POS Tagging and Sequence Models

JURAFSKY AND MARTIN CHAPTER 8

Recap: Sequence Models

A **sequence model** or **sequence classifier** is a model whose job is to assign a label or class to each unit in a sequence, thus mapping a sequence of observations to a sequence of labels.

A Hidden Markov Model (HMM) is a probabilistic sequence model: given a sequence of words, it computes a probability distribution over possible sequences of labels and chooses the best label sequence.

Unknown words

One useful feature for distinguishing parts of speech is **word shape** (proper nouns start with a capital).

The strongest feature is **morphology**.

Words that end in

- **-s** tend to be **plural nouns (NNS)**
- **-ed** tend to be **past participles (VBN)**
- **-able** tend to be **adjectives (JJ)**
- and so on

Learning suffix model

Store the final letter sequence (suffixes) for up to 10 letters.

For each such sequence, record the probability of the tag that it was associated with during training.

Use back-off to smooth these probabilities for. Successively shorter sequences.

Trigram HMM with unknown word handling:	96.7%
State of the art neural network POS tagging:	97%

Maximum Entropy Markov Models

Could we add features like word shape and suffixes directly into the model in a clean way? We had this for classification with **logistic regression**. But logistic regression isn't a sequence model, since it assigns a class to a single observation.

We can turn logistic regression into a discriminative sequence model simply by running it on successive words, using the class assigned to the prior word as a feature in the classification of the next word. This is called a **MEMM**.

MEMMs v HMMs

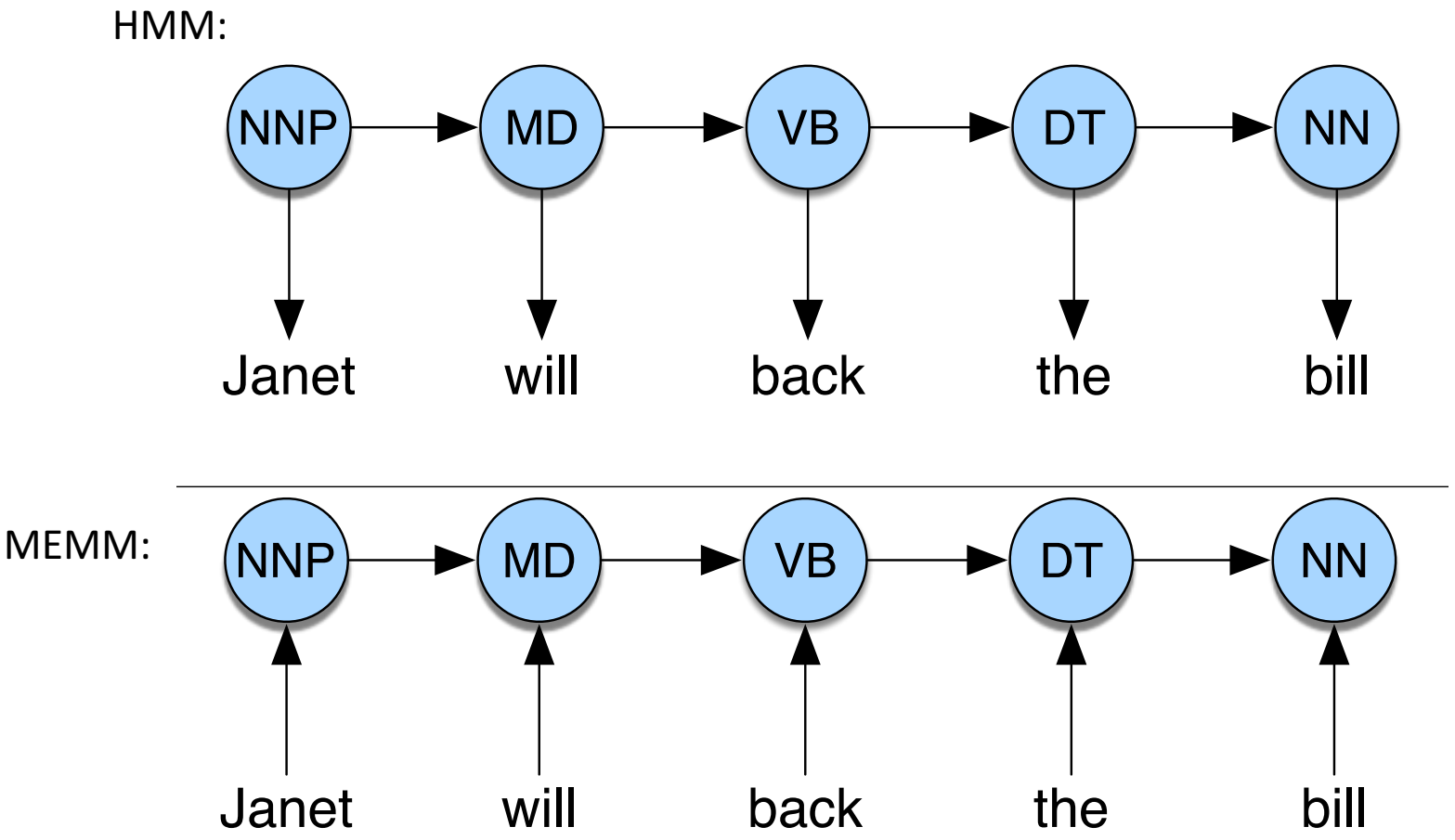
HMM:

$$\begin{aligned}\hat{T} &= \underset{T}{\operatorname{argmax}} P(T|W) \\ &= \underset{T}{\operatorname{argmax}} P(W|T)P(T) \\ &= \underset{T}{\operatorname{argmax}} \prod_i P(word_i|tag_i) \prod_i P(tag_i|tag_{i-1})\end{aligned}$$

MEMM:

