

From Keywords to Vectors: How AI Decodes Meaning

Published December 28, 2025 • Runtime: 18:30

<https://myweirdprompts.com/episode/ai-semantic-understanding-evolution/>

EPISODE SYNOPSIS

Ever wonder why you can search for "banana bread" with typos and get results, but your own computer fails to find a document if you miss one letter? In this episode of My Weird Prompts, Herman and Corn break down the shift from literal keyword matching to semantic understanding. They explore the fascinating history of "word math," from the linguistic theories of the 1950s to the revolutionary Transformer architecture that powers today's LLMs. You'll learn why local file search is still catching up, the trade-offs between precision and "vibes," and how "Approximate Nearest Neighbors" are changing the way we interact with data. Join us for a deep dive into the vector spaces that allow machines to finally understand what we mean, not just what we type.

DANIEL'S PROMPT

Daniel

"We've discussed semantic understanding as a foundation of large language models and its role in modern AI engineering—from context understanding to database design and proximity matching. While this numerical and statistical approach can seem abstract, the benefits are significant, such as the typo tolerance and flexibility that traditional keyword-based systems lack. Semantic understanding and technologies like vector matching and embeddings actually predate the current AI era. I'd love to learn more about the chronology of these developments. Also, given how useful semantic understanding is, why do we still see old-school, non-semantic search interfaces in modern systems, like computer file searches, instead of integrating fuzzy matching or semantic understanding?"

TRANSCRIPT

Corn

Hey everyone, welcome back to My Weird Prompts! I am Corn, and I am currently leaning very far back in my favorite chair, feeling every bit like the sloth that I am. And as always, I am joined by my much more energetic brother.

Herman

Herman Poppleberry, at your service! And yes, I have already finished two cups of coffee and read three research papers this morning, so I am ready to go. Our housemate Daniel sent us a really fascinating prompt today that gets into the literal gears and cogs of how modern artificial intelligence actually understands what we are saying.

Corn

Yeah, Daniel was asking about semantic understanding. It is one of those terms that sounds like it belongs in a philosophy textbook, but apparently, it is the reason why I can type a bunch of typos into a search bar and it still knows I am looking for a recipe for banana bread.

Herman

It is exactly that, Corn! And it is even more impressive when you realize that to a computer, language is just a series of numbers. Today we are going to dive into how we got here, the history of this technology, and the big question Daniel had: if this tech is so good, why does searching for a file on my own computer still feel like I am using a machine from nineteen ninety-five?

Corn

That is the part that really gets me. I can ask an artificial intelligence to write a poem about a lonely toaster and it gets the vibe perfectly, but if I search for a document named report on my laptop and I forget one letter, it just looks at me like I am speaking a dead language. It is frustrating!

Herman

It really is. But before we get to the frustration, let us talk about the magic. Or, as the engineers call it, the vector space. See, traditional search, what we call keyword matching, is very literal. If you search for dog, the computer looks for the letters d, o, and g in that exact order. If you have a file called canine, the computer says, sorry, never heard of it.

Corn

Right, because it does not know what a dog is. It just knows the shape of the word.

Herman

Exactly. Semantic understanding is the shift from looking at the shape of the word to looking at the meaning of the word. And the way we do that is by turning words into lists of numbers called embeddings. Imagine every word has a set of coordinates in a massive, multi-dimensional map. In this map, dog and canine are sitting right next to each other. Even cat and puppy are nearby. But a word like refrigerator is miles away in another zip code.

Corn

So when I search, the computer is not just looking for the word, it is looking for the neighborhood?

Herman

That is a perfect way to put it. It is called proximity matching. The computer looks at your search term, finds its coordinates, and then looks at all the other words or sentences in its database to see which ones are physically closest in that mathematical space.

Corn

That is wild. But Daniel mentioned that this stuff actually predates the current artificial intelligence craze. I always assumed this was some brand new invention from the last two years. How far back does this actually go, Herman? You are the history buff.

Herman

Oh, it goes back way further than people realize. The fundamental idea, which is called the Distributional Hypothesis, was actually popularized in the nineteen fifties! There was a linguist named John Rupert Firth who famously said, you shall know a word by the company it keeps.

Corn

Wait, the nineteen fifties? We barely had computers that could fit in a room back then, let alone understand the vibes of a sentence.

