Transformer Networks: How They Work and Why They Matter

Rakshit Agrawal

Co-Founder & CEO

Ryddle AI

Introduction to AI Evolution

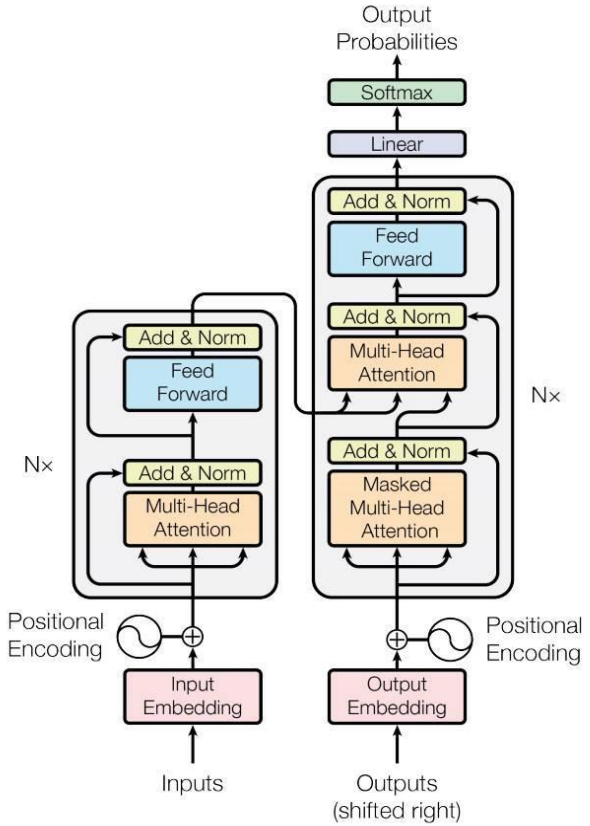
- In Natural Language Processing (NLP), in the early 2010s, word embedding models like Word2Vec and GloVe started capturing semantic meanings.
- In the mid 2010s, RNNs and LSTMs incorporated this into sequence-to-sequence models making it possible to generate continuous text sequences with deep learning.
- Introduction of attention mechanisms gave a significant boost to the performance of these models.
- In 2017, transformers proposed an architecture entirely based on attention, removing the limitations of recurrent neural networks
- Transformer based models such as BERT, GPT, ViT, etc., are shaping the new era of AI.

Importance of Transformers in Modern AI Research and Applications

- **Integrates Multiple Domains**: Unifies text, image, and sound data for cohesive learning.
- **Flexible Architecture**: Adapts to various data types without altering core structure.
- **Contextual Comprehension**: Processes diverse data types simultaneously for deeper insights.
- **Powers AI Applications**: Powers a new category of intelligent AI applications with prompting and transformer based systems.
- **Future Directions**: Poised to further blend and enhance inter-domain machine learning.

Understanding Transformers

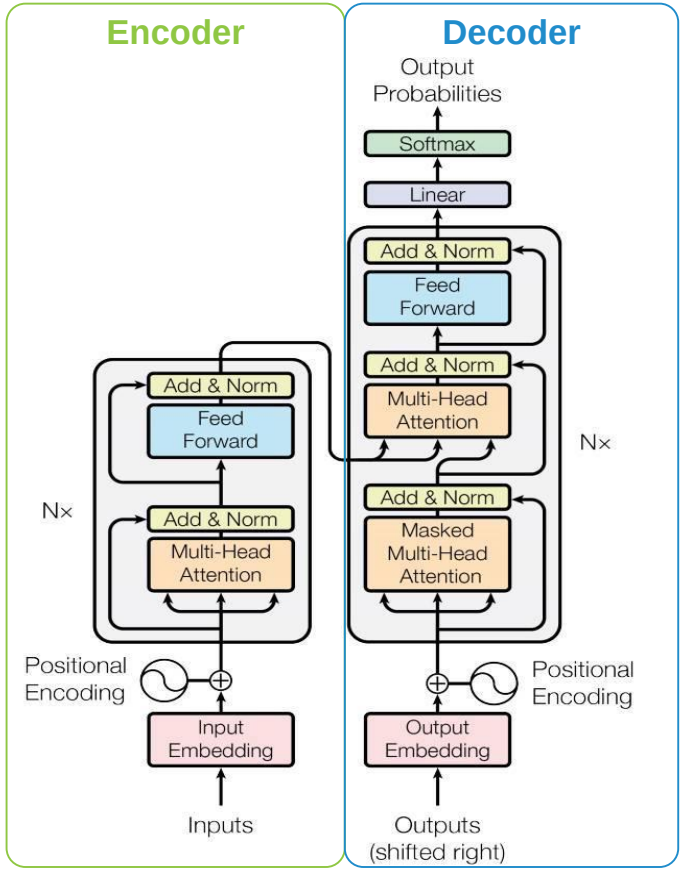
- Neural networks based entirely on attention, replacing recurrent layers.
- Processes entire sequences in parallel, boosting speed and efficiency.
- Highly scalable with increased computational power and data size.
- Versatile across multiple domains, including text, vision, and speech.



