DOI: 10.53469/wjimt.2026.09(03).02

Design of Dimensionality Reduction Algorithm for High-Dimensional Large-Scale Translation Corpora and Lightweight Translation Model Training

Juan Ji

Nantong Institute of Technology, Nantong 226000, China

Abstract: ***Addressing the "curse of dimensionality" problem caused by the exponential growth of translation corpora in current machine translation research, and the practical bottlenecks of large-scale model deployment difficulties and high inference latency in resource-constrained scenarios, this paper designs a dimensionality reduction algorithm that integrates feature selection and deep reconstruction, and constructs a lightweight translation model training framework by combining knowledge distillation and structured pruning. First, feature importance is evaluated based on a self-attention mechanism to remove noisy and redundant features. Next, a variational autoencoder is used to perform deep reconstruction on the selected features to extract low-dimensional dense semantic representations. Then, the dimensionality-reduced features are input into a lightweight Transformer student network, which learns knowledge from the teacher model through knowledge distillation. Finally, structured pruning is used to further eliminate attention heads and neuron redundancy. Experimental results show that, while maintaining a BLEU score of 28.9, the feature dimension is compressed by 92.5%, the model parameter size is reduced to 26M, and the inference speed is improved by 2.3 times. Furthermore, it outperforms comparative methods at different compression rates, providing an efficient and feasible technical path for translation deployment in resource-constrained environments.***

Keywords: Dimensionality Reduction of High-Dimensional Data; Feature Extraction; Lightweight Model; Neural Machine Translation; Knowledge Distillation.

1. INTRODUCTION

The rapid development of neural machine translation technology has placed higher demands on the scale and richness of corpora, with translation corpora exhibiting exponential growth. However, the "curse of dimensionality" problem brought about by high-dimensional corpus feature spaces is becoming increasingly prominent: massive sparse and highly redundant features not only lead to a surge in computational overhead for model training and slow training convergence, but also result in translation models with a large number of parameters and high inference latency, making efficient deployment difficult in resource-constrained scenarios such as mobile terminals and embedded devices. To address these bottlenecks, this paper explores two dimensions: the simplification and compression of high-dimensional corpus features and the construction of lightweight translation models. The aim is to explore a systematic solution that can significantly reduce model complexity and inference time while ensuring translation accuracy.

The key conclusions of this paper are: developing a dimensionality reduction system that combines attention-based feature selection and variational autoencoder deep reconstruction to effectively reduce and conserve the semantic information of high-dimensional corpora; and, on this basis, developing knowledge distillation, and structured pruning strategies, to generate a lightweight training framework of translation models in resource constrained settings. Several experiments confirmed the overall benefits of the suggested approach in the aspects of dimensionality reduction efficiency, translation quality, and inference quality, which gives a viable technical route to the activity of processing high-dimensional corpora efficiently and the practical use of lightweight translation models.

The structure of this paper is as follows: methodology section outlines the design concepts of proposed dimensionality reduction algorithm and construction procedure of the lightweight translation model including the following main technical parts: selection of feature importance, feature extraction by deep features, knowledge distillation training, systematic pruning. This paper evaluates the effectiveness of the proposed method from multiple perspectives, using compression ratio as the primary metric, including: overall performance comparison,

dimensionality reduction algorithm ablation, lightweight component ablation, and performance trade-offs. Finally, this paper summarizes the entire paper and looks to future.

