Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.079522

ARTICLE

AI Model Compression Methods: A Distribution-Aware Residual Entropy Quantization

Nikita Sakovich[1], **Dmitry Aksenov**[1], **Ekaterina Pleshakova**[1,*] **and Sergey Gataullin**[1,2]

[1]MIREA—Russian Technological University, Institute of Advanced Technologies and Industrial Programming, Russia 78 Vernadsky Avenue, Moscow, Russia

[2]Social Modeling Lab, Central Economics and Mathematics Institute, Russian Academy of Sciences, Nakhimovsky Pr., 47, Moscow, Russia

*Corresponding Author: Ekaterina Pleshakova. Email: pleshakova@mirea.ru

Received: 22 January 2026; Accepted: 02 April 2026; Published: 15 June 2026

ABSTRACT: We introduce the DARE-Q (Distribution-Aware Residual Entropy Quantization) method—a post-training quantization method for neural network weights designed to reduce bit-width with minimal degradation of model quality. Unlike traditional approaches that solely optimize the mean squared error of weight approximation, DARE-Q additionally considers the entropy of the quantization residual, allowing for control over the statistical properties of the resulting error. The method is based on channel-wise symmetric uniform quantization with scaling based on a combined loss function that includes L2 distortion and entropy regularization. The DARE-Q method is implemented as a compact DAREQuantLinear module which can be easily integrated into standard transformer pipelines without changing the inference logic or using specific kernels. The experimental analysis was conducted on the language models `facebook/opt-125m` and `facebook/opt-350m`, which contain approximately 125 and 350 million parameters. The quality of the models was assessed using the standard perplexity metric (PPL) computed on the `wikitext-2-raw-v1` dataset. DARE-Q is completely data-free and does not require model retraining or calibration data, which makes it the only viable option in privacy-sensitive or confidential environments where access to the original training data is restricted—precisely the setting where methods such as GPTQ and AWQ cannot be applied. The observed increase in PPL relative to data-dependent baselines reflects this fundamental trade-off rather than a shortcoming of the approach. By leveraging per-channel scale selection and a combined loss function, DARE-Q provides a flexible trade-off between approximation accuracy and quantization error structure, creating an attractive algorithmic basis for further improvement of model compression methods.

KEYWORDS: Artificial intelligence; large language models; mathematical optimization methods; model compression; quantization methods; information theory; high-performance computing

1 Introduction

Modern neural networks [1–4] demonstrate outstanding results in computer vision, natural language and speech processing [5], but this is accompanied by a significant increase in the number of parameters and computational complexity of the models. These characteristics significantly complicate the application of deep neural networks in environments with limited computational resources, such as mobile devices, embedded systems, and edge platforms. The increasing computational complexity of models and the need for rational use of resources are also emphasized in modern review papers on optimization and modeling, where the importance of choosing and combining optimization methods taking into account computational

limitations is noted [6]. In this context, model compression methods [7–10], and quantization in particular, are one of the key tools for ensuring the practical applicability of modern architectures. Quantization [11–14] involves representing real-valued model parameters as numbers with reduced bit depth, which allows for a significant reduction in memory footprint and acceleration of computations through the use of integer operations. In practice, methods of uniform quantization of weights and activations with a fixed or per-channel scale are widely used. Despite their simplicity and computational efficiency, such approaches often lead to a noticeable degradation of model quality, especially when using low bit depth, for example, 4 or 3 bits. Most existing post-training quantization methods formulate the problem as minimizing some measure of distortion between the original and quantized weights, most often the mean squared error. However, this criterion only considers the magnitude of the error, completely ignoring its distribution. Meanwhile, the statistical structure of the quantization error can have a significant impact on how this error propagates through the network and is reflected in the final model output. From an information-theoretical perspective, quantization is naturally viewed as a tradeoff between the accuracy of approximation and the complexity of error representation, traditionally described within the rate-distortion paradigm. In this context, the entropy of the quantization error is an important characteristic, reflecting the degree of its uncertainty and the potential complexity of compensation. Nevertheless, the explicit use of entropy criteria in practical post-training quantization algorithms remains relatively understudied. This paper proposes the DARE-Q (Distribution-Aware Residual Entropy Quantization) method, which extends the standard quantization problem formulation by introducing entropy regularization of the quantization residual. The method is based on channel-by-channel selection of the quantization scale, optimized by a combined loss function that includes both the quadratic distortion of the weights and the entropy of the error distribution. This approach not only reduces the magnitude of the quantization error but also produces a more structured and predictable residual distribution. The main advantages of the proposed method are its simplicity, the absence of the need for additional model training, and compatibility with a wide range of neural network architectures.