The Core of Transformers

Transformer Architecture

- Consists of encoder and decoder blocks
- Main components of a block:
  - Self-attention
  - Layer normalization
  - Feed-forward neural network
- Uses positional encodings

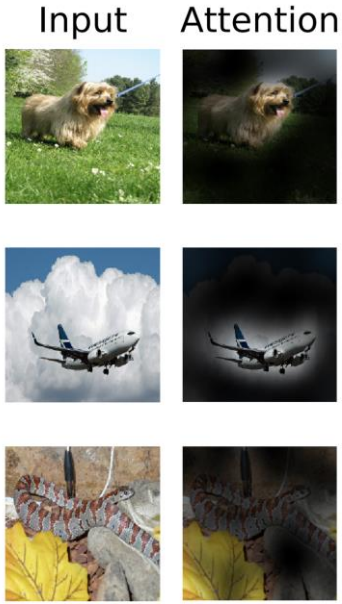


Encoders-Decoders

	Encoder	Decoder
Function	Input → Context	Context → Output
Process	Generates a representation encoding the entire input sequence.	Combines encoder output and previous decoder outputs to predict next symbol in sequence.
Components	• Self-attention • Layer normalization • Feed-forward neural network	• Self-attention • Layer normalization • Feed-forward neural network • Additional layer for outputs of encoder