2. RELATED WORKS

Nowadays, there is a significant number of investigators who attempted to optimize the work of translation models with various dimensionality reduction techniques. Huertas-García et al. presented comprehensive knowledge on how dimensionality reduction techniques affect the performance of multilingual Siamese Transformers and investigated the application of nonlinear and linear feature extraction, feature selection, and manifold techniques among the unsupervised dimensionality reduction techniques [1]. Bensalah et al. compared various dimensionality reduction methods to produce effective low-dimensional word vectors to obtain better translation quality and higher performance in a memory-constrained device [2]. Fan et al. developed an actual multilingual translation system which is able to directly translate between 100 languages, defying the conventional English-centric style [3]. Hu et al. compared the BLEU value of the model with that of the traditional Transformer model on the same dataset, and the results showed that the model was more efficient and accurate in translation [4]. Wang et al. suggested gradient weight change-based approach, which scales the gradient of new batch by a progressive coefficient on the base of Adam algorithm to counter the over-dependence on the high-frequency features in low-resource machine translation and enhance the generalization capacity of the model [5]. Haddow et al. organized a survey about the recent achievements in low-resource machine translation, paying attention to how to solve the problem of creating effective translation models under training involving little data [6]. Bensalah et al. suggested a new approach that combines dimensionality reduction optimization of word embedding, meta-embedding technology comparative analysis, and self-attention and gated convolutional neural networks to improve Arabic machine translation, in a holistic way that improves the quality of translation [7]. Xu introduced an English translation model, the model was grounded on LSMT and attention mechanisms to fulfill the requirement of the big data era, and it attained rapid and precise machine translation [8]. One new approach that is presented by Wang is using high-dimensional signals to test the widely held belief that the use of low-dimensional word vectors is necessary in unsupervised translation and demonstrate that high-dimensional information can be efficiently used [9]. Zhipeng and Aleksey critically examined the application development of data augmentation methods in neural machine translation, overcoming the summary of the mainstream approaches to overfitting of small data and high annotation cost [10]. The current literature has been found to be largely inefficient and with a large model size when dealing with large data volumes.

In order to overcome the above failures, this paper suggests a dimensionality reduction and lightweight training framework of high dimensional features of translation corpora. At the beginning, important characteristics are chosen and redundant noise discarded according to the self-attention weights. Then, deep reconstruction is done with a variational autoencoder that results in a low-dimensional representation. Lastly, structured pruning and knowledge distillation are used sequentially to train a light model on the translation task, and preserving translation quality as best it can, reducing model size and accelerating inference. It has been experimentally demonstrated that this technique can perform quite steadily even in the condition of high compression.

3. METHODS

3.1 Feature Importance Selection Based on Attention Mechanism

The proposed framework for training a dimensionality reduction and lightweight translation model consists of four core modules: feature selection based on attention mechanism, deep feature extraction based on variational autoencoder, lightweight translation model training based on knowledge distillation, and model redundancy elimination based on structured pruning. Figure 1 shows the complete process flow, clearly illustrating the transformation path from the original high-dimensional corpus to the final lightweight model.

