NTCIR 17 Conference: Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies, December 12-15, 2023, Tokyo, Japan
DOI: https://doi.org/10.20736/0002001294

VLP Methods at the MedNLP-SC Social Media Adverse Drug Event Detection of NTCIR-17

The-Quyen Ngo, Duy-Dao Do, Phuong Le-Hong*
Vietnam National University, Hanoi
Hanoi, Vietnam
(ngoquyenbg|doduydao|phuonglh)@hus.edu.vn

ABSTRACT

The VLP team participated in the MedNLP-SC subtask of the NTCIR-17. This paper reports our approach to solving the problem and discusses our experimental and official results. We present approaches which combine the training datasets using different methods, either vertically or horizontally across the languages. We use different text representation methods, either using a continuous embedding vector generated by a large pretrained language model or a discrete count vector generated by a simple bag-of-word method. Our proposed approaches achieve good performance – our system is ranked in the top two or three of the best performing systems for the task.

KEYWORDS

adverse drug detection, social text, multilingual text processing

TEAM NAME

VLP

SUBTASKS

SM-ADE-EN, SM-ADE-FR, SM-ADE-DE, and SM-ADE-JA

1 INTRODUCTION

Medical Natural Language Processing for Social media and Clinical texts (MedNLP-SC) is one of the core tasks in NTCIR-17 for medical natural language processing using clinical texts written by physicians (radiology reports) or social media data. The goal of this shared task is to promote the development of practical systems that support various medical services. The MedNLP-SC task has two corpus-based subtasks: (1) Social Media Subtask which focuses on Adverse Drug Event detection for social media texts in Japanese, English, German, and French and (2) Radiology Report Subtask which focuses on TNM staging for radiology reports in Japanese [18]. The VLP team participated in the first subtask. This paper reports our approach to solving the problem and discusses the official results.

2 RELATED WORK

The problem of the first subtask is essentially concerned with the problem of multi-label text classification, which has been of interest to researchers for many years. Multi-label text classification is applied in many practical applications, such as customer reviews of suppliers' services and products on e-commerce platforms, or movie reviews. Evaluations of health services, or health-related questions, can also be multi-labeled.

Previously, muti-label classification (MLC) was often approached based on existing approaches to the single-label classification problem [23]. Binary relevance algorithm was introduced by [1], in which the labels are considered independent of each other, and there will be a binary classifier built for each label. The classifier chains method was presented in [15], this method can model the correlation between labels. By mapping the label set to natural numbers, Tsoumakas [17] transforms MLC to n multi-class classifiers. Some popular machine learning methods for classification problems such as k-nearest neighbor, support vector machine, conditional random field or decision tree are also modified to apply to MLC [3, 8, 9, 22]. Approaches based on multi-class classification often have difficulty when applied to multi-label classification problems in cases where the number of labels is large and there are relationships between labels. Because then the number of classifiers will be large, or the label mapping set will be very large, leading to data shortage as well as data imbalance.

There have been a number of effective approaches for multi-label classification problems such as tree-base, embedding-base, deep learning, etc. also proposed by researchers. CRAFTML was proposed by [16], an algorithm based on decision tree combined with a very fast partitioning approach. This system outperforms other tree-based systems on nine large multi-label datasets. By combining tree-base and deep learning, Ronghui [20] proposed AttentionXML which is a label tree-based deep learning, which included a multi-label attention mechanism for raw text input and a shallow and wide probabilistic label tree to handle a large number of labels. AttentionXML shows good results on eight benchmark datasets, including datasets with a very large number of labels such as Amazon-3M, with three million labels. A commonly used method to represent textual contexts is pre-trained models. APLC-XLNet was introduced by [19], which is a fine-tuned deep learning model based on XLNet to learn the representation of an input text and Adaptive Probabilistic Label Clusters for label clustering. The BERT model is applied very effectively in many problems. In many recent studies, the authors have used transformer-based models for MLC. Zhang [21] proposed XR-Transformer, a novel recursive approach to increase speed of fine-tuning transformer models while also improving model performance.

Research related to biomedical data always attracts the attention of the community, and there have also been many studies related to multi-label classification on biomedical data. Chen [2] introduced LITMC-BERT, a method based on the transformer for multi-label biomedical data. LITMC-BERT used BioBERT [12] to create representations of the input text and a mechanism to share transformer backbone between labels. Pan [14] proposed FAMLC-BERT. ML-Net [7] is an end-to-end model for multi-label biomedical text data.

