Deep Learning for Natural Language Processing

Basic Models for Sequence Labeling

UNIVERSITY OF
GOTHENBURG

CHALMERS

WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

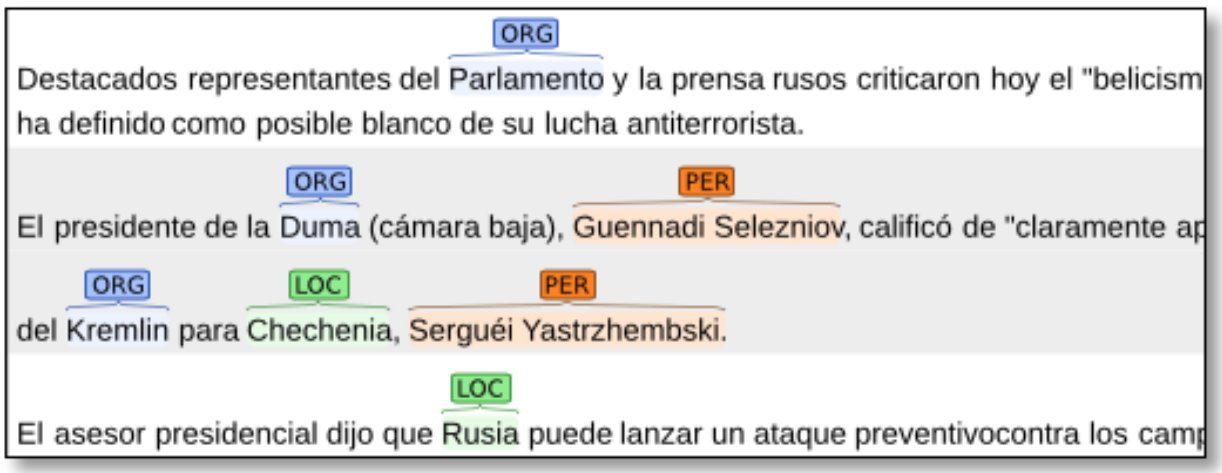
Richard Johansson

richard.johansson@gu.se

recap: main type of sequence prediction tasks

- **sequence labeling** tasks
- **segmentation** tasks
- **bracketing** tasks

Astronomers	announce	the	detection	of	water	in	the	atmosphere
NNS	VBP	DT	NN	IN	NN	IN	DT	NN



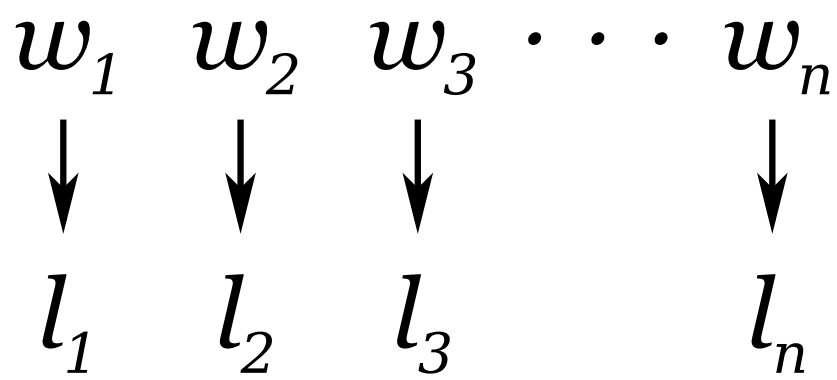
recap: BIO coding

- for segmentation and bracketing tasks, we often convert into sequence labeling by applying BIO coding or similar

The	**cases**	**of**	**metastatic**	**cancer**	**of**	**the**	**gall**	**bladder**
			Pathology				Organ	
O	**O**	**O**	**B-PAT**	**I-PAT**	**O**	**O**	**B-ORG**	**I-ORG**

structured prediction: basic terminology

- sequence labeling is a **structured prediction** task
  - the output is a complex object, not just a category
  - later in the course: trees, graphs, ...
- input: a sequence $\boldsymbol{x}$
- output: a sequence $\boldsymbol{y}$ of the same length as $\boldsymbol{x}$



Algorithmic approaches

- **Exhaustive search**

  Cast structured prediction as a combinatorial optimisation problem over the set of target representations.