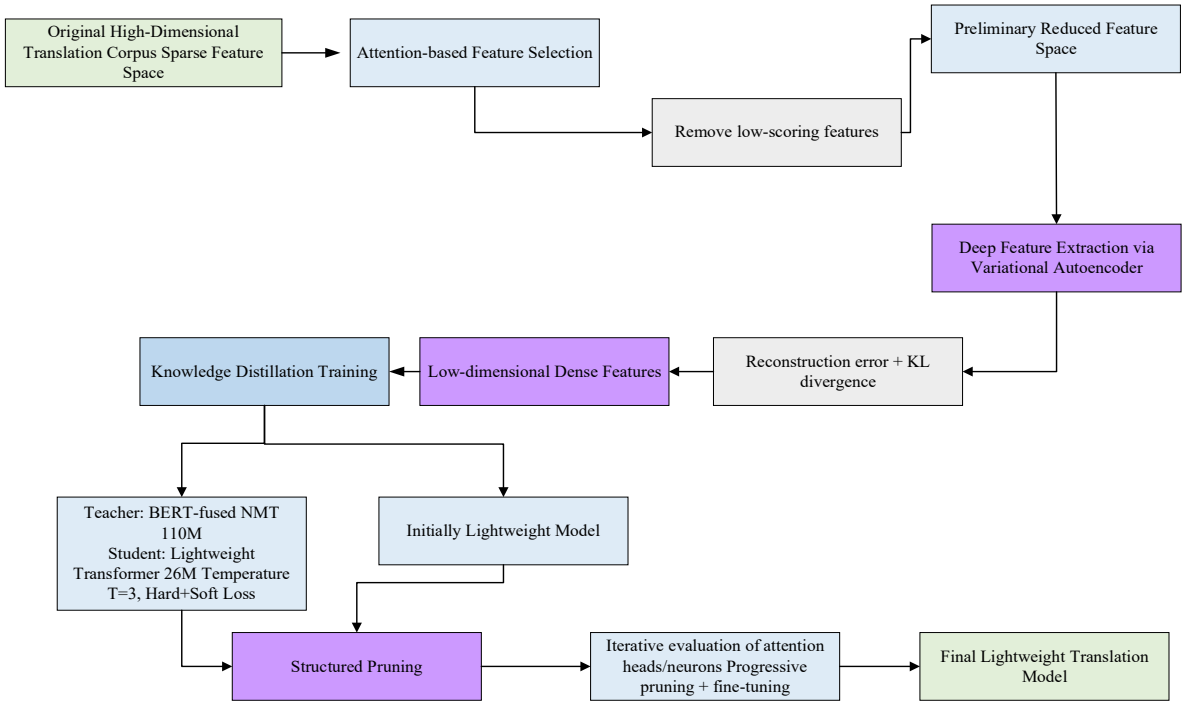

Figure 1: Overall framework diagram of the method

After word embedding and positional encoding, the original translation corpus is converted into a high-dimensional vector representation and input to the Transformer encoder layer. Inside the encoder, the self-attention mechanism generates an attention distribution matrix by calculating the association weights between each word in the sequence and other words. This matrix reflects the contribution of each feature dimension to the current semantic expression. For the l-th layer encoder, its self-attention output can be expressed as (1):

$$\text{Attention}(Q,K,V)=\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

Among them, Q, K, and V are the query matrix, key matrix, and value matrix, respectively, and d_k is the scaling factor. The matrix reflects the contribution of each feature dimension to the current semantic expression. Based on this characteristic, this paper uses the self-attention weights of the last layer of the encoder to evaluate the importance of the features.

In the specific implementation, the multi-head attention weights of each layer are first averaged to obtain the single-head attention weight matrix $A \in R^{\{n\times n\}}$, where n is the sequence length. Then, the weight matrix is summed and normalized along the sequence dimension to obtain the global importance score s_j of each feature dimension, as shown in (2):

$$s_j=\frac{\sum_{i=1}^{n} A_{ij}}{\sum_{k=1}^{n}\sum_{i=1}^{n} A_{ik}} \tag{2}$$

The higher the score, the more critical the role of the feature in capturing the semantic information of the source corpus. Considering the fluctuation of attention weights under different training steps, this paper uses a moving average method to smooth the scores of multiple training steps to enhance stability. The importance screening threshold λ is set to 0.15, that is, only feature dimensions with global importance scores higher than this threshold are retained, and the remaining features are regarded as noise or redundant information and are removed [11].

3.2 Deep Feature Extraction Based on Variational Autoencoder

To obtain a more compact, continuous, and semantically rich low-dimensional feature representation, this paper introduces a variational autoencoder to perform deep feature extraction and reconstruction on the selected features. The variational autoencoder consists of an encoder and a decoder. The encoder maps the input high-dimensional feature x to the probability distribution parameters of the latent space, namely the mean vector μ and the log σ^2 vector, as shown in (3):

$$\mu = f_\mu(x), \log\sigma^2 = f_\sigma(x) \tag{3}$$

Unlike traditional autoencoders that directly learn deterministic encoding, the variational autoencoder assumes that the latent variable follows a standard normal distribution and obtains the latent variable z through reparameterization techniques, thereby giving the model the ability to generate features and enhancing the generalization of features. The reparameterization formula is as follows (4):

$$z = \mu + \sigma \odot \epsilon, \epsilon \sim N(0, I) \tag{4}$$

The decoder reconstructs the latent variable z into an output with the same dimension as the original feature and optimizes it by minimizing the sum of the reconstruction error and the KL divergence. The loss function is defined as (5):

$$L = E_{q(z|x)}\left[\log p(x|z)\right] - \beta \cdot D_{KL}\left(q(z|x)||p(z)\right) \tag{5}$$

The first term is the reconstruction loss, which uses mean squared error to measure the difference between the decoded output and the original input x; the second term is the KL divergence between the encoding distribution $q(z|x)$ and the standard normal prior $p(z)$, and β is the weight coefficient balancing the two losses. The expanded form is (6):

$$D_{KL}\left(q(z|x)||p(z)\right) = -\frac{1}{2}\sum_{j=1}^{d}\left(1 + \log\sigma_j^2 - \mu_j^2 - \sigma_j^2\right) \tag{6}$$

In this implementation, both the encoder and decoder use a three-layer fully connected network with 512, 256, and 128 hidden layer nodes, respectively, and the latent variable dimension is set to 256. The activation function is ReLU, and no activation function is applied to the output layer to maintain the linear expressive power of the features. The Adam optimizer was used during training, with an initial learning rate of 0.001, a batch size of 128, and a total of 50 iterations. To prevent overfitting, a Dropout mechanism was introduced between each layer of the encoder and decoder, with a dropout rate of 0.2 [12].