Related studies have shown that approaches based on transformers with pre-trained models have very good results with multi-label classification problems on biomedical text data. Our proposed methods in this research are an extension of this line of research.

3 METHODS

This section describes our approach to solving the problem. Given a parallel input text in four languages, say x_e, x_f, x_g, x_j for English, French, German and Japanese respectively[1], we experiment with two data combination methods which results in two multilingual training corpora as follows:

(1) In the first method, we combine vertically all the samples in the four languages into a single corpus. Here, we use the term "vertically" to signify the union or concatenation of sample texts – for each language, there are about 9,957 texts; there are thus $4 \times 9,957$ samples in the vertically combined corpus, each text is a monolingual sample.
(2) In the second method, we combine horizontally all the parallel samples in the four languages into a single corpus. There are also about 9,957 texts in this single corpus, each text has four fields $x = [x_e, x_f, x_g, x_j]$ and can be seen as a multilingual sample.

We experiment with three different featurization methods:

(1) In the first method, we use a multilingual *large pretrained language model* (LPLM) to produce an embedding vector for each sample in the corpus. The chosen LPLM is XLM-R [5] which has the best results on a development dataset.[2] For a sample text x in the vertical corpus, the LPLM produces an embedding vector $\vec{v}(x) \in \mathbb{R}^d$. For a sample text x in the horizontal corpus, the LPLM produces four embedding vectors $[\vec{v}(x_e), \vec{v}(x_f), \vec{v}(x_g), \vec{v}(x_j)]$ and these vectors are combined in two ways, they are either (1) averaged to produce an embedding vector $\vec{v}(x) = \frac{1}{4}\sum \vec{v}(x.) \in \mathbb{R}^d$, or (2) concatenated to produce an embedding vector $\vec{v}(x) = \vec{v}(x_e) \oplus \vec{v}(x_f) \oplus \vec{v}(x_g) \oplus \vec{v}(x_j) \in \mathbb{R}^{4d}$. This method produces a continuous vector representation for each input text.
(2) In the second method, we use a traditional bag-of-word featurization technique to produce discrete count vectors for each language and concatenate these vectors together. Stopwords are first removed from each input text, then spaced-based tokenization is performed. Each token array is then vectorized by a count vectorizer. Three vectors corresponding to three monolingual samples obtained by this same preprocessing pipeline are then concatenated.[3] This method produces a discrete vector representation for each input text in the horizontal fashion.
(3) In the third method, we concatenate both continuous and discrete representations above to produce a mixed vector for each input text.

As described by the organizers of the task, the samples are first manually classified into two classes: those containing Adverse Drug Events (ADEs) versus those not containing ADEs. Then annotators labeled the occurring symptoms in the tweets containing ADEs with 22 most frequent symptoms. This is thus a multilabel classification problem where each sample is labeled with zero, one or more categories. We use a simple feed-forward neural network with one hidden unit for this task: an input feature vector $\vec{v}(x)$ is fed to a hidden layer of h dimensions using the ReLu activation function, and then the result is passed to the output layer of 22 dimensions (the number of labels) using the sigmoid activation function. At each output dimension, if the output value is not less than a predefined threshold τ then the corresponding label is selected. The best hyperparameters h and τ are tuned on a validation dataset.

[1]This is the same tweet which is translated into different languages.
[2]In addition to XLM-R, we experiment with other LPLMs, as described in Section 4.
[3]We don't use the Japanese dataset in this method to avoid having to use a specific tokenization method for this language which does not rely on the space character like in the other occidental languages.

4 EXPERIMENTS

This section discusses our experimental results and the official results of our proposed methods for the task on the test set as announced by the organizers.

4.1 Pretrained Language Models

We carried out experiments with a number of LPLMs including mBERT, RoBERTa, DeBERTa and XLM-R. We give a brief overview of these language models as follows:

mBERT is a multilingual version of BERT which follows the BERT recipe with the same training architecture and objective [6]. The main difference is the training set where it is trained on 104 languages from the Wikipedia corpus. mBERT has an encoder-only architecture with 180M parameters.

