ERMI at PARSEME Shared Task 2020: Embedding-Rich Multiword Expression Identification

Zeynep Yirmibeşoğlu and Tunga Güngör
Boğaziçi University
Department of Computer Engineering
34342 Bebek, Istanbul, Turkey
{zeynep.yirmibesoglu, gungort}@boun.edu.tr

Abstract

This paper describes the ERMI system submitted to the closed track of the PARSEME shared task 2020 on automatic identification of verbal multiword expressions (VMWEs). ERMI is an embedding-rich bidirectional LSTM-CRF model, which takes into account the embeddings of the word, its POS tag, dependency relation, and its head word. The results are reported for 14 languages, where the system is ranked 1st in the general cross-lingual ranking of the closed track systems, according to the Unseen MWE-based F_1.

1 Introduction

Multiword expressions (MWEs) are lexical items that consist of multiple lexemes. The challenge of identifying MWEs comes from the fact that their properties cannot directly be deducted from the lexical, syntactic, semantic, pragmatic, and statistical properties of their components (Baldwin and Kim, 2010). Addressing this challenge, the PARSEME shared task 2020 is a campaign that encourages the development of automatic verbal MWE (VMWE) identification models in a multilingual context. In this third edition of the PARSEME shared task, the focus is on identifying VMWEs that are unseen in training data. For this task, dev, test, train and raw corpora have been provided for 14 languages.

ERMI (Embedding-Rich Multiword expression Identification) is a multilingual system with a bidirectional LSTM-CRF architecture, which can take as input the embeddings of the word, its POS tag, dependency relation, and its head word. Since the main focus of the shared task is to identify unseen VMWEs, we experiment with how the addition of the head word embedding affects the prediction results for different languages. In addition, we also take advantage of the raw corpora in a semi-supervised teacher-student neural model carrying the same LSTM-CRF architecture for two languages (EL, TR). We use no external resources in the training of our system, thus participating in the closed track.

The results for all 14 languages in the closed track have been submitted where language-specific combinations of the above-mentioned embeddings have been used as input to the system. The system has been ranked 1st in the general cross-lingual ranking of the closed track systems for the Unseen MWE-based F_1, and 2nd for the Global MWE-based and Global Token-based F_1 metrics.

2 System Description

Named entity recognition (NER) and MWE detection can be considered similar tasks, thus encouraging similar architectures. Neural models have been preferred frequently for NER (Lample et al., 2016; Güngör et al., 2019), and for detecting VMWEs in the previous edition of PARSEME (Ehren et al., 2018; Boros and Burtica, 2018; Berk et al., 2018; Taslimipoor and Rohanian, 2018; Stodden et al., 2018; Zampieri et al., 2018).

In order to detect VMWEs, we develop a system[1] consisting of three (two supervised, one semi-supervised) neural network models, all of which carrying the same, bidirectional LSTM-CRF architecture, as proposed by Huang et al. (2015) for sequence tagging tasks. All models consist of three layers:

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by/4.0/.

[1] ERMI is freely available at https://github.com/zeynepyirmibes/ERMI

Joint Workshop on Multiword Expressions and Electronic Lexicons, pages 130–135
Barcelona, Spain (Online), December 13, 2020.

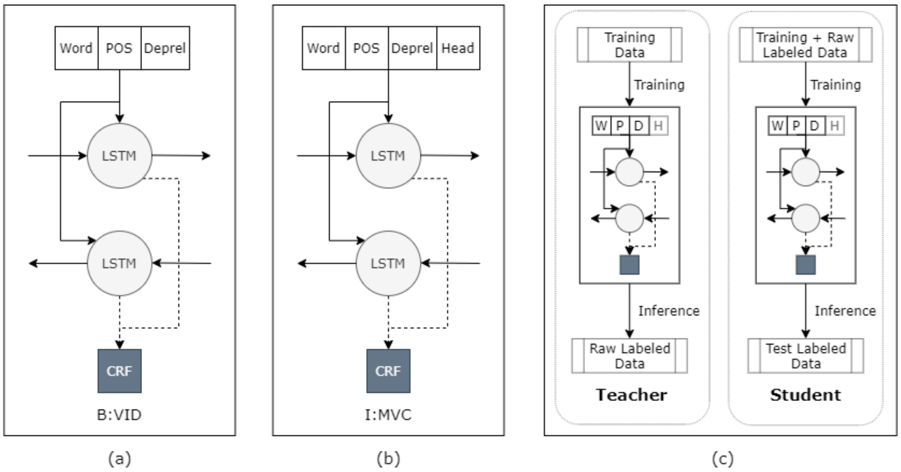

Figure 1: Our bi-LSTM-CRF models (**a**) ERMI; (**b**) ERMI-head; and (**c**) TeachERMI. For TeachERMI, the input layer may or may not contain the head word embedding, decided according to validation results per language.

