IN5550: Neural Methods in Natural Language Processing
Sub-lecture 7.1
RNNs and sequence labeling

Andrey Kutuzov

University of Oslo

03 March 2022



Contents



1. RNN applications

Recap: Common Applications of RNNs (in NLP)



- Acceptors e.g. sentence-level sentiment classification (Obligatory 2):

$$
\begin{aligned}
P(c = k|w_{1:n}) &= \hat{\boldsymbol{y}}_{[k]} \\
\hat{\boldsymbol{y}} &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}}); \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})])) \\
\boldsymbol{x_{1:n}} &= \boldsymbol{E_{[w_1]}}, \ldots, \boldsymbol{E_{[w_n]}}
\end{aligned}
$$

Recap: Common Applications of RNNs (in NLP)



- Acceptors e.g. sentence-level sentiment classification (Obligatory 2):

$$
\begin{aligned}
P(c = k | w_{1:n}) &= \hat{\boldsymbol{y}}_{[k]} \\
\hat{\boldsymbol{y}} &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[n]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[1]}])) \\
\boldsymbol{x_{1:n}} &= \boldsymbol{E}_{[w_1]}, \ldots, \boldsymbol{E}_{[w_n]}
\end{aligned}
$$

Recap: Common Applications of RNNs (in NLP)



- Acceptors e.g. sentence-level sentiment classification (Obligatory 2):

$$
\begin{aligned}
P(c = k|w_{1:n}) &= \hat{\boldsymbol{y}}_{[k]} \\
\hat{\boldsymbol{y}} &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[n]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[1]}])) \\
\boldsymbol{x_{1:n}} &= \boldsymbol{E_{[w_1]}}, \ldots, \boldsymbol{E_{[w_n]}}
\end{aligned}
$$

- transducers e.g. part-of-speech tagging (group session next week):

$$
\begin{aligned}
P(c_i = k|w_{1:n}) &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[i]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[i]}]))_{[k]} \\
\boldsymbol{x_i} &= \boldsymbol{E_{[w_i]}}
\end{aligned}
$$

Recap: Common Applications of RNNs (in NLP)



- Acceptors e.g. sentence-level sentiment classification (Obligatory 2):

$$
\begin{aligned}
P(c = k|w_{1:n}) &= \hat{\boldsymbol{y}}_{[k]} \\
\hat{\boldsymbol{y}} &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[n]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[1]}])) \\
\boldsymbol{x_{1:n}} &= \boldsymbol{E_{[w_1]}}, \ldots, \boldsymbol{E_{[w_n]}}
\end{aligned}
$$

- transducers e.g. part-of-speech tagging (group session next week):

$$
\begin{aligned}
P(c_i = k|w_{1:n}) &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[i]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[i]}]))_{[k]} \\
\boldsymbol{x_i} &= [\boldsymbol{E_{[w_i]}}; \text{RNN}^{\text{f}}_{\text{c}}(\boldsymbol{c_{1:l_i}}); \text{RNN}^{\text{b}}_{\text{c}}(\boldsymbol{c_{l_i:1}})]
\end{aligned}
$$

- character-level RNNs robust to unknown words; may capture affixation

Recap: Common Applications of RNNs (in NLP)



- Acceptors e.g. sentence-level sentiment classification (Obligatory 2):

$$
\begin{aligned}
P(c = k|w_{1:n}) &= \hat{\boldsymbol{y}}_{[k]} \\
\hat{\boldsymbol{y}} &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[n]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[1]}])) \\
\boldsymbol{x_{1:n}} &= \boldsymbol{E_{[w_1]}}, \ldots, \boldsymbol{E_{[w_n]}}
\end{aligned}
$$

- transducers e.g. part-of-speech tagging (group session next week):

$$
\begin{aligned}
P(c_i = k|w_{1:n}) &= \text{softmax}(\text{MLP}([\text{RNN}^{\text{f}}(\boldsymbol{x_{1:n}})_{[i]}; \text{RNN}^{\text{b}}(\boldsymbol{x_{n:1}})_{[i]}]))_{[k]} \\
\boldsymbol{x_i} &= [\boldsymbol{E_{[w_i]}}; \text{RNN}^{\text{f}}_{\text{c}}(\boldsymbol{c_{1:l_i}}); \text{RNN}^{\text{b}}_{\text{c}}(\boldsymbol{c_{l_i:1}})]
\end{aligned}
$$

- character-level RNNs robust to unknown words; may capture affixation
- encoder–decoder (sequence-to-sequence) models coming in a couple of weeks.

