

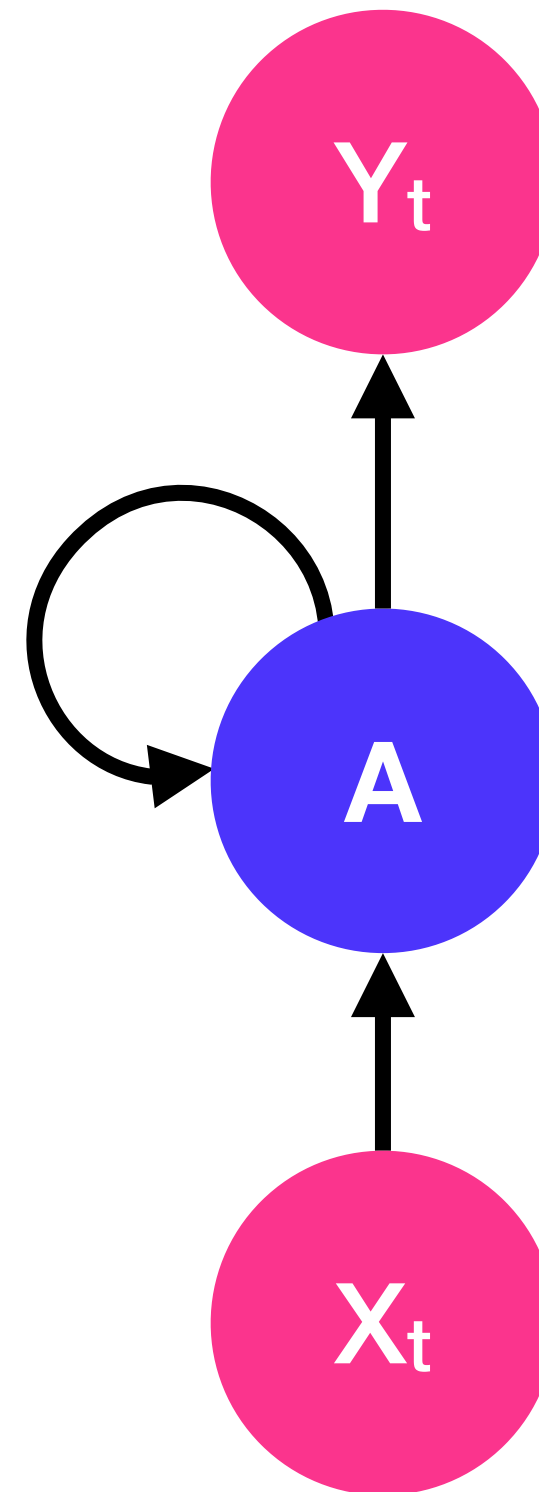
Processamento de Linguagem Natural

Redes Neurais Recorrentes

Yuri Malheiros (yuri@ci.ufpb.br)

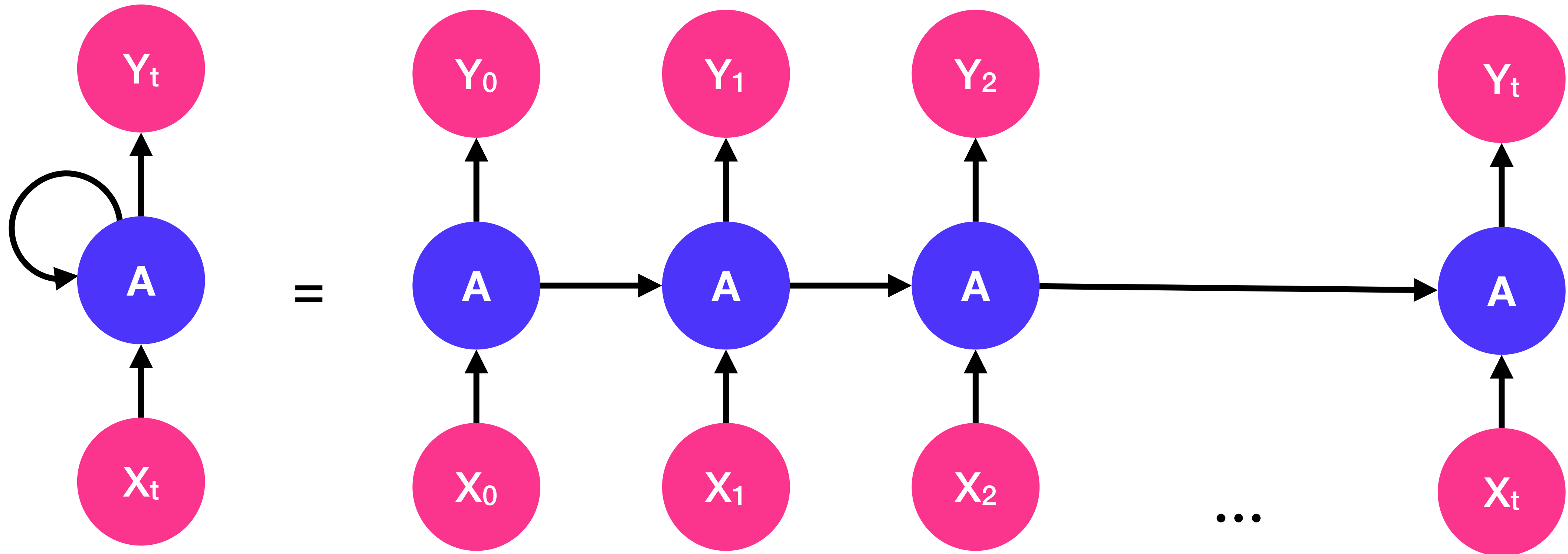
Redes Neurais Recorrentes

- Uma rede neural recorrente (RNN) é uma rede que contém ciclos
 - A saída da rede também é entrada da própria rede

