Natural Language Processing

Lecture 3.2

Sequence labelling with local search

Marco Kuhlmann
Department of Computer and Information Science

Approaches to sequence labelling

- **Local search**

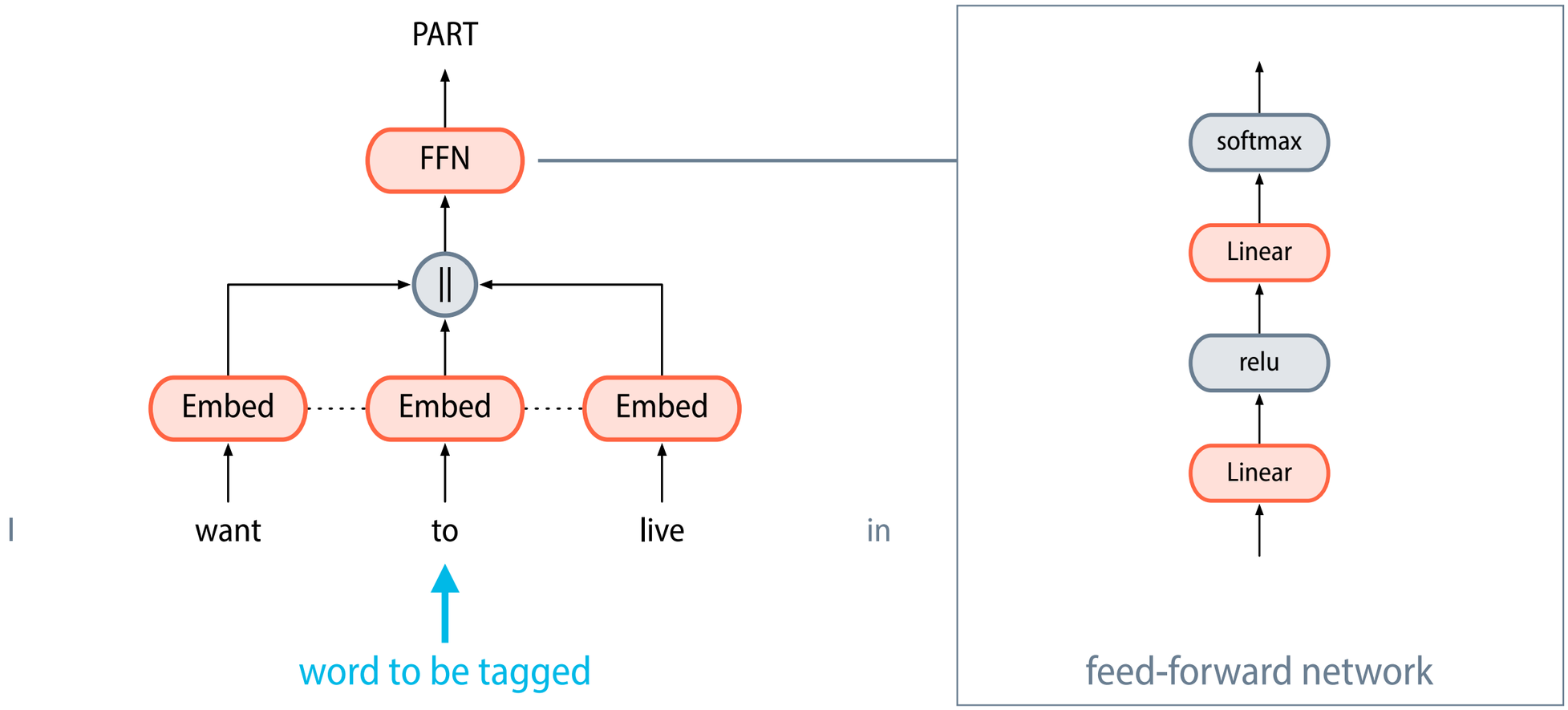
  Cast sequence labelling as a sequence of classification problems: predict the label for each position in the input sequence.

