

## Empacotamento de sequências

Para processar batches de sequências com tamanhos variáveis, aprendemos o artifício de empacotamento de sequências. Vimos que essa abordagem exige três passos:

- Ordenar as sequências dentro de um batch da maior para a menor,
- Adicionar elementos vazios (padding), igualando o comprimento de todas as sequências para em seguida transformá-las em um único tensor `X`.
- Guardar o comprimento original de cada sequência em uma tensor de tamanhos .

Com essas informações em mãos, podemos executar a seguinte sequência de passos na definição do forward da rede.

```
## Empacote a sequência antes de alimentar a unidade recorrente
packed_input = nn.utils.rnn.pack_padded_sequence(X, tamanhos)
## Forward recorrente
packed_output, hidden = self.rnn(packed_input, hidden )
## Desempacote a sequência para continuar o fluxo na rede.
output, output_lengths = nn.utils.rnn.pad_packed_sequence(packed_output)
```

[COPIAR CÓDIGO](#)

Sobre empacotamento de sequências, e considerando a formulação acima, marque as alternativas que apresentam vantagens do empacotamento de sequências.

Selezione 2 alternativas

**A**

Permite processar um batch de tamanho maior do que 1 com sequências de tamanhos variáveis.

**B**

A RNN processa somente a parte relevante do pacote, ignorando o padding.

**C**

Viabiliza o uso de RNNs multicamadas, mais eficientes que as camadas de célula (ex.: RNNCell ).