Subject: Re: Concepts Vectors
Posted by Wayne Parham on Tue, 03 Feb 2026 22:09:40 GMT
View Forum Message <> Reply to Message

My initial idea for concepts training is that it should be multi-modal. That was mostly because I imagined a baby learning concepts from things it experienced by sight, sound and tactile senses. And I still think those are very useful and important training modalities. Show a video of an item being put inside a container and then closing the lid. Lots of concepts there - inside/outside, open/closed, visible/hidden, etc.

But later, a thought occurred to me. Consider the plight of those born blind. They are trained concepts without images. They learn by tactile and audio senses. Once they learn language, they can learn concepts solely with language.

I am not saying that I think language models are all we need. I'm still very focused on concepts. But what I am saying is that concepts may be described using language. It may simplify the design.

So I have begun to do that. I am in the process of making a system that incorporates a list of concepts defined in JSON. When a user query is processed, it looks for matching concepts and provides scores for each concept. That can be used to create a list of parameters, which then forms a vector, much like models of other modalities.

Further, the concept model outputs not only vectors but also metadata. The metadata can be used to "explain" the concepts that have been found. The vectors help with analogical reasoning, finding concepts that are similar. And the metadata helps describe the concept, which can be used as retrieval augmented generation (RAG) content.

That information can help an LLM, keeping it more focused and "grounded." It won't depend solely on the similarities between words or passages. This is a shift towards concept-centric embeddings: vectors that represent concepts and their relationships, not just surface-form similarity.

Concepts are sort of like axioms or rules that are learned. They are much less fluid than relationships between words.

From embeddings to "concepts vectors"
A "concepts modeling" approach builds a system around concept classifiers and concept vectors:

A concept classifier detects whether some concept is present (e.g., inside vs. outside, equal vs. unequal, similar vs. dissimilar). Each classifier yields a multi-dimensional vector of parameters that characterizes the concept for the input—this becomes a concept vector. Instead of responding based on "this text is close to that text," the model responds based on "this input is close to these concepts."

Building large concept models: bottom-up
The proposed path is:

Establish primitive concepts first
Train reliable detectors for foundational distinctions like:

inside / outside
equal / unequal
similar / dissimilar

Compose higher-level concepts from primitives
Once primitives are stable, compose higher-level classifiers using:

outputs of primitive concept classifiers
associative classifiers
higher-order collective and correlation classifiers

Why this differs from standard embedding
Standard embeddings: often cluster by linguistic/visual similarity (distributional patterns).
Concept vectors: aim to cluster by conceptual similarity (shared underlying properties), enabling more analogical reasoning and more robust generalization across modalities.

A concrete example
If the input is an image of a cup in a box, a concept-centric system might output strong activations for:

inside (cup is inside box)
container (box is a container)
support (cup supported by box bottom)

Those concept activations (vectors) can then drive reasoning or retrieval—even if the text description uses different words.

Stay tuned!
