Neural Architectures for Nested NER through Linearization

Jana Straková and **Milan Straka** and **Jan Hajič**
Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
{strakova,straka,hajic}@ufal.mff.cuni.cz

Abstract

We propose two neural network architectures for nested named entity recognition (NER), a setting in which named entities may overlap and also be labeled with more than one label. We encode the nested labels using a linearized scheme. In our first proposed approach, the nested labels are modeled as multilabels corresponding to the Cartesian product of the nested labels in a standard LSTM-CRF architecture. In the second one, the nested NER is viewed as a sequence-to-sequence problem, in which the input sequence consists of the tokens and output sequence of the labels, using hard attention on the word whose label is being predicted. The proposed methods outperform the nested NER state of the art on four corpora: ACE-2004, ACE-2005, GENIA and Czech CNEC. We also enrich our architectures with the recently published contextual embeddings: ELMo, BERT and Flair, reaching further improvements for the four nested entity corpora. In addition, we report flat NER state-of-the-art results for CoNLL-2002 Dutch and Spanish and for CoNLL-2003 English.

1 Introduction

In nested named entity recognition, entities can be overlapping and labeled with more than one label such as in the example *"The Florida Supreme Court"* containing two overlapping named entities *"The Florida Supreme Court"* and *"Florida"*.[1]

Recent publications on nested named entity recognition involve stacked LSTM-CRF NE recognizer (Ju et al., 2018), or a construction of a special structure that explicitly captures the nested entities, such as a constituency graph (Finkel and Manning, 2009) or various modifications of a directed hypergraph (Lu and Roth, 2015; Katiyar and Cardie, 2018; Wang and Lu, 2018).

We propose two completely neural network architectures for nested nested named entity recognition which do not explicitly build or model any structure and infer the relationships between nested NEs implicitly:

- In the first model, we concatenate the nested entity multiple labels into one multilabel, which is then predicted with a standard LSTM-CRF (Lample et al., 2016) model. The advantages of this model are simplicity and effectiveness, because an already existing NE pipeline can be reused to model the nested entities. The obvious disadvantage is a large growth of NE classes.
- In the second model, the nested entities are encoded in a sequence and then the task can be viewed as a sequence-to-sequence (seq2seq) task, in which the input sequence are the tokens (forms) and the output sequence are the labels. The decoder predicts labels for each token, until a special label "<eow>" (end of word) is predicted and the decoder moves to the next token.

The expressiveness of the models depends on a non-ambiguous encoding of the nested entity structure. We use an enhanced BILOU scheme described in Section 4.1.

The proposed models surpass the current nested NER state of the art on four nested entity corpora: ACE-2004, ACE-2005, GENIA and Czech CNEC. When the recently introduced contextual embeddings – ELMo (Peters et al., 2018), BERT (Devlin et al., 2018) and Flair (Akbik et al., 2018) – are added to the architecture, we reach further improvements for the above mentioned nested entity corpora and also exceed current state of the art for CoNLL-2002 Dutch and Spanish and for CoNLL-2003 English.

[1] Example from ACE-2004 (Doddington et al., 2004), https://catalog.ldc.upenn.edu/LDC2005T09.

2 Related Work

Finkel and Manning (2009) explicitly model the nested structure as a syntactic constituency tree.

Ju et al. (2018) run a stacked LSTM-CRF NE recognizer as long as at least one nested entity is predicted, from innermost to outermost entities.

Wang and Lu (2018) build a hypergraph to capture all possible entity mentions in a sentence.

Katiyar and Cardie (2018) model nested entities as a directed hypergraph similar to Lu and Roth (2015), using RNNs to model the edge probabilities.

Our proposed architectures are different from these works because they do not explicitly build any structure to model the nested entities. The nested entity structure is instead encoded as a sequence of labels, and the artificial neural network is supposed to model the structural relationships between the named entities implicitly.

A sequence-to-sequence architecture similar to one of our approaches is used by (Liu and Zhang, 2017) to predict the hierarchy of constituents in order to extract lookahead features for a shift-reduce constituency parser.

