CINO: A Chinese Minority Pre-trained Language Model

Ziqing Yang[†], **Zihang Xu**[†], **Yiming Cui**[‡†*], **Baoxin Wang**[†], **Min Lin**,
Dayong Wu[†], **Zhigang Chen**[†§]
[†]State Key Laboratory of Cognitive Intelligence, iFLYTEK Research, Beijing, China
[‡]Research Center for SCIR, Harbin Institute of Technology, Harbin, China
[§]Jilin Kexun Information Technology Co., Ltd., Changchun, China
[†]{zqyang5,zhxu13,ymcui,bxwang2,dywu2,zgchen}@iflytek.com
[‡]ymcui@ir.hit.edu.cn

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 current 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, Yue Chinese, and six other ethnic minority languages. To evaluate the cross-lingual ability of the multilingual model on ethnic minority languages, we collect documents from Wikipedia and news websites, and construct two text classification datasets, WCM (Wiki-Chinese-Minority) and CMNews (Chinese-Minority-News). We show that CINO notably outperforms the baselines on various classification tasks. The CINO model and the datasets are publicly available at `http://cino.hfl-rc.com`.

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 low-resource languages and reduce the cost of 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 low-resource languages understanding:

- The size of pre-training corpora of some low-resource languages is small compared to the high-resource languages. This bias towards high-resource languages may harm the performance on low-resource languages.
- There are thousands of living languages in the world, but many languages have not been covered in the existing MPLMs, especially indigenous or ethnic 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 multilingual training objectives, such as leveraging parallel sentences to improve the alignment between different languages, and have improved notably over XLM-R. However, these models have not paid attention to the 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 ethnic minority languages. In this paper, we focus on Chinese minority languages. In China, Standard Chinese (Mandarin Chinese) is the predominant language. Besides Standard Chinese, we consider several most spoken minority languages. These languages are in different language families with varying writing systems, as summarized in Table 1.

Although each of the listed minority languages is spoken by at least millions of people, their digital corpus resources are quite limited. For example, in

*Email corresponding.

Proceedings of the 29th International Conference on Computational Linguistics, pages 3937–3949
October 12–17, 2022.

ISO Code	Language Name	Language Family	Writing System
zh	Standard Chinese (Mandarin)	Sino-Tibetan	Chinese characters
yue	Yue Chinese (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: Families and writing systems of the languages covered by CINO.

the CC-100 corpus used by XLM-R, the size of the Uyghur (ug) corpus is 0.4 GB, which is about 1% of the Chinese (Simplified) corpus (46.9 GB); also, there are no Tibetan (bo) or (traditional) Mongolian (mn) corpora in the CC-100.

We propose a multilingual pre-trained language model named CINO (**C**hinese M**ino**rity Pre-trained Language Model), which covers Standard Chinese, Yue Chinese (Cantonese) and six ethnic minority languages. As far as we know, this is the first multilingual pre-trained language model for the Chinese minority languages. CINO largely has the same structure as XLM-R and has been adapted for minority languages by resizing its vocabulary and adopting a fast masked language modeling objective for the pre-training.

The reason for training a multilingual pre-trained model rather than multiple monolingual pre-trained models is threefold. First, a multilingual model is more convenient than multiple monolingual models. Second, for 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 reduces 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 natural language understanding tasks in Chinese minority languages are extremely limited. In this work, we construct two multilingual datasets from two data sources to support evaluating the zero-shot cross-lingual ability of MPLMs on the Chinese minority languages: (1) The **WCM** (**W**iki-**C**hinese-**M**inority) dataset is a multilingual text classification dataset built from Wikipedia corpora, with 10 classes, consisting of 63k examples. (2) **CMNews** (**C**hinese **M**inority **N**ews) dataset is a multilingual news classification dataset with 8 classes, built from the crawled news and the pre-existing news datasets, consisting of 57k examples.

To evaluate CINO from different perspectives, we run experiments on Tibetan News Classification Corpus (TNCC), Korean news topic classification (YNAT), WCM, and CMNews. Results show that CINO has acquired the ability of minority language understanding and outperforms the existing baselines on the Chinese minority languages.

To summarize, our contributions are:

- We introduce CINO, the first multilingual pre-trained language model for Chinese minority languages. Besides Standard Chinese, CINO covers Yue Chinese and six ethnic minority languages.
- We construct two multilingual text classification datasets for Chinese minority languages. They are used for evaluating the cross-lingual and multilingual abilities of the ethnic minority language model.
- Experiments show that CINO achieves notable improvements over the baselines. Furthermore, by making the model public, CINO will be a useful resource on Chinese minority languages and facilitate related research.

