RAG Embeddings Report

RAG embedding and retrieval is an essential technology as AI becomes more and more prevalent. RAG models allow semantic models to function while only specifically referencing information explicitly given to them. This is useful in preventing model hallucination or in training highly specialized models. This report examines on a base level the effectiveness of two different models available to coders today.

The Models:

OpenAI's text-embedding-ada-002 model, originally released in late 2022, was a significant advancement in embedding technology at the time of its debut. While it has since been surpassed by newer, more advanced models in OpenAI's lineup, it remains a strong performer in many natural language processing tasks. Despite not being the latest offering, text-embedding-ada-002 continues to deliver competitive results when compared to current top-tier embedding models. One of its key advantages is its efficiency—it strikes a solid balance between performance and resource usage, making it well-suited for applications that require quality embeddings without the computational overhead of larger, more resource-intensive models. As a result, it is still widely used in production environments where cost and speed are important factors.

SBERT's all-MiniLM-L6-v2 is a widely adopted open-source embedding model that offers a strong balance between performance and efficiency. Developed as part of the Sentence-BERT (SBERT) framework, it is designed to generate high-quality sentence embeddings suitable for tasks like semantic search, clustering, and textual similarity. One of the standout features of all-MiniLM-L6-v2 is its accessibility—it is available under an open license and can be easily integrated into Python-based projects using popular libraries such as sentence-transformers With only six transformer layers, the model is lightweight and fast, making it an excellent choice

for real-time applications or resource-constrained environments, without significantly sacrificing embedding quality. Its open nature and robust community support make it a go-to solution for developers seeking a free, performant embedding model.

Model Comparison:

OpenAI's text-embedding-ada-002 and SBERT's all-MiniLM-L6-v2 are both popular embedding models, but they serve different needs and come with distinct trade-offs.

The text-embedding-ada-002 model offers high-quality embeddings and performs well across a wide range of retrieval and similarity tasks. It has strong semantic understanding and scales easily through OpenAI's cloud infrastructure. Since it is accessed via an API, there is no setup required, making it easy to use in any environment with internet access. However, it is not open source, which means users are dependent on OpenAI's infrastructure and pricing. It also cannot be used in offline environments and offers limited control, as users cannot fine-tune or inspect the model's internals. Additionally, reliance on an external API introduces potential latency and usage costs.

In contrast, SBERT's all-MiniLM-L6-v2 is open source and can be freely modified or deployed. It is lightweight and fast, making it suitable for use in low-resource environments or real-time applications. It can run locally without an internet connection and is easy to use through the Python sentence-transformers library. However, it may offer slightly lower embedding quality compared to larger proprietary models like ada-002. Because it runs on local infrastructure, it may not scale as easily for very large workloads, and users are responsible for maintaining and optimizing the system themselves. It also lacks official commercial support, relying instead on community resources.

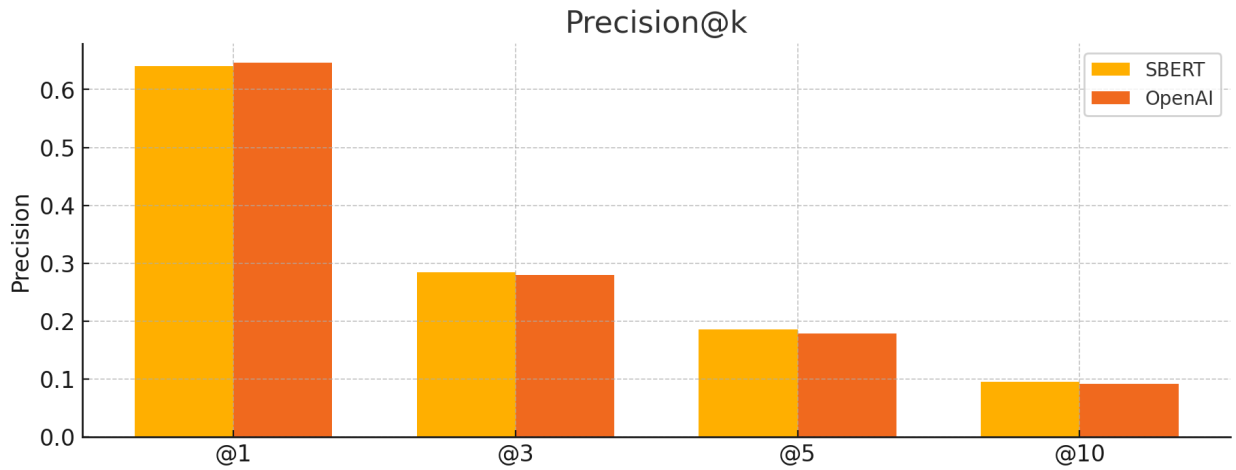
Overall, ada-002 is better suited for users who prioritize accuracy and ease of use via a managed service, while all-MiniLM-L6-v2 is ideal for developers who need a fast, free, and flexible solution that can be deployed locally.

