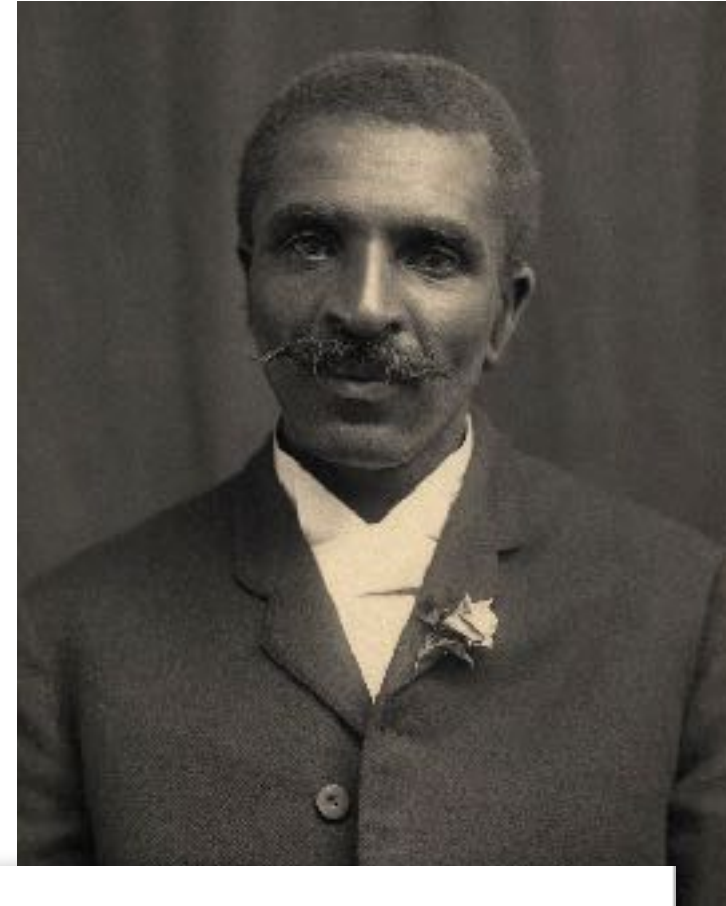
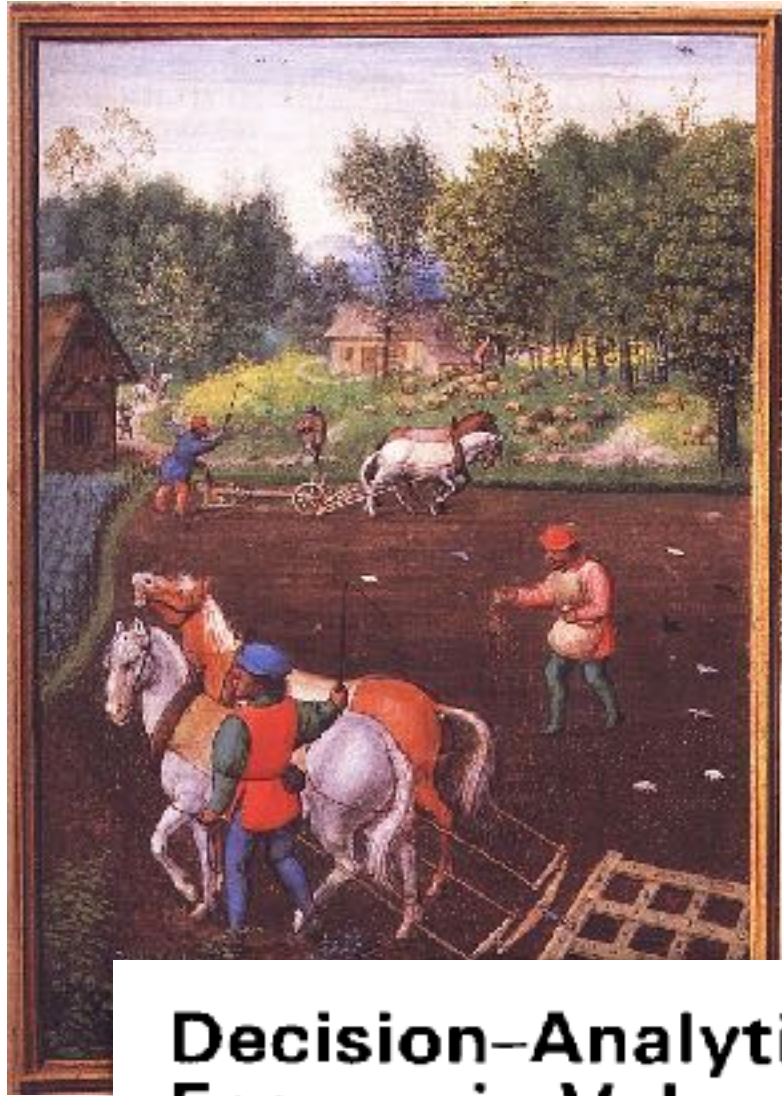


