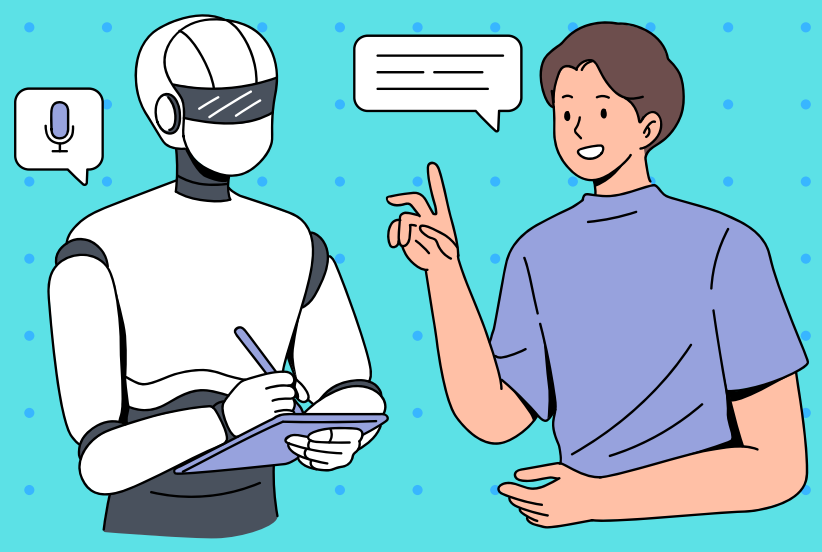
Understanding AI-Powered Language Models