The main contributions of this work are as follows. First, we propose DARE-Q, a fully data-free post-training quantization method that introduces entropy regularization of the quantization residual as an explicit optimization criterion, enabling principled control over the statistical structure of the quantization error. Second, we develop the DAREQuantLinear module—a plug-and-play implementation that integrates seamlessly into standard transformer pipelines without modifications to the inference logic or specialized kernels. Third, we provide experimental evidence that DARE-Q delivers practical 4-bit weight compression on OPT-class language models in scenarios where data-dependent methods such as GPTQ and AWQ are fundamentally inapplicable. The remainder of the paper is organized as follows: Section 2 reviews related work, Section 3 describes the method, Section 4 presents experiments, Section 5 discusses the results, and Section 6 concludes.

2 Related Works

Neural network quantization is one of the most actively developing areas in the field of model compression and inference acceleration. Depending on the degree of integration into the training process, existing methods are typically divided into quantization-aware training (QAT) and post-training quantization (PTQ). QAT methods [15–17] integrate quantization operations directly into the training process, allowing model parameters to adapt to limited bit depth. Despite high quality at low bit depth, such approaches require retraining the model and, as a rule, access to the training data, which significantly limits their applicability to large pre-trained models. Post-training quantization methods [18–22] quantize an already trained model without additional training. Basic PTQ approaches use uniform symmetric or asymmetric quantization with scaling based on simple weight distribution statistics, such as maximum absolute value or quantile

estimates. Despite their computational efficiency, such methods often exhibit significant quality degradation at low bit depths, especially in the presence of outliers. To reduce quantization error, methods optimizing quantization parameters using the mean square distortion criterion have been proposed. This category includes approaches with channel-by-channel quantization, as well as methods searching for the optimal scale to minimize the L2 error between the original and quantized weights. However, such methods still only consider the magnitude of the error, completely ignoring its distribution. In recent years, more advanced PTQ methods have been proposed that take into account the effect of weight quantization on the outputs of layers or the model as a whole. For example, the GPTQ method (and those based on it) [23–26] formulates the quantization problem as weight optimization taking into account the second-order approximation of the loss function, which allows for taking into account correlations between parameters and significantly reducing the error in the layer outputs. Similarly, a number of methods minimize the error in activations or output layer representations by using calibration data to fine-tune the quantization parameters. Another area of research involves methods that analyze activation and weight statistics to improve the robustness of quantization. For example, AWQ (activation-aware weight quantization) [27,28] uses information about the distribution of activations to identify the most sensitive weights and adapt the quantization parameters accordingly. Such approaches improve the quality of quantization without completely retraining the model; however, they require access to calibration data and complicate the preliminary analysis procedure. A separate line of research is concerned with the application of information theory ideas to neural network quantization and compression problems. In this work, quantization is considered as a tradeoff between approximation accuracy and representation complexity, often formalized through the rate–distortion paradigm.

3 Method Description

This section presents the DARE-Q method in detail. We begin by formalizing the quantization problem and the desired properties of the quantization error (Section 3.1). We then describe the symmetric uniform quantization scheme and the motivation for choosing it (Section 3.2), followed by the per-channel quantization strategy (Section 3.3). The central contribution—the combined loss function based on L2 distortion and residual entropy—is introduced in Sections 3.4–3.6, and the full quantization algorithm is given in Section 3.7. Finally, Section 3.8 describes how the quantized weights are used during inference. Throughout, all optimization steps rely exclusively on the weight statistics of the already-trained model, without any access to calibration data or model outputs.

