Hierarchical Region Learning for Nested Named Entity Recognition

Xinwei Long[1,2], **Shuzi Niu**[1]* **and Yucheng Li**[1]
[1]Institute of Software, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
longxinwei19@mails.ucas.ac.cn, {shuzi, yucheng}@iscas.ac.cn

Abstract

Named Entity Recognition (NER) is deeply explored and widely used in various tasks. Usually, some entity mentions are nested in other entities, which leads to the nested NER problem. Leading region based models face both the efficiency and effectiveness challenge due to the high subsequence enumeration complexity. To tackle these challenges, we propose a hierarchical region learning framework to automatically generate a tree hierarchy of candidate regions with nearly linear complexity and incorporate structure information into the region representation for better classification. Experiments on benchmark datasets ACE-2005, GENIA and JNLPBA demonstrate competitive or better results than state-of-the-art baselines.

1 Introduction

As a fundamental information extraction task, Named Entity Recognition (NER) is widely used in various downstream tasks, such as entity linking and entity search. Most studies assigns a label to each token of the sequence for the flat NER problem (Lample et al., 2016). However, it is common that entities are embedded in other entities in many domains (Kim et al., 2003; Ringland et al., 2019). Example from ACE-2005 dataset shown in Fig. 1 illustrates that the top-level PER entity includes a nested entity with ORG label. How to recognize all entities recursively from innermost to outermost is referred to as the Nested NER problem.

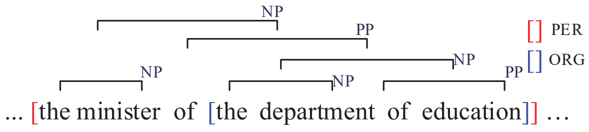

Figure 1: Illustration of nested entities and constituent parsing tree

Existing approaches mainly solve the nested NER problem by classifying all candidate subsequences (a.k.a regions). The key to region based methods lies in candidate region detection. one kind is the brute force method (Sohrab and Miwa, 2018) to enumerate all possible $\mathcal{O}(n^2)$ subsequences for each sentence with n words. The other kind (Zheng et al., 2019) is to generate and classify candidate regions in a two-stage paradigm, often leading to cascaded errors. Thus region based methods face efficiency and effectiveness challenges.

To tackle these challenges, we propose a **Hi**erarchical **Re**gion learning framework, referred to as HiRe. First, inspired by constituent parsing tree as the top of Fig. 1 and its neural syntactic distance (Shen et al., 2018), we introduce the coherence measure between adjacent regions. Then we generate a region tree for each sentence by merging two adjacent regions recursively based on this region coherence measure in a bottom-up manner. Finally, hierarchical regions are classified based on the boundary and merging word representation. We train the hierarchical region generation and classification tasks simultaneously.

Experimental results on three benchmark datasets ACE-2005, GENIA and JNLPBA demonstrate that HiRe shows the competitive or better performance than baselines. HiRe generates only $\mathcal{O}(n)$ candidate regions about 77.9% less than the brute-force method and achieves 98.1% true region recall in the GENIA dataset, a good trade-off between efficiency and effectiveness.

2 Related Work

Given a sentence of n words $(w_1, \dots, w_n)$, the nested named entity recognition task aims at identifying all the entities especially when one entity subsequence $(w_i, \dots, w_j), i < j$ contains others $(w_p, \dots, w_q), i \leq p < q \leq j$. According to reduced different problems, existing nested NER

[*] Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4788–4793
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

models mainly fall into three categories.

Sequence labeling models assign multiple labels to each word assuming that one word may belong to multiple entities, such as linearization method (Straková et al., 2019) and layered CRF (Ju et al., 2018).

Structured label classification models capture the label relationship of a sentence for better performance. (Lu and Roth, 2015; Wang and Lu, 2018) proposed hyper-graphs models to describe the label relationship, and either human designed or latent features were adopted for classification.

Region based models were summarized by (Lin et al., 2019) as obtaining all possible regions and assigning labels to regions. The key to region classification models is how to obtain candidate regions from a sentence. One is the brute-force method (Sohrab and Miwa, 2018; Xia et al., 2019), which enumerates all subsequences of a sentence for classification with high time complexity. The other is to formulate the task as a two-stage paradigm. (Zheng et al., 2019; Tan et al., 2020) detected a small set of candidate regions with high efficiency, but only about 80% entities could be found in the first stage, making a performance bottleneck. Some studies (Finkel and Manning, 2009) leveraged the external knowledge, such as constituent parsing tree, to guide the first step, which achieved impressive performance but suffered from the cubic time complexity and error propagation from external tools. Most methods above represented the region as the average or weighted sum of word representations, ignoring the region structure.

