**Absolutely! Here's a ranked comparison of the best local embedding models based on three key factors:**

**✅ Ranking Criteria**

* **🔢 Dimensions: Higher = richer semantic information, but slower.**
* **🎯 Accuracy: Based on** [**MTEB**](https://huggingface.co/spaces/mteb/leaderboard) **benchmarks, performance in RAG, retrieval, clustering.**
* **⚡ Latency: Speed of embedding generation (on local CPU/GPU). Lower = faster.**

**🥇 Top Embedding Models - Ranked Table**

| **Rank** | **Model Name** | **🔢 Dimensions** | **🎯 Accuracy (MTEB / practical)** | **⚡ Latency (CPU)** | **⚡ Latency (GPU)** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **Alibaba/gte-large** | **1024** | **⭐⭐⭐⭐⭐ (top-tier)** | **❌ (slow)** | **✅ (fast)** | **Best accuracy; ideal for GPU-heavy RAG** |
| **2** | **nomic-embed-text** | **768** | **⭐⭐⭐⭐½** | **✅ (fast)** | **✅ (fast)** | **Fast + accurate, great balance** |
| **3** | **BAAI/bge-base-en-v1.5** | **768** | **⭐⭐⭐⭐½** | **✅ (fast)** | **✅ (faster)** | **Extremely accurate for RAG** |
| **4** | **intfloat/e5-large** | **1024** | **⭐⭐⭐⭐½** | **❌ (slow)** | **✅ (fast)** | **Needs instruction formatting** |
| **5** | **thenlper/gte-base** | **768** | **⭐⭐⭐⭐** | **✅ (good)** | **✅ (fast)** | **General purpose + fast** |
| **6** | **intfloat/e5-base** | **768** | **⭐⭐⭐⭐** | **✅ (good)** | **✅** | **Best for QA if format used** |
| **7** | **Instructor-Large** | **768** | **⭐⭐⭐⭐** | **❌ (moderate)** | **✅** | **Needs instruction prefix** |
| **8** | **all-roberta-large-v1** | **1024** | **⭐⭐⭐⭐** | **❌ (slow)** | **❌ (slow)** | **Accurate but very slow** |
| **9** | **all-MiniLM-L6-v2** | **384** | **⭐⭐½** | **✅✅✅ (superfast)** | **✅✅✅** | **Great for fast CPU apps** |
| **10** | **distiluse-base-multilingual** | **512** | **⭐⭐½** | **✅✅** | **✅✅** | **Lightweight & multilingual** |
| **11** | **cohere multilingual (ONNX)** | **768** | **⭐⭐⭐½** | **✅ (if ONNX)** | **✅** | **Good for multilingual RAG** |

**🧠 Detailed Interpretation:**

**🥇 Best Overall (Accuracy + Latency):**

* **✅ nomic-embed-text**
* **✅ BAAI/bge-base-en-v1.5**

**These give strong accuracy, work very fast, and easy to run locally even on CPU.**

**🥈 Best for Enterprise & High Accuracy:**

* **✅ gte-large**
* **✅ e5-large**
* **✅ all-roberta-large**

**These are accurate, multi-purpose, and suitable for GPU setups, but can be slow on CPU.**

**⚡ Best for Speed (Low Latency):**

* **✅ MiniLM-L6-v2 – ultra fast (good for chatbots, CPU RAG)**
* **✅ distiluse-base-multilingual – multilingual + low latency**

**🌍 Best for Multilingual Use:**

* **✅ gte-base / distiluse-base-multilingual**
* **✅ cohere multilingual (via ONNX/Transformers)**

**✅ TL;DR: Recommendations by Goal**

| **Goal** | **Recommended Models** |
| --- | --- |
| **Best All-Round (Fast + Good)** | **nomic-embed-text, bge-base-en-v1.5** |
| **High Accuracy (GPU setups)** | **gte-large, e5-large** |
| **Fastest on CPU** | **MiniLM-L6-v2, distiluse-multilingual** |
| **Multilingual Embedding** | **cohere multilingual, gte-base** |
| **Instruction-aware Embeddings** | **Instructor-XL, e5-base** |

