



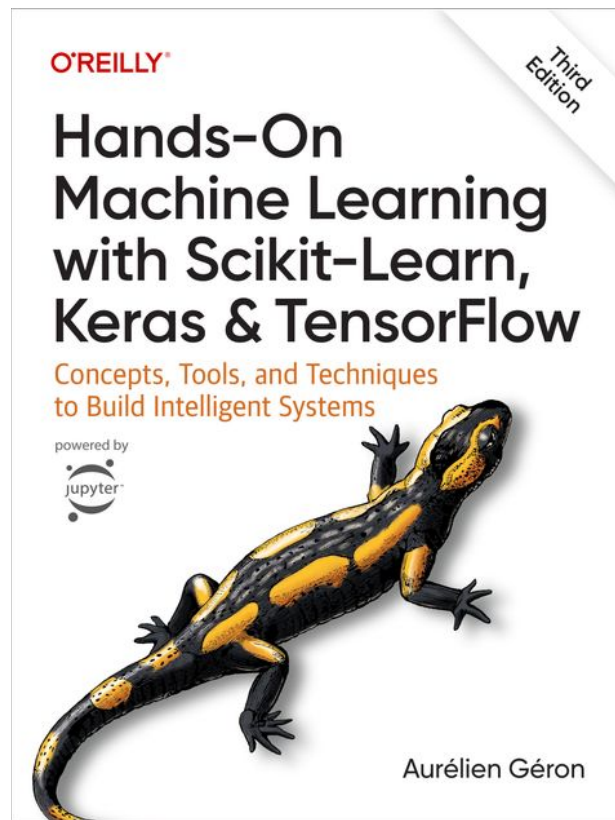
Redes Neurais Recorrentes

Inteligência Artificial - UFSM

Roberto J Mahl

01/07/2024

Bibliografia



Redes Neurais Recorrentes



- Incremento sobre Redes Feed-Forward
- Adição de recorrência nos neurônios – conexões apontando para trás
- Especialização – dados sequenciais
 - Séries temporais
 - Frases
 - Documentos
 - Áudio

*“RNNs are **not the only** types of neural networks capable of handling sequential data. For small sequences, a **regular dense network** can do the trick, and for very long sequences, such as audio samples or text, **convolutional neural networks** can actually work quite well too.”*

Utilidades



- Previsão de séries temporais:
 - Preço de ações na bolsa de valores
 - Temperatura
 - Crescimento populacional
 - Nível de poluição
- Previsão do próximo frame de um vídeo
- Processamento de linguagem natural:
 - Previsão da próxima palavra em um texto
 - Tradução automática
 - Speech-to-text
- Geração de legendas

"Para entender o final de uma frase, você precisa saber o que foi dito antes."

Neurônios e camadas recorrentes

- Representação padrão *versus* desenrolada
 - Time-step, frame
- Dois conjuntos de pesos por neurônio (w_x e w_y)
- Entrada: vetor de qualquer tamanho (x)
- Saída: também é um vetor (\hat{y})

Equation 15-1. Output of a recurrent layer for a single instance

$$\hat{y}_{(t)} = \varphi(\mathbf{W}_x^T \mathbf{x}_{(t)} + \mathbf{W}_y^T \hat{\mathbf{y}}_{(t-1)} + \mathbf{b})$$

- Função de ativação: ReLU, tanh, ...

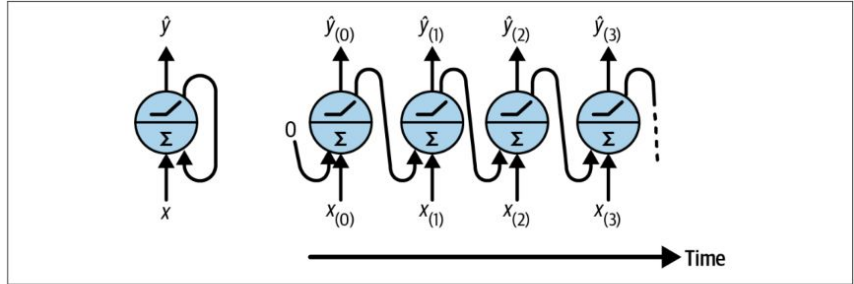


Figure 15-1. A recurrent neuron (left) unrolled through time (right)

