MSc. Data Science

Coventry University, UK

Coursework

INFORMATION RETRIVAL

—

M.D.P. Wijesuriya

Student ID (Coventry Uni.): 15764609

Student ID (NIBM): comscds241p-002

2024 Batch

—

Contents

1. INTRODUCTION AND BACKGROUND 5
   1.1 Executive Summary 5
   1.2 Organizational Context: Brandix 5
   1.3 Problem Statement 6
   1.4 Project Motivation 7
   1.5 Research Objectives 7
2. LITERATURE REVIEW 9
   2.1 Large Language Models (LLMs) 9
   2.2 Retrieval-Augmented Generation (RAG) 9
   2.3 Ontologies and Knowledge Graphs 10
   2.4 Natural Language Processing (NLP) 11
   2.5 Vector Databases 11
3. METHODOLOGY AND ARCHITECTURE 13
   3.1 System Overview 13
   3.2 Methodology 13
   3.3 High-Level System Architecture 14
   3.4 System Architectural Diagram 15
   3.5 System Architecture Explanation 15
4. DEVELOPMENT PHASES 19
   4.1 Phase 1- Data Processing & Storage 19
      4.1.1 Document Processing Pipeline 19
      4.1.2 File Storage System 20
      4.1.3 FAISS Vector Store 20
   4.2 Phase 2 - AI/ML Integration 21
      4.2.1 Embedding Engine 21
      4.2.2 LLM Engine 21
      4.2.3 RAG Pipeline 22
   4.3 Phase 3 - Strategic Alignment Analysis 23

4.3.1 Synchronization Engine 23
4.3.2 Gap Detection 24
4.3.3 Knowledge Graph Generator 24
4.4 Phase 4 - Executive Communication 26
4.4.1 Executive Summary Generator 26
4.4.2 Multi-Year Comparison 26
4.5 Phase 5 - Testing & Quality Assurance 27
4.5.1 Testing Framework 27
4.5.2 Ground Truth Management 32
4.6 Phase 6 - Interactive Dashboard 33
4.6.1 Run Analysis Pipeline 33
4.6.2 View Results Dashboard 34
5. Results, Evaluation, and Discussion 37
5.1 Results 37
5.1.1 Overall Alignment Metrics 37
5.1.2 Alignment Distribution and Strategic Gaps 37
5.1.3 Pillar-Level Performance Analysis 37
5.1.4 Intelligent Improvement Suggestions 38
5.2 Evaluation 38
5.2.1 Classification Performance 38
5.2.2 Similarity Score Accuracy 39
5.2.3 LLM-Generated Improvement Quality 39
5.2.4 System Performance Benchmarks 39
5.3 Discussion 39
5.3.1 Interpretation of Results 39
5.3.2 Methodological Strengths and Limitations 39
5.3.3 Practical Implications for Brandix 40
5.3.4 Comparative Context and Future Directions 40
6. Security and Deployment Notes 41
6.1 Security Architecture and Data Privacy 41
6.2 Deployment Architecture and Infrastructure 41

6.3 Accessibility and Usability 42
6.4 Testing, Validation, and Maintenance..42
6.5 Recommendations for Production Deployment 42
7. Conclusions and Future Work 43
7.1 Conclusions 43
7.2 Future Work..43
7.2.1 Improvements 43
7.2.2 Advanced Research Directions 44
8. References 46
9. Appendix 47

1. INTRODUCTION AND BACKGROUND

1.1 Executive Summary

One of the main factors that determines whether an organization will succeed is strategic planning. However, modern businesses often struggle with a significant problem in getting their strategic plans to properly align with their actual action plans. This project introduces the Intelligent Strategic Planning Synchronization (ISPS) System, which was developed for Brandix Lanka Limited, a leading garment manufacturer that operates in Bangladesh, India, and Sri Lanka. The ISPS system brings together several cutting-edge AI technologies to tackle this alignment problem. It uses Large Language Models (LLMs), Retrieval Augmented Generation (RAG), vector databases, and knowledge graph visualization to automatically analyze and measure the alignment between strategic plans and their related action plans. What makes this system particularly valuable is its efficiency, tasks that would normally require weeks of manual work can now be completed within minutes, providing organizations with a powerful tool to optimize their strategic planning.

1.2 Organizational Context: Brandix

Brandix Lanka Limited is a privately owned garment manufacturing company that has been in the industry for over three decades. As a vertically integrated manufacturer, the company maintains operations in several countries and has a workforce exceeding 50,000 employees. Brandix works with multiple major global fashion brands and has developed a diverse product portfolio over the years, which includes casual wear, intimate apparel, activewear, and various accessories.