RoBERTa – Robustly Optimized BERT Pretraining Approach is built on BERT and offers better training setup [13]. Some modifications include changing key hyperparameters, removing the next-sentence pretraining objective and training with much larger mini-batches and learning rates. It uses a byte-level BPE (Byte-Pair Encoding) as subtoken and not characters for handling unicode characters. RoBERTa outperformed BERT in a number of standard NLP benchmarks.

DeBERTa – Decoding-enhanced BERT with disentangled attention improves the BERT and RoBERTa models using two novel techniques [11]. The first is the disentangled attention mechanism, where each word is represented using two vectors that encode its content and position, respectively, and the attention weights among words are computed using disentangled matrices on their contents and relative positions. Second, an enhanced mask decoder is used to replace the output softmax layer to predict the masked tokens for model pretraining. These two techniques significantly improve the efficiency of model pre-training and performance of downstream tasks. The latest version is DeBERTaV3 which improves the original DeBERTa model by replacing masked language modeling with replaced token detection [10]. This pretraining approach was proposed by ELECTRA [4] which is more efficient. DeBERTaV3 proposes the addition of a novel gradient disentangled embedding layer which helps improve both training efficiency and the quality of the pretrained model.

XLM-R is XLM-RoBERTa, a model based on RoBERTa. It is a large multilingual language model, trained on 2.5TB of

filtered CommonCrawl data of 100 different languages [5]. It has been shown to outperform mBERT on a variety of cross-lingual benchmarks. XLM-R performs particularly well on low-resource languages. This model does not require *lang* tensors to understand which language is used, and should be able to determine the correct language from the input. In addition, it does not use the translation language modeling objective as in some other XLM multilingual models; it only uses masked language modeling one sentence coming from one language.

Our extensive experiments showed that the XLM-R model gives the best performance compared to these other models.

4.2 Experimental Results

As described in the previous section, in the vertical data combination method, we combine all the data in four languages into one large corpus of about 28,670 samples. This corpus is randomly divided into the training set, validation set and test set with ratios 80%, 10% and 10% respectively. More precisely, the training data has 25,484 samples, the test and validation data have 3,186 samples. The data is trained using the XLM-R (large) pretrained model. This approach uses the first featurization method as described in the previous section.

Some hyper-parameters of our models are as follows. The batch size is 128, the learning rate is 2×10^{-5}, the weight decay is 0.01 and the model is trained in 50 epochs. The embeddings layer is fine-tuned along with the model.

Tables 1, 2, 3 show the results of this method on the test set; all the scores are obtained by the evaluation script provided by the organizers.[4]

Table 1: **Binary scores (ADE vs. no ADE) of the vertical method**

	precision	recall	f1-score	support
no ADE	0.99	0.98	0.98	2229
ADE	0.95	0.97	0.96	957
accuracy			0.98	3186
macro avg	0.97	0.98	0.97	3186
weighted avg	0.98	0.98	0.98	3186

Table 2: **(Individual) Per Class Scores of the vertical method**

label	precision	recall	f1-score	support
C0027497	0.96	0.98	0.97	320
C0011991	0.94	0.96	0.95	196
C0015672	0.97	0.99	0.98	92
C0042963	0.94	0.99	0.96	82

Continued on next page

Table 2: **(Individual) Per Class Scores of the vertical method** (Continued)

label	precision	recall	f1-score	support
C0003123	0.98	0.99	0.98	96
C0018681	0.98	1.00	0.99	102
C0015967	0.94	0.97	0.95	65
C0206062	1.00	1.00	1.00	4
C0023895	1.00	0.94	0.97	17
C0012833	0.92	1.00	0.96	48
C0030193	0.94	0.90	0.92	114
C0002170	0.93	1.00	0.96	27
C0004096	0.98	1.00	0.99	42
C0022658	0.86	1.00	0.92	6
C0020517	0.91	0.96	0.93	67
C0917801	0.92	0.97	0.95	37
C0009806	0.91	0.95	0.93	22
C0005956	1.00	0.62	0.77	8
C0000737	0.92	0.96	0.94	136
C0010692	0.50	1.00	0.67	2
C0015230	0.95	0.95	0.95	39
C0149745	0.79	1.00	0.88	15
micro avg	0.95	0.97	0.96	1537
macro avg	0.92	0.96	0.93	1537
weighted avg	0.95	0.97	0.96	1537
samples avg	0.29	0.29	0.29	1537

Table 3: **(Full) Per Label Scores of the vertical method**

	precision	recall	f1-score	support
0	1.00	1.00	1.00	68555
1	0.95	0.97	0.96	1537
accuracy			1.00	70092
macro avg	0.97	0.98	0.98	70092
weighted avg	1.00	1.00	1.00	70092
Exact accuracy		0.9698		

It is observed that this method gives quite a good performance. Most of the labels have an f1 score greater than 0.90. The rare labels with low support values have mediocre scores: C0010692 (*hemorrhagic cystitis*) of 0.67 f1, C0005956 (*bone marrow dysfunction*) of 0.77 f1. The exact accuracy of this simple method on our test set is also is quite high, of 96.98%.