2 Related Work

2.1 Pre-trained Language Models

Multilingual Pre-trained Language Models. Devlin et al. (2019) introduced the first multilingual pre-trained language model mBERT trained with Masked Language Modeling (MLM). Conneau and Lample (2019) proposed Translation Language Modeling (TLM) to train the multilingual model with cross-lingual supervision. Since then, various kinds of multilingual pre-training objectives have been proposed. Unicoder (Huang et al., 2019) trains the model with the objectives including cross-lingual word recovery, cross-lingual paraphrase classification and cross-lingual MLM. InfoXLM (Chi et al., 2021) proposed a pre-training task based on contrastive learning from an information-theoretic perspective. Pan et al. (2021) also introduced an alignment method based on contrastive learning. Cao et al. (2020) proposed an explicit word-level alignment procedure. ERNIE-M (Ouyang et al., 2021) integrates back-translation into the pre-training process. VECO (Luo et al., 2021) uses a cross-attention module to build the interdependence between languages explicitly. In this work, we only use non-parallel data and an objective similar to MLM for pre-training CINO.

Non-English Pre-trained Language Models and Benchmarks. Many pre-trained models have been trained on English corpora, or corpora that are heavily biased toward English. To make NLP techniques accessible to people from different cultures, researchers have developed pre-trained models and benchmarks targeting different languages: FlauBERT and the FLUE benchmark for French (Le et al., 2020), KLUE-BERT and the KLUE benchmark for Korean (Park et al., 2021), IndoBERT and the IndoLEM benchmark for Indonesian (Koto et al., 2020), and there are Chinese-BERT-wwm (Cui et al., 2021) and Arabic BERT AraBERT (Antoun et al., 2020). However, there are no pre-trained language models targeting Chinese ethnic minority languages.

2.2 Language Diversity in China

There are 56 ethnic groups and more than 80 languages in China. Standard Chinese (Mandarin) is the official language, spoken mainly by ethnic Han Chinese, which accounts for more than 90% of the total population. Ethnic minorities have their own languages. According to the study in Moseley (2010), the ethnic minority languages Mongolian, Uyghur, Kazakh, Tibetan,Yi, and Korean are safe (five of them are covered by CINO), which are spoken by about 25 million people, while the rest are in unsafe or endangered status.

Besides the ethnic minority languages, there are dialects and varieties of Chinese across the country. In this work, we consider Yue Chinese (also known as Cantonese), a widely used group of varieties of Chinese in Southern China and have been carried by immigrants to Southeast Asia and many other parts of the world.

Some languages in Table 1 are spoken and widely used in more than one country, such as Korean, Mongolian and Kazakh. In this work, we named them as *minority* languages based on their status in China.

3 CINO Model

In this section, we present the CINO model structure and the pre-training methodology. We denote by N the number of pre-training languages, $\mathcal{C}_i$ the monolingual corpus of the ith language ($i = 1, \ldots, N$). Let n_i be the number of sentences and l_i be the mean sequence length in $\mathcal{C}_i$. Let c_i represent the total number of tokens of $\mathcal{C}_i$.

3.1 Model Structure

CINO is a multilingual transformer-based model with the same architecture as XLM-R. For the CINO-base, it has 12 layers, 768 hidden states, and 12 attention heads; for the CINO-large, it has 24 layers, 1024 hidden states, and 16 attention heads. The main differences between CINO and XLM-R are the word embeddings and the tokenizer. We start from the word embeddings and the tokenizer of XLM-R and adapt them for the minority languages by vocabulary extension and vocabulary pruning, as depicted in Figure 1.

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.

We train sentence-piece tokenizers for Tibetan and Mongolian on their monolingual pre-training corpora respectively. Each of the tokenizers has a vocabulary size of 16,000. Then we merge the vocabularies 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 0.0 and variance 0.02.

Vocabulary Pruning. Next, we prune the word

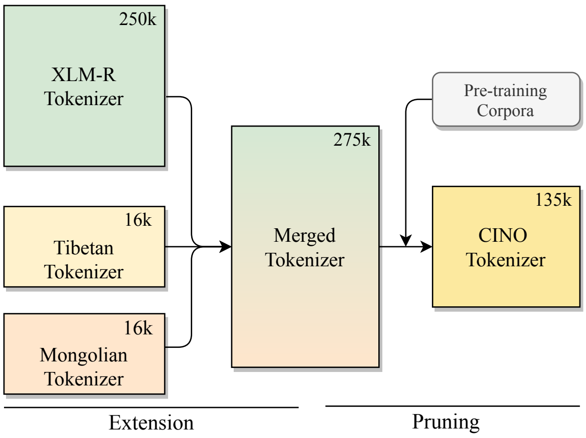

Figure 1: We extend the XLM-R tokenizer with a Tibetan tokenizer and a Mongolian tokenizer, then remove the redundant tokens to obtain the CINO tokenizer.

