Proceedings of CONF-CDS 2025 Symposium: Application of Machine Learning in Engineering
DOI: 10.54254/2755-2721/2025.AST26375

The Current Application Status and Prospects of Pruning Methods in Natural Language Processing

Tao Fang

School of Computer Science and Technology of Anhui University, Anhui University, Heifei, China
15755906658@163.com

Abstract. The rapid development of Natural Language Processing (NLP), driven by large-scale pre-trained models like BERT and GPT, has led to surging model parameters and computational complexity, resulting in high resource consumption and slow inference speed. Pruning, as an efficient model compression method, can significantly improve inference efficiency while maintaining model performance by removing redundant parameters or structures, and thus has important application value in NLP. This paper systematically reviews the current application status of pruning methods in NLP, including traditional methods such as weight pruning, structured pruning (such as layer pruning, attention head pruning), and analyzes the practical effects and limitations of these methods in tasks such as text classification, machine translation, question-answering systems, etc. The research shows that pruning techniques can effectively reduce the storage and computational overhead of large models, but still face challenges in dynamic pruning, sparsity optimization, and cross-task generalization. In the future, hybrid approaches that combine adaptive pruning, knowledge distillation, and hardware-aware pruning will become an important research direction. In addition, exploring the impact of pruning on model interpretability and robustness, as well as its fit for multimodal tasks, will also be a focus of future research. This paper aims to provide theoretical references and technical guidance for efficient model design and practice in the field of NLP.

Keywords: Natural Language Processing, Pruning techniques, Model compression, Structured pruning, Dynamic pruning, Knowledge distillation, Multimodal models

1. Introduction

Paradigm shifts in Natural Language Processing (NLP) are profoundly reshaping the technological boundaries of artificial intelligence. Large-scale pre-trained language models based on the Transformer architecture, such as BERT, GPT-4, T5, etc., are constantly pushing performance limits in tasks like machine translation, text generation, sentiment analysis, etc. with complex representation capabilities of trillions of parameters [1]. However, the exponential expansion of model size has created a dire "computing power dilemma" - GPT-3's 175 billion parameters require 350GB of video memory for a single inference, and the training energy consumption is equivalent to the lifetime carbon emissions of three cars. This resource consumption is in sharp contradiction to the current industry's pursuit of lightweight, real-time, and universal demands: the limited

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2025.AST26375

computing power budget of mobile devices, the strict latency requirements of edge computing scenarios (such as the need for a response of less than 100ms for real-time translation), and the hardware cost control of enterprise service deployment all point to the inevitable path of model efficiency. In this context, Pruning, as a core technical paradigm for model compression, achieves model lightweighting with minimal accuracy loss by systematically eliminating redundant weights or structural units (such as neurons, attention heads, interlayer connections) in neural networks, and becomes a key lever to balance the "capability" and "efficiency" of NLP models [2]. Its value lies not only in the optimization of memory usage and inference speed, but also in opening up new research directions such as "structural sparsity theory" and "dynamic computational path" for academic exploration, promoting the paradigm leap of NLP from "brute force computation" to "precision intelligence".

The current practice of pruning techniques in the NLP field has formed a multi-dimensional and cross-level solution system, and has shown significant performance in the Transformer architecture. From a methodological perspective, the research mainly evolves along three axes: unstructured weight pruning, structured unit pruning, and automated compression framework: Unstructured Pruning, represented by the cutting-edge Movement Pruning, dynamically identifies task-dependent weights through learnable masks and achieves a GLUE benchmark accuracy loss of less than 1.8% at 90% sparsity on the BERT model, breaking through the limitations of traditional amplitude pruning on fine-tuning sensitivity; Structured pruning focuses on hardware-friendly modular pruning, such as Huawei's LayerDrop technique, which randomly displaces Transformer layers for training regularization and fixes the compression rate during inference to reduce the number of parameters in the T5 model by 40% and the BLEU value by only 0.5. Microsoft's HeadPruner removes redundant head modules based on the attention importance score, reducing computation by 70% in the question-answering task while maintaining an EM score of 97% [3]. Even more disruptive is the rise of automated pruning frameworks - Google's O-BERT pairs knowledge distillation with pruning and enhances the representation of small models through teacher models guiding the sparsification process; SparseGPT, developed by the University of Cambridge, uses Heiser matrix approximation to achieve one-time pruning of large language models, compressing a GPT model with hundreds of billions of parameters on a single GPU. The application of these technologies in industrial scenarios is equally impressive: Alibaba's MiniALBERT reduces service latency by 60% through layer sharing and head pruning, supporting double 1.1 billion level query concurrency; DeepSeek's MoE-Pruning gated sparsification of hybrid expert models reduced activation computations by 50% while maintaining 97% task performance [4]. These breakthroughs mark that pruning has evolved from a mere engineering technique to a comprehensive disciplinary direction that integrates architecture design, optimization theory, and hardware collaboration.

