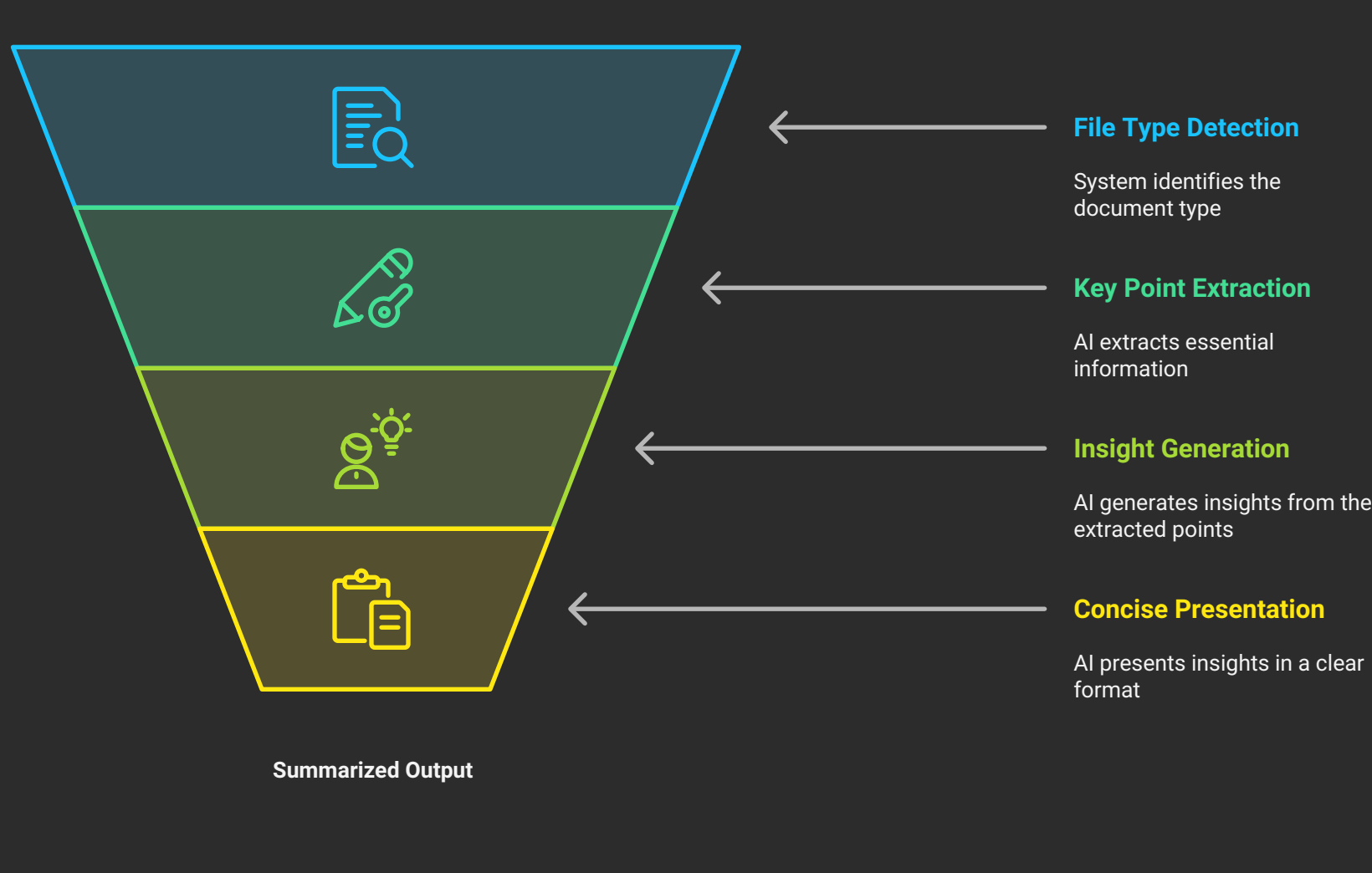


Aitalk Summarization Feature Documentation

This document outlines the functionality of the Aitalk summarization feature, which allows users to summarize the content of various file types, including .txt, .pdf, and .docx. By leveraging AI, the tool extracts key points and insights from the provided documents, presenting them in a clear and concise manner. The following sections detail the usage, operational mechanics, and tips for effective use of the summarization feature.