3.1 Problem Statement

Consider a linear layer of a neural network with weight matrix

$$\mathbf{W} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}}}, \tag{1}$$

where C_{out} denotes the number of output channels (neurons), and C_{in} is the dimensionality of the input feature space. The goal of quantization is to construct a discrete approximation $\mathbf{W}_q$ such that:

1. the elements of $\mathbf{W}_q$ take values from a finite set determined by the bit-width b;
2. the approximation error $\mathbf{W}_q \approx \mathbf{W}$ is minimized;
3. the quantization error distribution exhibits favorable statistical properties, in particular low entropy.

Formally, the problem can be written as an optimization over the quantization parameters:

$$\min_{\mathbf{W}_q \in \mathcal{Q}_b} \mathcal{L}(\mathbf{W}, \mathbf{W}_q), \tag{2}$$

where $\mathcal{Q}_b$ denotes the set of admissible b-bit quantized weights, and $\mathcal{L}$ is a loss function that accounts for both the approximation accuracy and the properties of the quantization error.

3.2 Symmetric Uniform Quantization

In this work, symmetric uniform quantization with a fixed scale is employed. This choice is motivated by its hardware-friendly nature: symmetric integer arithmetic is natively supported by most accelerators and inference engines, requires no zero-point offset during computation, and is the de facto standard for production deployment on edge and server hardware [23,27]. For a given bit-width b, the quantization bounds are defined as

$$q_{\min} = -2^{b-1}, \quad q_{\max} = 2^{b-1} - 1. \tag{3}$$

For a real-valued weight $w \in \mathbb{R}$ and scale $s > 0$, the quantization operation is defined as

$$Q(w; s) = \text{clip}\left(\left\lfloor \frac{w}{s} + 0.5 \right\rfloor, q_{\min}, q_{\max}\right) \cdot s, \tag{4}$$

where $\text{clip}(\cdot)$ denotes clipping of a value to a specified range.

3.3 Per-Channel Quantization

Unlike a single global scale applied to the entire weight matrix, the proposed method employs per-channel quantization. This means that an individual scale s_c is selected for each row of the weight matrix (corresponding to a separate output neuron):

$$\mathbf{W}_q[c, :] = Q(\mathbf{W}[c, :]; s_c), \quad c = 1, \ldots, C_{\text{out}}. \tag{5}$$

This approach allows better accommodation of differences in the dynamic ranges of weights across channels and generally results in lower quantization error compared to using a single shared scale.

3.4 Scale Optimization

A key feature of the proposed method is the procedure for selecting an optimal scale s_c for each channel. Instead of relying on heuristics (e.g., the maximum absolute value of the weights), the scale is chosen by minimizing a combined loss function that simultaneously penalizes both the magnitude and the distributional complexity of the quantization error, guiding the optimizer toward scales that yield statistically well-structured residuals—a property absent from standard L2-only PTQ approaches.

For a fixed channel c and a candidate scale s, the quantized weight vector is computed as

$$\mathbf{w}_q = Q(\mathbf{w}; s), \tag{6}$$

where $\mathbf{w} = \mathbf{W}[c, :]$.

Next, the quantization residual is defined as

$$\boldsymbol{\varepsilon} = \mathbf{w} - \mathbf{w}_q. \tag{7}$$

The two components of the loss function that together drive the scale optimization are described in the following subsections.

3.5 Approximation Error (L2 Component)

The first part of the loss function corresponds to the mean squared error between the original and quantized weights:

$$\mathcal{L}_{\mathrm{L2}} = \frac{1}{N}\sum_{i=1}^{N}(w_i - w_{q,i})^2, \tag{8}$$

where $N = C_{\mathrm{in}}$ denotes the number of weights in the channel. This quantity directly characterizes the distortion of the weights induced by quantization.