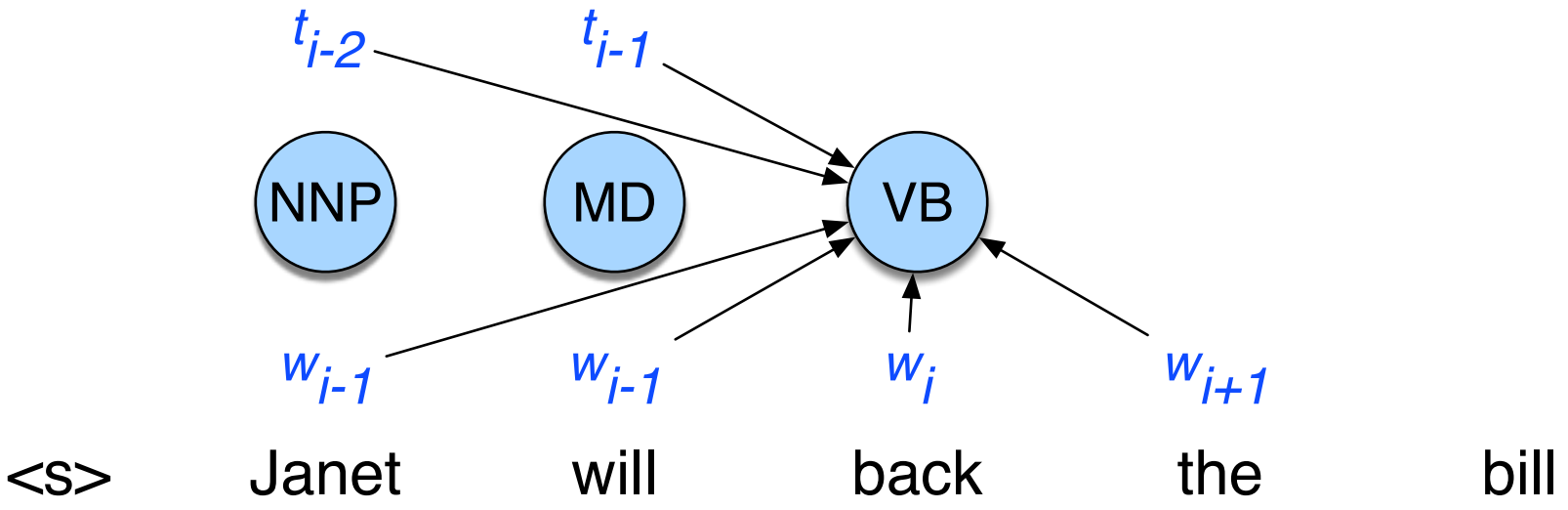
$$\begin{aligned}\hat{T} &= \underset{T}{\operatorname{argmax}} P(T|W) \\ &= \underset{T}{\operatorname{argmax}} \prod_i P(t_i|w_i, t_{i-1})\end{aligned}$$

MEMMs v HMMs


HMM:
NNP
MD
VB
DT
NN
Janet
will
back
the
bill
MEMM:
NNP
MD
VB
DT
NN
Janet
will
back
the
bill

Features in a MEMM

We can build MEMMs that don't just condition on w_i and t_{i-1}. It is easy to incorporate lots of features in a discriminative sequence model.



Feature templates

A basic MEMM part-of-speech tagger conditions on the observation word it- self, neighboring words, and previous tags, and various combinations, using feature templates like the following

$$\langle t_i, w_{i-2}\rangle, \langle t_i, w_{i-1}\rangle, \langle t_i, w_i\rangle, \langle t_i, w_{i+1}\rangle, \langle t_i, w_{i+2}\rangle$$
$$\langle t_i, t_{i-1}\rangle, \langle t_i, t_{i-2}, t_{i-1}\rangle,$$
$$\langle t_i, t_{i-1}, w_i\rangle, \langle t_i, w_{i-1}, w_i\rangle \langle t_i, w_i, w_{i+1}\rangle,$$

Janet/NNP will/MD back/VB the/DT bill/NN, when w_i is the word *back*

$$t_i = \text{VB and } w_{i-2} = \text{Janet}$$
$$t_i = \text{VB and } w_{i-1} = \text{will}$$
$$t_i = \text{VB and } w_i = \text{back}$$
$$t_i = \text{VB and } w_{i+1} = \text{the}$$
$$t_i = \text{VB and } w_{i+2} = \text{bill}$$
$$t_i = \text{VB and } t_{i-1} = \text{MD}$$
$$t_i = \text{VB and } t_{i-1} = \text{MD and } t_{i-2} = \text{NNP}$$
$$t_i = \text{VB and } w_i = \text{back and } w_{i+1} = \text{the}$$

Features for unknown words

w_i contains a particular prefix (from all prefixes of length ≤ 4)
w_i contains a particular suffix (from all suffixes of length ≤ 4)
w_i contains a number
w_i contains an upper-case letter
w_i contains a hyphen
w_i is all upper case
w_i's word shape
w_i's short word shape
w_i is upper case and has a digit and a dash (like *CFC-12*)
w_i is upper case and followed within 3 words by Co., Inc., etc.

Features for *well-dressed*

prefix(w_i) = `w`
prefix(w_i) = `we`
prefix(w_i) = `wel`
prefix(w_i) = `well`
suffix(w_i) = `ssed`
suffix(w_i) = `sed`
suffix(w_i) = `ed`
suffix(w_i) = `d`
has-hyphen(w_i)
word-shape(w_i) = `xxxx-xxxxxxx`
short-word-shape(w_i) = `x-x`

Other sequence labeling tasks

A **sequence model** or **sequence classifier** is a model whose job is to assign a label or class to each unit in a sequence, thus mapping a sequence of observations to a sequence of labels.

What other sequence labeling tasks can you think of?

Named Entity Recognition

Citing high fuel prices, [ORG United Airlines] said [TIME Friday] it has increased fares by [MONEY $6] per round trip on flights to some cities also served by lower-cost carriers. [ORG American Airlines], a unit of [ORG AMR Corp.], immediately matched the move, spokesman [PER Tim Wagner] said. [ORG United], a unit of [ORG UAL Corp.], said the increase took effect [TIME Thursday] and applies to most routes where it competes against discount carriers, such as [LOC Chicago] to [LOC Dallas] and [LOC Denver] to [LOC San Francisco].

Type	Tag	Sample Categories	Example sentences
People	PER	people, characters	**Turing** is a giant of computer science.
Organization	ORG	companies, sports teams	The **IPCC** warned about the cyclone.
Location	LOC	regions, mountains, seas	The **Mt. Sanitas** loop is in **Sunshine Canyon**.
Geo-Political Entity	GPE	countries, states, provinces	**Palo Alto** is raising the fees for parking.
Facility	FAC	bridges, buildings, airports	Consider the **Tappan Zee Bridge**.
Vehicles	VEH	planes, trains, automobiles	It was a classic **Ford Falcon**.

Category Ambiguity in NER

Name	Possible Categories
Washington	Person, Location, Political Entity, Organization, Vehicle
Downing St.	Location, Organization
IRA	Person, Organization, Monetary Instrument
Louis Vuitton	Person, Organization, Commercial Product

Examples of type ambiguities in the use of the name *Washington:*

[$_{\text{PER}}$ Washington] was born into slavery on the farm of James Burroughs.
[$_{\text{ORG}}$ Washington] went up 2 games to 1 in the four-game series.
Blair arrived in [$_{\text{LOC}}$ Washington] for what may well be his last state visit.
In June, [$_{\text{GPE}}$ Washington] passed a primary seatbelt law.
The [$_{\text{VEH}}$ Washington] had proved to be a leaky ship, every passage I made...

