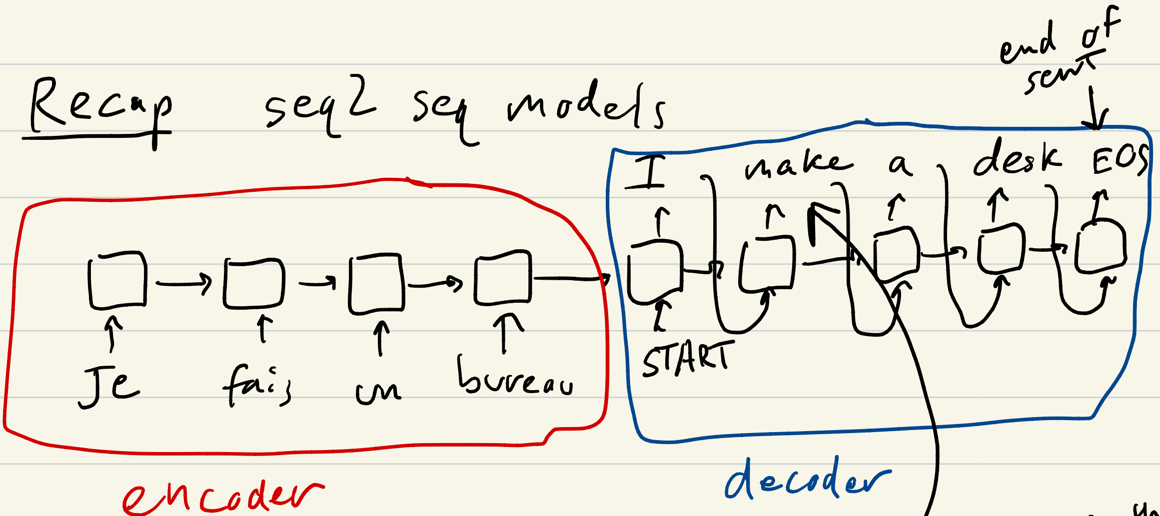
# CS 378 Lecture 20

**Today**
- Attention in seq2seq models
- Neural machine translation

**Recap** seq2seq models




$$P(t_i \mid \bar{s}, t_1, \ldots, t_{i-1})$$

update both enc + dec

**Training** Assume everything correct through $i-1$, maximize log prob of word $t_i$

teacher forcing

$$\text{loss} = \sum -\log P(t_i^* \mid \bar{s}, t_1^*, \ldots, t_{i-1}^*)$$

## Announcements

- A4 due
- A5 out tonight (due in 1 week)

## Problems with seq2seq models

① Model repeats itself in a loop

Je fais ... ⇒ I make a desk a desk
a desk a desk ...

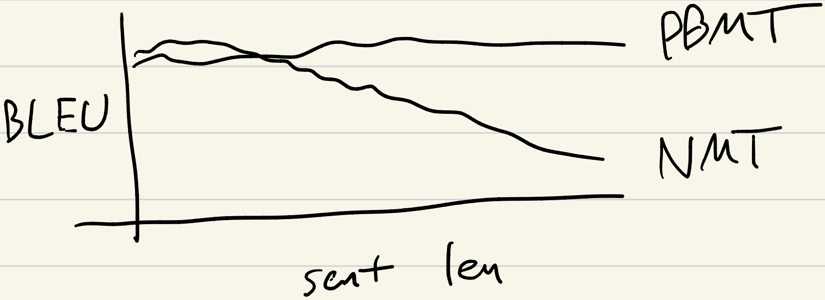
Why didn't this happen in phrase-based?
We had a notion of coverage!

RNN doesn't track "progress"

② Fixed vocabulary

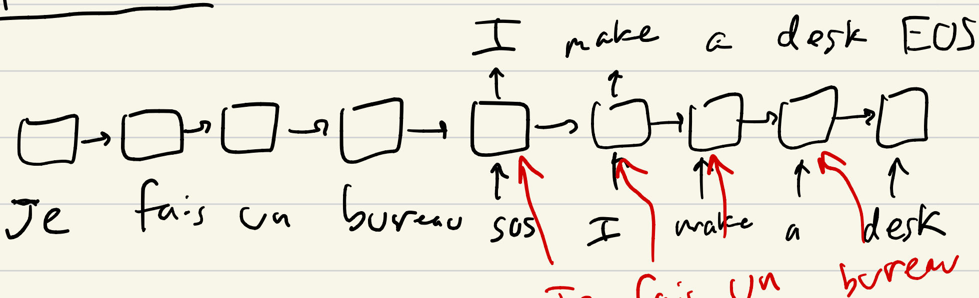
Elle est allée à Pont-de-Buis ⇒ She went to
UNK

## ③ Bad at long sentences




LSTMs have fixed hidden state, 50+ timesteps is hard

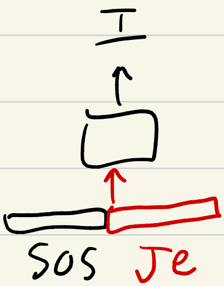
## Attention




Requires modifying LSTM to take 2 inputs

Suppose it was always a word-by-word translation in order

Decoder:



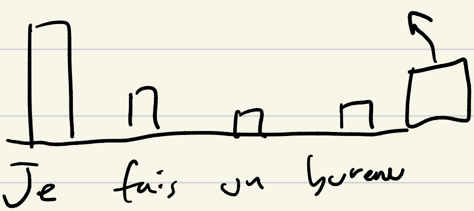

RNN just needs to map Je to I

Can learn this more easily + with fewer params than a normal seq2seq.

