Recurrent Neural Networks

CISC 7026: Introduction to Deep Learning

University of Macau

Agenda

1. Review
2. Sequence Modeling
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Agenda

1. **Review**
2. Sequence Modeling
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Agenda

1. **Review**
2. Sequence Modeling
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Agenda

1. Review
2. **Sequence Modeling**
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Sequence Modeling

- Conv is good at some Things
- Generalizes to n dimensions, etc
- Shortcomings - locality not good in all cases
- Equivariance not good in all cases
- Imagine memories created over a lifetime
  - Targeted remembering and forgetting of important events
- RNNs developed specifically for temporal problems
- Are there other ways to process sequences

Convolution is not the only way to process sequential data

Sequence Modeling

We previously used convolution to model signals

Sequence Modeling

We previously used convolution to model signals

Some signals, such as audio, occur over time

Sequence Modeling

We previously used convolution to model signals

Some signals, such as audio, occur over time

Convolution approaches temporal data from an electrical engineering approach

Sequence Modeling

We previously used convolution to model signals

Some signals, such as audio, occur over time

Convolution approaches temporal data from an electrical engineering approach

Today, we will process temporal data using a psychological approach

Sequence Modeling

Convolution makes use of locality and translation equivariance properties

Sequence Modeling

Convolution makes use of locality and translation equivariance properties

This makes learning more efficient, but not all problems benefit from locality and equivariance

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 2: Your parent changes your diaper

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 2: Your parent changes your diaper

- Not equivariant! Ok if you are a baby, different meaning if you are an adult!

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 2: Your parent changes your diaper

- Not equivariant! Ok if you are a baby, different meaning if you are an adult!

Example 3: You kiss person A, then 5 years later marry person B

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 2: Your parent changes your diaper

- Not equivariant! Ok if you are a baby, different meaning if you are an adult!

Example 3: You kiss person A, then 5 years later marry person B

- Not equivariant or local! If you marry person B then kiss person A, it is different…

Example 1: You like dinosaurs as a child, you grow up and study dinosaurs for work

- Not local! Two events occur separated by 20 years

Example 2: Your parent changes your diaper

- Not equivariant! Ok if you are a baby, different meaning if you are an adult!

Example 3: You kiss person A, then 5 years later marry person B

- Not equivariant or local! If you marry person B then kiss person A, it is different…

Question: Any other examples?

Sequence Modeling

If your problem has local and equivariant structure, use convolution

Sequence Modeling

If your problem has local and equivariant structure, use convolution

For other problems, we need something else!

Sequence Modeling

If your problem has local and equivariant structure, use convolution

For other problems, we need something else!

Humans experience time and process temporal data

Sequence Modeling

If your problem has local and equivariant structure, use convolution

For other problems, we need something else!

Humans experience time and process temporal data

Can we use this to come up with a new neural network architecture?

Agenda

1. **Review**
2. Sequence Modeling
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Agenda

1. Review
2. **Sequence Modeling**
3. Trace Theory
4. Elman Networks
5. Backpropagation through Time
6. LSTM
7. GRU
8. Linear Recurrent Models
9. Coding

Sequence Modeling

How do humans experience time?

Sequence Modeling

How do humans experience time?

Humans create memories

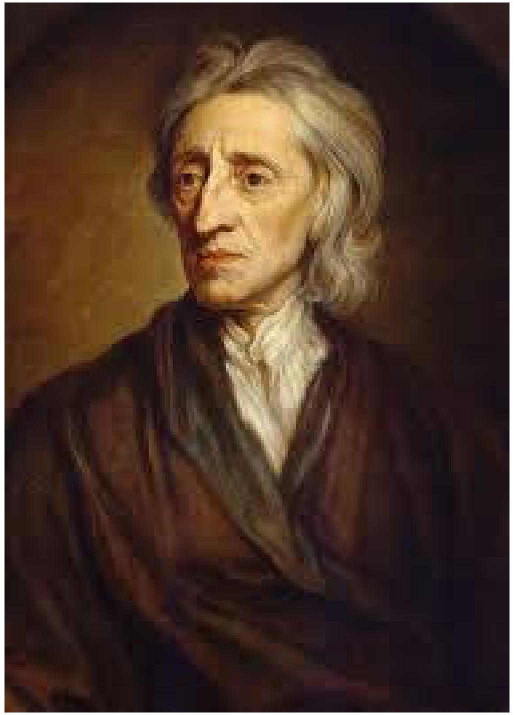
Sequence Modeling

How do humans experience time?