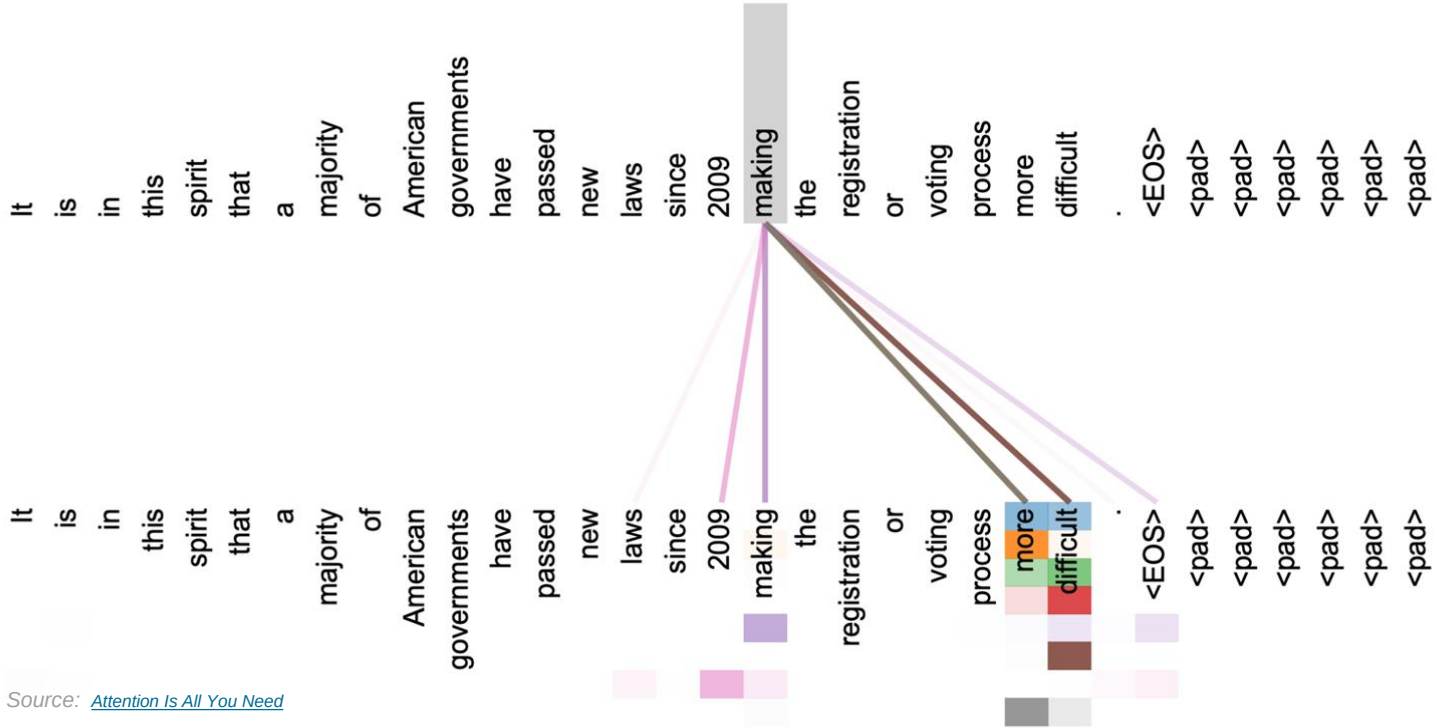
Self-Attention Mechanism

- A mechanism that allows each position in the decoder to attend to all positions in the encoder of the previous layer.
- Self-attention computes a weighted sum of all input representations with respect to their relevance.
- Benefits of self-attention
  - Allows the model to dynamically focus on different parts of the sequence.
  - Provides a more nuanced understanding and representation of the sequence.
  - Facilitates parallel processing, unlike RNNs which process data sequentially.



Source: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Self-Attention Mechanism: Visualizing Attention



Source: Attention Is All You Need

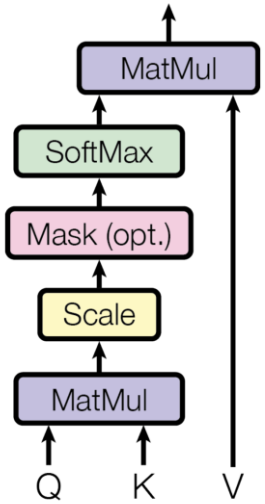
Components of Self-Attention in Transformers

Component	Description	Function
Query, Key, Value Vectors	Each input token is transformed into Q, K, V vectors.	Enables calculation of attention scores and retrieval of information.
Attention Score Calculation	Scores calculated via dot product of Q and K vectors.	Determines focus on different parts of the sequence.
Softmax Layer	Applies softmax to attention scores.	Normalizes scores into a probability distribution.
Weighted Sum	Weighted sum of V vectors using softmax probabilities.	Aggregates information from the sequence based on attention.
Output	Resultant vector from weighted sum.	Serves as input for the next layer, represents aggregated information.

