CINO: A Chinese Minority Pre-trained Language Model

Anonymous ACL submission

Abstract

Multilingual pre-trained language models have shown impressive performance on cross-lingual tasks. It greatly facilitates the applications of natural language processing on low-resource languages. However, there are still some languages that the existing multilingual models do not perform well on. In this paper, we propose CINO (Chinese Minority Pre-trained Language Model), a multilingual pre-trained language model for Chinese minority languages. It covers Standard Chinese, Cantonese, and six other Chinese minority languages. To evaluate the cross-lingual ability of the multilingual models on the minority languages, we collect documents from Wikipedia and build a text classification dataset WCM (Wiki-Chinese-Minority). We test CINO on WCM and two other text classification tasks. Experiments show that CINO outperforms the baselines notably. The CINO model and the WCM dataset will be made publicly available.

1 Introduction

The multilingual pre-trained language model (MPLM) is known for its ability to understand multiple languages and its surprising zero-shot cross-lingual ability (Wu and Dredze, 2019). The zero-shot cross-lingual transfer ability enables the MPLM to be applied on the target languages with limited or even no annotated data by fine-tuning the MPLM on the source language with rich annotated data. MPLMs greatly facilitate transferring the current NLP technologies to the low-resource languages and reduce the cost for developing NLP applications for low-resource languages.

The existing public MPLMs such as mBERT (Devlin et al., 2019), XLM (Conneau and Lample, 2019) and XLM-R (Conneau et al., 2020) can handle 100 languages, but there are still some challenges on processing low-resource languages:

- For some low-resource languages, the pre-training corpora are small compared to the high-resource languages. Thus, during the pre-training the bias towards high-resource languages may harm the performance on low-resource languages.
- There are thousands of living languages in the world, so a large amount of languages has not been covered in the existing MPLMs, especially indigenous or minority languages. For example, Tibetan, a language spoken mainly by Tibetans around Tibetan Plateau, is absent from the CC-100 corpus. Therefore, the XLM-R tokenizer can not tokenize Tibetan scripts correctly and XLM-R is not good at understanding Tibetan texts.

Recently, more advanced MPLMs have been proposed, such as ERNIE-M (Ouyang et al., 2021), VECO (Luo et al., 2021) and Unicoder (Huang et al., 2019). These models focus on sophisticated training task design, such as leveraging parallel sentences to improve the alignment between different languages. They have achieved notable improvements over XLM-R. However, these models have not paid attention to low-resource languages, so the problem remains unsolved.

For the above reasons, it is necessary to develop multilingual pre-trained language models for low-resource and minority languages. In this paper, we focus on Chinese minority languages. In China, Standard Chinese is the predominant language. There are in addition several hundred minority languages. Among these languages, we concentrate on several most spoken minority languages together with Standard Chinese and Cantonese (a dialect of Chinese). These languages are in different language families with different writing systems, as summarized in Table 1.

Although each of the minority languages is spoken by millions of people, their digital corpora are scarce. For example, in the CC-100 corpus used in XLM-R, the size of the Uyghur corpus is 0.4 GB, which is about 1% of the Chinese (Simplified) cor-

ISO	Language	Language Family	Writing System
zh	Standard Chinese	Sino-Tibetan	Chinese characters
yue	Cantonese	Sino-Tibetan	Chinese characters
bo	Tibetan	Sino-Tibetan	Tibetan script
mn	Mongolian	Mongolic	Traditional Mongolian script
ug	Uyghur	Turkic	Uyghur Arabic alphabet
kk	Kazakh	Turkic	Kazakh Arabic alphabet
za	Zhuang	Kra-Dai	Latin alphabet
ko	Korean	Isolate	Hangul

Table 1: Language families and writing systems of the languages covered by CINO.

pus (46.9 GB); there are no Tibetan or (traditional) Mongolian corpora in the CC-100.

We proposed a multilingual pre-trained language model named CINO (**C**hinese M**ino**rity Pre-trained Language Model), which covers Standard Chinese, Cantonese and six minority languages. This is the first multilingual pre-trained language model for the Chinese minority languages.

The reason to train a multilingual pre-trained model rather than multiple monolingual pre-trained models is three folds. First, a multilingual model is more convenient than multiple monolingual models. Second, for the low-resource languages, multilingual pre-training leads to better performance than monolingual pre-training (Conneau et al., 2020; Wu and Dredze, 2020). Third, a multilingual pre-trained model provides cross-lingual transfer ability, which can reduce the data annotation cost for low-resource languages. Studies have also shown that pre-training with more languages leads to better cross-lingual performance on low-resource languages (Conneau et al., 2020).