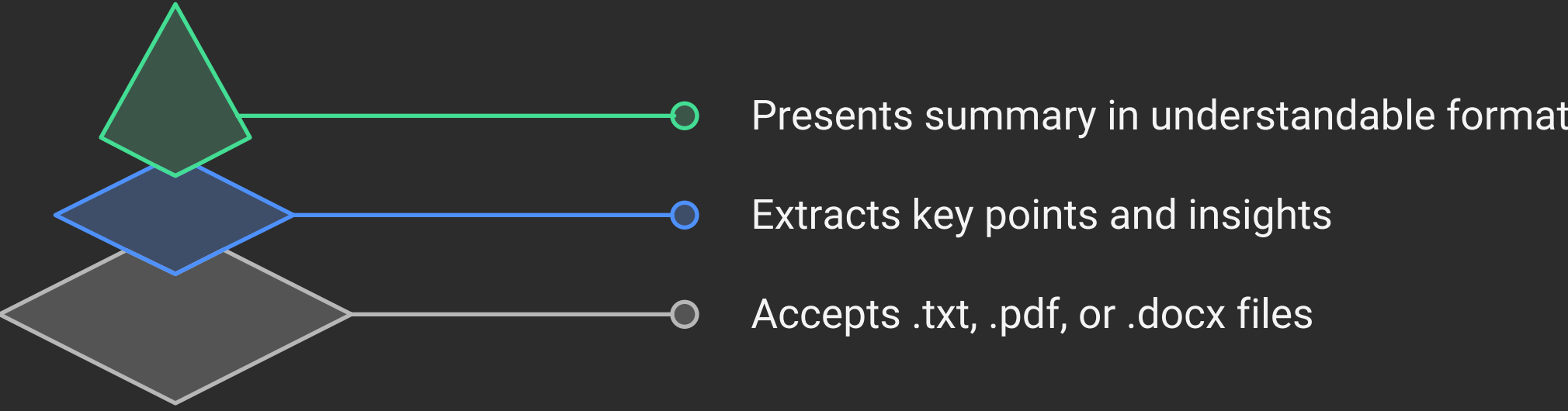
AI Document Summarization Process



Purpose

Summarize the content of a .txt, .pdf, or .docx file using AI, extracting the most important points and insights in a human-readable format.