Recurrent Neural Networks

Prof. Tamara Broderick

Edited From 6.036 Fall21 Offering



Decision-Analytic Assessment of the Economic Value of Weather Forecasts: The Fallowing/Planting Problem

RICHARD W. KATZ

National Center for Atmospheric Research, U.S.A.

and

BARBARA G. BROWN* and ALLAN H. MURPHY

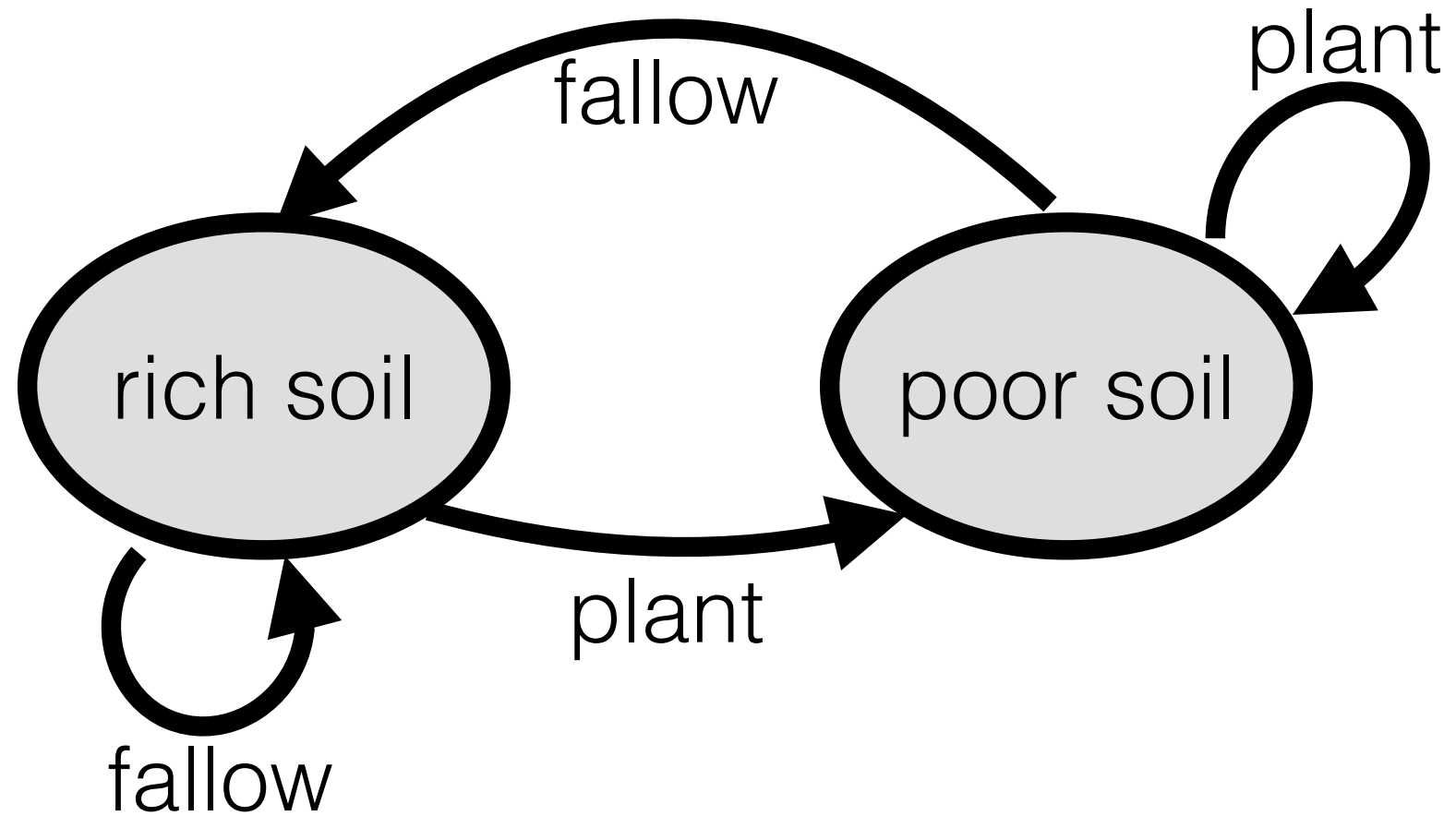
Oregon State University, U.S.A.