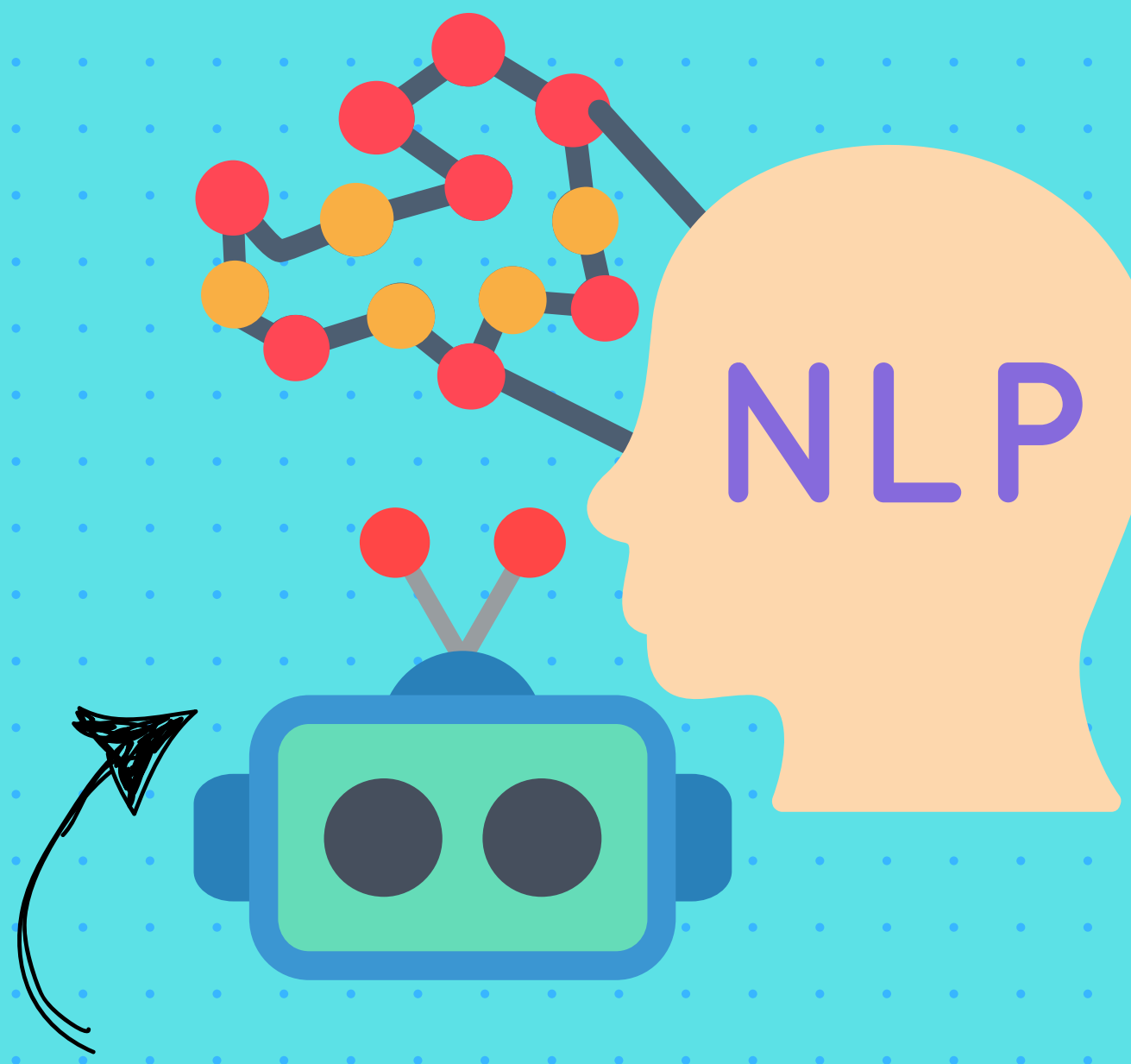


# From RNNs to Transformers: explained simply

A beginner-friendly guide to NLP's key models—RNNs, Transformers, & Attention Mechanisms

@Dzan Dedukic

NLP Series #2




# Following themes are covered

1. 📘 **Understanding NLP Sequence Models**
2. ⚙️ **Why Sequence Matters in NLP**
3. 📚 **History of Sequence Models**
4. 🔄 **Feedforward vs. Recurrent Networks**
5. 📚 **From RNNs to Transformers: The Evolution**
6. 🎯 **Attention Mechanisms**
7. ⚠️ **Challenges & Future Directions**
8. 🌟 **Real world transformer Models**
9. 📍 **Key Takeaways**

# 1. 🚀 Understanding NLP sequence models

**Did you know?**

AI models like GPT, BERT, and T5 use sequence-based learning to process language—helping chatbots, translators, and search engines sound more human!

**Definition:**
Sequence models enable AI to process language in a structured way, preserving word order, context, and dependencies between words.

**Fun fact:**
This approach powers your favorite tools like chatbots 🤖, search engines 🔍, and voice assistants 🗣️ to make them sound more human!

# 2. Why Sequence Matters in NLP

**Under the Hood:**

Language is context-dependent—the meaning of a word changes based on surrounding words.

Sequence models retain memory, allowing AI to process sentences logically and cohesively.