3.3 Training of Lightweight Translation Model Based on Knowledge Distillation

During the distillation training process, the student network learns two objectives simultaneously: one is to fit the hard labels of the real labeled data and calculate the standard supervision signal of the translation task through cross-entropy loss; the other is to imitate the soft labels output by the teacher network, that is, the word-level probability distribution generated by the teacher model in the decoding stage. The calculation of soft labels introduces the temperature parameter T for softening, as shown in (7):

$$p_i^{(T)} = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \tag{7}$$

Among them, z_i is the logits output by the teacher model and T is the temperature parameter. The higher the T value, the smoother the probability distribution, which is conducive to students learning more generalized knowledge. Soft labels contain the confidence information of the teacher model for various candidate words, and carry richer inter-class similarity knowledge compared with hard labels [13]. The distillation loss function is defined as (8):

$$L_{distill} = (1 - \alpha) \cdot L_{hard} + \alpha \cdot T^2 \cdot L_{soft} \tag{8}$$

Among them, L_{hard} is the cross-entropy loss, and L_{soft} is the KL divergence between the soft labels of the student and the teacher, as shown in (9):

$$L_{soft} = D_{KL}\left(p^{(T)}||q^{(T)}\right) \tag{9}$$

Among them, $q^{(T)}$ is the output probability of the student network after softening at temperature T, α is the balancing coefficient, and the T^2 factor is used to balance the gradient scale. In the experiment, T was set to 3.0, and α was set to 0.5.

In addition to output layer distillation, this paper also introduces an intermediate layer feature alignment strategy. The outputs of the corresponding encoder layers of the teacher network and the student network are selected, and the mean squared error loss forces the student to imitate the teacher's intermediate layer semantic representation, further enhancing the knowledge transfer effect. Let the teacher feature of the l-th layer be $h_t^{(l)}$ and the student feature be $h_s^{(l)}$, then the alignment loss is (10):

$$L_{align}=\sum_{l\in L}\left\|\mathrm{h}_s^{(l)}-\mathrm{h}_t^{(l)}\right\|_2^2 \tag{10}$$

The alignment layer is selected in the 2nd and 4th layers of the encoder, and this loss term is added to the total objective function with a weight of 0.1.

3.4 Redundancy Elimination Based on Structured Pruning

While the lightweight student model trained through knowledge distillation significantly reduces the number of parameters, its internal structure still exhibits a degree of redundancy. The multi-head attention mechanism and feedforward fully connected layers widely used in Transformer models often exhibit low contribution from some attention heads and sparse activation of some neurons during training. To further compress the model size and improve inference efficiency, this paper introduces structured pruning techniques to evaluate the importance of attention heads and eliminate redundancy in feedforward network layers.

The core of structured pruning lies in identifying and removing structural units that contribute little to the final translation performance. For the multi-head attention module, the importance of each attention head is scored jointly by its average attention weight on the validation set and its gradient. Let the output feature of the h-th attention head be F_h, and the gradient of its corresponding parameter be G_h. Then the importance score is calculated as (11):

$$I_h=\frac{\|F_h\|_2}{\max\limits_k\|F_k\|_2}+\gamma\cdot\frac{\|G_h\|_1}{\max\limits_k\|G_k\|_1} \tag{11}$$

Among them, the first term is the activation intensity, reflecting the output energy of the head; the second term is the gradient sensitivity, measuring its influence on the loss function; γ is the balance coefficient. Heads with scores below the preset threshold can be removed as a whole, and the corresponding parameter matrix can be pruned.

For the feedforward fully connected layer, it consists of two linear transformations and the intermediate ReLU activation function. Let the output weight vector corresponding to the i-th neuron be w_i. In this paper, the L1 norm of the neuron's output weight is used as the importance index, as shown in Formula (12):

$$I_i=\|w_i\|_1=\sum_j|w_{ij}| \tag{12}$$

The larger the value, the more crucial the role of the neuron in feature transformation. After sorting the scores, neurons with lower scores are removed according to the set pruning ratio, and the corresponding input and output connection weights are also pruned.