Attention in Transformers

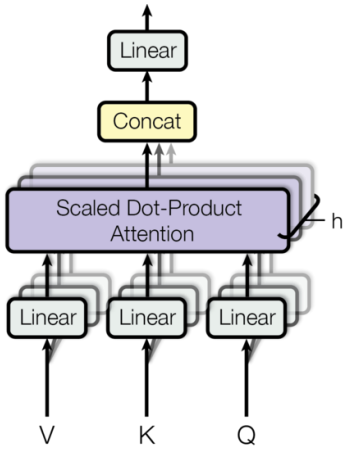
Scaled Dot-Product Attention

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



Multi-Head Attention

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$



Positional Encodings

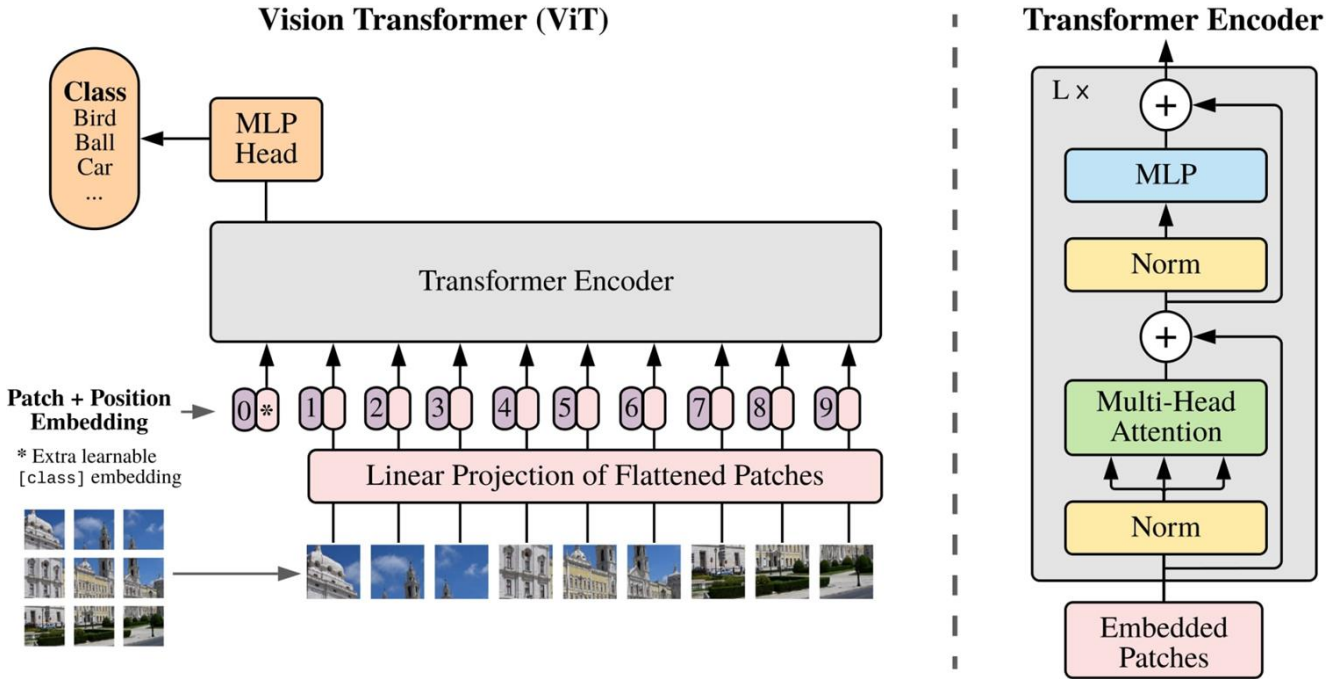
- **Role**: Enables sequence recognition by providing unique positional signals.
- **Types**: Mainly sinusoidal or trainable learned encodings.
- **Integration**: Added to input embeddings before self-attention layers.
- **Purpose**: Maintains position information throughout the transformer.
- **Impact**: Enhances handling of sequence-dependent tasks effectively.

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

where *pos* is the position and *i* is the dimension from d_{model} dimensions of the model

Positional Encodings in Vision Transformer (ViT)



Source: [An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale]

Additional Components

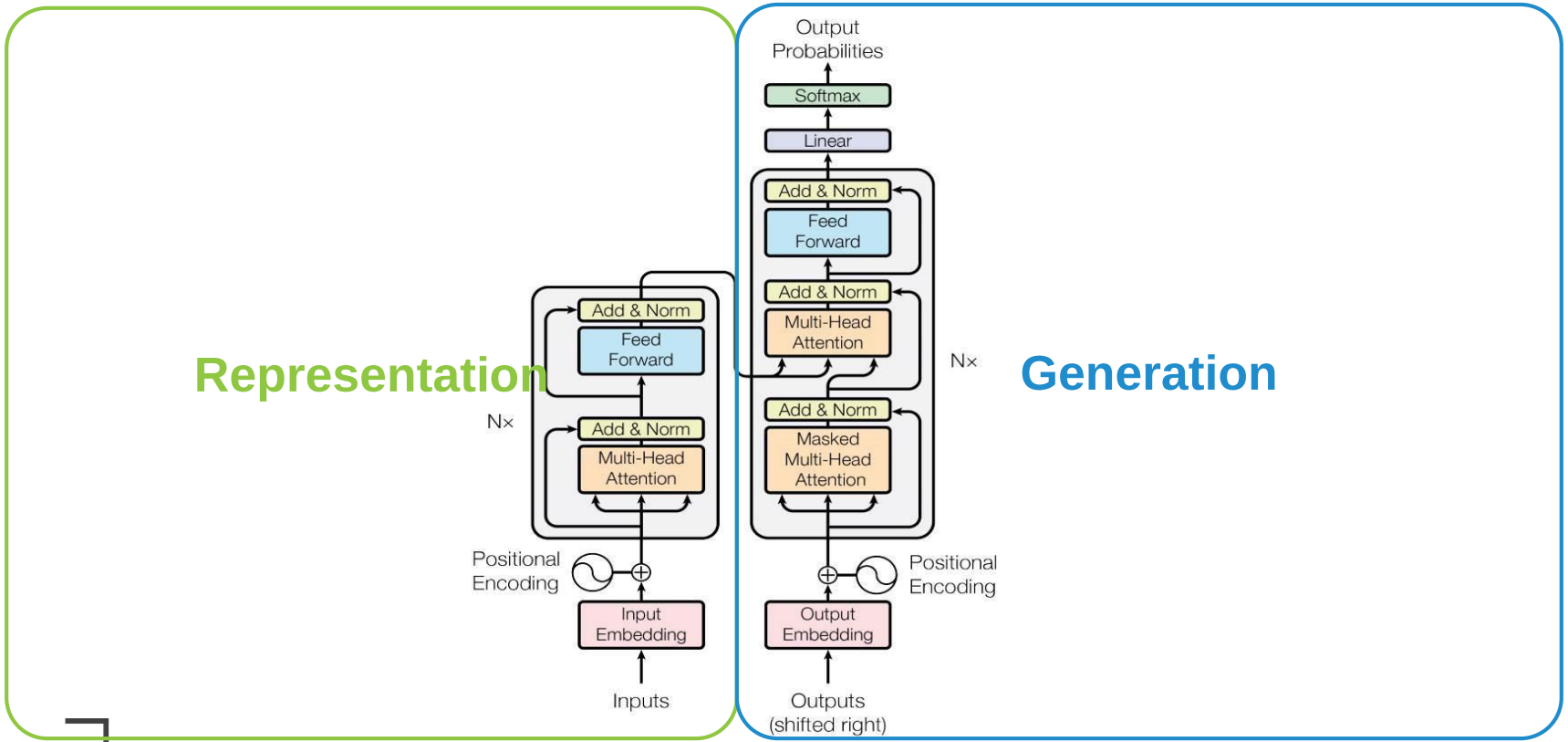
- Transformers use additional components similar to other neural networks
  - Embeddings
  - Layer normalization
  - Feed-forward neural network
  - Softmax
- Transformers utilize traditional neural network training
  - Adam is generally used as the optimizer
  - Regularization like residual dropout and label smoothing are commonly applied.