Humans create memories

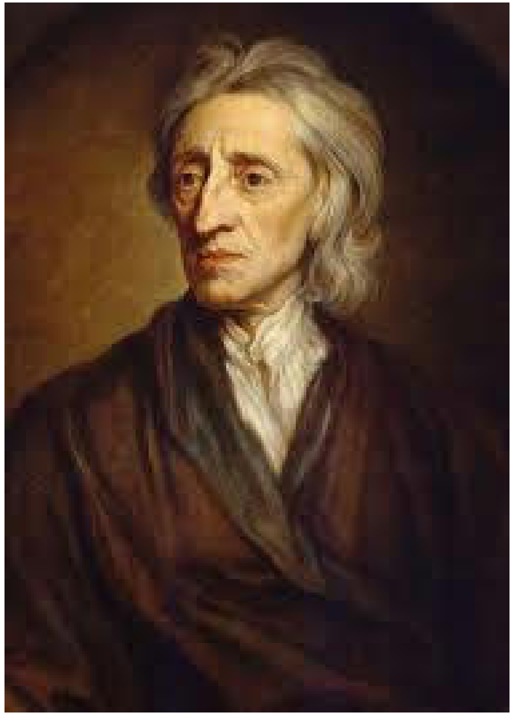
We experience time by reasoning over our memories

Sequence Modeling



John Locke (1690) believed that conciousness and identity arise from memories

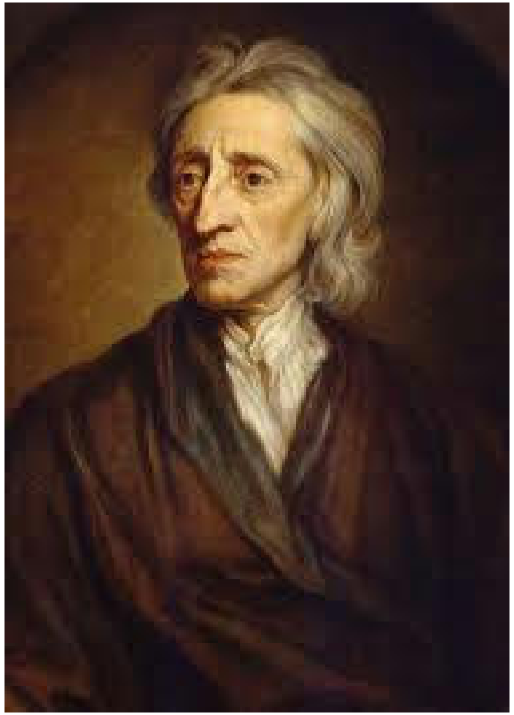
Sequence Modeling



John Locke (1690) believed that conciousness and identity arise from memories

If all your memories were erased, you would be a different person

Sequence Modeling



John Locke (1690) believed that conciousness and identity arise from memories

If all your memories were erased, you would be a different person

Without the ability to reason over memories, we would simply react to stimuli like bacteria

Sequence Modeling

So how do we model memories in humans?

Sequence Modeling

So how do we model memories in humans?

$$f : \underbrace{H}_{\text{Memories}} \times \overbrace{X}^{\text{Sensory information}} \times \underbrace{\Theta}_{\text{Neurons}} \mapsto \overbrace{H}^{\text{Updated memories}}$$

Sequence Modeling

So how do we model memories in humans?

$$f : \underbrace{H}_{\text{Memories}} \times \overbrace{X}^{\text{Sensory information}} \times \underbrace{\Theta}_{\text{Neurons}} \mapsto \overbrace{H}^{\text{Updated memories}}$$

All your memories represented as a vector $\boldsymbol{h} \in H$

Sequence Modeling

So how do we model memories in humans?

$$f : \underbrace{H}_{\text{Memories}} \times \overbrace{X}^{\text{Sensory information}} \times \underbrace{\Theta}_{\text{Neurons}} \mapsto \overbrace{H}^{\text{Updated memories}}$$

All your memories represented as a vector $\boldsymbol{h} \in H$

Everything you currently sense (sight, touch, sound, emotions) is a vector $\boldsymbol{x} \in X$

Sequence Modeling

So how do we model memories in humans?

$$f: \underbrace{H}_{\text{Memories}} \times \overbrace{X}^{\text{Sensory information}} \times \underbrace{\Theta}_{\text{Neurons}} \mapsto \overbrace{H}^{\text{Updated memories}}$$

