Paper Summary Report

Paper: Attention Is All You Need

Authors:

· Ashish Vaswani (Google Brain)
· Noam Shazeer (Google Brain)
· Niki Parmar (Google Research)
· Jakob Uszkoreit (Google Research)
· Llion Jones (Google Research)
· Aidan N. Gomez (University of Toronto)
· ·ukasz Kaiser (Google Brain)
· Illia Polosukhin (Google Research)

English Summary

1. The paper introduces the Transformer, a novel neural network architecture for sequence transduction.
2. Traditional sequence transduction models rely on recurrent or convolutional neural networks (RNNs or CNNs).
3. The Transformer uses only attention mechanisms, eliminating recurrence and convolutions.
4. This architecture offers improved parallelization and faster training.
5. Experiments on machine translation tasks demonstrate superior performance compared to existing models.
6. The Transformer achieves a 28.4 BLEU score on the WMT 2014 English-to-German translation task, surpassing previous best results by over 2 BLEU.
7. On the WMT 2014 English-to-French translation task, it sets a new single-model state-of-the-art BLEU score of 41.0.
8. Training the model for English-to-French took only 3.5 days on eight GPUs, a fraction of the training cost of other top-performing models.
9. Recurrent neural networks (RNNs), including LSTMs and GRUs, are established sequence modeling approaches.
10. However, RNNs' sequential nature limits parallelization during training, especially with long sequences.

11. Attention mechanisms enhance sequence modeling by capturing dependencies regardless of distance.
12. Most existing attention mechanisms are used with recurrent networks.
13. The Transformer uses self-attention, relating different positions within a sequence.
14. Self-attention has been used successfully in various tasks such as reading comprehension and summarization.
15. The Transformer is the first transduction model solely relying on self-attention without RNNs or convolutions.
16. The Transformer employs an encoder-decoder structure.
17. The encoder maps an input sequence to a continuous representation.
18. The decoder generates an output sequence based on the encoder's representation.
19. Both encoder and decoder use stacked self-attention and point-wise fully connected layers.
20. Residual connections and layer normalization are used in both the encoder and decoder.
21. The decoder's self-attention prevents attending to future positions, maintaining auto-regressive property.
22. The attention function maps queries and key-value pairs to an output.
23. The output is a weighted sum of values, where weights are determined by a compatibility function.
24. Scaled Dot-Product Attention is used, scaling dot products by 1/·dk to handle large dk values.
25. Multi-Head Attention employs multiple attention layers in parallel with different learned linear projections.
26. This allows attending to information from different representation subspaces.
27. The Transformer uses multi-head attention in three ways: encoder-decoder attention, encoder self-attention, and decoder self-attention.
28. Position-wise Feed-Forward Networks are applied to each position separately.
29. Learned embeddings convert input and output tokens to vectors.
30. The same weight matrix is shared between embedding layers and the pre-softmax linear transformation.
31. Positional encodings are added to input embeddings to incorporate sequence order information.
32. Sinusoidal functions of different frequencies are used for positional encoding.
33. Self-attention layers offer advantages over recurrent and convolutional layers.

34. Self-attention has a lower computational complexity for shorter sequences (n < d).
35. For longer sequences, restricted self-attention can be used to improve efficiency.
36. Self-attention allows for more parallelization compared to recurrent layers.
37. Self-attention has shorter paths between long-range dependencies than recurrent or convolutional layers.
38. The models were trained on the WMT 2014 English-German and English-French datasets.
39. Byte-pair encoding and word-piece vocabulary were used.
40. Training was done on a machine with 8 NVIDIA P100 GPUs.
41. The Adam optimizer with a specific learning rate schedule was used.
42. Three types of regularization were employed: residual dropout, label smoothing.
43. The big Transformer model achieved a BLEU score of 28.4 on English-to-German and 41.0 on English-to-French.
44. The training cost was significantly lower compared to previous state-of-the-art models.
45. Variations in the model architecture were tested, examining the effects of the number of attention heads, key size, and dropout.
46. The Transformer achieved state-of-the-art results on machine translation tasks.
47. Future work includes applying the Transformer to other tasks and modalities.
48. The code is available on GitHub.
49. The authors acknowledge Nal Kalchbrenner and Stephan Gouws for their contributions.

Equations

Attention(Q, K, V) = softmax(QKT/·dk)V

FFN(x) = max(0, xW1 + b1)W2 + b2

lrate = dmodel·0.5 * min(step_num·0.5, step_num * warmup_steps·1.5)