = b_1 + b_2 + b_3$

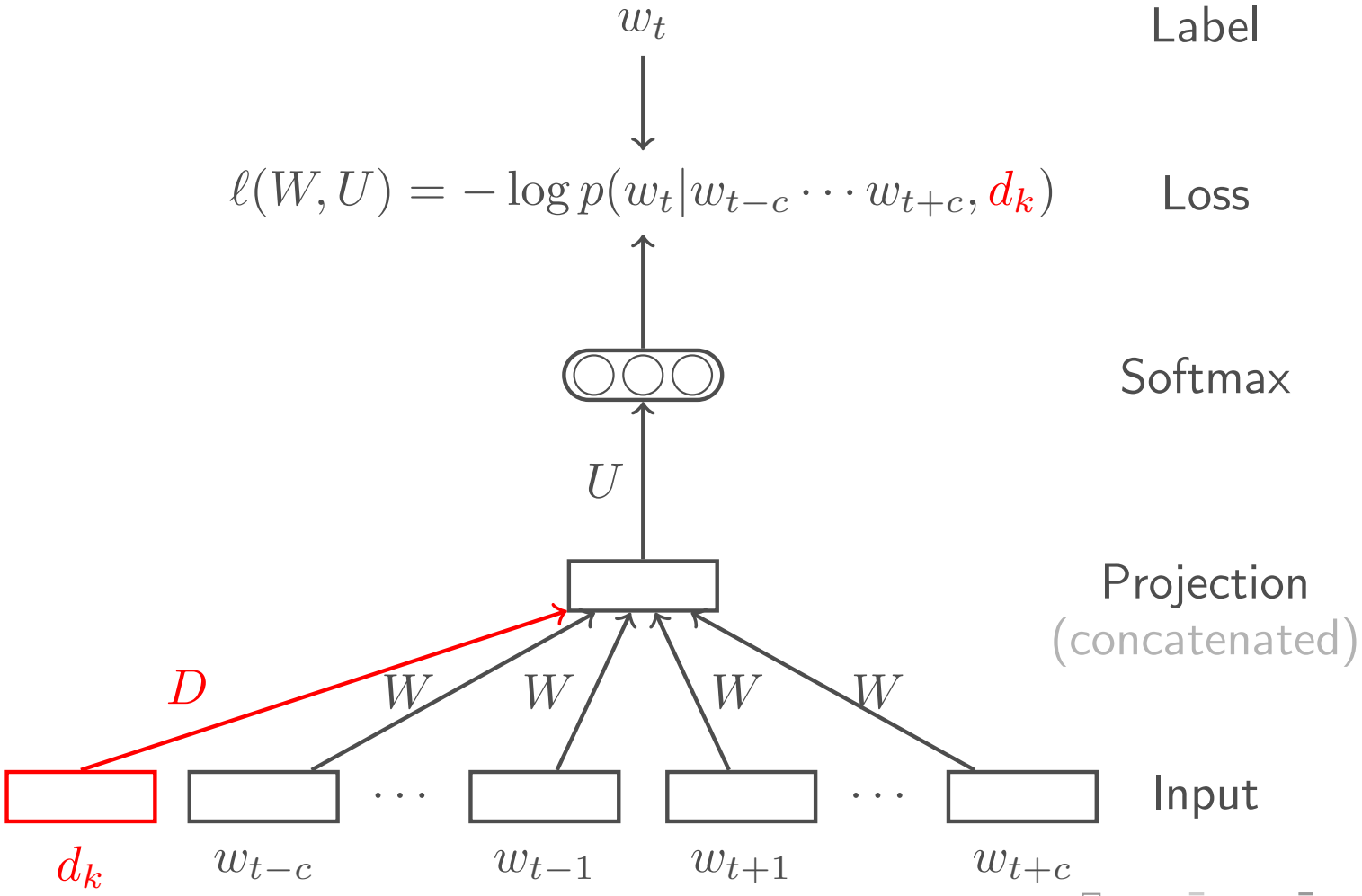
$$
\begin{aligned}
cos(A, B) &= \frac{A \cdot B}{||A|| \cdot ||B||} \\
&= \frac{1}{||A|| \cdot ||B||}(a_1 \cdot b_1 + a_1 \cdot b_2 + a_1 \cdot b_3 + \\
&\qquad a_2 \cdot b_1 + a_2 \cdot b_2 + a_2 \cdot b_3 + \\
&\qquad a_3 \cdot b_1 + a_3 \cdot b_2 + a_3 \cdot b_3) \\
&= \text{weighted all-pairs similarity over } A \text{ and } B
\end{aligned}
$$

Paragraph vector (a.k.a doc2vec)

Le & Mikolov (2014)

Distributed memory

- Predict target w_t given context $w_{t-c}, \ldots, w_{t-1}, w_{t+1}, \ldots, w_{t+c}$ and doc label d_k
- At test time, hold U, W fixed and back-prop into expanded D

