Joint Training of POS Tagging and Dependency Parsing Models

Xuan-Dung Doan
Viettel Cyberspace Center, Viettel Group
Hanoi, Vietnam
dungdx4@viettel.com.vn

***Abstract*—We propose a joint model for POS tagging and dependency parsing. Our model consists of a BiLSTM-CNN-CRF-based POS tagger [27] and a Deep Biaffine Attention-based dependency parser [25]. A combined objective function is used to jointly train both models.**

***Keywords*-Dependency Parsing; Biaffine Attention; Joint training;**

I. Introduction

Dependency parsing is the task of automatically identifying binary grammatical relations between tokens in a sentence. There are two common approaches to dependency parsing: transition-based [12], [18], and graph-based [10], [19]. Recently, there has been a surge in the use of deep learning approaches to dependency parsing [5], [3], [9], [25], [26], [4], which help alleviate the need for hand-crafted features, take advantage of the vast amount of raw data through word embeddings, and achieve state-of-the-art results. Long Short-Term Memory networks (LSTM) [23] and attention mechanism [8], [24] are popular techniques used.

Among the deep learning approaches, joint multi-task models are gaining traction [29], [14], [6], [7], due to the intuition that information sharing among related tasks during training can help the model generalize better on each single task [16], [22]. In a joint multi-task model, the tasks can be ordered in a hierarchical structure, in which the output of one task is served as input to another task, or learned concurrently through multiple prediction heads in the last layer.

In this work, we present a joint part-of-speech (POS) tagging and dependency parsing model. We use a BiLSTM-CNN-CRF model [27] for our POS tagger, a graph-based Biaffine Attention model [25] for our dependency parser, and jointly train both models through a combined objective function. We evaluate our model on the VLSP 2019 dataset [1].

II. Related works

Several works have attempted to use the multitask learning paradigm to simultaneously solve several NLP tasks within a single model. With regards to dependency parsing, [29] is one of the earlier works to jointly learn POS tagging and dependency tagging for the Chinese language. The authors use learned POS tags as features for graph-based dependency parsers, and also to decide the spanning context in the decoding phase. For deep learning-related methods, [14] uses a cascaded model to jointly learn five different tasks, achieving state-of-the-art or competitive performances in all individual tasks. Their dependency parsing component takes as features the outputs of the lower-level chunking component, which in turn takes as features the outputs of the lower-level POS tagging component.

Recently, Nguyen et al. [6] propose a novel neural network model for joint POS tagging and graph-based dependency parsing. Their model uses bidirectional LSTMs to learn feature representations shared for both POS tagging and dependency parsing tasks. Experiments on 19 languages from the UD v1.2 treebanks show that their model obtains state-of-the-art results in both POS tagging and dependency parsing. After that, they [7] improve their model by extending the BIST graph-based dependency parser [9] by incorporating a BiLSTM-based tagging component to produce automatically predicted POS tags for the parser.

III. Methodology

In this section, we present each component of the joint model and the combined objective function. In our model, an input sentence of n words $w = w_1, w_2, ..., w_n$ is fed to each of the two component networks to learn separate token embeddings. We describe the learning process below.

Figure 1 illustrates the architecture of our model in detail.

A. Graph-based Dependency Parsing

Graph-based Dependency Parsing follows the common structured prediction paradigm [19], [1]:

$$predict(w) = \underset{y \in \mathcal{Y}(w)}{\operatorname{argmax}}\, score_{global}(w, y) \quad (1)$$

$$score_{global}(w, y) = \sum_{part \in y} score_{local}(w, part) \quad (2)$$

Given an input sentence w (and the corresponding sequence of the vectors $w_{1:n}$), we look the highest-score parse tree y in the space $\mathcal{Y}$(w) of valid dependency trees over w. In order to make the search tractable, the scoring function is decomposed to the sum of local scores for each part independently.