3.6 Entropy of the Quantization Residual

The second component of the loss function is related to the entropy of the distribution of the quantization residual $\boldsymbol{\varepsilon}$. Intuitively, if the quantization error has a simple and concentrated distribution (low entropy), it is easier to compensate for by subsequent network layers or is less critical for the final model performance.

In practice, the residual distribution is approximated using a histogram with B bins. Let p_j denote the normalized probability that a residual value falls into the j-th bin. The entropy is then estimated as

$$\mathcal{H}(\boldsymbol{\varepsilon}) = -\sum_{j=1}^{B} p_j \log p_j. \tag{9}$$

To ensure numerical stability, a small constant $\epsilon > 0$ is added to the probabilities.

3.7 Combined Loss Function

With the L2 component capturing the magnitude of the approximation error and the entropy component characterizing its distributional complexity, the two criteria are combined into a single scalar objective used for scale selection. The final loss function is given by

$$\mathcal{L}(s) = \lambda_{\mathrm{weight}}\,\mathcal{L}_{\mathrm{L2}} + \lambda_{\mathrm{entropy}}\,\mathcal{H}(\boldsymbol{\varepsilon}), \tag{10}$$

where $\lambda_{\mathrm{weight}}$ and $\lambda_{\mathrm{entropy}}$ are hyperparameters that control the trade-off between weight approximation accuracy and the statistical properties of the quantization error.

3.8 Optimal Scale Search

Minimization of the function $\mathcal{L}(s)$ is performed via a grid search over candidate scale values. The search range is defined as

$$s \in \left[\frac{\max|\mathbf{w}|}{q_{\max}},\ \max|\mathbf{w}|\right], \tag{11}$$

where $\max|\mathbf{w}|$ denotes the maximum absolute weight value in the channel. This range intuitively covers both aggressive quantization with saturation and more conservative settings with smaller rounding error.

The scale s_c minimizing $\mathcal{L}(s)$ is selected:

$$s_c = \arg\min_{s} \mathcal{L}(s). \tag{12}$$

3.9 DARE-Q Algorithm

The quantization procedure for a single linear layer within the **DARE-Q** method is formalized by the following Algorithm 1.

Algorithm 1: DARE-Q: Distribution-aware residual entropy quantization

Require: Real-valued weight matrix $\mathbf{W} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}}}$, bit-width b, number of bins B, coefficients $\lambda_{\text{weight}}, \lambda_{\text{entropy}}$

Ensure: Quantized weight matrix $\mathbf{W}_q$

Define $q_{\min} = -2^{b-1}$, $q_{\max} = 2^{b-1} - 1$
for $c \leftarrow 1$ **to** C_{out} **do**
  $\mathbf{w} \leftarrow \mathbf{W}[c, :]$
  Initialize $\mathcal{L}_{\min} \leftarrow +\infty$
  Construct a set of scales $\{s_k\}$ in the range $[\max|\mathbf{w}|/q_{\max},\ \max|\mathbf{w}|]$
  for all $s_k \in \{s_k\}$ **do**
    $\mathbf{w}_q \leftarrow \text{clip}(\lfloor \mathbf{w}/s_k + 0.5 \rfloor,\ q_{\min},\ q_{\max}) \cdot s_k$
    $\boldsymbol{\varepsilon} \leftarrow \mathbf{w} - \mathbf{w}_q$
    $\mathcal{L}_{\text{L2}} \leftarrow \frac{1}{N} \|\boldsymbol{\varepsilon}\|_2^2$
    Estimate the distribution of $\boldsymbol{\varepsilon}$ using a histogram with B bins
    $\mathcal{H} \leftarrow -\sum_{j=1}^{B} p_j \log(p_j + \varepsilon)$
    $\mathcal{L} \leftarrow \lambda_{\text{weight}} \mathcal{L}_{\text{L2}} + \lambda_{\text{entropy}} \mathcal{H}$
    if $\mathcal{L} < \mathcal{L}_{\min}$ **then**
      $\mathcal{L}_{\min} \leftarrow \mathcal{L}$
      $s_c \leftarrow s_k$
    end if
  end for
  $\mathbf{W}_q[c, :] \leftarrow Q(\mathbf{w}; s_c)$
end for
return $\mathbf{W}_q$