[https://en.wikipedia.org/wiki/Sowing#/media/File:Simon_Bening_-_September.jpg]

[https://en.wikipedia.org/wiki/George_Washington_Carver#/media/File:George_Washington_Carver_c1910_-_Restoration.jpg]

State Machine

- $\mathcal{S}$ = set of possible states
- $\mathcal{X}$ = set of possible inputs
- $s_0 \in \mathcal{S}$: initial state
- $f: \mathcal{S} \times \mathcal{X} \rightarrow \mathcal{S}$: transition function
- $\mathcal{Y}$: set of possible outputs
- $g: \mathcal{S} \rightarrow \mathcal{Y}$: output function
 - e.g. $g(s) = s$
 - e.g. $g(s) = \text{soil-moisture-sensor}(s)$



Example

$s_0 = \text{rich}$
 $s_1 = f(s_0, \text{plant}) = \text{poor};$
 $y_1 = g(s_1) = \text{poor}$
 $s_2 = f(s_1, \text{fallow}) = \text{rich};$
 $y_2 = g(s_2) = \text{rich}$

Text prediction

Final product

VicePresidentOfCompany@HopefullyNotARealEmailA..

Final product

All the documents are finished. Please see attached



WIKIPEDIA
The Free Encyclopedia

English

6 183 000+ articles

Español

1 637 000+ artículos

日本語

1 235 000+ 記事

Deutsch

2 495 000+ Artikel

Русский

1 672 000+ статей

Français

2 262 000+ articles

Italiano

1 645 000+ voci

中文

1 155 000+ 條目



Português

1 045 000+ artigos

Polski

1 435 000+ haseł

autocomp

EN



Autocomplete

Application that predicts the rest of a word a user is typing.

Search suggest drop-down list

Automotive industry in India

Automotive industry in the United States

ivoyage

e travel guide

inews

e news source

Text prediction: supervised learning

- Training data: lots of text
 - “what happens to a dream deferred”

features	label
w	h
wh	a
wha	t
what	–
what_	h
what_h	a
what_ha	p
what_hap	p
what_happ	e

- Classification with 27 classes
- How to featurize?
- Idea: use all previous characters. But so far we’ve said $x^{(i)} \in \mathbb{R}^d$; i.e. fixed dimension
- Idea: just use last character. But lose info
- Idea: use last m characters

Text prediction: supervised learning

- Training data: lots of text
 - “what happens to a dream deferred”

features	label
w	h
wh	a
wha	t
what	–
what_	h
what_h	a
what_ha	p
what_hap	p
what_happ	e

- Classification with 27 classes
- How to featurize?
- Idea: use all previous characters. But so far we’ve said $x^{(i)} \in \mathbb{R}^d$; i.e. fixed dimension
- Idea: just use last character. But lose info
- Idea: use last $m = 3$ characters