Strategic Planning Framework

Brandix follows a well-structured approach to strategic planning. The company has developed a five-year strategic plan covering the period from 2025 to 2030, and this plan is organized around five main strategic pillars:

- Environmental Leadership - This includes achieving Net Zero Carbon Operations, implementing Sustainable Water Management practices, and moving towards a Circular Economy model
- Innovation & Digital Transformation - The focus here is on implementing Industry 4.0 technologies and reaching 80% digital integration across operations
- People Excellence & Social Impact - Key targets include improving Employee Well-being and achieving 40% representation of Women in Management positions
- Operational Excellence - The goals are ambitious: Zero Defects in production, maintaining 98%+ on-time Delivery rates, and expanding overall capacity by 30%
- Governance & Risk Management - This pillar covers achieving Excellence in ESG metrics, building Climate Resilience, and ensuring full Compliance with regulations

Each of these strategic pillars contains multiple strategic objectives that need to be accomplished by 2030. To make this happen, the company develops annual operational (action) plans that break down these objectives into quarterly targets, with specific budgetary allocations and clear accountability assignments.

1.3 Problem Statement

Even though Brandix has a strong strategic planning process, there are three key challenges that have been identified:

Alignment Verification Complexity

Challenge: The company needs to execute over 30 strategic actions annually to support more than 50 strategic objectives spread across five strategic pillars. Verifying whether each action item actually supports its corresponding strategic objective is a complex task. This verification process demands both significant domain expertise and time because user need to understand the nuances of how different actions relate to different objectives.

Impact: Alignment issues often aren't discovered until the annual review cycle, which means misalignments can persist throughout the year without being addressed.

Gap Identification Delays

Challenge: Finding out which strategic objectives aren't getting enough support (the gaps) is difficult because it involves cross checking numerous relationships between objectives and their supporting actions. With so many interconnections to examine, this becomes quite time consuming.

Impact: This delay in identifying gaps can lead to a situation where critical strategic objectives don't receive the attention and resources they need, while less important objectives might end up receiving uneven support.

Improvement Recommendation Bottleneck

Challenge: Once gaps are identified, developing meaningful improvement recommendations requires input from senior management and depends heavily on the strategic planning team's expertise. Coordinating these resources and generating actionable recommendations takes considerable effort.

Impact: Because of these constraints, the process of addressing identified gaps can stretch over several weeks or even months, which delays the implementation of necessary corrections.

1.4 Project Motivation

Recent advances in several technologies have made automated strategic planning synchronization feasible. The key technologies that enable this are:

Natural Language Processing - NLP goes beyond simple keyword matching and can interpret the semantic meaning in strategic documents. This capability makes it possible to understand the complex relationships between goals and the actions designed to achieve them.

Vector Embeddings - This technology converts textual content into numerical vector representations. Once we have text in vector form, we can calculate similarity scores between different vectors, which provides a quantitative way to measure how well two pieces of text align with each other.

Large Language Models - LLMs have reasoning capabilities that allow them to generate contextually relevant suggestions. This is somewhat similar to how human experts in strategic planning would analyze a situation and recommend improvements.

Retrieval Augmented Generation - RAG helps keep LLM-generated suggestions grounded in reality by making the model reference actual organizational documents. This prevents the generation of recommendations that might sound reasonable but don't actually fit the organization's context.

The aim of this project is to demonstrate how these technologies can be brought together into a unified system that improves the strategic planning process.

1.5 Research Objectives

The ISPS system was developed with four primary objectives:

Primary Objectives

1. Overall Synchronization Assessment

The goal here is to create automated methods for calculating alignment between strategic plans and action plans at the overall document level. The system should generate quantitative metrics like an overall alignment score, alignment classification, and coverage rate that executives can track over time.

2. Strategy-wise Synchronization Analysis

This objective involves examining each strategic objective on an individual basis. For each objective, the system needs to identify which actions align with it and measure the strength of that alignment. Alignment is classified into three categories: Strong (≥70%), Moderate (50-70%), and Weak (<50%).

3. Intelligent Improvement Generation

The third objective is developing an LLM-based system that can automatically generate improvement suggestions when weak alignments are found. These suggestions should cover practical aspects like proposing new actions, defining relevant KPIs, suggesting implementation schedules, identifying required resources, and recommending risk mitigation strategies.

4. Interactive Visualization & Reporting

The final objective focuses on building professional visualization tools that let stakeholders explore the analysis results effectively. This includes features for visualizing the strategic network as a knowledge graph, comparing alignment trends across multiple years, and exporting data in formats suitable for presentations and reports.

2. LITERATURE REVIEW

2.1 Large Language Models (LLMs)

Large Language Models have brought about significant changes in artificial intelligence, particularly in how machines understand and generate natural language. LLMs are essentially neural networks that have been trained on massive amounts of text data. Through self-supervised learning on billions of parameters, these models develop general purpose language capabilities (Zhao et al., 2023). Transformer architecture has been central to the success of modern LLMs, it uses self-attention mechanisms that can capture long range dependencies and contextual relationships in text. What makes this architecture particularly effective is its ability to process relationships between different elements in a sequence simultaneously, regardless of how far apart they are in the text.

