1. Parameter-Efficient Fine-Tuning (PEFT):
Definition: A technique that updates only a small subset of parameters in large models during fine-tuning, improving efficiency and reducing computational costs.
Key points:
• Allows the use of large pre-trained models without the need for extensive computational resources.
• Maintains model performance while minimizing the number of parameters that need to be trained.
2. Transfer Learning:
Definition: A machine learning method where a model trained on one task is reused on a second, related task, leveraging learned features to improve performance.
Key points:
• Reduces training time and data requirements by utilizing existing knowledge from pre-trained models.
• Commonly used in natural language processing (NLP) with models like BERT and GPT.
3. Fine-Tuning:
Definition: The process of taking a pre-trained model and training it further on a specific task, adjusting its weights to optimize performance for that task.
Key points:
• Involves updating all model parameters, which can be computationally expensive with large models.
• Often leads to improved accuracy on specialized tasks compared to using the model as-is.
4. Sparse Fine-Tuning:
Definition: A method of fine-tuning where only a small number of parameters are updated, often focusing on the most critical weights in the model.
Key points:
• Reduces the computational burden by limiting the number of parameters that need adjustment.
• Can maintain performance close to full fine-tuning while using significantly fewer resources.
5. Lottery Ticket Hypothesis:
Definition: A theory suggesting that within a randomly initialized neural network, there exist smaller subnetworks (winning tickets) that can be trained to achieve high performance.
Key points:
• Supports the idea that large models can be pruned effectively to find efficient subnetworks.
• Winning tickets can be identified through pruning techniques, leading to reduced model sizes.
6. Pruning:
Definition: The process of removing weights or neurons from a neural network to create a smaller, more efficient model while maintaining performance.
Key points:
• Can significantly reduce model size and computational requirements without sacrificing accuracy.
• Common methods include magnitude pruning, where the smallest weights are removed based on their magnitude.
7. Adapter Functions:
Definition: Small modules inserted into pre-trained models that allow adaptation to new tasks without modifying the original model weights extensively.
Key points:
• Adapters enable task-specific adjustments while keeping the base model intact, enhancing modularity.
• They facilitate quick adaptation to new tasks with minimal additional parameters.
8. Low-Rank Adaptation (LoRA):
Definition: A technique that introduces low-rank matrices into transformer layers to approximate weight updates, allowing for efficient fine-tuning.
Key points:
• Reduces the number of parameters that need to be trained while maintaining high performance.
• LoRA is particularly effective in large models like GPT-3, achieving significant parameter reduction.
9. Prompting:
Definition: A technique in NLP where specific input prompts are used to guide model responses, often replacing traditional fine-tuning.
Key points:
• Enables in-context learning, allowing models to perform tasks without extensive retraining.
• Performance can vary significantly based on prompt wording and structure.
10. Catastrophic Forgetting:
Definition: A phenomenon where a neural network forgets previously learned information upon learning new information, particularly in sequential learning tasks.
Key points:
• Can be mitigated by using modular representations that allow for better retention of past knowledge.
• Important consideration when training models on multiple tasks over time.
11. Neural Network Architecture:
Definition: The structure of a neural network, defined by the arrangement of layers, nodes, and connections that determine how data is processed.
Key points:
• Different architectures (e.g., convolutional, recurrent) are suited for different types of data and tasks.
• The choice of architecture significantly impacts model performance and training efficiency.
12. In-Context Learning:
Definition: A method where models learn to perform tasks based on examples provided in the input context without explicit retraining.
Key points:
• Useful for adapting to new tasks quickly with minimal additional training.
• Performance can be highly sensitive to the quality and structure of the provided examples.
13. Modular Representations:
Definition: A design approach in neural networks where different components (modules) can be independently trained and combined for various tasks.
Key points:
• Promotes flexibility and reusability of model components across different tasks.
• Can help alleviate issues like catastrophic forgetting by segregating task-specific knowledge.
14. Weight Magnitude:
Definition: A criterion used in pruning that evaluates the importance of weights based on their absolute values, with smaller weights often being removed first.
Key points:
• Effective in identifying which weights contribute less to model performance, facilitating efficient pruning.
• Weight magnitude pruning is a straightforward method that can lead to significant model size reduction.
15. Early-Stopping Criterion:
Definition: A technique used during training to stop the process when the model's performance on a validation set begins to degrade, preventing overfitting.
Key points:
• Helps in selecting the optimal model parameters by avoiding unnecessary training epochs.
• Can improve the generalization of the model to unseen data by reducing overfitting.