  Viterbi algorithm, Eisner algorithm

- **Greedy search**

  Cast structured prediction as a sequence of classification problems: at each point in time, predict one of several options.

  window-based part-of-speech tagging, arc-standard algorithm

Algorithmic approaches

- **Exhaustive search**

  Cast structured prediction as a combinatorial optimisation problem over the set of target representations.

  Viterbi algorithm, Eisner algorithm

- **Greedy search**

  Cast structured prediction as a sequence of classification problems: at each point in time, predict one of several options.

  window-based part-of-speech tagging, arc-standard algorithm

simplest approach: word-level classification

- what if we train a model that classifies each word independently of other words?

She	**plays**	**in**	**many**	**plays**
pronoun	verb	preposition	adverb	noun

Manchester	**United**	**will**	**return**	**to**	**the**	**United**	**States**
B-ORG	**I-ORG**	**O**	**O**	**O**	**O**	**B-LOC**	**I-LOC**

better: window-based classification

Manchester United will

better: window-based classification

Manchester United will

↓

I-ORG

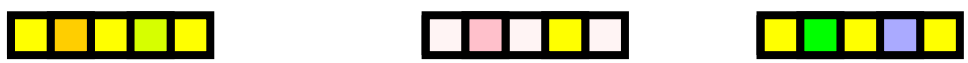
better: window-based classification

better: window-based classification

window-based classification: implementation

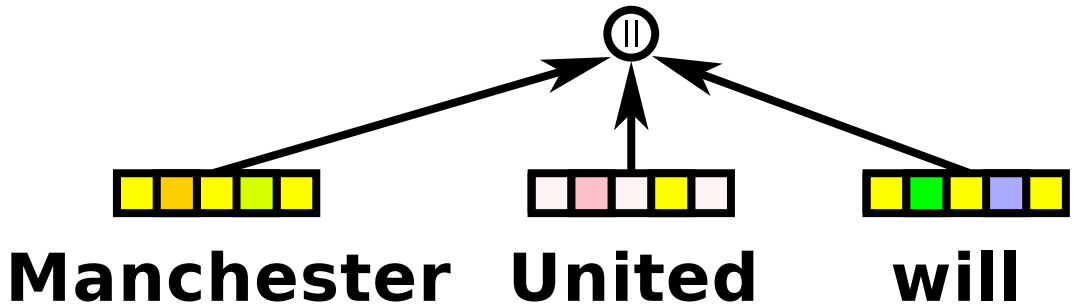
Manchester United will

window-based classification: implementation

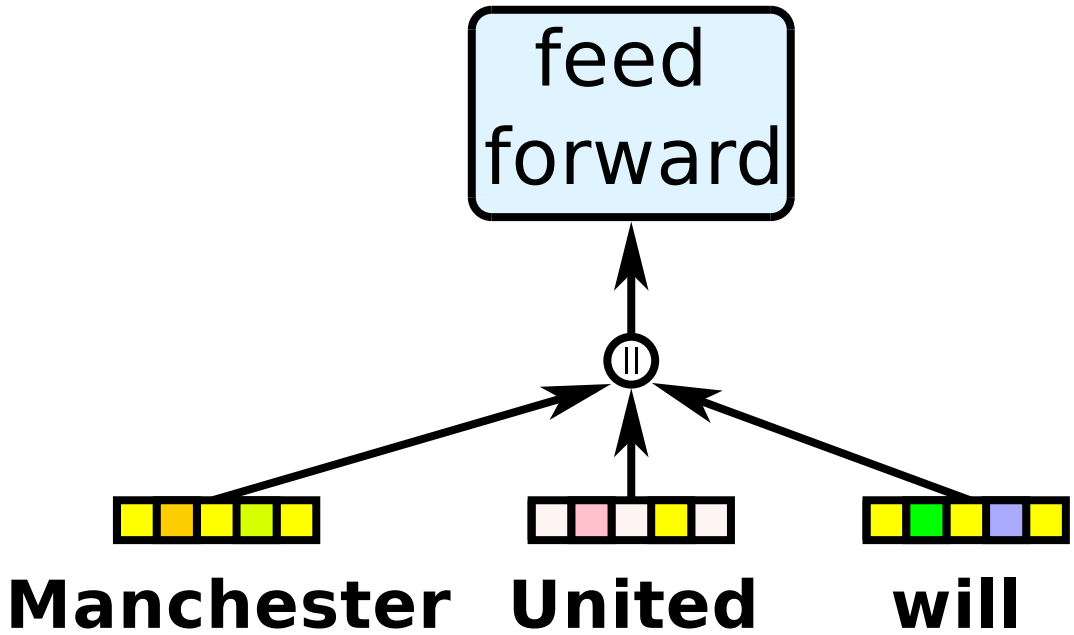


Manchester United will

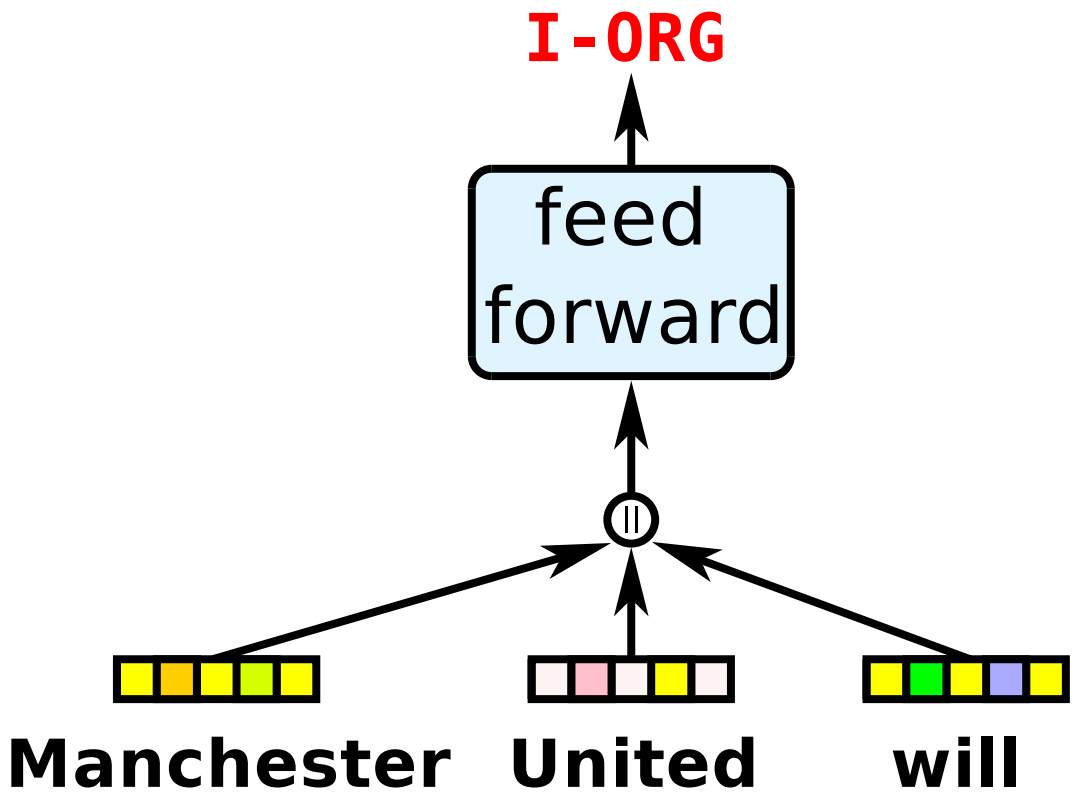
window-based classification: implementation