A state machine: writing & predicting text

- Recall state machines:
 - Set of possible states $\mathcal{S}$
 - Set of possible inputs $\mathcal{X}$
 - Initial state
 - Transition function $f(s, x)$
 - Set of possible outputs
 - Output function $p = g(s)$
- Example:
 - Every sequence of 3 chars
 - Every single character
 - $s_0 = \wedge\wedge\wedge$
 - E.g. $f(\wedge\wedge w, h) = \wedge wh$
 - Output: 27 prediction probs
 - E.g. $p = g(\wedge\wedge\wedge) = [0.08, 0.02, \dots, 0.001, 0.01]$

previously “y”

t	$x_t^{(1)}$	$s_t^{(1)}$
0		$\wedge\wedge\wedge$
1	w	$\wedge\wedge w$
2	h	$\wedge wh$
3	a	wha
4	t	hat
5	_	at_

- $x^{(1)}$: “what happens to a dream deferred”
- $x^{(2)}$: “if you can keep your head when all about you”
- $x^{(3)}$: “you may write me down in history”

A state machine: writing & predicting text

- Recall state machines:
 - Set of possible states $\mathcal{S}$
 - Set of possible inputs $\mathcal{X}$
 - Initial state
 - Transition function $f(s, x)$
 - Set of possible outputs
 - Output function $p = g(s)$
- Example:
 - Every sequence of 3 chars
 - Every single character
 - s_0
 - E.g. $f\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}, x\right) = \begin{bmatrix} c_2 \\ c_3 \\ x \end{bmatrix}$
 - Output: v prediction probs
 - E.g. $p = g\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}\right)$
- Example with alphabet $\{0, 1\}$ ($v=2$); state is last $m = 3$ chars

$$s_t = f_1 \left(\begin{bmatrix} w_1^{sx} \\ w_2^{sx} \\ w_3^{sx} \end{bmatrix} x_t + \begin{bmatrix} w_{11}^{ss} & w_{12}^{ss} & w_{13}^{ss} \\ w_{21}^{ss} & w_{22}^{ss} & w_{23}^{ss} \\ w_{31}^{ss} & w_{32}^{ss} & w_{33}^{ss} \end{bmatrix} s_{t-1} + \begin{bmatrix} w_{0,1}^{ss} \\ w_{0,2}^{ss} \\ w_{0,3}^{ss} \end{bmatrix} \right)$$

$$\begin{aligned} p_t &= g(s_t) \\ &= f_2(W^o s_t + W_0^o) \end{aligned}$$

same as final layer in all of our neural networks so far, except for notation

A state machine: writing & predicting text

- Recall state machines:
 - Set of possible states $\mathcal{S}$
 - Set of possible inputs $\mathcal{X}$
 - Initial state
 - Transition function
 - Set of possible outputs
 - Output function $o(s) = g(s)$
- Example:
 - Every sequence of 3 chars
 - Every single character
 - s_0
 - E.g. $f\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}, x\right) = \begin{bmatrix} c_2 \\ c_3 \\ x \end{bmatrix}$
 - Output: v prediction probs
 - E.g. $p = g\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}\right)$
- Example: alphabet $\{0, 1\}$ ($v=2$); state is last $m = 3$ chars

$$s_t = f_1 \left(\begin{bmatrix} w_1^{sx} \\ w_2^{sx} \\ w_3^{sx} \end{bmatrix} x_t + \begin{bmatrix} w_{11}^{ss} & w_{12}^{ss} & w_{13}^{ss} \\ w_{21}^{ss} & w_{22}^{ss} & w_{23}^{ss} \\ w_{31}^{ss} & w_{32}^{ss} & w_{33}^{ss} \end{bmatrix} s_{t-1} + \begin{bmatrix} w_{0,1}^{ss} \\ w_{0,2}^{ss} \\ w_{0,3}^{ss} \end{bmatrix} \right)$$