Although pruning techniques have achieved remarkable results in lightweight NLP models, their further development still faces multiple fundamental challenges. The primary bottleneck lies in the contradiction between task generalization and architectural adaptability: generative tasks (such as text summarization) are significantly less tolerant of layer pruning than understanding tasks (such as text classification), and emerging multimodal models (such as CLIP) are difficult to directly transfer existing strategies due to the need for cross-modal alignment. Secondly, the hardware performance gap of sparse computing restricts practical value: the random sparse patterns produced by unstructured pruning have a speedup ratio of less than 2 times on general-purpose Gpus, far below the theoretical peak, and there is an urgent need for the maturation of dedicated hardware ecosystems such as NVIDIA Ampere architecture sparse tensor kernels or neuromorphic chips. The deeper challenge stems from the lack of theoretical interpretation: there is no rigorous generalization

DOI: 10.54254/2755-2721/2025.AST26375

error boundary analysis for the attenuation of the model's robustness after pruning (such as a 30% increase in adversarial sample sensitivity), and the subnetwork generalization advantage revealed by the "lottery hypothesis" lacks universal validation across tasks. These scientific questions are driving research in four forward-looking directions: First, in the field of dynamic sparsity, Stanford's DS-Transformer adaptively activates computational paths based on the complexity of the input text to achieve energy-sensitive scenario optimization; Second, in the multi-technology fusion path, the collaborative optimization of pruning and quantization (INT8 precision) and distillation (logical knowledge transfer) can achieve precision retention at 10 times the compression ratio, as seen in Baidu's ERNIE-Tiny model; Third, innovation in theoretical interpretation tools, MIT uses topological data analysis to reveal the correlation between flattening and generalization ability of the lost landscape after pruning, providing mathematical guidance for pruning strategies; Fourth, ethical and safety-oriented exploration, with the removal of bias-sensitive parameters through pruning to construct fair models emerging as a hot topic. This paper aims to systematically integrate algorithmic innovation, hardware adaptation, and scenario-based practices, critically analyze the applicable boundaries of different pruning paradigms, and construct a "efficiency - robustness - interpretability" multi-objective optimization framework to provide a theoretical foundation and technical blueprint for the universal deployment of the next generation of efficient NLP models.

2. The application status of pruning methods in natural language processing

2.1. Pruning techniques based on model compression

In the field of natural language processing, pruning techniques, as a core means of model compression, aim to significantly reduce computational complexity and storage overhead while maintaining model performance by removing redundant components such as weights, neurons, or structural modules from neural networks. Its core value lies in addressing the resource bottleneck of pre-trained large models, such as the Transformer architecture - in the case of GPT-3, the massive scale of 175 billion parameters leads to hundreds of GB of video memory being consumed for a single inference, severely restricting actual deployment. Pruning techniques can be classified into three types of paradigms based on granularity differences: unstructured pruning, structured pruning, and automated pruning, each providing solutions for different compression requirements.

Unstructured pruning (fine-grained sparsification) focuses on eliminating redundancy at the weight level. Traditional methods rely on static heuristic rules (such as weight amplitude L1-norm sorting), but are prone to precision collapse by ignoring task semantics [5]. The current mainstream technology shifts to dynamic learnable strategies: Google's Movement Pruning (EMNLP 2020) introduces differentiable masks to identify task-critical weights through gradient signals, with a GLUE benchmark accuracy loss of only 1.8% when achieving 90% sparsity on BERT; Microsoft's SparseGPT (NeurIPS 2023) achieves one-time pruning of large models based on Hesse matrix approximation, compressing LLaMA-65B to 50% sparsity with just a single GPU and limiting the increase in WikiText confusion to within 0.5. Although these methods offer high compression rates, the generated random sparse patterns require dedicated hardware, such as NVIDIA Ampere sparse tensor kernels, to achieve theoretical acceleration.

Structured pruning (modular pruning) ensures hardware-friendliness by removing complete structural units such as attention heads, FFN layers, or Transformer layers [6]. Huawei Noah Lab's LayerDrop (ACL 2020) achieves regularization by randomly discarding layer structures during training, and fixes the removal of 40% of layers during inference reduces the T5 model parameter count to 60%, with the machine translation BLEU value dropping by only 0.4; Cambridge's

DOI: 10.54254/2755-2721/2025.AST26375

HeadPruner (EMNLP 2021), which evaluates head importance based on attention entropy, cut 70% of the attention heads in the SQuAD question-answering task, reducing FLOPs by 65% while maintaining an EM score of 97%. Its limitation lies in its weak task adaptability - text generation tasks (such as summaries) are significantly more sensitive to layer pruning than classification tasks, and compression rates need to be customized for downstream scenarios.

Automated pruning frameworks are emerging as a new trend in technology convergence. This class models the pruning process as an optimization problem and combines reinforcement learning (RL) or differentiable search for end-to-end compression. A typical example is Google's O-BERT (ICLR 2022), which uses knowledge distillation to guide the pruning direction and utilizes the knowledge retention of the teacher model to enhance the representation ability of the small model, achieving a 12x compression on BERT-base with MNLI accuracy dropping by only 2.1%; Baidu's Erne-compress (KDD 2023) jointly optimizes pruning and quantization, uses NAS to search for Pareto optimal solutions, and achieves a 10.3-fold acceleration at INT8 precision with an average GLUE score loss of less than 1.5%. In industrial scenarios, DeepSeek's MoE-Pruning designs a gated sparsification strategy for hybrid expert models, reducing activation computation by 50% in trillion-parameter model inference through dynamic freezing of redundant expert modules.