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle *Obama* *visits* *UiO* *today* .

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.
PERS			ORG		

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.
PERS			ORG		
PERS	PERS	—	ORG	—	—

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.
PERS			ORG		
PERS	PERS	—	ORG	—	—
B_{PERS}	I_{PERS}	O	B_{ORG}	O	O

Sequence Labeling in Natural Language Processing



- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.
PERS			ORG		
PERS	PERS	—	ORG	—	—
B_{PERS}	I_{PERS}	O	B_{ORG}	O	O
B_{PERS}	E_{PERS}	O	S_{ORG}	O	O

Sequence Labeling in Natural Language Processing



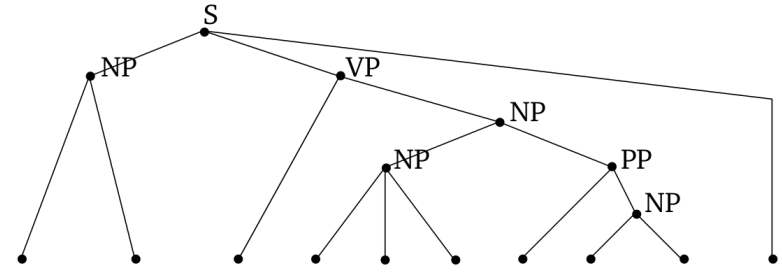
- Token-level class assignments in sequential context, aka tagging
- e.g. phoneme sequences, parts of speech; chunks, named entities, etc.
- some structure transcending individual tokens can be approximated

Michelle	*Obama*	*visits*	*UiO*	*today*	.
NNP	NNP	VBZ	NNP	RB	.
PERS			ORG		
PERS	PERS	—	ORG	—	—
B_{PERS}	I_{PERS}	O	B_{ORG}	O	O
B_{PERS}	E_{PERS}	O	S_{ORG}	O	O

- IOB (aka BIO) labeling scheme — and variants — encodes chunkings.

Constituent Parsing as Sequence Labeling

Linearization of constituent trees:



Linearized tree (absolute scale):

2NP 1S 2VP 4NP 4NP 3NP 4PP 5NP 1S

Linearized tree (relative scale):

2NP -1S 1VP 2NP 0NP -1NP 1PP 1NP -4S

Constituent parsing as sequence labeling

[Gómez-Rodríguez and Vilares, 2018]

Constituent Parsing as Sequence Labeling

Carlos Gómez-Rodríguez
Universidade da Coruña
FASTPARSE Lab, LyS Group
Departamento de Computación
Campus de Elviña s/n, 15071
A Coruña, Spain
carlos.gomez@udc.es

David Vilares
Universidade da Coruña
FASTPARSE Lab, LyS Group
Departamento de Computación
Campus de Elviña s/n, 15071
A Coruña, Spain
david.vilares@udc.es

Abstract

We introduce a method to reduce constituent parsing to sequence labeling. For each word w_t, it generates a label that encodes: (1) the number of ancestors in the tree that the words w_t and w_{t+1} have in common, and (2) the nonterminal symbol at the lowest common ancestor. We first prove that the proposed encoding function is injective for any tree without unary branches. In practice, the approach is made extensible to all constituency trees by collapsing unary branches. We then use the PTB and CTB treebanks as testbeds and propose a set of fast baselines. We achieve 90% F-score on the PTB test set, outperforming the Vinyals et al. (2015) sequence-to-sequence parser. In addition, sacrificing some accuracy, our approach achieves the fastest constituent parsing speeds reported to date on PTB by a wide margin.

1 Introduction

Zhang, 2017; Fernández-González and Gómez-Rodríguez, 2018).