(Could even delete encoder)

This is too rigid. Instead we want the decoder to <u>softly</u> pick where it looks in the input.

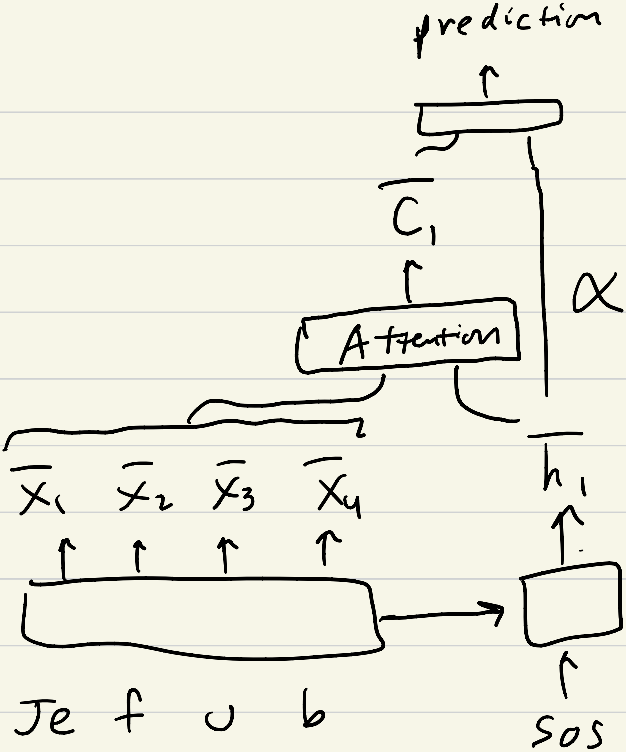
Attention:




distribution over input positions

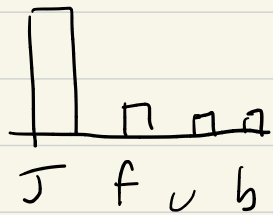
↳ weighted average of input

↳ used for prediction




"loop over all inputs i, compute this score for $\bar{x}_i$, softmax"

$$\alpha = \text{softmax}_i\left(\bar{h}_1^T W \bar{x}_i\right)$$




form $\bar{c}_1 = \sum_{i=1}^{n} \alpha_i \bar{x}_i$

$$P(t_1 \mid \bar{s}) = \underset{\text{(vocab)}}{\text{softmax}}\left(V \begin{bmatrix} \bar{h}_1 \\ \bar{c}_1 \end{bmatrix}\right)$$

vector concat

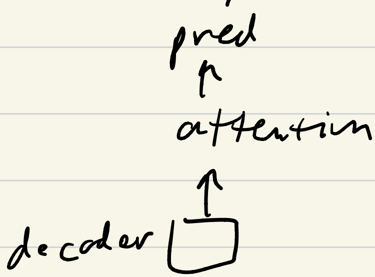
$\bar{c}_1 \approx$ "Je"

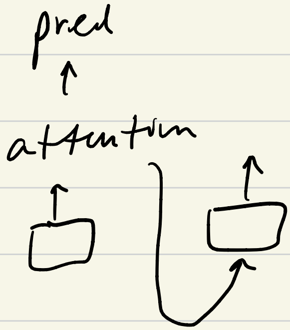
Do this at every timestep!

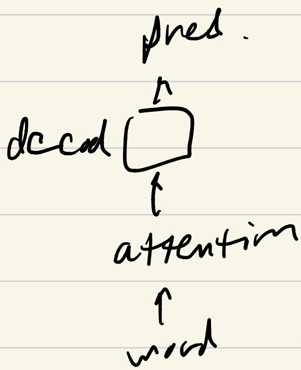
Training: with backprop

# Details:

- Many ways to set this up




$$\alpha = \text{softmax}(f(\bar{h}_1, \bar{x}_i))$$

f: dot product,

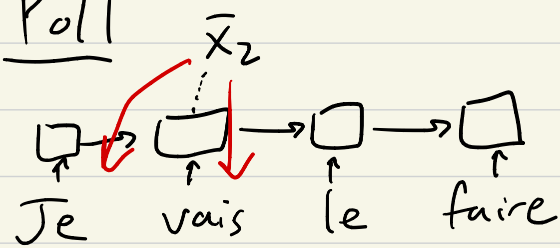
$\bar{h}_1^T W \bar{x}_i$

one-layer NN

$\vdots$

Overall idea: form context vector $\bar{c}$ that captures input directly

# Poll




? look back at the verb
look back at position 2 in the (index) source

I am going to do it

1. am : look at vais
2. look at Je (subject)
   look at other stuff?
   (other dependencies of the verb)
3. Encoder tracks : <u>content</u>

   <u>position</u>

Decoder timestep 1 : model wants to attend to enc. timestep 1