Newsletter Data & AI

28/10/2024

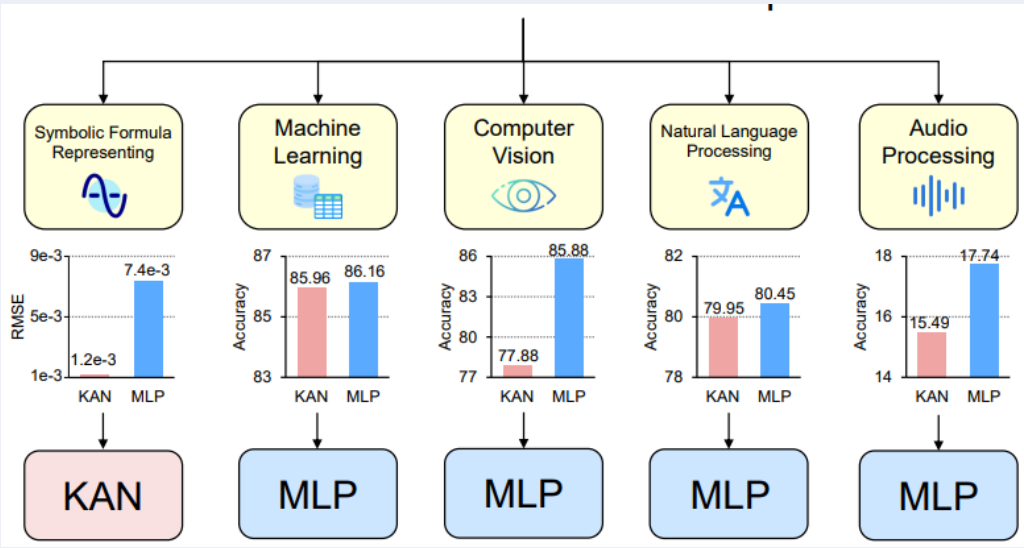
THE SHORT OF IT ⏱️

- **LLM Advancements :** A new model called the AI Scientist automates the entire research process, from generating ideas to conducting experiments and writing papers, aiming to make scientific discovery more accessible and cost-effective.
- **V-Information and Deep Learning Insights:** Research on V-information reveals that deep learning models learn simpler features first, while more complex ones emerge later but have a smaller impact on the model's decision-making process.

Trends

- [Paper] KAN or MLP: A Fairer Comparison

  A recent study from NSU reveals that MLP outperforms KAN in most tasks, except symbolic formula representation, where KAN's B-spline activation provides an edge. Applying B-spline to MLP enhances its performance in this domain, while MLP shows overall superior performance.



- [Paper] Speculative RAG: Enhancing Retrieval Augmented Generation through Drafting

  Researchers from Google DeepMind and UC San Diego introduce SPECULATIVE RAG, a framework that improves retrieval-augmented generation by using a smaller specialist model to generate diverse drafts, which are then verified by a larger generalist model. This method enhances performance and reduces latency, achieving up to a 13% accuracy improvement and a 50% reduction in latency across multiple benchmarks, including PubHealth and TriviaQA.