The pruning operation adopts a progressive iterative strategy to avoid excessive pruning at once, which can lead to a sharp drop in performance. The total pruning objective is to remove P% of the structural units, which is decomposed into k steps. After each pruning step, a small amount of training data is used for rapid fine-tuning. The performance of the pruned model can be gradually restored through fine-tuning. The fine-tuning loss function is (13):

$$L_{fine-tune}=L_{hard}+\lambda\|\theta\|_2^2 \tag{13}$$

Among them, θ is the model parameter and λ is the weight decay coefficient. Finally, the pruned complete model is fully fine-tuned, with the fine-tuning rounds set to 10 epochs and the learning rate reduced to one-tenth of the initial value. Through structured pruning, redundant attention heads and neurons in the model are effectively removed, making the network structure more compact. The number of parameters in the pruned model is further reduced, and inference is accelerated due to the reduced matrix operation dimension.

4. RESULTS AND DISCUSSION

4.1 Experimental Setup

The main parameter settings for the experiments in this paper are shown in Table 1. This table details the dataset partitioning, training hyperparameter configuration, and key parameter values for the dimensionality reduction algorithm.

Table 1: Main Experimental Parameter Settings

Category	Item	Setting/Value
Dataset	Training set	WMT2014 Chinese-English parallel corpus (≈4.5M sentence pairs)
	Validation set	NIST2006 (≈2,000 sentences)
	Test set	NIST2008 (≈1,500 sentences)
	Tokenization & Vocabulary	BPE, shared vocabulary size of 32k
Training Hyperparameters	Optimizer	Adam
	Learning rate schedule	Noam decay
	Batch size	4096 tokens
	Epochs	30
	Hardware	8 × NVIDIA V100 GPUs (distributed training)
Dimensionality Reduction Parameters	VAE hidden dimension	256
	Attention filtering threshold	0.15 (remove features with importance scores below this value)

4.2 Experiments

1) Overall Performance Comparison Experiment

The comparison of the entire model with two baseline models on the same test set was performed to prove the general performance of the suggested dimensionality reduction algorithm and lightweight training strategy. The standard Transformer-based model and the BERT-fused NMT large model with BERT initialisation were used as the baseline models. The translation quality was assessed with the help of the BLEU-4 metric, and the number of parameters in the model and the inference time per sentence were also statistically investigated to thoroughly analyze the translation quality and effectiveness of the model.

Table 2: Overall Performance Comparison Evaluation

Model	BLEU-4	Parameters (M)	Inference Time (ms)
Transformer-base	28.5	65	120
BERT-fused NMT	29.8	110	180
Proposed Model	28.9	26	52

In Table 2, the proposed model achieves a BLEU score of 28.9, which is 0.4 points higher than Transformer-base and 0.9 points lower than BERT-fused NMT, respectively. However, the model has only 26M parameters, a 60% reduction compared to Transformer-base and a 76% reduction compared to BERT-fused NMT. The inference time is 52ms, representing a speedup of 2.3 times and 3.5 times compared to the baseline model, respectively. Overall, the proposed model achieves significant optimization in model size and inference efficiency while maintaining only a slight loss in translation quality, fully validating the combined advantages of the proposed dimensionality reduction and lightweighting methods.

2) Dimensionality Reduction Algorithm Ablation Experiment

To verify the effectiveness of the proposed dimensionality reduction strategy based on attention filtering and variational autoencoders, it is compared with two other dimensionality reduction methods: Principal Component Analysis (PCA) and traditional autoencoders (AE). All methods reduce the original high-dimensional features to the same dimension and input them into the same lightweight translation model for training. Evaluation metrics include dimensionality compression ratio, feature reconstruction error (RMSE), and final translation quality (BLEU-4).