Temporal Expression Extraction

Absolute	Relative	Durations
April 24, 1916	yesterday	four hours
The summer of ’77	next semester	three weeks
10:15 AM	two weeks from yesterday	six days
The 3rd quarter of 2006	last quarter	the last three quarters

Lexical triggers for temporal expressions:

Category	Examples
Noun	*morning*, *noon*, *night*, *winter*, *dusk*, *dawn*
Proper Noun	*January, Monday, Ides, Easter, Rosh Hashana, Ramadan, Tet*
Adjective	*recent, past, annual, former*
Adverb	*hourly, daily, monthly, yearly*

Fine grained entity recognition

person	doctor	**organization**	terrorist_organization		
actor	engineer	airline	government_agency		
architect	monarch	company	government		
artist	musician	educational_institution	political_party		
athlete	politician	fraternity_sorority	educational_department		
author	religious_leader	sports_league	military		
coach	soldier	sports_team	news_agency		
director	terrorist				
location	body_of_water	**product**	camera	**art**	written_work
city	island	engine	mobile_phone	film	newspaper
country	mountain	airplane	computer	play	music
county	glacier	car	software		
province	astral_body	ship	game	**event**	military_conflict
railway	cemetery	spacecraft	instrument	attack	natural_disaster
road	park	train	weapon	election	sports_event
bridge				protest	terrorist_attack
building	time	chemical_thing	website		
airport	color	biological_thing	broadcast_network		
dam	award	medical_treatment	broadcast_program		
hospital	educational_degree	disease	tv_channel		
hotel	title	symptom	currency		
library	law	drug	stock_exchange		
power_station	ethnicity	body_part	algorithm		
restaurant	language	living_thing	programming_language		
sports_facility	religion	animal	transit_system		
theater	god	food	transit_line		

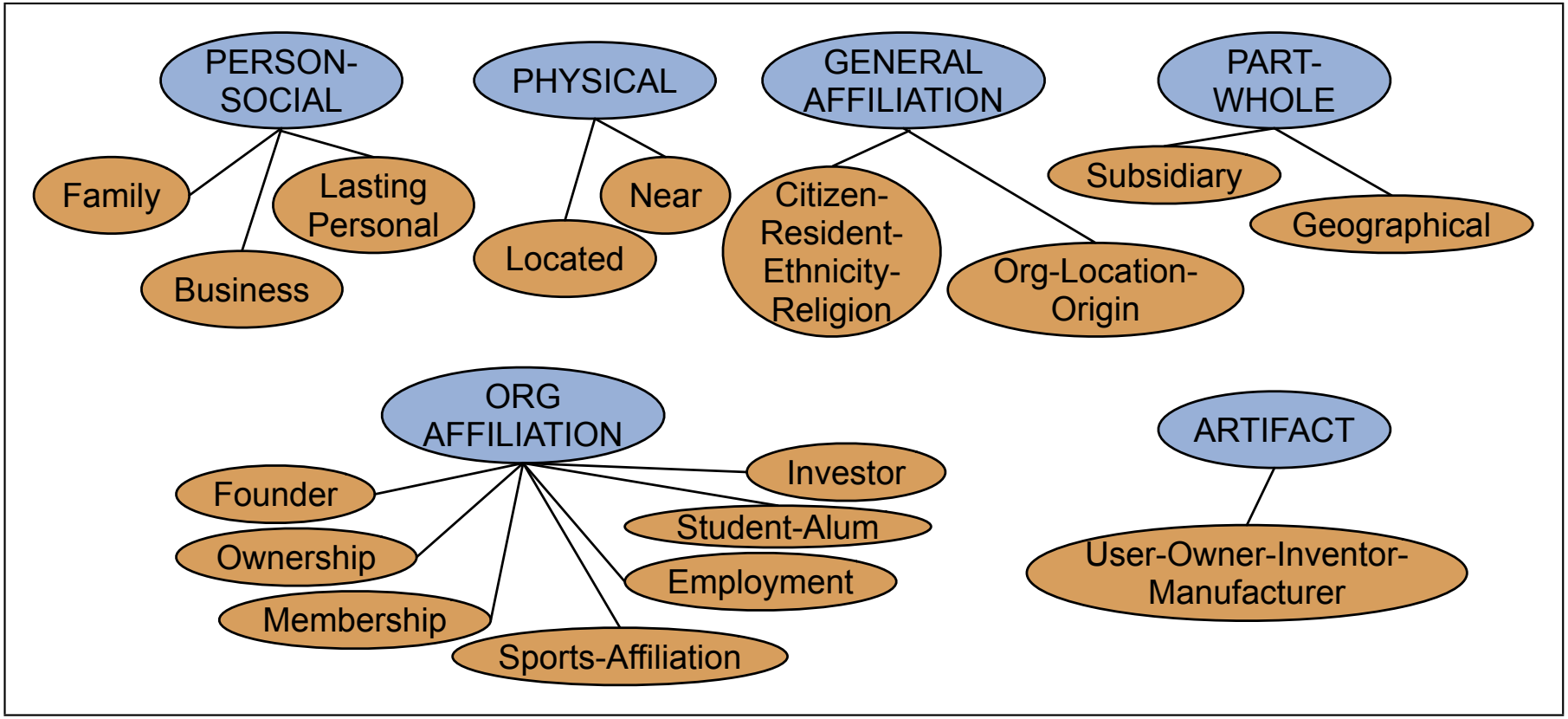
Relation Extraction

Citing high fuel prices, [ORG United Airlines] said [TIME Friday] it has increased fares by [MONEY $6] per round trip on flights to some cities also served by lower-cost carriers. [ORG American Airlines], a unit of [ORG AMR Corp.], immediately matched the move, spokesman [PER Tim Wagner] said. [ORG United], a unit of [ORG UAL Corp.], said the increase took effect [TIME Thursday] and applies to most routes where it competes against discount carriers, such as [LOC Chicago] to [LOC Dallas] and [LOC Denver] to [LOC San Francisco].

Tim Wagner **is a spokesman for** American Airlines

United **is a unit of** UAL Corp

American **is a unit of** AMR



Relations	Types	Examples
Physical-Located	PER-GPE	**He** was in **Tennessee**
Part-Whole-Subsidiary	ORG-ORG	**XYZ**, the parent company of **ABC**
Person-Social-Family	PER-PER	**Yoko**'s husband **John**
Org-AFF-Founder	PER-ORG	**Steve Jobs**, co-founder of **Apple**...