[1] http://vlsp.org.vn/vlsp2019/eval/udp

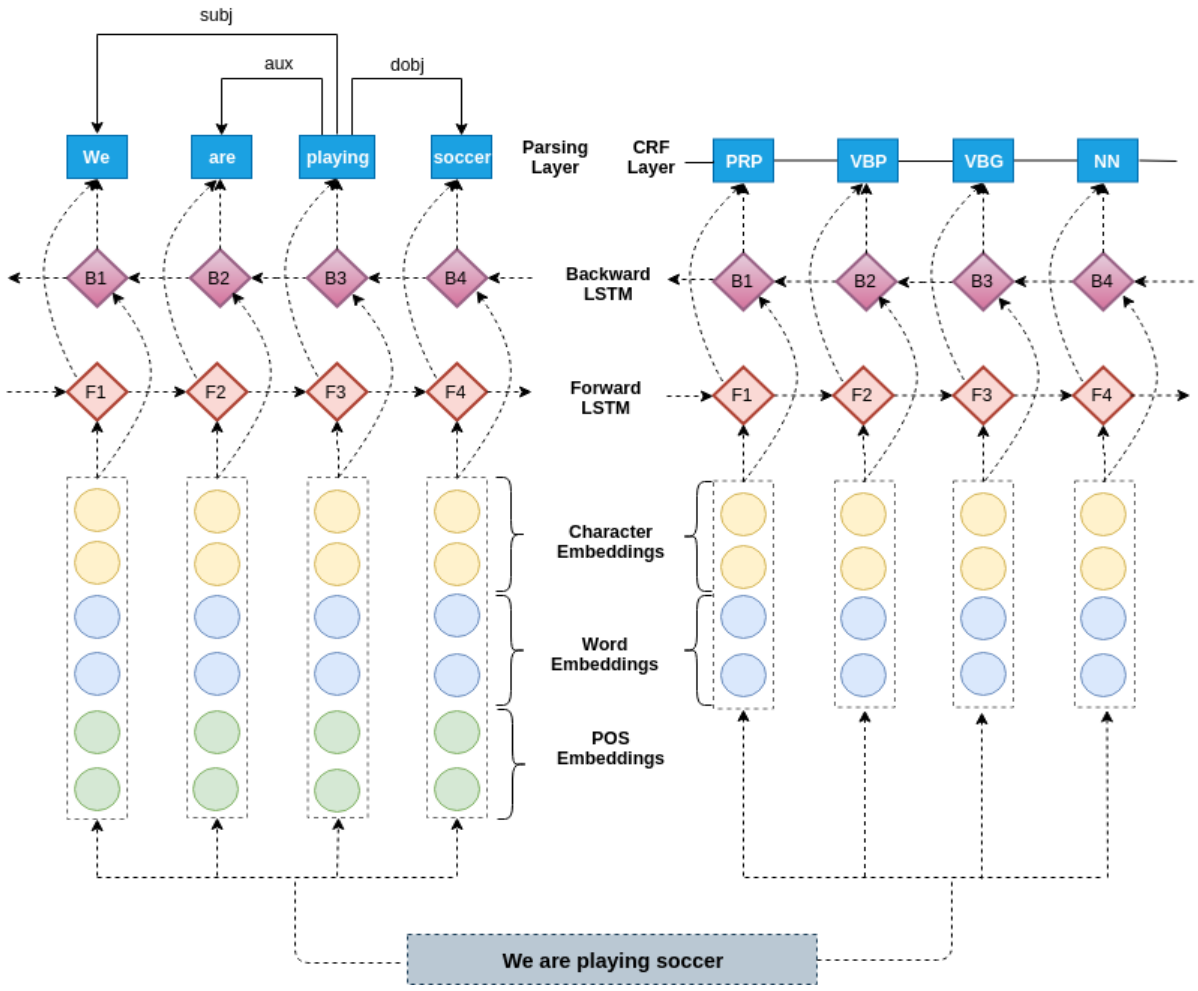

Figure 1. Illustration of our new model joint training Postag and Deep Biaffine Attention for Neural Dependency Parsing

B. Encoder

The encoder of our parsing model is based on the bi-directional LSTM-CNN architecture (BiLSTM-CNN) [27]. The CNN encodes character-level information of a word into its character-level representation, which is then concatenated with its corresponding word embedding and POS tag embedding before being fed to a BiLSTM layer. The BiLSTM is used to capture context information of each word. For the dependency parsing task, we also add POS tag embeddings to further enrich each word's representation. Finally, the encoder outputs a sequence of hidden states s_i.