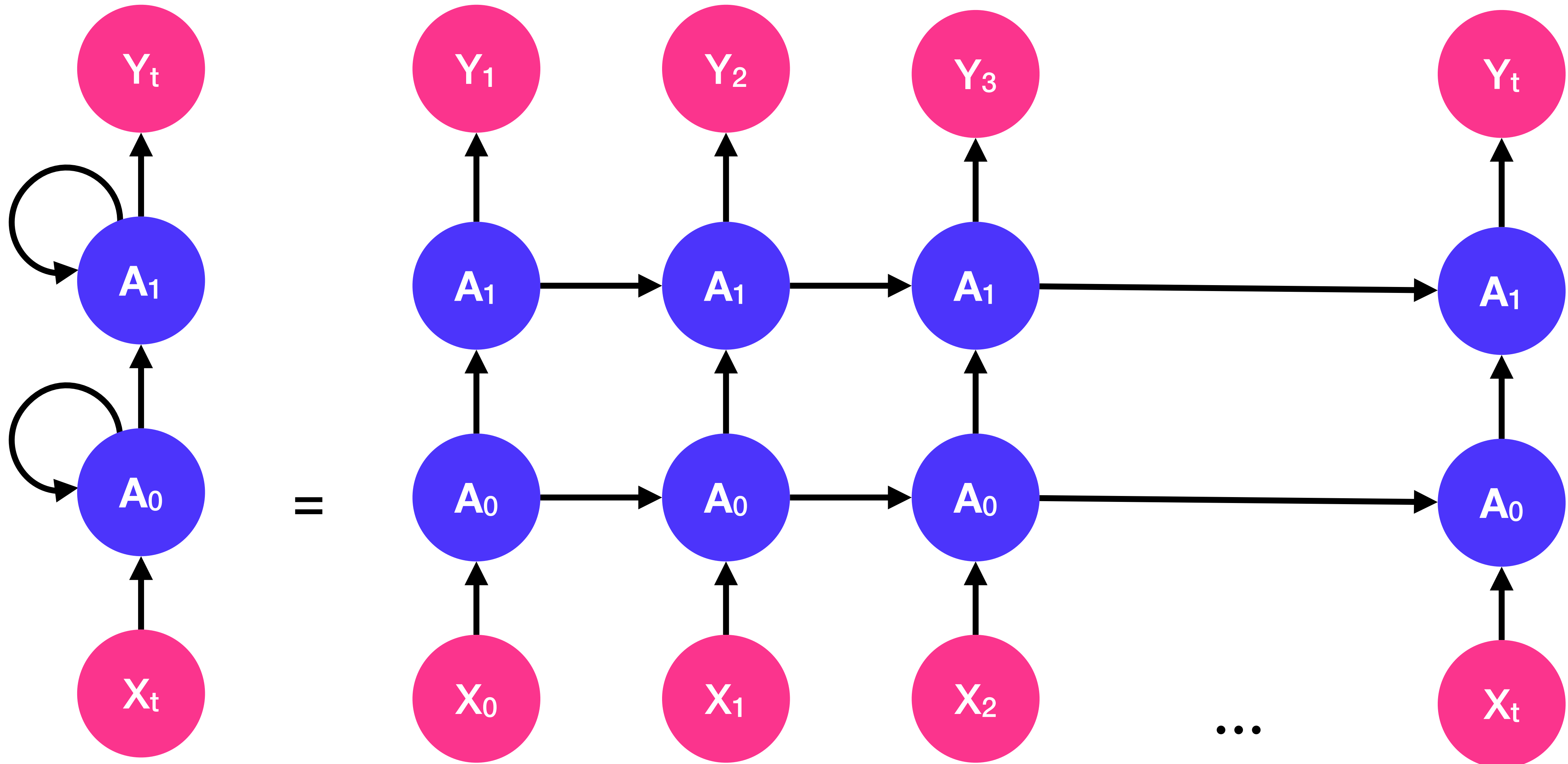


Redes Neurais Recorrentes

- O loop pode ser interpretado assim:



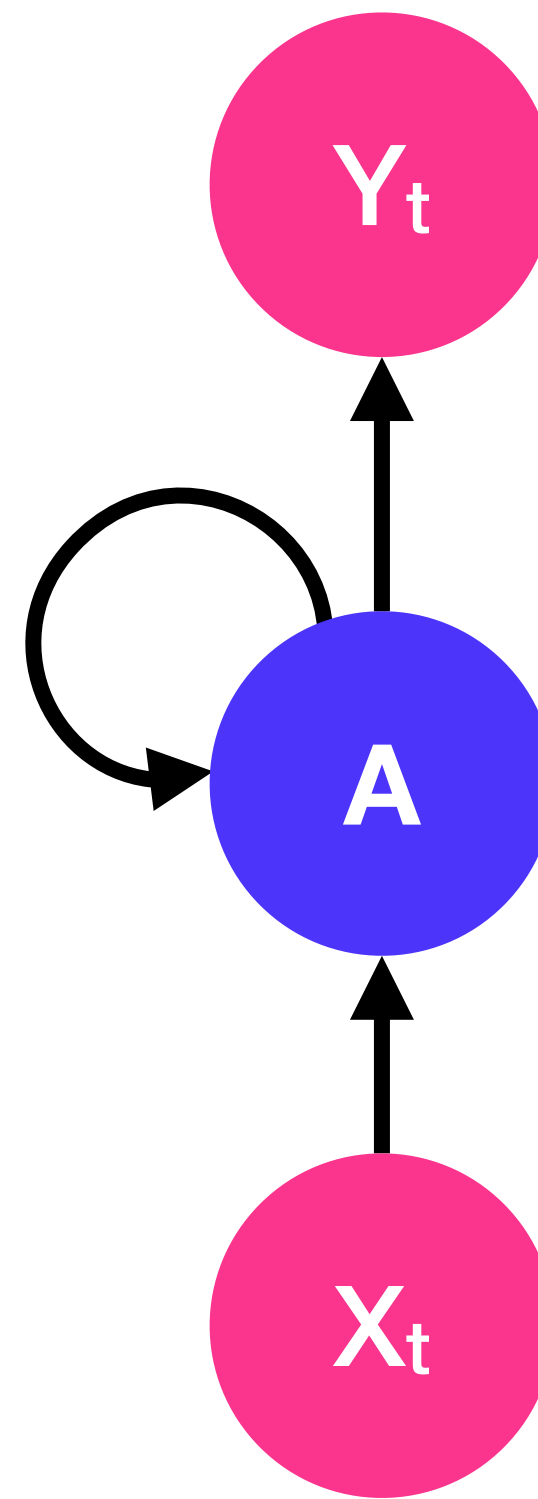
- Podemos ter múltiplas camadas recorrentes