3 Datasets

We evaluate our results on four nested NE corpora:

- English **ACE-2004**, (Doddington et al., 2004)[2]. We reuse the train/dev/test split used by most previous authors (Lu and Roth, 2015; Muis and Lu, 2017; Wang and Lu, 2018).
- English **ACE-2005**[3]. Again, we use the train/dev/test split by Lu and Roth (2015); Muis and Lu (2017); Wang and Lu (2018).
- English **GENIA** (Kim et al., 2003). We use the 90%/10% train/test split used by previous authors (Finkel and Manning, 2009; Lu and Roth, 2015; Muis and Lu, 2017; Wang and Lu, 2018).
- Czech **CNEC** – Czech Named Entity Corpus 1.0. As previous authors (Straková et al., 2016), we predict the 42 fine-grained *NE types* and 4 *containers* from the first annotation round.

We evaluate flat NER on these four languages: **CoNLL-2003** English and German (Tjong Kim Sang and De Meulder, 2003) and **CoNLL-2002** Dutch and Spanish (Tjong Kim Sang, 2002).

In all cases, we use the train portion of the data for training and the development portion for hyperparameter tuning, and we report our final results on models trained on concatenated train+dev portions and evaluated on the test portion, following e.g. (Ratinov and Roth, 2009; Lample et al., 2016).

Our evaluation is a strict one: each entity mention is considered correct only when both the span and class are correct.

4 Methods

4.1 Nested NE BILOU Encoding

Our goal is to encode the nested entity structure into a CoNLL-like, per-token BILOU encoding,[4] as in the following example for sentence *"in the US Federal District Court of New Mexico ."*:

```
in              O
the             B-ORG
US              I-ORG|U-GPE
Federal         I-ORG
District        I-ORG|U-GPE
Court           I-ORG
of              I-ORG
New             I-ORG|B-GPE
Mexico          L-ORG|L-GPE
.               O
```

The mapping from tokens to multilabels is defined by the two following rules: (1) entity mentions starting earlier have priority over entities starting later, and (2) for mentions with the same beginning, longer entity mentions have priority over shorter ones. A multilabel for a word is then a concatenation of all intersecting entity mentions, from the highest priority to the lowest.

Another, more formalized look at the BILOU encoding is that it is a BILOU encoding of an unfolded directed hypergraph similar to Katiyar and Cardie (2018), in which the shared entity labels are not collapsed and the `O` is used only for tokens outside any entity mention.

We use a trivial heuristic during decoding, matching labels of consecutive words by order only. Therefore, an `I-` or `L-` label is merged with a preceding `B-` or `I-` if they appear on the same position in neighboring multilabels and have the same type.

[2] https://catalog.ldc.upenn.edu/LDC2005T09

[3] https://catalog.ldc.upenn.edu/LDC2006T06

[4] `B-` (beginning), `I-` (inside), `U-` (unit-length entity), `L-` (last) or `O` (outside) labels (Ratinov and Roth, 2009).

4.2 Neural Models for Nested NER

Both our models are encoder-decoder architectures:

LSTM-CRF: The encoder is a bi-directional LSTM and the decoder is a CRF (Lample et al., 2016), modeling multilabels from Section 4.1.

Sequence-to-sequence (seq2seq): The encoder is a bi-directional LSTM and the decoder is a LSTM. The tokens are viewed as the input sequence, and the encoded labels are predicted one by one by the decoder, until the decoder outputs the `"<eow>"` (end of word) label and moves to the next token. We use a hard attention on the word whose label(s) is being predicted, and predict labels for a word from highest to lowest priority as defined in Section 4.1.

We train the network using the lazy variant of the Adam optimizer (Kingma and Ba, 2014), which only updates accumulators for variables that appear in the current batch,[5] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.98$. We use mini-batches of size 8. As a regularization, we apply dropout with rate 0.5 and the word dropout replaces 20% of words by the unknown token to force the network to rely more on context. We did not perform any complex hyperparameter search.

In our baseline versions, we use the following word- and character-level word embeddings:

- pretrained word embeddings: For English, we train our own word embeddings of dimension 300 with `word2vec`[6] on the English Gigaword Fifth Edition.[7] For other languages (German, Dutch, Spanish and Czech) we use the FastText word embeddings (Bojanowski et al., 2017).[8]
- end-to-end word embeddings: We embed the input forms and lemmas (256 dimensions) and POS tags (one-hot).
- character-level word embeddings: We use bidirectional GRUs (Cho et al., 2014; Graves and Schmidhuber, 2005) of dimension 128 in line with Ling et al. (2015): we represent every Unicode character with a vector of dimension 128, and concatenate GRU outputs for forward and reversed word characters.