The public text classification datasets in the minority languages are extremely limited, thus we build the WCM (**W**iki-**C**hinese-**M**inority) dataset. The WCM is a multilingual text classification dataset with 10 classes, built from Wikipedia corpora, consisting of 63k examples. The purpose of WCM is to evaluate the zero-shot cross-lingual ability of MPLMs on the Chinese minority languages.

We evaluate the CINO model on Tibetan News Classification Corpus (TNCC), Korean news topic classification (YNAT) and WCM, and compare it with the existing XLM-R model. Results show that CINO has acquired the ability of minority languages understanding and outperforms the baselines on the Chinese minority languages.

2 CINO Model

CINO is a multilingual transformer-based model which has the same model architecture as XLM-R. For the CINO-base, it has 12 layers, 768 hidden states, 12 attention heads; for the CINO-large, it has 24 layers, 1024 hidden states and 16 attention heads. The main difference between CINO and XLM-R is the word embeddings and the tokenizer. We take the XLM-R model's word embeddings and XLM-R tokenizer as our starting point. To adapt them for the minority languages, we conduct vocabulary extension and vocabulary pruning.

Vocabulary extension. The original XLM-R tokenizer does not recognize Tibetan scripts and Traditional Mongolian scripts, so we extend the XLM-R tokenizer and XLM-R word embeddings matrix with additional tokens.

To extend the tokenizer, we train sentence-piece tokenizers for Tibetan and Mongolian on the pre-training corpora respectively. Each of the tokenizers has a vocabulary size of 16,000. Then we merge the vocabulary from the Tibetan and Mongolian tokenizers into the original XLM-R tokenizer. The merged tokenizer has a vocabulary size of 274,701.

To extend the word embeddings, we resize the original word embeddings matrix of shape $V \times D$ to $V' \times D$ by appending new rows, where D is the hidden size, V is the original vocabulary size, V' is the new vocabulary size. The new rows represent the word vectors of the new tokens from the merged tokenizer. They are initialized with a Gaussian distribution of mean 0 and variance 0.02.

Vocabulary pruning. We prune the word embeddings matrix to reduce the model size. We tokenize the pre-training corpora with the merged tokenizer, and remove all the tokens that have not appeared in the corpora from the merged tokenizer's vocabulary and the word embeddings matrix. After the operation, we discard 139,342 tokens. A smaller vocabulary size not only leads to a memory-friendly model, but also leads to a faster model by reducing the cost of computing the log-softmax in the MLM (masked language model) task. The forward pass time is reduced by approximately 35% by pruning the vocabulary size from 270k to 140k.

Finally, we obtain the CINO model structure with vocabulary size 135,359, total model size 728 MB for the base model, 1.7 GB for the large model, 68% and 79% size of XLM-R-base and XLM-R-large respectively.

3 WCM Dataset

WCM is based on the data from Wikipedia. It covers seven languages including Mongolian, Tibetan, Uyghur, Kazakh, Korean, Cantonese and Standard Chinese. We build the dataset from the Wikipedia page dumps and the Wikipedia category dumps [1] of the languages in question.

We firstly generate a category graph for each language, where each node represents a category and each edge stands for the affiliation between a pair of categories. By referring to the category system of Chinese Wikipedia, we choose 10 categories for the classification task: Art, Geography, History, Nature, Science, Personage, Technology, Education, Economy and Health. Then, we start from the categories of each page and backtrack along the routes in the category graph until meeting one of the 10 target categories and we set this category as the label of that page. Owing to some affiliation conflicts like subcategory A belongs to two categories in the 10 categories simultaneously, we reconstructed the graph by modifying certain edges between the 10 target categories and their subcategories which are assessed as unreasonable by our human evaluation team.

After getting the labeled data, we apply several strategies to improve the quality of the datasets. In the first place, we removed dirty data like large blocks of URLs, file paths. Then, the examples are restricted by their lengths (after being tokenized by the CINO tokenizer) and we filter out those examples which are shorter than 20 or longer than 1024 tokens. Furthermore, since there are both high-resource languages like Korean and extremely low-resource languages like Uyghur, we down-sample the languages and the categories with abundant data to balance the numbers of examples among different languages and different categories in each language. Finally, we obtain 63,137 examples. WCM contains the train/dev/test set for Standard Chinese and only test sets for other languages.

The data distribution is shown in the Table 3 [2].

[1] https://dumps.wikimedia.org/other

Model	TNCC		YNAT
	test	dev	dev
TextCNN	63.4	65.1	-
base models			
XLM-R-base	18.4	21.4	85.5
CINO-base	67.3	69.6	85.2
large models			
XLM-R-large	33.1	34.4	86.9
CINO-large	**68.6**	**71.3**	**87.4**

Table 2: Model performance on the Tibetan text classification task TNCC and Korean text classification task YNAT. The metric is macro-F1.