All your memories represented as a vector $\boldsymbol{h} \in H$

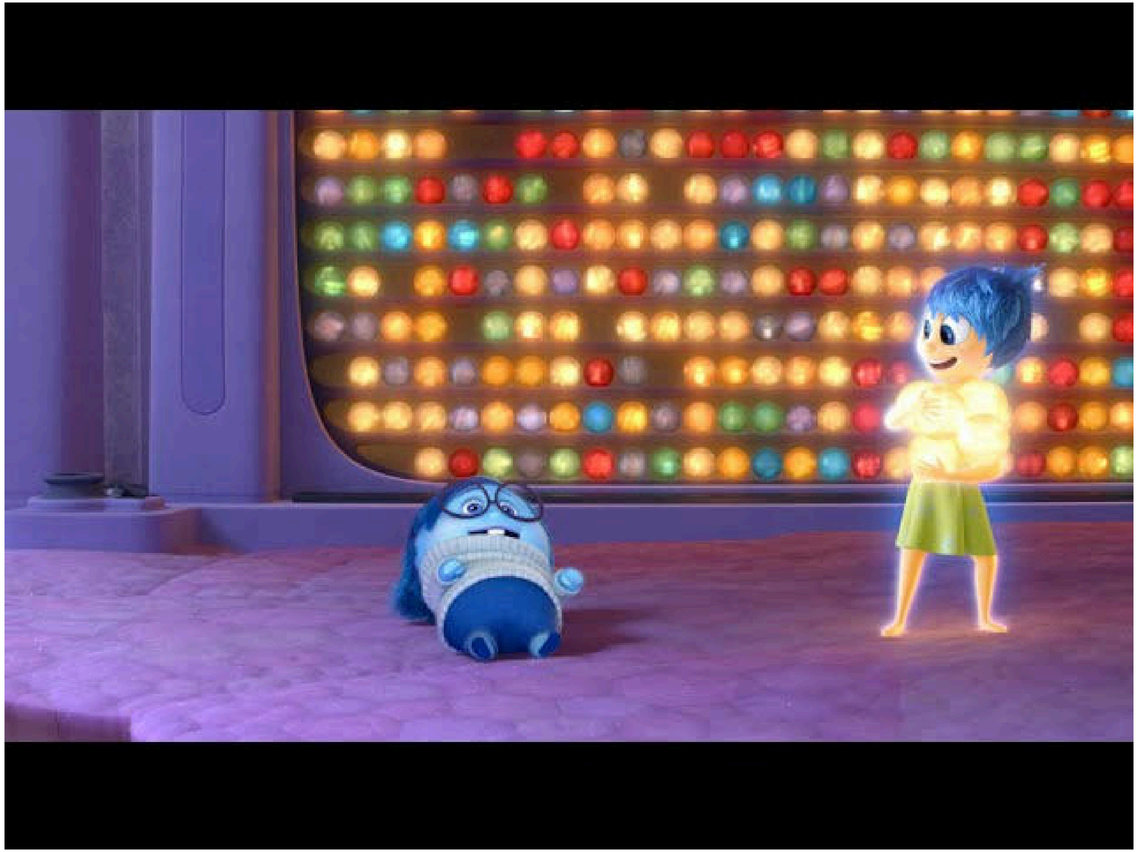
Everything you currently sense (sight, touch, sound, emotions) is a vector $\boldsymbol{x} \in X$

We update our memories following

$$\boldsymbol{s}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1})$$

Sequence Modeling

$$\boldsymbol{s}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1})$$



Sequence Modeling

After we have constructed our memories $\boldsymbol{h}_t$, we do not recall all information at once

Sequence Modeling

After we have constructed our memories $\boldsymbol{h}_t$, we do not recall all information at once

Example: I ask you your favorite ice cream flavor

Sequence Modeling

After we have constructed our memories $\boldsymbol{h}_t$, we do not recall all information at once

Example: I ask you your favorite ice cream flavor

You recall previous times you ate ice cream, but not your phone number

Sequence Modeling

After we have constructed our memories $\boldsymbol{h}_t$, we do not recall all information at once

Example: I ask you your favorite ice cream flavor

You recall previous times you ate ice cream, but not your phone number

We should model this too

$$g : H \times X \times \Theta \mapsto Y$$

Sequence Modeling

After we have constructed our memories $\boldsymbol{h}_t$, we do not recall all information at once

Example: I ask you your favorite ice cream flavor

You recall previous times you ate ice cream, but not your phone number

We should model this too

$$g : H \times X \times \Theta \mapsto Y$$

$$\boldsymbol{y}_t = g(\boldsymbol{h}_t, \boldsymbol{x}_t, \boldsymbol{\theta})$$

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}); \quad \boldsymbol{y}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_t)$$