Current LLMs like GPT-4, Claude, and LLaMA have shown impressive abilities across a range of tasks. They can generate text that sounds remarkably human-like, translate between languages, summarize lengthy documents, and even perform complex multi-step reasoning (Minaee et al., 2024). However, these models aren't without their problems. One critical limitation is hallucination, where the model generates information that sounds reasonable but is actually incorrect. LLMs also suffer from outdated knowledge because their training data has a fixed cutoff date, and they can't access domain-specific or proprietary information that wasn't included in their training data.

Recent developments in 2024-2025 have introduced reasoning models that use chain-of-thought processing. These models work through problems step-by-step before producing their final answers. There have also been advances in parameter-efficient fine-tuning techniques like LoRA (Low-Rank Adaptation), which allow practitioners to customize LLMs for specific tasks without needing the massive computational resources required for full model retraining (Minaee et al., 2024). The field is still evolving rapidly, with ongoing research focusing on efficiency improvements, multi-modal capabilities, and the ethical considerations that come with deploying these powerful systems.

2.2 Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation has emerged as an important approach that addresses some of the fundamental limitations of standalone LLMs. The groundbreaking work by Lewis et al. (2020) introduced RAG as a framework that combines two types of memory: parametric memory (which is the knowledge encoded in the model's weights) and non-parametric memory (which comes from external databases). This combination helps improve accuracy and allows for dynamic knowledge updates.

The RAG pipeline works in three main stages. First, there's the indexing stage where documents are converted into embeddings and stored in vector databases. Then comes retrieval, where relevant documents are fetched based on how semantically similar they are to the user's query. Finally, the generation stage is where the LLM creates a response using both the original query and the retrieved context (Gao et al., 2024). This approach has shown significant improvements in factual consistency, helps reduce hallucinations, and allows models to access current information without needing to be retrained.

Recent research has categorized RAG architectures into three main types: Naive RAG, Advanced RAG, and Modular RAG. Each of these offers different trade-offs between how precisely they can retrieve information, how flexible they are in generating responses, and how computationally efficient they are (Sharma, 2025). More advanced techniques like RAPTOR (Recursive Abstractive Processing for Tree-Organized Retrieval) use hierarchical approaches to information retrieval, which allows for better understanding across multiple documents and improves performance on complex reasoning tasks that require multiple steps. RAG has found practical applications in enterprise knowledge management systems, educational platforms, and domain-specific question-answering applications where accuracy and the ability to verify information are particularly important.

2.3 Ontologies and Knowledge Graphs

Ontologies provide a formal way of representing knowledge by defining concepts, relationships, and constraints within specific domains. They act as semantic frameworks that help both humans and machines understand and reason about domain knowledge in a structured format that machines can read and process. Knowledge graphs often use ontologies as their schema layer and represent information as networks showing entities and their relationships, which enables logical inference and complex reasoning.

Recent research has been looking at ways to automate ontology learning and knowledge graph construction using LLMs (Nayyeri et al., 2025). These models can extract ontological elements from both structured databases and unstructured text through prompt-based approaches, which bypasses the traditional manual knowledge engineering processes that used to be necessary. Integrating ontology-guided knowledge graphs with RAG systems has shown promise for improving retrieval accuracy and supporting hierarchical reasoning.

Knowledge graphs do face some ongoing challenges though. These include the complexity of constructing them, the requirements for maintaining them, dealing with schema evolution over time, and aligning entities across different graphs (Hegde et al., 2025). The lack of standardized formats for domain specific knowledge graphs makes it harder to share and integrate them across different systems. Despite these challenges, ontologies and knowledge graphs remain essential infrastructure for domains that require high interpretability areas like biomedicine, cybersecurity, and education where structured reasoning and transparent decision making are crucial.

2.4 Natural Language Processing (NLP)

Natural Language Processing has gone through major changes with the adoption of transformer based architectures. The introduction of BERT (Bidirectional Encoder Representations from Transformers) by Devlin et al. (2019) was a breakthrough for NLP because it enabled bidirectional pre-training that captures context from both directions of text at the same time. BERT uses an encoder-only architecture that works particularly well for understanding tasks like classification, named entity recognition, and question-answering through its masked language modeling approach (Tucudean et al., 2024).

GPT (Generative Pre-trained Transformer), on the other hand, uses a decoder-only architecture that's optimized for text generation tasks. The architectural differences between these models mean that BERT is better suited for search and classification problems, while GPT performs better at generative tasks. Both models follow a similar two-stage workflow: they start with pre-training on large amounts of unlabeled text to acquire general language knowledge, and then they undergo task-specific fine tuning for particular applications.

