Representative Data Selection for Sequence-to-Sequence Pre-training

Haiyue Song[1,2] Raj Dabre[2] Zhuoyuan Mao[1] Chenhui Chu[1] Sadao Kurohashi[1]
[1]Kyoto University [2]NICT
{song, zhuoyuanmao, chu, kuro}@nlp.ist.i.kyoto-u.ac.jp
{raj.dabre}@nict.go.jp

Abstract

Pre-trained sequence-to-sequence models such as BART [1] have helped improve natural language generation quality. However, training large models is resource-consuming. We propose a data selection algorithm that selects a tiny but representative subset from billion-scale datasets. Experimental results show that pre-training with 0.26% data and 4.4% energy consumption achieves about 90% BLEU scores on **machine translation (MT)** tasks and ROUGE scores on text summarization tasks, compared to pre-training on the entire dataset. Compared to random selection baselines, it shows lower **perplexity (PPL)**, higher BLEU and ROUGE scores.

1 Introduction

Pre-training and then fine-tuning is a widely-used paradigm for natural language processing [2, 3]. However, training pre-trained models such as BART [1, 4], IndicBART [5], mT5 [6] usually takes hundreds to thousands of GPU days. Previous works focus on reducing the parameters of the model [7], but there are very few studies [8] related to shrinking the dataset, which can also reduce computational costs.

In this work, we propose a clustering-based representative data selection algorithm. As illustrated in Figure 1, we first convert discrete sentences into continuous embeddings. Then, we perform large-scale and efficient clustering of the sentences based on the embeddings. From each cluster, we select several centroid points according to the scale of the cluster. The centroids from each cluster are combined to form the representative subset. Furthermore, we propose to combine an unsupervised outlier detection method to remove noisy data points. We calculate the center point of the entire embedding space and filter out points distant from the center. Experimental results show that with 0.26% data of the entire dataset and 4.4% energy comsumption, it can obtain relatively high performance on MT and text summarization tasks, only 2 to 4 points lower in terms of BLEU and ROUGE scores.

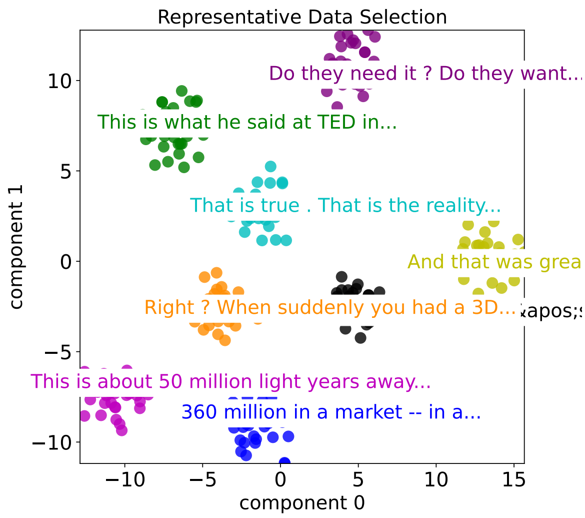

Figure 1: Centroids of clusters as representative data. Each component stands for one cluster with close sentence embeddings. Mapped to 2-D with t-SNE [9].

2 Related Work

Supervised Data Selection In-domain data selection [10, 11, 12] focuses on extracting sentences from a large general-domain corpus that are most relevant to a target domain. Trusted data or clean selection [13, 14] aims to select trusted (clean) data from a general-domain corpus given a small trusted (clean) dataset. They all rank data according to the cross-entropy difference [10] between a general **language model (LM)** and a target LM, where the target LM is trained on in-domain data, trusted data, or clean data. However, they require supervision and only solve one particular downstream task.

Small Scale Representative Data Selection Representative data selection finds a small subset of the original dataset that captures the most information. Previous methods include calculating the mutual information and relative entropy [15], converting to a sparse multiple measurement vector problem [16]. They are slow and require large memory, therefore, can only handle approximately 10k data points; however, billions of sentences are used in pre-training.