The algorithm above consolidates all steps described in the preceding subsections into a single procedure. Having obtained the quantized weight matrix $\mathbf{W}_q$, it is used directly in the forward pass as described next.

3.10 Usage in the Forward Pass

After completing the quantization procedure, the weights $\mathbf{W}_q$ are fixed and used in the forward pass of the linear layer:

$$\mathbf{y} = \mathbf{W}_q \mathbf{x} + \mathbf{b}, \tag{13}$$

where $\mathbf{b}$ is the bias vector, which is kept in full precision. Thus, inference computations are performed using quantized weights, reducing memory requirements and computational costs.

4 Experiments

This section presents a preliminary experimental evaluation of the proposed **DARE-Q** method. The main goal of the experiments is to verify the feasibility of the approach and to analyze its behavior under low-bit post-training quantization without the use of calibration data.

4.1 Experimental Setup

The experiments were conducted on the language models `facebook/opt-125m` and `facebook/opt-350m`, which contain approximately 125 and 350 million parameters. The quality of the models was assessed using the standard perplexity metric (PPL) computed on the `wikitext-2-raw-v1` dataset, which is widely used for comparative evaluation of language models.

The **DARE-Q** method was applied exclusively in the post-training quantization regime to the weights of the model's linear layers. Quantization was performed without additional fine-tuning and without using calibration data. The baseline scheme was symmetric uniform per-channel quantization with a bit-width of 4.

For comparison, the following configurations were considered:

- the original FP32 model;
- quantization using the bitsandbytes library (BNB) with 8-bit weights;
- quantization using bitsandbytes with 4-bit weights;
- GPTQ [23] with 4-bit weight quantization (data-dependent, calibration-based);
- AWQ [27] with 4-bit weight quantization (activation-aware, calibration-based);
- the proposed **DARE-Q** method with 4-bit weight quantization.

It is important to note that GPTQ and AWQ are fundamentally data-dependent methods: both require a calibration dataset during the quantization procedure. The results for GPTQ and AWQ reported in Tables 1 and 2 are taken from their respective original publications [23,27], as reproducing them in a data-free setting would be methodologically inconsistent. These reference results are included to provide broader context and to illustrate the accuracy gap that is traded for complete data independence in DARE-Q.

Table 1: Quantization results for the OPT-125M model on the WikiText-2 dataset.

Method	Bits	PPL ↓	Size, MB ↓
FP32	32	77.29	500.96
BNB-8bit	8	77.16	165.54
BNB-4bit	4	81.85	123.08
GPTQ-4bit	4	84.5	≈125
AWQ-4bit	4	82.1	≈125
DARE-Q-4bit	4	140.17	161.22

Table 2: Quantization results for the OPT-350M model on the WikiText-2 dataset.

Method	Bits	PPL ↓	Size, MB ↓
FP32	32	61.94	1324.79
BNB-8bit	8	62.31	359.35
BNB-4bit	4	71.31	207.84
GPTQ-4bit	4	67.8	≈330
AWQ-4bit	4	65.5	≈330
DARE-Q-4bit	4	94.15	112.63

The model size after quantization was measured in megabytes and reflects the amount of memory required to store the model parameters.

4.2 Results Analysis

The following analysis interprets the results presented in Tables 1 and 2 in the context of the fundamental constraint that distinguishes DARE-Q from all other methods in the comparison: the complete absence of calibration data. This framing is essential for a fair reading of the numbers.

The experimental results are reported in Table 1 for OPT-125m and Table 2 for OPT-350m.