Herman

True, but the theory was there. The idea was that words that appear in similar contexts tend to have similar meanings. If you see the word bark near the word tree, it means one thing. If you see it near the word tail, it means another. In the late nineteen eighties and early nineteen nineties, we got something called Latent Semantic Analysis. This was the first real attempt to use math to uncover these hidden relationships between words in large bodies of text.

Corn

So what changed? Why did it take until now for us to feel like we are actually talking to the machines?

Herman

Computational power and better algorithms. The real explosion happened around twenty thirteen. That was when a team at Google, led by Tomas Mikolov, released something called Word to Vec. That was the moment we figured out how to create these word embeddings very efficiently on a massive scale. It allowed us to do things like word math.

Corn

Word math? Like, addition and subtraction?

Herman

Literally! You could take the vector for the word king, subtract the vector for the word man, add the vector for the word woman, and the resulting mathematical coordinate would be almost exactly where the word queen is located.

Corn

Okay, now you are just messing with me. You are telling me king minus man plus woman equals queen in computer math?

Herman

It is true! It proved that these numbers were not just random; they were actually capturing human concepts and relationships. Then, in twenty seventeen, we got the Transformer architecture, which is what the T in G P T stands for. That allowed the models to look at the entire context of a sentence at once, rather than just one word at a time. It made the semantic understanding much more nuanced and powerful.

Corn

It is amazing how much work went into making things feel natural for us. It feels like we are just talking, but underneath it is all this heavy lifting with millions of coordinates.

Herman

It really is a monumental engineering feat. But as Daniel pointed out, we still have these weird gaps. We have this incredible technology, yet we still deal with systems that are incredibly rigid.

Corn

Yeah, let us get into that. But first, I think I hear our favorite salesman lurking in the hallway. Let us take a quick break. Larry: Are you tired of your thoughts being disorganized? Do you wish you could browse your own memories with the click of a button? Introducing the Neuro-Linker Nine Thousand! This revolutionary headband uses patented static-cling technology to reorganize your brain's electromagnetic field. Simply strap it on before bed, and by morning, your dreams will be sorted alphabetically! No more searching for that childhood memory of a blue bicycle—it is right there between "Bipolar Disorder" and "Blueberry Muffins." Side effects may include temporary loss of color vision, a sudden craving for copper wire, and a permanent humming sound in your left ear. The Neuro-Linker Nine Thousand: because your brain is a mess and we know it. BUY NOW!

Herman

...I really hope no one actually buys anything from Larry. Sorting dreams alphabetically sounds like a literal nightmare.

Corn

Honestly, I would settle for just sorting my sock drawer. Anyway, back to the topic. We were talking about why my computer's file search is still so bad compared to these artificial intelligence models. If we have had this semantic tech since twenty thirteen, why is my laptop still acting like it is nineteen ninety-five?

Herman

That is the million-dollar question, and there are actually a few really good reasons for it. The first one is pure computational cost. To do a semantic search, you have to turn every single file on your computer into a vector. You have to run it through a model to get those coordinates.

Corn

That sounds like a lot of work for a little laptop.

Herman

It is! Think about how many files are on your computer. Thousands, maybe tens of thousands. Every time you edit a document, the computer would have to re-calculate that vector. If you did that for every file in real-time, your laptop battery would die in twenty minutes and your fan would sound like a jet engine taking off.

Corn

Okay, that makes sense. But couldn't they just do it once a day or something?

Herman

They could, and some systems are starting to do that. But there is a second issue: reliability and precision. Sometimes, you do not want a fuzzy match. If I am looking for a file specifically named Budget Final Version Two, I do not want the computer to show me a bunch of files about money and accounting. I want that exact file.

Corn

Right. If I am looking for my taxes, I do not want the computer to give me a poem about the IRS because the vibes are similar.

Herman

Exactly. Keyword search is what we call deterministic. It is either a match or it is not. It is very reliable for finding specific things. Semantic search is probabilistic. It is giving you its best guess based on similarity. For a file system, being wrong is often worse than being slow.

Corn

That is a fair point. I guess I would rather have to type the name perfectly than have to dig through fifty similar files that are not what I want. But surely there is a middle ground?