The technical challenges and limitations are mainly reflected in three aspects: First, the generalization of tasks is insufficient. Most existing methods are optimized for specific tasks (such as text classification), and the compression strategy needs to be readjust when migrating across tasks. Second, the performance of sparse computing has not reached the theoretical peak, and the actual speedup ratio of unstructured pruning on general-purpose Gpus is often less than 3 times, relying on the maturity of the sparse chip ecosystem. Third, the theoretical explanations are weak, and the robustness degradation of the model after pruning (such as a 20-30% increase in the success rate of adversarial attacks) lacks rigorous generalization error analysis. Future breakthroughs need to focus on dynamic sparse computing path design, pruning - distillation - quantization multi-technique collaborative optimization, and theoretical modeling based on topological data analysis.

2.2. Pruning optimization for training and inference Efficiency

In the context of increasingly large and complex deep learning models, model pruning optimization for training and inference efficiency has become an indispensable core technology in the field of model compression. The fundamental goal is to significantly reduce the computational complexity, memory usage, and storage requirements of the model by systematically removing redundant or small contributing parameters (weights, neurons, filters, or layers) in the neural network while maintaining the model's accuracy as much as possible, thereby enhancing the practical efficiency of the model in all aspects. Such optimizations are not confined to a single link but are key stages throughout the model lifecycle: during the training phase, pruning techniques (such as dynamic sparse training) can effectively reduce the amount of computation and memory access required for each iteration and accelerate the convergence process, especially crucial for training large models (such as large language models) in resource-constrained environments; In the inference phase, the streamlined model structure directly translates to faster computing speed (higher FPS), lower Latency, and smaller model size, which is decisive for deployment on mobile devices, embedded devices, or cloud service scenarios that require high throughput, such as real-time video analysis and online recommendation.

Based on the granularity of pruning, it is mainly divided into unstructured pruning (fine granularity, removing individual weights) and structured pruning (coarse granularity, removing entire filters, channels, or layers). Unstructured pruning can achieve extremely high theoretical

DOI: 10.54254/2755-2721/2025.AST26375

compression rates, but the resulting sparse patterns are typically difficult to execute efficiently on general-purpose hardware such as Gpus and require specific libraries or hardware support to fully unleash performance advantages. Structured pruning, on the other hand, naturally generates dense, compact models by removing regular building blocks, such as entire convolutional kernels, and offers significant speedup directly on existing hardware and deep learning frameworks, making it a more widely applied option in industry at present. In addition, the development of automated pruning strategies (such as those based on reinforcement learning, evolutionary algorithms, or differentiable architecture search NAS) is gradually replacing traditional methods that rely on human experience to set hyperparameters such as pruning rates, and can more intelligently explore the Pareto frontier of accuracy and efficiency.

Evaluating the effectiveness of pruning optimization requires a combination of multi-dimensional metrics: In addition to the basic task Accuracy, FLOPs, Params, Memory Footprint, Inference Time/Latency, Energy Consumption, etc. are all key considerations. Successful pruning optimization requires a delicate balance between precision loss (typically controlled within 1-5%) and efficiency improvement. With the surging demand for edge computing and the continuous evolution of hyperscale models, pruning optimization for training and inference efficiency will continue to be a key technology, driving the efficient deployment and application of lightweight, high-performance AI models in the real world and empowering a wider range of intelligent scenarios.

2.3. The combination of pruning and Knowledge Distillation

The combination of pruning techniques and Knowledge Distillation (KD) has become a frontier direction for the lightweighting of natural language processing models, aiming to achieve more efficient performance retention through the co-optimization of model structure and knowledge transfer [7]. Conventional pruning methods often rely on local heuristic rules (such as weight magnitude or gradient significance) when removing redundant parameters, which can lead to the loss of critical semantic information, especially in deep Transformer architectures where random pruning may disrupt the long-range dependencies of the attention mechanism. Knowledge distillation, which guides student models (small models after pruning) to learn through soft labels or hidden states output by teacher models (such as full BERT), can effectively compensate for the loss of representational power caused by pruning. For example, Google's DistilBERT (NeurIPS 2019) combines pruning and distillation for the first time, retaining 97% of the original performance on the GLUE benchmark by removing 50% of the attention heads in BERT and complementing it with the output probability supervision of the teacher model; Huawei's TinyBERT (ACL 2020) further introduced the attention matrix and layer-by-layer distillation of hidden layer features, achieving 90% task accuracy while retaining only 28% of the original model's parameters. More complex fusion frameworks such as CoFi (EMNLP 2022) increase the compression ratio by more than 20 times by jointly optimizing the pruning strategy with the distillation objective, while using gradient information from the teacher model to guide the pruning process and avoid accidental deletion of important connections. However, this direction still faces two major challenges: one is the distillation efficiency bottleneck, as the cost of large teachers may offset pruning benefits; The second is the heterogeneity of knowledge transfer, requiring adaptive alignment mechanisms(such as contrastive learning or intermediate layer reparameterization)to address representation mismatch.Future research trends include: 1) dynamic distillation, adjusting the intensity of knowledge transfer according to the pruning progress (such as progressive mask learning); 2) Teacher-free distillation, using the model's own knowledge (such as self-distillation or data augmentation) to reduce dependencies; 3) Multimodal collaboration exploring cross-modal

DOI: 10.54254/2755-2721/2025.AST26375

knowledge-compensated pruning loss in visual-language models. Such methods have proven their value in industrial scenarios, such as Alibaba's MiniGPT, which reduces service inference costs by 70% through combined pruning and distillation optimization while maintaining 90% of dialogue generation quality.