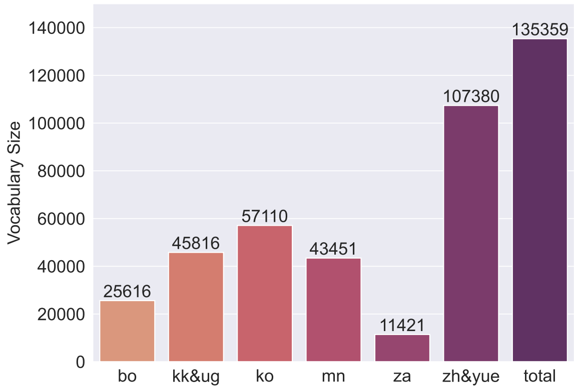

Figure 2: The vocabulary size counted from the corpus of each language. We merge the vocabularies of the languages that have similar writing systems.

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. The above process discards 139,342 tokens.

Finally, we obtain the CINO model structure with a vocabulary size of 135,359, a model size of 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. A smaller vocabulary size leads to not only a memory-friendly model but also a faster model by reducing the cost of computing the log-softmax in the MLM task. The time cost of each iteration in pre-training is reduced by approximately 35% by reducing the vocabulary size from 270k to 140k.

3.2 Pre-training

We adopt the MLM objective for pre-training. In addition, we apply the following strategies for balancing training data and faster pre-training.

3.2.1 Resampling Strategy

To balance the data size between high-resource and low-resource languages, Conneau and Lample (2019) and Chi et al. (2021) have applied a multinomial sampling strategy. An example in the ith language is sampled with the probability

$$p_i = \frac{n_i^\alpha}{\sum_k^N n_k^\alpha}, \tag{1}$$

where $\alpha \in (0, 1]$ is a hyperparameter.

However, if the mean sequence lengths of different corpora are different, it may lead to an undesired data bias.[1] To see this, we use $\tilde{c}_i$ to denote the number of tokens seen during training. We have $\tilde{c}_i \propto p_i l_i$ and $\tilde{c}_i = K c_i$ for all $i = 1 \ldots N$ if $\alpha = 1$. K is a constant that only depends on the number of training steps. If two languages i and j that have the same number of tokens, i.e., $c_i = c_j$, but with $n_i > n_j$ and $l_i < l_j$. With the sampling ratio in (1), we get $\tilde{c}_i < \tilde{c}_j$ if $\alpha < 1$ although the original corpora are of the same size. To remedy this, we introduce the dependence on the mean sequence length l_i. The sampling probability is

$$p_i = \frac{n_i^\alpha / l_i^\beta}{\sum_k^N n_k^\alpha / l_k^\beta}, \tag{2}$$

where $\beta \in [0, 1]$. Setting $\beta = 1 - \alpha$, the number of training tokens in the ith language is

$$\tilde{c}_i \propto p_i l_i \propto n_i^\alpha l_i^{1-\beta} = (n_i l_i)^\alpha = c_i^\alpha. \tag{3}$$

Therefore, corpora of equal size will be trained with an equal number of tokens.

3.2.2 Fast Masking Language Modeling

Table 1 shows that the languages we consider have distinguished writing systems, which implies that the vocabulary of each language only takes up a fraction of the whole vocabulary, as shown in Figure 2. By taking advantage of this fact, the computational costs can be reduced if the model only makes MLM predictions over the vocabulary of the specific language of the input examples rather than the whole vocabulary.

[1]In most cases, we could join short sequences to form long sequences of a uniform length. But some corpora we use consist of short sentences. Joining them as a long sequence leads to semantically incoherence.

Suppose the example is in the ith language. We denote by $\mathcal{V}$ the full vocabulary, and $\mathcal{V}_i \subset \mathcal{V}$ the vocabulary of the ith language, which is obtained by tokenizing the ith language's monolingual corpus. Let (c, x) denote the input text sequence, where x is the masked token, and c is the context. By limiting the prediction of the masked token to $\mathcal{V}_i$, the MLM loss of the masked token x is

$$\mathcal{L}_{\text{MLM}}^{(i)} = -\log \frac{\exp(g(c) \cdot E(x))}{\sum_{x' \in \mathcal{V}_i} \exp\left(g(c) \cdot E(x')\right)}, \quad (4)$$

where $g(\cdot)$ is the transformer encoder and $E(\cdot)$ is the look-up operation that returns the embeddings.