**Would you like a visual graph comparing them or recommendation based on your machine specs (CPU/GPU available)?**

Final List

| **🏅Rank** | **🧠 Model Name** | **🔢 Dimensions** | **🎯 Accuracy (MTEB / Real)** | **⚡ CPU Latency** | **⚡ GPU Latency** | **🤝 LLM Compatibility** | **📌 Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 🥇 1 | thenlper/gte-large | 1024 | ⭐⭐⭐⭐⭐ (Top-tier) | ❌ (slow) | ✅ (fast) | ✅✅✅ | **Best accuracy**, especially great for RAG with LLMs like Llama2/Mistral |
| 🥈 2 | BAAI/bge-base-en-v1.5 | 768 | ⭐⭐⭐⭐½ | ✅ (fast) | ✅✅ (very fast) | ✅✅✅ | **Highly accurate**, optimized for dense retrieval |
| 🥉 3 | stella-embed-en-v2 | 1024 | ⭐⭐⭐⭐½ | ✅✅ (very fast) | ✅✅ (very fast) | ✅✅✅ | Best balance of **speed + 1024-dim**, great for local CPU/GPU setups |
| 4 | nomic-embed-text-v1 | 768 | ⭐⭐⭐⭐½ | ✅ (fast) | ✅✅ | ✅✅ | General-purpose, **fast**, ideal for wide use in apps or dashboards |
| 5 | jinaai/jina-embedding-v3-base | 1024 | ⭐⭐⭐⭐½ | ✅ | ✅ | ✅ (non-commercial) | Long-context support, **top MTEB**, use in **non-commercial** RAG only |
| 6 | KaLM-Embedding | 64–896 | ⭐⭐⭐⭐ | ✅✅✅ (superfast) | ✅✅✅ | ✅✅ | **Dimension-scalable**, good for fast + memory-constrained environments |
| 7 | Alibaba/gte-large | 1024 | ⭐⭐⭐⭐⭐ | ❌ (slow) | ✅✅ | ✅✅✅ | **Alibaba's best for RAG**, needs GPU, highest semantic quality |
| 8 | intfloat/e5-large | 1024 | ⭐⭐⭐⭐½ | ❌ | ✅ | ✅✅✅ | Needs input formatting (e.g., query: ...) |
| 9 | thenlper/gte-base | 768 | ⭐⭐⭐⭐ | ✅ | ✅✅ | ✅✅ | **Smaller + faster GTE**, good for mid-range local setups |
| 10 | intfloat/e5-base | 768 | ⭐⭐⭐⭐ | ✅ | ✅ | ✅✅ | Fast, accurate for QA if used with prefix formatting |
| 11 | Instructor-Large | 768 | ⭐⭐⭐⭐ | ❌ | ✅ | ✅✅ | Requires instructions in input ("Represent this sentence for...") |
| 12 | all-roberta-large-v1 | 1024 | ⭐⭐⭐⭐ | ❌ (very slow) | ❌ (slow) | ✅ | Accurate but **too slow** for real-time use |
| 13 | all-MiniLM-L6-v2 | 384 | ⭐⭐½ | ✅✅✅ | ✅✅✅ | ✅✅ | **Fastest CPU**, great for speed-critical tasks |
| 14 | distiluse-base-multilingual | 512 | ⭐⭐½ | ✅✅ | ✅✅ | ✅✅ | Lightweight, **multilingual**, ideal for global apps |
| 15 | cohere multilingual (ONNX) | 768 | ⭐⭐⭐½ | ✅ | ✅ | ✅✅ | Solid multilingual + ONNX support |

Latest

| **Model Name** | **Dimensions** | **Released** | **License** | **Local Compatible** | **Notes** |
| --- | --- | --- | --- | --- | --- |
| nomic-ai/nomic-embed-text-v1 | 768 | 2024–2025 | Apache 2.0 | ✅ | High-quality general-purpose embeddings |
| BAAI/bge-base-en-v1.5 | 768 | 2024 | Apache 2.0 | ✅ | Top for RAG & semantic search |
| thenlper/gte-large | 1024 | 2024 | Apache 2.0 | ✅ | High MTEB score; large, needs GPU |
| jinaai/jina-embedding-v3-base | 1024 | 2024 | CC-BY-NC-4.0 | ✅ (non-commercial) | Long context, supports downscaling |
| stella-embed-en-v2 | 1024 | 2024 | MIT | ✅ | Lightweight, powerful |
| KaLM-Embedding | 64–896 | 2025 | MIT | ✅ | Matryoshka downscaling |