Redes Neurais Recorrentes

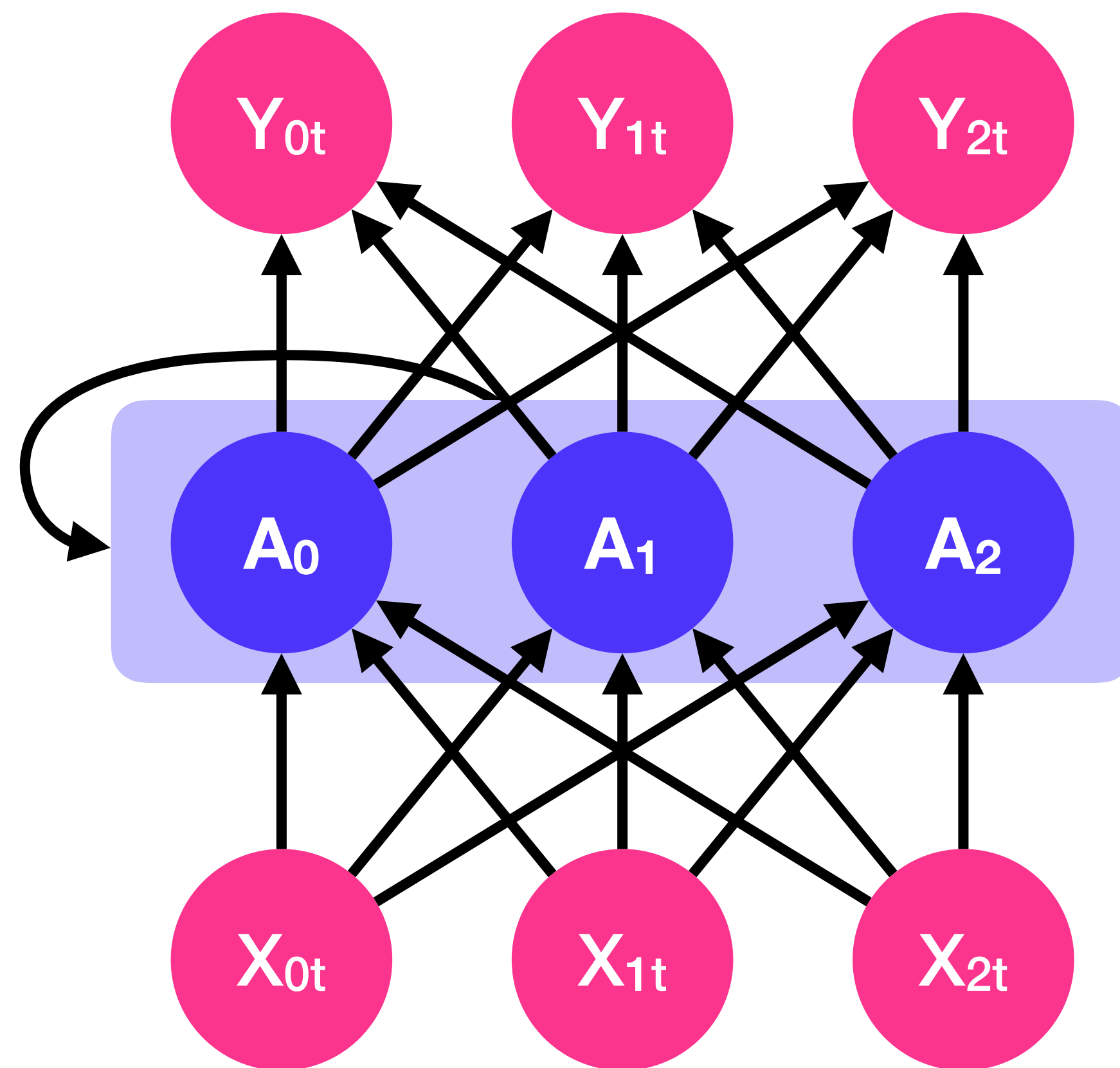
- As RNN são usadas com entradas sequenciais
 - Por exemplo: texto
- Nesse caso, os elementos da sequência são processados um por um
- A rede recebe uma entrada e o valor de sua saída na etapa anterior
- O valor da saída da rede fornece uma espécie de memória ou contexto
- Esta abordagem não coloca limite em relação ao tamanho da sequência