1. Code Generation:
Definition: The process of automatically generating source code from high-level specifications, often using machine learning models trained on large codebases.
Key points:
• Utilizes models like Codex and GitHub Copilot to assist developers in writing code efficiently.
• Can improve productivity by automating repetitive coding tasks and suggesting code snippets.
2. Question Answering (QA):
Definition: A natural language processing task where systems are designed to automatically answer questions posed in natural language using relevant information.
Key points:
• Applications include chatbots, virtual assistants, and customer support systems to enhance user interaction.
• Evaluates machine understanding of text, serving as a benchmark for NLP capabilities.
3. Large Language Models (LLMs):
Definition: Advanced neural network models trained on vast amounts of text data to understand and generate human-like text.
Key points:
• Examples include GPT-3 and BERT, which excel in various NLP tasks like translation and summarization.
• Can be fine-tuned for specific applications, improving performance on targeted tasks.
4. Retrieval-Augmented Generation (RAG):
Definition: A model that combines retrieval of relevant documents with generative capabilities to produce answers or summaries from the retrieved information.
Key points:
• Enhances performance on open-domain question answering by leveraging external knowledge sources.
• Allows models to generate contextually relevant responses, improving accuracy and relevance.
5. Attention Mechanism:
Definition: A technique in neural networks that allows models to focus on specific parts of the input data when making predictions.
Key points:
• Crucial in transformer architectures, enabling better handling of long-range dependencies in text.
• Improves model interpretability by highlighting which parts of the input influence the output.
6. Bidirectional Encoder Representations from Transformers (BERT):
Definition: A transformer-based model that processes text bidirectionally, capturing context from both left and right of a word for better understanding.
Key points:
• Achieves state-of-the-art results in various NLP tasks, including reading comprehension and sentiment analysis.
• Pre-trained on large text corpora, allowing fine-tuning for specific applications with minimal data.
7. Sequence-to-Sequence Models:
Definition: Neural network architectures designed to transform input sequences into output sequences, commonly used in translation and summarization.
Key points:
• Utilize encoder-decoder structures where the encoder processes the input and the decoder generates the output.
• Can be enhanced with attention mechanisms to improve performance on longer sequences.
8. Natural Language Processing (NLP):
Definition: A field of artificial intelligence focused on the interaction between computers and human language, enabling machines to understand, interpret, and generate text.
Key points:
• Applications include sentiment analysis, chatbots, and machine translation, impacting various industries.
• Involves techniques like tokenization, parsing, and named entity recognition to process and analyze text.
9. Evaluation Metrics in QA:
Definition: Quantitative measures used to assess the performance of question answering systems, such as F1 score and exact match.
Key points:
• F1 score evaluates the balance between precision and recall, providing insights into model accuracy.
• Exact match measures the percentage of answers that exactly match the ground truth, indicating reliability.
10. Pre-trained Language Models:
Definition: Models trained on extensive datasets to understand language patterns before being fine-tuned for specific tasks.
Key points:
• Reduce the amount of labeled data needed for training new models, facilitating quicker deployment.
• Examples include GPT, BERT, and RoBERTa, which have transformed NLP applications.
11. Code Documentation Generation:
Definition: The automatic creation of documentation for codebases, explaining functionality and usage to aid developers.
Key points:
• Improves code maintainability and understanding, especially in large projects with multiple contributors.
• Can be integrated into development environments to provide real-time documentation suggestions.
12. Bug Detection and Fixing:
Definition: The process of identifying and correcting errors in software code, often enhanced by machine learning models.
Key points:
• Tools like static analyzers and LLMs can detect potential bugs before code execution, improving software quality.
• Automated fixing suggestions can expedite the debugging process, saving developers time.
13. HumanEval Dataset:
Definition: A benchmark dataset designed to evaluate code generation models by providing programming tasks with hidden tests.
Key points:
• Helps assess the functional correctness of generated code by requiring it to pass specific unit tests.
• Serves as a standard for comparing the performance of different code generation systems.
14. Semantic Similarity:
Definition: A measure of how alike two pieces of text are in meaning, regardless of the exact wording used.
Key points:
• Essential for tasks like paraphrase detection, information retrieval, and question answering.
• Techniques include embedding-based methods, where sentences are represented in vector space for comparison.
15. Model Fine-tuning:
Definition: The process of taking a pre-trained model and training it further on a specific dataset to improve performance on a particular task.
Key points:
• Allows leveraging existing knowledge while adapting the model to new, often smaller datasets.
• Crucial for achieving high accuracy in specialized applications like medical text analysis or legal document review.