- **Global search**

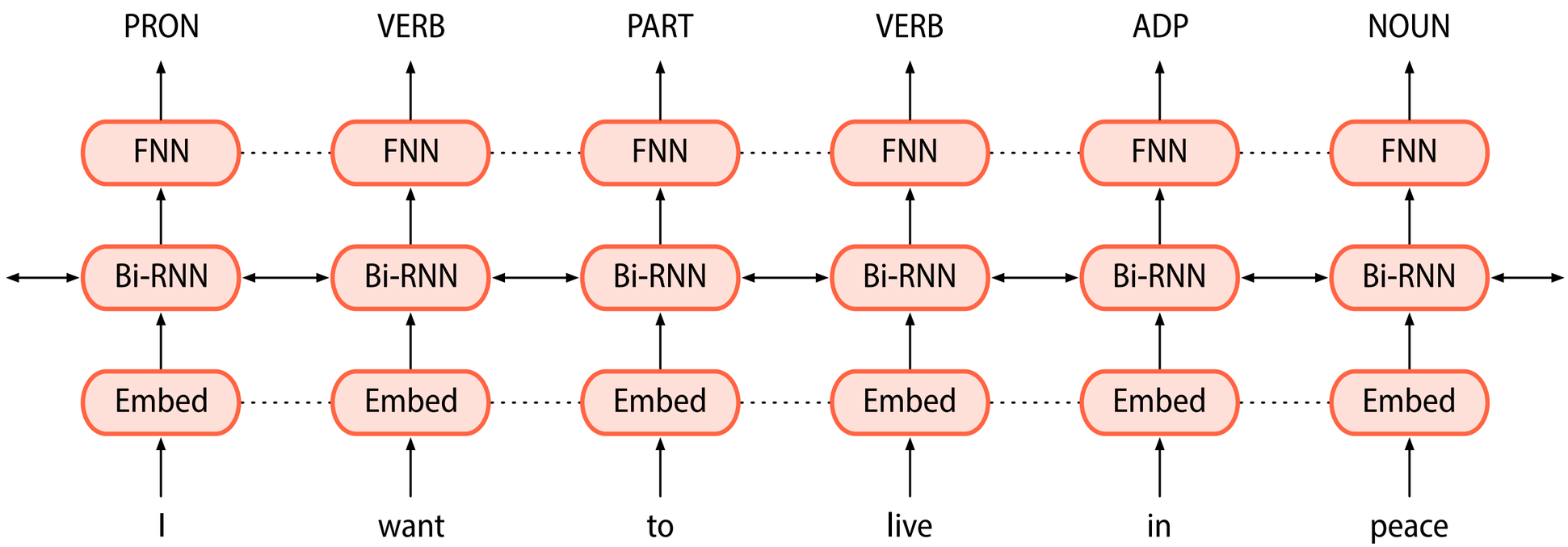
  Cast sequence labelling as a combinatorial optimisation problem over the full set of candidate sequences.

  requires specialised algorithms

Fixed-window model



Bidirectional RNN model



loss = sum or mean of the token-specific losses

Practical issues when training the Bi-RNN model

- The output of the Bi-RNN model is a 3-dimensional tensor of shape [# sentences, sequence length, # labels].
- The sequence length needs to be uniform, so we have to use padding to bring all sequences in a batch to the same length.
- We do not want to compute the loss for the padding tokens, so we need to mask those tokens or use padding-aware losses.