Redes Neurais Recorrentes

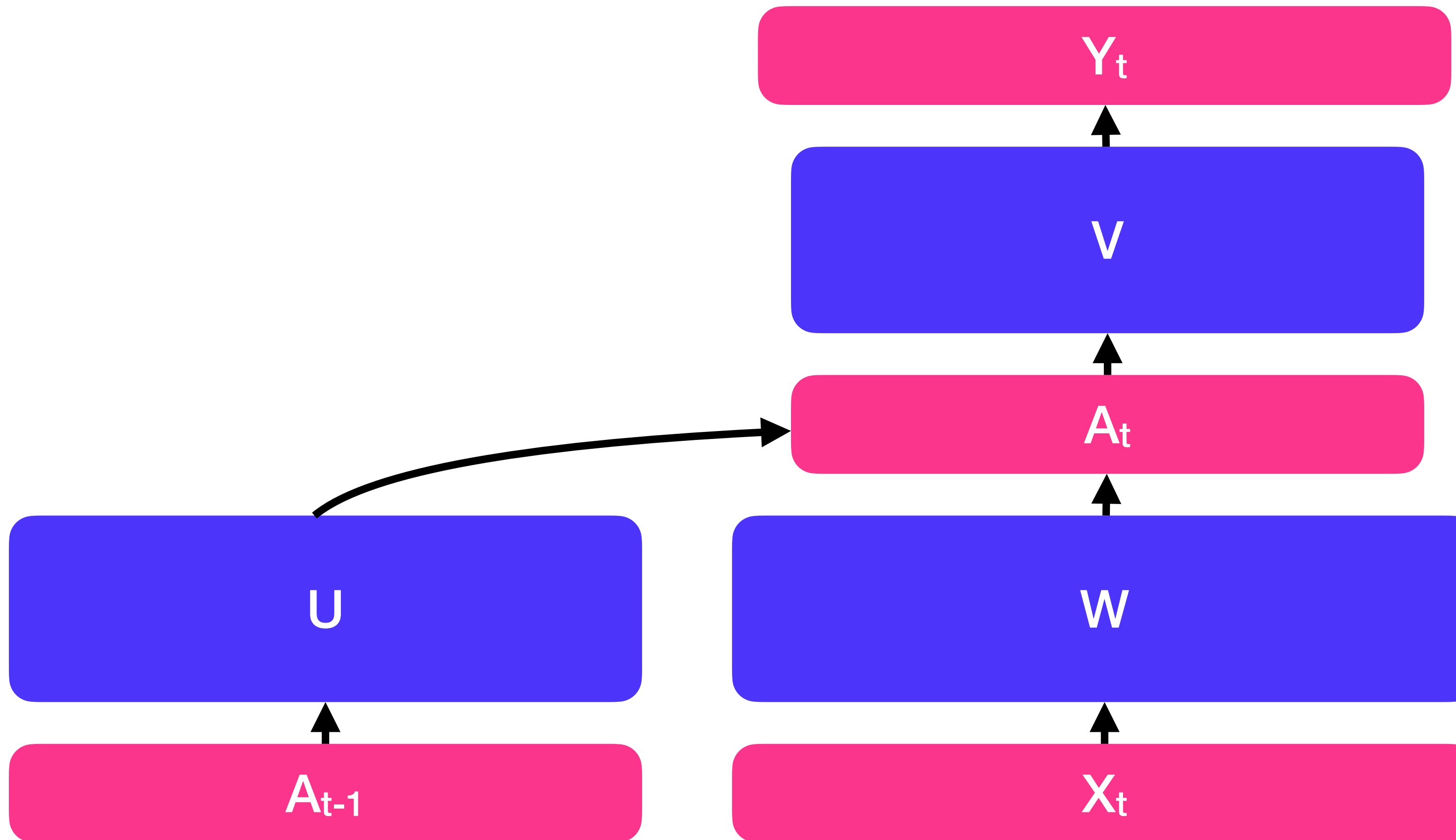


Redes Neurais Recorrentes

- Mais detalhadamente, essa é a estrutura de uma RNN



- As RNNs não são muito diferentes das redes neurais multicamadas



Redes Neurais Recorrentes

- Para computar a saída da RNN primeiro calculamos o valor de a_t
 - $a_t = g(Ua_{t-1} + Wx_t)$
 - g é uma função de ativação
- Em seguida, calculamos a saída y_t
 - $y_t = f(Va_t)$

Redes Neurais Recorrentes

- É importante analisar as dimensões das matrizes W , V e U
- W tem dimensões $d_a \times d_{entrada}$
- U tem dimensões $d_a \times d_a$
- V tem dimensões $d_{saida} \times d_a$

Redes Neurais Recorrentes

- Implementar RNN para classificação

Modelo de Linguagem Neural

- Podemos utilizar RNN para criar modelos de linguagem
- Nesse caso, a RNN processa uma sequência de palavras com o objetivo de prever a próxima palavra
- RNNs não têm o limite de contexto que modelo n-gramas possui

Modelo de Linguagem Neural

- Passando uma sequência de embeddings de palavras a saída é um vetor y representando a probabilidade das palavras de um vocabulário aparecerem depois da sequência de entrada
- $a_t = g(Ua_{t-1} + We_t)$ e_t é o embedding da palavra
- $y_t = \text{softmax}(Va_t)$

Modelo de Linguagem Neural

- A probabilidade de uma sequência é o produto de cada palavra na sequência para um terminado tempo t

