Transformers: Understanding the Pipeline

Input → Computation → Output → WHY

Week 5: Transformers

The Simple Goal

INPUT:

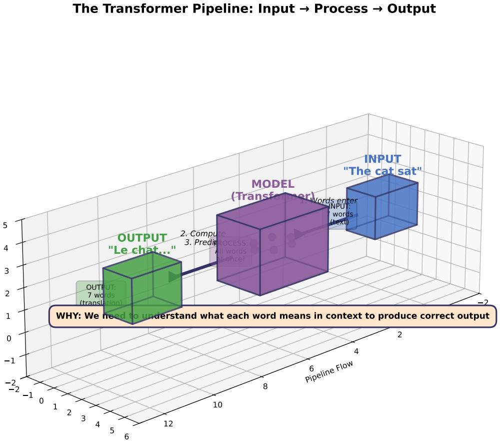
- Text: "The cat sat on the mat"
- 7 words (English)

OUTPUT:

- Text: "Le chat était assis sur le tapis"
- 7 words (French)

THE TASK:

- Translate
- Predict next word
- Answer questions

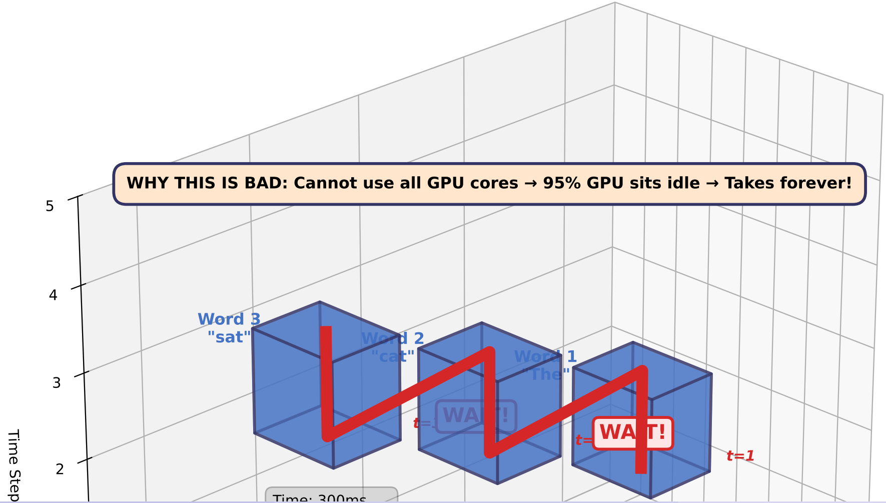


The Transformer Pipeline: Input → Process → Output

WHY: We need to understand what each word means in context to produce correct output

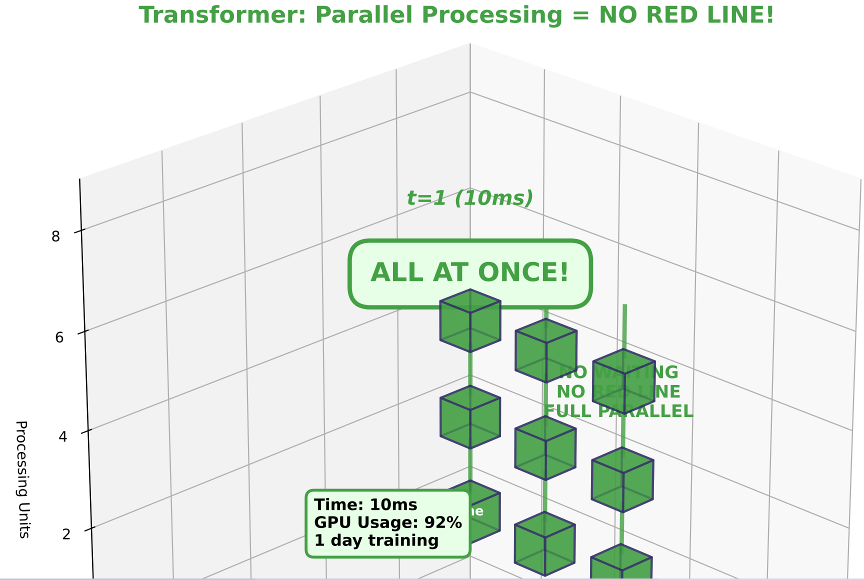
The Old Way: RNN (Sequential Processing)