4 Experiments

4.1 Pre-training

The CINO is trained with the standard MLM objective. The masking probability is 0.15 with a max length 256. The monolingual corpora involve the languages listed in Table 1, with a total size of 36 GB. We randomly sample a subset dataset from the public base version of WuDaoCorpora (Yuan et al., 2021) as the Standard Chinese corpus. The corpora for the other languages are in-house.

We initialize the parameters of CINO with XLM-R. We use the AdamW optimizer (Loshchilov and Hutter, 2019) with the max learning rate 5e-5. The learning rate is scheduled with 10k warm-up steps followed by a linear decay. We train the model with the batch size of 2,048 for 150k steps (large model) or 200k steps (base model) on 16 Nvidia A100 GPUs.

4.2 Downstream Evaluation

We evaluate the CINO and the baselines on Tibetan News Classification Corpus (Qun et al., 2017) (TNCC), Korean news topic classification (Park et al., 2021) (YNAT) and WCM. On TNCC and YNAT, we evaluate the model in-language performance, i.e., we train and evaluate the model on the same language. On WCM, we evaluate the

[2] We show the detailed distribution and samples of the WCM dataset in the appendix.

	mn	bo	ug	kk	ko	yue	zh (test)	zh (dev)	zh (train)	Total
# examples	2,973	1,110	300	6,258	6,558	5,943	4,000	3,995	32,000	63,137

Table 3: Data distribution of the WCM dataset.

Model	**mn**	**bo**	**ug**	**kk**	**ko**	**yue**	**zh**	**Average**
base models								
XLM-R-base	41.2	25.7	84.5	23.0	43.1	66.1	88.3	53.1
CINO-base	71.1	38.3	**89.6**	**37.6**	45.9	67.1	89.2	62.7
large models								
XLM-R-large	38.3	14.5	83.9	19.1	45.6	67.3	88.4	51.0
CINO-large	**72.8**	**43.4**	**89.6**	36.4	**47.6**	**68.0**	**90.1**	**64.0**

Table 4: Model performance on the WCM test sets. The metric on each language is macro-F1.

cross-lingual ability, i.e., we train the model on the Chinese training set and evaluate the model on the test sets of all the languages.

4.2.1 TNCC

TNCC is a Tibetan classification dataset with 12 classes. It uses the macro-F1 score as the evaluation metric. In the paper Qun et al. (2017), the authors proposed two tasks: news title classification and news document classification. Here we conduct the news document classification only. The task is to predict the topic of each document. Since no official splits of the dataset are given, We split the dataset into a training set, a development set and a test set with a ratio of 8:1:1. The results are listed in Table 2. The TextCNN (Kim, 2014) is trained from the scratch, and we have searched for the best architecture. The XLM-R model has not been pre-trained on the Tibetan corpus, therefore it has a very low score. Compared to the baselines, the CINO model boosts the performance notably.

4.2.2 YNAT

YNAT is a Korean text classification dataset with 7 classes. The task is to predict the topic of each text snippet. The macro-F1 score is used as the evaluation metric. The results are listed in Table 2. The CINO-large outperforms XLM-R-large, while the CINO-base is slightly lower than XLM-R-base. Notice that Korean is not low-resourced (in the CC-100, the size of Korean corpus is 54 GB), XLM-R may have learned Korean well. For the CINO-base model, we expect that a longer pre-training time would help improve its performance on Korean.

4.2.3 WCM

We train the CINO on the Standard Chinese training set and test it on the test sets of all the languages. We use the weighted-F1 (Pedregosa et al., 2011) as the metric on each language to account for the imbalance between the categories

$$\text{weighted-F1}(y, \hat{y}) = \frac{1}{\sum_l |\hat{y}_l|} \sum_l |\hat{y}_l| \cdot \text{F1}(y_l, \hat{y}_l),$$

where y_l is the set of examples with predicted label l, $\hat{y}_l$ is the set of examples with true label l. The summation is over all labels. The overall metric is the average of weighted-F1 over all the languages. The results are listed in Table 4. CINO has a superior zero-shot performance over XLM-R. A curious observation is that XLM-R-base performs better than XLM-R-large on some languages, especially on those which have not been pre-trained on: Mongolian, Tibetan, and Kazakh. We leave this problem for future study.