In order to calculate the loss (4) efficiently, during training, we group examples by language so that each batch contains examples in a single language.

With the objective (4) for pre-training, we have observed 10% time reduction and no significant performance drop compared to the original MLM objective, which predicts over the whole vocabulary. Combined with the speedup by vocabulary pruning, the pre-training time cost is reduced by about 40% in total.

4 Text Classification Datasets for Minority Languages

Multilingual tasks have been used widely to evaluate the cross-lingual transferability of multilingual models (Hu et al., 2020). Nevertheless, the pre-existing multilingual datasets hardly cover the Chinese ethnic minority languages. For example, Tibetan, Mongolian and Uyghur have never appeared in any task in the XTREME benchmark. To evaluate the cross-lingual transferability of CINO, we construct two text classification datasets **WCM** (**W**ikipedia-**C**hinese-**M**inority) and **CMNews** (**C**hinese-**M**inority-**N**ews).

4.1 WCM Dataset

Data Collection and Annotation. WCM is based on the data from Wikipedia. It covers seven languages: Mongolian, Tibetan, Uyghur, Kazakh, Korean, Cantonese, and Standard Chinese. We build the dataset from the Wikipedia page dumps and the Wikipedia category dumps[2] of the languages in question.

To annotate the data, we first generate a category graph for each language. 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 ten 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 reaching one of the ten target categories, and we set this category as the label of that page. Owing to some affiliation conflicts, like one subcategory belonging to two categories simultaneously, we reconstructed the graph by removing certain edges between the 10 target categories and their subcategories which are assessed as unreasonable by our human evaluation team.

Data Cleaning. After getting the labeled data, we apply several strategies to improve the quality of the datasets. We remove dirty data like large blocks of URLs and file paths. Then, the examples are filtered by their lengths (after being tokenized by the CINO tokenizer) by removing those examples shorter than 20 or longer than 1024 tokens.[3]

Subsampling. Since there are both high-resource languages like Korean and low-resource languages like Uyghur, we down-sample the data in the high-resource languages and the high-resource categories to balance the numbers of examples among different languages and different categories. We fix the size of the training set (Chinese articles) to 32K and downsample the datasets of the languages with abundant articles to about $5\% \sim 20\%$ size of the training set. Similarly, we also down-sampled some categories if they dominate in some languages. We did not apply the above process to Uyghur due to its extreme scarcity.

Finally, we obtain 63,137 examples. WCM contains the train/dev/test set for Standard Chinese and only test sets for other languages. The detailed distribution is listed in Appendix C.

4.2 CMNews Dataset

Data Collection and Annotation. To collect the minority language examples, we crawl the news from the news websites in ethnic minority languages and record the category to which each news item belongs. To collect the Chinese news, we reuse the pre-existing dataset SogouCS News (Wang et al., 2008) and CAIL 2018 (Xiao et al., 2018). We select the appropriate categories and

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

[3]We discard examples that are too long because long examples likely cover multiple topics while we assign a single label to each example.

Dataset		mn	bo	ug	kk	ko	yue	zh	Total
WCM	**# Samples**	27	5	4	52	43	49	20	200
	# Correctly Labeled	24	4	4	49	34	43	19	177
	Matching Acc	88.9%	80%	100%	94.2%	79.1%	87.8%	95.0%	88.5%
CMNews	**# Samples**	11	34	24	14	10	23	84	200
	# Correctly Labeled	8	31	24	14	10	20	80	187
	Matching Acc	72.7%	91.2%	100%	100%	100%	87.0%	95.2%	93.5%

Table 2: Results of human evaluation of the sampled examples from WCM and CMNews.

down-sample the two datasets to make the whole dataset more balanced.

After gathering the raw data from all the languages, we first merge the categories that have similar meanings (for example, we merge the categories *Finance* and *Economy*). Since the definition of news category may vary from website to website and language to language, we remove the categories that are not consistent in different languages by manually checking a sampled subset. We also remove the categories that do not appear in more than two languages. Finally, we obtain a dataset containing eight categories: Education, Sports, Health, Tourism, Legal, Economy, Culture, and Society.

Data Cleaning. The crawled news is much cleaner than the Wikipedia pages, and each document naturally belongs to only one category. Therefore we only perform length filtering by keeping the documents that contain more than 30 tokens after tokenization.