Unified Medical Language System

Entity	Relation	Entity
Injury	disrupts	Physiological Function
Bodily Location	location-of	Biologic Function
Anatomical Structure	part-of	Organism
Pharmacologic Substance	causes	Pathological Function
Pharmacologic Substance	treats	Pathologic Function

Given a medical sentence like:

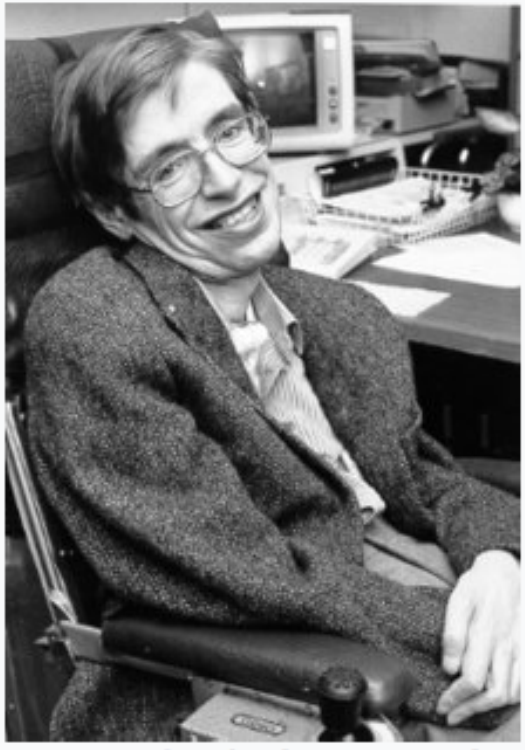
Doppler echocardiography can be used to diagnose left anterior descending artery stenosis in patients with type 2 diabetes

Extract the UMLS relation:

Echocardiography (Doppler) **Diagnoses** *Acquired stenosis*

Wikipedia info boxes

Stephen Hawking
CH CBE FRS FRSA



Hawking at NASA's StarChild Learning Center, ca. 1999

Born	Stephen William Hawking 8 January 1942 Oxford, England
Died	14 March 2018 (aged 76) Cambridge, England
Education	St Albans School, Hertfordshire
Alma mater	University of Oxford (BA) University of Cambridge (MA, PhD)
Known for	Hawking radiation Penrose–Hawking theorems Bekenstein–Hawking formula Hawking energy Gibbons–Hawking ansatz Gibbons–Hawking effect Gibbons–Hawking space Gibbons–Hawking–York boundary term Thorne–Hawking–Preskill bet
Spouse(s)	Jane Wilde (m. 1965; div. 1995) Elaine Mason (m. 1995; div. 2006)
Children	3, including Lucy
Awards	Adams Prize (1966) Eddington Medal (1975) Maxwell Medal and Prize

Your next HW assignment

Develop a NER system for Dutch and Spanish.

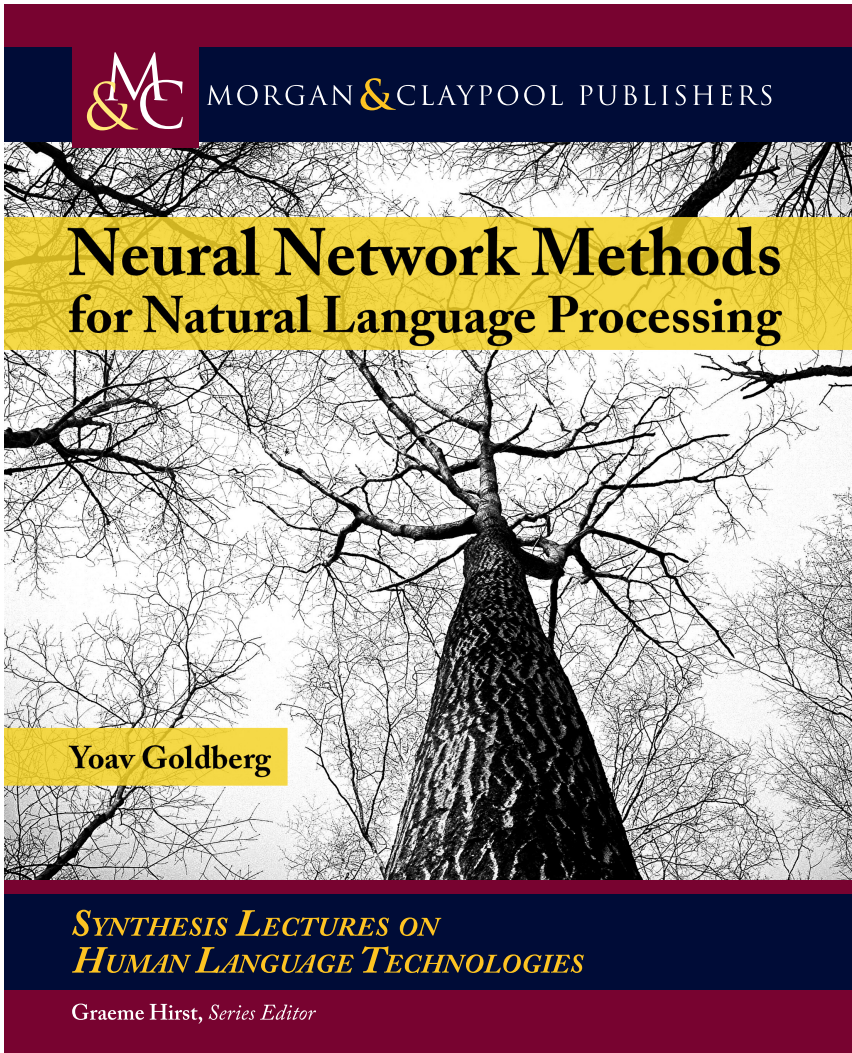
http://computational-linguistics-class.org/assignment7.html