Challenges with Transformers

- **Training Complexity**: Transformers are costly and time-consuming to train due to their complexity.
- **Model Interpretability**: Their complex mechanisms hinder understanding of decision processes.
- **Resource Requirements**: High memory, processing and data demands limit use in low-resource environments.
- **Latency Issues**: Significant model size can result in high inference latency, impacting real-time applications.
- **Environmental Impact**: Substantial energy consumption for training of large transformer models raises concerns about the environmental footprint.

Transformers in Modern Research and Applications

Transformers for Representation & Generation



Transformers for Representation & Generation

Attributes	Encoder	Decoder
Role	Representation	Generation
Input	Entire input sequence	Encoded embeddings + partially generated sequence
Output	Contextual embeddings of input	Next token prediction or entire output sequence
Token Visibility	All input tokens visible to each other	Restricted to previous and current tokens only
Autoregressive	No	Yes
Models	BERT - Bidirectional Encoder Representations	GPT – Generative Pretrained Transformer

Applications of Transformers

Model Type	Description	Applications
GPT (Generative Pre-trained Transformer)	Autoregressive model that generates text by predicting one word at a time based on previous words.	Text completion, creative writing, chatbots.
BERT (Bidirectional Encoder Representations from Transformers)	Uses masked language modeling to predict missing words from context in both directions, but primarily used for understanding rather than generation.	Text classification, question answering.
ViT (Vision Transformers)	Adapts the Transformer architecture for image classification by treating image patches as tokens in a sequence.	Image classification, object detection, image generation.
LLaVA (Large Language and Vision Assistant)	Autoregressive multi-modal version of LLMs fine-tuned for chat/ instructions.	Video question answering.
CLIP (Contrastive Language-Image Pre-training)	Learns visual concepts from natural language supervision, capable of understanding and generating both text and images.	Image captioning, text-to-image synthesis.
DALL-E	Generative model capable of creating images from textual descriptions, based on GPT-3 specially adapted for images.	Image creation from text, art generation.

Conclusion

- **Revolutionary Impact**: Transformers have reshaped natural language processing and expanded into vision and multimodal applications.
- **Performance Excellence**: They consistently outperform older models in both accuracy and computational efficiency across diverse tasks.
- **Scalable Architecture**: Designed to benefit from increasing data and computational power, making them highly effective in large-scale applications.
- **Present Challenges**: Resource intensity, complexity, and interpretability remain significant challenges.
- **Future Potential**: Research continues to enhance their efficiency, effectiveness, and broaden their applicability.

Resources

Understanding Transformers

“Attention Is All You Need” by Vaswani et al. (2017)
arxiv.org/abs/1706.03762

Language Translation with Transformers
pytorch.org/tutorials/beginner/translation_transformer

Hugging Face - Transformers
huggingface.co/docs/transformers

Rakshit Agrawal

Co-Founder & CEO

Ryddle AI

LinkedIn: linkedin.com/in/rakshit-agrawal/