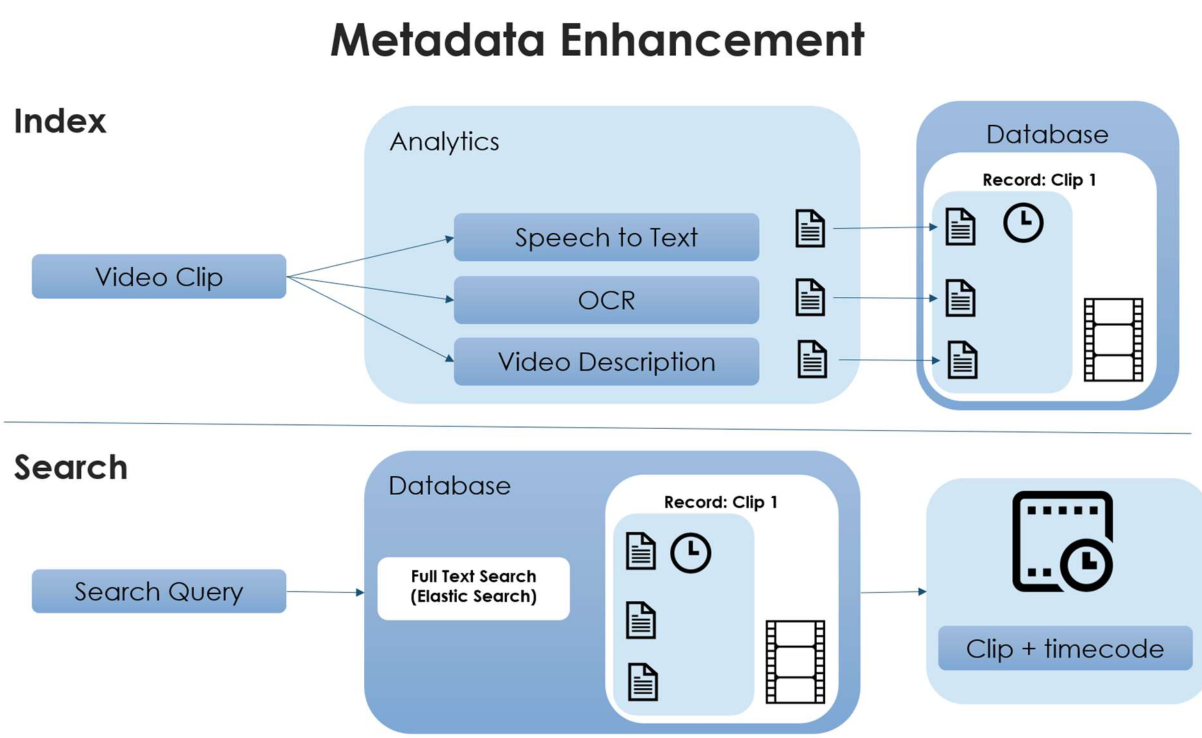
What is Semantic Search and why use it?

Video archives are everywhere. PAM, MAMs, deep archives. Finding the right moment inside hours of footage remains one of the hardest problems in media production. Traditional search tools rely on metadata: structured tags, transcripts, and scene descriptions that someone has to define and generate up front. If you didn't think to label it, you can't find it.

Semantic search offers a fundamentally different approach. By using multimodal AI to understand the visual and audio content of a clip — rather than just its labels — it becomes possible to search video the way you'd describe it to another person: "show me a happy woman in a red dress" or "find a mom hugging a kid." No manual tagging required.

In this article, I'll walk through how this works — starting with the traditional metadata augmentation approach, then explaining how semantic search works under the hood, and finally showing a real-world example with a 90-minute TV show.

1. Metadata Augmentation



Let’s start with the traditional approach I call “metadata augmentation.” The idea is to run analytics — speech-to-text, OCR, scene description (some of which are AI-powered) — to extract additional metadata and associate it with each clip in a Media Asset Management (MAM) system.

All extracted metadata is associated with the clip and timestamped.

