Approach and Methodology: SHL Assessment Recommendation Engine

Chandra Shekhar

1. Problem Understanding

The objective of this project is to design an intelligent recommendation system that maps a natural language hiring or assessment requirement to the most relevant SHL individual test solutions. The system must operate on SHL's assessment catalogue, support semantic understanding of queries, and expose results through an accessible interface. In addition to building the recommendation engine, the assignment emphasizes transparent evaluation using labeled data and clear system accessibility through APIs.

The key challenges in this task include handling unstructured natural language input, ensuring relevance beyond keyword matching, dealing with heterogeneous assessment descriptions, and evaluating the quality of recommendations in a measurable manner.

2. Data Collection and Preparation

The foundation of the system is the SHL product catalogue, which was crawled directly from the SHL website. Only individual test solutions were retained, while pre-packaged job solutions were explicitly excluded in line with the assignment requirements. During crawling, care was taken to avoid duplicate links, pagination artifacts, and non-assessment pages.

Each assessment page was processed to extract its textual description, which serves as the primary source of semantic information. The extracted content was cleaned to remove navigation elements, repetitive footer text, and unrelated metadata. The final dataset consists of structured assessment records, each associated with a unique URL and descriptive text.

Although the assignment specifies a target of at least 377 individual assessments, the publicly accessible catalogue endpoints limited discoverability of some solutions. The final curated dataset contains 264 unique individual test solutions, which are transparently used throughout the system.

3. Semantic Representation and Retrieval

To support meaningful recommendations, the system relies on semantic similarity rather than keyword-based matching. Each assessment description is transformed into a dense vector representation using a pre-trained sentence embedding model (all-MiniLM-L6-v2). This model was selected due to its balance between performance and computational efficiency, as well as its proven effectiveness in semantic retrieval tasks.

All assessment embeddings are indexed using FAISS with an inner-product similarity metric. Since embeddings are normalized, this configuration effectively computes cosine similarity. This approach enables fast and accurate retrieval of relevant assessments even when the user query does not share explicit keywords with the assessment text.

At query time, the user's input is embedded using the same model and compared against the indexed assessment vectors. The system retrieves the top-K most similar assessments, ranked by semantic relevance.

4. Evaluation Methodology

To validate the effectiveness of the recommendation engine, the provided labeled training dataset was used for evaluation. This dataset contains natural language queries paired with human-annotated relevant assessment URLs. Since the labeled data includes multiple URL formats, a normalization step was introduced to extract consistent assessment identifiers before comparison.

Evaluation focuses on Recall@K metrics, specifically Recall@5 and Recall@10. These metrics measure the proportion of relevant assessments retrieved within the top-K recommendations. Recall was chosen as the primary metric because the task emphasizes surfacing relevant options rather than strict ranking precision.

The evaluation process demonstrates that the system consistently retrieves a meaningful subset of the human-labeled relevant assessments, indicating that the semantic retrieval approach aligns well with human judgment.

5. System Interface and Deployment

To ensure usability and compliance with the assignment specifications, the recommendation engine is exposed through a RESTful API built using FastAPI. Two endpoints are provided: a health check endpoint for service validation and a recommendation endpoint that accepts a query and returns structured assessment information.

In addition to the API, a lightweight Streamlit-based web frontend was developed. This interface allows users to enter natural language requirements and view recommended assessments interactively. The frontend communicates directly with the FastAPI backend and serves as a simple demonstration tool for evaluators.

6. Design Decisions and Limitations

Several design decisions were guided by practicality and clarity. A retrieval-based architecture was chosen over generative approaches to ensure deterministic behavior, transparent evaluation, and reproducibility. Lightweight models and indexing methods were preferred to keep the system efficient and easy to run locally.

The primary limitation of the system is the incomplete coverage of the full SHL catalogue due to constraints in public endpoint visibility. This limitation is acknowledged, and future work could explore additional catalogue entry points or structured data sources to expand coverage.

7. Conclusion

This project delivers a complete, evaluated, and accessible assessment recommendation system built on SHL's product catalogue. By combining semantic embeddings, efficient vector search, objective evaluation, and clean interfaces, the system meets the core requirements of the assignment while maintaining transparency in design choices and limitations.