RNN: Sequential Processing = RED LINE Bottleneck

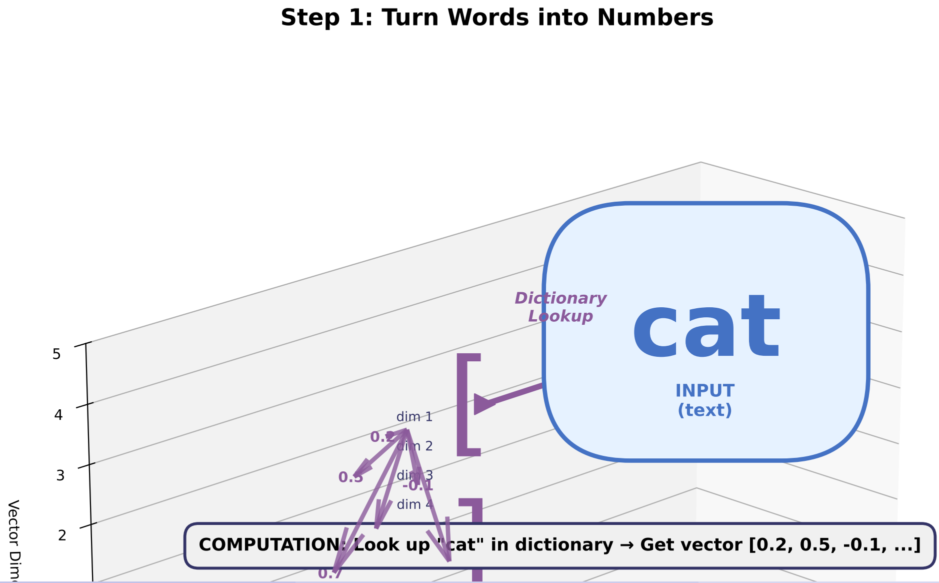


WHY THIS IS BAD: Cannot use all GPU cores → 95% GPU sits idle → Takes forever!

The New Way: Transformer (Parallel Processing)



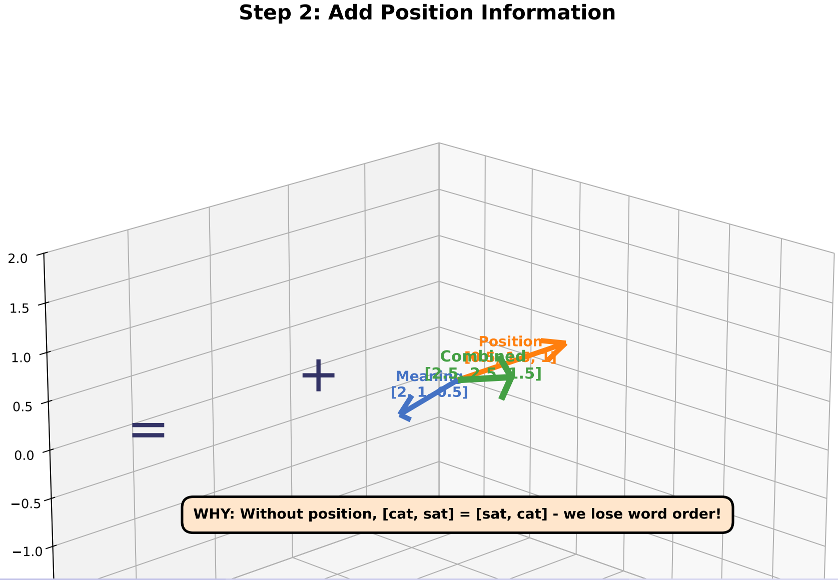
Step 1: Turn Words into Numbers



Step 1: Turn Words into Numbers

COMPUTATION: Look up "cat" in dictionary → Get vector [0.2, 0.5, -0.1, ...]

Step 2: Add Position Information

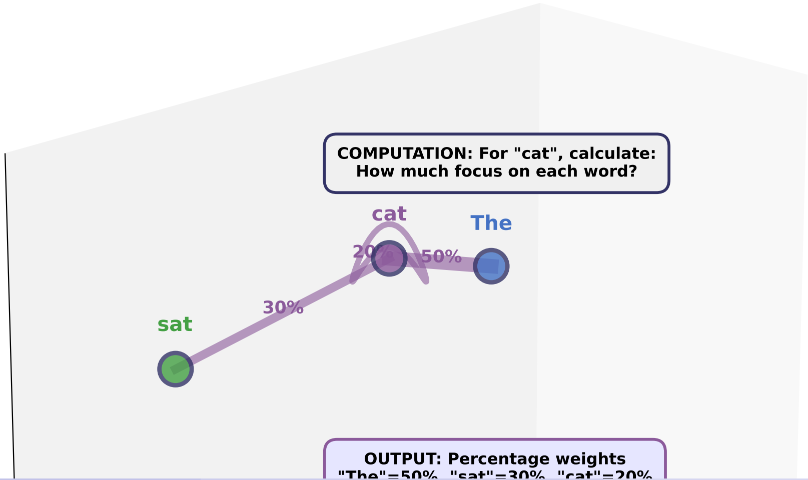


Step 2: Add Position Information

WHY: Without position, [cat, sat] = [sat, cat] - we lose word order!