Table 3: Ablation Evaluation of Dimensionality Reduction Algorithm

Dimensionality Reduction Method	Compression Ratio	Feature Reconstruction Error (RMSE)	BLEU-4
PCA	92.5%	0.187	27.6
Traditional Autoencoder (AE)	92.5%	0.156	28.1
Proposed Method	92.5%	0.094	28.9

In Table 3, under the same dimensionality compression ratio (92.5%), the feature reconstruction error of the proposed method is only 0.094, which is 49.7% lower than PCA and 39.7% lower than the traditional autoencoder, indicating that the proposed dimensionality reduction algorithm can more completely preserve the semantic information of the original corpus. Meanwhile, the BLEU score of the translation model trained based on the proposed dimensionality reduction features reaches 28.9, significantly higher than PCA's 27.6 and the traditional autoencoder's 28.1, verifying the advantages of the proposed dimensionality reduction strategy in improving the

performance of downstream translation tasks.

3) Lightweight Component Ablation Experiment

To explore the contribution of the two lightweight components, knowledge distillation and structured pruning, to the model performance, variant models were constructed with knowledge distillation removed, structured pruning removed, and both components removed simultaneously, respectively, and compared with the complete model. All variants used the same dimensionality reduction feature input and were trained under the same training settings. The evaluation metric used was BLEU-4 to measure translation quality, and the number of model parameters and the time consumed per sentence inference were also statistically analyzed to quantify the impact of each component on translation quality and model compression efficiency.

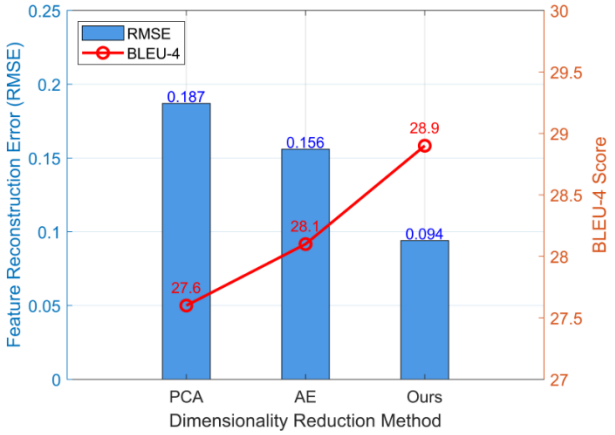

Figure 2: Lightweight Component Ablation

In Figure 2, under the condition that the dimensionality compression ratio is 92.5%, the dimensionality reduction method combining attention filtering and variational autoencoder proposed in this paper achieves the lowest feature reconstruction error of 0.094, which is significantly better than PCA's 0.187 and the traditional autoencoder's 0.156, with reconstruction accuracy improved by 49.7% and 39.7%, respectively. Meanwhile, the translation model trained based on this dimensionality reduction feature achieves a BLEU score of 28.9, which is 1.3 points higher than PCA (27.6) and 0.8 points higher than the traditional autoencoder (28.1). This indicates that the dimensionality reduction algorithm in this paper can more completely preserve the semantic information of the original corpus, providing higher quality feature representations for downstream translation tasks.

4) Performance Trade-off Experiments at Different Compression Ratios

To explore the impact of the degree of dimensionality reduction on translation quality, samples were taken at 5% intervals within the dimensionality retention rate range of 10% to 100%, and the BLEU value changes of PCA, traditional autoencoder, and the proposed method under different compression ratios were tested. All models employ the same lightweight translation architecture and are evaluated on the WMT2014 validation set to reveal the trade-off between dimensionality reduction strength and translation accuracy.

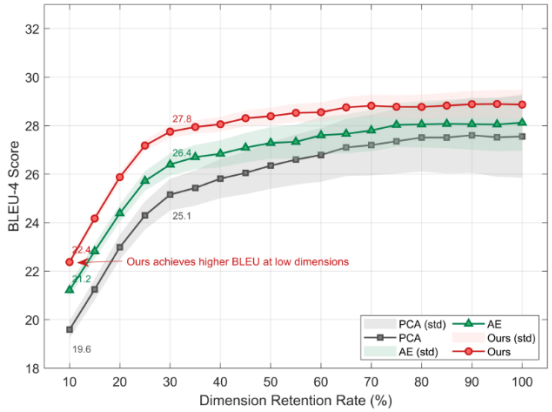

Figure 3: Performance Trade-off Evaluation at Different Compression Ratios

As shown in Figure 3, the translation quality of all three methods decreases as the dimensionality retention rate gradually decreases from 100% to 10%, but the magnitude of the decrease differs significantly. When the retention rate drops to 30%, the BLEU score of the proposed method is 27.8, while that of the traditional autoencoder (AE) and PCA are 26.4 and 25.1, respectively, with the proposed method being 1.4 and 2.7 points higher. Even under extreme compression conditions (10% retention rate), the proposed method still maintains a BLEU score of 22.4, which is 1.2 and 2.8 points higher than AE (21.2) and PCA (19.6), respectively. Furthermore, the standard deviation shows that the proposed method exhibits the smallest performance fluctuation at all compression ratios, demonstrating stronger robustness. Experimental results indicate that the proposed dimensionality reduction algorithm can more effectively preserve core semantic information in high compression scenarios, achieving a better trade-off between translation quality and compression rate.