AI Summarization Process

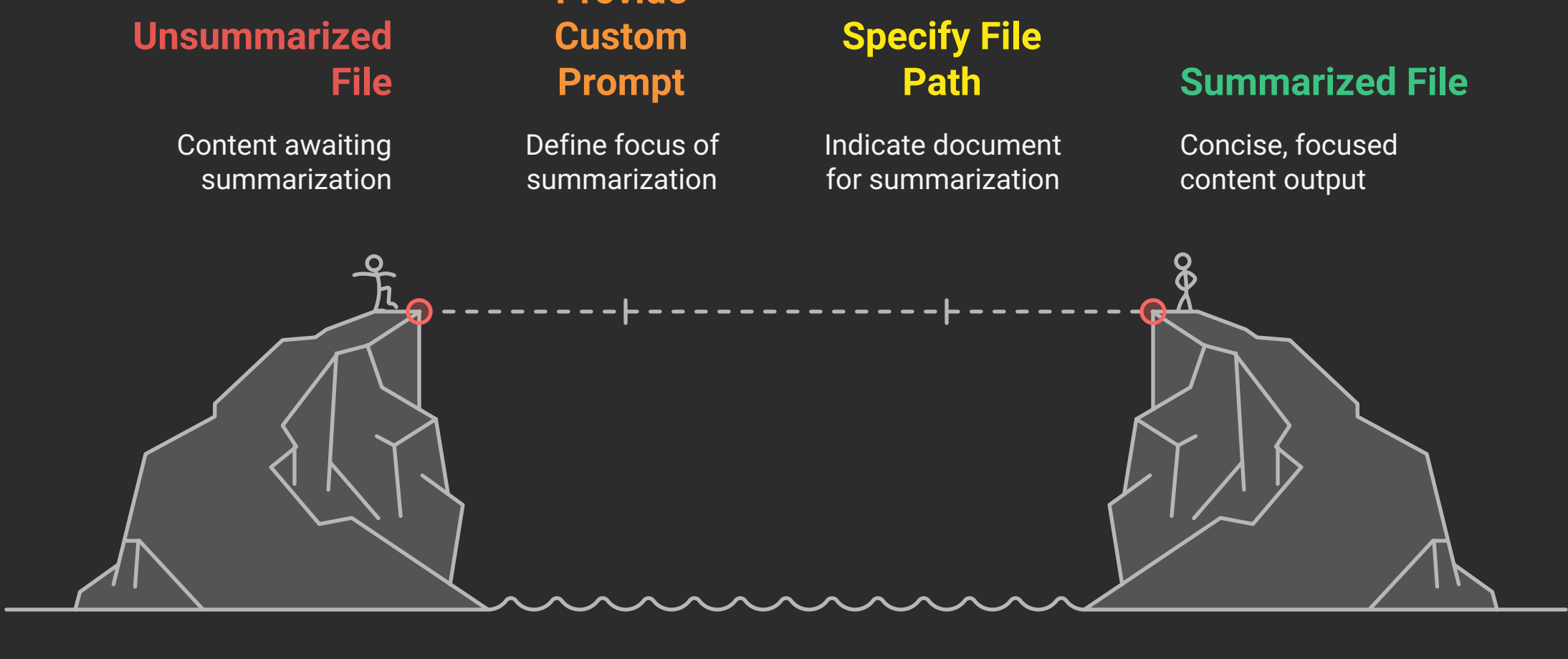


Usage

To utilize the summarization feature, you need to provide two arguments:

1. Your custom prompt (what you want the summary to focus on).
2. The path to the file you want summarized.

Utilizing the Summarization Feature



Example Command

```
aitalk --summarise "summarise this file" file.txt
```

How It Works

File Type Detection & Extraction

The tool automatically detects the file extension and processes the file accordingly:

- .txt: Reads the file as plain text.
- .pdf: Utilizes PyMuPDF to extract text from each page.
- .docx: Employs python-docx to extract text from each paragraph.

Prompt Construction

The tool combines your custom prompt with the extracted content from the file into a single prompt for the Language Learning Model [LLM].