We further add contextual word embeddings to our baselines:

- **+ELMo** (Peters et al., 2018): pretrained contextual word embeddings of dimension 512 for English.
- **+BERT** (Devlin et al., 2018): pretrained contextual word embeddings of dimension 1024 for English[9] and 768 for other languages[10]. For each token, we generate the contextual word embedding by averaging all BERT subword embeddings in the last four layers (Devlin et al., 2018) without finetuning.
- **+Flair** (Akbik et al., 2018): pretrained contextual word embeddings of dimension 4096 for all languages except Spanish.[11]

We use the implementation provided by Akbik et al. (2018) to generate the Flair and ELMo word embeddings.[12]

We do not use any hand-crafted classification features in any of our models.

5 Results

Table 1 shows the F1 score for the nested NER and Table 2 shows the F1 score for the flat NER.

When comparing the results for the nested NER in the baseline models (without the contextual word embeddings) to the previous results in literature, we see that **LSTM-CRF** reaches comparable, but suboptimal results in three out of four nested NE corpora, while **seq2seq** clearly outperforms all the known methods by a wide margin. We hypothesize that **seq2seq**, although more complex (the system must predict multiple labels per token, including the special label `"<eow>"`), is more suitable for more complex corpora. The gain is most visible in ACE-2004 and ACE-2005, which contain extremely long named entities and the level of "nestedness" is greater than in the other nested corpora. According to Wang and Lu (2018), 39% of train sentences contain overlapping mentions in ACE-2004, as opposed to 22% of train sentences with overlapping mentions in GENIA. With shorter and less overlapping entities, such as in GENIA, and ultimately in flat

[5] `tf.contrib.opt.lazyadamoptimizer` from `www.tensorflow.org`

[6] Skip-gram, for tokens with at least 10 occurrences, window = 5, dimension = 300, negative sampling = 5.

[7] `https://catalog.ldc.upenn.edu/LDC2011T07`

[8] `https://fasttext.cc/docs/en/crawl-vectors.html`

[9] BERT-Large Uncased from `https://github.com/google-research/bert`

[10] BERT-Base Multilingual Uncased from `https://github.com/google-research/bert`

[11] Not yet available in December 2018.

[12] `https://github.com/zalandoresearch/flair`

model	ACE-2004	ACE-2005	GENIA	CNEC 1.0
(Finkel and Manning, 2009)**	–	–	70.3	–
(Lu and Roth, 2015)**	62.8	62.5	70.3	–
(Muis and Lu, 2017)**	64.5	63.1	70.8	–
(Katiyar and Cardie, 2018)	72.70	70.5	73.6	–
(Ju et al., 2018)*	–	72.2	74.7	–
(Wang and Lu, 2018)	75.1	74.5	75.1	–
(Straková et al., 2016)	–	–	–	81.20
LSTM-CRF	72.26	71.62	*76.23*	80.28
LSTM-CRF+ELMo	*78.72*	*78.36*	*75.94*	–
LSTM-CRF+BERT	*81.48*	*79.95*	*77.80*	*85.67*
LSTM-CRF+Flair	*77.65*	*77.25*	*76.65*	*81.74*
LSTM-CRF+BERT+ELMo	*80.07*	*80.04*	*76.29*	–
LSTM-CRF+BERT+Flair	*81.22*	*80.82*	*77.91*	*85.70*
LSTM-CRF+ELMo+BERT+Flair	*80.19*	*79.85*	*76.56*	–
seq2seq	*77.08*	*75.36*	*76.44*	*82.96*
seq2seq+ELMo	*81.94*	*81.95*	*77.33*	–
seq2seq+BERT	*84.33*	*83.42*	*78.20*	*86.73*
seq2seq+Flair	*81.38*	*79.83*	*76.63*	*83.55*
seq2seq+BERT+ELMo	*84.32*	*82.15*	*77.77*	–
seq2seq+BERT+Flair	**84.40**	**84.33**	**78.31**	**86.88**
seq2seq+ELMo+BERT+Flair	*84.07*	*83.41*	*78.01*	–