Feed forward vs Recurrent NNs

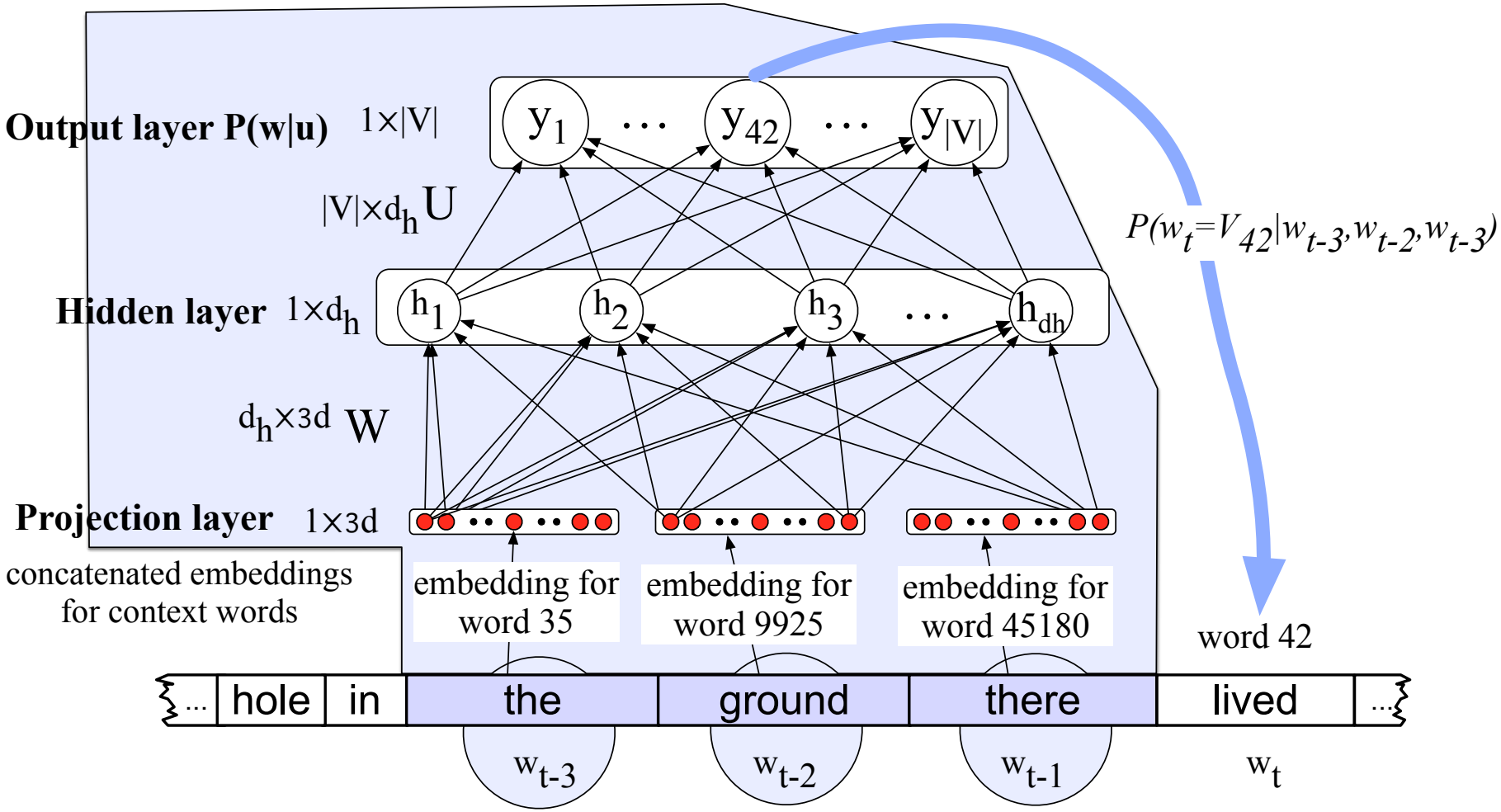
READ CHAPTER 8 AND CHAPTER 14 FROM

YOAV GOLDBER'S BOOK NEURAL NETWORKS METHODS FOR NLP

(IT'S FREE TO DOWNLOAD FROM PENN'S CAMPUS!)



Feed forward neural network



Fixed inputs

Feed-forward networks assume a fixed dimensional input. For feature-extraction (like in our MEMM), we can extract a fixed number of features, and represent each feature as a vector, and then concatenate all of the vectors. Each region of the resulting input vector corresponds to a different feature.

In some cases the number of features is not known in advance.

Variable length inputs

If we want to use each word in the sentence as a feature, then we need to represent a variable number of features using a fixed size vector.

One way of achieving this is through a so-called continuous bag of words (CBOW) representation. We discard order information, and either sum or average the embeddings

$$\mathrm{CBOW}(f_1, \ldots, f_k) = \frac{1}{k} \sum_{i=1}^{k} v(f_i)$$

Are feed forward networks good for sequences?

When dealing with language data, it is very common to work with sequences, such as words (sequences of letters), sentences (sequences of words), and documents.

Feed-forward networks can accommodate sequences through the use of vector concatenation and vector addition (CBOW). CBOW representations allows to encode arbitrary length sequences as fixed sized vectors. However, CBOW is limited, and it disregards order.

Recurrent neural networks (RNNs)

RNNs can represent any size sequential inputs as fixed-size vectors. They model the structured properties of the input.

RNNs are very powerful at capturing statistical regularities in sequential inputs. They are arguably the strongest contribution of deep-learning to the statistical NLP tool-kit.

There are various types of RNNs: simple RNNs, Long-short-term Memory (LSTM) and the Gated Recurrent Unit (GRU).

Recurrent neural networks (RNNs)

RNNs allow for language models that do not make the Markov assumption, and condition the next word on the entire sentence history (all the words preceding it).

This ability opens the way to *conditioned generation models*, where a language model that is used as a generator is conditioned on some other signal, such as a sentence in another language.

We'll talk about this application when we come to Neural Machine Translation next week.

RNN Abstraction

$$\boldsymbol{y_{1:n}} = \text{RNN}^{\star}(\boldsymbol{x_{1:n}})$$

$$\boldsymbol{y_i} = \text{RNN}(\boldsymbol{x}_{1:i})$$

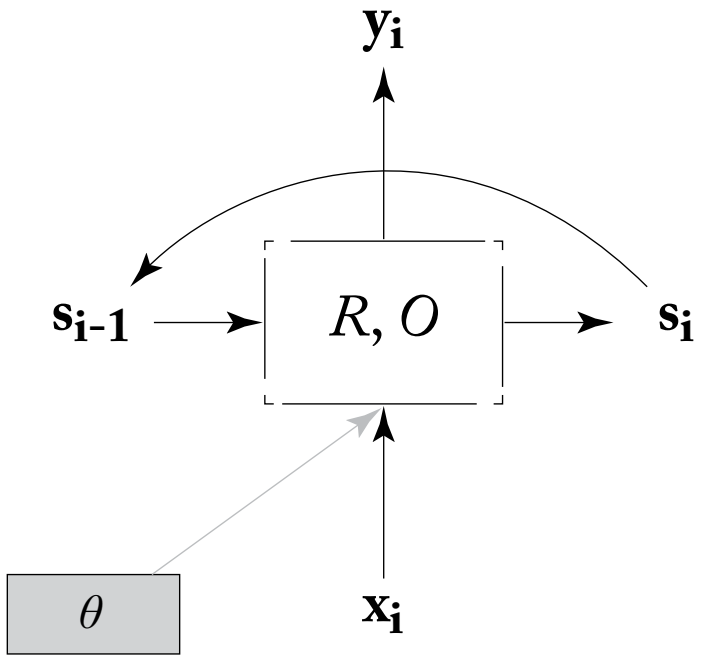
$$\boldsymbol{x_i} \in \mathbb{R}^{d_{in}} \quad \boldsymbol{y_i} \in \mathbb{R}^{d_{out}}$$