3 Methods

To tackle efficiency and effectiveness challenges in region based methods, we propose a **Hi**erarchical **Re**gion learning framework for nested NER problem, namely **HiRe** in Fig. 2.

3.1 Overall Architecture

Specifically, we first obtain word representations through the encoder layer. Then, we introduce a word coherence measure based on word representations through word coherence layer. Next, region coherence measure is derived from the word coherence, two adjacent regions are recursively merged based on this measure, and a tree of regions is generated for each sentence. Finally, we use a ranking loss of region boundaries for region generation task and cross entropy loss of labeling candidate regions for entity recognition task in a multi-task framework.

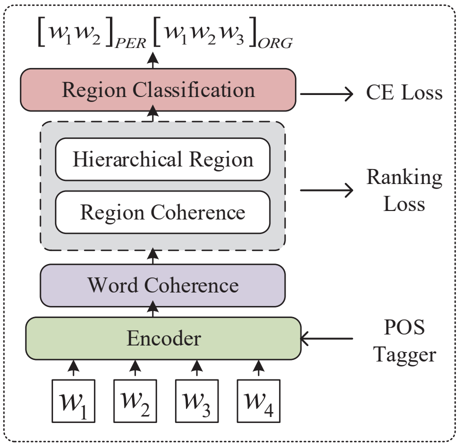

Figure 2: Architecture of HiRe.

Encoder Layer. Consider the i-th word w_i in a sentence with n words, we represent it by concatenating their word embedding x_i^w, part-of-speech(POS) embedding x_i^p and character-level embedding x_i^c together. The character-level embeddings are generated by a convolutional neural network module with the same setting as (Yang et al., 2018) to capture the orthographic and morphological features of the word. Then, we employ a bi-directional LSTM to obtain the long-term context-aware representation as:

$$x_i^t = [x_i^w; x_i^p; x_i^c], \tag{1}$$

$$\overrightarrow{h}_i = LSTM(x_i^t, \overrightarrow{h}_{i-1}), \tag{2}$$

$$\overleftarrow{h}_i = LSTM(x_i^t, \overleftarrow{h}_{i+1}), \tag{3}$$

$$h_i = [\overrightarrow{h}_i; \overleftarrow{h}_i], \tag{4}$$

Word Coherence. Word context representations $\{h_t\}_{t=0}^{n-1}$ are fed to the convolutional kernel with window size 2 to obtain the local feature between adjacent words $g_0, g_2, ...g_{n-2} = CONV(h_0, h_1, \dots, h_{n-1})$. Then these features are input into a 2-layers feed-forward network (FFN) to obtain the word coherence measure $\{d_t\}_{t=0}^{n-2}$, where d_t indicates the affinity between word w_t and w_{t+1}. The higher this measure, the more coherent adjacent words.

Region Coherence. A subsequence of the sentence composed of consecutive words is called a region denoted as $\mathcal{R}_{i,j} = (w_i, \dots, w_j)$. Based on the word coherence measure, we define the region coherence based on adjacent words between two adjacent regions in Eq.(5). It indicates how likely two adjacent regions are to be a whole.

$$d(\mathcal{R}_{i,j}, \mathcal{R}_{j+1,k}) = d_j, i \leq j < k, \qquad (5)$$

Hierarchical Region Generation. Based on region coherence measure, we build the region hierarchy from bottom to up recursively as follows. At 1-st level for initialization, each word is treated as a region and the leaf node in this tree. At t-th level, two regions $\mathcal{R}_{i,k}$ and $\mathcal{R}_{k+1,j}$ will be merged into $\mathcal{R}_{i,j}$ at the merging point k if $d(\mathcal{R}_{i,k}, \mathcal{R}_{k+1,j}) > d(\mathcal{R}_{p,i-1}, \mathcal{R}_{i,k})$ and $d(\mathcal{R}_{i,k}, \mathcal{R}_{k+1,j}) > d(\mathcal{R}_{k+1,j}, \mathcal{R}_{j+1,q})$. $\mathcal{R}_{i,j}$ will be used at the following levels instead of $\mathcal{R}_{i,k}$ and $\mathcal{R}_{k+1,j}$. Because each k has one chance to be the merging point, this merging operation will be repeated at most $n-1$ times. The process will generate about $\mathcal{O}(n)$ candidate regions. Fig. 3 illustrates this generation process of the example sentence from Fig. 1, where blocks with the same color are of the same region. Practically, it is not essential to generate the whole tree with the restraint of maximum entity length, which further reduces the number of candidate regions.