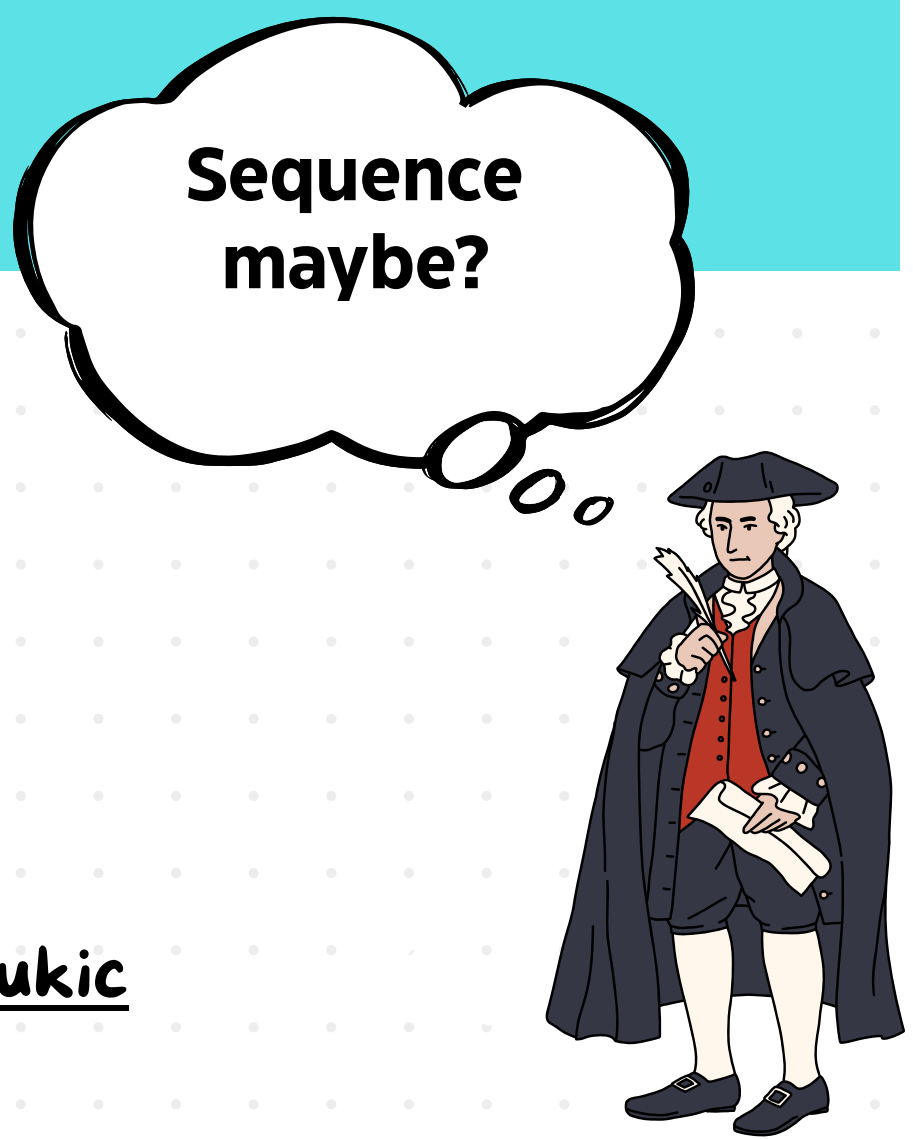
🚀 **Why This Is Important**
🎯 Without sequence models, AI couldn't understand full sentences or context.
📍 Essential for chatbots, translation, summarization, and AI-powered assistants.
💡 NLP advancements depend on better sequence models to improve AI comprehension.

@Dzan Dedukic

# 3. History of Sequence Models

## 📅 When Were Sequence Models Introduced?

- Early NLP models used statistical approaches (e.g., Markov Chains).
- RNNs became popular in the 1980s–1990s but struggled with long-range dependencies.
- LSTMs (1997) improved memory retention, paving the way for modern Transformers (2017, Attention is All You Need).