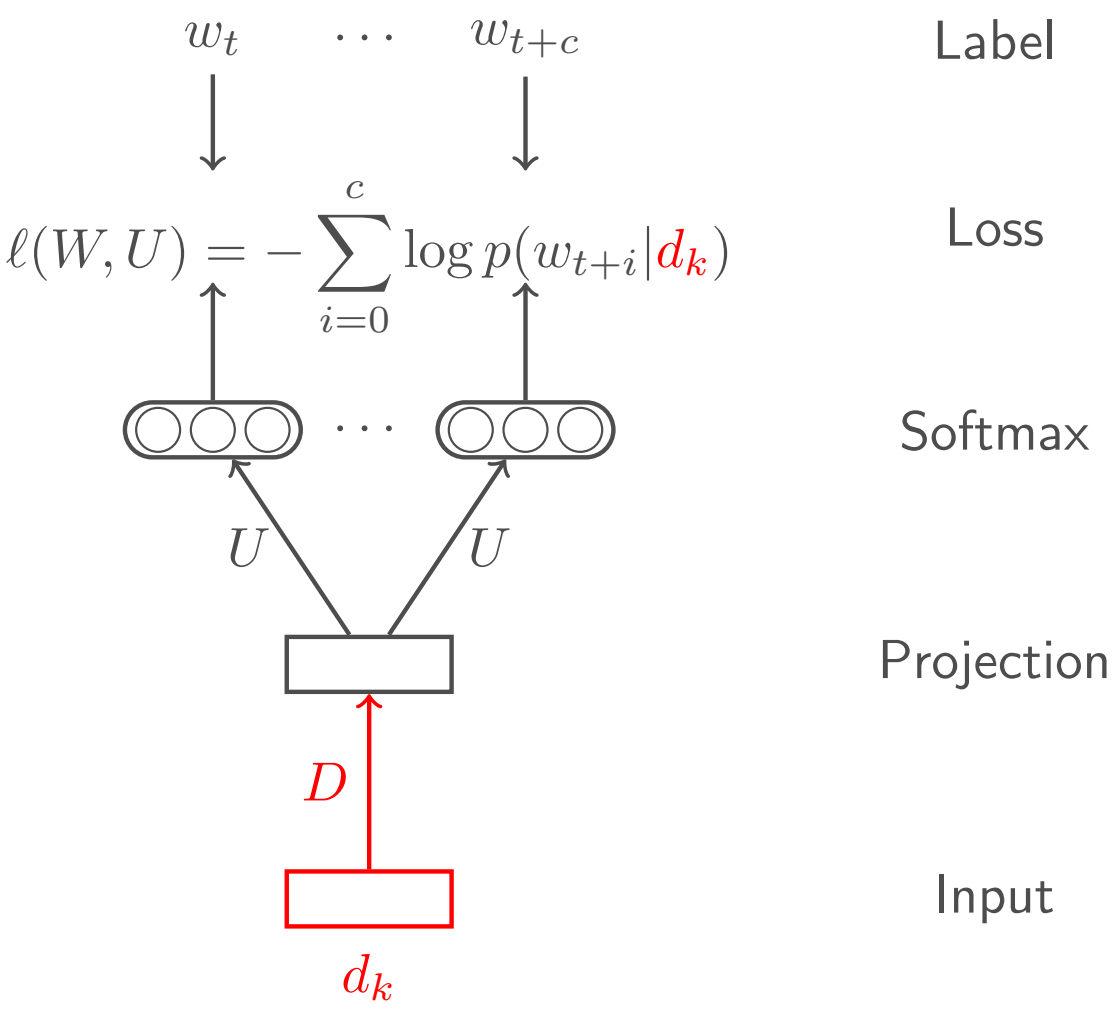


Paragraph vector (a.k.a doc2vec)

Le & Mikolov (2014)

Distributed Bag-of-Words (DBOW)

- Predict target n-grams $w_t, \ldots, w_{t+c}$ given doc label d_k
- At test time, hold U fixed and back-prop into expanded D



Semantics are elusive

Visitors saw her duck with binoculars .

Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

Visitors saw her duck with binoculars .

Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

Visitors	saw	her	duck	with	binoculars	.
NNS	VBD	PRP$	NN	IN	NNS	.

Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

Visitors	saw	her	duck	with	binoculars	.
NNS	VBD	PRP	VB	IN	NNS	.

Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

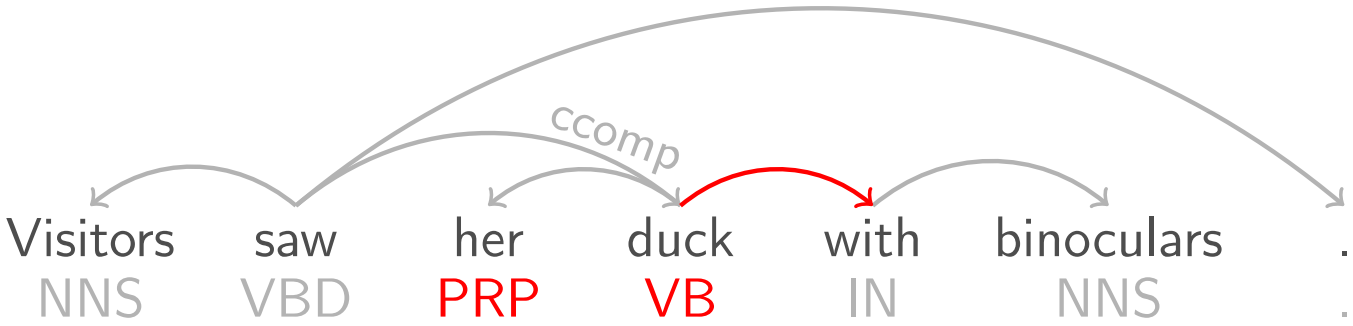
Visitors	saw	her	duck	with	binoculars	.
NNS	VBD	PRP	VB	IN	NNS	.

Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

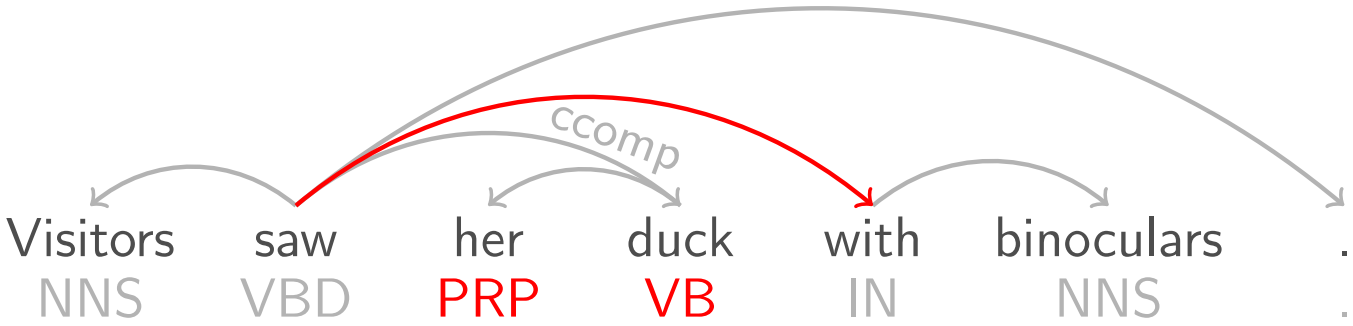


Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

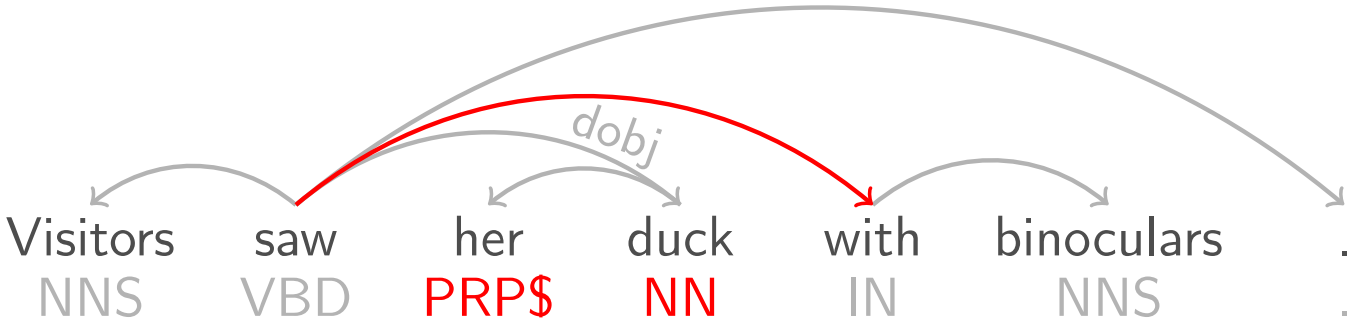


Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

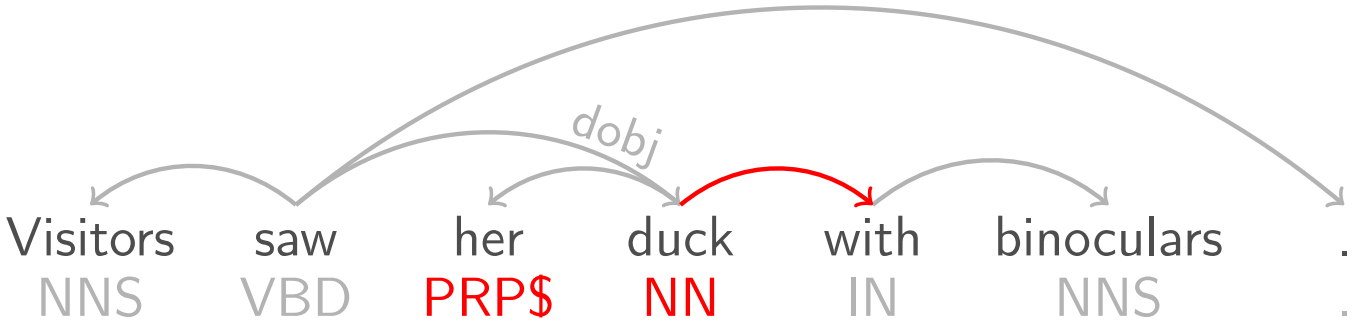


Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive

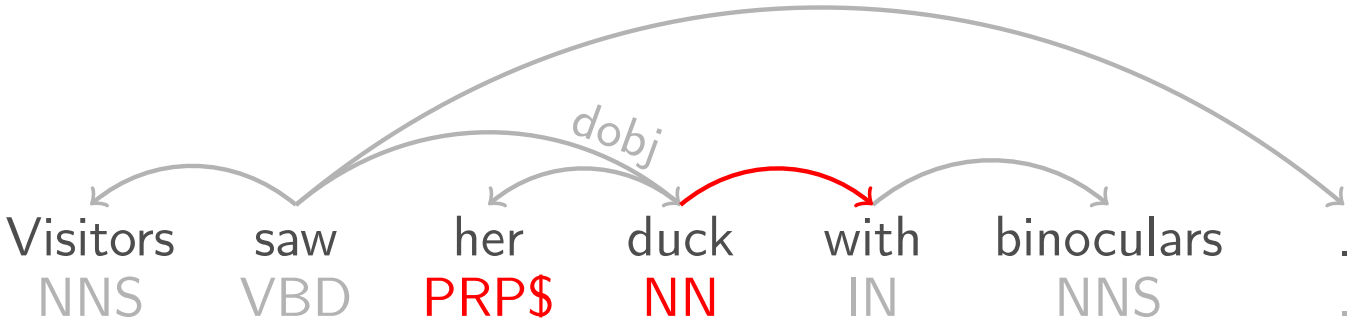


Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Semantics are elusive



Did she duck or does she *have* a duck?

Who has the binoculars?

How many pairs of binoculars are there?

Task-specific representations

Many NLP tasks fall into classification or sequence tagging

Classification

- Given variable-length text $w_1 \cdots w_n$ (sentence, document, etc), find label y
- Normal discriminative approach:
  - Extract features over the input text
  - Train a linear classifier

Tagging

- Given variable-length text $w_1 \cdots w_n$, label spans $z_1, \ldots, z_m$
- Normal discriminative approach:
  - Distribute labels over input words (e.g., BIO, BILOU encodings)
  - Extract features over input words
  - Train a linear-chain conditional random field

Convolutional NNs

Collobert et al (2011)

SENNA

- Part-of-speech tagging
- Chunking
- Named entity recognition
- Semantic role labeling

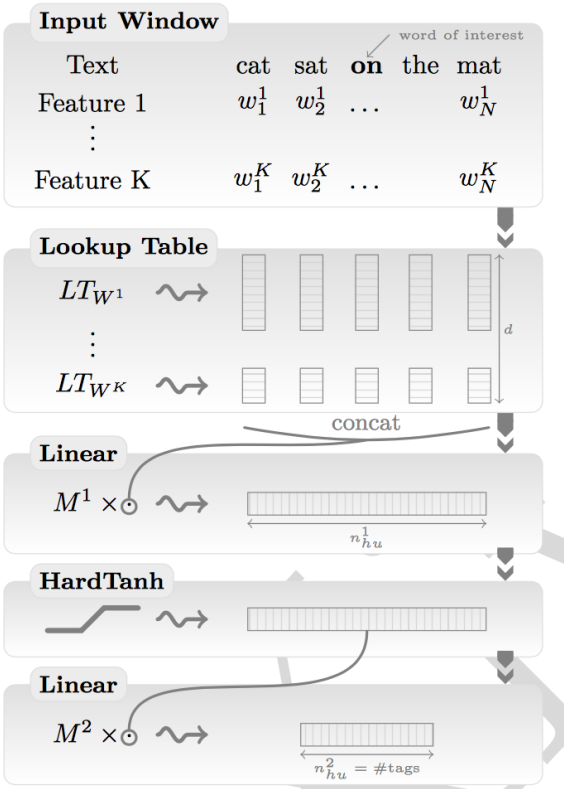


Figure 1: Window approach network.