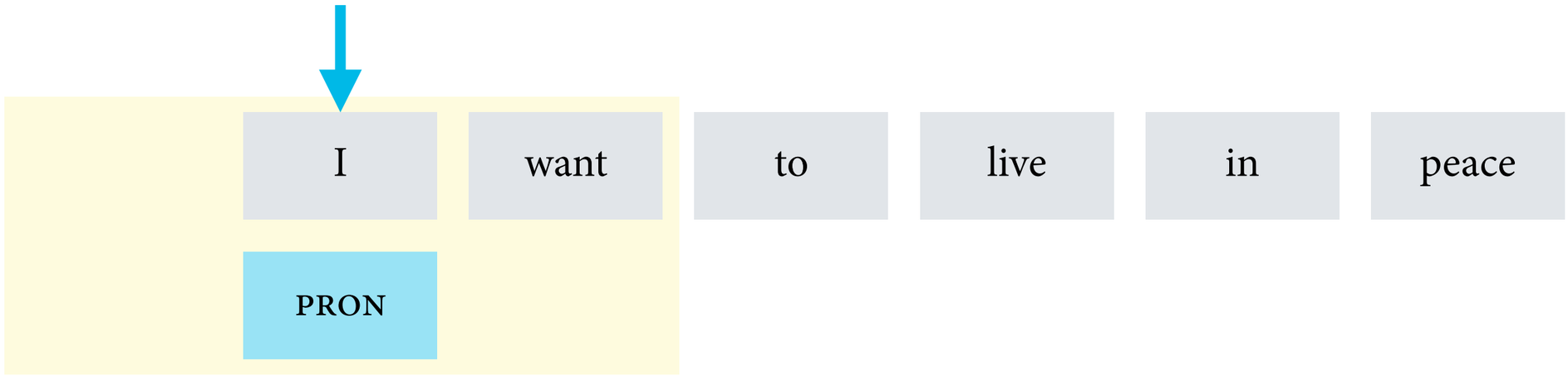
  Loss functions in PyTorch have the 'ignore_index' keyword.

Labels are interdependent

I	want	to	live	in	peace	
PRON	VERB	PART	VERB	ADP	NOUN	71%
PRON	VERB	ADP	VERB	ADP	NOUN	1%

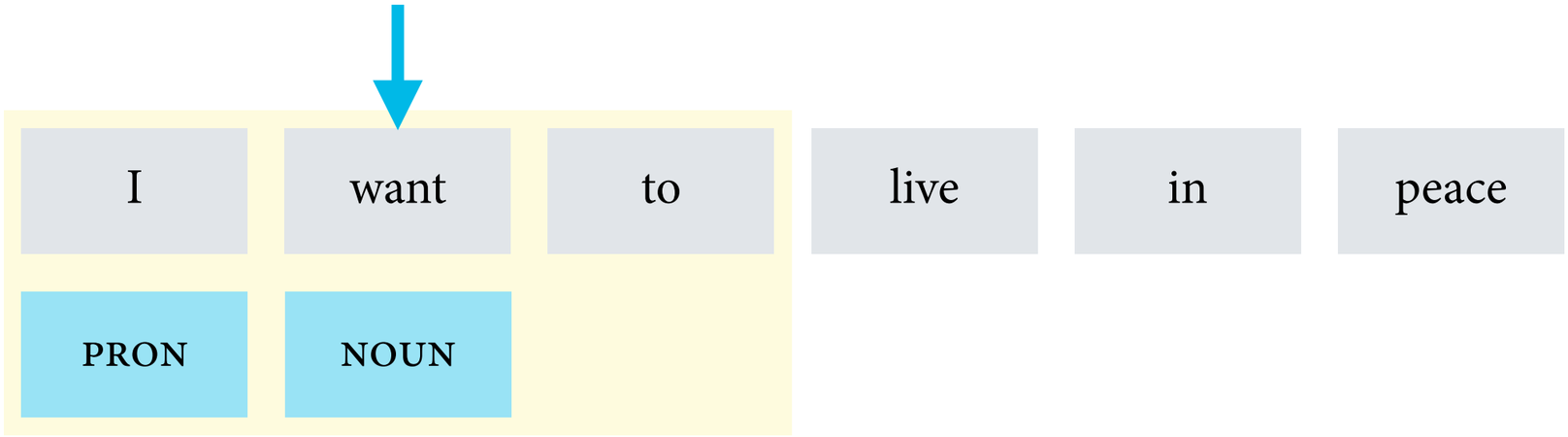
Some combinations of part-of-speech tags
are more likely than others.