@Dzan Dedukic

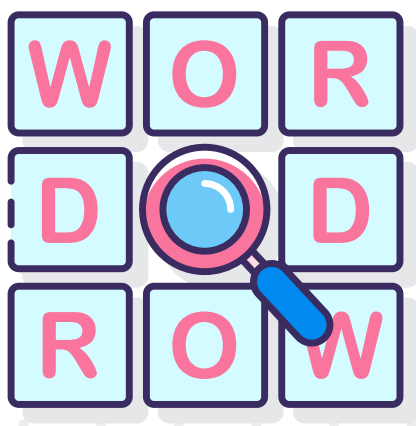
# 4. ➡ Feedforward network

**What is it**

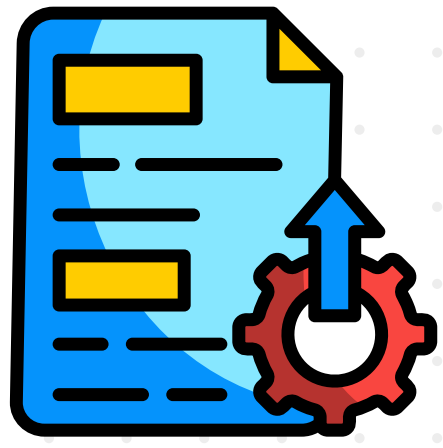
- The simplest type of neural network, where information moves in one direction—from input to output.
- Process input word by word, without memory.
- Each word is treated separately, meaning context is lost.
- Limitation: Cannot understand sequence or relationships between words.

**Input Layer**

**Hidden Layer**




**Output Layer**



# 4. ➡ Feedforward network: Drawbacks and use-case

**Used for:**
Commonly used for tasks like image classification but not ideal for language processing.

🧠 **Why FFNs Struggle in NLP?**

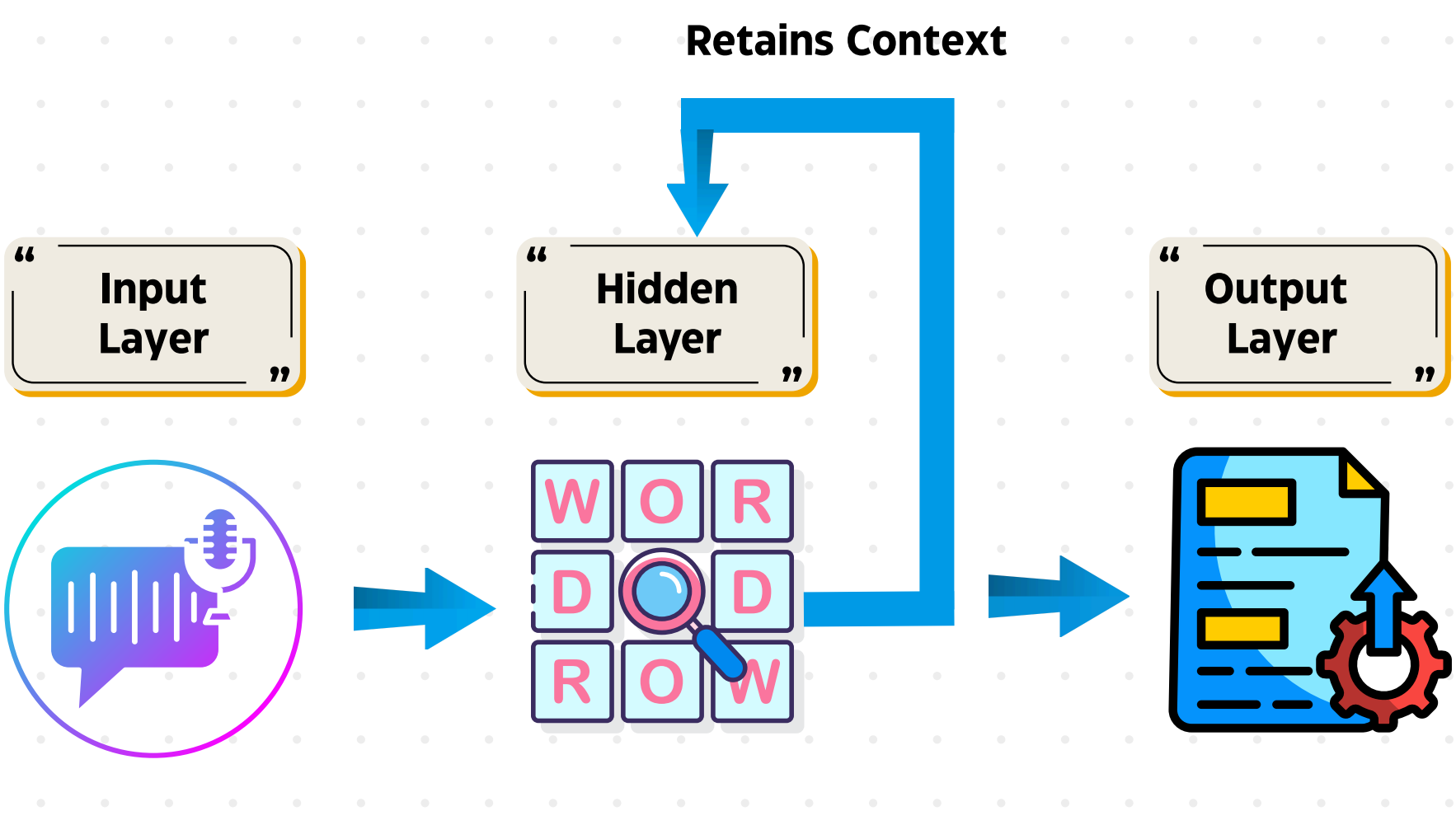
❌ No memory—can’t retain information from previous words.
❌ Fails to understand word relationships or sentence structure.
❌ Ignores context, making it ineffective for sequential tasks like speech recognition or translation.