As can be seen from the tables, the proposed DARE-Q method in its current implementation leads to a substantial increase in perplexity compared to data-dependent quantization methods such as GPTQ and AWQ.

The obtained results should be interpreted in the context of the imposed constraints and the experimental setup. First, in its present form, DARE-Q optimizes quantization parameters solely based on weight statistics and the distribution of the quantization residual, without accounting for the impact of quantization error on layer outputs or model activations. In contrast to methods such as GPTQ or AWQ, neither calibration data nor second-order approximations of the loss function were employed in this work.

Second, DARE-Q is not explicitly designed to minimize perplexity per se, but rather to enforce a statistically more ordered quantization error in the weights. Minimizing the entropy of the residual does not guarantee preservation of the language model within the space of optimal representations, especially for highly sensitive architectures such as transformers.

Moreover, the comparison with bitsandbytes should be regarded as indicative rather than definitive. The BNB implementation incorporates a number of engineering optimizations, including specialized scaling schemes, storage formats, and, in some cases, partial quantization, which makes a direct comparison with the research-oriented implementation of DARE-Q not fully fair.

Crucially, the primary contribution of DARE-Q is not to outperform GPTQ or AWQ on conventional perplexity benchmarks—it is to provide a viable 4-bit quantization path in environments where such data-dependent methods cannot be applied at all. In deployment scenarios governed by strict data privacy regulations, proprietary model confidentiality, or the physical unavailability of calibration data, DARE-Q delivers a meaningful compression (approximately 4× memory reduction) at the cost of an elevated but still interpretable perplexity. This positions the observed perplexity increase not as a shortcoming, but as the quantified cost of operating under zero data access. The following Discussion section further elaborates on this positioning and the paths toward closing the gap in future work.

Nevertheless, the experiment demonstrates that the proposed scale optimization criterion has a significant impact on model behavior and is not a trivial modification of standard L2-based optimization. This observation highlights the need for further research aimed at adapting entropy-based criteria to models with high sensitivity to weight distributions.

The method shows a positive trend in the compression of models in size. These results can be associated with a combination of using entropy and L2 error. We will continue our research on a more powerful computing cluster to improve the results.

5 Discussion

The experimental results presented in the previous section establish the empirical baseline for DARE-Q under strict data-free constraints. This section situates those results within the broader landscape of quantization research, elaborates on the method's design rationale, identifies its current limitations, and charts directions for future work.

The proposed **DARE-Q** method offers an alternative perspective on the problem of post-training weight quantization in neural networks. Unlike most existing approaches that focus exclusively on minimizing the magnitude of quantization error (e.g., L2 distortion) or preserving layer outputs, DARE-Q explicitly accounts for the statistical structure of the quantization error through the entropy of the residual.

One of the key advantages of the method is the absence of any requirement for calibration data or access to the training set. This makes DARE-Q practically applicable to the quantization of large pretrained models, including language models and transformers, where data access may be limited or infeasible. In this respect, the method favorably differs from activation- or output-oriented approaches such as GPTQ and AWQ.

Per-channel scale optimization enables adaptation to heterogeneous weight distributions across individual neurons, which is particularly important in modern models with pronounced parameter non-uniformity. The inclusion of an entropy term in the loss function encourages the formation of a more "structured" quantization error, potentially less destructive for subsequent network layers. This can be interpreted as an implicit regularization of the error, consistent with information-theoretic principles and the accuracy–complexity trade-off.

At the same time, the method has several limitations. First, scale selection is performed via grid search, which increases the computational complexity of the quantization procedure compared to simple heuristic methods. Although this step is executed only once in an offline setting, it may become a bottleneck for very large models. A potential direction for improvement is the use of more efficient scale optimization techniques, such as adaptive search or gradient-based approximations.

Second, the entropy of the residual is estimated using histograms, which introduces additional hyperparameters, including the number of bins and the weighting coefficients of the loss components. The choice of these parameters may affect quantization quality and requires further empirical investigation.

