

VIDEO INDEXING

STEP 4: METADATA CONSOLIDATION



Gather and structure metadata to describe video segments for indexing.

Metadata includes:

Technical: File format, resolution, duration.

Descriptive: Keywords (e.g., "dog," "park"), sentiment (e.g., positive).



Techniques Used



TECHNICAL METADATA EXTRACTION



TOOL

FFmpeg (ffprobe) to extract file details.



HOW

Parses video file to get format (e.g., MP4), resolution (e.g., 1920x1080), duration.



WHY

Enables fast lookup of segments (e.g., “dog” returns segment IDs).



Techniques Used



KEYWORD EXTRACTION AND CLEANING



TOOL

spaCy for natural language processing (NLP).



HOW

Tokenization: Splits text into words.
Lemmatization: Reduces words to base form (e.g., "running" → "run").
Stop-word Removal: Excludes noise (e.g., "the," "is").



WHY

Creates clean, meaningful keywords for search (e.g., "dog," "park").



Techniques Used



SENTIMENT ANALYSIS



TOOL

Hugging Face transformers with BERT-based model.



HOW

Classifies text as positive, negative, or neutral using contextual understanding.



WHY

Adds emotional context (e.g., "positive" for happy scenes).



Techniques Used



METADATA STRUCTURING



TOOL

Python's json module.

HOW

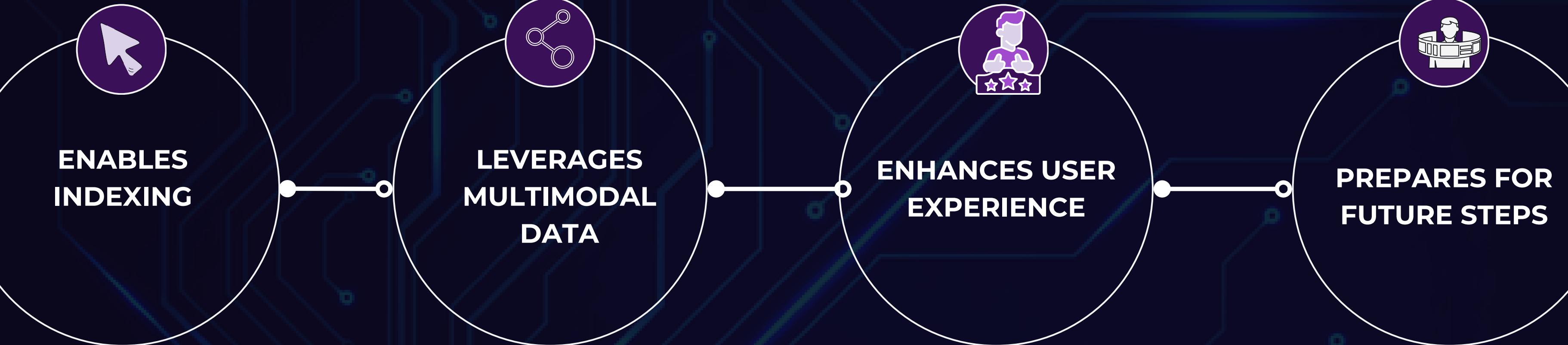
Formats metadata into a JSON schema with segment IDs, timestamps, and metadata.



WHY

Ensures compatibility with indexing tools like Elasticsearch.

Importance of Step 4



Structured metadata feeds into inverted indices or vector databases for fast retrieval.

Example: Keywords like "dog" map to segment IDs in an inverted index.

Combines modalities (text, audio, visual) for richer queries.

Example: Search "dog" finds segments with "dog" in text and visuals.

Sentiment and keywords allow thematic searches (e.g., "find positive dog scenes").

Sets up Step 5 (indexing) and Step 6 (search/retrieval).