the input layer, LSTM layer, and CRF layer, implemented using Keras (Chollet and others, 2015) with Tensorflow backend (Abadi et al., 2015). The architecture of each model is shown in Figure 1.

The input of our neural networks is an embedding layer, where we provide the model with the concatenation of the embeddings of the word, its POS tag, its dependency relation to the head, and the head of the word (for some languages). We do not use a pre-trained word embedding model. Instead, we exploit the provided raw corpora [2], which are gathered specifically for this task and are in the same domain as the annotated corpora[3]; and train FastText word embedding models for each of the 14 languages separately, using Gensim's Fasttext implementation (Řehůřek and Sojka, 2010). The embedding vector dimension for all languages is 300 (for word and head word embeddings), whereas the vocabulary size of the embedding models vary due to different sizes of raw corpora. Due to computational limitations, we only use a portion of the raw corpora for FR, PL and SV languages.

2.1 Supervised ERMI

We develop two supervised neural models (ERMI, and ERMI-head) differing only in the input layer. For the input (embedding) layer, word and head word embeddings each of dimension 300 are extracted from the Fasttext embedding models that we pretrained from the raw corpora. Dependency relation and POS tag embeddings are represented as one-hot encodings, and then converted into embeddings during training. Hence, the dimension of the dependency relation embedding for each language is the number of unique DEPREL tags encountered in the training data plus one, for unknown tags in the test data. The same logic holds for the POS tag embedding.

For our basic ERMI model, we use as input the concatenation of the embeddings of the word (CoNLL-U's FORM), its POS tag (UPOS), and its dependency relation to the head word (DEPREL). For our second supervised model, ERMI-head, we also concatenate the embedding of the head of the word (CoNLL-U's HEAD) to the input layer, in order to incorporate the relationship the word has to its dependent word, which, as we observe for some languages (EU, FR, HE, HI, PL, TR), aids in the decision of whether a word is to be annotated as part of a VMWE.

Differing only in the input layer, both models pass the input features to the bidirectional LSTM layer, where past (via forward LSTM states) and future (via backward LSTM states) information are taken into account. The output of the LSTM layer is then passed to the CRF layer, which connects consecutive output layers to produce the final output. With this approach, we incorporate both past and future information using the bi-LSTM architecture, and also the sentence level tag information using the CRF

[2] http://hdl.handle.net/11234/1-3416

[3] http://hdl.handle.net/11234/1-3367

layer.

2.2 Semi-supervised ERMI

In the third edition of PARSEME, raw (unlabeled) corpora are provided for all languages, thus enabling the possibility of semi-supervised learning. Hence, we exploit a portion of the raw corpus in addition to the annotated training corpus, and propose a teacher-student model (TeachERMI). The aim is to be able to also train on unlabeled data, as suggested by Wu et al. (2020), where they train a teacher-student cross-lingual Named Entity Recognition (NER) model.

In this approach, we first train a teacher model for every language separately, on the labeled training set. The teacher model is one of ERMI, or ERMI-head, depending on the validation results per language. Afterwards, we take a portion of the unlabeled raw corpus (corresponding to the half of the size of the training corpus for that language), and label it using the teacher model that we trained. Then, we combine the annotated training corpus with the raw corpus labeled by the teacher model, and train a student model. We observe that this approach only performs better than the teacher model (ERMI or ERMI-head) for Greek (EL) and Turkish (TR). Thus, we employ this approach (TeachERMI) for only two languages.

3 Experimental Setup

Tagging Scheme: During pre-processing, we adopt the bigappy-unicrossy tagging scheme proposed by Berk et al. (2019) to better represent overlapping (nesting and crossing) and discontinuous MWEs.