Think of it like assigning a human analyst to watch the clip and document:

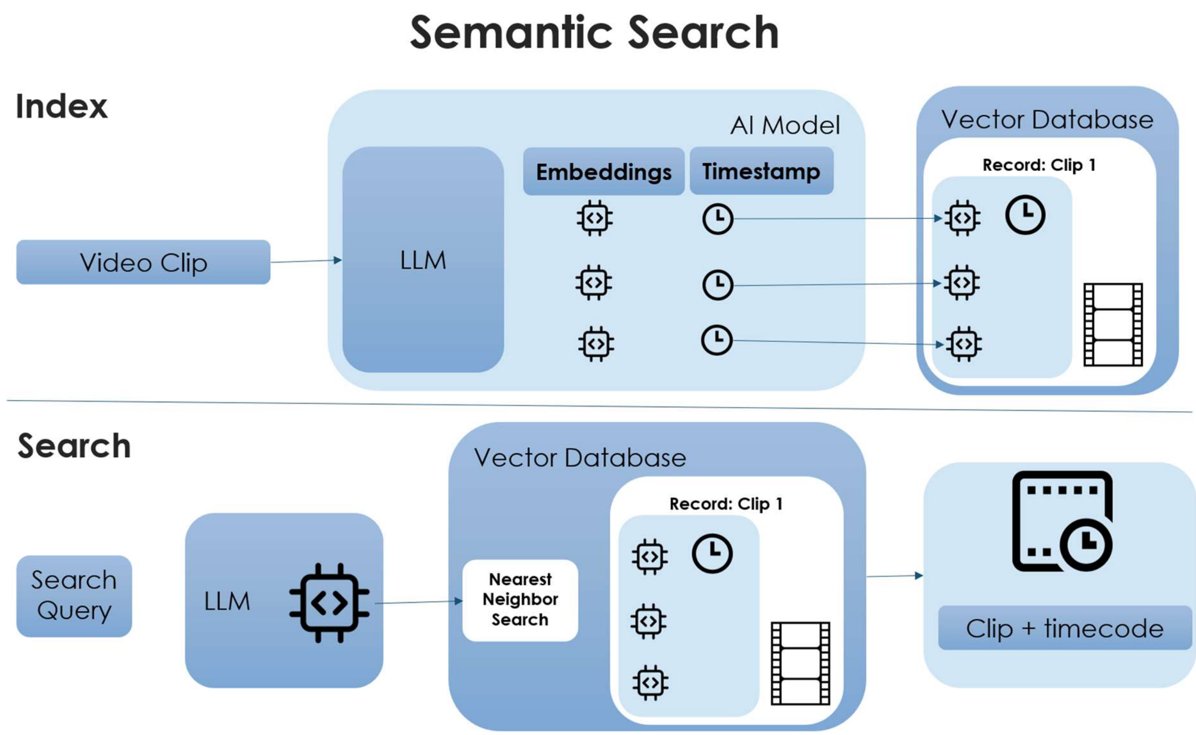
- produce a timestamped transcript of all spoken audio
- transcribe all on-screen text
- for each scene, write down what you see.

Everything produced is timed text — each piece of information is anchored to a specific moment in the clip.

I can then use text-based search to find the specific clips and timecodes where a search term appears. Unstructured search engines like Elasticsearch handle this kind of lookup very efficiently.

The key limitation: if something was never written down as text, it simply cannot be found. You are constrained by what you thought to extract up front.

2. Sematic Search



Semantic search works differently. Instead of pre-defining what to extract, I use a multimodal LLM to index the clip and generate embeddings. Embeddings are dense sequences of numbers — often thousands of values — that capture the meaning of a scene. The exact significance of each value is not human-readable; it emerges from the model's training process.

In practice, I generate an embedding for each scene in a clip, pair it with the scene's timestamp, and store it in a vector database — a specialized database designed for fast similarity matching across high-dimensional embeddings.

When searching, I type in the search phrase. This phrase is also converted to an embedding using the same model.