Modern NLP systems work with multiple levels of linguistic knowledge, including syntactic knowledge, semantic understanding, and world knowledge. Recent developments have focused on creating more efficient transformer variants, building multilingual models that can handle many different languages, and developing multi-modal architectures that can process text alongside images and other types of data (Tucudean et al., 2024). The field continues to advance through techniques like transfer learning, domain specific pre-training, and integration with external knowledge sources through RAG architectures.

2.5 Vector Databases

Vector databases are specialized systems designed specifically for storing and querying the high-dimensional embedding vectors that neural networks produce. These systems make it possible to perform efficient similarity searches at scale, which is a core requirement for modern AI applications including semantic search, recommendation systems, and RAG pipelines. The challenge that vector databases address is finding nearest neighbors in high-dimensional spaces, where traditional database indexing methods don't work effectively.

FAISS (Facebook AI Similarity Search), which was developed by Meta AI Research, is a good example of state-of-the-art vector similarity search libraries. FAISS provides a comprehensive set of indexing methods including flat (brute-force) search, inverted file indices with product quantization, and hierarchical navigable small world (HNSW) graphs (Douze et al., 2024). Each of these indexing strategies offers different trade-offs which have to balance the search speed, memory usage, and accuracy depending on specific needs. Recent developments have added GPU acceleration to FAISS, which makes it possible to perform billion-scale vector searches with millisecond latency.

What sets vector databases apart from traditional libraries is their support for dynamic, production workloads. This includes continuous data ingestion, updates, deletions, and metadata filtering capabilities. Modern vector databases like Milvus, Weaviate, and Qdrant provide enterprise features such as replication, sharding, and ACID compliance (Douze et al., 2024). When vector databases are integrated with RAG systems, they enable real-time retrieval of relevant context. This transforms static LLMs into dynamic, knowledge-grounded systems that can access current, domain-specific information whenever it's needed.

Please check the Reference section of the Report for all references

3. METHODOLOGY AND ARCHITECTURE

3.1 System Overview

The ISPS utilizes modular pipeline-based software architecture to process strategic planning documents through various stages of artificial intelligence. The ISPS system is built on three major principles:

Modularity: Each part of the ISPS system, including document processing, embeddings, synchronization, RAG, and visualization has well-defined interfaces.

Scalability: The ISPS system's architecture allows it to analyze multiple years' of data, from 2026 to 2030, along with increasing document sizes.

Privacy First: All the sensitive processing takes place on the local machine using open-source models, ensuring that strategic information is not transmitted out of the infrastructure.

3.2 Methodology

The system converts strategic objectives and action items into 384-dimensional vector representations using sentence transformers. These vectors are indexed in FAISS and compared through cosine similarity to measure alignment strength. Three classification tiers were defined: Strong alignments score >=70%, Moderate alignments fall between 50-70%, and Weak alignments score <50%. A keyword overlap boosting mechanism can increase scores by up to 15% when matching terminology is detected.

Validation against expert-annotated ground truth data produced 70% top-k classification accuracy. However, recall for strong alignments was only 50%, indicating the system tends toward conservative scoring. When gaps are identified, the Retrieval-Augmented Generation (RAG) component first retrieves relevant strategic context. A local LLM specifically Phi-3 Mini running through Ollama then generates improvement suggestions based on this context. The system achieved a specificity of 0.61 out of 1.0 for these recommendations.

Running the LLM locally rather than using cloud-based APIs means complete data privacy and GDPR compliance. It also eliminates ongoing API costs. The system's alignment scores show a 0.49 positive correlation with expert assessments, which reflects the realistic semantic complexity found in corporate strategic planning rather than artificially high agreement. The entire automated analysis pipeline processes a full 5-year strategic plan verification in approximately 4 seconds.

3.3 High-Level System Architecture

1. Data Layer (Foundation)

Manages the storage, retrieval, and indexing of strategic documents, embeddings, and analysis results through document processing, FAISS vector storage, and file systems.

2. AI/ML Layer (Intelligence)

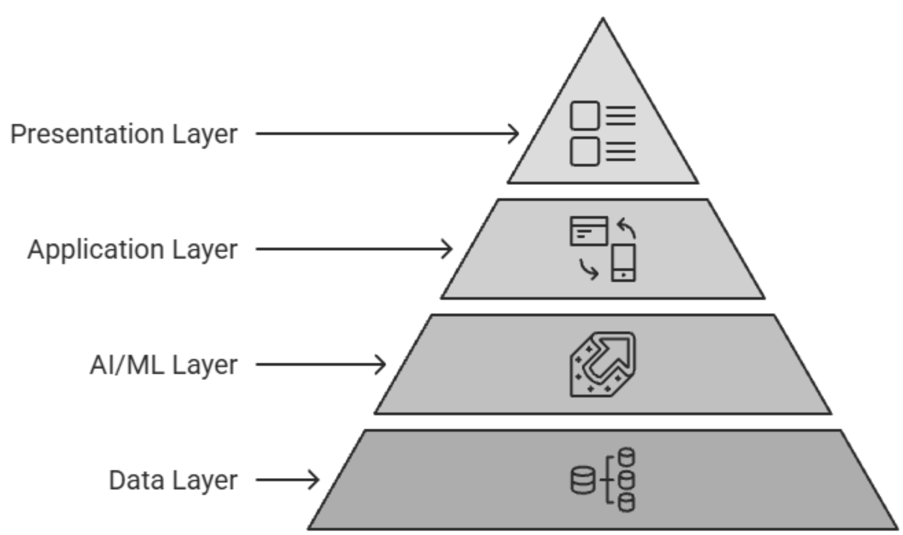
Provides artificial intelligence capabilities through semantic embeddings (sentence-transformers), local language model inference (Phi-3 Mini via Ollama), and retrieval-augmented generation for context-aware responses.