Datasets: During the validation runs (results of which are explained in Section 4.1), we concatenate the training and development corpora for each language, and randomly split 90% for training and 10% for testing. For the teacher-student model, we also use a portion of the raw corpora (roughly half the size of the training sets). After selecting the best system (out of ERMI, ERMI-head, and TeachERMI) for each language, we train our final models using the combined training and development sets, and use the blind test data for testing. For Turkish (TR) and Greek (EL), we develop a teacher-student model, using 10,796 and 9,510 sentences, respectively, of the provided raw corpora in addition to the development and training sets.

Hyperparameters: We choose the mini batch size and number of epochs with respect to the size of training sets for each language (ref. Table 1). We limit the mini batch size between 8-32, drawn from the conclusions of Reimers and Gurevych (2017), where they experiment with five sequence tagging tasks with LSTM architectures, and deduct the optimal mini batch size for large training corpora. We use a fixed dropout rate of 0.1 for all bi-LSTM layers.

4 Results and Discussion

We make validation runs on the training and development data, and compare our three neural models for each language. Afterwards, we report the official results of the selected systems on the blind test set.

4.1 Validation Results

The validation results of our three systems (ERMI, ERMI-head, TeachERMI) are compared for all languages, and the best-performing system (with respect to Unseen MWE-based, Global MWE-based, and Global Token-based F_1) for each language is selected for the final submission. In Table 1, we report the validation results together with the hyperparameters used during training.

For us, the most interesting part of evaluating the validation runs is the comparison between ERMI and ERMI-head. We observe that the addition of head word embeddings to the input layer improves the Unseen MWE-based F_1 score significantly for the EU, FR, HE, HI, PL and TR languages (4.98% on average for these languages). We also have the opportunity to observe that the teacher-student model enables the enlargement of the training corpus by around 50%, thus enabling better generalization for EL and TR.

	System	Batch Size	Epochs	Unseen MWE-based F_1	Global MWE-based F_1	Global Token-based F_1		System	Batch Size	Epochs	Unseen MWE-based F_1	Global MWE-based F_1	Global Token-based F_1
DE	**ERMI**	16	20	**21.93**	**52.59**	**57.51**	IT	**ERMI**	16	15	17.18	**43.12**	**44.73**
	ERMI-head			20.49	47.27	55.13		ERMI-head			7.74	31.63	33.96
	TeachERMI			21.89	48.70	53.71		TeachERMI			**18.40**	38.40	40.35
EL	ERMI	32	20	30.77	60.07	**66.02**	PL	ERMI	32	20	25.45	69.94	72.13
	ERMI-head			30.51	55.13	59.76		**ERMI-head**			**32.03**	**70.85**	**72.23**
	TeachERMI			**33.33**	**60.11**	64.82		TeachERMI			22.97	63.61	64.57
EU	ERMI	16	15	39.39	77.55	80.12	PT	**ERMI**	32	20	22.64	**57.89**	**58.23**
	ERMI-head			**42.42**	**77.81**	**80.50**		ERMI-head			**25.59**	53.57	55.18
	TeachERMI			28.57	69.16	72.01		TeachERMI			21.05	54.12	54.16
FR	ERMI	32	20	24.83	59.62	**66.31**	RO	**ERMI**	16	15	24.24	**82.88**	**84.38**
	ERMI-head			**27.18**	**62.75**	66.15		ERMI-head			**28.57**	80.57	81.20
	TeachERMI			20.39	57.39	63.10		TeachERMI			27.91	81.85	83.59
GA	**ERMI**	8	12	**4.82**	**9.80**	**27.23**	SV	**ERMI**	8	12	**30.23**	**60.16**	**60.41**
	ERMI-head			2.27	1.92	23.08		ERMI-head			25.00	55.85	58.26
	TeachERMI			~0	~0	~0		TeachERMI			27.50	51.33	50.28
HE	ERMI	32	20	10.67	**27.87**	31.02	TR	ERMI	32	20	42.86	64.95	64.92
	ERMI-head			**14.18**	27.19	**34.65**		ERMI-head			45.71	65.98	66.50
	TeachERMI			7.21	13.04	11.76		**TeachERMI**			**52.25**	**67.32**	**68.72**
HI	ERMI	8	12	42.11	54.84	69.35	ZH	**ERMI**	32	20	**42.99**	**62.39**	**66.67**
	ERMI-head			**53.66**	**63.64**	**74.63**		ERMI-head			39.30	59.43	63.83
	TeachERMI			35.90	53.12	70.97		TeachERMI			39.44	59.61	61.96

Table 1: Validation results and hyperparameters of our three models for each language.