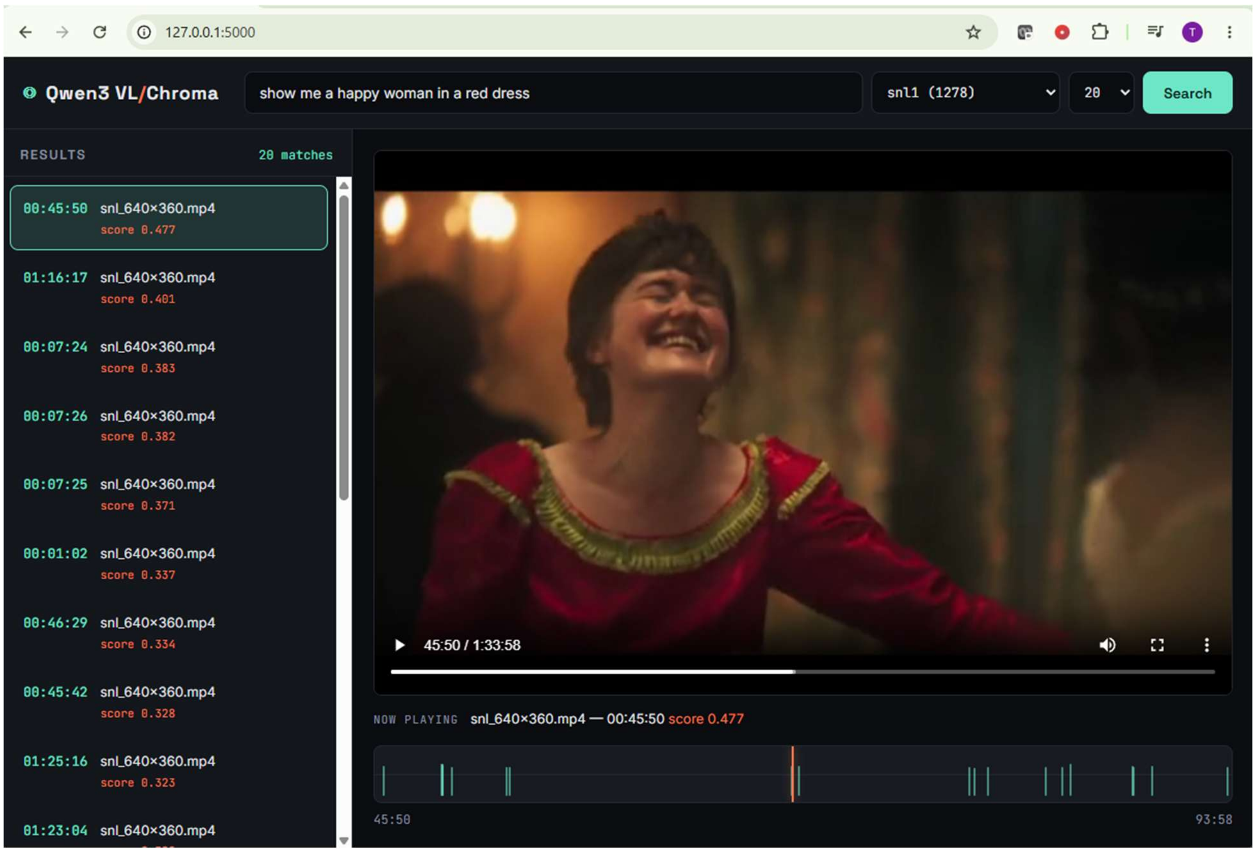
The text "frying pan" and an image of a frying pan will produce embeddings that are numerically close to each other — because the model has learned that both represent the same concept.

So when I search for "frying pan," I convert the query to an embedding and look in the vector database for scenes whose embeddings are numerically closest to it. The results are ranked by similarity and returned — no keyword matching required.