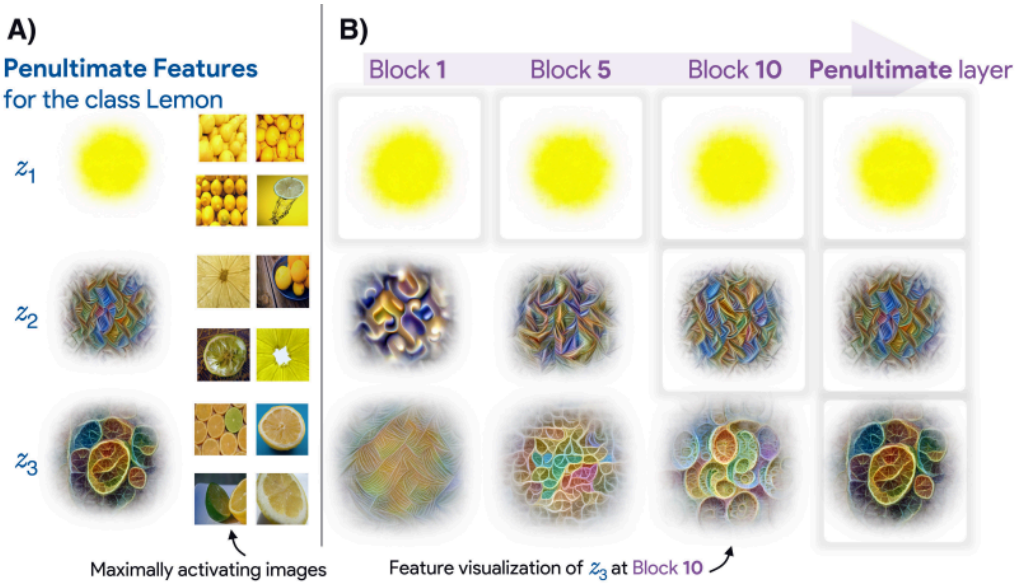
State Of The Art

- [Paper] The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery

  The paper introduces *The AI Scientist*, a framework for fully automating scientific discovery, allowing large language models to independently generate ideas, conduct experiments, and write complete research papers. This approach, applied across machine learning subfields, produces research papers at under $15 each, helping democratize research.

- [Paper] Understanding Visual Feature Reliance through the Lens of Complexity

  A study by Brown University and DeepMind introduces *V-information*, a metric to quantify feature complexity in deep learning models. Analyzing 10,000 features from an ImageNet model, the research finds that simpler features dominate early training, with more complex ones emerging later. Surprisingly, complex features are less critical to the model's decisions, while important features emerge earlier, resembling a "sedimentation process" in the learning hierarchy.



- [Paper] Extracting Prompts by Inverting LLM Outputs

  This work addresses the challenge of language model inversion, focusing on recovering the original prompt from a model's output. The authors introduce *output2prompt*, a black-box method that relies only on standard user queries, eliminating the need for model logits or adversarial techniques. With a novel sparse encoding for improved memory efficiency, *output2prompt* shows zero-shot transferability across multiple language models.

Miscellaneous

- [Blog] How to Prune and Distill Llama-3.1 8B to an NVIDIA Llama-3.1-Minitron 4B Model

  NVIDIA's blog discusses a new approach to improving training efficiency for large language models (LLMs) through *speculative decoding*. This technique uses a smaller model to generate multiple predictions in parallel, which are then verified by a larger model, accelerating inference while maintaining high accuracy. The method shows a

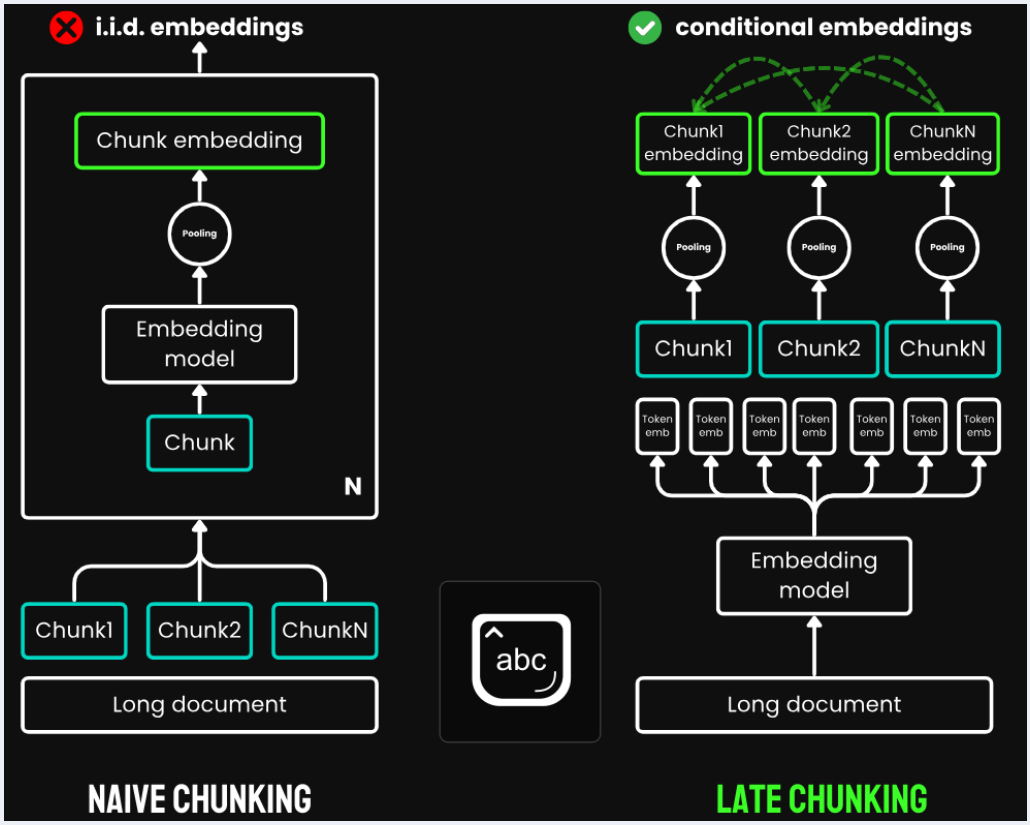
significant reduction in computation time, making LLMs more practical for real-world applications. For more details, refer to the full paper here.