3 Representative Data Selection

We introduce the representative data selection approach to extract a fraction from a multi-million to billion scale dataset. It consists of the following steps:

Continuous Embedding Conversion In order to perform clustering, we first convert discrete data such as sentences into continuous representations in a common space. Suppose there is a set containing n sentences $S = \{s_1, ..., s_n\}$. We convert S into embedding set $E = \{e_1, ..., e_n\}$ as following:

$$e_i = f(s_i|\theta), \qquad f : \mathbb{S} \rightarrow \mathbb{R}^d \tag{1}$$

where f denotes the sentence-to-vector model, θ is the parameters of f, d is the dimension of the output vector and $\mathbb{S}$ is a set of all the natural language sentences. Here f can be sent2vec [17] or sentBERT [18].

Outlier Detection We apply an outlier detection algorithm [19] to eliminate noisy data in an unsupervised manner. We first calculate the center of the embedding space e_c and filter outliers whose distance from e_c is greater than two standard deviations. The de-noised embedding set contains m points, $E' = \{e'_1, ..., e'_m\}$. More precisely:

$$\begin{aligned} e_c &= center(E) = \frac{1}{n}\sum_i e_i \\ E' &= \{e'_i \mid e'_i \in E, ||e'_i - e_c|| < 2\sigma\} \end{aligned} \tag{2}$$

where σ denotes the standard deviation.

Clustering and Selection Suppose we select a subset S' containing k sentences. We first apply efficient K-Means algorithm on GPUs [20] to create k clusters $c_1, ..., c_k$ from E'. For each cluster c_i with size $|c_i|$, we select $\frac{k}{m} * |c_i|$ sentences whose embeddings are the nearest from the center of c_i, forming $S'_{c_i} = \{e_1^{(c_i)}, ..., e_N^{(c_i)}\}$:

$$S'_{c_i} = \underset{\{e_j^{(c_i)} \in c_i\}}{\arg\min} \sum_{j=1}^{N} ||e_j^{(c_i)} - center(c_i)|| \tag{3}$$

where $N = \frac{k}{m}|c_i|$.

The representative set S' of the entire dataset S is the union of all representative sets from each cluster:

$$S' = \bigcup_{i=1}^{k} S'_{c_i}, \quad |S'| = \sum_{i=1}^{k} \frac{k}{m} * |c_i| = k \tag{4}$$

4 Experiments

4.1 Settings

■ Datasets

- **Pre-train**: IndicCorp [21] that contains a total of 458M sentences in 11 Indian languages and English.
- **MT**: PMI dataset [22] from WAT2021 MultiIndicMT task [23].
- **Summarization**: data in 7 Indic languages from multilingual XLSum dataset [24].

We applied script unification for all Indic languages to Devanagari, following mBART50 [4] and IndicBART [5]. Across all experiments, we used the IndicBART vocabulary of 64k subwords.[1)]

■ Pre-train Methods Comparison

- **w/o Pre-train**: directly train on downstream tasks from random parameters initialization.
- **Random**: pre-trained on k randomly selected sentences.
- **Random+RemoveOutlier (RO)**: first remove outliers, then apply **Random**.
- **Repre**: pre-train on k sentences by representative data selection w/o outlier detection.
- **Repre+RemoveOutlier (RO)**: first remove outliers, then apply **Repre**.
- **Full**: use 458M monolingual sentences in the IndicCorp dataset.

We compare proposed **Repre** and **Repre+RO** methods with two baselines **Random** and **Random+RO**. We set k to $1.2M$ and select sentences from different languages while keeping their proportions in the IndicCorp dataset. We follow fine-tuning settings in [5].

1) Download: https://github.com/AI4Bharat/indic-bart

■ **Representative Data Selection Settings**

- **Continuous Embedding Conversion**: we trained one Sent2vec [17] model for each language on IndicCorp data with default settings and sentence embedding dimension to 768.
- **Clustering Algorithm**: we used GPU-implemented K-Means in the Faiss toolkit [20].