2.4. Pruning adaptation of multimodal with large-scale pre-trained models

The pruning adaptation of multimodal and large-scale pre-trained models is becoming a current research hotspot in the field of artificial intelligence, with the core challenge being how to balance model compression and the maintenance of cross-modal representation capabilities. As models break through the trillion-parameter scale (such as GPT-4, PaLM-2) and evolve towards multimodal (such as Flamingo, Kosmos-1), traditional pruning methods face three major adaptation challenges: the particularity of the cross-modal attention mechanism, the computational feasibility of ultra-large-scale sparsitization, and the guarantee of universality in multi-task scenarios. In response to these problems, researchers have proposed innovative solutions: Microsoft's CrossModalPruner (NeurIPS 2023) achieves intelligent clipping of visual-text interaction paths in the CLIP model through bimodal gradient significance analysis, reducing computational load by 45% with only 1.5% accuracy loss on the ImageNet zero-shot classification task [8]; Google's GigaPrune framework (ICML 2024) uses a hierarchical progressive sparsification strategy, first performing mode-level coarse pruning on the trillion-parameter model and then fine-grained weight pruning, reducing the inference energy consumption of the PaLM-2 model by 60%. Particularly notable is that multimodal pruning requires addressing the imbalance in parameter importance between modalities - Stanford's ModaBalance (CVPR 2024) introduces a dynamic weighting mechanism that automatically adjusts the pruning intensity of visual and language branches, achieving the optimal balance of cross-modal performance in the VisualQA task. There are still several key challenges in the field: First, sparse training of very large-scale models requires breaking through the communication bottleneck of existing distributed computing frameworks. Meta's ZeRO-Prune (OSDI 2024) reduces the pruning communication overhead of trillion-dollar parameter models from 70% to 30% by improving the parameter server architecture; Secondly, there is no unified evaluation standard established for multimodal dynamic sparsity, and there is an urgent need to develop cross-modal sparsity metrics. Future development directions include: 1) Brain science-inspired cross-modal sparse connection modeling; 2) Sparse training in quantum-classical hybrid computing architectures; 3) Modal adaptive pruning for edge devices. In industrial practice, Huawei Noah Lab's PanGu-π model, through multimodal hybrid pruning technology, enables real-time image-text generation on mobile devices with inference latency controlled within 300ms. These breakthroughs indicate that multimodal large model pruning technology is moving from laboratory research to large-scale application, laying the foundation for the deployment of the next generation of high-performance multimodal intelligent systems.

3. Technical challenges of pruning methods in natural language processing

3.1. The trade-off dilemma between performance and efficiency

In NLP, although pruning techniques can significantly improve the inference efficiency of models and reduce resource consumption, one of the core challenges lies in the trade-off dilemma between performance and efficiency. The essence of pruning is to achieve lightweighting by removing redundant parameters or structures from the model, but this process inevitably loses some of the

DOI: 10.54254/2755-2721/2025.AST26375

model's capabilities, especially in complex tasks where this loss is more significant. For example, in long text tasks, the sparsity of the attention mechanism causes the model to be unable to effectively capture long-distance dependencies, thereby affecting task performance [9]. Studies have shown that when the pruning rate exceeds a certain threshold, the accuracy or generation quality of the model drops sharply, while the actual acceleration effect may fail to meet the theoretical expectations due to hardware limitations.This contradiction is particularly prominent in generative tasks such as text summarization or machine translation, which rely heavily on structural integrity. In addition, although dynamic pruning techniques can adaptively adjust computational paths based on input, their implementation requires complex scheduling algorithms and hardware support, further increasing the difficulty of actual deployment.Current solutions often adopt hybrid strategies,such as combining knowledge distillation or progressive pruning, to mitigate performance loss, but still cannot completely eliminate this predicament. Future research will need to make breakthroughs in three areas: algorithm design, hardware adaptation, and theoretical analysis, such as developing task-aware pruning criteria, optimizing the hardware efficiency of sparse computing, and establishing a quantitative evaluation framework for pruning to model representation ability, in order to achieve a better balance between performance and efficiency.