RNN Abstraction

$$\text{RNN}^{\star}(\boldsymbol{x_{1:n}}; \boldsymbol{s_0}) = \boldsymbol{y_{1:n}}$$

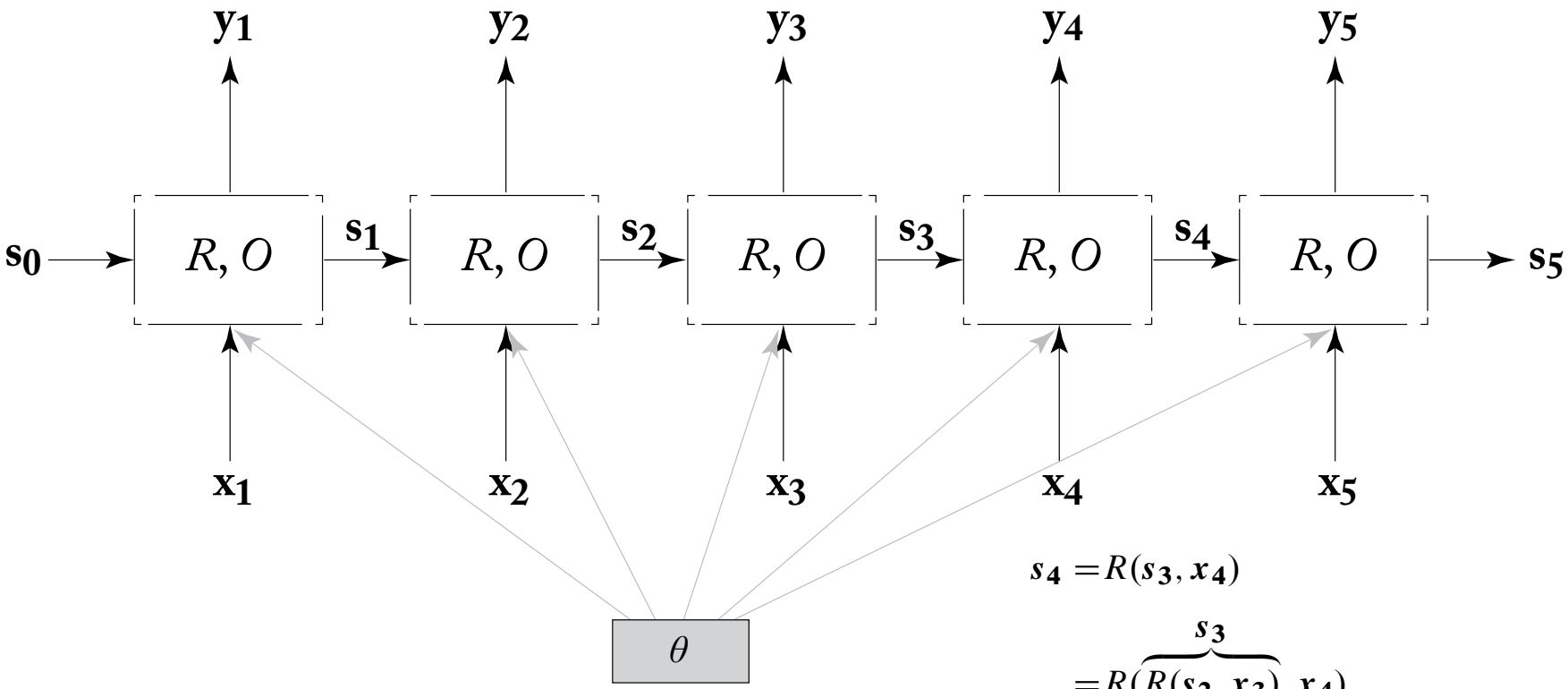
$$\boldsymbol{y_i} = O(\boldsymbol{s_i})$$

$$\boldsymbol{s_i} = R(\boldsymbol{s_{i-1}}, \boldsymbol{x_i})$$



$$\boldsymbol{x_i} \in \mathbb{R}^{d_{in}}, \quad \boldsymbol{y_i} \in \mathbb{R}^{d_{out}}, \quad \boldsymbol{s_i} \in \mathbb{R}^{f(d_{out})}$$

RNN unrolled



$$s_4 = R(s_3, x_4)$$

$$= R(\overbrace{R(s_2, x_3)}^{s_3}, x_4)$$

$$= R(R(\overbrace{R(s_1, x_2)}^{s_2}, x_3), x_4)$$

$$= R(R(R(\overbrace{R(s_0, x_1)}^{s_1}, x_2), x_3), x_4)$$

RNN training

To train an RNN network, all we need to do is to create the unrolled computation graph for a given input sequence, add a loss node to the unrolled graph, and then use the backpropogation algorithm to compute the gradients with respect to that loss. For RNNs, This is referred as *backpropagation through time* (BPTT)

To do this, we need a training objective.

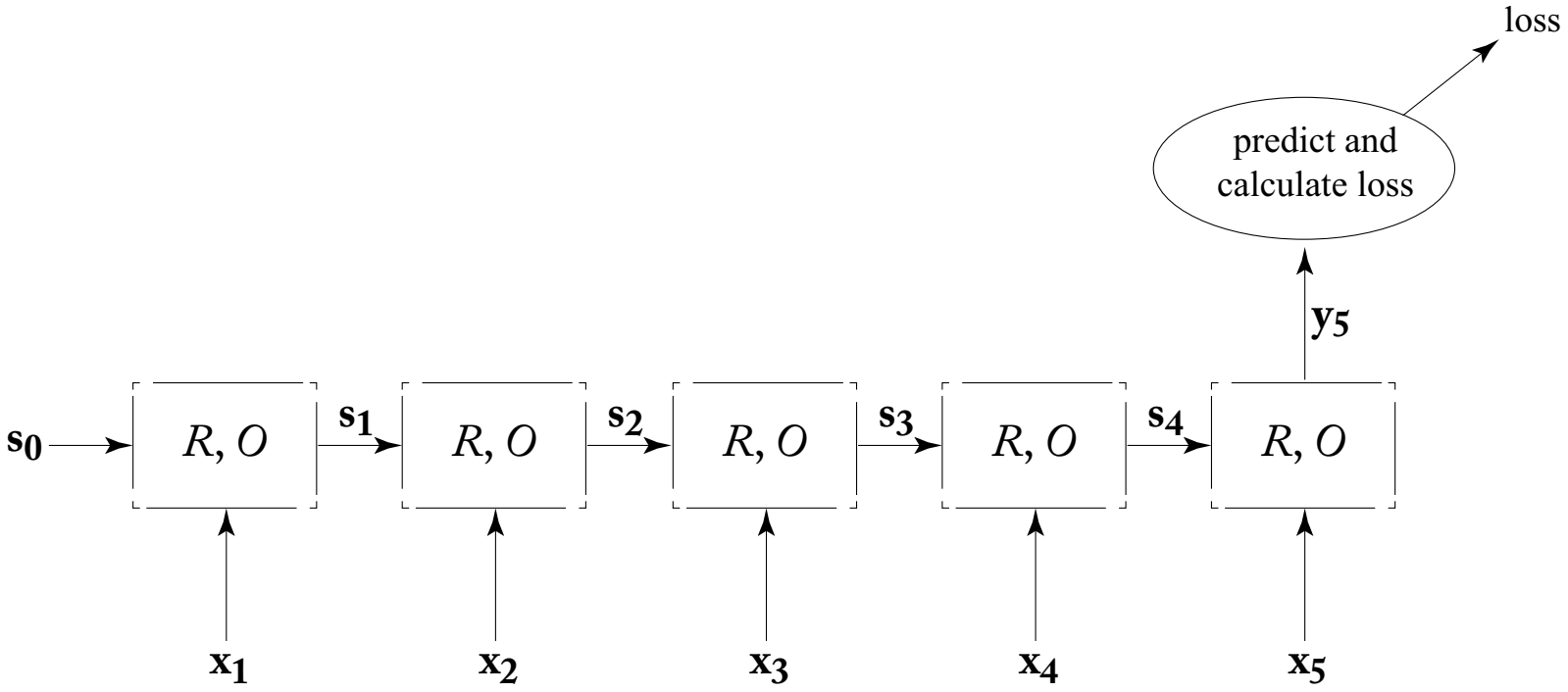
What is the objective of the training?

