```
def simple_elementwise_apply(fn, packed_sequence):
    """applies a pointwise function fn to each element in packed_sequence"""
    return torch.nn.utils.rnn.PackedSequence(fn(packed_sequence.data), packed_sequence.batch_sizes)
```

→ Modèle séquence à séquence $x_1, ..., x_N \longrightarrow y_1, ..., y_M$ $(N = M)$

utilisation → $(N = M)$ tagging   je vais en TP

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ S V CC CC

$\qquad\qquad$ ↘ $(N \neq M)$ traduction   un chat → a cat

- Packed Sequence (optionnel)

$\qquad\qquad\qquad\qquad\qquad$ A   B   C
$\qquad\qquad\qquad\qquad\qquad$ D   E   → Packed_sequence → data   ADF|BE|C
$\qquad\qquad\qquad\qquad\qquad$ F   ↑   batch_sizes 3, 2, 1
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ torch.nn.utils.rnn.pack_sequence

- [OOV] = out of vocabulary → apprentissage = remplacer un token de temps
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ en temps par oov

- [SOS] = start of sequence

$\qquad\qquad\qquad\qquad\qquad\qquad$ un chat : [SOS] → a → cat → [EOS]

$\qquad\qquad\qquad\qquad h_0 \to h_1 \to h_2 \dashrightarrow \quad h_0 \longrightarrow h_1 \longrightarrow h_2 \cdots$
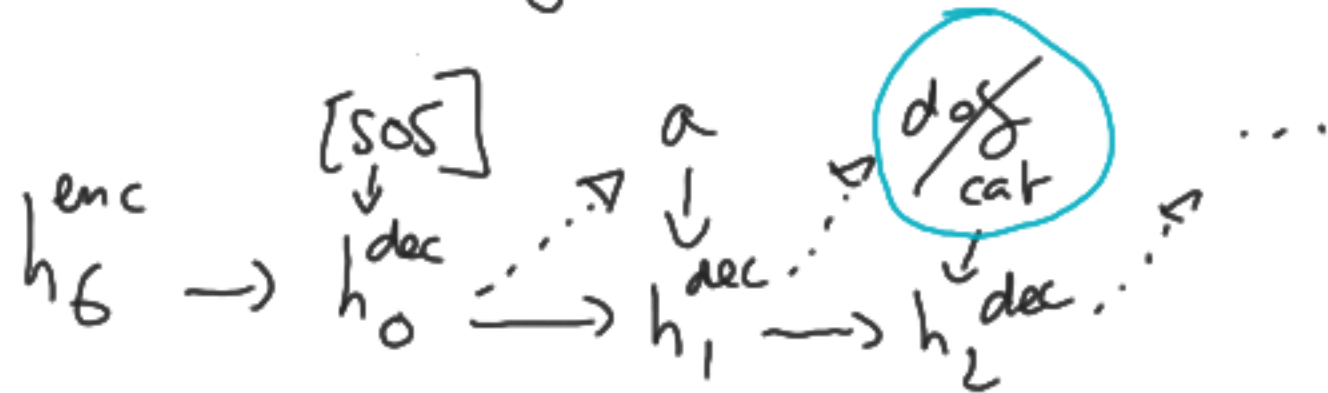
$\qquad\qquad\qquad\qquad\qquad\qquad$ RNN-enc $\qquad\qquad$ RNN-dec

"Curriculum learning"

Un chat saute sur le toit ($\rightarrow h_6^{enc}$)

A cat jumps on the roof (cible)

$$h_6^{enc} \rightarrow h_0^{dec} \xrightarrow{} h_1^{dec} \xrightarrow{} h_2^{dec} \dots$$

[SOS] $\rightarrow$ a $\rightarrow$ dog/cat $\dots$

En inférence   A dog ....

$\hookrightarrow$ à calculer h_2^{dec}

Byte-pair encoding (BPE)

"You | shoulda | got | David | Carr | of | Third | Day | to | do | it"

mang | er
    | é
    eait
    eais

....