[4] https://github.com/lraithel/ntcir_2023_SMA

4.3 Official Results

We use all the provided training data to train models and perform inference on the official test set for official submissions. In general, the vertical data combination approach gives better results than those of the horizontal approach.

Table 4: Binary Scores (ADE vs. no ADE) of the vertical approach using the XLM-R model.

	ja	fr	en	de
f1-score no ADE	0.92	0.91	0.91	0.91
f1-score ADE	0.80	0.79	0.79	0.78
accuracy	0.88	0.88	0.88	0.87

Using the same XLM-R language model, the horizontal data combination approach has an f1-score no-ADE of 0.84, an f1-score ADE of 0.67 and an accuracy of 0.79 for all the four languages.[5]

Table 5 shows the individual scores of our approaches. The first four columns are test results of the vertical system on the four corresponding languages. The last column contains scores of the horizontal system. However, by mistake, we used the universal pretrained embeddings for the horizontal system rather than the intended XLM-R pretrained embeddings. Therefore, these results are not comparable.

Table 5: **(Individual) Per Class Scores of our approaches**

label	ja	fr	en	de	horiz.
C0027497	0.9	0.88	0.87	0.87	0.77
C0011991	0.84	0.78	0.8	0.79	0.63
C0015672	0.86	0.83	0.85	0.82	0.70
C0042963	0.82	0.83	0.76	0.77	0.72
C0003123	0.85	0.78	0.84	0.81	0.57
C0018681	0.85	0.85	0.88	0.83	0.80
C0015967	0.67	0.64	0.67	0.62	0.43
C0206062	0.5	0	0	0	0
C0023895	0.5	0	0	0	0.40
C0012833	0.67	0.69	0.61	0.71	0.62
C0030193	0.56	0.52	0.53	0.5	0.28
C0002170	0.82	0.82	0.89	0.78	0.64
C0004096	0.91	0.91	0.94	0.91	0.82
C0022658	0.8	0.67	0.8	0.8	0
C0020517	0.76	0.77	0.74	0.74	0.58
C0917801	0.69	0.69	0.7	0.67	0.48

Continued on next page

[5]Note that in this approach, it is obvious that we have the same scores for all the languages.

Table 5: **(Individual) Per Class Scores of our approaches** (Continued)

label	ja	fr	en	de	horiz.
C0009806	0.81	0.82	0.79	0.79	0.79
C0005956	0.67	0.5	0.5	0.5	0
C0000737	0.8	0.7	0.78	0.76	0.55
C0010692	0.5	0.67	0.5	0.57	0
C0015230	0.71	0.71	0.73	0.72	0.56
C0149745	0.73	0.68	0.72	0.69	0.33
micro avg	0.79	0.76	0.78	0.76	0.62
macro avg	0.74	0.67	0.68	0.67	0.50
weighted avg	0.79	0.75	0.77	0.76	0.60
samples avg	0.23	0.21	0.22	0.22	0.18

Finally, Table 6 shows the performance of our systems on the official test set as announced by the organizers. The second and the third featurization methods do not result in better performance in our experiments. Their results are therefore excluded in this paper.

Table 6: (Full) Per Label Scores of our systems

	ja	fr	en	de	horiz.
0	1.0	1.0	1.0	1.0	0.99
1	0.79	0.76	0.78	0.76	0.62
accuracy	0.99	0.99	0.99	0.99	0.98
macro avg	0.89	0.88	0.89	0.88	0.80
weighted avg	0.99	0.99	0.99	0.99	0.98
Exact accuracy	0.83	0.83	0.84	0.83	0.70

5 CONCLUSIONS

In this paper, we described our approaches to the MedNLP-SC subtask of the NTCIR-17. We use two data combination methods along with large multilabel pretrained language models and a simple neural network models. Experimental results show that these approaches have a promising results despite their simplicity.