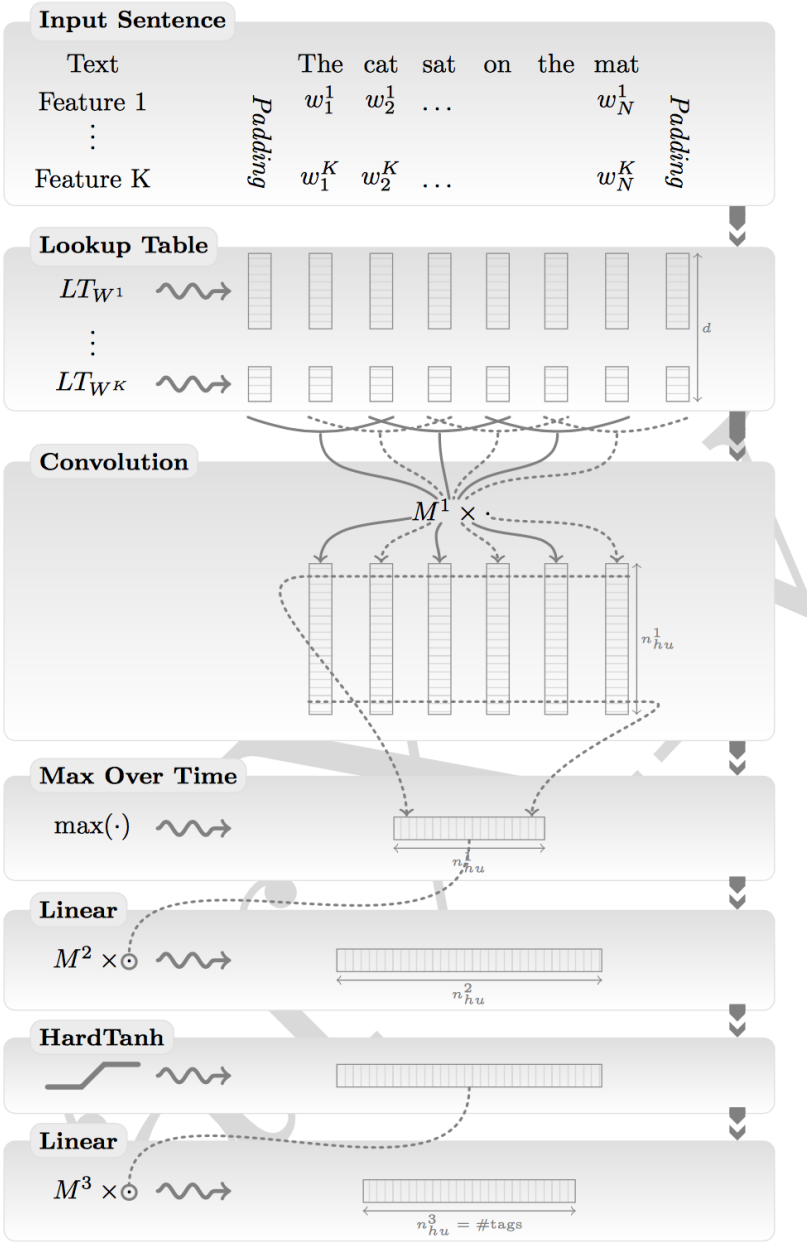
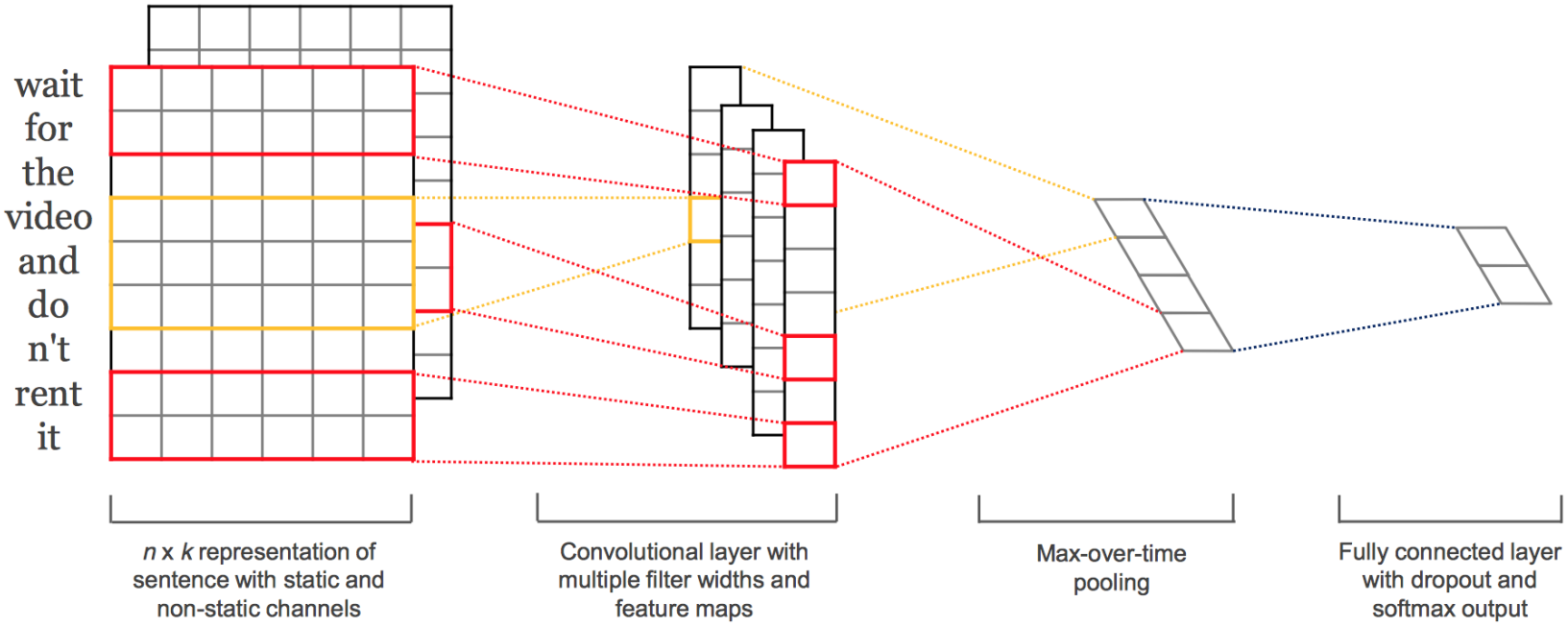


Figure 2: Sentence approach network.

Convolutional NNs

Kim (2014)



- Max-over-time pooling
- Two input embedding "channels" — one updated during training

Convolutional NNs

Lei et al (2015)

R-CNNs

- Capture interactions between non-consecutive n-grams
  e.g., not bad, not so bad, not nearly as bad

- Each n-gram context vector includes a weighted average over prior n-gram states

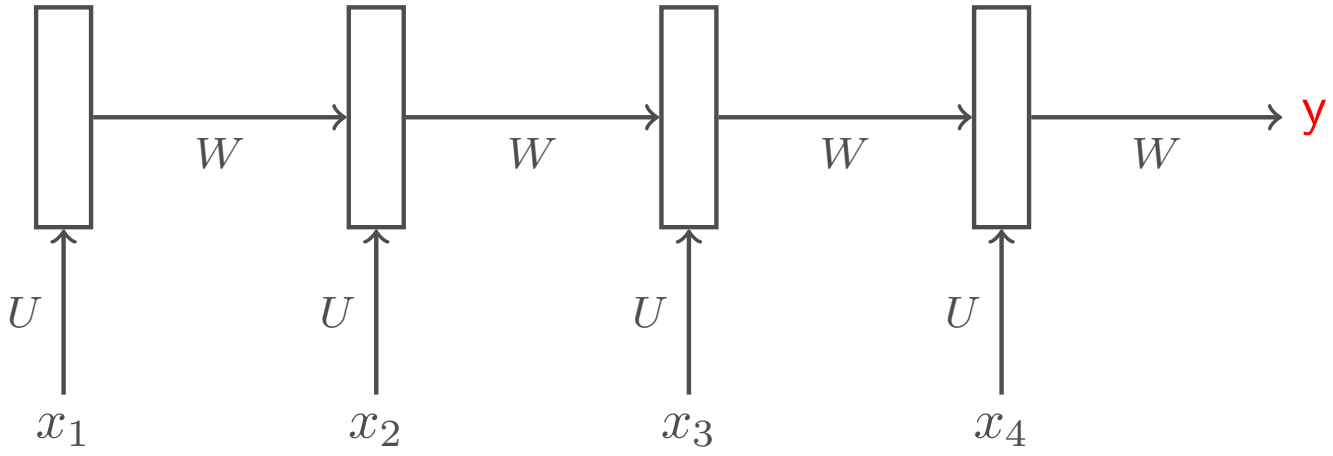
$$c_{ij} = U^\top (W_1 x_i \odot W_2 x_j)$$
$$c_t = \sum_{i<t} \lambda_t c_{it}$$
$$h_t = \tanh(c_t + b)$$