- $$P(w_{1:n}) = \prod_{i=1}^n y_i[w_i]$$

- $y_i[w_i]$ é a probabilidade da palavra w_i no tempo i

Redes Neurais Recorrentes

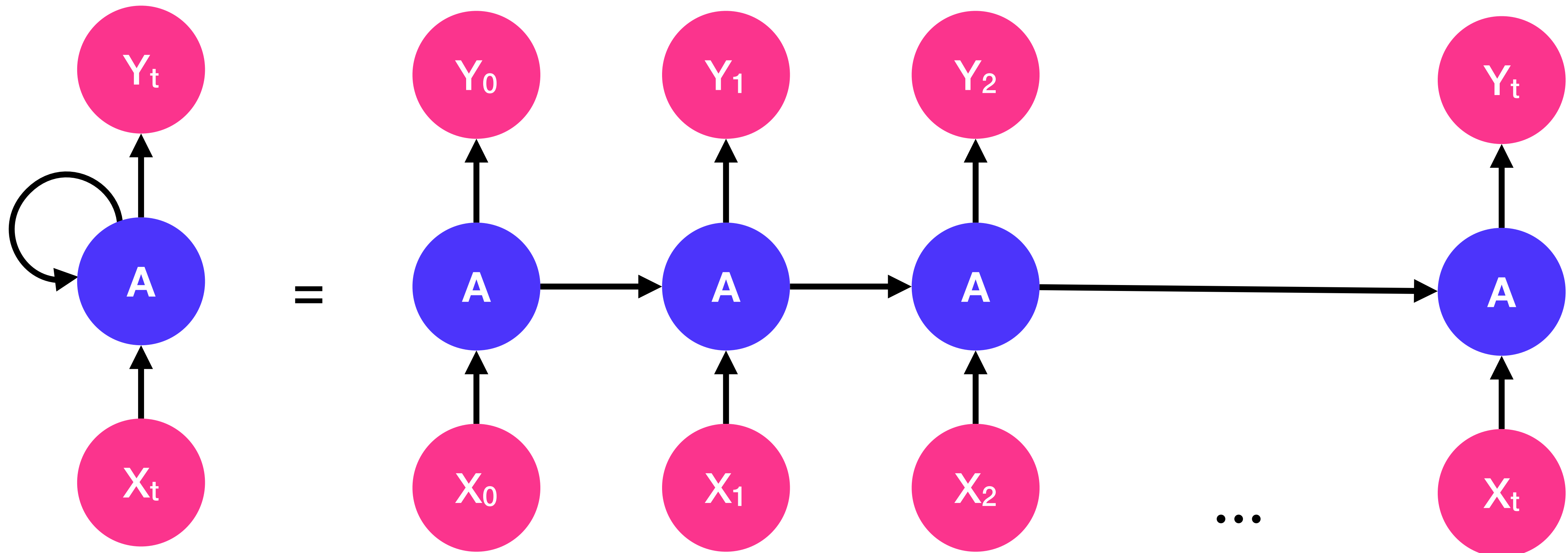
- Com as RNNs temos um modelo sequencial
 - Mas para sequências muito longas, sabemos que na prática ela pode não funcionar muito bem

LSTM

- Redes LSTM (Long Short Term Memory) são um tipo especial de RNN
- Elas possuem melhor capacidade para trabalhar com sequências grandes
- As redes LSTM possuem mecanismos para lembrar de informações por longos períodos

LSTM

- LSTM possui a mesma estrutura sequencial:



LSTM

- Entretanto, ao invés de termos apenas uma camada, nós teremos quatro que interagem de uma maneira especial

LSTM

- Continuamos tendo entradas e saídas



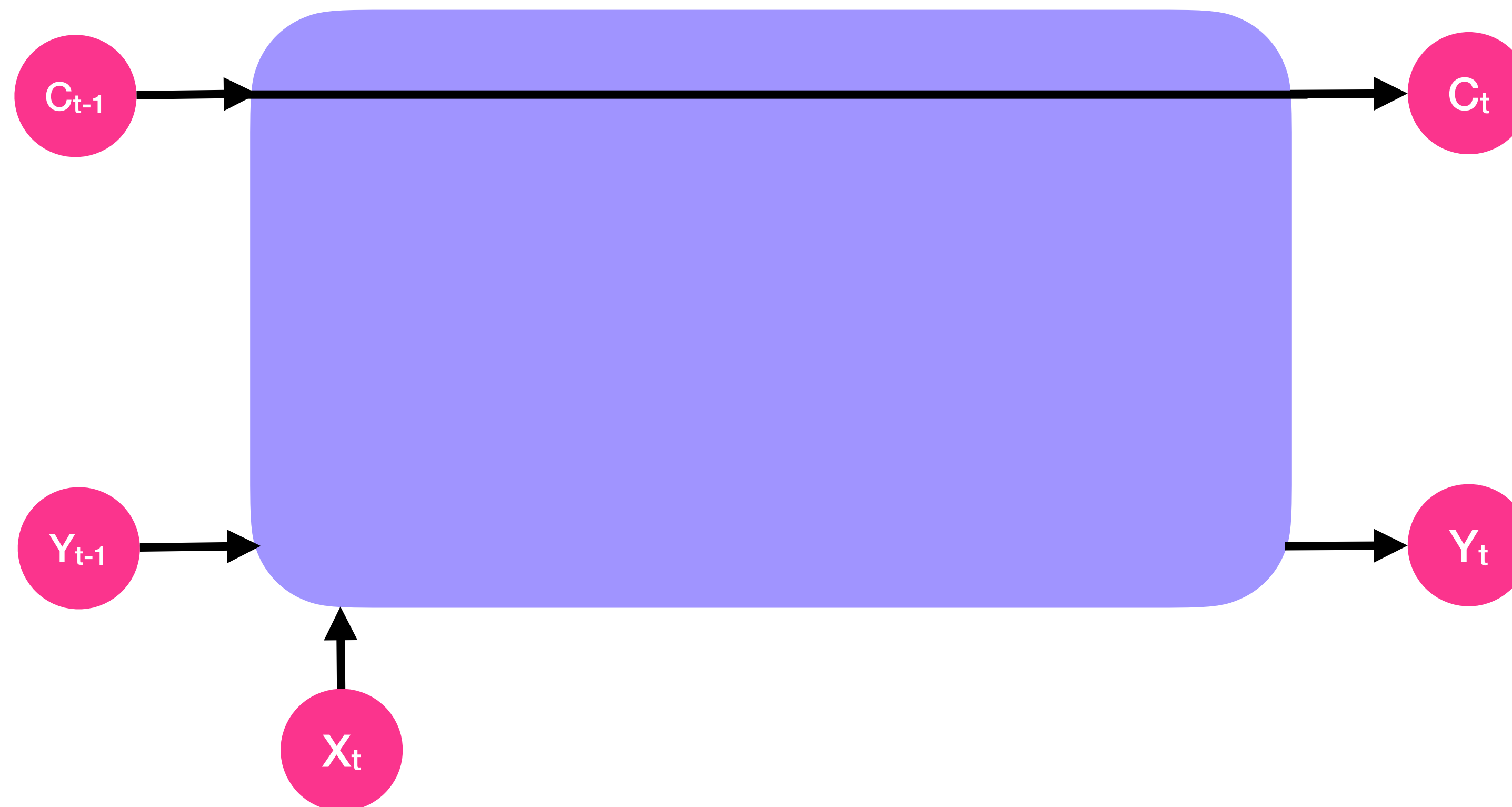
LSTM

- Além disso, temos o estado de célula, que auxilia a passagem de informações ao longo da sequência