💡 Usage Example:
FFNs **work well for single-word classifications** like:
✅ Spam detection → Classifying individual words as spam/not spam.
✅ Text sentiment on single words → "Happy" = positive, "Sad" = negative.

# 4. 🔁 Recurrent Neural Network

**What is it**

🔁 A type of neural network designed to handle sequential data, where past inputs influence future outputs.
🔄 Processes word-by-word but retains memory, storing context from previous words.
🔗 Uses hidden states to remember past information, making it effective for language tasks.
⚠️ Limitation: Struggles with long-term dependencies due to vanishing gradients.



# 4. 🔁 Recurrent Neural Network: Use case

🎯 **Used for:**
✅ Text generation → AI writing coherent sentences.
✅ Machine translation → Understanding sentence structure when translating languages.
✅ Speech recognition → Recognizing words based on previous sounds.

**Why RNNs Are Better for NLP?**
✔ Retains memory—previous words affect next predictions.
✔ Understands sequence relationships instead of treating words in isolation.
✔ Captures context, making it useful for tasks like chatbots, translation, and speech processing.
💡 Usage Example:
✅ Chatbots → "Hi, how are..." helps predict "you today?"
✅ Predictive text → Suggests next words based on previous input.

# 5. From RNNs to Transformers

## 🔁 Why Did We Need a Better Model Than RNNs?

- RNNs process words sequentially, making them slow for long sentences.
- Struggles with long-range dependencies due to vanishing gradients.
- Hard to parallelize, making large-scale NLP inefficient.

**So, what is tranformer then..?**

✓ A neural network architecture designed for handling sequential data without recurrence.

✓ Uses Self-Attention to process all words simultaneously, rather than step by step like RNNs.

✓ Enables parallel computation, making language tasks faster and more efficient.

**Scroll to next slide to see examples.**

# 5. From RNNs to Transformers

Think of a transformer like a super-focused reader.
It doesn't just read words one by one—it looks at every word in a sentence and figures out how they're all connected, instantly!

🔍 **For example:**

- **"She poured water from the pitcher to the cup until it was full."**
- The transformer knows "it" means "cup," not "pitcher."
- **If you change it to "until it was empty,"** the model knows "it" now means "pitcher."

💡 No More Tedious Labeling!

- Old days: You needed to hand-label tons of data for AI.
- Now: Transformers can "teach themselves" by finding patterns in unlabeled data.