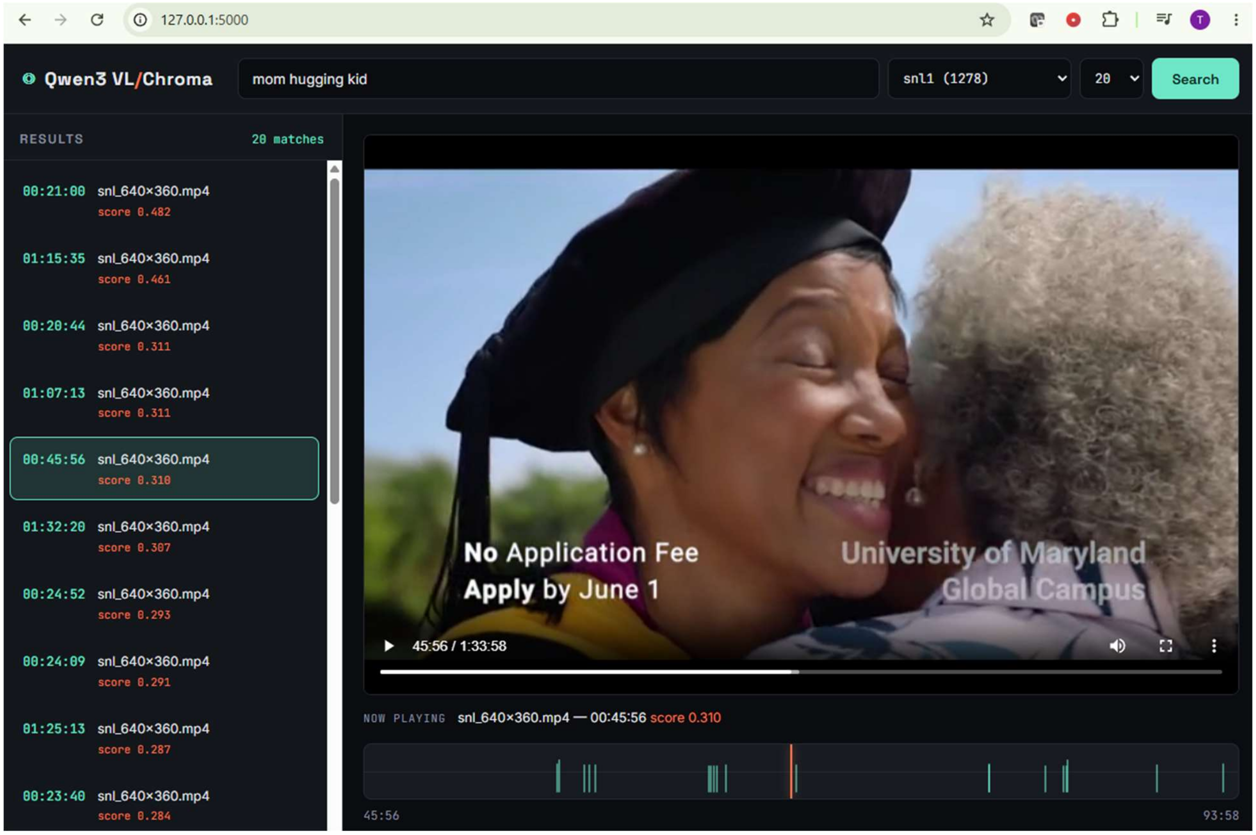
3. A Real World Example

A 90-minute TV show was indexed using Qwen 3-VL, a multimodal vision-language model. An embedding was generated for each scene and stored in ChromaDB, a vector database.

The following search was performed “show me a happy woman in a red dress”. The embedding of the query was then compared to all the embeddings for the scenes and the top 20 are displayed in the timeline.



Below is another example. The search term was “mom hugging kid”.



Semantic search is like asking someone with a perfect memory to watch a video — and then being able to ask them anything about it afterward.

They won’t always give you a perfect or exhaustive answer — but crucially, you don’t need to know what you’re looking for ahead of time. That flexibility is what makes semantic search so powerful for video content.

In the next articles in this series, I’ll explore practical applications and walk through a simple implementation you can run yourself. The code is already available on GitHub: tpflaum/qwen_video_search — including the search UI shown above. Give it a try and let me know what you think.

#SemanticSearch #VideoSearch #VideoAI #ComputerVision #ContentManagement #DigitalMedia