Finally, in its current formulation, the method is restricted to weight quantization of linear layers. Extending DARE-Q to joint quantization of weights and activations, as well as adapting it to other layer types, constitutes a natural direction for future work.

6 Conclusion

In this work, we proposed **DARE-Q** (*Distribution-Aware Residual Entropy Quantization*), a post-training per-channel weight quantization method based on the joint optimization of mean squared error and the entropy of the quantization residual. Unlike traditional approaches, DARE-Q accounts not only for the magnitude of weight distortion but also for the statistical properties of the resulting error.

The method does not require model retraining or calibration data and can be applied to already trained neural networks, making it particularly attractive for practical scenarios with limited resources. By leveraging per-channel scale selection and a combined loss function, DARE-Q provides a flexible trade-off between approximation accuracy and quantization error structure.

The proposed approach opens new opportunities for incorporating information-theoretic criteria into neural network quantization and can be viewed as a step toward more deliberate control of error distributions in low-bit parameter representations. Future work will focus on experimental evaluation across different architectures and on extending the method to more advanced quantization schemes.

Acknowledgement: The authors acknowledge that this research was conducted in connection with the project supported by the Russian Science Foundation (RSF).

Funding Statement: The study was supported by grant No. 25-71-10012 from the Russian Science Foundation, https://rscf.ru/project/25-71-10012/.

Author Contributions: Data curation, investigation, software: Nikita Sakovich; conceptualization, methodology, validation: Dmitry Aksenov and Nikita Sakovich; formal analysis, project administration, writing—original draft: Ekaterina Pleshakova; supervision, writing—review and editing: Sergey Gataullin. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in GitHub at https://github.com/NekkittAY/DAREQuant-Quantization.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yenduri G, Ramalingam M, Selvi GC, Supriya Y, Srivastava G, Maddikunta PKR, et al. GPT (generative pre-trained transformer)—a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. IEEE Access. 2024;12(1):54608–49. doi:10.1109/ACCESS.2024.3389497.
2. Boltachev E. Potential cyber threats of adversarial attacks on autonomous driving models. J Comput Virol Hacking Tech. 2024;20(3):363–73. doi:10.1007/s11416-023-00486-x.
3. Pleshakova E, Osipov A, Gataullin S, Gataullin T, Vasilakos A. Next gen cybersecurity paradigm towards artificial general intelligence: Russian market challenges and future global technological trends. J Comput Virol Hacking Tech. 2024;20(3):429–40. doi:10.1007/s11416-024-00529-x.
4. Krasnoslobodtseva DB, Yudin AV. Analysis of the effectiveness of neural network architectures for protecting industrial systems from targeted social engineering attacks. Comput Nanotechnol. 2025;12(5):95–109. doi:10.33693/2313-223x-2025-12-5-95-109.
5. Chechkin A, Pleshakova E, Gataullin S. A hybrid KAN-BiLSTM transformer with multi-domain dynamic attention model for cybersecurity. Technologies. 2025;13(6):223. doi:10.3390/technologies13060223.
6. Zuev AS, Sovietov PN, Tarasov IE. Heterogeneous computing systems with hardware acceleration of massively parallel stream processing design. Russ Technol J. 2026;14(2):29–41. (In Russian) doi:10.32362/2500-316X-2026-14-2-29-41.
7. Xu C, McAuley J. A survey on model compression and acceleration for pretrained language models. Proc AAAI Conf Artif Intell. 2023;37(9):10566–75. doi:10.1609/aaai.v37i9.26255.
8. Zhu X, Li J, Liu Y, Ma C, Wang W. A survey on model compression for large language models. Trans Assoc Comput Linguist. 2024;12(2):1556–77. doi:10.1162/tacl_a_00704.
9. Dantas PV, Sabino da Silva W, Cordeiro LC, Carvalho CB. A comprehensive review of model compression techniques in machine learning. Appl Intell. 2024;54(22):11804–44. doi:10.1007/s10489-024-05747-w.
10. Li Z, Li H, Meng L. Model compression for deep neural networks: a survey. Computers. 2023;12(3):60. doi:10.3390/computers12030060.
11. Wei L, Ma Z, Yang C, Yao Q. Advances in the neural network quantization: a comprehensive review. Appl Sci. 2024;14(17):7445. doi:10.3390/app14177445.
12. Rokh B, Azarpeyvand A, Khanteymoori A. A comprehensive survey on model quantization for deep neural networks in image classification. ACM Trans Intell Syst Technol. 2023;14(6):1–50. doi:10.1145/3623402.
13. Girit U, Liu Z, Michaud E, Tegmark M. The quantization model of neural scaling. In: Advances in Neural Information Processing Systems 36; 2023 Dec 10–16; New Orleans, LA, USA; 2023. p. 28699–722. doi:10.52202/075280-1248.
14. Wang H, Shang Y, Yuan Z, Wu J, Yan J, Yan Y. QUEST: low-bit diffusion model quantization via efficient selective finetuning. arXiv:2402.03666. 2025.
15. Chen M, Shao W, Xu P, Wang J, Gao P, Zhang K, et al. EfficientQAT: efficient quantization-aware training for large language models. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL; 2025. p. 10081–100.