The dataset contains 56,764 examples in total. We split the dataset into a training set and a development set. The detailed distribution is listed in Appendix C.

4.3 Human Evaluation

To assess the quality of the datasets, we randomly sample 200 examples from WCM and 200 examples from CMNews and manually check whether the contents of the examples match their labels. The results are shown in Table 2. **Matching Acc** denotes how many examples match their labels under human evaluation. We find that 88.5% of the sampled examples from WCM and 93.5% of the sampled examples from CMNews are correctly labeled, which shows CMNews has less noise.

5 Experiments

5.1 Pre-training Setup

Pre-training Data. We randomly sample a subset dataset from the public base version of WuDaoCorpora (Yuan et al., 2021) as the Standard Chinese corpus; the corpora of the minority languages are in-house data, consisting of short monolingual sentences. The total corpora size is 28 GB. The statistics of the pre-training corpora are listed in Appendix A.

Experiment Settings. CINO is trained with the fast MLM objective (4) with the masking probability is 0.2 and the max sequence length 256. We initialize the parameters of CINO with XLM-R. We use the AdamW optimizer (Loshchilov and Hutter, 2019) with the peak learning rate of 2e-4 for the base model and 1e-4 for the large model. The learning rate is scheduled with 10k and 5k warmup steps followed by a linear decay for the base and the large model respectively. The sampling hyperparameter α is set to 0.7. We train the model with the batch size of 4,096 for 150k steps for the base model, and the batch size of 8,192 for 75k steps for the large model. The pre-training is performed on 16 NVIDIA A100 GPUs. The full pre-training hyperparameters are summarized in Appendix B.1.

5.2 Downstream Evaluation

How does CINO perform on the newly introduced languages? How does CINO perform on the languages pre-existing in XLM-R? Does CINO show multilingual and cross-lingual abilities? To answer these questions, we evaluate CINO on (1) Tibetan News Classification Corpus (Qun et al., 2017) (**TNCC**); (2) Korean news topic classification (Park et al., 2021) (**YNAT**); (3) **WCM** and **CMNews**. On TNCC and YNAT,[4] we evaluate the in-language model performance, i.e., we train and evaluate the model on the same language. On WCM and CMNews, we evaluate the cross-lingual ability. We describe the details in Section 5.4.

For each task and each model, we run the experiment five times with different seeds and report the mean metrics. The fine-tuning hyperparameters of

[4]The splitting sizes of TNCC and YNAT are listed in Appendix C.

each experiment are listed in Appendix B.2.

5.3 Baselines

Besides the common multilingual pre-trained models **mBERT** and **XLM-R**, we compare CINO models with the following baselines on some tasks.

XLM-R-Ext. We extend and prune the vocabulary of XLM-R as described in Section 3.1. This model is the un-pretrained CINO. The embeddings of Tibetan and Mongolian are randomly initialized, and the other parameters are the same as XLM-R.

KLUE-BERT-base. This is a Korean pre-trained model proposed in Park et al. (2021). Although KLUE-BERT-base is a base-sized model, it outperforms other large models on the YNAT task except for XLM-R-large.

TextCNN is a simple and light-weight model for text classification tasks (Kim, 2014). The word embedding dimension is set to 300. After the embedding layer, we apply three convolution layers in parallel with the number of out-channels 100, kernel size 3,4, and 5, respectively. Finally, we concatenate the outputs from the convolution layers and apply a two-layer fully-connected network with ReLU activation to perform the classification. We train the TextCNN from scratch with randomly initialized model parameters and word embeddings.

Word2vec (Tibetan). We first train the word embeddings using word2vec (Mikolov et al., 2013a,b) on the TNCC training set. The embedding dimension is set to 300. To perform the classification task, we average the word embeddings of each sample, then feed the results to a trainable linear layer that outputs the logits.

5.4 Results and Discussions

5.4.1 TNCC

How does CINO perform on the newly introduced language? We evaluate CINO on TNCC, a Tibetan classification dataset with 12 classes. The original work (Qun et al., 2017) proposes a news title classification and a news document classification. Here we conduct the news document classification only. The task is to predict the topic of each document. Because there are no official splits available, we split the dataset into a training set, a development set and a test set with a ratio of 8:1:1. Since the texts in the dataset have been pre-tokenized (spaces have been added between words), we remove the spaces between words and tokenize the texts with the pre-trained tokenizer unless otherwise specified. We select the best checkpoint based on its macro-F1 score. We also report the accuracy score for reference.