■ **Model Hyperparameters** We followed the settings of IndicBART[2] and used the yanmmt toolkit[3] based on Hugging Face.[4]

- **Architecture**: transformer model of 6 encoder layers and 6 decoder layers with 16 attention heads.
- **Training**: we used 8 GPUs with a batch size of 4,096 tokens during pre-training and 2,048 tokens during fine-tuning. We trained 200k steps in pre-training and apply early stopping to fine-tuning.

4.2 Pre-trained Model Perplexity

We report our results in terms of the perplexities obtained on a mix of all dev sets from the PMI dataset that contains high-quality data from 10 Indian languages and English. As shown in Figure 2, proposed **Repre** method showed approximately 0.15 lower minimal PPL than **Random**. Furthermore, **RO** is effective for both **Random** and **Repre** methods.

4.3 Energy Consumption Comparison

- **Full**: trained on 48 V100 GPUs for 750k steps [5].
- **Proposed**: trained on 8 V100 GPUs for 200k steps.

Therefore, our approach reduces the energy consumption to 4.4%[5] compared with **Full**.[6]

4.4 MT Results

As presented in Table 3, proposed methods yield the highest BLEU scores for all pairs compared with baselines. With 4.4% energy consumption, our results are only 2-4 BLEU points lower than **Full**. Additionally, **RO** helps both **Random** and **Repre**.

2) https://github.com/AI4Bharat/indic-bart
3) https://github.com/prajdabre/yanmtt
4) https://huggingface.co
5) (8*200k)/(48*750k)=4.4%
6) Training sent2vec models and clustering consumes very little energy in comparison.

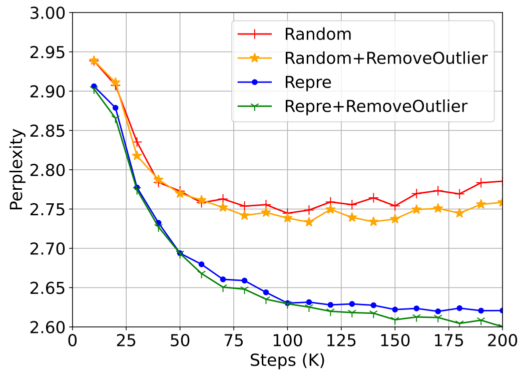

Figure 2: Perplexity curves of pre-trained models on the PMI dev sets.

4.5 Summarization Results

As expressed by Table 1, proposed methods achieve higher ROUGE-L F-scores than baselines. Especially for low-resource **bn** language that contains only 80k training points, **Repre** is more robust than **Random**.

Table 1: ROUGE-L F1 scores on the summarization task.

Method	bn	gu	hi	mr	pa	ta	te	Avg
Baselines								
Rand (1.2M)	5.7	15.1	28.3	16.5	22.0	16.1	11.4	16.4
+RO	9.7	15.9	28.7	17.6	21.4	16.0	11.1	17.2
Proposed								
Repre (1.2M)	**15.1**	15.9	29.3	18.3	**22.3**	12.6	12.0	17.9
+RO	13.0	**16.4**	**29.4**	**18.6**	20.0	**17.2**	**12.4**	**18.1**
Reference								
Full (458M)	17.2	17.9	32.2	20.1	24.0	19.3	14.6	20.8

4.6 Outlier Detection Examples

We show examples of normal sentences and outliers. We extract 300 English sentences from IndicCorp and apply the outlier detection algorithm to form Figure 3 together with Table 2. We can find that outlier sentences contain more proper nouns and disfluent phrases.

Table 2: The corresponding sentences in Figure 3.

Type	Sentences
Center	This never happened before ...
Normal	Suddenly, there's something that was happening... So at some point it became, you know...
Outlier	You have Palestine-Loves-Israel. They have graphic designers. What?

Table 3: Performance on the MT task. Report sacreBLEU [25] scores on the WAT2021 MultiIndicMT test set.