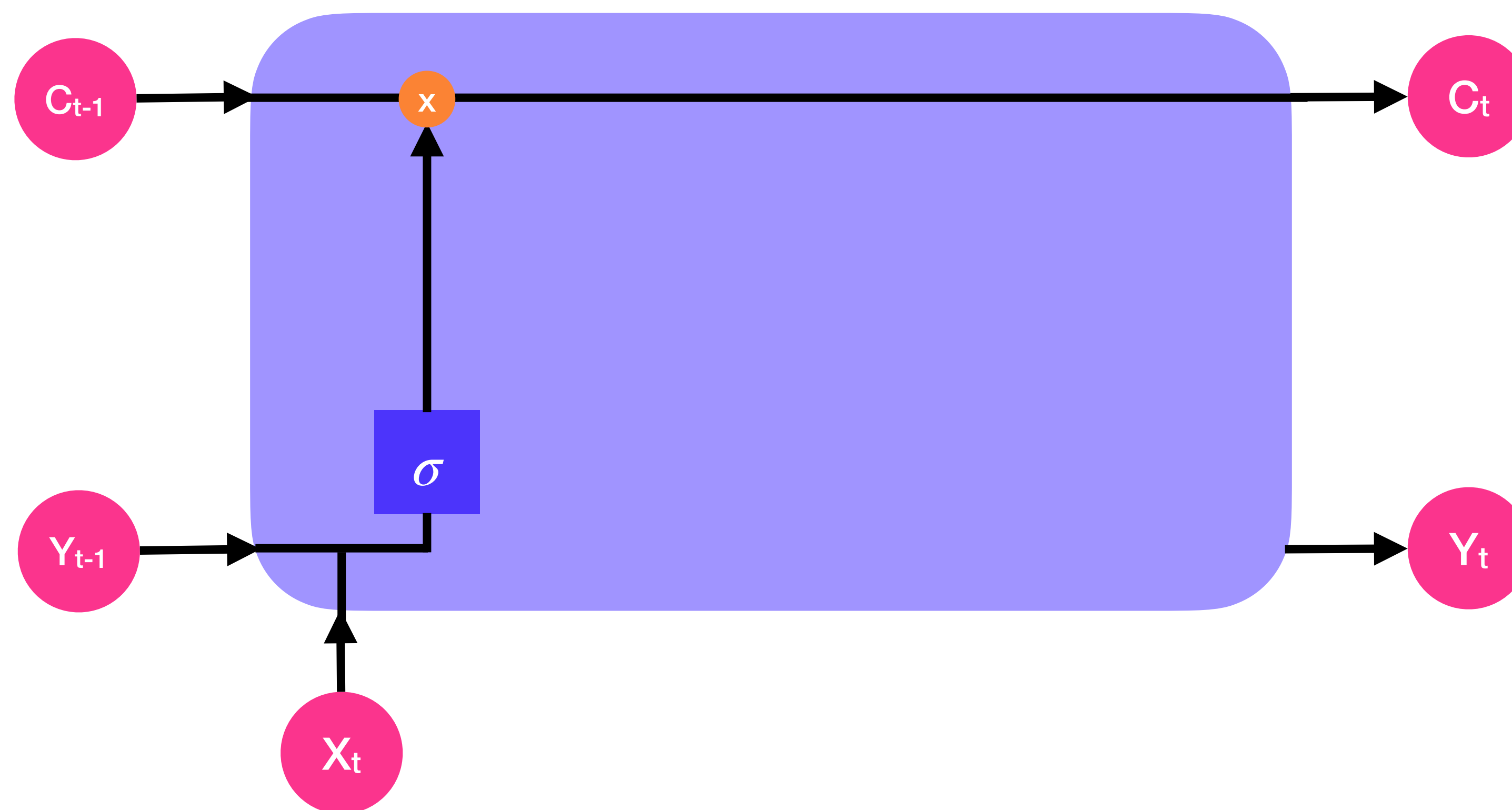


LSTM

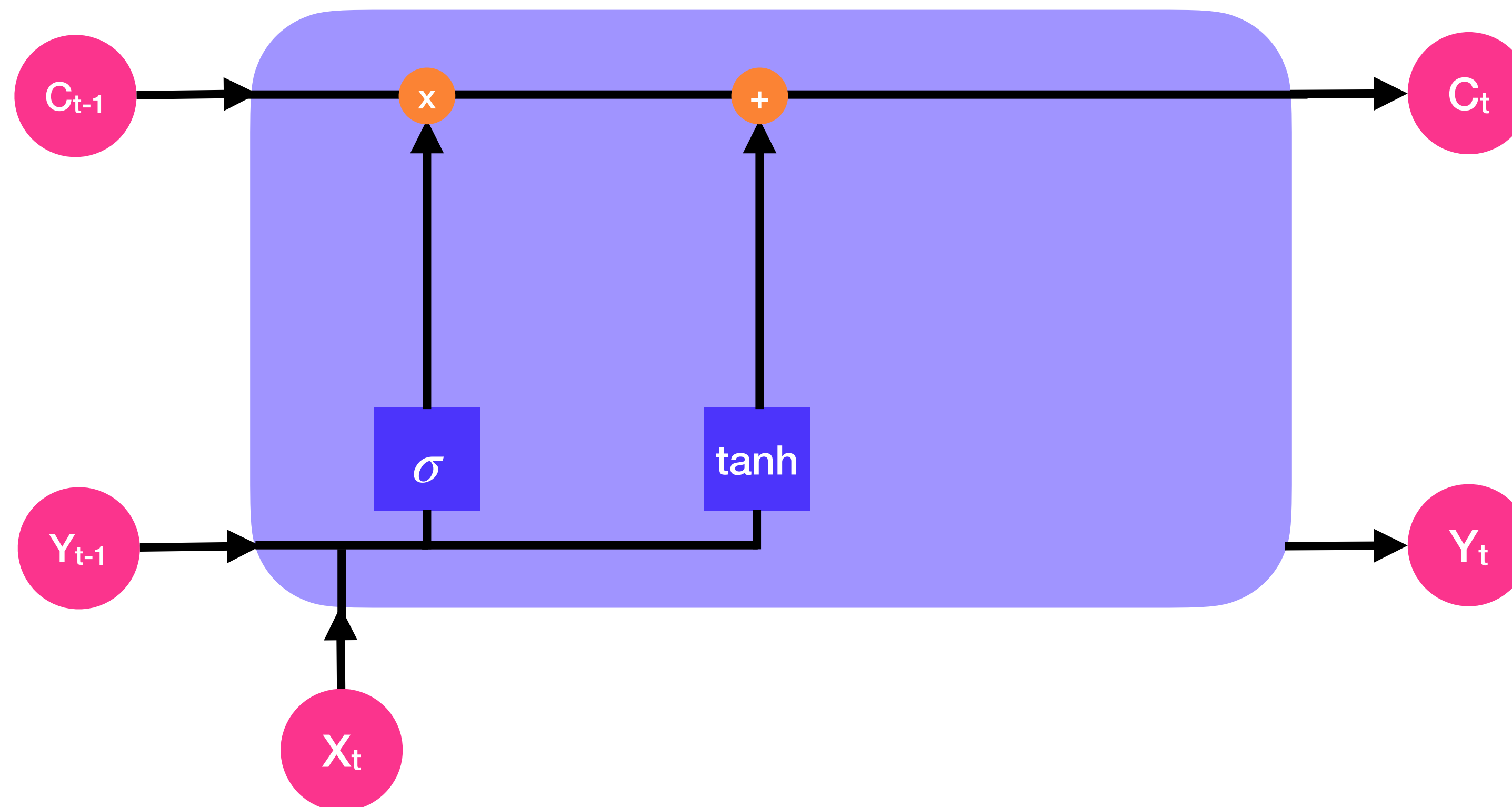
- Além disso, temos o estado de célula, que auxilia a passagem de informações ao longo da sequência



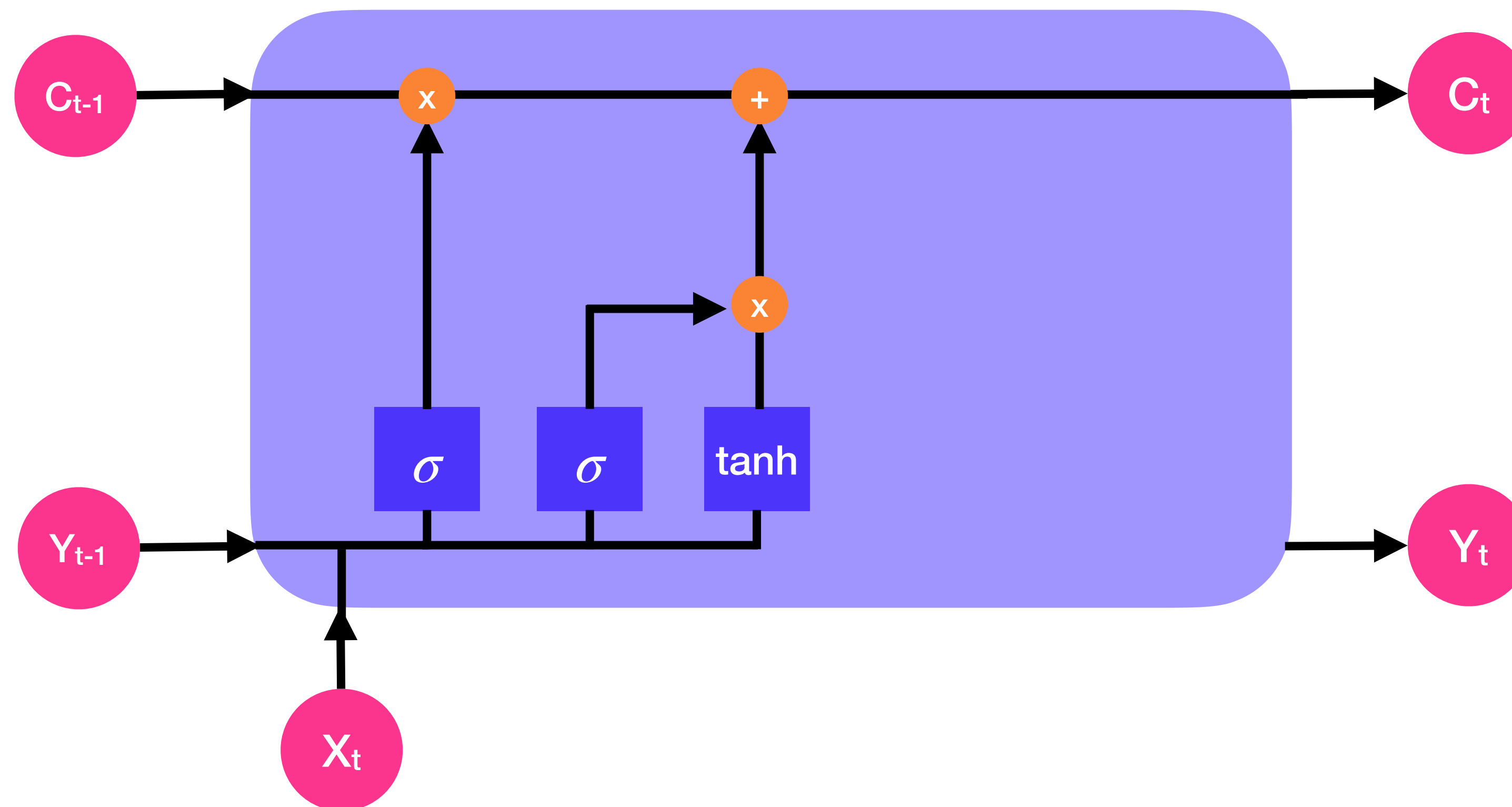
- Ao receber a entrada, o LSTM decide o quanto ele quer manter da informação do estado da célula
- A camada sigmoid é conhecida como o “forget gate”, ela retorna um número entre 0 e 1
- O valor é combinado com o estado da célula usando uma multiplicação