- Exponential computations avoided by dynamic programming

- Averaging weight learned as a recurrent gate

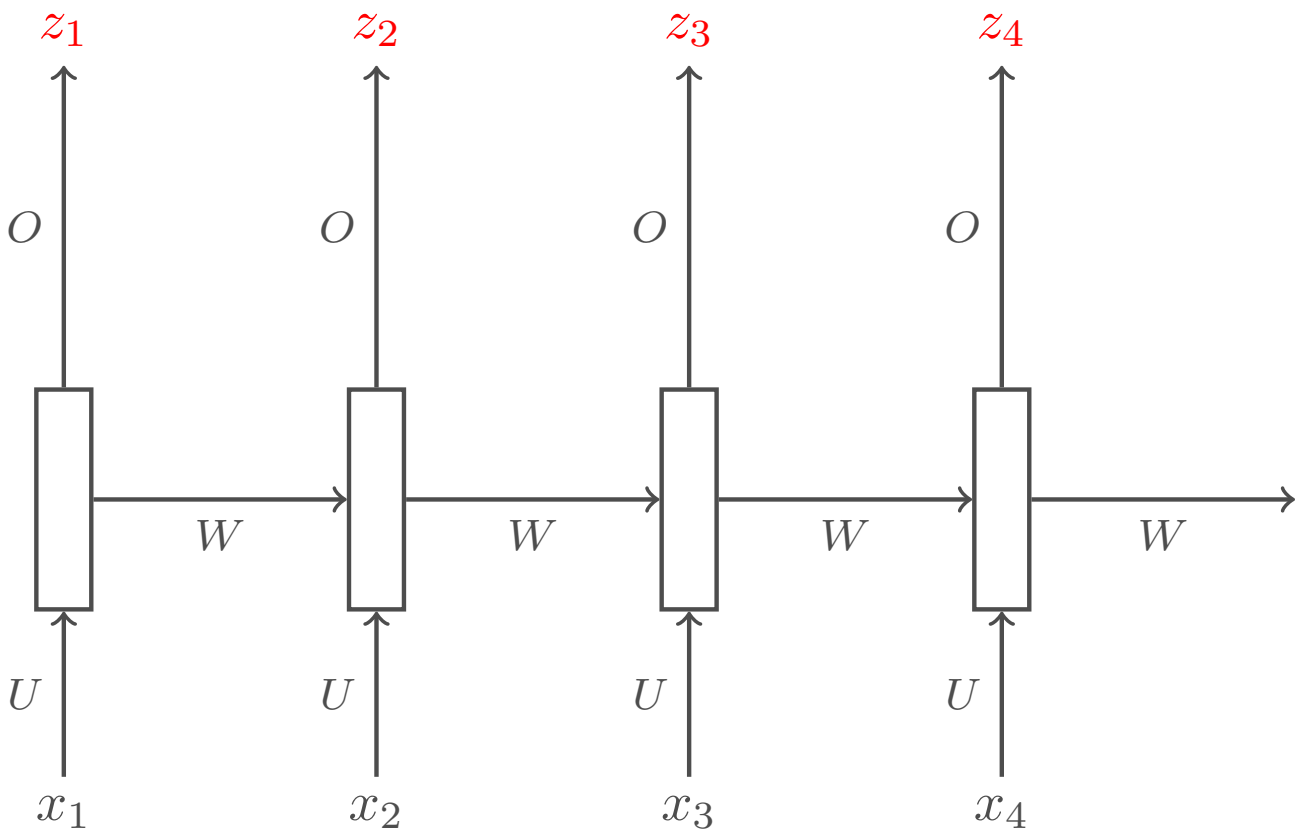
$$\lambda_t = \sigma(W_\lambda x_t + U_\lambda h_{t-1} + b)$$

Recurrent NNs



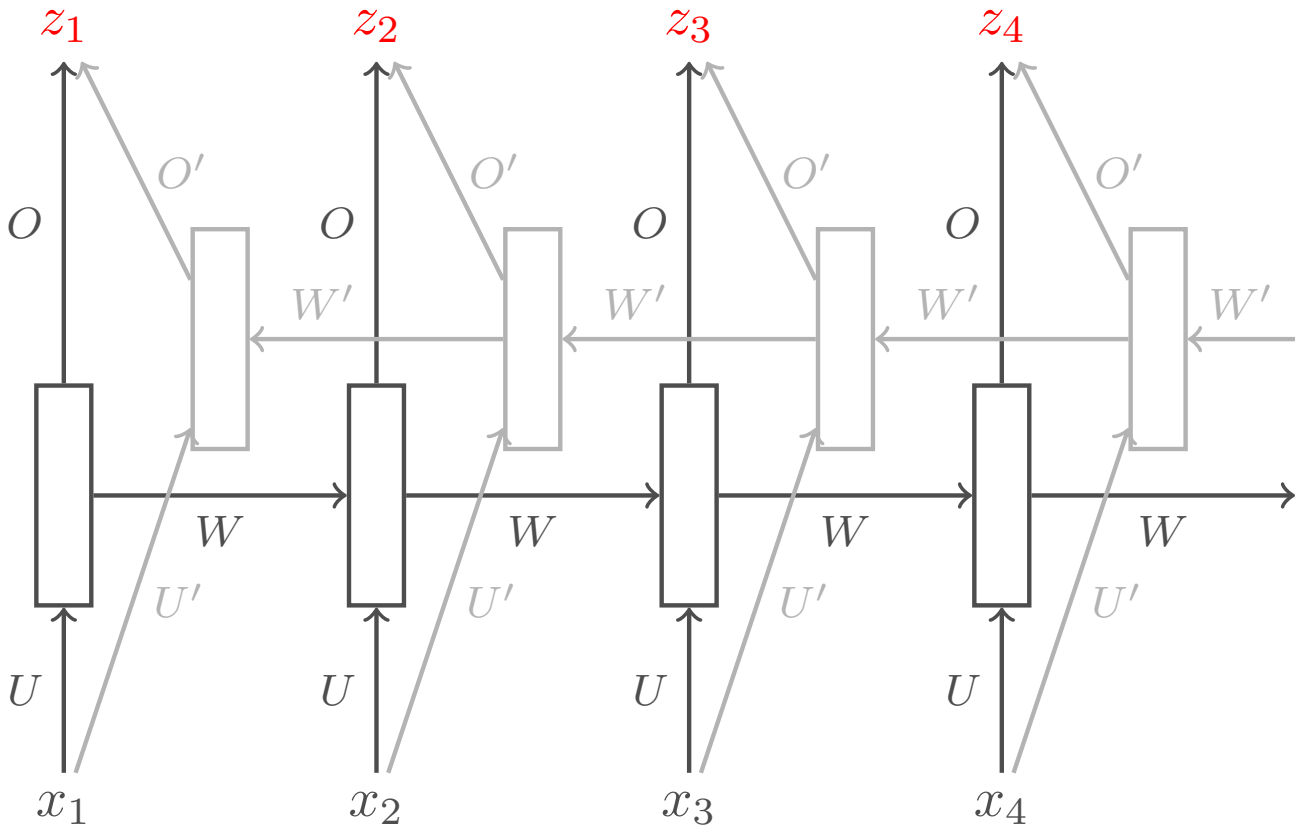
- Flexible models for classification and/or tagging
- Typically used with LSTM units or GRUs
- Bidirectional RNNs better for long inputs