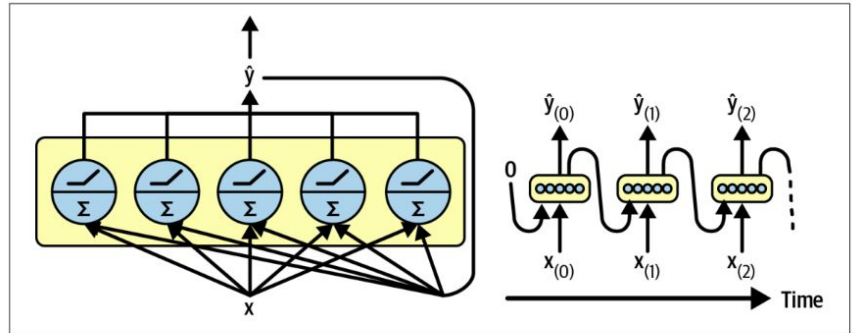


Figure 15-2. A layer of recurrent neurons (left) unrolled through time (right)

Células de memória

- Tipo de memória – preserva um tipo de estado
- A saída em um dado tempo t é função das entradas dos frames anteriores
- $h(t) = f(x(t), h(t-1))$
- LSTMs (Long Short-Term Memories) e GRUs (Gated Recurrent Networks)

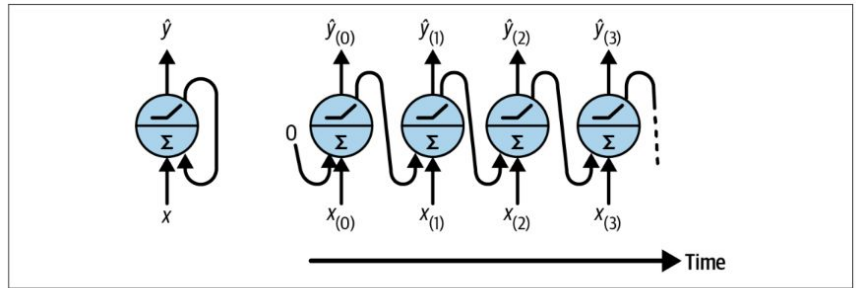


Figure 15-1. A recurrent neuron (left) unrolled through time (right)

Tipos

- **sequence-to-sequence**
 - previsão de consumo energético com base no histórico
- **sequence-to-vector**
 - definição de score com base em review
- **vector-to-sequence**
 - criação de legenda para imagem
- **encoder-decoder**
 - tradução aperfeiçoada

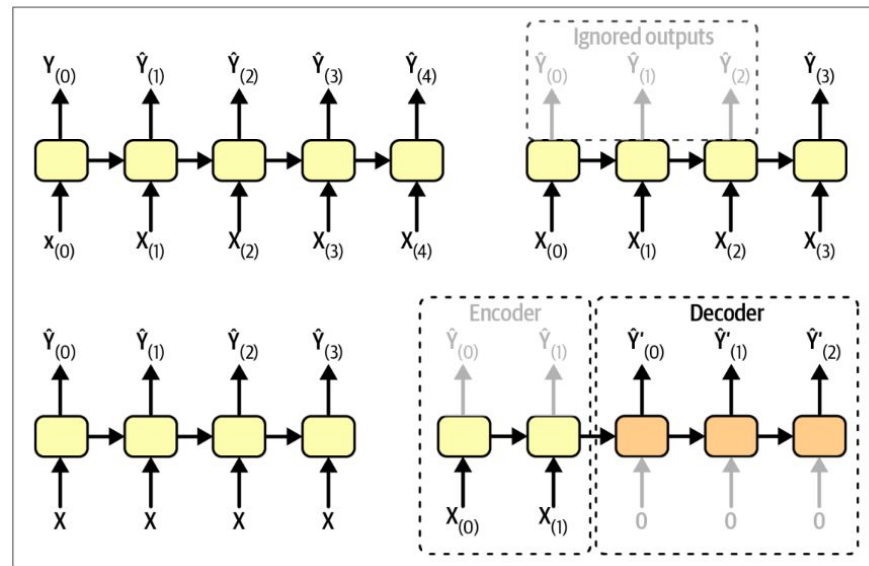


Figure 15-4. Sequence-to-sequence (top left), sequence-to-vector (top right), vector-to-sequence (bottom left), and encoder-decoder (bottom right) networks

Deep RNNs

- Obtidas ao empilharmos múltiplas camadas de células
- Pesos e parâmetros são compartilhados pelas camadas