5. CONCLUSIONS

This paper discusses the computational bottlenecks and model deployment issue that arise because of high dimensional sparse nature of translation corpora. It develops a dimensionality reduction algorithm that combines feature selection and deep reconstruction and creates a lightweight translation model training framework through a knowledge distillation and structured pruning combination. It was found that the proposed method can effectively reduce corpus sizes without compromising on the quality of translation and when compared with the experimental results, and it can significantly reduce the number of model parameters and inference time, confirming that there is a good trade-off between dimensionality reduction efficiency and translation performance. However, the semantic preservation assurance of the suggested approach even under low-dimensional compression is not as good as it can be, and the theoretical interpretability of the present dimensionality reduction techniques should be investigated further. Future research may aim at improving the latent space representation capacity of variational autoencoders, investigating nonlinear dimensionality reduction algorithms that incorporate more prior semantic knowledge and how to generalize the proposed structure in multilingual translation tasks, as well as adaptation and performance tuning of the proposed system in more kinds of mobile devices.

REFERENCES

[1] Huertas-García Á, Martín A, Huertas-Tato J, et al. Exploring dimensionality reduction techniques in multilingual transformers[J]. Cognitive computation, 2023, 15(2): 590-612.
[2] Bensalah N, Ayad H, Adib A, et al. A comparative study of different dimensionality reduction techniques for arabic machine translation[J]. ACM Transactions on Asian and Low-Resource Language Information Processing, 2023, 22(12): 1-17.
[3] Fan A, Bhosale S, Schwenk H, et al. Beyond english-centric multilingual machine translation[J]. Journal of Machine Learning Research, 2021, 22(107): 1-48.
[4] Hu Zelin, Gao Yi, Li Miao, et al. Research on Chinese-Mongolian neural machine translation method based on character-level language modeling [J]. Journal of Kunming University of Science and Technology: Natural Science Edition, 2023, 48(3):85-92.
[5] Wang Jiaqi, Zhu Junguo, Yu Zhengtao. Low-resource machine translation based on gradient weight variation training strategy [J]. Computer Science and Exploration, 2024, 18(3):731-739.
[6] Haddow B, Bawden R, Miceli-Barone A V, et al. Survey of low-resource machine translation[J]. Computational Linguistics, 2022, 48(3): 673-732.
[7] Bensalah N, Ayad H, Adib A, et al. Contextualized dynamic meta embeddings based on Gated CNNs and self-attention for Arabic machine translation[J]. International Journal of Intelligent Computing and Cybernetics, 2024, 17(3): 605-631.
[8] Xu J. Multi-region English translation synchronization mechanism driven by big data[J]. Evolutionary Intelligence, 2023, 16(5): 1539-1546.
[9] Wang S. Accessing higher dimensions for unsupervised word translation[J]. Advances in Neural Information Processing Systems, 2023, 36(1): 69098-69116.
[10] Zhipeng Z, Aleksey P. Research on the Development of Data Augmentation Techniques in the Field of Machine Translation[J]. International Journal of Open Information Technologies, 2023, 11(5): 33-40.
[11] Liu L, Zhu M. Bertalign: Improved word embedding-based sentence alignment for Chinese–English parallel corpora of literary texts[J]. Digital Scholarship in the Humanities, 2023, 38(2): 621-634.
[12] Jiang Y, Niu J. A corpus-based search for machine translationese in terms of discourse coherence[J]. Across Languages and Cultures, 2022, 23(2): 148-166.
[13] Minyun X. Machine Translation Based on Neural Network: Challenge or Chance?[J]. Educalitra: English Education, Linguistics, and Literature Journal, 2023, 2(2): 41-51.