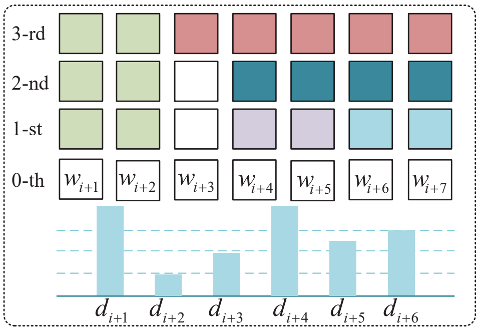

Figure 3: Hierarchical Region Generation for Fig. 1, where w_{i+l} represents the $(i+l)$-th word in the sequence. The blue histograms on the bottom represent the coherence scores, and the blocks with the same color in each layer indicate they have been merged into a region.

Region Classification. Here a region is composed of two sub-regions. For a region $\mathcal{R}_{i,j}$ with its merging point k generated by the above steps, we adopt g_k as the representation of its sub-regions $\mathcal{R}_{i,k}$ and $\mathcal{R}_{k+1,j}$. To make the classifier more sensitive to entity boundaries, both boundary and merging word representations are concatenated as region $\mathcal{R}_{i,j}$'s representation $v_{[i,j]} = [h_i; g_k; h_j]$, namely hierarchical region representation. If $i = j$, we set $v_{[i,i]}$ to $[h_i; h_i; h_i]$. Next, a 2-layer feed-forward network is to predict the probability that region $\mathcal{R}_{i,j}$ belongs to entity category c as Eq.(6).

$$p(c|\mathcal{R}_{i,j}) = Softmax(FFN(v_{[i,j]})) \qquad (6)$$

3.2 Learning and Inference

We train both the hierarchical region generation and classification tasks simultaneously in a multi-task framework as Eq.(7).

$$\mathcal{L} = \alpha\mathcal{L}_{region} + (1-\alpha)\mathcal{L}_{label} \qquad (7)$$

For the hierarchical region generation task, we propose to optimize the pairwise ranking loss $\mathcal{L}_{region}$ in Eq.(8) to emphasize the partial order between inner and boundary word coherence instead of their values. The predicted partial order is determined by the learned boundary and inner word coherence scores. The loss function is reduced to each region difference between the predicted and ground truth region hierarchy.

However, The ground truth partial order is unavailable in datasets. To solve this problem, we generate the ground truth coherence scores based on the rule that the boundary word w_{i-1} and w_j coherence is always smaller than the inner word $\{w_t\}_{t=i}^{j-1}$ coherence for each ground truth entity region $\mathcal{R}_{i,j}$. Considered the hierarchy of entity, we define the ground truth word coherence as a logarithmic function of length. Specifically, Ground truth boundary word coherences $\bar{d}_{i-1}$ and $\bar{d}_j$ are defined as $-(\lfloor \log_2(j-i+2) \rfloor + 1)$. Ground truth inner word coherence $\{\bar{d}_m\}_{m=i}^{j-1}$ are randomly generated from $[-1, -\lfloor \log_2(j-i+2) \rfloor]$. Predicted word coherences $\{d_t\}_{t=i-1}^{j}$ are derived through above layers.

$$\sum_{\forall \mathcal{R}_{i,j}} \sum_{\substack{l = i-1, j \\ m \in [i, j-1]}} [1 - sign(\bar{d}_l - \bar{d}_m)(d_l - d_m)]^+ \qquad (8)$$

For the region classification task, the cross entropy loss function $\mathcal{L}_{label}$ is utilized with a softmax classifier based on the probabilities in Eq.(6).

4 Experiments

To investigate the effectiveness and efficiency of our proposed method, we conduct comprehensive experiments on three benchmark NER datasets.

