interweb

Generated on 15/11/2026 00:16

KNOWLEDGE BASE · PUBLIC DOCUMENT

Embeddings and semantic search: finding things by meaning in your own content

How AI Works

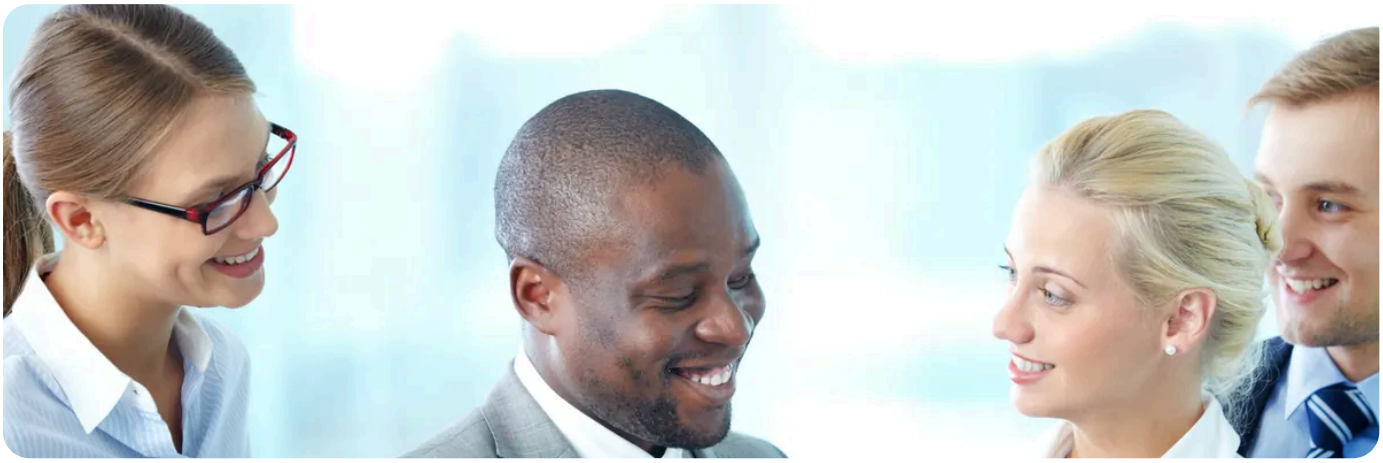
CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-1580	v2026.10.10	Angola · English

Interweb Angola
RLX Angola, Lda. · Group RLX Holdings
Tax ID (NIF) 500 260 5724 · Commercial registration no. 26464-25/250724

https://www.meu.interweb.ao/knowledgebase/1580

Embeddings and semantic search: finding things by meaning in your own content

How AI Works Created on **October 10, 2026** 3 min read



An **embedding** is a list of numbers that represents the meaning of a text. Two texts that say similar things get similar lists, even if they share no words. That is what makes **semantic search** possible: someone who types “I cannot get into my account” finds the article “recovering access”, which does not contain the word “get into”. It is the piece underneath assistants that answer from documents (RAG), but it also works on its own.

Searching by words or by meaning

	By words (the normal kind)	By meaning (embeddings)
How it finds	Looks for the written words	Compares the meaning of the question with that of the texts
Good for	Codes, references, exact names, invoice numbers	Natural-language questions, synonyms, vague descriptions
Fails on	People who use other words for the same thing	Exact codes and names: it may return “similar” things instead of the one asked for
What it needs	Nothing special (WordPress search, the database)	An embedding model and somewhere to store the numbers
Cost and privacy	Stays on your server	The text goes to a model, a provider’s or your own

The best result usually comes from **combining the two**: word search for what is exact and meaning search for the rest.

What it is for on a small site

1. **Site search that understands questions.** The visitor types the way they speak and finds the right page.
2. **“Similar articles” at the end of each article,** without choosing them by hand.
3. **Routing support requests:** compare a message with the frequently asked questions and suggest the closest answer.
4. **Finding duplicates** or repeated texts on a large site.

What you need to build

1. **Cut the content into pieces** that each deal with one thing (a section, a question).
2. **Generate one embedding per piece,** with a model made for that. Answering models and embedding models are different; each has its own pricing page.
3. **Store the numbers** in a database built to search by closeness, or in a file if the content is small. See **installing PostgreSQL on a VPS**.
4. **At search time,** turn the question into an embedding with the *same* model and fetch the closest ones.
5. **Redo the embeddings** when the text changes. New text with old numbers gives wrong results.

Quality depends a lot on **how you cut** the text. Pieces that are too small lose context; pieces that are too big mix subjects. Try real questions and look at what the search returns before connecting it to the site.

Mixing models ruins everything.
One model’s numbers cannot be compared with another’s. If you change model, you have to redo all the embeddings.

For a site of a few dozen pages, it may not be worth building anything:
a good frequently asked questions page
and a well-tuned normal search solve most of it. See **RAG explained**.

RECOMMENDED

Want to build a search like this on a server of your own? Have a look at the VPS plans.

See VPS plans →

SEE ALSO

- AI that answers from your own documents: RAG explained →
- Tokens explained: how AI model APIs are counted →
- Installing PostgreSQL on a VPS →

interweb

Interweb Angola

RLX Angola, Lda. · Group RLX Holdings

TAX ID (NIF)	500 260 5724
COMMERCIAL REGISTRATION NO.	26464-25/250724
REGISTERED OFFICE	Rua Direita do Patriota, Bairro Dangereux Belas, Luanda, Angola
CONTACT	info@interweb.ao · +244 942 011 191 · www.interweb.ao
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-AO-KB-1580 · v2026.10.10 · Generated on 15/11/2026 00:16