Model	TNCC Dev		TNCC Test	
	Acc	Macro-F1	Acc	Macro-F1
TextCNN	69.4	65.7	62.8	66.6
Word2vec (Tibetan)	70.1	67.7	70.2	68.0
base models				
mBERT	22.9	4.8	22.8	5.5
mBERT (p.t.)	63.9	56.2	61.8	56.4
XLM-R-base	35.1	20.2	31.1	21.1
XLM-R-base (p.t.)	34.2	21.5	31.4	19.9
XLM-R-Ext-base	55.7	43.2	55.0	42.1
CINO-base	74.8	71.4	73.1	70.0
large models				
XLM-R-large	35.7	26.4	32.8	27.3
XLM-R-Ext-large	31.6	13.0	29.2	12.2
CINO-large	**76.3**	**73.7**	**75.4**	**72.9**

Table 3: Model performance on the Dev and Test sets of Tibetan text classification task TNCC. *p.t.* is short for *pre-tokenized*.

Model	YNAT Dev	
	Acc	Macro-F1
mBERT (Park et al., 2021)	-	82.6†
XLM-R-base (Park et al., 2021)	-	84.5†
XLM-R-large (Park et al., 2021)	-	87.3†
KLUE-RoBERTa-large (Park et al., 2021)	-	85.9†
KLUE-BERT-base (Park et al., 2021)	-	87.0†
base models		
mBERT	82.9	82.8
XLM-R-base	85.1	85.0
KLUE-BERT-base	87.0	**87.1**
CINO-base	86.1	85.9
large models		
XLM-R-large	87.0	86.8
CINO-large	**87.3**	87.0

Table 4: Model performance on the Dev set of Korean text classification task YNAT. The results marked with † are taken from the KLUE paper (Park et al., 2021). The rest results are from our experiments.

The results are listed in Table 3. Compared among the pre-trained models, XLM-R series have low scores since the vocabulary is not adapted for the Tibetan language and has not been pre-trained on the Tibetan corpus. While XLM-R-Ext-base has an extended vocabulary and significantly outperforms XLM-R-base even without being pre-trained on the target language. Finally, by pre-training on the minority languages corpora, CINO is adapted to the new language and outperforms XLM-R and XLM-R-Ext notably.

mBERT achieves better results when fine-tuned

	Model	bo	kk	ko	mn	ug	yue	zh	Avg (Minorities)	Avg (All)
WCM	*base models*									
zh → min.	XLM-R-base	19.0	16.7	43.2	15.2	23.3	58.3	78.1	29.3	36.2
	CINO-base	36.2	43.2	**44.9**	39.1	**33.4**	59.7	78.0	42.6	47.6
	large models									
	XLM-R-large	18.4	32.9	43.8	22.2	27.8	**60.0**	77.3	34.2	40.3
	CINO-large	**40.6**	**44.8**	**44.8**	**41.6**	28.8	59.8	**79.2**	**43.3**	**48.4**
	Model	**bo**	**kk**	**ko**	**mn**	**ug**	**yue**	**zh**	**Avg (Minorities)**	**Avg (All)**
CMNews	*base models*									
min. → zh	XLM-R-base	38.1	69.6	88.3	35.1	77.5 (67.7/88.6)	**87.8**	58.6	66.1	65.0
	CINO-base	85.5	79.2	89.0	77.3	77.4 (77.0/78.0)	86.9	68.8	82.6	80.6
	large models									
	XLM-R-large	30.1	80.8	88.9	30.8	**85.1** (76.4/91.0)	87.5	63.6	67.2	66.7
	CINO-large	**86.8**	**83.0**	**90.3**	**79.4**	78.8 (68.4/91.3)	**87.9**	**71.2**	**84.4**	**82.5**

Table 5: Model performance on the WCM and CMNews. The metric on each language is macro-F1. **Avg (Minorities)** is the mean score over languages other than zh; **Avg (All)** is the mean score over all languages. We bold any score within 0.1 of the best on each language. The results in the parentheses are the min and the max values of five runs.

on the pre-tokenized data (but there are still many tokens being mapped to `[UNK]`). Due to the difference in the tokenization algorithms used by mBERT and XLM-R, XLM-R does not benefit from using pre-tokenized data.

TextCNN and Word2vec (Tibetan) surprisingly achieve competitive scores and outperforms XLM-R-Ext-base. It is possibly due to the difficulty in the optimization of large models such as XLM-R with limited training data. As we continue increasing the model size, the performance gets worse, as can be seen from comparing the scores of XLM-R-base-Ext and XLM-R-large-Ext.