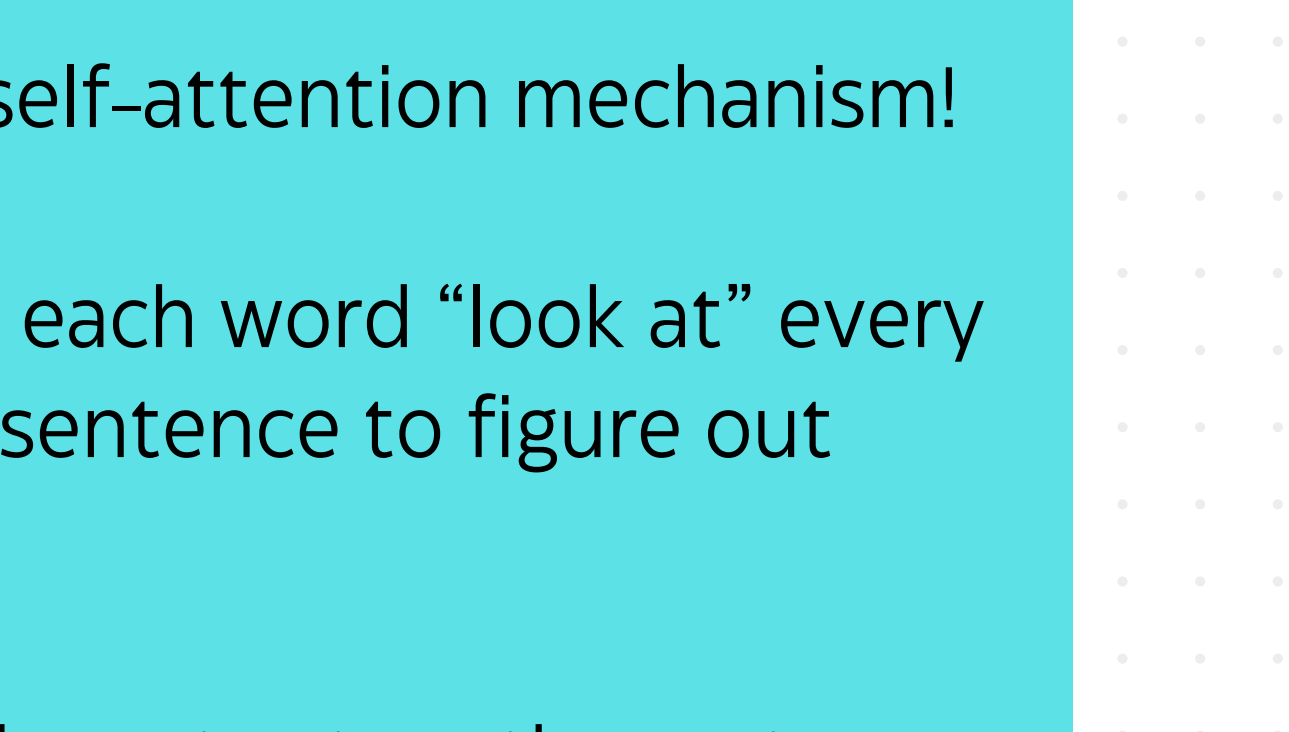
# 6. Self-attention mechanism

**How Do Transformers Work really then?**

Transformers process sequences (like sentences) all at once—not word by word like RNNs.
Their secret? The self-attention mechanism!

Self-attention lets each word “look at” every other word in the sentence to figure out what’s important.

For example, in “The cat sat on the mat because it was warm,” the model uses self-attention to understand that “it” refers to “the mat.”



@Dzan Dedukic

# 6. Self-attention mechanism: Under the hood

- Each input word is turned into a vector (embedding), and self-attention computes how much each word should pay attention to every other word—using matrices and dot products.
- These attention scores help the model build better, context-aware word representations.
- Transformers use multi-head attention (many attention mechanisms in parallel) for richer understanding.
- They stack layers of these attention blocks, allowing deeper learning of complex patterns.
- No recurrence, no convolution, just attention and position encodings to keep track of word order!

