Recent Progress in Understanding Transformer and Developing Its Surpasser: A Survey

Presenter: Maohua Liu

The 2024 IEEE Signal Processing in Medicine and Biology Symposium
December 7th, 2024

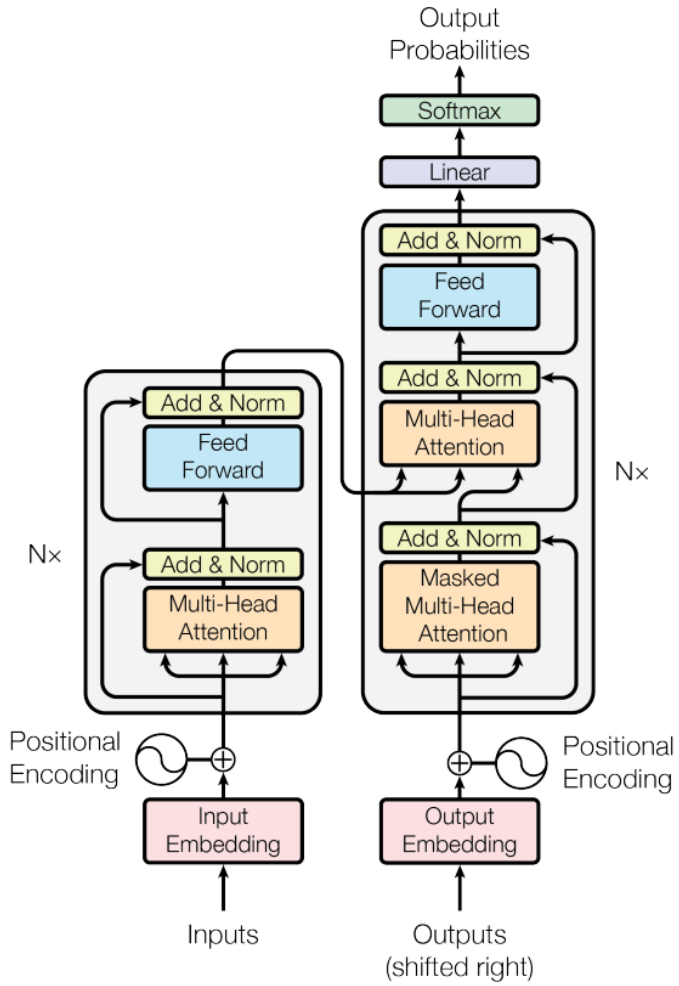
UNIVERSITY OF GEORGIA

Outline

- Introduction of Transformer
- Transformer's comparison with CNN and RNN
- Efforts in understanding Transformer
- Efforts in developing Transformer's surpasser

Introduction of Transformer

- Introduced by Vaswani et al. (2017).
- Revolutionized NLP
- New architectural concept
- Attention mechanism
- Long-term dependency
- Applications



Introduction of Transformer

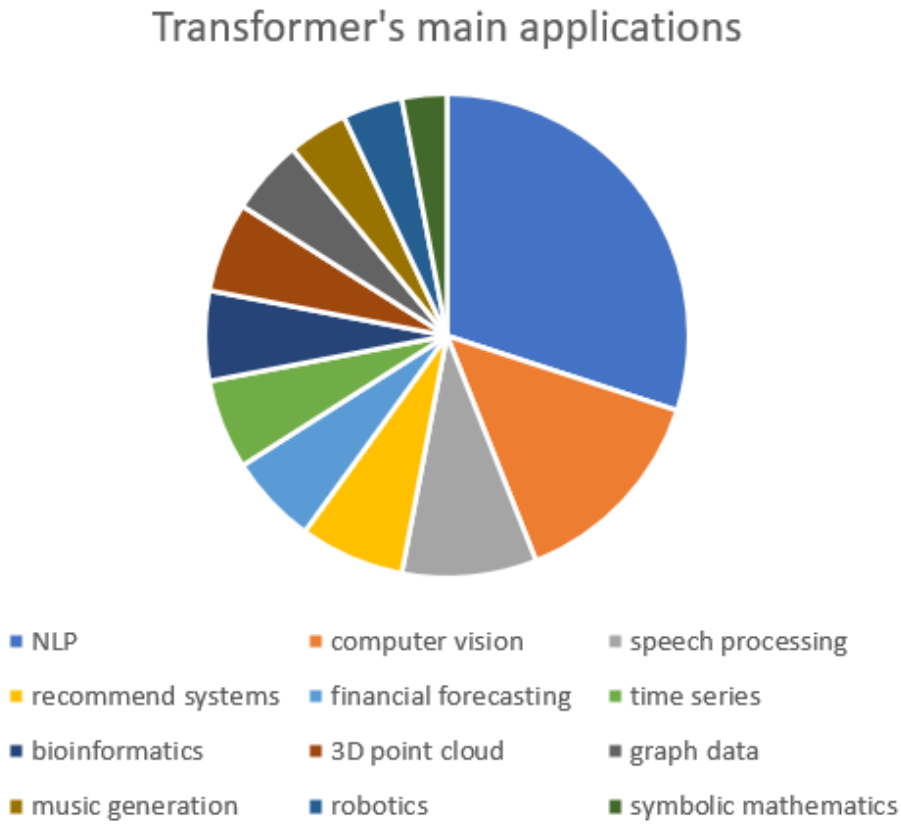
- Introduced by Vaswani et al. (2017).
- Revolutionized NLP
- New architectural concept
- Attention mechanism
- Long-term dependency
- Applications

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

Q (queries), K (keys), V (values) are matrices packed together, and d_k is the dimension of the keys.