REFERENCES

[1] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. *Pattern Recognition* 37, 9 (2004), 1757–1771.

[2] Qingyu Chen, Jingcheng Du, Alexis Allot, and Zhiyong Lu. 2022. LitMC-BERT: Transformer-Based Multi-Label Classification of Biomedical Literature With An Application on COVID-19 Literature Curation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 19 (2022), 2584–2595.

[3] Amanda Clare and Ross King. 2001. Knowledge Discovery in Multi-Label Phenotype Data. 42–53.

[4] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather than Generators. In *Proceedings of ICLR*.

[5] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 8440–8451.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*. Minnesota, USA, 1–16.

[7] Jingcheng Du, Qingyu Chen, Yifan Peng, Yang Xiang, Cui Tao, and Zhiyong Lu. 2018. ML-Net: multi-label classification of biomedical texts with deep neural networks. *Journal of the American Medical Informatics Association : JAMIA* (2018).

[8] André Elisseeff and Jason Weston. 2001. A Kernel Method for Multi-Labelled Classification. *14th Advances in Neural Information Processing Systems* 14, 681–687.

[9] Nadia Ghamrawi and Andrew Mccallum. 2005. Collective multi-label classification. 195–200.

[10] Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. In *The Eleventh International Conference on Learning Representations*.

[11] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-Enhanced BERT with Distangled Attention. In *International Conference on Learning Representations*.

[12] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36 (2019), 1234 – 1240.

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692 [cs.CL]

[14] Disheng Pan, Xi zi Zheng, Weijie Liu, Mengya Li, Meng Ma, Ying Zhou, Li Yang, and Ping Wang. 2020. Multi-label Classification for Clinical Text with Feature-level Attention. *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)* (2020), 186–191.

[15] Jesse Read, Bernhard Pfahringer, Geoffrey Holmes, and Eibe Frank. 2009. Classifier Chains for Multi-label Classification. *Machine Learning* 85, 254–269.

[16] Wissam Siblini, Frank Meyer, and Pascale Kuntz. 2018. CRAFTML, an Efficient Clustering-based Random Forest for Extreme Multi-label Learning. In *International Conference on Machine Learning*.

[17] Grigorios Tsoumakas, Ioannis Katakis, and I. Vlahavas. 2011. Random k-Labelsets for Multi-Label Classification. *IEEE Trans. Knowl. Data Eng.* 23 (07 2011), 1079–1089.

[18] Shoko Wakamiya, Lis Kanashiro Pereira, Lisa Raithel, Hui-Syuan Yeh, Peitao Han, Seiji Shimizu, Tomohiro Nishiyama, Gabriel Herman Bernardim Andrade, Noriki Nishida, Hiroki Teranishi, Narumi Tokunaga, Philippe Thomas, Roland Roller, Pierre Zweigenbaum, Yuji Matsumoto, Akiko Aizawa, Sebastian Möller, Cyril Grouin, Thomas Lavergne, Aurélie Névéol, Patrick Paroubek, Shuntaro Yada, and Eiji Aramaki. 2023. NTCIR-17 MedNLP-SC Social Media Adverse Drug Event Detection: Subtask Overview. In *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies, NTCIR-17*. National Institute of Informatics (NII).

[19] Hui Ye, Zhiyu Chen, Da han Wang, and Brian D. Davison. 2020. Pretrained Generalized Autoregressive Model with Adaptive Probabilistic Label Clusters for Extreme Multi-label Text Classification. In *International Conference on Machine Learning*.

[20] Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. 2018. AttentionXML: Label Tree-based Attention-Aware Deep Model for High-Performance Extreme Multi-Label Text Classification. In *Neural Information Processing Systems*.

[21] Jiong Zhang, Wei-Cheng Chang, Hsiang-Fu Yu, and Inderjit S. Dhillon. 2021. Fast Multi-Resolution Transformer Fine-tuning for Extreme Multi-label Text Classification. *ArXiv* abs/2110.00685 (2021).

[22] Min-Ling Zhang and Zhi-Hua Zhou. 2007. ML-KNN: A lazy learning approach to multi-label leaming. *Pattern Recognition* 40 (07 2007), 2038–2048.

[23] Min-Ling Zhang and Zhi-Hua Zhou. 2014. A Review on Multi-Label Learning Algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2014), 1819–1837.