3.2. The gap between theory and practice of dynamic pruning

Dynamic pruning techniques can theoretically significantly improve computational efficiency without sacrificing performance by adaptively adjusting the sparse patterns of the model according to the input data [10]. However, it faces a significant gap between theory and practice in practical applications, especially the lack of hardware acceleration support becomes a major bottleneck. Dynamic sparsification requires models to adjust computational paths in real time during inference, such as activating or skipping specific neurons, attention heads, or network layers, but existing general-purpose hardware, such as Gpus and cpus, are mainly optimized for dense matrix computations and have limited support for sparse matrix operations. Although new-generation Gpus such as the NVIDIA Ampere architecture introduce sparse tensor cores that can accelerate computations of specific sparse patterns, the sparse structures generated by dynamic pruning are often irregular and vary with the input, making it difficult to fully utilize these dedicated hardware units. In addition, efficiency losses in memory access, thread scheduling, and cache utilization in sparse computing can result in actual speedup ratios far below theoretical expectations, and even increase latency in some cases due to scheduling overhead. For example, in NLP tasks, dynamic pruning of Transformer models can cause the GPU's parallel computing power to be underutilized due to the irregularity of sparse attention patterns, resulting in limited improvement in actual inference speed. At the same time, the software ecosystem for dynamic sparsity is not yet mature. Mainstream deep learning frameworks such as PyTorch and TensorFlow have insufficient native support for dynamic sparse computing, and developers often have to rely on custom kernel implementations or third-party libraries, further increasing deployment complexity. Breaking through this gap in the future requires collaborative design of hardware and algorithms: on the one hand, hardware needs to support more flexible sparse computing models, such as reconfigurable data flow architectures or neuromorphic chips; On the other hand, algorithms need to consider hardware-friendliness, such as constraining dynamic pruning to generate regularized sparse structures, or combining compilation optimization techniques to achieve efficient mapping of sparse patterns. In addition, establishing unified standards and evaluation benchmarks for dynamic sparse computing and promoting the ecosystem of software frameworks are also key to narrowing this gap.

DOI: 10.54254/2755-2721/2025.AST26375

Only through interdisciplinary collaboration can dynamic pruning technology truly realize its theoretical potential and promote the wide application of efficient AI models.

3.3. Limitations of generalization capabilities across tasks and languages

The ability to generalize across tasks and languages is one of the significant challenges currently faced by pruning techniques [11]. Although pruning methods can achieve good compression results on specific tasks or languages, performance often drops significantly when the pruned models are transferred to new tasks or different languages. This limitation is particularly prominent in low-resource language scenarios, where the lack of sufficient training data to recalibrate the structure of the pruned model makes it difficult for the model to adapt to new language features or task requirements. For example, a pruned model optimized for an English text classification task fails to effectively capture the key semantic information of a new language when migrated to a low-resource language with a very different grammatical structure, such as Swahili or Bengali, because weights sensitive to cross-language common features are removed during the pruning process. Similarly, when a model pruned for a machine translation task is directly applied to a question-answering system during cross-task migration, its performance often plummets due to the over-pruning of task-specific structures. This lack of generalization ability stems from the fact that most of the existing pruning methods rely on task-driven local optimizations, lacking consideration of the model's cross-domain robustness. In addition, pruning of multilingual pre-trained models, such as mBERT or XLM-R, often ignores the parameter sharing feature between languages, leading to the dominance of pruning decisions by high-resource languages and further exacerbating the performance degradation of low-resource languages. Addressing this limitation requires breakthroughs in multiple directions: one is to develop language-independent or task-independent pruning criteria, such as global evaluation based on parameter sensitivity rather than task-specific loss; 2 Introduce meta-learning or adaptive pruning mechanisms that enable models to dynamically adjust sparse patterns based on a small sample size of new tasks or languages; Third, explore a multi-task joint pruning framework that retains key structures that are common to a variety of tasks and languages during compression. In the future, as multilingual and multitask learning progresses further, pruning techniques need to pay more attention to the design of the generalization ability of models in order to truly achieve the goal of "one pruning, multiple applications" and promote the popularization of lightweight models in global applications.

3.4. Lack of explainability and theoretical support

The current pruning technique still has significant deficiencies in terms of explainability and theoretical support, which has become a key bottleneck restricting its further development. The pruning process is often regarded as a "black box" operation, and researchers can usually only observe the performance changes after pruning through experiments, but have difficulty understanding why some parameters or structures are retained while others are removed [12]. This opacity in decision-making poses a significant risk to the application of pruning models in safety-critical areas. For example, in scenarios such as medical text analysis or financial risk assessment, unexplicable pruning decisions lead to the model's neglect of important features, resulting in systemic biases or security risks. More worryingly, a large number of empirical studies have shown that although pruned models can maintain good performance on standard test sets, their robustness is often significantly reduced - the success rate of adversarial sample attacks increases by 20-30%, and the sensitivity to input perturbations also increases significantly. There is currently a lack of a

DOI: 10.54254/2755-2721/2025.AST26375

rigorous theoretical explanation for this degradation of robustness, and traditional generalization error theories are difficult to fully apply to analyzing the behavioral characteristics of structured sparse models. Existing theoretical frameworks such as the "lottery hypothesis" offer some explanations for the effectiveness of pruning, but cannot predict the extent to which a specific pruning strategy will affect the robustness of the model. This stems from the complexity of assessing parameter importance and the unclear link between pruning-induced loss landscape changes and robustness. To break through this limitation, frontier research is moving in three directions: first, developing new interpretive tools based on topological data analysis to predict model behavior by analyzing changes in the geometric characteristics of lost landscapes before and after pruning; Second, build a quantitative assessment framework for pruning robustness, such as introducing an adversarial sparsity metric to measure the stability of the pruned model under different attack intensities; Third, explore theory-guided pruning methods that incorporate generalized error boundary analysis directly into the design of pruning criteria. Only by establishing a more complete interpretability framework and theoretical basis can pruning techniques truly become a reliable and credible model optimization tool and exert their lightweight value in high-risk fields such as healthcare and finance.