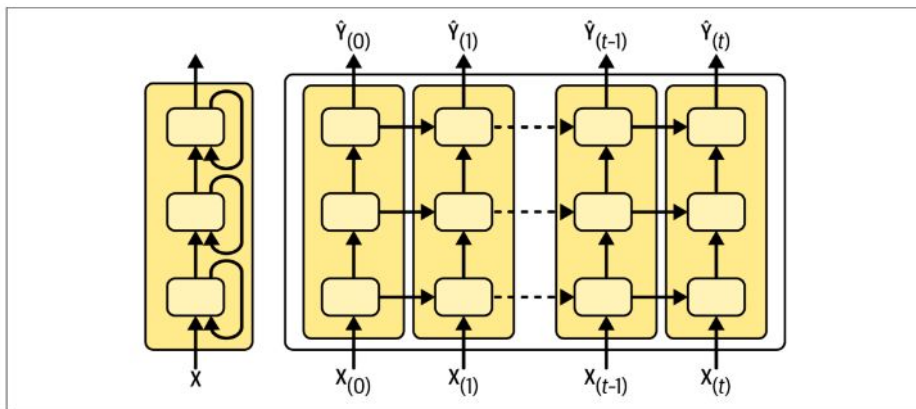


Figure 15-10. A deep RNN (left) unrolled through time (right)

```
deep_model = tf.keras.Sequential([
    tf.keras.layers.SimpleRNN(32, return_sequences=True, input_shape=[None, 1]),
    tf.keras.layers.SimpleRNN(32, return_sequences=True),
    tf.keras.layers.SimpleRNN(32),
    tf.keras.layers.Dense(1)
])
```


Treinamento - BPTT

Através de Backpropagation Through Time (BPTT)

- Decorrência da recorrência, desenrolar dos frames

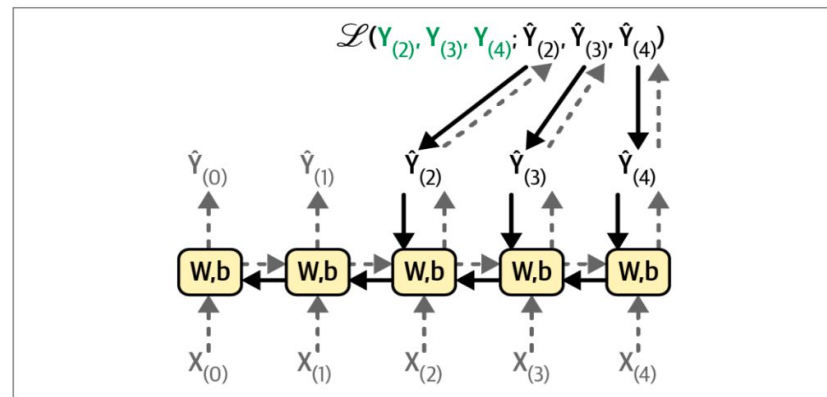


Figure 15-5. Backpropagation through time

1. Aplicação de função de cálculo de perda (*loss function*) considerando rótulos e previsões*

$$w \leftarrow w - \eta \frac{\partial L}{\partial w}$$

2. Os gradientes da perda são propagados de volta através da rede desenrolada

3. Uma vez que a backward-phase foi completada e todos os gradientes foram computados, os parâmetros podem ser atualizados usando os gradientes acumulados

* Em algumas RNNs, algumas saídas podem ser ignoradas na loss function. Por exemplo, em uma RNN sequence-to-vector apenas a última saída é considerada

** η = taxa de aprendizado, um hiperparâmetro que controla o tamanho do passo de atualização

Sequências longas – Problemas



Sequências longas \Rightarrow Muitos passos \Rightarrow Rede desenrolada mais profunda

- **Gradientes instáveis:** treinamento pode levar muito tempo ou ser instável
- **Problema de Memória de Curto Prazo:** esquecimento gradual das primeiras entradas da sequência

Gradientes instáveis

Função não-saturada: $\lim_{z \rightarrow \infty} f(z) = \infty$

Vanishing gradient ou Explosion gradient

Explosion gradient:

1. Pesos são iguais em todos os passos
2. Peso atualizado
3. Leve aumento na primeira saída
4. Leve aumento em todas as saídas subsequentes
5. Explosão da saída

Medidas para redução de risco:

- Menor taxa de aprendizado
- Aplicação de função de ativação saturada (tanh, sigmoid)

Vanishing gradient:

1. À medida em os gradientes são propagados para trás eles podem se tornar muito pequenos devido às multiplicações repetidas em cada time-step
2. Com gradientes pequenos, pesos são pouco atualizados
3. Saídas permanecem com pouca ou nenhuma alteração

Problema de Memória de Curto Prazo

Devido às transformações que os dados passam, parte da informação é perdida a cada passo da recorrência. Após algum tempo, a RNN possui quase nenhuma referência às primeiras entradas.

- Introdução de diversos tipos de células com memória de longo prazo
- Redes de Memória de Curto Prazo Longo (Long Short-Term Memory - LSTM)

- **LM task:** The writer of the books ____
is
are
- **Correct answer:** The writer of the books is planning a **sequel**
- **Syntactic recency:** The writer of the books is (correct)
- **Sequential recency:** The writer of the books are (incorrect)

Próximo passo



Exemplos