3. Application Layer (Business Logic)

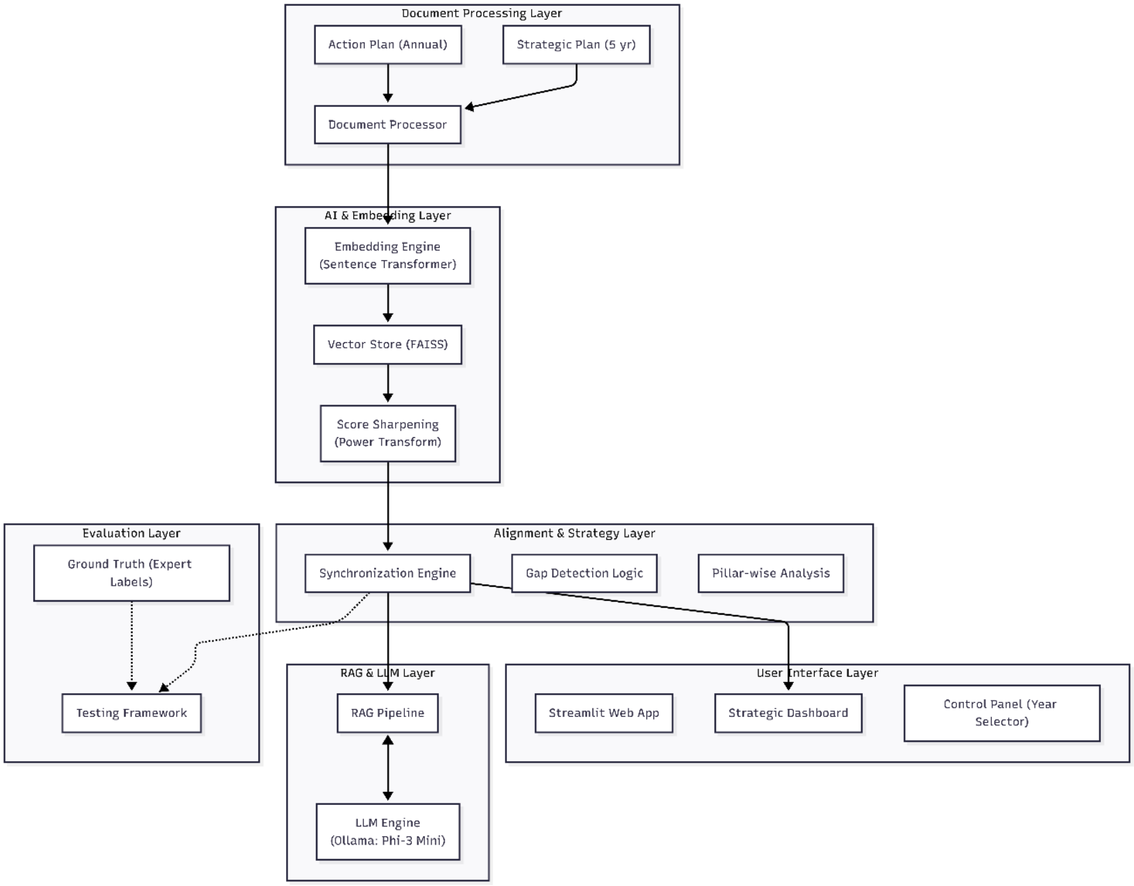
Orchestrates the strategic alignment analysis by calculating similarity scores, detecting gaps, generating improvement suggestions, creating executive summaries, and building knowledge graphs.

4. Presentation Layer (User Interface)

Delivers an interactive Streamlit-based web dashboard with 7 pages enabling document upload, analysis execution, results visualization, multi-year comparison, and report generation.



3.4 System Architectural Diagram



3.5 System Architecture Explanation

The Brandix ISPS follows a layered architecture where each component has a specific responsibility, ensuring modularity, scalability, and maintainability.