window-based classification: implementation

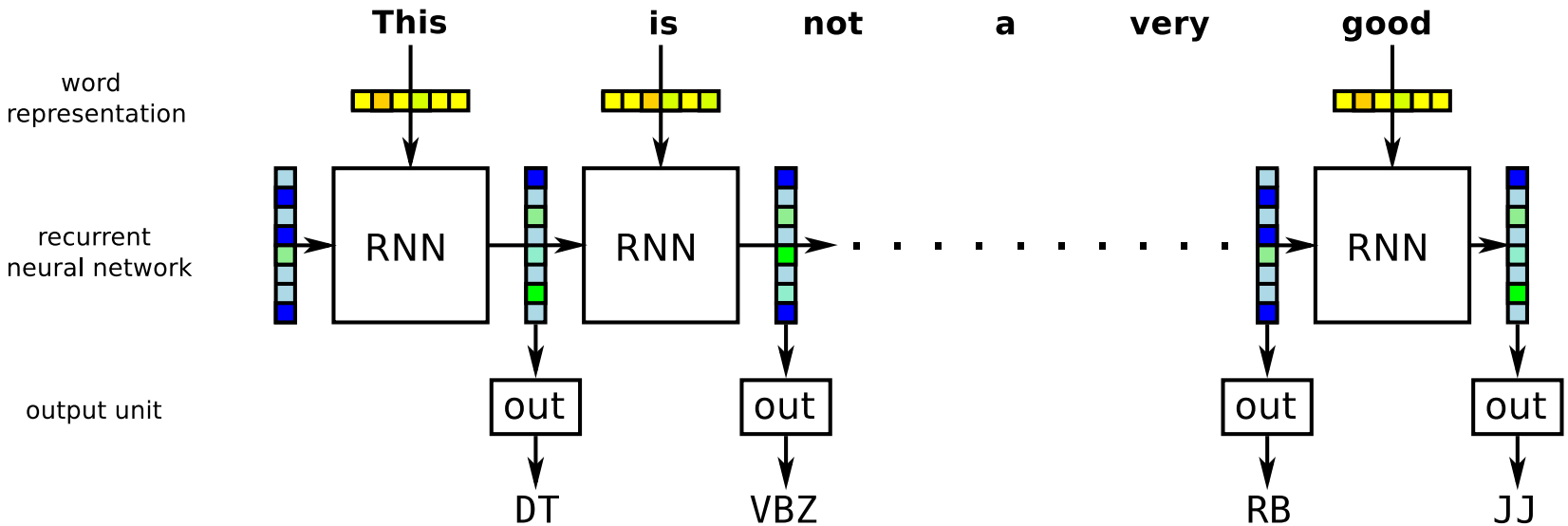


window-based classification: implementation

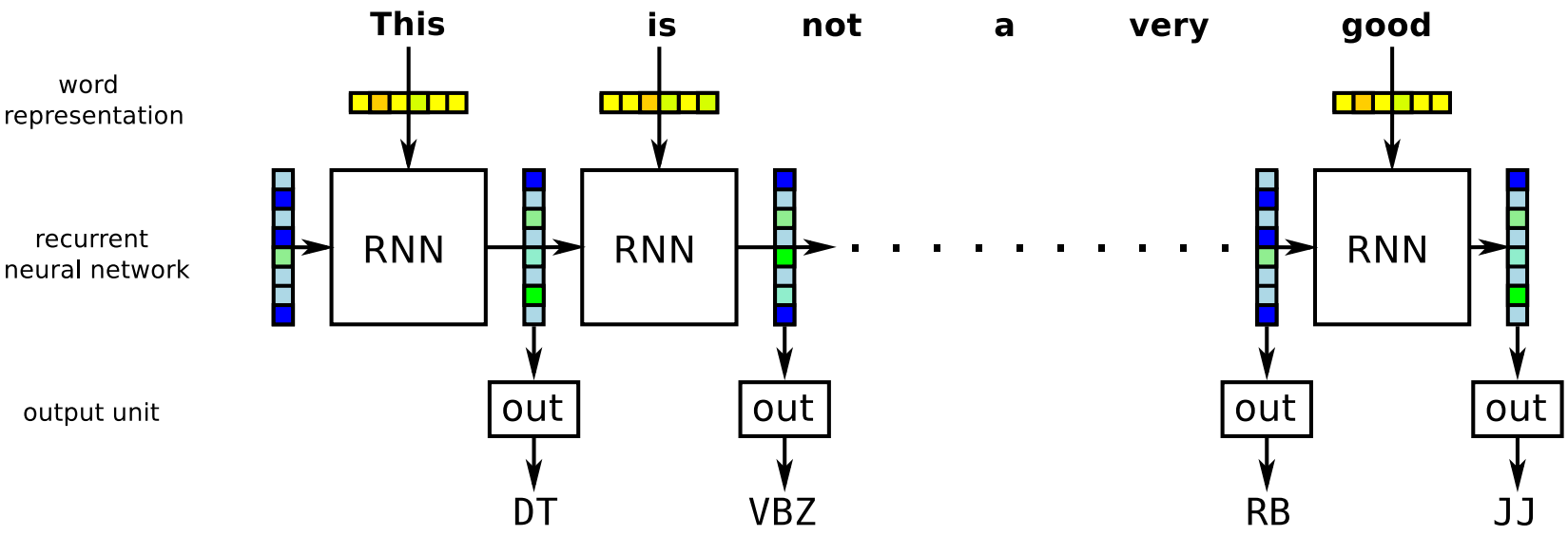


RNN-based sequence labeling