Autoregressive tagging with a fixed-window model



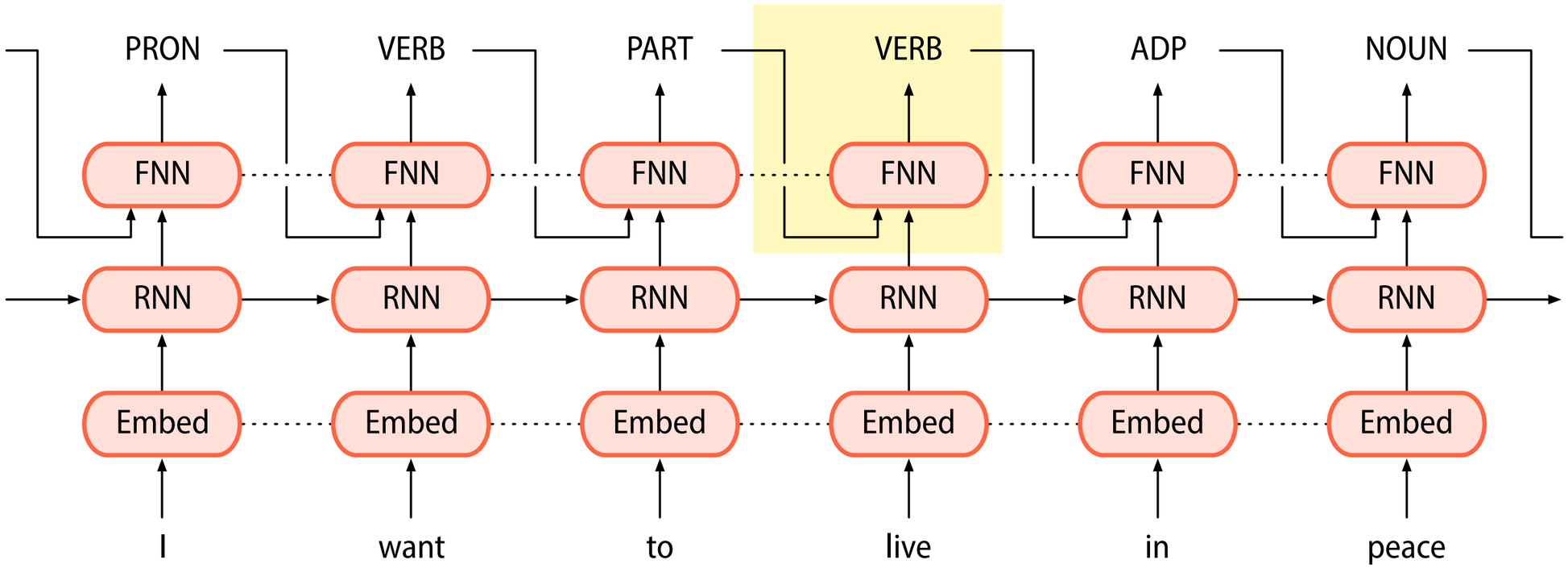
The model predicts the tag for the first word in the sentence.
Features are extracted from a context window.

Autoregressive tagging with a fixed-window model



The prediction at the second position can use features defined over the tags that have already been predicted.

Autoregressive RNN model



The label for the previous position becomes an extra input.

Training autoregressive models

- At test time, we run the model incrementally, and feed it with its own predicted labels.
- At training time, the label that we feed as an additional input is the gold-standard label. This regime is called **teacher forcing**.
- Teacher forcing can be problematic, because the model does not learn to deal with its own prediction errors.

  difference between training time and prediction time (exposure bias)