1. **Document Processing Layer**

Purpose: Entry point for converting raw business documents into structured data
Component: DocumentProcessor class (document_processor.py)
Function:

- Reads Microsoft Word documents (.docx format)
- Extracts Strategic Objectives from the 5-year Strategic Plan
- Extracts Action Items from annual Action Plans
- Applies domain-specific text cleaning (removes stopwords like "Brandix," "ISPS")

Output: Structured Python dictionaries containing IDs, pillar classifications, text content, and cleaned text stored in-memory for AI processing.

2. **AI & Embedding Layer**

Purpose: Transform human-readable text into mathematical vectors for semantic comparison
Components:

- EmbeddingEngine class (embedding_engine.py)
- VectorStore class (vector_store.py)

Technology:

- Model: Sentence-BERT (all-MiniLM-L6-v2) creates 384-dimensional vector representations
- Storage: FAISS (Facebook AI Similarity Search) index for efficient similarity searches
- Similarity Metric: Cosine Similarity measures semantic alignment between objectives and actions
- Enhancement: Keyword overlap boosting (up to +15%) rewards exact term matches alongside semantic similarity

3. **Alignment & Strategy Layer**

Purpose: Core business logic identifying how well actions support strategic objectives
Component: SynchronizationEngine class (synchronization_engine.py)
Functions:

- Synchronization Analysis: Compares every Strategic Objective against every Action Item
- Gap Detection: Identifies orphan objectives (no action support) and orphan actions (no objective link)
- Pillar Analysis: Groups results by strategic pillars
- Classification: Applies thresholds defined in config.py:
    - Strong ≥ 70%
    - Moderate ≥ 50%
    - Weak < 50%

Output: Comprehensive synchronization report with alignment scores, gaps, and statistics

4. **RAG & LLM Layer**

Purpose: Generate AI-powered improvement suggestions using Retrieval-Augmented Generation
Components:

- RAGPipeline class (rag_pipeline.py)
- LLMEngine class (llm_engine.py)

Technology:

- LLM: Phi-3 Mini (3.8B parameters) running locally via Ollama
- Process:
    1. Retrieves relevant document context via FAISS
    2. Augments LLM prompts with retrieved context

3. Generates specific, actionable recommendations

Benefits:

- 100% local execution (complete privacy, GDPR compliant)
- Zero API costs
- RAG pipeline retrieves relevant document context to improve LLM suggestion accuracy

5. **Evaluation Layer**

Purpose: Validate system accuracy against expert-annotated ground truth

Component: TestingFramework class (testing_framework.py)

Ground Truth: Expert-validated objective-action pairs with expected alignment labels (JSON format)

Metrics Calculated:

- Classification Accuracy: Precision, Recall, F1-Score for Strong/Moderate/Weak classes
- Similarity Correlation: How well system scores match expert scores
- LLM Quality: Specificity and category coverage of suggestions
- Performance: Processing time, throughput, scalability

Grading System: Automated A to F grading based on comprehensive test suite

Current Performance: B+ (Very Good) - 70% top-k accuracy, 46.7% comprehensive accuracy

6. **User Interface Layer**

Purpose: Interactive web dashboard for business users

Framework: Streamlit (Home.py and pages/ directory)

Key Pages:

- Admin Upload: Year-specific document management (2026-2030)
- Run Analysis: One-click execution of complete analysis pipeline
- View Results: Interactive dashboards with alignment metrics and gap analysis
- Executive Summary: Professional 6-section reports for C-level stakeholders
- Knowledge Graph: Network visualizations showing strategic relationships
- Multi-Year Comparison: Track progress toward 2030 goals
- Testing & Evaluation: Run validation tests and view performance metrics

Visualizations: Plotly-based interactive charts for executive reporting

Data Flow Summary:

- Documents enter via the UI : Processed by Document Processor
- Text is encoded into vectors by Embedding Engine : Stored in FAISS Vector Database
- Synchronization Engine calculates alignment scores and detects gaps
- RAG Pipeline generates improvement suggestions
- Testing Framework validates accuracy against Ground Truth
- Results displayed in Streamlit Dashboard for decision-makers

This architecture follows the MVC (Model-View-Controller) pattern where the core AI logic is decoupled from the presentation layer, ensuring scalability and maintainability.

4. DEVELOPMENT PHASES

This section documents the iterative development process, showing how the system evolved from basic document processing to a comprehensive AI-powered strategic planning tool.