4.1 Experimental Setting

NER datasets with some nested entities are referred to as nested NER datasets, while NER

Model	ACE-2005			GENIA		
	P	R	F	P	R	F
Layered-CRF (Ju et al., 2018)$^{\diamondsuit}$	74.2	70.3	72.2	78.5	71.3	74.7
Segm. HG (Wang and Lu, 2018)***[POS]**	76.8	72.3	74.5	77.0	73.3	75.1
Exhaustive (Sohrab and Miwa, 2018) [5]	-	-	-	74.6	68.2	71.2
ARNs (Lin et al., 2019)**[POS]**	76.2	73.6	74.9	75.8	73.9	74.8
M&L (Fisher and Vlachos, 2019)	75.1	74.1	74.6	-	-	-
Bound. Aware (Zheng et al., 2019)	-	-	-	75.9	73.6	74.7
BENSC (Tan et al., 2020)**[POS]**	77.1	74.2	75.6	78.9	72.7	75.7
Our Model(LSTM)**[POS]**	78.5	74.6	76.5	77.4	73.9	75.6

Table 1: Experimental results on ACE-2005 and GENIA. $\diamondsuit$ represents the sequence labeling model, $\star$ represents the structure label classification model, and others are region classification models. **[POS]** represents these methods use POS tags as features.

Model	P	R	F
(Xia et al., 2019)	79.0	77.3	78.2
(Fisher and Vlachos, 2019)	82.7	82.1	82.4
(Tan et al., 2020)	83.8	83.9	83.9
Our Model	83.0	86.3	84.6

Table 2: Experimental results on ACE-2005 with pre-trained language models.(Xia et al., 2019) use ELMo, and the others use uncased BERT-Base.

datasets without nested entities are called as flat NER datasets. We evaluate our model on the nested NER dataset ACE-2005 [1] and GENIA[2](Kim et al., 2003), which contain 36.4% and 21.8% nested entities respectively. We follow the same dataset setup as previous work (Wang and Lu, 2018; Lin et al., 2019). We also conduct ablation experiments on the flat NER dataset JNLPBA (Collier and Kim, 2004), and pre-processed data is obtained from (Zheng et al., 2019).

HiRe was implemented by Pytorch[3]. Stanford CoreNLP toolkit(Manning et al., 2014) was used to split sentences and for POS tagging. We use ADAM(Kingma and Ba, 2015) for optimization with batch size 32 and learning rate 0.001. Word embeddings are initialized with pretrained 200-dimension Glove vectors(Pennington et al., 2014)[4]. Dimensions of POS tag embedding, character embedding, LSTM layer and hidden units are 50, 100, 2 and 256 respectively. The dropout ratio is 0.2 and α is 0.4. We use BERT$_{base}$ for word representations and fine tune parameters with learning rate $3e-5$. The maximum number of hierarchical layer t is set as 8, 6, 6 on ACE, GENIA and JNLPBA separately.

4.2 Effectiveness Analysis

Table 1 shows the performance comparison between HiRe and baselines on ACE-2005 and GENIA datasets using Bi-LSTM as the encoder. On ACE-2005, F1 score of HiRe achieves 76.5% and is improved by 0.9% compared with SOTA. On GENIA, its F1 score is 75.6%, which is competitive to baselines. The performance gain on ACE-2005 is due to the high recall in the region generation step and the incorporation of region structure into its representation in region classification step. Higher performance on ACE-2005 means that HiRe performs better on datasets with more nested entities.

Considering baselines with pre-trained language model, we replace LSTM encoder with BERT$_{base}$ in HiRe. Experimental results are listed in Table 2. Our model significantly outperforms baselines. As far as we know, the only reported higher F1 score (Li et al., 2019) on ACE-2005 is obtained from BERT$_{large}$ with three times parameter number of BERT$_{base}$ to learn and infer with low efficiency.

4.3 Efficiency Analysis

Given a sentence with n words, the brute force method enumerates $\mathcal{O}(n^2)$ candidate regions. HiRe generates $\mathcal{O}(n)$ candidate regions. (Zheng et al., 2019) finds candidate regions through a token-wise classification with $\mathcal{O}(n)$ time complexity. For sentences in GENIA, the number of candi-

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

[2] http://www.geniaproject.org/genia-corpus/term-corpus

[3] https://pytorch.org/

[4] http://nlp.stanford.edu/data/glove.6B.zip

[5] Due to different experimental settings, we reproduced (Sohrab and Miwa, 2018) under the same setting with other baselines and obtained performances similar to results in (Zheng et al., 2019). The other results were taken from their papers

Model	P	R	F
(Lample et al., 2016)	69.1	74.2	71.6
(Sohrab and Miwa, 2018)	69.4	73.1	71.2
(Zheng et al., 2019)	-	-	73.6
Our Model	72.5	75.6	74.0

Table 3: Experimental results on JNLPBA.

date regions generated by HiRe is 77.9% less than that of the enumeration method discarding 1.3% long entities and more than that of (Zheng et al., 2019). However, the true recall of candidate regions generated by the enumeration method and HiRe are 98.7% and 98.1%, respectively. The recall of the start/end boundary generated by (Zheng et al., 2019) is 84.3%/87.2%. In this sense, HiRe finds a relatively smaller (20% or so) but higher quality (true recall 98.1%) subset of all regions, which is a good trade-off between efficiency and effectiveness.