Step 3: Calculate Attention (Who Looks at Who)

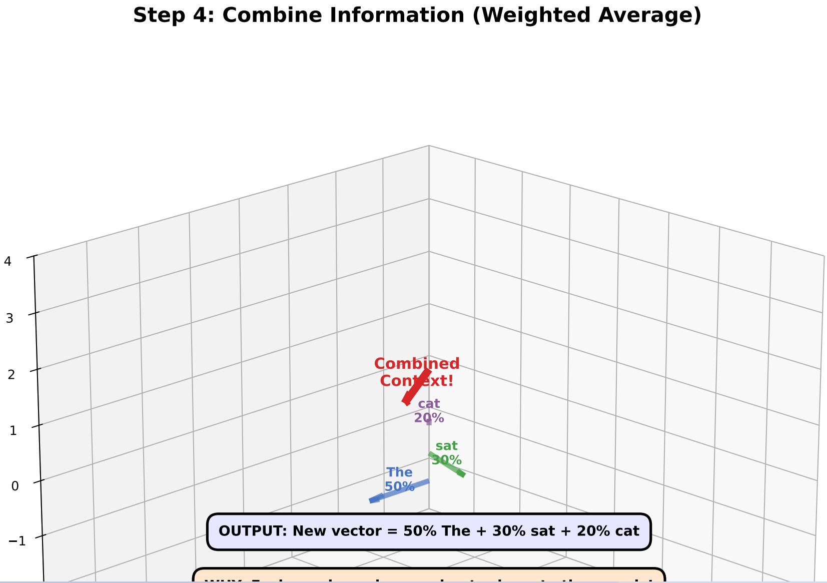
Step 3: Calculate Attention (Who Looks at Who)



COMPUTATION: For "cat", calculate:
How much focus on each word?

OUTPUT: Percentage weights
"The"=50%, "sat"=30%, "cat"=20%

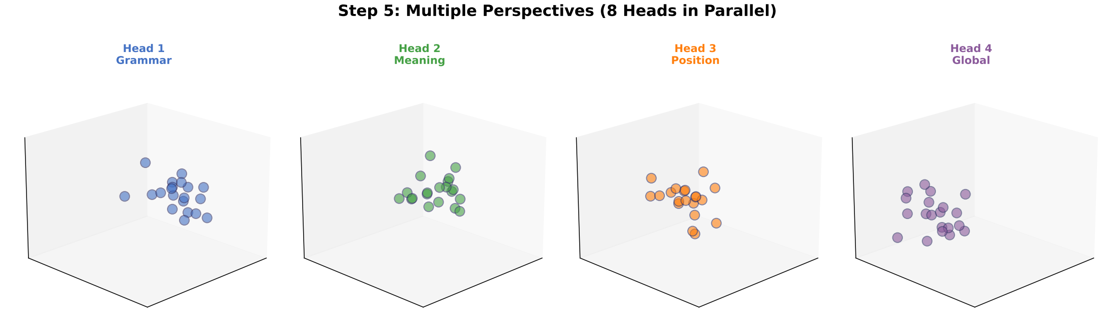
Step 4: Combine Information (Weighted Average)



Step 4: Combine Information (Weighted Average)

OUTPUT: New vector = 50% The + 30% sat + 20% cat

Step 5: Multiple Perspectives (Multi-Head Attention)



Step 5: Multiple Perspectives (8 Heads in Parallel)

Head 1
Grammar

Head 2
Meaning

Head 3
Position

Head 4
Global

Step 6: Final Prediction

INPUT: Context-enriched vectors

- Each word knows about:
  - Its meaning
  - Its position
  - Related words (8 perspectives)

COMPUTATION:

- Feed through prediction layer
- Calculate probabilities for each possible next word