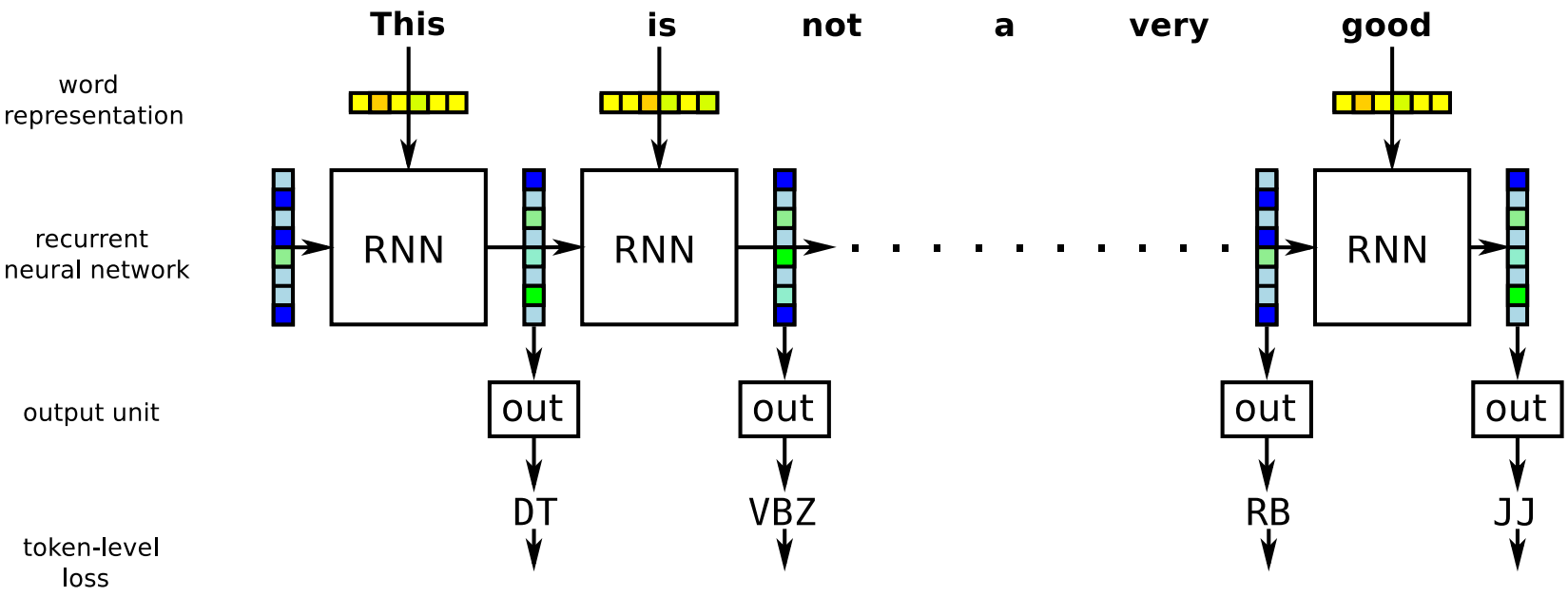
- we have already seen different types of **recurrent neural networks**, which operate on sequences
- it is straightforward to apply an output layer on top of an RNN