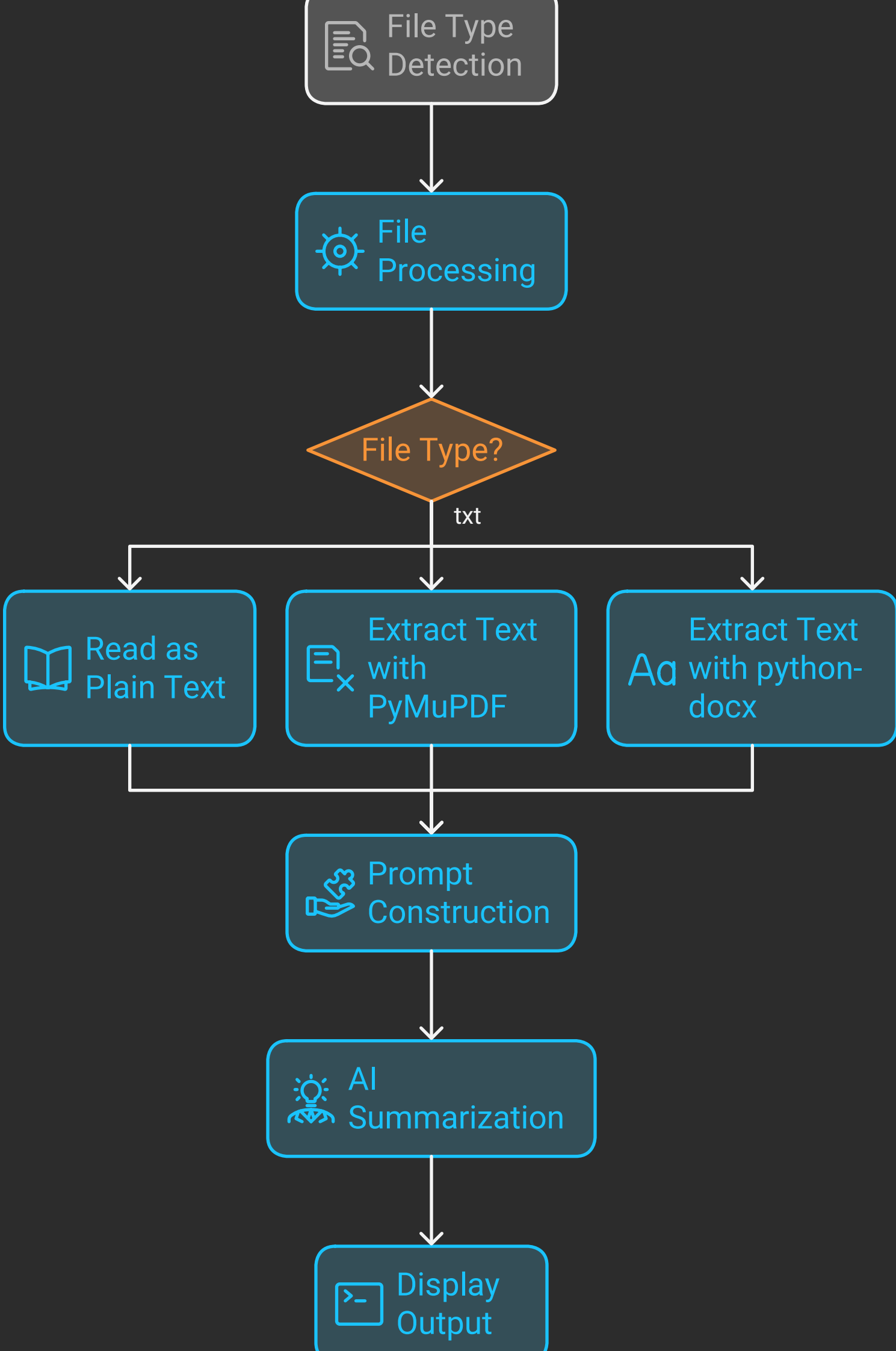
AI Summarization

The constructed prompt and file content are sent to the Groq LLM, which generates a concise, human-readable summary.

Display Output

The summary is printed directly in your terminal for easy access.

AI-Powered File Processing and Summarization



Code Explanation

- **Entry Point:** aitalk.py
 - Detects the --summarise flag, extracts the prompt and file path, and invokes the summarise_file function.
- **Main Logic:** summarise_utils.py
 - extract_text: Handles reading and extracting text from supported file types.
 - summarise_file: Builds the prompt, calls the LLM, and prints the summary.

Tips

- The tool works seamlessly with .txt, .pdf, and .docx files.
- For larger files, be aware that summarization may be truncated due to the LLM's context window limitations.
- Ensure that the required Python packages [PyMuPDF, python-docx] are installed for proper functionality.

Ready for the next feature? Just say "next"!

Tool Usage and File Handling