The RNN does not do much on its own. Instead, it serves as a trainable component in a larger network.

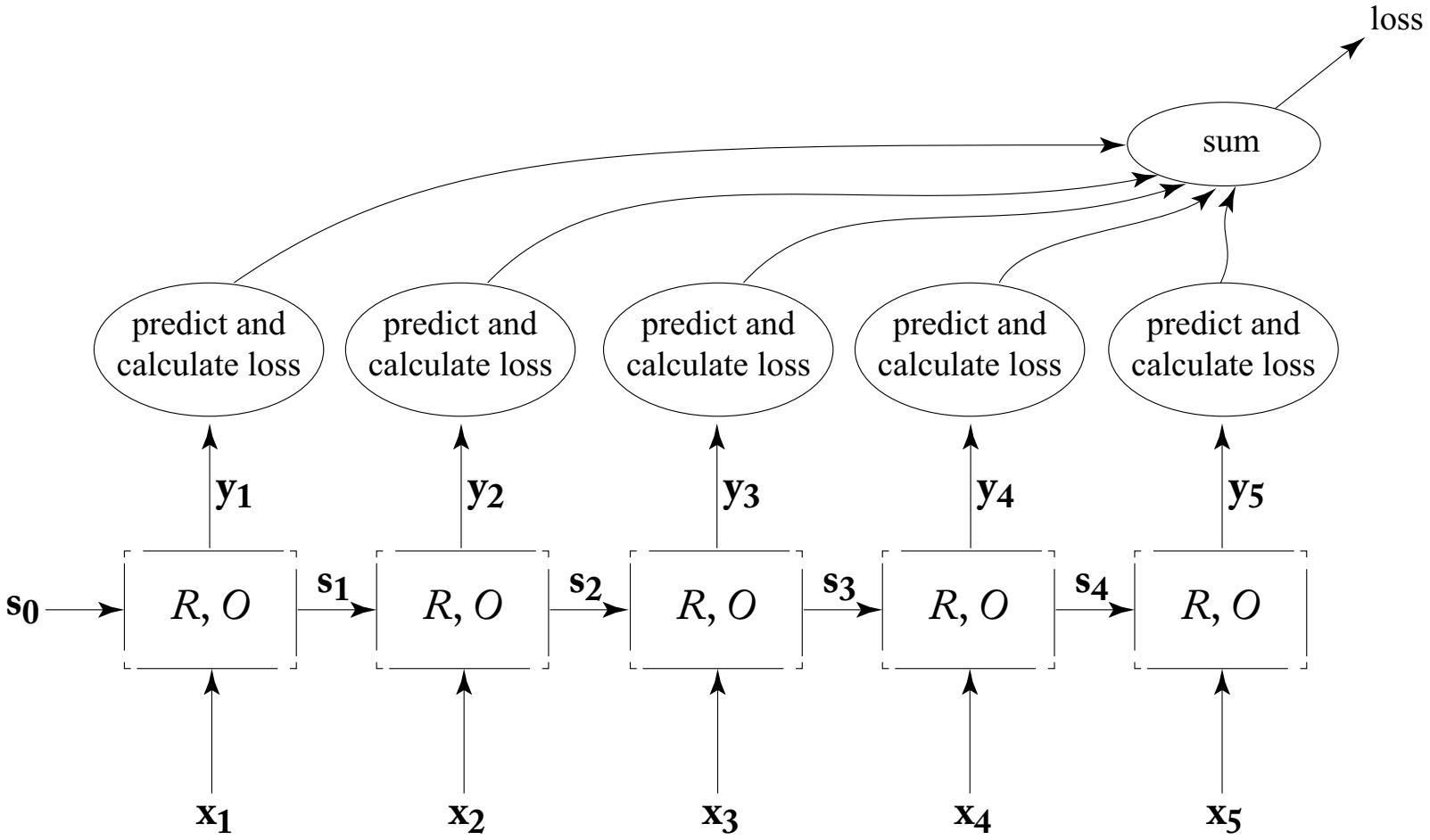
The final prediction and loss computation are performed by that larger network, and the error is back-propagated through the RNN.

This way, the RNN learns to encode properties of the input sequences that are useful for the further prediction task.