		Unseen MWE-based				Global MWE-based				Global Token-based			
Language	System	P	R	F1	Rank	P	R	F1	Rank	P	R	F1	Rank
DE	ERMI	24.02	20.27	21.98	1	63.23	44.66	52.35	2	76.14	42.66	54.68	2
EL	TeachERMI	28.70	31.00	29.81	1	67.20	56.16	61.19	2	75.19	58.82	66.00	2
EU	ERMI-head	21.13	37.33	26.99	1	75.95	70.35	73.04	2	80.10	72.25	75.97	2
FR	ERMI-head	18.54	35.67	24.40	1	61.52	61.30	61.41	2	70.86	65.53	68.09	2
GA	ERMI	14.79	6.98	9.48	1	32.62	13.99	19.58	2	69.71	21.15	32.45	1
HE	ERMI-head	11.49	6.62	8.40	1	41.81	24.85	31.17	2	46.72	26.22	33.59	2
HI	ERMI-head	37.09	41.67	39.25	1	63.48	56.32	59.69	1	79.48	62.00	69.66	1
IT	ERMI	17.44	10.00	12.71	1	66.27	32.75	43.84	2	75.45	32.55	45.48	2
PL	ERMI-head	23.28	29.24	25.92	1	73.92	64.91	69.12	2	77.87	65.86	71.36	2
PT	ERMI	24.63	33.33	28.33	1	68.84	59.46	63.81	2	73.62	58.80	65.38	2
RO	ERMI	16.45	30.10	21.28	1	85.67	81.57	83.57	1	88.69	82.97	85.74	1
SV	ERMI	31.16	28.67	29.86	1	72.68	55.73	63.08	2	77.24	52.53	62.53	2
TR	TeachERMI	37.28	35.67	36.46	1	67.11	61.86	64.38	1	69.11	62.42	65.60	1
ZH	ERMI	47.49	34.67	40.08	1	66.67	55.98	60.86	1	70.92	58.99	64.41	1
Total		25.25	27.23	26.20	1	64.78	52.85	58.21	2	73.65	54.48	62.63	2

Table 2: Official Language-specific Results of ERMI

4.2 Test Results

We evaluate the validation runs (Table 1), and train the ERMI system for DE, GA, IT, PT, RO, SV and ZH languages; the ERMI-head system for EU, FR, HE, HI, and PL languages. For Turkish (TR), we train TeachERMI using the ERMI-head input layer (including the head word embedding), and for Greek (EL), we train TeachERMI using the ERMI input layer (excluding the head word embedding), judging from these languages' validation results. Having selected the most appropriate system for each language, we present the official results in the closed track for all 14 languages on the blind test data in Table 2.

4.3 Discussion

We have been able to observe from the validation results that the addition of head word embeddings to the input layer significantly aided in detecting unseen VMWEs for EU, FR, HE, HI, PL and TR. In order to observe the effect of head word embeddings on VMWE detection in the final test set, we removed the head word embeddings from the input layer for one of those languages (EU), and obtained a 24.64% Unseen MWE-based F1 score from the ERMI model, as compared to the 26.99% that we've obtained in the official results with ERMI-head.

For DE, GA, IT, PT, RO, SV and ZH, our ERMI model (without head word embeddings in the input layer) performed better than ERMI-head and TeachERMI during the validation runs. To examine this phenomenon in the blind test set, we also trained the ERMI-head system for one of those languages (IT). The 43.84% Global MWE-based F1 score of ERMI for IT drops to 36.88% when head-word embeddings are added to the input layer.

Analyzing the presence and absence of head-word embeddings in the embedding layer for each language, we deduct that feeding a language-specific input layer to the neural models increased our overall performance. Using also the raw corpora for EL and TR languages with the teacher-student model, we have been able to benefit from training on unlabeled data, which may be preferable for low resource scenarios. For TR, the validation results show the superiority of ERMI-head over ERMI, and of TeachERMI over ERMI-head. Hence, the final system for Turkish is TeachERMI with the ERMI-head input layer. We also run ERMI-head for the final test set, where we obtain a Global MWE-based F1 score of %63.47, whereas the official score of TeachERMI for TR is %64.38, showing us the benefit of using the teacher-student model for this language.