5.4.2 YNAT

How does CINO perform on the minority languages pre-existing in XLM-R? We evaluate CINO on YNAT, a Korean text classification dataset with 7 classes. We select the best checkpoint based on its macro-F1 score. The results are listed in Table 4. CINO-base outperforms XLM-R-base, while CINO-large is better than XLM-R-large by our reimplementation but lower than the score reported in Park et al. (2021). CINO-large is also comparable to KLUE-BERT-base.

Notice that Korean is not a low-resource language in XLM-R (the size of the Korean corpus is 54 GB in the CC-100), thus XLM-R may have learned Korean well. To significantly outperform XLM-R and KLUE-BERT-base, we expect that longer training time and more data are required.

5.4.3 WCM and CMNews

Does CINO show multilingual and cross-lingual abilities? We use these two datasets to evaluate the cross-lingual and multilingual abilities. We take macro-F1 as the metric on each language, and the *Avg* is the arithmetic mean of the macro-F1 scores.

On the WCM dataset, we train models on the Chinese training set and test it on all the languages, so the results show how well the model transfers the knowledge from Chinese to the minority languages; the best checkpoint of each run is selected based on its score on Chinese; On the CMNews dataset, we train models on the minority languages and the Chinese data is zero-shot; the best checkpoint is selected based on its score on minority languages. The results are listed in Table 5.

On WCM, **Avg (Minorities)** score shows that CINO has superior zero-shot performance over XLM-R. By inspecting the detailed performance on each language, we see that CINO most significantly outperforms XLM-R on Tibetan, Kazakh, Mongolian and Uyghur, which have been insufficiently pre-trained in XLM-R.

On CMNews, because CINO has been adapted to minority languages, it learns more effectively than XLM-R by leveraging the examples in all the languages. **zh** score shows that CINO transfers better than XLM-R. CINO also outperforms XLM-R on almost all the minority languages except for **ug**, where there is a large gap. To find out the reason, we list the min and the max **ug** scores of five runs. We see that there is a large variance.

CINO-large achieves the highest score among all runs, but its average score is lower than XLM-R-large. The unstable performance may be the main reason that explains the gap.

6 Discussion on Limitations

Coverage of ethnic minority languages. Due to the scarcity of minority language corpora, CINO only covers Standard Chinese and some of the most popular minority languages and dialects. While being spoken by millions of people, some languages, such as the Yi language, are omitted in this study since we can not find sufficient data for pre-training.

Pre-training objectives. In our early trials of multilingual pre-training, we leveraged both monolingual and bilingual parallel data, and combined the MLM objective with a cross-lingual alignment objective, similar to the TLM objective used in Chi et al. (2021) and Conneau and Lample (2019). Intuitively, parallel data contain more information than monolingual data. However, we have not observed significant improvements over pre-training with only monolingual data and the MLM objective. The performance of CINO may be improved if parallel data can be effectively used.

Languages from different cultures. Among the languages in Table 1, some are cross-border languages. The cross-border languages are spoken in more than one country and are influenced by local cultures. How well does the model that has been trained on the corpus collected in one country transfer to the corpus collected in another country? If the writing systems of the language are different (for example, Mongolian is written in Cyrillic in Mongolia, while it is written in traditional Mongolian script in China), to what extent do writing systems influence the model performance? We expect future work to address these questions.

7 Conclusion

In this paper, we introduce CINO, a multilingual pre-trained language model for Chinese minority languages. It takes the same structure as XLM-R but with a different vocabulary and is pre-trained with an adapted MLM objective to reduce computational costs. We build multilingual text classification datasets WCM from Wikipedia and CMNews from ethnic minority news for zero-shot ability evaluation on the Chinese minority languages. We evaluate CINO on several text classification tasks. The results show that CINO achieves notable improvements over the existing baselines.

Acknowledgments

We would like to thank all anonymous reviewers for their thorough review and for providing constructive comments to improve our paper.

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

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.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514.

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.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

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.

Fajri Koto, Afshin Rahimi, Jey Han Lau, and Timothy Baldwin. 2020. IndoLEM and IndoBERT: A benchmark dataset and pre-trained language model for Indonesian NLP. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 757–770, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

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.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

C. Moseley. 2010. *Atlas of the World's Languages in Danger*. Memory of peoples Series. UNESCO Publishing.

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.

Lin Pan, Chung-Wei Hang, Haode Qi, Abhishek Shah, Saloni Potdar, and Mo Yu. 2021. Multilingual BERT post-pretraining alignment. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 210–219, Online. 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.

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.