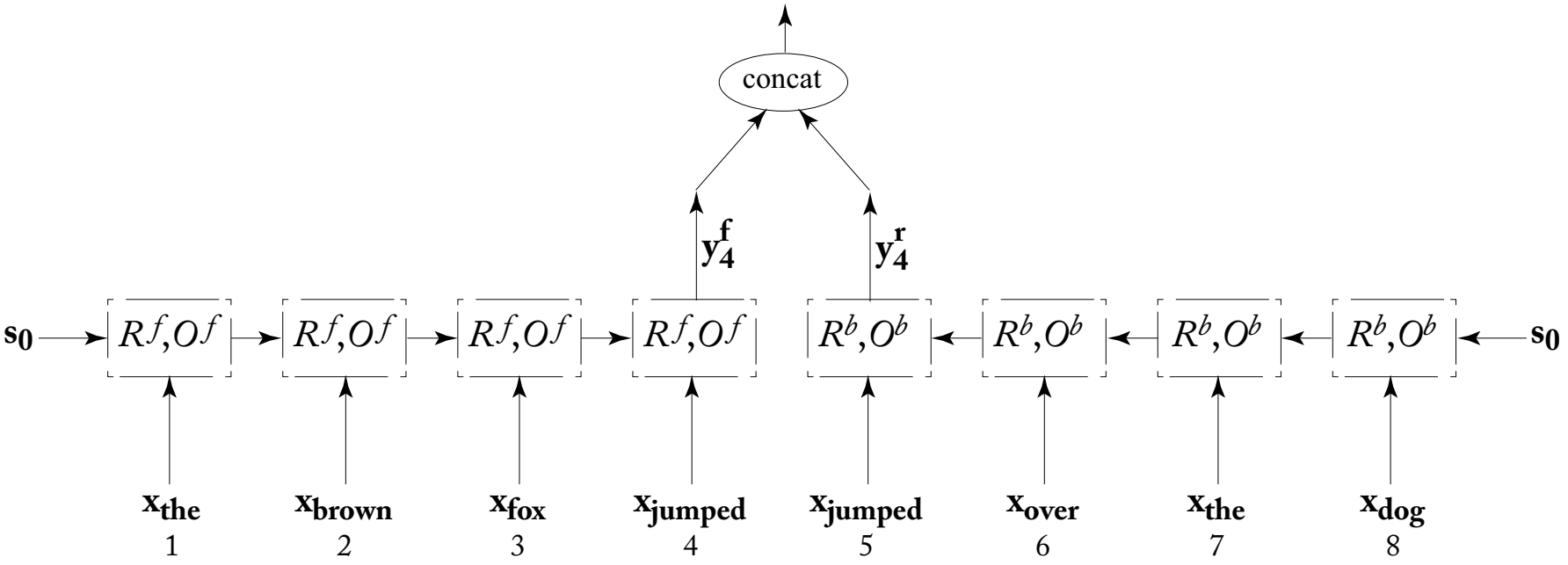
Acceptor RNN training graph



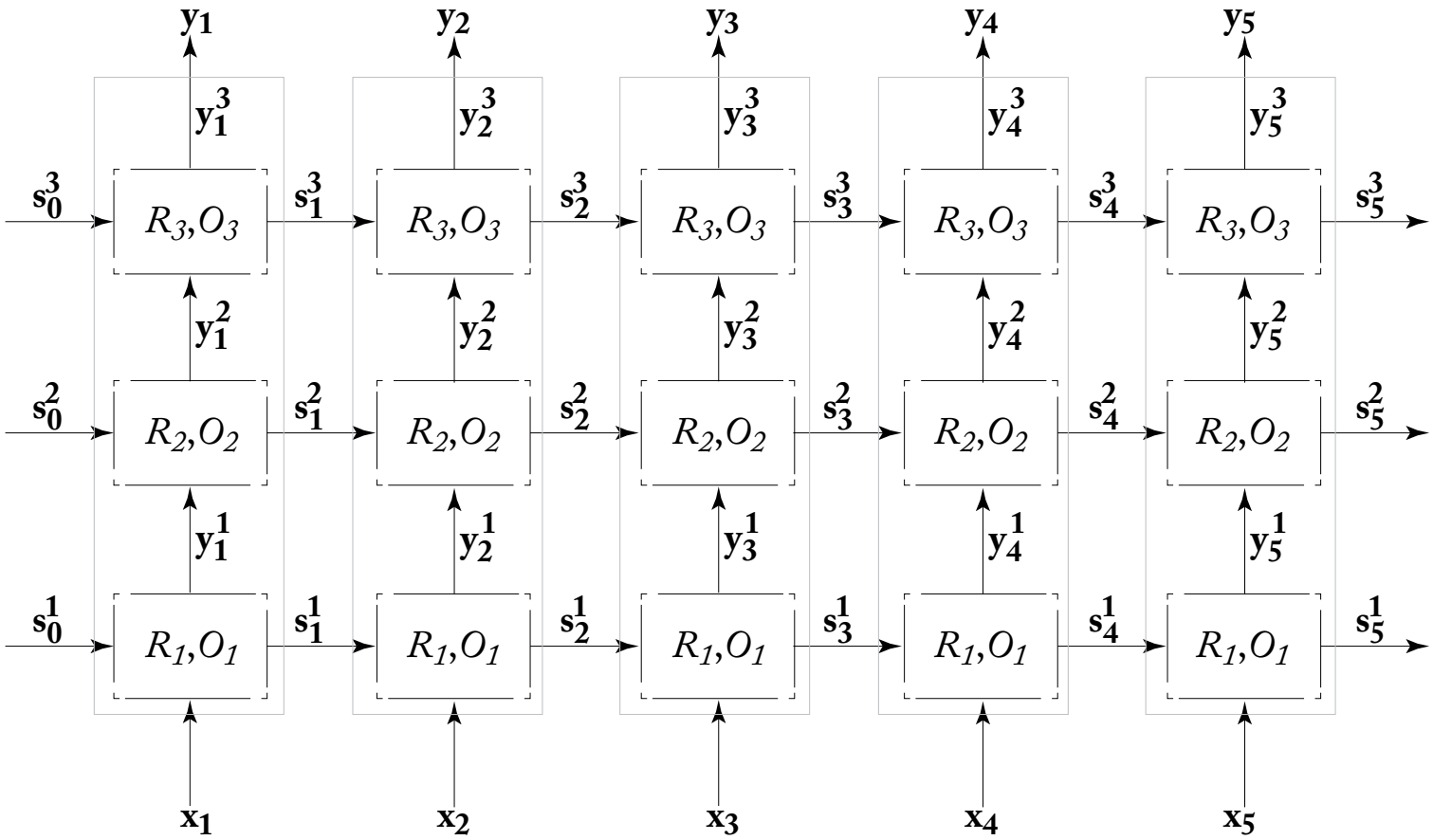
Transducer RNN training graph



Bidirectional RNNs (biRNNs)



Deep RNNs


y_1 y_2 y_3 y_4 y_5
y_1^3 y_2^3 y_3^3 y_4^3 y_5^3
s_0^3 R_3,O_3 s_1^3 R_3,O_3 s_2^3 R_3,O_3 s_3^3 R_3,O_3 s_4^3 R_3,O_3 s_5^3
y_1^2 y_2^2 y_3^2 y_4^2 y_5^2
s_0^2 R_2,O_2 s_1^2 R_2,O_2 s_2^2 R_2,O_2 s_3^2 R_2,O_2 s_4^2 R_2,O_2 s_5^2
y_1^1 y_2^1 y_3^1 y_4^1 y_5^1
s_0^1 R_1,O_1 s_1^1 R_1,O_1 s_2^1 R_1,O_1 s_3^1 R_1,O_1 s_4^1 R_1,O_1 s_5^1
x_1 x_2 x_3 x_4 x_5