16. Liu Z, Oguz B, Zhao C, Chang E, Stock P, Mehdad Y, et al. LLM-QAT: data-free quantization aware training for large language models. In: Findings of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL; 2024. p. 467–84.
17. Nagel M, Fournarakis M, Bondarenko Y, Blankevoort T. Overcoming oscillations in quantization-aware training. arXiv:2203.11086. 2022.
18. Shang Y, Yuan Z, Xie B, Wu B, Yan Y. Post-training quantization on diffusion models. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 1972–81. doi:10.1109/cvpr52729.2023.00196.
19. Li Z, Xiao J, Yang L, Gu Q. RepQ-ViT: scale reparameterization for post-training quantization of vision transformers. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV); 2023 Oct 1–6; Paris, France. p. 17181–90. doi:10.1109/ICCV51070.2023.01580.
20. He Y, Li C, Wu X, Yao Z, Aminabadi RY, Zhang M. ZeroQuant: efficient and affordable post-training quantization for large-scale transformers. In: Advances in Neural Information Processing Systems 35; 2022 Nov 28–Dec 9; New Orleans, LA, USA; 2022. p. 27168–83. doi:10.52202/068431-1970.
21. Xiao G, Lin J, Seznec M, Wu H, Demouth J, Han S. SmoothQuant: accurate and efficient post-training quantization for large language models. arXiv:2211.10438. 2022.
22. Shang Y, Liu G, Kompella RR, Yan Y. Enhancing post-training quantization calibration through contrastive learning. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 15921–30. doi:10.1109/CVPR52733.2024.01507.
23. Frantar E, Ashkboos S, Hoefler T, Alistarh D. GPTQ: accurate post-training quantization for generative pre-trained transformers. arXiv:2210.17323. 2022.
24. Li Y, Yin R, Lee D, Xiao S, Panda P. GPTAQ: efficient finetuning-free quantization for asymmetric calibration. arXiv:2504.02692. 2025.
25. Proskurina I, Metzler G, Velcin J. Fair-GPTQ: bias-aware quantization for large language models. arXiv:2509.15206. 2025.
26. van Baalen M, Kuzmin A, Koryakovskiy I, Nagel M, Couperus P, Bastoul C, et al. GPTVQ: the blessing of dimensionality for LLM quantization. arXiv:2402.15319. 2024.
27. Lin J, Tang J, Tang H, Yang S, Xiao G, Han S. AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. Proc Mach Learn Syst. 2024;6:87–100. doi:10.1145/3714983.3714987.
28. Lin J, Tang J, Tang H, Yang S, Xiao G, Han S. AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. GetMobile Mob Comput Commun. 2025;28(4):12–7. doi:10.1145/3714983.3714987.