C. Biaffine Attention Mechanism

We use the Biaffine attention mechanism described in [25] for our dependency parser. The task is posed as a classification problem, where given a dependent word, the goal is to predict the head word (or the incoming arc). Formally, let s_i and h_t be the BiLSTM output states for the dependent word and a candidate head word respectively, the score for the arc between s_i and h_t is calculated as:

$$e_i^t = h_t^T W s_i + U^T h_t + V^T s_i + b \tag{3}$$

Where W, U, V, b are parameters, denoting the weight matrix of the bi-linear term, the two weight vectors of the linear terms, and the bias vector. An additional multilayer perceptron (MLP) is added after the BiLSTM layer to reduce dimensionality and overfitting.

Similarly, the dependency label classifier also uses a biaffine function to score each label, given the head word vector h_t and child vector s_i as inputs. Again, we use MLPs to transform h_t and s_i before feeding them into the classifier.

D. Part-of-Speech Tagging

As proposed by [27], we use the BiLSTM-CNN-CRF architecture for our POS tagging model. The hidden states obtained from the encoder described in section B are fed through a Conditional Random Field (CRF) layer [13], [2] to predict the POS tags.

E. Joint Training of POS Tagging and Dependency Parsing

We implement joint training of both dependency parsing and POS tagging models by optimizing a combined objective the function of three individual losses: the POS tagging loss $\mathcal{L}_{POS}$, the arc classifying loss $\mathcal{L}_{arc}$, and the relation labeling loss $\mathcal{L}_{rel}$:

$$\mathcal{L} = \mathcal{L}_{POS} + \mathcal{L}_{arc} + \mathcal{L}_{rel} \quad (4)$$

F. Dependency Parsing Decoding

The decoding problem of this parsing model is solved by using the Maximum Spanning Tree (MST) algorithm [20].

IV. Experiments

A. Dataset

We use a dataset from VLSP 2019. The dataset is on news articles and literary text. The statistics of the dataset are summarized in Table I, II.

Table I
STATISTICS OF THE PUBLIC DATASET

Number of sentences	News articles	Literary text
train set	2920	936
test set	80	20

Table II
STATISTICS OF THE PRIVATE TEST SET

	News articles	Literary text	Social media
Number of sentences	400	101	100

B. Setup

We used Glove[2] pre-trained word embedding released by Stanford on 6.3G segment text. We adopt similar hyper-parameters as [25], [26] and [27]. Table III summarizes the hyper-parameters that we use in our experiments.

Table III
HYPER-PARAMETERS IN OUR EXPERIMENTS

	Layer	Hyper-Parameter	Value
Input	Word	dimension	128
	POS	dimension	100
	Char	dimension	64
LSTM	Encoder	encoder layer	3
		encoder size	512
	MLP	arc MLP size	512
		label MLP size	128
	Training	Dropout	0.33
		optimizer	Adam
		learning rate	0.001
		batch size	80

We create a train set contains news articles and literary text in public train set. We used k-fold Cross-Validation for the train set, where k is equal to 3, 5, and 7. Parsing performance is measured using UAS metric (Unlabeled Attachment Score) and LAS metric (Labeled Attachment Score) by comparing the gold relations of the test set and relations returned by the system. We used the evaluation script published at the CoNLL 2018 [3].

C. Main Results

The results on the private test set are shown in Table IV, V.

Table IV
THE RESULTS (UAS%/LAS%) ON THE DOMAIN OF THE PRIVATE TEST SET

	news articles	literary text	social media
k=3	68.07/55.69	81.36/73.55	67.22/55.52
k=5	68.21/55.95	81.73/72.91	65.65/54.12
k=7	68.93/56.61	81.55/72.55	66.47/54.86

Table V
THE RESULTS ON THE PRIVATE TEST SET

	UAS%	LAS%
k=3	69.18	57.28
k=5	69.17	57.29
k=7	**69.82**	**57.87**

Our model achieves the best performance on both UAS and LAS when k is equal to 7 on the private test set.

V. Conclusion

We present a joint POS tagging and graph-based dependency parsing model. In the future, we plan to evaluate our model on more datasets, and assess the viability of using shared embeddings and shared LSTM representations between the Biaffine Attention and BiLSTM-CNN-CRF models. We also look to improve our model by using self-attention approaches as proposed by [30], [31], [32].

References

[1] Benjamin Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. 2005. Learning structured prediction models: A large margin approach. In Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005, pages 896–903.

[2] Charles Sutton, Andrew McCallum. 2006. An introduction to conditional random fields for relational learning.

[3] Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In Proceedings of ACL-2015 (Volume 1: Long Papers). Beijing, China, pages 334–343.