Introduction of Transformer

- Introduced by Vaswani et al. (2017).
- Revolutionized NLP
- New architectural concept
- Attention mechanism
- Long-term dependency
- Applications



Comparison

Table 1: Comparison of Transformer, RNN and CNN.

Architecture	Transformer [1][2][4]	RNN [21][22][30][31]	CNN [23][26][27][32]
Basic structure	Self-attention layers, feed-forward neural networks, positional encoding	Sequential processing with recurrent connections for temporal data	Convolutional layers for spatial data processing
Parallelization	Highly parallelizable, processes entire sequences parallelly	Sequential processing, making it less parallelizable	Highly parallelizable, processes multiple parts of the data simultaneously
Long-term dependencies	Efficiently captures long-term dependencies through self-attention	Struggles with long-term dependencies due to vanishing gradient problem	Effective in capturing local features but less suited for long-term dependencies
Training efficiency	Efficient, especially on large datasets, thanks to parallel processing	Slower training due to sequential nature	Faster training due to parallel processing and efficient use of local patterns
Data type	Primarily sequential data (text, time series, speech), but adaptable for images and multimodal data	Sequential data (text, speech, time series)	Primarily spatial data (images, videos), but adaptable to text and sequential data
Advantages	Can handle long-term dependencies efficiently, Highly parallelizable, Scales well with large datasets	Maintains sequence order, Good for temporal data, Suitable for tasks like language modeling	Excellent for spatial feature, Parallelizable and efficient, Effective in recognizing patterns in images
Disadvantages	Resources consuming, Can be data-intensive, May need large datasets for effective training	Struggles with long-term dependencies, Slower training, Vanishing/exploding gradients	Limited in handling long-term dependencies, Not as efficient for sequential data

Understanding Transformer: primary stage

- **Techniques:**
  - Activation-based: Analyzing neuron activities.
  - Gradient-based: Layer-wise Relevance Propagation.
  - Pruning: Removing non-essential components.
  - Perturbation: LIME and feature perturbations.
- **Limitation:** Post-hoc interpretability risks misleading conclusions.

Understanding Transformer: intermediate stage

- **Focus:** Leveraging attention for insights.
- **Methods:**
  - Attention visualization.
  - Attribution methods.
  - Self-explaining models.
  - Probing tasks.
- **Criticism:** Attention weights may not always indicate importance.

Understanding Transformer: advanced stage

- **CRATE:**
  - Sparse Rate Reduction (SRR) for compact representations.
  - Iterative optimization improves data compression.

- **Advantages:**
  - Mathematical interpretability and competitive performance.
  - Scalability for diverse applications.

Efforts in surpassers

- Mamba
- Diffusion Model
- Diffusion Transformer

Table 2: Comparison between Transformer and its surpassers

Architecture	Transformer [42][43][44][45]	Mamba [46][47][48][53]	Diffusion Model [54][55][57]	Diffusion Transformer [58]
Basic Structure	Self-attention layers, feed-forward neural networks, positional encoding	Combines self-attention mechanisms with hierarchical design, focusing on efficiency	Sequential process of adding and removing noise to generate data	Integrates diffusion processes with self-attention mechanisms for sequential data
Mechanism	Self-attention to capture dependencies across entire sequences	Efficient attention mechanisms, designed to optimize speed and resource usage	Stepwise diffusion, starting from noise to create or denoise samples	Combines diffusion (probabilistic modeling) with attention for efficient data generation
Parallelization	Highly parallelizable, processes entire sequences at once	Designed for better parallel processing and scalability	Not inherently parallel; sequential processing for diffusion steps	Combines parallelizable elements of Transformers with sequential steps of diffusion
Applications	NLP, Computer Vision, Time Series, Multimodal tasks	Optimized for fast processing of large-scale data	Image and data generation, noise removal, generative tasks	Advanced generative tasks, combining benefits of Transformers and Diffusion Models
Long-term Dependencies	Efficiently captures long-term dependencies via self-attention	Improved handling of long-term dependencies	Effective for structured data but not designed for dependencies	Captures dependencies while maintaining generative quality through diffusion
Advantages	Can handle long-term dependencies efficiently, Highly parallelizable, Scales well with large datasets,	Optimized for speed and resource efficiency, Scalable for large tasks, Reduced computational cost	Effective for generating high-quality data, Excellent at capturing fine-grained details	Combines strengths of both Transformer and Diffusion models, Maintains generative power with efficient sequence processing
Disadvantages	Requires significant computational resources, Can be data-intensive, May need large datasets for effective training	Newer model with less extensive testing across diverse tasks, May need fine-tuning for specific use cases	Sequential nature slows down training and sampling, Resource-intensive for high-quality output	Combines complexity of both Transformers and Diffusion, making it more resource-intensive
Excels In	NLP (e.g., GPT, BERT), Computer Vision (e.g., Vision Transformer), Time Series (e.g., Informer)	Optimized for large-scale data processing, Applications requiring fast attention computation, Potential use in NLP and large data tasks	Image Generation (e.g., DALL·E 2) Speech Synthesis, Data Denoising, Generative Art	Complex generative tasks, Data generation across multiple domains (e.g., multimodal tasks), Advanced language modeling combined with generation

Summary

- Understanding Transformer
- Developing its surpasser

Thank you all ☺