With an aim more related to our work, other authors have reduced constituency parsing to tasks that can be solved faster or in a more generic way. Fernández-González and Martins (2015) reduce phrase structure parsing to dependency parsing. They propose an intermediate representation where dependency labels from a head to its dependents encode the nonterminal symbol and an attachment order that is used to arrange nodes into constituents. Their approach makes it possible to use off-the-shelf dependency parsers for constituency parsing. In a different line, Vinyals et al. (2015) address the problem by relying on a sequence-to-sequence model where trees are linearized in a depth-first traversal order. Their solution can be seen as a machine translation model that maps a sequence of words into a parenthesized version of the tree. Choe and Charniak (2016) re-

Two Definitions of 'Sequence Labeling'



- Hmm, actually, what exactly does one mean by sequence labeling?

Two Definitions of 'Sequence Labeling'



- Hmm, actually, what exactly does one mean by sequence labeling?
- gentle definition: class predictions for all elements, in context;
  - pointwise classification; each individual decision is independent;
  - no (direct) model of wellformedness conditions on class sequence;

Two Definitions of 'Sequence Labeling'



- Hmm, actually, what exactly does one mean by sequence labeling?
- gentle definition: class predictions for all elements, in context;
  - pointwise classification; each individual decision is independent;
  - no (direct) model of wellformedness conditions on class sequence;
- strict definition: sequence labeling performs structured prediction;
  - search for 'globally' optimal solution, e.g. most probable sequence;
  - models (properties of) output sequence explicitly, e.g. class bi-grams;
  - later time points impact earlier choices, i.e. revision of path prefix;

Two Definitions of 'Sequence Labeling'



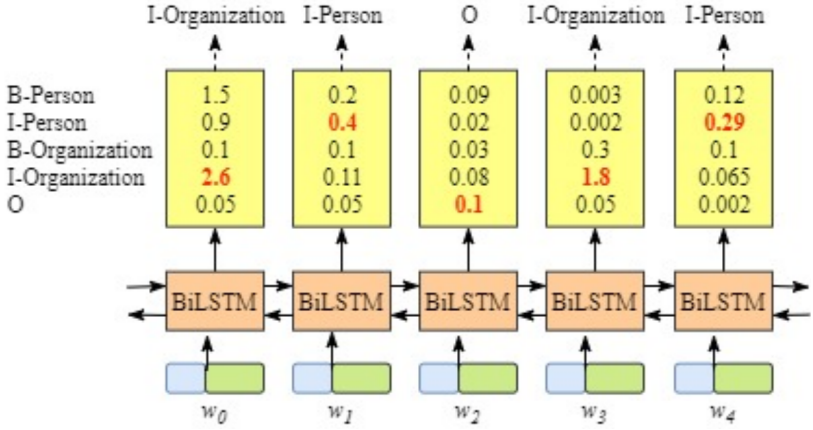
- Hmm, actually, what exactly does one mean by sequence labeling?
- gentle definition: class predictions for all elements, in context;
  - pointwise classification; each individual decision is independent;
  - no (direct) model of wellformedness conditions on class sequence;
- strict definition: sequence labeling performs structured prediction;
  - search for 'globally' optimal solution, e.g. most probable sequence;
  - models (properties of) output sequence explicitly, e.g. class bi-grams;
  - later time points impact earlier choices, i.e. revision of path prefix;
  - search techniques: dynamic programming, conditional random fields (CRF), beam search, re-ranking, etc

Two Definitions of 'Sequence Labeling'



- Hmm, actually, what exactly does one mean by sequence labeling?
- gentle definition: class predictions for all elements, in context;
  - pointwise classification; each individual decision is independent;
  - no (direct) model of wellformedness conditions on class sequence;
- strict definition: sequence labeling performs structured prediction;
  - search for 'globally' optimal solution, e.g. most probable sequence;
  - models (properties of) output sequence explicitly, e.g. class bi-grams;
  - later time points impact earlier choices, i.e. revision of path prefix;
  - search techniques: dynamic programming, conditional random fields (CRF), beam search, re-ranking, etc
- Will do part of speech tagging (sequence labeling) with RNNs next week at the group session.

Sequence-Level Output Constraints



Question: What constraint is broken here?

References I



Gómez-Rodríguez, C. and Vilares, D. (2018).
Constituent parsing as sequence labeling.
In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1314–1324, Brussels, Belgium. Association for Computational Linguistics.