Recurrent NNs



- Flexible models for classification and/or tagging
- Typically used with LSTM units or GRUs
- Bidirectional RNNs better for long inputs

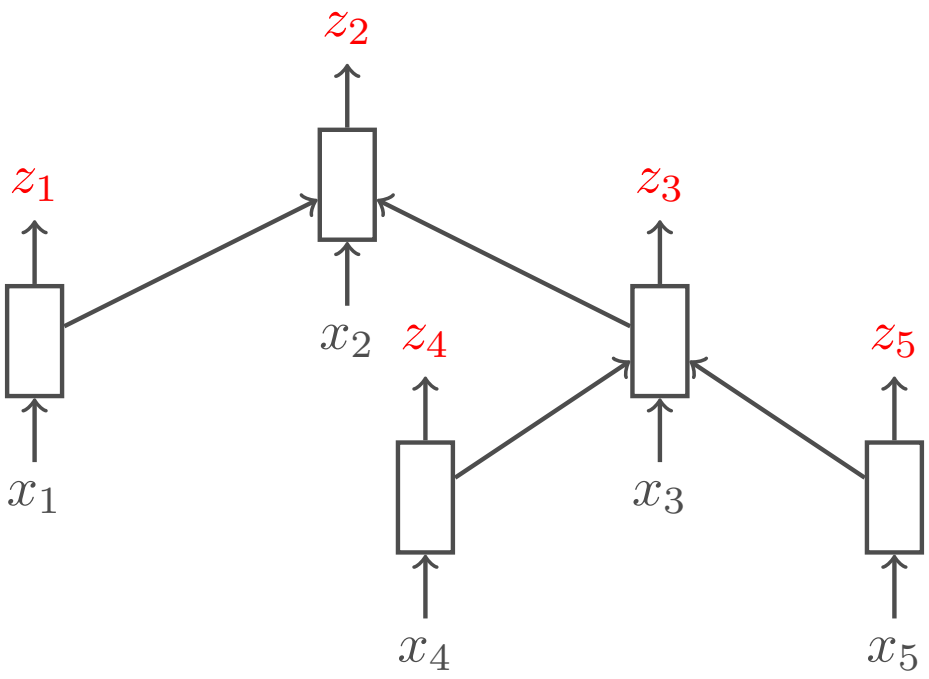
Recurrent NNs



- Flexible models for classification and/or tagging
- Typically used with LSTM units or GRUs
- Bidirectional RNNs better for long inputs

Recursive NNs (a.k.a Tree-RNNs)

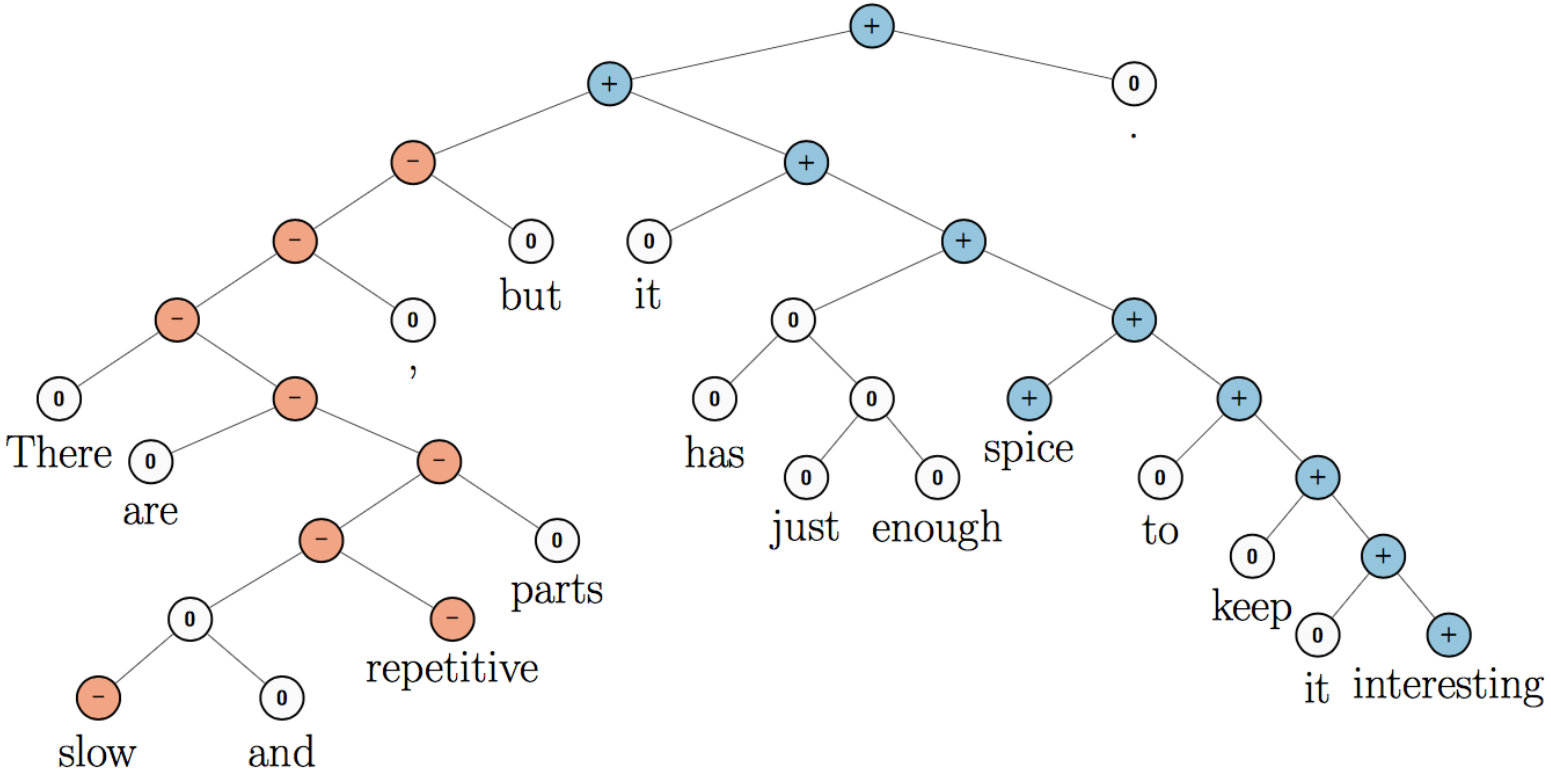
Tai et al (2015)