@Dzan Dedukic

# 6. Self-attention mechanism: How model pays attention

**Sentence**:
The keys open the doors when they are turned.
1 **Embeddings**:
Each word (“keys”, “doors”, “they”...) is turned into a vector representing its meaning and position.
2 **Self-Attention**:
For every word, the model calculates how much it should “pay attention” to all other words.
For “they”, most attention goes to “keys” (because “they” = “keys”).
3 **Context-Aware Representation**:
The model builds a new, smarter representation for each word, using these attention scores.
4 **Multi-Head Attention**:
It repeats this in parallel (“multi-heads”), so the model can capture different types of relationships.
5 **Stacked Layers**:
Several layers of attention blocks are stacked, letting the model understand complex patterns.
**Result**:
The model figures out that “they” refers to “keys”—all thanks to self-attention!

# 7. ⚠ Challenges & Future Directions

### 🚧 Current Challenges in NLP

- ❌ High Computational Cost:
- Transformers require massive computing power.
- ❌ Bias in AI Models:
- Language models can inherit biases from their training data.
- ❌ Long Context Limitations:
- They still struggle to process very long documents effectively.

### 🔮 Future Directions in NLP

- 🚀 Sparse & Efficient Transformers:
- Reducing complexity while keeping accuracy.

### 🧠 Beyond Transformers?

- Exploring new architectures for even better performance.

# 8. 🌟Real world transformers

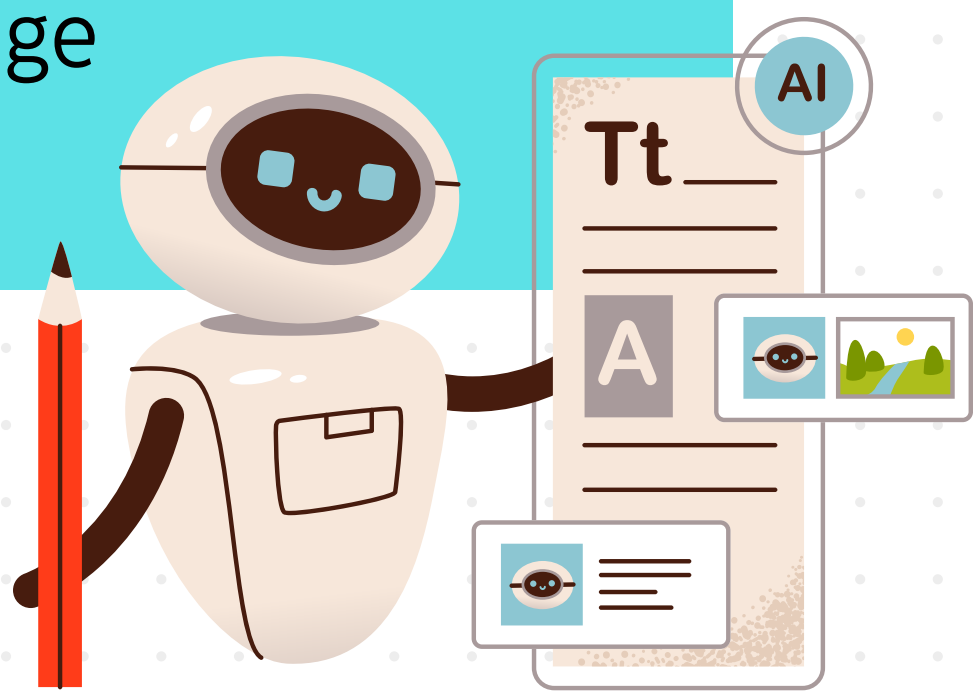
🌟 **Real-World Transformer Models (few examples)**

- BERT (Google):
  - Powers Google Search by understanding queries and web content contextually.
- GPT (OpenAI):
  - Generates human-like text, powers chatbots, code assistants, and creative writing.
- T5 (Google):
  - Excels at translation, summarization, and answering questions.
- AlphaFold (DeepMind):
  - Predicts protein structures—revolutionizing biology and drug discovery.
- GatorTron (University of Florida):
  - Analyzes clinical health records to help medical research and patient care.