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. ELRA. `http://is.muni.cz/publication/884893/en`.

Canhui Wang, Min Zhang, Shaoping Ma, and Liyun Ru. 2008. Automatic online news issue construction in web environment. In *Proceedings of the 17th*

International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008, pages 457–466. ACM.

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.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2018. CAIL2018: A large-scale legal dataset for judgment prediction. *ArXiv preprint*, abs/1807.02478.

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 Pre-training Corpora

The corpus size and mean sequence length for pre-training are listed in Table 6. The sequence lengths are obtained by counting the tokens after tokenization. For Standard Chinese (zh), we concatenate or truncate each example to the max sequence length, while for other languages, we do not concatenate the examples but keep them unchanged.

Language	# Tokens	Mean Sequence Length
bo	130M	13.4
kk	238M	60.7
ko	170M	20.0
mn	337M	25.7
ug	1B	23.1
yue	276M	12.6
za	23M	58.1
zh	1.2B	254

Table 6: Corpus size and mean sequence length of each language in the pre-training data.

B Hyperparameters

B.1 Pre-training Hyperparameters

Hyperparameter	Base Model	Large Model
Batch Size	4,096	8,192
Warmup Steps	10k	5k
Training Steps	150k	75k
Peak Learning Rate	2e-4	1e-4
Max Length	256	256
MLM probability	0.2	0.2
Adam ϵ	1e-8	1e-8
Adam β_1	0.9	0.9
Adam β_2	0.999	0.999
Gradient Clipping	1.0	1.0
Weight Decay	0	0
Sampling α	0.7	0.7

Table 7: Hyperparameters used for pretraining CINO models.

Table 7 presents the full set of the hyperparameters used for pre-training CINO models.

B.2 Fine-tuning Hyperparameters

The hyperparameters for fine-tuning on the downstream tasks is listed in Table 9. The batch size is 32 for all experiments except Word2vec (Tibetan), of which batch size is 16. The learning rate is scheduled with 10% warmup steps followed by a linear decay.

We use Gensim (Řehůřek and Sojka, 2010) to train the Word2vec embeddings, and set `min_count` = 1, `vector_size` = 300. Other parameters take the default values.

Dataset	# Train	# Dev	# Test	# Classes
TNCC	7,359	191	923	12
YNAT	45,678	9,106	-	7

Table 8: Number of examples in TNCC and YNAT.

C Statistics of the Datasets

The sizes of TNCC and YNAT are shown in Table 8. Detailed data distribution of WCM is listed in Table 10. Detailed data distribution of CMNews is listed in Table 11.

Model	TNCC		YNAT		WCM		CMNews	
	LR	**Epochs**	**LR**	**Epochs**	**LR**	**Epochs**	**LR**	**Epochs**
Word2vec (Tibetan)	3e-2	20	-	-	-	-	-	-
TextCNN	1e-4	40	-	-	-	-	-	-
mBERT	3e-5	40	2e-5	5	-	-	-	-
KLUE-BERT-base	-	-	3e-5	3	-	-	-	-
XLM-R-base	5e-5	40	3e-5	3	1e-5	20	3e-5	5
CINO-base	5e-5	40	3e-5	3	1e-5	20	3e-5	5
XLM-R-large	3e-5	40	2e-5	3	1e-5	20	3e-5	5
CINO-large	3e-5	40	2e-5	3	1e-5	20	3e-5	5

Table 9: Hyperparameters used for downstream fine-tuning.

Category	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 10: Number of examples in each category and language in WCM.

Split	Category	bo	kk	ko	mn	ug	yue	zh
Train	**Education**	626	364	378	187	423	880	1979
	Sports	66	133	321	556	1216	70	1978
	Health	1309	153	40	31	240	1358	2000
	Tourism	1128	12	43	102	1078	0	1998
	Legal	433	283	283	294	19	22	2000
	Economy	399	107	192	510	0	1080	1877
	Culture	1834	231	228	118	0	0	1995
	Society	898	149	147	543	1132	169	1935
	Total	6693	1432	1632	2341	4108	3579	15762
Dev	**Education**	418	243	253	125	282	587	1000
	Sports	44	89	215	371	811	48	1000
	Health	874	103	28	21	160	906	1000
	Tourism	752	8	30	68	719	0	1000
	Legal	289	190	189	196	14	15	1000
	Economy	266	72	129	341	0	721	1000
	Culture	1223	155	152	80	0	0	1000
	Society	600	100	99	362	756	113	1000
	Total	4466	960	1095	1564	2742	2390	8000

Table 11: Number of examples in each category and language in CMNews.