Pre-train Model	bn	gu	hi	kn	ml	mr	or	pa	ta	Avg
	Others→English									
w/o Pre-train	13.5	27.4	30.9	22.5	16.5	18.4	18.4	27.1	17.1	21.31
Baselines										
Random (1.2M)	18.9	31.2	34.1	26.6	23.0	23.0	23.7	31.1	22.4	26.00
+RemoveOutlier	19.1	31.1	34.1	27.1	22.9	23.0	24.6	31.5	22.4	26.20
Proposed										
Repre (1.2M)	**19.6**	**32.2**	33.7	27.1	**23.8**	23.2	**24.7**	31.6	22.6	26.50
+RemoveOutlier	19.5	31.9	**34.5**	**27.6**	23.7	**23.8**	24.5	**32.0**	**23.1**	**26.73**
Reference										
Full (458M)	23.4	35.7	37.6	31.5	28.3	27.3	28.4	36.0	27.0	30.58
	English→Others									
w/o Pre-train	4.5	17.9	21.7	12.1	3.9	10.0	9.2	17.9	7.2	11.60
Baselines										
Random (1.2M)	6.4	21.1	23.8	15.4	**5.6**	13.1	10.5	22.9	9.0	14.20
+RemoveOutlier	6.8	21.2	24.2	15.3	5.5	13.3	10.7	22.7	8.9	14.29
Proposed										
Repre (1.2M)	6.9	**21.8**	23.9	15.7	**5.6**	13.6	10.9	**22.9**	9.1	14.49
+RemoveOutlier	**7.3**	21.3	**24.7**	**16.1**	5.4	**13.8**	**11.2**	22.8	**9.6**	**14.69**
Reference										
Full (458M)	8.2	23.4	26.3	17.6	6.4	16.5	12.3	25.3	10.5	16.28

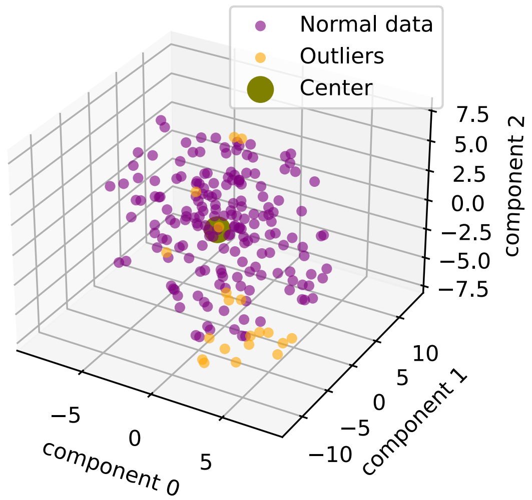

Figure 3: Outlier detection. High-dimensional embeddings are mapped into 3-D points by t-SNE [9].

4.7 Sentence Clustering Examples

Table 4 shows the clustering results. In each cluster, the centroid is the most relevant from all other points. For example, in the first cluster, sentences are related to "Earth", "Jupiter", "ocean planet", "Life on Earth" and the sentence related to "Earth" is the centroid.

Table 4: Example of clusters. The **centroids** of the clusters are representative data.

Clus	Sentences
#1	**It is the Earth as we know it.** This is the planet Jupiter. This is an ocean planet. Life on Earth is the size of the Earth.
#2	**And he started asking me questions.** Would you ask me those questions? So I started to ask myself questions about it. Number one question I get asked.
#3	**One is the beginning of the music video.** And now to introduce their music video We have a video to show you. Now this is video of a session.

5 Conclusion

In this paper, we propose a representative data selection algorithm together with an unsupervised outlier detection algorithm. With only 0.26% data and 4.4% energy consumption of the full model, proposed methods show reasonable performance on MT and text summarization tasks, and much higher performance compared to baselines.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers JP21J23124.

References

[1] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *ACL.2020*, pp. 7871–7880, Online, July 2020.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL.2019*, pp. 4171–4186, Minneapolis, Minnesota, June 2019.