- [Blog] Revolutionizing AI Embeddings with Geometry

  The blog discusses *Angular Embeddings (AE)*, which leverage complex geometry to improve embeddings by focusing on angular information rather than magnitudes, overcoming limitations of cosine similarity in text embeddings. It also introduces *RotatE*, a method for modeling relationships in knowledge graphs as rotations in complex space, offering more robust relation modeling and better semantic distinctions than traditional approaches.

- [Blog] Late Chunking in Long-Context Embedding Models

  The article introduces *Late Chunking*, a method that preserves context in long documents for tasks like Retrieval-Augmented Generation (RAG). By first applying a transformer to the entire text before chunking, it ensures richer contextual embeddings, improving retrieval performance, especially in longer documents.



- [Conference] Scikit-Learn can do THAT?!

  In this PyData 2024 presentation, various lesser-known but powerful features of scikit-learn are explored, going beyond its popular .fit() and .predict() functionalities. The talk covers advanced capabilities such as handling sparse datasets and models, working with larger-than-memory datasets, using sample weight techniques, and performing image classification via embeddings. It also examines data deduplication, tabular embeddings, and pipeline caching, highlighting how these features can enhance machine learning workflows effectively.

- [Package] Darts

  Darts is a versatile Python library for time series forecasting and anomaly detection, supporting models from ARIMA to deep learning techniques like N-BEATS and Transformers. It handles both univariate and multivariate series, offers probabilistic forecasting, and simplifies backtesting, anomaly detection, and model explainability. Built

on PyTorch Lightning, Darts efficiently scales to large datasets and integrates external covariates for enhanced predictions.

Latest Releases

- [Minor Release] Tensorflow 2.18.0

  TensorFlow 2.18.0 now supports NumPy 2.0 by default, with changes in type promotion rules that may affect precision. TensorFlow Lite introduces an optional fourth parameter in `TfLiteOperatorCreate` for a cleaner API and adds support for `SignatureRunner` on models without signatures. Additionally, TensorRT support is disabled in CUDA builds, while Hermetic CUDA ensures more reproducible builds by using specific, downloadable CUDA versions in Bazel targets.

- [Minor Release] Pytorch 2.5.0

  PyTorch 2.5.0 introduces significant performance boosts with a new CuDNN backend for SDPA, offering up to 75% speed-ups on NVIDIA H100 GPUs. TorchInductor sees major improvements, including FP16 support and enhanced CPU performance. The release also brings regional compilation to reduce startup times for repetitive models and introduces FlexAttention for flexible attention mechanisms. Expanded hardware support includes Intel GPUs and TorchInductor compatibility on Windows.

Thank you for your engagement. We eagerly anticipate sharing further advancements in AI with you.