4. Future research directions and prospects

4.1. An automated and intelligent pruning framework

The automated and intelligent pruning framework represents the cutting-edge paradigm evolution in the field of deep learning model compression. Its core lies in replacing manual heuristic rules with autonomous algorithmic decision-making to efficiently eliminate redundant parameters of the model, while also taking into account performance balance under multi-objective constraints. Traditional pruning methods rely on local weight amplituities or regularized sparse patterns (such as hardware incompatibility due to unstructured pruning and accuracy damage due to coarse-grained structured pruning), making it difficult to reconcile the complex trade-offs of compression ratio, accuracy loss, and hardware acceleration performance. The new generation of intelligent frameworks breaks through this limitation with three innovations: graph representation learning, multi-objective joint optimization, and hardware-aware decision-making. In terms of graph representation learning, SACP (Structure-Aware Automatic Channel Pruning) transforms neural network topologies into graph structures, uses graph convolutional networks (GCN) to encode inter-layer dependencies (such as residual connections), and learns the global importance of channels Significantly reduce the mispruning rate [13].For example, on ResNet, the method reduced the miscut rate in ImageNet tasks by 23% via capturing cross-layer dependencies. Similarly, the AutoSculpt framework leverages a Graph Attention network (GATv2) and reinforcement learning, achieving high compression rates (55% on ResNet, 45% on Vision Transformers) with <3% precision loss and 29% lower latency.The multi-objective joint optimization mechanism coordinates conflicting objectives through a unified search space. The TinyML edge device joint optimization framework incorporates pruning rate, quantization bit width, and hardware latency into the evolutionary algorithm search to generate Pareto optimal solution 5; AutoCompress, combining hybrid structured pruning (filter + channel) with ADMM (alternating direction multiplier method) dynamic regularization, achieves 54.2x compression with zero precision loss on ResNet-18 and boosts mobile inference speed to 2.7 milliseconds 9. Such methods rely on adaptive reward and punishment functions (such as AutoSculpt's FLOPs- accuracy trade-off function) to avoid the suboptimality of staged optimization. The core of intelligent decision engines is reinforcement

DOI: 10.54254/2755-2721/2025.AST26375

learning and meta-learning. MetaPruning predicts the weight distribution of the pruning structure through a meta-network (PruningNet), and combines evolutionary algorithms to search for the optimal configuration to achieve hardware-independent automatic compression on MobileNet-V2. FLAP (FLuctuation based Adaptive Pruning) proposes a "fluctuation metric" to assess the recoverability of feature maps after weight removal, maintaining over 90% of task performance in LLM pruning without retraining. The current framework has proven value in edge computing and generative tasks: (1) Edge deployment: TinyML's combined pruning - quantization toolchain reduces MobileNet-V3 memory to 500KB and lowers the energy consumption of industrial predictive maintenance models by 60%; (2) Generative task optimization: Attribution-guided Pruning uses interlayer correlation propagation (LRP) to locate task-key subgraphs (such as indirect object recognition circuits), maintaining core semantic integrity in Llama models when compression exceeds 50%. (3) Future research needs to deepen towards tripole collaboration: Developing task-aware pruning criteria at the algorithmic end (such as modular sparsity in SparseLLM); Hardware-optimized in-memory computing architecture for sparse computing; Establish a quantitative correlation between pruning instability and model flatness at the theoretical end (generalized stability theory). Only through cross-level collaborative innovation can we approach the Pareto optimal boundary under resource constraints and empower the universal deployment of lightweight AI.

4.2. Collaborative design of hardware with dynamic sparsity

Dynamic sparsifying hardware co-design is a key paradigm for addressing the efficiency bottleneck of AI computing. The core lies in achieving dynamic perception of sparse patterns and adaptive scheduling of computing resources through deep coupling of algorithms and hardware, thereby breaking through the inherent limitations of traditional static pruning [14]. The core challenge in this field lies in the irregular computing patterns resulting from dynamic sparsity (such as input-dependent pixel-level activation or attention head selection), which are difficult to be efficiently supported by traditional hardware due to fixed data flow architectures and redundant memory access. To address this, in-memory sparse computing architectures significantly reduce index overhead by embedding sparsity information near the storage cell: Wang's team used molybdenum disulfide ferroelectric transistors to integrate weight storage and sparsity encoding transistors within a single cell, reducing sparse training energy consumption by an order of magnitude and achieving software-hardware co-optimization through vector approximation update algorithm (VAU) to maintain 98.4% EMNIST classification accuracy at 75% sparsity. Similarly, sparse-aware readout circuits (such as the SAC scheme) adaptively adjust the quantization range and resolution based on model sparsity, doubling the throughput of in-memory computations in resistive memory (RRAM) and improving energy efficiency by 25.35%. In multimodal scenarios, cross-phase dynamic sparse frameworks become the breakthrough point: Dynamic-LLaVA pruned visual tokens in the pre-fill phase and sparsified text tokens in the decoding phase through an image predictor and an output predictor, and combined with online KV Cache compression, achieved a 75% reduction in pre-fill computational overhead, a 50% reduction in decoding memory usage, and lossless visual understanding capability.