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}); \quad \boldsymbol{y}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_t)$$

$$\boldsymbol{h}_{t+1} = f(\boldsymbol{x}_{t+1}, \boldsymbol{h}_t); \quad \boldsymbol{y}_{t+1} = g(\boldsymbol{x}_{t+1}, \boldsymbol{h}_{t+1})$$

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}); \quad \boldsymbol{y}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_t)$$

$$\boldsymbol{h}_{t+1} = f(\boldsymbol{x}_{t+1}, \boldsymbol{h}_t); \quad \boldsymbol{y}_{t+1} = g(\boldsymbol{x}_{t+1}, \boldsymbol{h}_{t+1})$$

$$\boldsymbol{h}_{t+2} = f(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+1}); \quad \boldsymbol{y}_{t+2} = g(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+2})$$

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}); \quad \boldsymbol{y}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_t)$$

$$\boldsymbol{h}_{t+1} = f(\boldsymbol{x}_{t+1}, \boldsymbol{h}_t); \quad \boldsymbol{y}_{t+1} = g(\boldsymbol{x}_{t+1}, \boldsymbol{h}_{t+1})$$

$$\boldsymbol{h}_{t+2} = f(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+1}); \quad \boldsymbol{y}_{t+2} = g(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+2})$$

$$\vdots$$

Sequence Modeling

$$f : H \times X \times \Theta \mapsto H; \quad g : H \times X \times \Theta \mapsto Y$$

The function f is **recurrent** because it outputs a future input

$$\boldsymbol{h}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}); \quad \boldsymbol{y}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_t)$$

$$\boldsymbol{h}_{t+1} = f(\boldsymbol{x}_{t+1}, \boldsymbol{h}_t); \quad \boldsymbol{y}_{t+1} = g(\boldsymbol{x}_{t+1}, \boldsymbol{h}_{t+1})$$

$$\boldsymbol{h}_{t+2} = f(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+1}); \quad \boldsymbol{y}_{t+2} = g(\boldsymbol{x}_{t+2}, \boldsymbol{h}_{t+2})$$

$$\vdots$$

If f, g are neural networks, then we call it a **recurrent neural network** (RNN)

Sequence Modeling

The simplest recurrent neural network is called an **Elman Network**

Sequence Modeling

The simplest recurrent neural network is called an **Elman Network**

$$f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}_1^\top \boldsymbol{h}_{t-1} + \boldsymbol{\theta}_2^\top \overline{\boldsymbol{x}}_t)$$

Sequence Modeling

The simplest recurrent neural network is called an **Elman Network**

$$f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}_1^\top \boldsymbol{h}_{t-1} + \boldsymbol{\theta}_2^\top \overline{\boldsymbol{x}}_t)$$

$$g(\boldsymbol{h}_t, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma\left(\boldsymbol{\theta}_3^\top \overline{\boldsymbol{h}}_t\right)$$

Sequence Modeling

The simplest recurrent neural network is called an **Elman Network**

$$f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}_1^\top \boldsymbol{h}_{t-1} + \boldsymbol{\theta}_2^\top \overline{\boldsymbol{x}}_t)$$

$$g(\boldsymbol{h}_t, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma\left(\boldsymbol{\theta}_3^\top \overline{\boldsymbol{h}}_t\right)$$

Sequence Modeling

TODO BPTT TODO Forgetting

Sequence Modeling

Add forgetting

$$f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}_1^\top \overline{\boldsymbol{x}}_t + \boldsymbol{\theta}_2^\top \boldsymbol{h}_{t-1})$$

$$f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma\left(\boldsymbol{\theta}_3^\top \overline{\boldsymbol{x}}_t + \boldsymbol{\theta}_4^\top \boldsymbol{h}_{t-1} \odot f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta})\right)$$

Minimal gated unit (MGU)

Minimal gated unit (MGU)

$$f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma\left(\boldsymbol{\theta}_1^\top \overline{\boldsymbol{x}}_t + \boldsymbol{\theta}_2^\top \boldsymbol{h}_{t-1}\right)$$

$$f_2(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) = \sigma_{\tanh}\left(\boldsymbol{\theta}_3^\top \overline{\boldsymbol{x}}_t + \boldsymbol{\theta}_4^\top \left(f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) \odot \boldsymbol{h}_{t-1}\right)\right)$$

$$f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) =$$

$$\left(\left(1 - f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta})\right) \odot \boldsymbol{h}_{t-1} + f_{\text{forget}}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta}) \odot f_2(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t, \boldsymbol{\theta})\right)$$

Sequence Modeling

My PhD research focused on making RNNs more like human memory

https://openreview.net/forum?id=KTfAtro6vP