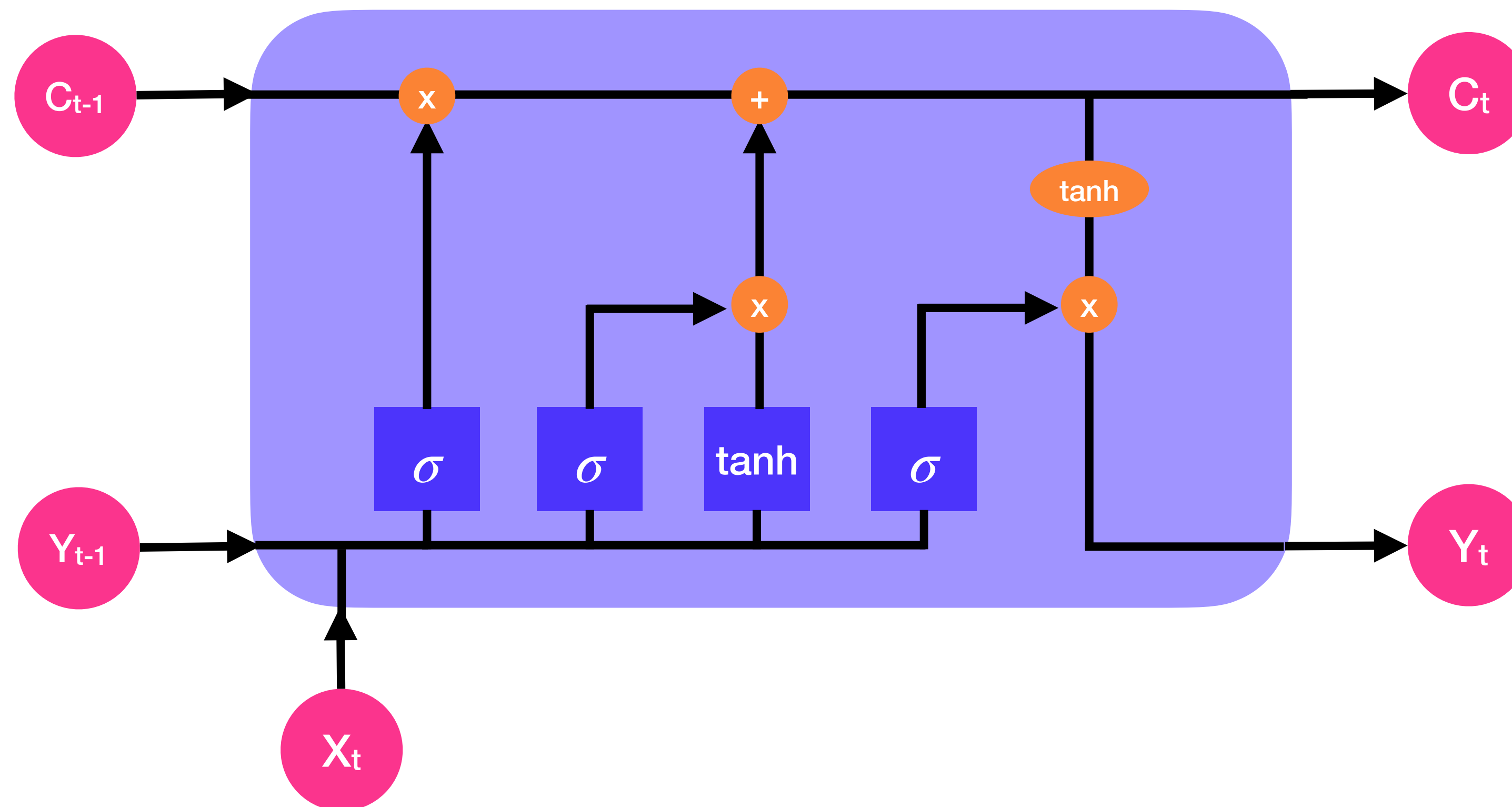
- O próximo passo é adicionar novas informações ao estado da célula
- Isto é feito através de uma camada com função de ativação tangente hiperbólica
- A saída da camada é adicionada ao estado da célula



- Antes de adicionar informação ao estado da célula, a saída de tanh é multiplicada pela saída de outra camada sigmoid
- Essa camada é conhecida como “input gate”



- Por fim, vamos definir a saída
- Ela é baseada no estado da célula, na entrada e na saída da célula anterior



Part of Speech tagging

- Podemos usar uma rede neural para atribuir um rótulo para cada elemento de uma sequência
- Por exemplo, como no part of speech tagging
- Nesse caso, usando uma rede recorrente, a entrada é o embedding de uma palavra e a saída são as probabilidades dos rótulos

Part of Speech tagging