To enhance hardware flexibility, programmable sparse accelerators adapt to dynamic patterns by reconstructing data streams [14]. Intel patent proposes coarse-grained programmable computing units with advanced schedulers to support runtime reconstruction of sparse matrices, significantly enhancing the parallel efficiency of image generation and text tasks; The SALO2 framework from Shanghai Jiao Tong University is designed with a dedicated hardware pipeline that supports static

DOI: 10.54254/2755-2721/2025.AST26375

and dynamic sparse switching of the attention mechanism, which is 13.65 times faster than Gpus in long sequence tasks. In addition, the dual-core heterogeneous architecture focuses on dynamic sparse convolution: the saliency map computing module (lightweight CNN) is decoubrated from the sparse convolution module, dynamically masking redundant pixels through salience-driven binarization masks, reducing invalid computations by more than 50% and hardware resource consumption by 60%. The future direction requires further integration of bio-inspired architectures (such as the dynamic regulation mechanism of astrocytes) with cross-layer optimization theories (such as the CNN-accelerator collaborative search framework CNNBench/AccelBench) to achieve three-level synergy of algorithms, circuits, and applications, ultimately approaching the Pareto frontier of energy efficiency and performance.

4.3. Green pruning techniques for sustainable AI

Green pruning techniques for sustainable AI aim to significantly reduce the energy consumption and carbon footprint of AI models through the synergistic optimization of algorithms and hardware while maintaining their performance efficiency. The core of the technology lies in multi-dimensional collaborative innovation: at the algorithmic level, model pruning achieves lightweighting by removing redundant parameters, combined with quantization techniques to reduce computational precision requirements, which can significantly reduce inference energy consumption by sacrificing only about 15% accuracy while reducing the number of parameters by 90%. Meta's CATransformers framework is a typical example, which uses a multi-objective Bayesian optimization engine to balance latency, accuracy and carbon emissions. For instance, the CarbonCLIP-S model reduces carbon emissions by 17% while maintaining accuracy, and edge inference latency is controlled within 15 milliseconds [15]. The dynamic pruning technique further ADAPTS the computational path based on the input, but addresses the additional energy consumption brought about by the complexity of hardware scheduling. At the hardware level, green pruning relies on the underlying support of energy efficiency improvements. Silicon photonic integrated circuit (PIC) technology uses optical neural networks instead of traditional electronic computing, which is 140 times more energy-efficient than digital hardware such as Gpus. The III-V group semiconductor heterogeneous integration solution also enhances the scalability of photonic devices, providing an efficient execution environment for high sparsity models. Innovations in data center infrastructure, such as Delta's 800V high-voltage DC power architecture, boost grid-to-chip energy efficiency to 92% and address thermal management for high-density computing through liquid cooling technologies, such as cabinet-level CDU. System-level optimization emphasizes carbon management throughout the entire life cycle. CATransformers' practice shows that optimizing delays alone can lead to a 2.4x increase in "embodied carbon" in the hardware manufacturing process, while an integrated carbon emissions design strategy can achieve a 20% reduction in total emissions. In the future, task-aware pruning criteria (such as dual attention importance scores) and theoretical evaluation frameworks (such as manifold expression ability analysis) will need to be integrated with sparse computing hardware (such as NVIDIA Ampere tensor kernels) and renewable energy microgrids to build a closed-loop green algorithm-hardware-energy system.

In summary, green pruning technology drives the transformation of AI from a high-energy-consuming paradigm to a sustainable one through an innovative path of "software-hardware synergy and carbon-efficiency balance", providing key technical support for global emission reduction targets.

DOI: 10.54254/2755-2721/2025.AST26375

4.4. Lightweight adaptation of multimodal and hyperscale models

A key AI trend is the deep integration of multimodal information (text, image, audio, video) to build more capable general agents. Multimodal hyperscale models represented by GPT-4V, Gemini, Claude 3, etc. have made significant breakthroughs in this direction. However, these models often have hundreds of billions or even trillions of parameters (such as GPT-4, which is estimated to have about 1.8 trillion parameters) and the resulting astonishing computational overhead (training costs often reach tens of millions of dollars, and the energy consumption for a single inference can be hundreds of times that of traditional tasks) It poses a huge obstacle to its widespread deployment in resource-constrained edge devices such as mobile phones and Internet of Things terminals, latency-sensitive real-time applications such as autonomous driving and industrial quality inspection, and research and development scenarios that require frequent iterations. As a result, efficient lightweight adaptation of multimodal hyperscale models has become a key technical challenge to drive their actual application. The core objective of lightweight adaptation is to significantly reduce the computational complexity, storage requirements, and energy consumption of the model through a series of model compression and optimization techniques while maintaining its excellent performance as much as possible. The mainstream technical approaches include structured and unstructured pruning, systematically removing redundant or slightly contributing parts of the model weights; Quantization techniques that convert model weights and activation values from 32-bit floating-point numbers (FP32) to 8-bit integers (INT8) or even 4-bit (INT4) significantly reduce storage usage and memory access bandwidth pressure; Knowledge distillation, using a large model (teacher model) to guide the training of a more compact small model (student model) that learns and approximates the behavior and output distribution of the teacher model; And design efficient model architectures, such as sparsely activated hybrid expert models (MoE), to activate only a subset of parameters for each inference. These lightweight techniques are not isolated and are often combined to form a better solution [16]. Successful lightweight adaptation not only extends model deployment to mobile and embedded devices, accelerating response times (significantly reducing inference latency), but also greatly reduces cloud service costs (reducing computing resource consumption) and carbon emissions (reducing energy consumption), It is decisive for achieving large-scale and civilian application of multimodal large models in key areas such as medical image analysis, real-time educational assistance, inclusive finance, and intelligent security, and serves as a key bridge connecting cutting-edge AI research with actual industrial value.