Table 1: Nested NER results (F1) for ACE-2004, ACE-2005, GENIA and CNEC 1.0 (Czech) corpora. **Bold** indicates the best result, *italics* results above SoTA and gray background indicates the main contribution. * uses different data split in ACE-2005. ** non-neural model

model	English	German	Dutch	Spanish
(Gillick et al., 2016)	86.50	76.22	82.84	82.95
(Lample et al., 2016)	90.94	78.76	81.74	85.75
ELMo (Peters et al., 2018)	92.22	–	–	–
Flair (Akbik et al., 2018)	93.09	**88.32**	–	–
BERT (Devlin et al., 2018)	92.80	–	–	–
LSTM-CRF	90.72	79.89	*87.42*	*86.34*
LSTM-CRF+ELMo	92.58	–	–	–
LSTM-CRF+BERT	92.94	84.53	*92.48*	*88.77*
LSTM-CRF+Flair	92.25	82.35	*88.31*	–
LSTM-CRF+BERT+ELMo	92.93	–	–	–
LSTM-CRF+BERT+Flair	*93.22*	84.44	**92.69**	–
LSTM-CRF+ELMo+BERT+Flair	**93.38**	–	–	–
seq2seq	90.77	79.09	*87.59*	*86.04*
seq2seq+ELMo	92.43	–	–	–
seq2seq+BERT	92.98	84.19	*92.46*	**88.81**
seq2seq+Flair	91.87	82.68	*88.67*	–
seq2seq+BERT+ELMo	92.99	–	–	–
seq2seq+BERT+Flair	93.00	85.10	*92.34*	–
seq2seq+ELMo+BERT+Flair	93.07	–	–	–

Table 2: Flat NER results (F1) for CoNLL-2002 and CoNLL-2003. **Bold** indicates best result, *italics* results above SoTA.

corpora, the simplicity of **LSTM-CRF** wins over **seq2seq**.

We also report a substantial increase in the F1 score when recently published contextual embeddings (ELMo, BERT, Flair) are added as pretrained word embeddings on input (Peters et al., 2018; Devlin et al., 2018; Akbik et al., 2018) in all languages and corpora, although in the case of CoNLL-2003 German, our results stay behind those of Akbik et al. (2018).

6 Conclusions

We presented two neural architectures for nested named entities and a simple encoding algorithm to allow the modeling of multiple NE labels in an enhanced BILOU scheme. The LSTM-CRF modeling of NE multilabels is better suited for putatively less-nested and flat corpora, while the sequence-to-sequence architecture captures more complex relationships between nested and complicated named entities and surpasses the current state of the art in nested NER on four nested NE corpora. We also report surpassing state-of-the-art results with the recently published contextual word embeddings on both nested and flat NE corpora.

Acknowledgements

The work described herein has been supported by OP VVV VI LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (project CZ.02.1.01/0.0/0.0/16_013/0001781) and it has been supported and has been using language resources developed by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071).

We would like to thank the reviewers for their insightful comments.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

KyungHyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *CoRR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The Automatic Content Extraction (ACE) program-tasks, data, and evaluation. *Proceedings of LREC*, 2.

Jenny Rose Finkel and Christopher D. Manning. 2009. Nested Named Entity Recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 141–150.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2016. Multilingual language processing from bytes. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1296–1306. Association for Computational Linguistics.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, pages 5–6.

Meizhi Ju, Makoto Miwa, and Sophia Ananiadou. 2018. A neural layered model for nested named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1446–1459. Association for Computational Linguistics.

Arzoo Katiyar and Claire Cardie. 2018. Nested named entity recognition revisited. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 861–871. Association for Computational Linguistics.

Jing-Dong Kim, Tomoto Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. GENIA corpus—A semantically annotated corpus for bio-textmining. *Bioinformatics (Oxford, England)*, 19 Suppl 1:180–182.

Diederik Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics.

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W. Black, and Isabel Trancoso. 2015. Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. *CoRR*.

Jiangming Liu and Yue Zhang. 2017. Shift-Reduce Constituent Parsing with Neural Lookahead Features. *Transactions of the Association for Computational Linguistics*, 5:45–58.

Wei Lu and Dan Roth. 2015. Joint mention extraction and classification with mention hypergraphs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 857–867. Association for Computational Linguistics.

Aldrian Obaja Muis and Wei Lu. 2017. Labeling Gaps Between Words: Recognizing Overlapping Mentions with Mention Separators. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2608–2618. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Lev Ratinov and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *CoNLL '09: Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155. Association for Computational Linguistics.

Jana Straková, Milan Straka, and Jan Hajič. 2016. Neural Networks for Featureless Named Entity Recognition in Czech. In *Text, Speech, and Dialogue: 19th International Conference, TSD 2016, Brno , Czech Republic, September 12-16, 2016, Proceedings*, pages 173–181. Springer International Publishing.

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 Shared Task: Language-independent Named Entity Recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

Bailin Wang and Wei Lu. 2018. Neural segmental hypergraphs for overlapping mention recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 204–214. Association for Computational Linguistics.