- Computation graph follows dependency or constituent parse
- Child-sum:
  - Good for arbitrary fan-out or unordered children
  - Suited to dependency trees (input x_i is head word)
- N-ary:
  - Fixed number of children, each parameterized separately
  - Suited to binarized constituency parses (leaves take word inputs x_i)

Recursive NNs (a.k.a Tree-RNNs)

Socher et al (2013)



- Computation graph follows dependency or constituent parse
- Child-sum:
  - Good for arbitrary fan-out or unordered children
  - Suited to dependency trees (input x_i is head word)
- N-ary:
  - Fixed number of children, each parameterized separately
  - Suited to binarized constituency parses (leaves take word inputs x_i)

Text generation

Probabilistic language modeling

- Distribution over sequences of words $p(w_1, \ldots, w_T)$ in a language
- Typically made tractable via conditional independence assumptions

$$p(w_1, \ldots, w_n) = \prod_{t=1}^{T} p(w_t | w_{t-1}, \ldots w_{t-n})$$

- n-gram counts estimated from large corpora
- Distributions *smoothed* to tolerate data sparsity, e.g., Laplace (add-one) smoothing, Kneser-Ney smoothing
- Evaluate on *perplexity* over held-out data

$$2^{\frac{1}{N} \sum_{i=1}^{N} \log_2 p\left(w_1^{(i)} \ldots w_{T_i}^{(i)}\right)}$$

Discriminative language modeling

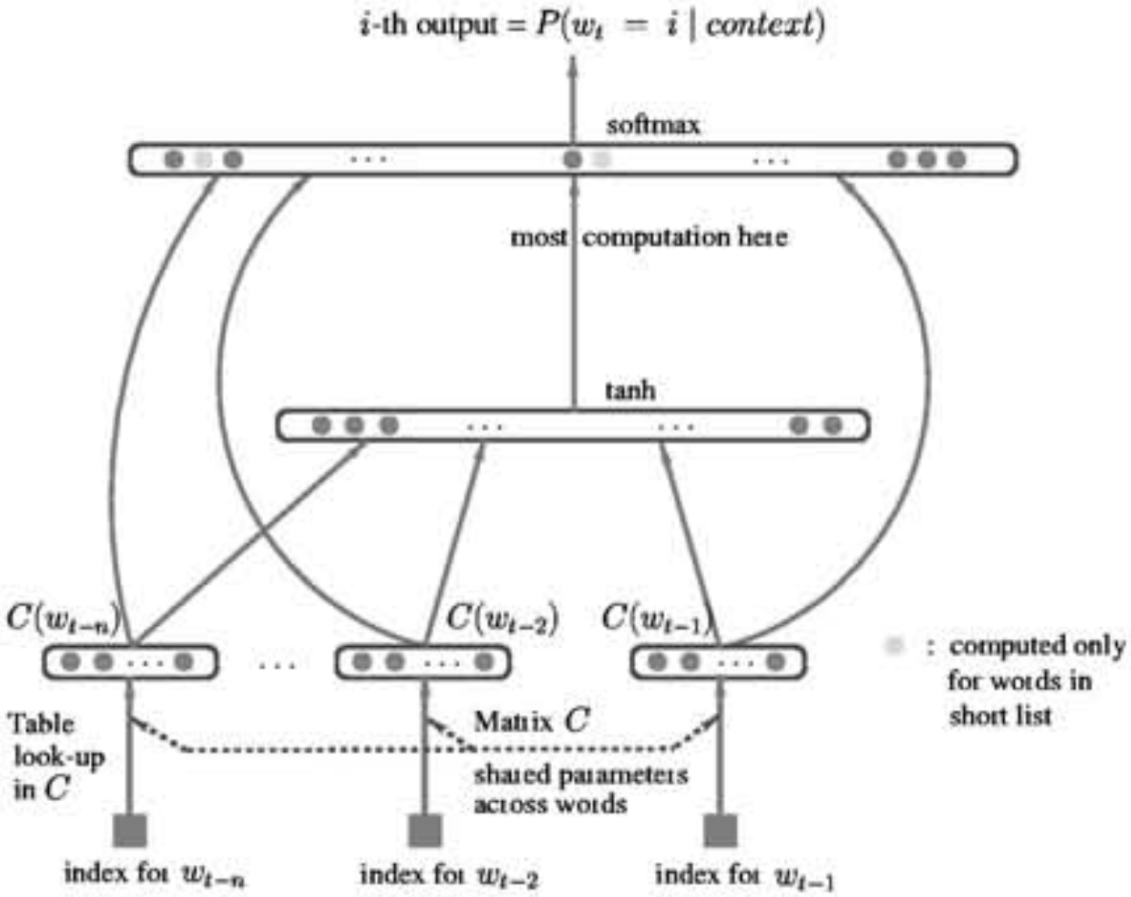
- Estimate n-gram probabilities with a discriminative model

$$p(w_t | w_{t-1}, \ldots w_1) \approx f(w_1, \ldots, w_t)$$

NNLM

Bengio et al (2003)

Model $p(w_t|w_{t-1}, \ldots w_{t-n})$ with a feed-forward neural net



RNNLM

Mikolov et al (2010)