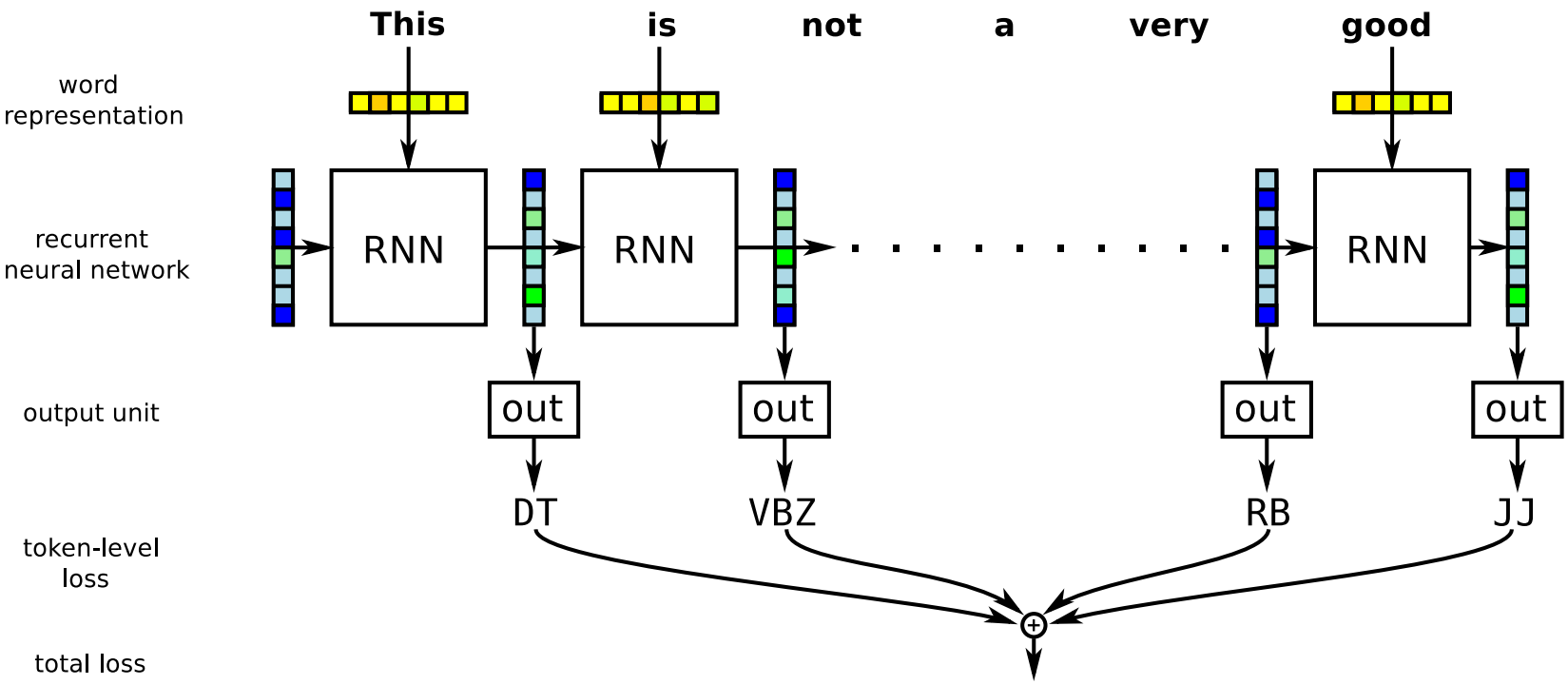
computing the loss in an RNN-based sequence labeler