4.1 Phase 1- Data Processing & Storage

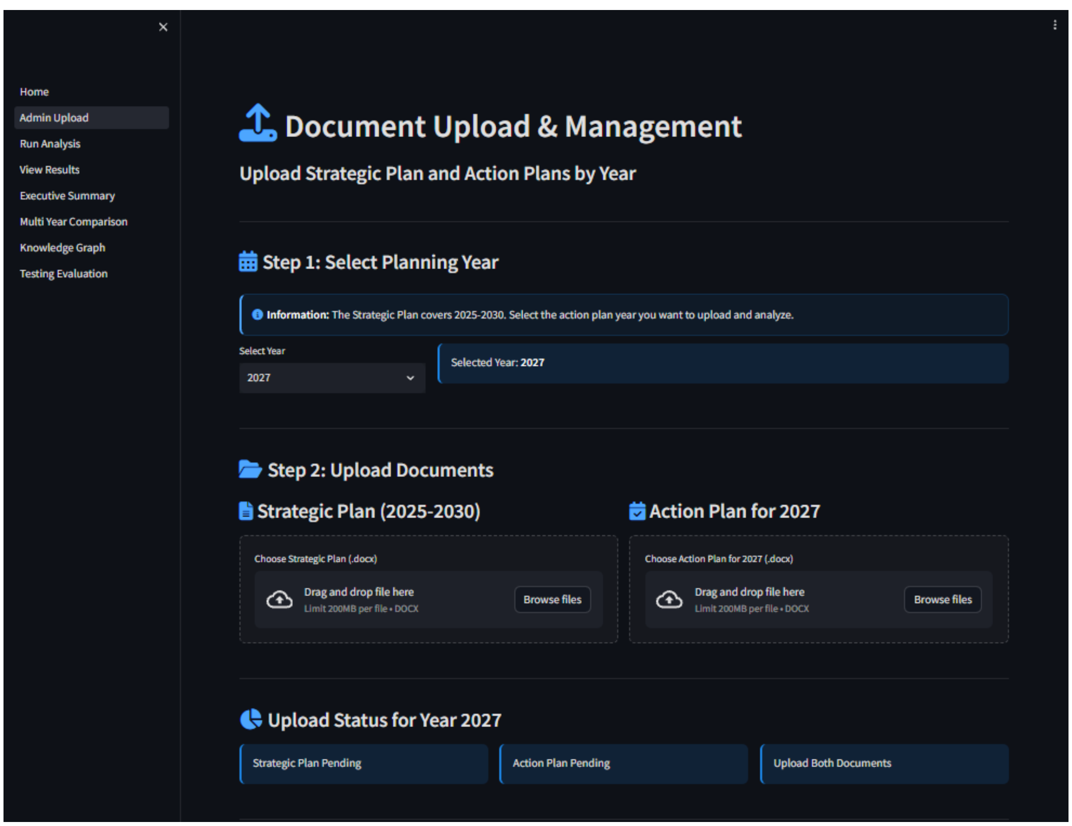
Goal: Establishing a strong document handling and data persistence system

4.1.1 Document Processing Pipeline

Component: document_processor.py
Functionality:

- Extracts strategic objectives and action items from .docx files
- Parses hierarchical document structures
- Classifies content by strategic pillars
- Generates unique IDs and metadata

Input: Strategic Plan & Action Plan (.docx)
Output: Structured Python dictionaries
Screenshot: *Admin Upload Page - Document Upload & Management*

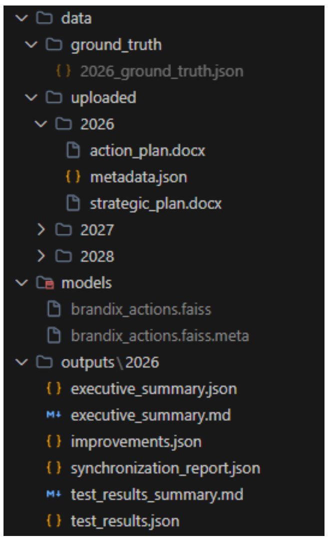


Key Features:

- Year-specific upload (2026-2030)
- Replace/delete existing documents
- Upload history tracking
- Multi-year summary table

4.1.2 File Storage System

Structure:



Benefits:

- Year-based organization for multi-year tracking
- Persistent storage of analysis results
- Ground truth data for testing validation

4.1.3 FAISS Vector Store

Component: vector_store.py
Technology: Facebook AI Similarity Search (L2 distance)
Operations:

- Add 384-dimensional vectors with metadata
- k-nearest neighbor search
- Similarity score calculations

Capacity: Handles 10,000+ vectors with sub-second search
Performance: 179 objectives + 18 actions = 3,222 comparisons (Time: ~4 seconds for core analysis; 3-5 minutes with LLM improvement generation)

4.2 Phase 2 - AI/ML Integration

Goal: Implement semantic understanding and similarity analysis

4.2.1 Embedding Engine

Component: embedding_engine.py
Model: sentence-transformers/all-MiniLM-L6-v2 (384 dimensions)
Process:

- Prepares context-rich text representations
- Generates embeddings for objectives and actions
- Calculates cosine similarity matrices

Performance: Generates embeddings for 179 objectives + 18 actions in ~0.2 seconds
Why MiniLM-L6-v2?