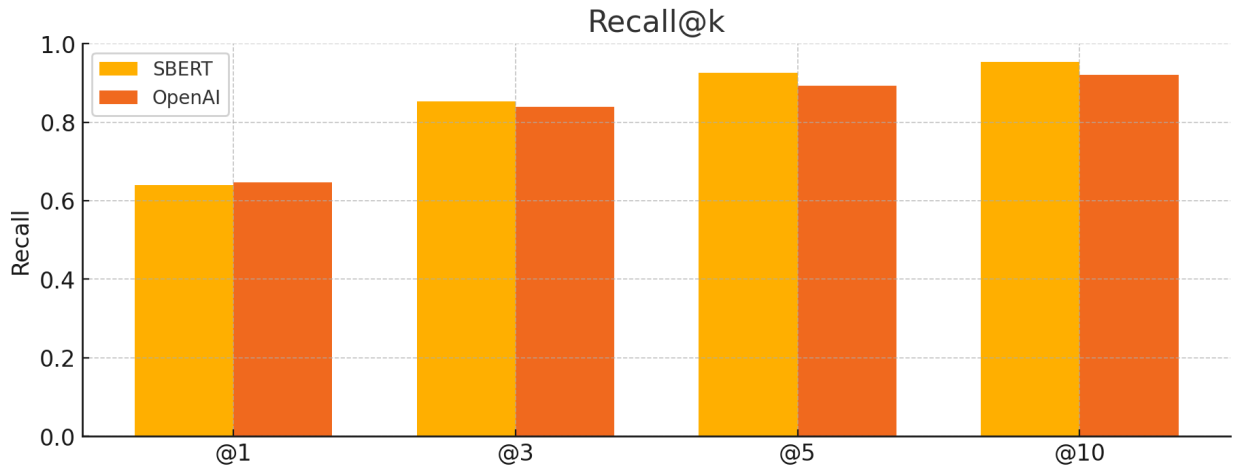
The Tests:

To test the models I used two different scripts, first a very basic embedding using only five documents tracking precision and mean reciprocal rank of the models performance. The purpose of this test was to see how the models perform in ideal conditions proving they can perform as advertised. The initial test resulted in perfect precision and MRR for both models showing that both models have a strong foundation.

Since both models performed so well in the first test I extended the test to a larger dataset, specifically the Stanford Question and Answers Dataset (SQuAD) which contains 100,000 questions and answers. This dataset is ideal for testing RAG as the ground truth of each document is very easy to establish. The dataset demonstrates the ability of the embedding model to process semantic information and return valuable relevant information. Note that I only used SQuAD 1.1 not the full 2.0 dataset which contains unanswerable questions.

To evaluate each model I measured the precision and recall of each model at k documents retrieved up to ten, and the mean reciprocal rank overall. Both models performed similarly but in most categories SBERT edged out OpenAI's model surprisingly. Specifically OpenAI had a higher precision and recall on the first document retrieved and lower in all categories in the following documents retrieved.





Conclusion:

SBERT outperformed OpenAI's embedding model in nearly all evaluated metrics, including precision and recall at various top-k levels, although the margin was not always large. This slight edge in performance, combined with SBERT's open-source nature, makes it an appealing choice for developers and researchers who value transparency, flexibility, and control. Being open source means SBERT can be fine-tuned, modified, or integrated into local workflows without dependence on external APIs or cloud services, which is particularly advantageous in privacy-sensitive or offline environments.

On the other hand, OpenAI's text-embedding-ada-002 model holds its own in terms of quality and offers a significant advantage in terms of ecosystem integration. Because it is part of OpenAI's managed platform, it can be seamlessly combined with other OpenAI services such as GPT models, file search, or assistant APIs. This tight integration may simplify development for applications already built on OpenAI's infrastructure and can reduce operational complexity for teams that prefer managed services over maintaining their own model infrastructure.

Overall, while SBERT provides slightly better performance and greater flexibility, OpenAI's model remains highly competitive and may be preferable in environments where ease of integration, scalability, or access to the broader OpenAI suite is a priority. In practical terms, both models are capable and effective for most semantic search or embedding use cases, and the final choice may depend more on project constraints and infrastructure preferences than on raw performance alone.