What Are Embeddings?

An **embedding** is a **vector representation of data** — usually a word, sentence, or token — that captures its **meaning**, **context**, or **features**, using a dense list of numbers.

Instead of representing a word as just a unique ID (like `word_id = 102`), we represent it as something like:

```
"cat" → [0.2, -1.5, 0.9, 0.3, ..., 0.7]  # A 300-dimensional vector
```

Words that are **similar in meaning** will be **close together** in embedding space.

Why Embeddings Matter

Embeddings are the foundation of how language models "understand" things. They help models:

- **Recognize synonyms** (“king” ≈ “monarch”)
- **Understand analogies** (“Paris is to France” as “Berlin is to Germany”)
- **Generalize** across similar patterns (e.g., recognizing that “dogs bark” and “wolves howl” are similar relationships)

Types of Embeddings

1. Static Embeddings (same vector for a word no matter the context)

- **Word2Vec**
- **GloVe**
- **FastText**

2. Contextual Embeddings (change depending on the sentence)

- **ELMo**
- **BERT**
- **GPT**

These are extracted from **hidden layers** of transformer models.

What Is Self-Attention?

Self-attention is a mechanism that allows a model to **look at all the words in a sentence at once** and decide **which other words are important** for understanding each word.

It answers:

> "When processing word X, which other words should I pay attention to?"

How It Works (At a High Level)

For each word, the model creates three vectors:

- **Query (Q)** – What am I looking for?
- **Key (K)** – What do I offer?
- **Value (V)** – What info do I carry?

Then for each word:

1. It computes the similarity between its **query** and all the **keys** in the sentence.
2. These scores become **attention weights**.
3. It uses the weights to **blend the value vectors** from all the words.
4. The result is a **context-aware vector** for that word.

Example:

Sentence:
"The cat sat on the mat."

When processing "sat", the model might pay more attention to "cat" (who sat) and "mat" (where it sat), rather than "the".

Why It Matters

- **Captures relationships between words**, no matter how far apart they are.
- Allows parallel processing (unlike RNNs, which go word by word).
- Is the core reason transformers outperform older models in NLP.

Summary

> Self-attention lets every word in a sentence look at every other word and decide what's important — helping the model understand language in a smart, flexible way.