5. Conclusion

This study shows that lightweight adaptation of multimodal and hyperscale models is a key bridge to moving cutting-edge AI technologies from the laboratory to widespread industrial application. Although large multimodal models such as GPT-4V and Gemini demonstrate powerful perception and understanding capabilities, the huge computational overhead (tens of millions of dollars in training costs and hundreds of times in single-inference energy consumption) caused by their hundreds of billions or even trillions of parameters (such as GPT-4 about 1.8 trillion parameters) Deployment in edge devices, real-time scenarios, and pervasive applications has been severely constrained. To break through this bottleneck, this study systematically demonstrates a synergistic optimization path that integrates structured/unstructured pruning, low-bit quantization, knowledge distillation, and efficient architecture design (such as sparse activation of MoE). These techniques significantly reduce computational complexity, storage requirements, and energy consumption while maintaining the core performance of the model by removing redundant parameters, compressing

DOI: 10.54254/2755-2721/2025.AST26375

data accuracy, migrating representational capabilities, and dynamically scheduling computing resources. Particularly notable is the adaptation innovations for multimodal characteristics - such as GigaPrune's hierarchical progressive sparsification and ModaBalance's dynamic weighting of cross-modal pruning intensity - effectively addressing the feasibility challenges of parameter importance imbalance between modalities and ultra-large-scale sparsification. The successful lightweight adaptation not only extends large models to devices such as mobile phones and iot terminals (such as Huawei PanGu-π achieving edge-side image generation within 300ms), reducing cloud service costs and carbon emissions, but also creates scalable application value in key areas such as medical imaging, industrial quality inspection, and real-time education. Further research on dynamic sparsity hardware collaboration (such as in-memory computing architecture), green pruning and multimodal evaluation standards is needed in the future to build a more efficient, robust and sustainable lightweight technology ecosystem and ultimately achieve the universal deployment of multimodal intelligence.

References

[1] Min E, Chen R, Bian Y, et al. Transformer for graphs: An overview from architecture perspective [J]. arXiv preprint arXiv: 2202.08455, 2022.
[2] Liu Z, Sun M, Zhou T, et al. Rethinking the value of network pruning [J]. arxiv preprint arxiv: 1810.05270, 2018.
[3] Zhang Liang, Zhang Zeng, Shu Weihua, et al. Structured pruning of convolutional layers based on YOLOv3 [J]. Journal of Computer Engineering & Applications, 2021, 57(6).
[4] Xie Y, Zhang Z, Zhou D, et al. MoE-Pruner: Pruning Mixture of Experts Large Language Model using the Hints from Its Router [J]. arxiv preprint arxiv: 2410.12013 2024.
[5] Wei, Song Hantao. Heuristic discovery of association rules Based on similarity [J]. Journal of Beijing Institute of Technology (Natural Science Edition), 2002 (1): 98-100.
[6] Liu Yu, Lei Xuemei. Structured Pruning method incorporating Mobile NetV3 features [J]. Journal of Shanghai Jiao Tong University (1006-2467), 2023, 57(9).
[7] Xu X, Li M, Tao C, et al. A survey on knowledge distillation of large language models [J]. arxiv preprint arxiv: 2402.13116, 2024.
[8] Hai T, Wang W, Yan F, et al. A Deep Learning-Based Approach to Apple Tree Pruning and Evaluation with Multi-Modal Data for Enhanced Accuracy in Agricultural Practices [J]. Agronomy, 2025, 15(5): 1242.
[9] Zuo Meng, Liu Yiyang, Cui Hao, et al. Point cloud Semantic segmentation method based on sparse convolution and attention mechanism [J]. Advances in Laser & Optoelectronics, 2023, 60(20): 2015002.
[10] Lin T, Stich S U, Barba L, et al. Dynamic model pruning with feedback [J]. arxiv preprint arxiv: 2006.07253, 2020.
[11] Liu. Generalization Ability of Deep Learning in natural Language Processing [J]. Dissertation of the Institute of Electrical Engineering, National Taiwan University, 2021: 1-61.
[12] Liebenwein L, Baykal C, Carter B, et al. Lost in pruning: The effects of pruning neural networks beyond test accuracy [J]. Proceedings of Machine Learning and Systems, 2021, 3: 93-138.
[13] Xu Bingbing, Cen Keting, Huang Junjie, et al. Graph Convolutional neural network Review [J]. Journal of Computer, 2020, 43(5): 755-780.
[14] Qian Jiaming, Lou Wenqi, Gong Lei, et al. Co-optimization of algorithm compression-hardware design for 3D-CNN [J]. Journal of Computer Engineering & Applications, 2023, 59(18).
[15] Wang I, Ardalani N, Elhoushi M, et al. Carbon Aware Transformers Through Joint Model-Hardware Optimization [J]. arxiv preprint arxiv: 2505.01386, 2025.
[16] Zhou Chengyang, Liu Wei, Wu Tianrun, et al. Rock slice image classification based on Hybrid expert model [J]. Journal of Jilin University (Science Edition), 2024, 62(4): 905-914.