5 Conclusion

In this paper, we introduce CINO, a multilingual pre-trained language model for Chinese minority languages. We build a multilingual text classification dataset WCM for zero-shot ability evaluation on the Chinese minority languages. We evaluate CINO on the Tibetan text classification task TNCC, Korean text classification task YNAT and WCM. The results show that CINO has acquired the ability of minority languages understanding and outperforms the baselines. In the future, we will explore more advanced pre-training techniques and collect more data to further improve its performance.

References

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*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7057–7067.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. 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)*, pages 2485–2494, Hong Kong, China. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Fuli Luo, Wei Wang, Jiahao Liu, Yijia Liu, Bin Bi, Songfang Huang, Fei Huang, and Luo Si. 2021. VECO: Variable and flexible cross-lingual pre-training for language understanding and generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3980–3994, Online. Association for Computational Linguistics.

Xuan Ouyang, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. ERNIE-M: Enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 27–38, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Jiyoon Han, Jangwon Park, Chisung Song, Junseong Kim, Yongsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice Oh, Jung-Woo Ha, and Kyunghyun Cho. 2021. Klue: Korean language understanding evaluation.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Nuo Qun, Xing Li, Xipeng Qiu, and Xuanjing Huang. 2017. End-to-end neural text classification for tibetan. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data - 16th China National Conference, CCL 2017, - and - 5th International Symposium, NLP-NABD 2017, Nanjing, China, October 13-15, 2017, Proceedings*, volume 10565 of *Lecture Notes in Computer Science*, pages 472–480. Springer.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Sha Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, Xiao Liu, Yukuo Cen, Xu Zou, Zhilin Yang, and Jie Tang. 2021. Wudaocorpora: A super large-scale chinese corpora for pre-training language models. *AI Open*, 2:65–68.

A Statistics of the Datasets

The sizes of TNCC and YNAT are shown in Table 5. Detailed data distribution of WCM is listed in Table 6.

Dataset	# Train	# Dev	# Test	# Classes
TNCC	7,363	920	920	12
YNAT	45,678	9,170	9,170	7

Table 5: Number of examples in TNCC and YNAT.

B Samples from the WCM dataset

Samples from the WCM Dataset in show in Figure 1.

Categories	mn	bo	ug	kk	ko	yue	zh-train	zh-test	zh-dev
Arts	135	141	3	348	806	387	2657	335	331
Geography	76	339	256	572	1197	1550	12854	1644	1589
History	66	111	0	491	776	499	1771	248	227
Nature	7	0	7	361	442	606	1105	110	134
Natural Science	779	133	20	880	532	336	2314	287	317
Personage	1402	111	0	169	684	1230	7706	924	953
Technology	191	163	8	515	808	329	1184	152	134
Education	6	1	0	1392	439	289	936	118	130
Economy	205	0	0	637	575	445	922	109	113
Health	106	111	6	893	299	272	551	73	67
Total	2973	1110	300	6258	6558	5943	32000	4000	3995

Table 6: Number of examples of each category in each language in WCM.

Language	Text	Label
mn	[illegible] (1937.03.06 [illegible]) — [illegible] · БНМАУ [illegible] ··	Personage
bo	ཚྭ་ནི་དངོས་པོ་དཀར་པོ་ཞིག་སྟེ་ཆུ་ནང་དུ་ཞུ་རང་བ་དང་ཟཆར་འགྲོ་བ་འདོན་ཆེད་དང་ཟ་ཆ་ཉར་ཚགས་བྱེད་རྒྱུར་འཕན་ཕྱིད་ཡིན།	Science
ug	قوڭۇر يىلپىز بولسا سۈت ئەمگۈچى ھايۋانلار (ياكى سۈت ئەمگۈچىلەر)	Nature
kk	فيبروبەتون — تسەمەنت-بەتون ٴتۇرى. فيبروبەتون مودۋلدىك جانە ٴمونوليتتى قۇرىلىمداردا پايدالانىلادى. فيبروبەتون سوزىلۋعا بەرىك جانە ٴتوزىمدى ك	Technology
ko	다이시도 역(일본어: 太子堂駅)은 일본 미야기현 센다이시 다이하쿠구에 있는 동일본 여객철도 도호쿠 본선의 역이다. 섬식 승강장 1면 2선의 구조를 지닌 고가역이다. 국도 제4호선 센다이 시 교통국 지하철 난보쿠 선 도미자와 역	Geography
yue	哮喘係種慢性氣管炎，病發時會氣喘、咳同呼吸困難。呢隻病嘅成因係由基因同環境一齊影響嘅。	Health
zh	独弦琴，是京族所特有的弹拨乐器，流行于中国广西等地区以及越南。 独弦琴俗称匏弦，早期采用竹子为材料，今亦采用其它材料制作。 需左右手协调配合，一挑一拨。	Art

Figure 1: Samples from the WCM dataset.

.