When we look at the performance of our system for the MWE-based F1 score per VMWE category, we can see that our system outperforms the other closed track system in the LVC.full category for HI, TR and ZH, and is ranked 2nd among all seven (open and closed track) systems for HI. Our system also predicts MVCs better than other systems that submitted their results for IT and PT.

Our overall system ranked 1st among 2 systems in the closed track, and 3rd among 9 systems in both open and closed tracks with respect to Unseen MWE-based F1, which was the focus of this edition of PARSEME. It is worth noting that, although we did not make use of any external resources (participating in the closed track), we outperformed most of the systems in the open track that exploit such resources. Our system also ranked 1st in the closed track for the HI, RO, TR and ZH languages in the Global MWE-based F1 metric and 5th for all 14 languages among all systems in the Global MWE-based and Token-based F1 metric.

5 Conclusion

In this paper we proposed an embedding-rich bidirectional LSTM-CRF system. In addition to word, POS and dependency relation embeddings, we exploited head word embeddings, especially to tackle the issue of predicting unseen VMWEs. Within the closed track, we have used the raw corpora to train word embeddings, as well as proposing a semi-supervised teacher-student model, providing the opportunity of training on unlabeled data for VMWE identification. These methods have increased the generalisation power, enabling our system to perform best in predicting unseen VMWEs in the closed track.

Acknowledgements

The numerical calculations reported in this paper were partially performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources).

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Timothy Baldwin and Su Nam Kim. 2010. Multiword expressions. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*, pages 267–292. CRC Press, Taylor and Francis Group, Boca Raton, FL. ISBN 978-1420085921.

Gözde Berk, Berna Erden, and Tunga Güngör. 2018. Deep-BGT at PARSEME shared task 2018: Bidirectional LSTM-CRF model for verbal multiword expression identification. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 248–253, Santa Fe, New Mexico, USA, August. ACL.

Gözde Berk, Berna Erden, and Tunga Güngör. 2019. Representing overlaps in sequence labeling tasks with a novel tagging scheme: bigappy-unicrossy. In Alexander Gelbukh, editor, *20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, La Rochelle, France.

Tiberiu Boros and Ruxandra Burtica. 2018. GBD-NER at PARSEME shared task 2018: Multi-word expression detection using bidirectional long-short-term memory networks and graph-based decoding. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 254–260, Santa Fe, New Mexico, USA, August. ACL.

François Chollet et al. 2015. Keras. `https://keras.io`.

Rafael Ehren, Timm Lichte, and Younes Samih. 2018. Mumpitz at PARSEME shared task 2018: A bidirectional LSTM for the identification of verbal multiword expressions. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 261–267, Santa Fe, New Mexico, USA, August. ACL.

Onur Güngör, Tunga Gungor, and Suzan Uskudarli. 2019. The effect of morphology in named entity recognition with sequence tagging. *Natural Language Engineering*, 25:147–169, 01.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

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, San Diego, California, June. ACL.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark, September. ACL.

Regina Stodden, Behrang QasemiZadeh, and Laura Kallmeyer. 2018. TRAPACC and TRAPACCS at PARSEME shared task 2018: Neural transition tagging of verbal multiword expressions. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 268–274, Santa Fe, New Mexico, USA, August. ACL.

Shiva Taslimipoor and Omid Rohanian. 2018. SHOMA at parseme shared task on automatic identification of VMWEs: Neural multiword expression tagging with high generalisation. *arXiv preprint arXiv:1809.03056*.

Qianhui Wu, Zijia Lin, Börje F Karlsson, Jian-Guang Lou, and Biqing Huang. 2020. Single-/multi-source cross-lingual NER via teacher-student learning on unlabeled data in target language. *arXiv preprint arXiv:2004.12440*.

Nicolas Zampieri, Manon Scholivet, Carlos Ramisch, and Benoit Favre. 2018. Veyn at PARSEME shared task 2018: Recurrent neural networks for VMWE identification. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 290–296, Santa Fe, New Mexico, USA, August. ACL.