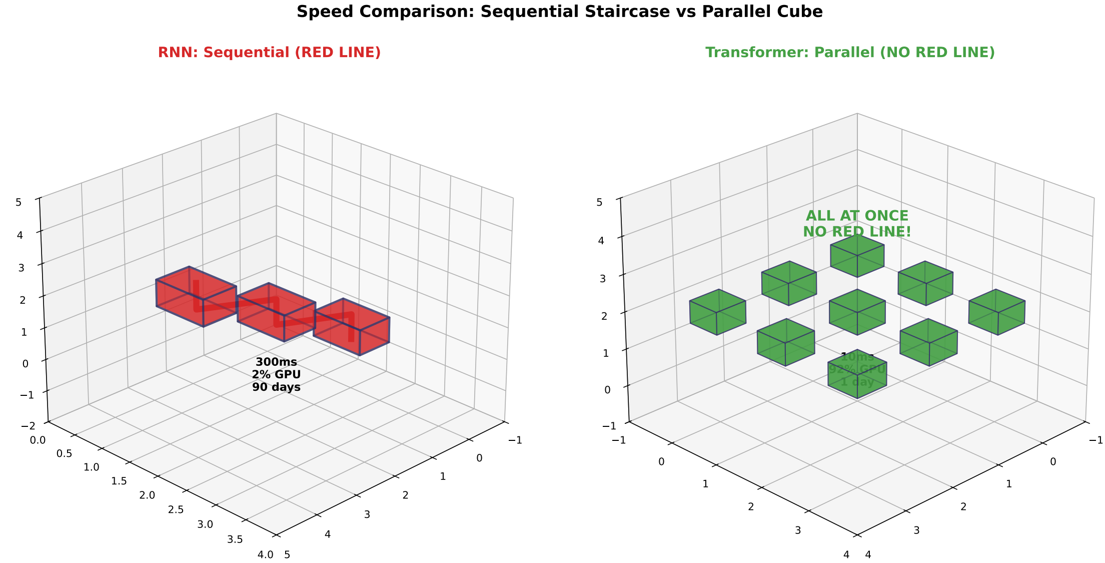
OUTPUT:

- Next word probabilities:
  - "Le": 85%
  - "The": 10%
  - Other: 5%
- Pick highest: "Le"

Result: Translation complete!

WHY: This is what we wanted all along - accurate prediction from context!

The Speed Secret: No Red Line!



Speed Comparison: Sequential Staircase vs Parallel Cube

RNN: Sequential (RED LINE)

Transformer: Parallel (NO RED LINE)

Real Numbers: The Proof

Actual Experimental Results ("Attention Is All You Need", 2017):

Model	Training Time	GPU Usage	Quality (BLEU)
RNN	90 days	2%	24.5
RNN + Attention	45 days	5%	28.4
Transformer	**1 day**	**92%**	**28.4**

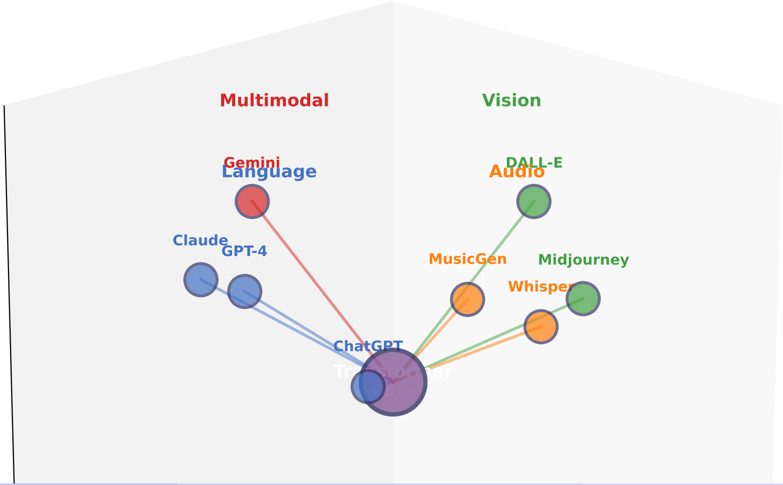
KEY INSIGHT:

- Same quality in 45x less time
- Better GPU utilization (2% → 92%)
- Enabled modern AI scale

WHY BELIEVE THIS: Published results, reproduced worldwide, powers all modern LLMs

What This Enabled: 2024 Landscape

2024 Landscape: Transformers Power Everything



The Tradeoff: What We Gave Up

Advantages (PRO):

- 100x faster training
- No sequential bottleneck
- 92% GPU utilization
- Works on any data type
- Enabled modern AI

Disadvantages (CON):

- More memory (quadratic)
- Needs more data
- Limited sequence length
- More complex to tune

THE DECISION: Speed + quality ¿ memory cost for modern AI

WHY ACCEPT TRADEOFF: Memory is cheap, time is expensive. Better to train fast even if uses more RAM.

Summary: The Pipeline Recap

The 6-Step Pipeline (NO RED LINE!):

1. **Words → Numbers:** Dictionary lookup (embeddings)
2. **Add Position:** Vector addition (meaning + position)
3. **Calculate Attention:** Who looks at who? (percentage weights)
4. **Combine Information:** Weighted average (context-enriched)
5. **Multiple Perspectives:** 8 heads in parallel (grammar, meaning, position, ...)
6. **Predict Output:** Final layer (translation/next word)

KEY INSIGHT: All in parallel - NO RED LINE!

- Result: 90 days → 1 day (90x speedup)
- Enabled: ChatGPT, GPT-4, DALL-E, Whisper, ...

Next Week: Pre-training & Fine-tuning - Now that training is fast, we can train HUGE models!

Transformers

Understanding the Pipeline

Input → Computation → Output → WHY

Questions?