**Transformers are everywhere:**
From recommending your next movie 🎬 to breaking language barriers 🌏, and even helping scientists fight diseases 🧬!

# 9. Key Takeaways


📌 **NLP needs sequence models** → Without them, AI can't understand word relationships.
📡 **Feedforward Networks** process words individually but lack memory.
🔁 **RNNs** introduced memory but struggled with long sentences.
🎯 **Attention Mechanisms** improved AI focus on important words dynamically.
🚀 **Transformers** revolutionized NLP with parallel processing & self-attention.
⚠️ **Challenges remain** → Computational costs, biases, and long-context limitations.
🔮 **Future directions** → More efficient models & ethical AI language development.



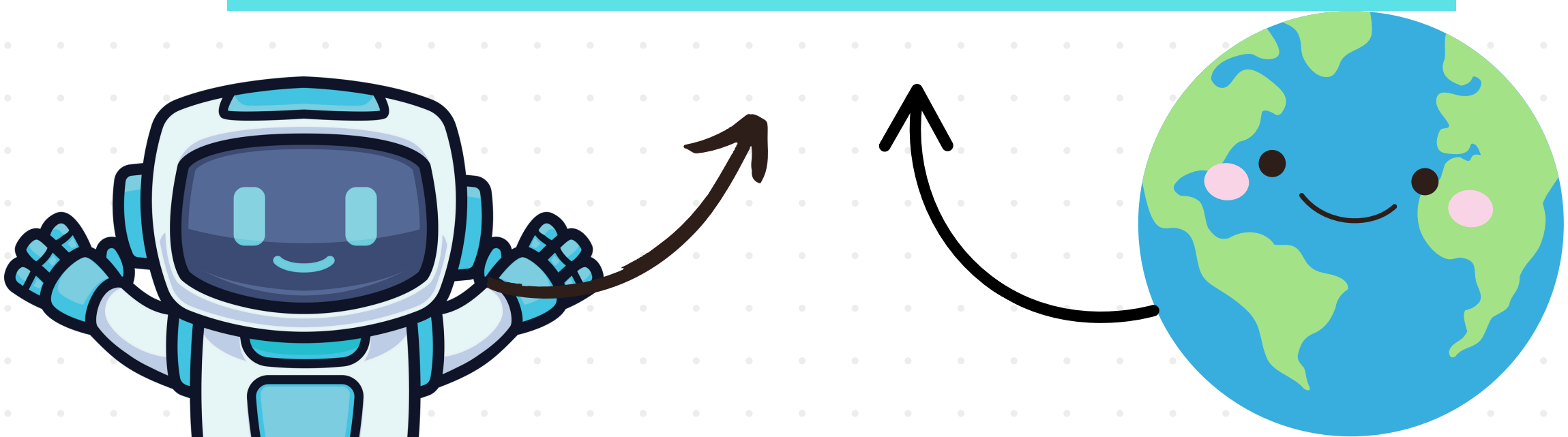
# 10. Ethical AI: Building a Transparent & Inclusive Future

🛠️ AI & ML are powerful tools—but their impact depends on how we use them.

💡 Transparency, fairness, and accountability must be at the heart of every AI-driven solution. Technology should empower, not harm, and ensure inclusivity rather than bias.

🌍 The key is in our hands. By prioritising ethical decision-making, we can build AI systems that enhance lives, foster trust, and contribute to a better world for all of humanity.

🔍 Join the AI movement that's making powerful language models accessible to everyone for more inclusive and better world.



# Are you AI enthusiast as well? 🚀

Did you enjoy this carousel and find it helpful?

Follow me for more content like this, and let’s share knowledge to grow together in the world of AI! 💡✨

FOLLOW

@Dzan Dedukic