4.4 Ablation Study

To prove our model can also work on flat NER task, we conduct ablation experiments on JNLPBA dataset. We compare our model with a standard flat NER benchmark (Lample et al., 2016) and two nested NER methods. Our model achieves 74.0% in F1 measure, which outperforms these baselines showed in Table 3.

To analyze the role of **H**ierarchical **R**egion **R**epresentation, denoted as HRR in HiRe, we compare performances of HiRe with and without it on ACE-2005. HiRe without HRR employs **A**verage **W**ord **R**epresentation (denoted as AWR) instead with precision 78.3%, recall 73.7% and F1 measure 75.9%. In contrast to HiRe$_{AWR}$, the absolute F1 measure improvement of HiRe$_{HRR}$ is 0.6%. In all, HRR plays an essential part in HiRe.

The reason lies in that the HRR treats each region as a hierarchical structure composed of two subregions rather than a flat structure as AWR does. The hierarchical structure will put more emphasis on some words while the flat structure treats each word equally in AWR. For example, *the minister of the department of education* composed of *the minister* and *of the department of education* two regions should be labeled with PER but may be misclassified into ORG with AWR.

5 Conclusion

Leading region based approaches to nested NER face the efficiency and effectiveness challenges. We propose a hierarchical region framework to generate hierarchical regions and assign those regions with hierarchical representation an entity categorical label together. Experimental results demonstrate a significant improvement of our proposed framework in terms of efficiency and effectiveness than SoTA baselines. In future work, how to represent hierarchical regions better will be considered.

Acknowledgements

We sincerely thank all reviewers and AC for their comments and suggestions. This research work was funded by the National Natural Science Foundation of China under Grant No. 62072447.

References

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, NLPBA/BioNLP 2004, Geneva, Switzerland, August 28-29, 2004*.

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, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 141–150.

Joseph Fisher and Andreas Vlachos. 2019. Merge and label: A novel neural network architecture for nested NER. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5840–5850. Association for Computational Linguistics.

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, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1446–1459.

Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. GENIA corpus - a semantically annotated corpus for bio-textmining. In *Proceedings of the Eleventh International Conference on Intelligent Systems for Molecular Biology, June 29 - July 3, 2003, Brisbane, Australia*, pages 180–182.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 260–270.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019. A unified mrc framework for named entity recognition.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2019. Sequence-to-nuggets: Nested entity mention detection via anchor-region networks. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5182–5192.

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, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 857–867.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, System Demonstrations*, pages 55–60.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543.

Nicky Ringland, Xiang Dai, Ben Hachey, Sarvnaz Karimi, Cécile Paris, and James R. Curran. 2019. NNE: A dataset for nested named entity recognition in english newswire. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5176–5181.

Yikang Shen, Zhouhan Lin, Athul Paul Jacob, Alessandro Sordoni, Aaron C. Courville, and Yoshua Bengio. 2018. Straight to the tree: Constituency parsing with neural syntactic distance. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1171–1180.

Mohammad Golam Sohrab and Makoto Miwa. 2018. Deep exhaustive model for nested named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2843–2849.

Jana Straková, Milan Straka, and Jan Hajic. 2019. Neural architectures for nested NER through linearization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5326–5331.

Chuanqi Tan, Wei Qiu, Mosha Chen, Rui Wang, and Fei Huang. 2020. Boundary enhanced neural span classification for nested named entity recognition. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9016–9023. AAAI Press.

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, Brussels, Belgium, October 31 - November 4, 2018*, pages 204–214.

Congying Xia, Chenwei Zhang, Tao Yang, Yaliang Li, Nan Du, Xian Wu, Wei Fan, Fenglong Ma, and Philip S. Yu. 2019. Multi-grained named entity recognition. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1430–1440. Association for Computational Linguistics.

Jie Yang, Shuailong Liang, and Yue Zhang. 2018. Design challenges and misconceptions in neural sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3879–3889.

Changmeng Zheng, Yi Cai, Jingyun Xu, Ho-fung Leung, and Guandong Xu. 2019. A boundary-aware neural model for nested named entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 357–366.