$$\begin{aligned} p_t &= g(s_t) \\ &= f_2(W^o s_t + W_0^o) \end{aligned}$$

same as final layer in all of our neural networks so far, except for notation

A state machine: writing & predicting text

- Recall state machines:
 - Set of possible states $\mathcal{S}$
 - Set of possible inputs $\mathcal{X}$
 - Initial state
 - Transition function $f(s, x)$
 - Set of possible outputs
 - Output function $p = g(s)$
- Example:
 - Every sequence of 3 chars
 - Every single character
 - s_0
 - E.g. $f\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}, x\right) = \begin{bmatrix} c_2 \\ c_3 \\ x \end{bmatrix}$
 - Output: v prediction probs
 - E.g. $p = g\left(\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}\right)$
- Example with alphabet $\{0, 1\}$ ($v=2$); state is last $m = 3$ chars

$$s_t = f_1(W^{sx}x_t + W^{ss}s_{t-1} + W_0^{ss})$$

$$\begin{aligned} p_t &= g(s_t) \\ &= f_2(W^o s_t + W_0^o) \end{aligned}$$

same as final layer in all of our neural networks so far, except for notation

A state machine: writing & predicting text

- Recall state machines:
 - Set of possible states $\mathcal{S}$
 - Set of possible transitions
 - Initial state
 - Transition function
 - Set of possible outputs
 - Output function
- Example:
 - Every sequence of 3 chars
 - Character
 - Transition probs

Recall:

- In 2-layer, fully-connected NNs, we learned the features.
- In CNNs, we learned the filters.
- Analogous idea here.

$$s_t = f_1(W^{sx}x_t + W^{ss}s_{t-1} + W_0^{ss})$$

$$\begin{aligned} p_t &= g(s_t) \\ &= f_2(W^o s_t + W_0^o) \end{aligned}$$

Put it all together:

$$p^{(i)} = \text{RNN}(x^{(i)}; W, W_0)$$

“recurrent neural network”

Recurrent Neural Networks

Recall: familiar pattern

1. Choose how to predict label (given features & parameters)
2. Choose a loss (between guess & actual label)
3. Choose parameters by trying to minimize the training loss

In our example application:

1. Use an RNN: $p^{(i)} = \text{RNN}(x^{(i)}; W, W_0)$

$$s_t = f_1(W^{sx}x_t + W^{ss}s_{t-1} + W_0^{ss})$$

$$p_t = f_2(W^os_t + W_0^o)$$

2. Loss: for q sequences, where the i th has length $n^{(i)}$

$$\text{sequence } L_{\text{seq}}(p^{(i)}, y^{(i)}) = \sum_{t=1}^{n^{(i)}} L_{\text{elt}}(p_t^{(i)}, y_t^{(i)}) \text{ element}$$

$$J(W, W_0) = \sum_{i=1}^q L_{\text{seq}}(p^{(i)}, y^{(i)})$$

3. Stochastic gradient descent

The collage consists of several overlapping elements:

- Top Left:** The word "nature" in a large, bold, black font.
- Top Center:** A navigation bar with a hamburger menu icon, the letters "n p r" in a red box, and a "Subscribe" button.
- Top Right:** The text "MIT Technology Review" and another "Subscribe" button.
- Middle Left:** A section titled "TechRepublic" with a logo and the text "Mach new e". Below it is a green circular logo with a white speech bubble.
- Middle Center:** A section titled "REUTERS" with a logo and the text "A small g". Below it is a grid of small portraits of men.
- Middle Right:** A section titled "SCIENTIFIC AMERICAN" with a logo and the text "COMPUTING". Below it is the headline "Artificial Intelligence Develops an Ear for Birdsong" and a sub-headline "Machine-learning algorithms can quickly process thousands of hours of natural soundscapes". The author is listed as "By Harini Barath on April 26, 2021".
- Bottom Left:** A section titled "De str of" with a large green circular logo and a portrait of a man. Below it is the text "A new mach massive dat".
- Bottom Center:** A section titled "The pat" with a logo and the text "Publish". Below it is a portrait of a man and the text "A ma demo predi".
- Bottom Right:** A photograph of a yellow bird perched on a branch, singing.

- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate
- **Notes:** ML is a tool with pros & cons. ML is built on math

Thank you!