computing the loss in an RNN-based sequence labeler

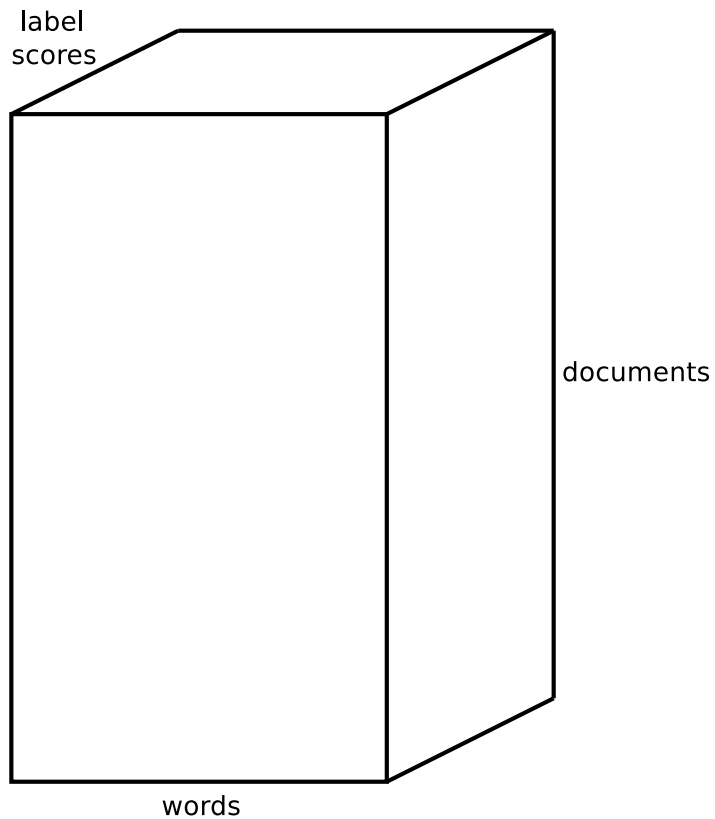


computing the loss in an RNN-based sequence labeler



a note about working with sequences in PyTorch

- the final output is a 3-dimensional tensor of shape `(n_docs, n_words, n_labels)`
- we use **padding** to make sure all documents in a batch have the same length
- if we don't want to compute the loss for the padded dummy tokens, we can use a **mask**



padding and masking example

- label sequences:

```
O       O
B-PER   O       O
O       B-LOC   I-LOC   O
```

- encoded and padded:

```
1   1   0   0
2   1   1   0
1   4   5   1
```

- loss mask:

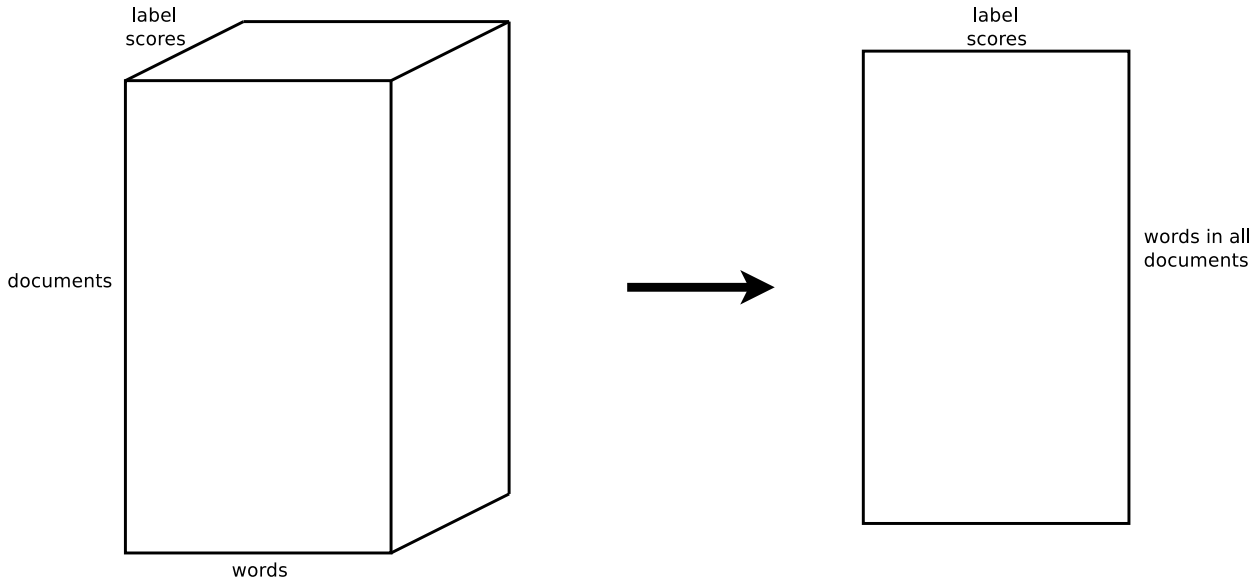
```
1   1   0   0
1   1   1   0
1   1   1   1
```

easier solution

- PyTorch provides automatic masking for some loss functions:

```
loss = nn.CrossEntropyLoss(ignore_index=pad_label_id)
```

- note that most loss functions expect 2-dimensional tensors, so we need to reshape before computing the loss



exercise 1

- simple models for **named entity recognition**

Manchester	United	will	return	to	the	United	States
B-ORG	I-ORG	O	O	O	O	B-LOC	I-LOC

- we will investigate more complex models in a second exercise