Here's a structured explanation of the data processing pipeline:

1. Data Acquisition & Ingestion:

- **Web Scraping Architecture:**
    - Paginated catalog scraping with Firecrawl/BeautifulSoup
    - Two-tiered collection: Product listings + individual product details
    - Robust error handling with retries and incremental saves
    - Rate limiting (10s delay between requests)
- **Key Technologies:**
    - FirecrawlApp for managed scraping
    - BeautifulSoup for HTML parsing
    - Pandas for incremental CSV storage

2. Data Transformation & Feature Engineering:

- **LangChain Worker Pipeline:**
    - Specialized AI workers for feature extraction:
        1. **Test Type Classification** (Letter-code taxonomy)
        2. **Skill Extraction** (Hard/soft skills detection)
        3. **Job Level Identification** (11-tier categorization)
        4. **Language Detection** (50+ language support)
        5. **Time Limit Parsing** (Duration pattern matching)
        6. **Testing Type Detection** (Remote/adaptive flags)
- **Normalization Steps:**
    - Skill term standardization (e.g., "JS" → "JavaScript")
    - Job level mapping to enterprise hierarchy
    - Duration conversion to minutes
    - Language name normalization

3. Embedding Generation & Indexing:

- **Semantic Encoding:**
    - paraphrase-MiniLM-L6-v2 model for dense embeddings
    - Combined "Skills_JobLevel" text feature:

python

Copy

"Python, cloud architecture, problem-solving , Mid-Professional"

- **FAISS Optimization:**
    - L2 normalization for cosine similarity
    - Flat index for exact nearest neighbors
    - 384-dimensional embedding space

4. Persistence & Deployment Prep:

- **Artifact Storage:**
    - metadata.parquet: Processed records with features
    - precomputed_faiss_index.bin: Binary index file
    - Columnar storage for efficient retrieval
- **Performance Considerations:**
    - Parquet format for column-based access
    - Batch processing of 50 candidates per query
    - GPU-accelerated embedding generation

Key Innovation Points:

1. **Hybrid AI Pipeline:** Combines LLM-based feature extraction with traditional text processing
2. **Domain-Specific Taxonomies:** Custom classification systems for HR tech
3. **Incremental Processing:** Resume-safe CSV appending for fault tolerance
4. **Semantic-Aware Indexing:** Job context-aware similarity matching

This pipeline transforms raw web content into a search-optimized knowledge base, enabling high-performance assessment recommendations based on multidimensional feature matching.