Herman

There is! It is called hybrid search. A lot of modern databases, like the ones Daniel mentioned in his prompt, are now using both. They use keywords to find exact matches and then use semantic understanding to find things that are related or to handle typos.

Corn

So why isn't my operating system doing that yet?

Herman

It is starting to happen! In late twenty twenty-four and throughout twenty twenty-five, we have seen major updates to Windows and Mac OS that are integrating these features. They are using smaller, more efficient models that can run locally on your computer's chip without killing the battery. But it is a slow rollout because of the third big issue: privacy.

Corn

Privacy? How does searching for a file affect my privacy?

Herman

To create these semantic embeddings, the system has to read the content of your files. If that processing happens in the cloud, you are essentially sending all your private documents to a server to be indexed. People are understandably very nervous about that.

Corn

Oh, yeah. I definitely do not want my personal journals or bank statements being sent off to some server just so I can find them easier.

Herman

Precisely. So the challenge for engineers right now is making these models small enough to run entirely on your device, so your data never leaves your sight. We are just now getting to the point where the hardware in our laptops and phones is powerful enough to do that efficiently.

Corn

So we are basically living in the transition period. We have the fancy tech in the cloud, and we are waiting for the local versions to catch up.

Herman

That is exactly it. It is also worth noting that keyword search is incredibly fast. Like, incredibly fast. You can index millions of words and search them in milliseconds using something called an inverted index. Semantic search, where you have to compare your search vector against every other vector in the database, is much slower mathematically. Engineers have to use tricks like Approximate Nearest Neighbor search to make it feel fast.

Corn

Approximate Nearest Neighbor? That sounds like what I do when I am looking for my keys. I don't look everywhere, I just look in the places that are close to where I usually leave them.

Herman

That is actually a great analogy, Corn! Instead of checking every single point in the multi-dimensional map, the computer groups similar points into clusters. It finds the cluster that is closest to your search and only looks inside that group. It is a bit like looking for a book in a library. You don't look at every spine; you go to the history section and then look for the specific shelf.

Corn

I like that. It makes the whole thing feel a bit more human. So, what are the big takeaways here for someone like Daniel, or for our listeners who are using these tools every day?

Herman

Well, the first takeaway is that we should appreciate the hybrid nature of search. We are moving toward a world where the computer understands our intent, not just our input. If you are a developer, understanding how to combine traditional keyword search with these new vector databases is the key to building tools that feel intuitive.

Corn

And for regular people like me, I guess the takeaway is to have a little patience with our old-school folders and files. The tech is coming, it just needs to be done in a way that does not melt our computers or spy on our secrets.

Herman

Exactly. And honestly, there is a lot of value in being precise with our language. Even as semantic understanding gets better, the clearer we are with our prompts and our file names, the better the results will be. The machine is trying to meet us halfway.

Corn

I can do that. I can meet a machine halfway. As long as it does not expect me to do any of that multi-dimensional math you were talking about. My brain only works in two dimensions: napping and eating.

Herman

I think that is a very healthy way to live, brother.

Corn

Well, this has been a really eye-opening conversation. I feel like I understand why my search bar is the way it is now. It is not just being stubborn; it is actually dealing with some pretty heavy math and privacy concerns.

Herman

It really is. And as we move into twenty twenty-six, I think we are going to see these semantic features become so common that we will forget how frustrating it used to be. We will just expect our computers to know what we mean, even if we are a bit fuzzy on the details.

Corn

I am looking forward to that day. Imagine a world where I can just tell my computer, find that thing I wrote about the squirrels, and it actually finds it!

Herman

We are almost there, Corn. We are almost there.

Corn

Alright, I think that is a wrap for today. Thank you so much for the deep dive, Herman Poppleberry. And thanks to Daniel for sending in such a great prompt. It is always fun to peek under the hood of the technology we use every day.

Herman

My pleasure! It is always a joy to explain the nerdier side of things to you.

Corn

This has been My Weird Prompts! If you enjoyed the show, you can find us on Spotify and at our website, myweirdprompts.com. We have an RSS feed for subscribers and a contact form if you want to send us your own weird prompts. We would love to hear from you!

Herman

Yes, please send us your questions! No topic is too strange or too technical for us to tackle.

Corn

Take care, everyone. We will see you in the next episode!

Herman

Bye everyone! Stay curious!