[3] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A Robustly Optimized BERT Pre-training Approach with Post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pp. 1218–1227, Huhhot, China, August 2021. Chinese Information Processing Society of China.

[4] Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual Translation with Extensible Multilingual Pretraining and Finetuning, 2020.

[5] Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh M. Khapra, and Pratyush Kumar. IndicBART: A Pre-trained Model for Natural Language Generation of Indic Languages, 2021.

[6] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *NAACL-HLT.2021*, pp. 483–498, Online, June 2021.

[7] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, 2019.

[8] Ernie Chang, Xiaoyu Shen, Hui-Syuan Yeh, and Vera Demberg. On Training Instance Selection for Few-Shot Neural Text Generation. In *ACL-IJCNLP.2021*, pp. 8–13, Online, August 2021. Association for Computational Linguistics.

[9] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, Vol. 9, No. 86, pp. 2579–2605, 2008.

[10] Robert C. Moore and William Lewis. Intelligent Selection of Language Model Training Data. In *ACL.2010*, pp. 220–224, Uppsala, Sweden, July 2010.

[11] Amittai Axelrod, Xiaodong He, and Jianfeng Gao. Domain Adaptation via Pseudo In-Domain Data Selection. In *EMNLP.2011*, pp. 355–362, Edinburgh, Scotland, UK., July 2011.

[12] Marlies van der Wees, Arianna Bisazza, and Christof Monz. Dynamic Data Selection for Neural Machine Translation. In *EMNLP.2017*, pp. 1400–1410, Copenhagen, Denmark, September 2017.

[13] Wei Wang, Taro Watanabe, Macduff Hughes, Tetsuji Nakagawa, and Ciprian Chelba. Denoising Neural Machine Translation Training with Trusted Data and Online Data Selection. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 133–143, Brussels, Belgium, October 2018. Association for Computational Linguistics.

[14] Wei Wang, Isaac Caswell, and Ciprian Chelba. Dynamically Composing Domain-Data Selection with Clean-Data Selection by "Co-Curricular Learning" for Neural Machine Translation. In *ACL.2019*, Florence, Italy, July 2019.

[15] Feng Pan, Wei Wang, A.K.H. Tung, and Jiong Yang. Finding representative set from massive data. In *ICDM.2005*, pp. 8 pp.–, 2005.

[16] Ehsan Elhamifar, Guillermo Sapiro, and René Vidal. See All by Looking at a Few: Sparse Modeling for Finding Representative Objects. In *CVPR.2012*, pp. 1600–1607, 2012.

[17] Prakhar Gupta, Matteo Pagliardini, and Martin Jaggi. Better Word Embeddings by Disentangling Contextual n-Gram Information. In *NAACL-HLT.2019*, pp. 933–939, Minneapolis, Minnesota, June 2019.

[18] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP.2019*, pp. 3982–3992, Hong Kong, China, November 2019.

[19] Stefan Larson, Anish Mahendran, Andrew Lee, Jonathan K. Kummerfeld, Parker Hill, Michael A. Laurenzano, Johann Hauswald, Lingjia Tang, and Jason Mars. Outlier Detection for Improved Data Quality and Diversity in Dialog Systems. In *NAACL-HLT.2019*, pp. 517–527, Minneapolis, Minnesota, June 2019.

[20] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-Scale Similarity Search with GPUs. *IEEE Transactions on Big Data*, Vol. 7, No. 3, pp. 535–547, 2021.

[21] Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages.

[22] Barry Haddow and Faheem Kirefu. PMIndia – A Collection of Parallel Corpora of Languages of India, 2020.

[23] Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, and Sadao Kurohashi. Overview of the 8th Workshop on Asian Translation. In *WAT.2021*, pp. 1–45, Online, August 2021.

[24] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. In *Findings of ACL-IJCNLP.2021*, pp. 4693–4703, Online, August 2021.

[25] Matt Post. A Call for Clarity in Reporting BLEU Scores.