Tokenization Fundamentals

Breaking Text Into Processing Units

What Are Tokens?

- Smallest units of text that LLMs can process.
- Not always words: Can be parts of words, whole words, or punctuation.
- Subword tokenization: Most modern LLMs use BPE (Byte-Pair Encoding).
- Language agnostic: Works across different languages and scripts.

Tokenization Examples

"Hello, world!"

```
["Hello", ",", " world", "!"]
```

→ Token count: 4 tokens

"Understanding tokenization"

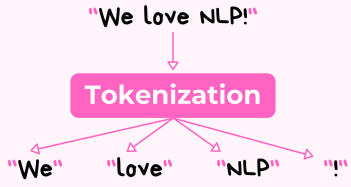
```
["Under", "standing", " token", "ization"]
```

→ Token count: 4 tokens

"GPT-4 is amazing!"

```
["GPT", "-", "4", " is", " amazing", "!"]
```

→ Token count: 6 tokens



Key Implications

- **Context Limits:** Token count determines what fits in a context window.
- **Cost Factor:** Many APIs charge per token processed.
- **Performance Impact:** More tokens can lead to longer processing times.
- **Language Efficiency:** Some languages are more "token-efficient" than others.

🛠️ **Pro Tip:** Use tokenizer tools (like tiktoken for OpenAI models) to count tokens before API calls to optimize cost and performance.