[2] https://nlp.stanford.edu/projects/glove

[3] https://universaldependencies.org/conll18/conll18_ud_eval.py

[4] Daniel Fernández-González and Carlos Gómez-Rodríguez, Left-to-Right Dependency Parsing with Pointer Networks, to appear in Proc. of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2019), Minneapolis, USA, 2019.

[5] Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In Proceedings of EMNLP-2014. Doha, Qatar, pages 740–750.

[6] Dat Quoc Nguyen, Mark Dras and Mark Johnson. 2017. A Novel Neural Network Model for Joint POS Tagging and Graph-based Dependency Parsing. In Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies (CoNLL), pages 134-142.

[7] Dat Quoc Nguyen and Karin Verspoor. 2018. An improved neural network model for joint POS tagging and dependency parsing. In Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies (CoNLL), pages 81-91.

[8] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Proceedings of ICLR-2015.

[9] Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. Transactions of the Association for Computational Linguistics 4:313–327.

[10] Jason M. Eisner. 1996. Three New Probabilistic Models for Dependency Parsing: An Exploration. In Proceedings of COLING. pages 340–345.

[11] Joakim Nivre, Mitchell Abrams, et al. 2018. Universal Dependencies 2.2. http://hdl.handle.net/11234/12837.

[12] Joakim Nivre. An efficient algorithm for projective dependency parsing. In Proceedings of the 8th International Workshop on Parsing Technologies (IWPT), pages 149-160, 2003.

[13] John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of ICML-2001, volume 951, pages 282–289.

[14] Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A Joint Many-Task Model: Growing a Neural Network for Multiple NLP Tasks. The 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017).

[15] Kiet Van Nguyen, Ngan Luu Thuy Nguyen, Error Analysis for Vietnamese Dependency Parsing, The 7th International Conference on Knowledge and System Engineering(KSE), 10-2015, Hochiminh, Vietnam.

[16] Rich Caruana. 1997. Multitask Learning. Machine Learning 28(1): 41-75.

[17] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa. Natural Language Processing (Almost) from Scratch. Journal of Machine Learning Research, 12:2493-2537, 2011.

[18] Ryan McDonald and Fernando Pereira. 2006. Online Learning of Approximate Dependency Parsing Algorithms. In Proceedings of EACL. pages 81–88.

[19] Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In Proceedings of ACL. pages 91–98.

[20] Ryan McDonald, Fernando Pereira, Kiril Ribarov, andJan Hajic. 2005b. Non-projective dependency pars-ing using spanning tree algorithms. InProceedingsof HLT/EMNLP-2005. Vancouver, Canada, pages523–530.

[21] Ryan McDonald and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. Computational Linguistics 37(1):197–230.

[22] Sebastian Ruder. 2017. An Overview of Multi-Task Learning in Deep Neural Networks. arXiv preprint arXiv:1706.05098.

[23] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural Computation 9(8):1735–1780.

[24] Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In Proceedings of EMNLP-2015. Lisbon, Portugal, pages 1412–1421.

[25] Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In Proceedings of ICLR-2017 (Volume 1: Long Papers). Toulon, France.

[26] Xuezhe Ma, Zecong Hu, Jingzhou Liu, Nanyun Peng, Graham Neubig, and Eduard H. Hovy. 2018. Stack-pointer networks for dependency parsing. 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 1403–1414.

[27] Xuezhe Ma and Eduard Hovy. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016), pages 1064-1074, Berlin, Germany. August 2016.

[28] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, Lawrence D Jackel. 1989. Backpropagation applied to handwritten zip code recognition. Neural computation, pages 541-551.

[29] Zhenghua Li, Min Zhang, Wanxiang Che, Ting Liu, Wenliang Chen, Haizhou Li. 2011. Joint Models for Chinese POS Tagging and Dependency Parsing. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP-2011). 2011.07, pp. 1180-1191. Edinburgh, Scotland, UK.

[30] Wasi Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, Nanyun Peng, On Difficulties of Cross-Lingual Transfer with Order Differences: A Case Study on Dependency Parsing, NAACL 2019.

[31] Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. pages 55-64.

[32] Wenhui Wang, Baobao Chang, Mairgup Mansur. 2018. Improved Dependency Parsing using Implicit Word Connections Learned from Unlabeled Data. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pages 2857-2863.