Model $p(w_t|w_{t-1}, \ldots w_{t-n})$ with an RNN

Or an ensemble of multiple RNNs, randomly initialized

Model	Perplexity
Kneser-Ney 5-gram	141
Random forest [Xu 2005]	132
Structured LM [Filimonov 2009]	125
Feedforward NN LM	116
Syntactic NN LM [Emami 2004]	110
RNN trained by BP	113
RNN trained by BPTT	106
4x RNN trained by BPTT	98

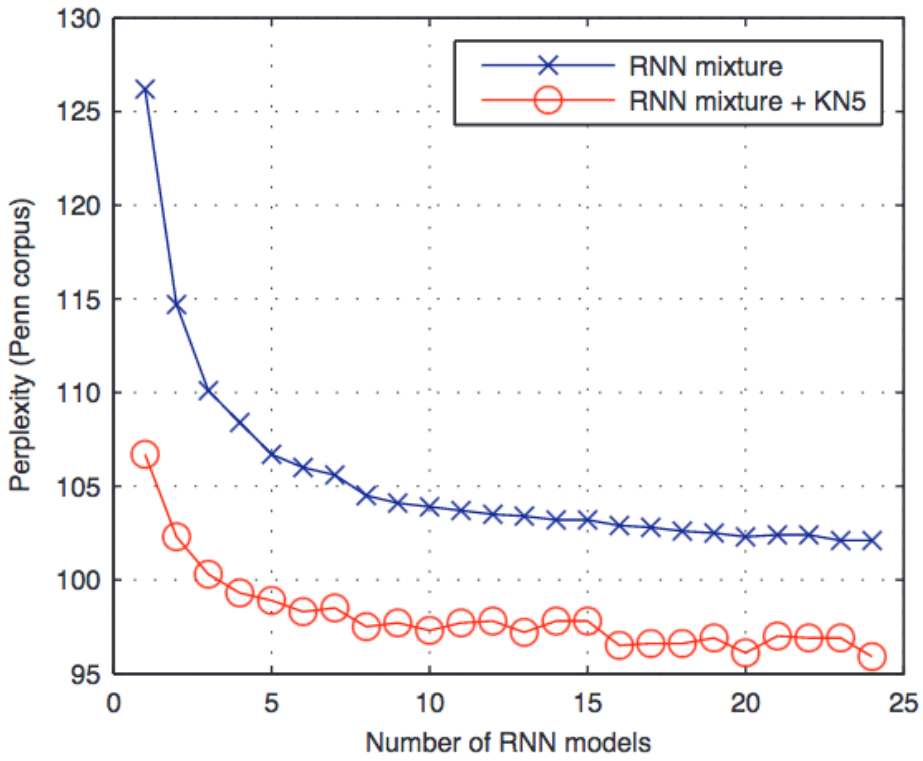
Results on Penn Treebank corpus

RNNLM

Mikolov et al (2010)

Model $p(w_t|w_{t-1}, \ldots w_{t-n})$ with an RNN

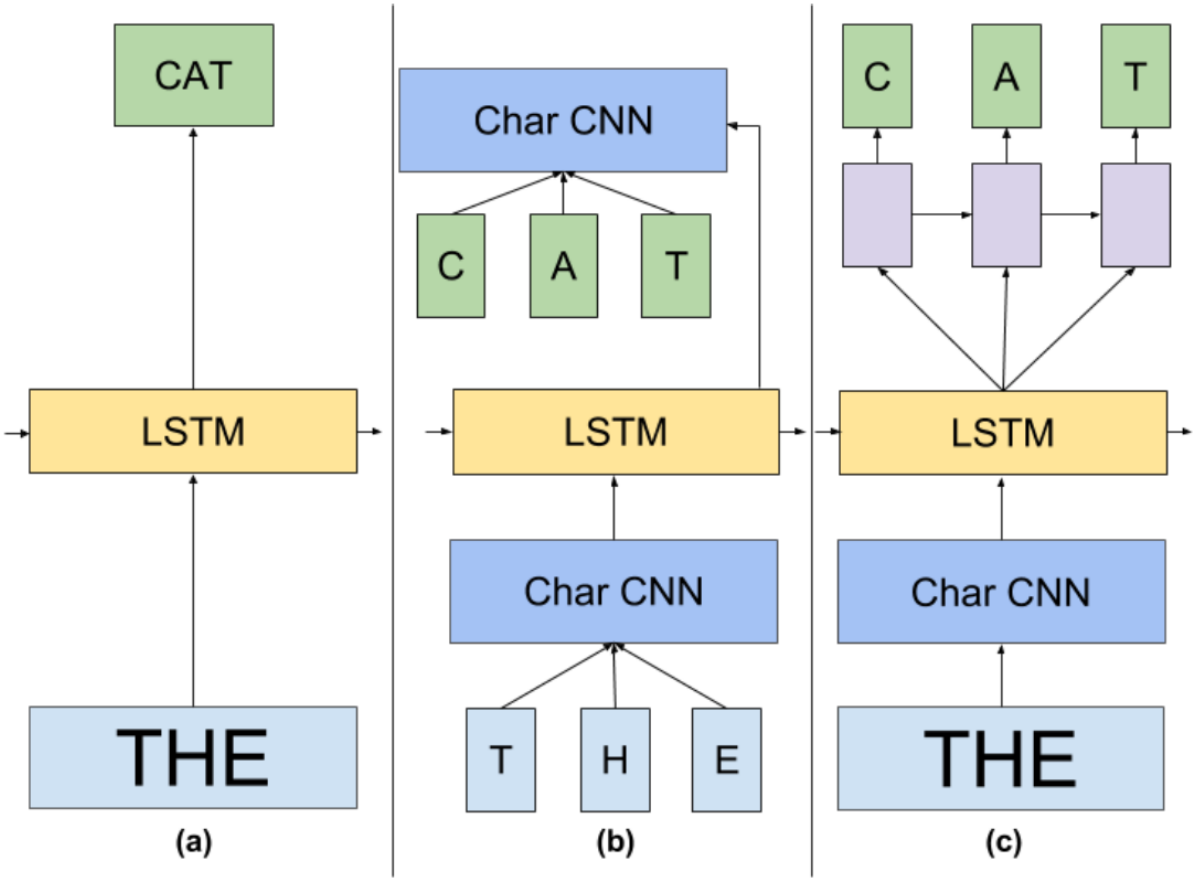
Or an ensemble of multiple RNNs, randomly initialized



Comparison of single RNN vs RNN ensembles

RNNLMs

Jozefowicz et al (2015)



Recent models with character-CNN inputs and softmax alternatives

RNNLMs

Jozefowicz et al (2015)

Word	Top-1	Top-2	Top-3
incerdible	incredible	nonedible	extendible
www.a.com	www.aa.com	www.aaa.com	www.ca.com
7546	7646	7534	8566
TownHal1	TownHall	DJc2	Moodswing360
Komarski	Koharski	Konarski	Komanski

Nearest neighbors in character-CNN embedding space