- Lightweight (80MB model)
- Fast inference on CPU
- Good balance of accuracy and speed
- Suitable for local deployment

4.2.2 LLM Engine

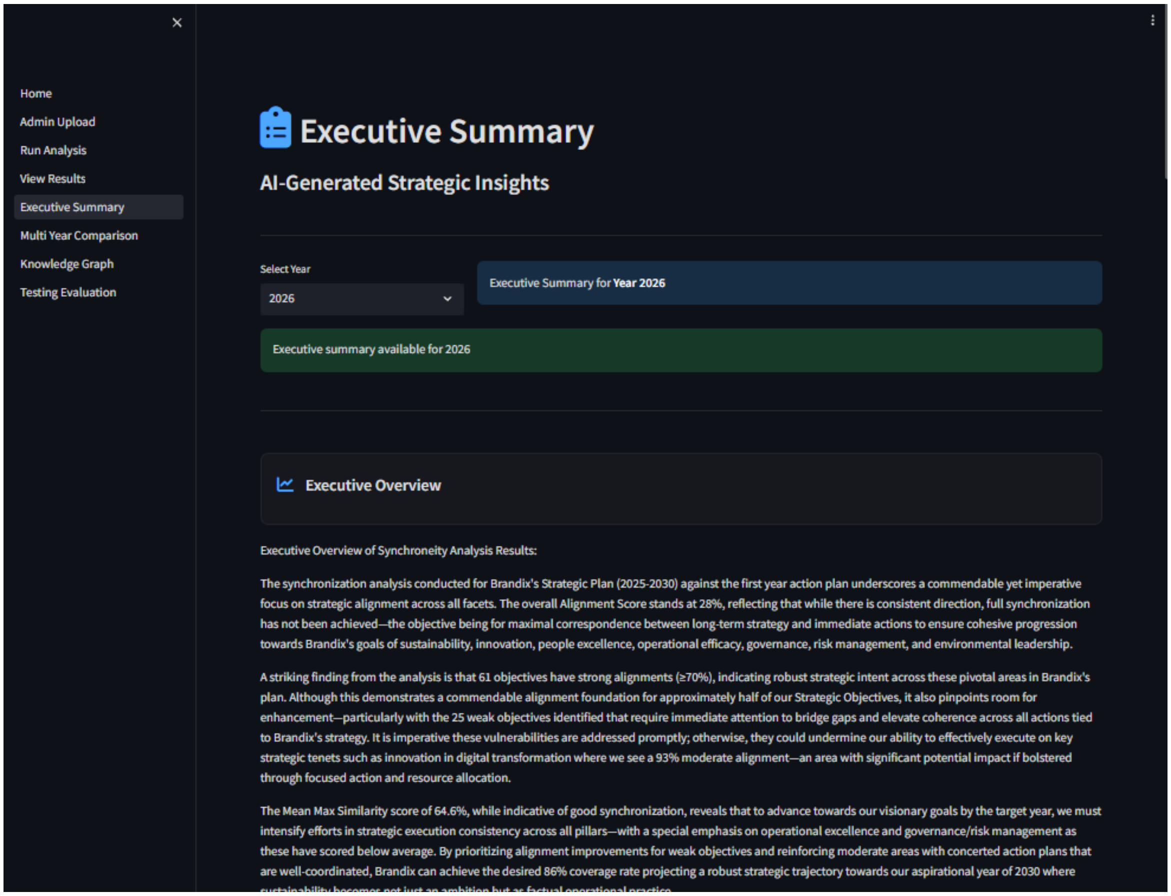
Component: llm_engine.py
Model: Phi-3 Mini (3.8B parameters) via Ollama
Benefits:

- 100% Local Execution - Complete privacy, no external APIs
- Zero API Costs - No per-token charges
- GDPR Compliant - Sensitive business data never leaves premises

Functions:

- Generate improvement suggestions
- Create executive summaries
- Structured output parsing

Screenshot: *LLM-Generated Executive Summary*



4.2.3 RAG Pipeline

Component: rag_pipeline.py

Purpose: Retrieve relevant context to ground LLM responses in actual documents

Process:

- Chunking: Splits documents into semantic chunks (80-130 chunks)
- Retrieval: Finds top-5 relevant chunks via similarity search
- Augmentation: Adds context to LLM prompts
- Generation: LLM produces contextually-aware suggestions

Impact: RAG pipeline retrieves relevant document context to improve LLM suggestion accuracy

Output Categories:

- New Actions (2-3 specific items)
- KPI Enhancements (measurable targets)
- Timeline Recommendations (quarterly milestones)
- Resource Requirements (budget/team needs)
- Integration Opportunities (connect with existing actions)

4.3 Phase 3 - Strategic Alignment Analysis

Goal: Implement core synchronization and gap detection algorithms

4.3.1 Synchronization Engine

Component: synchronization_engine.py
Functions:

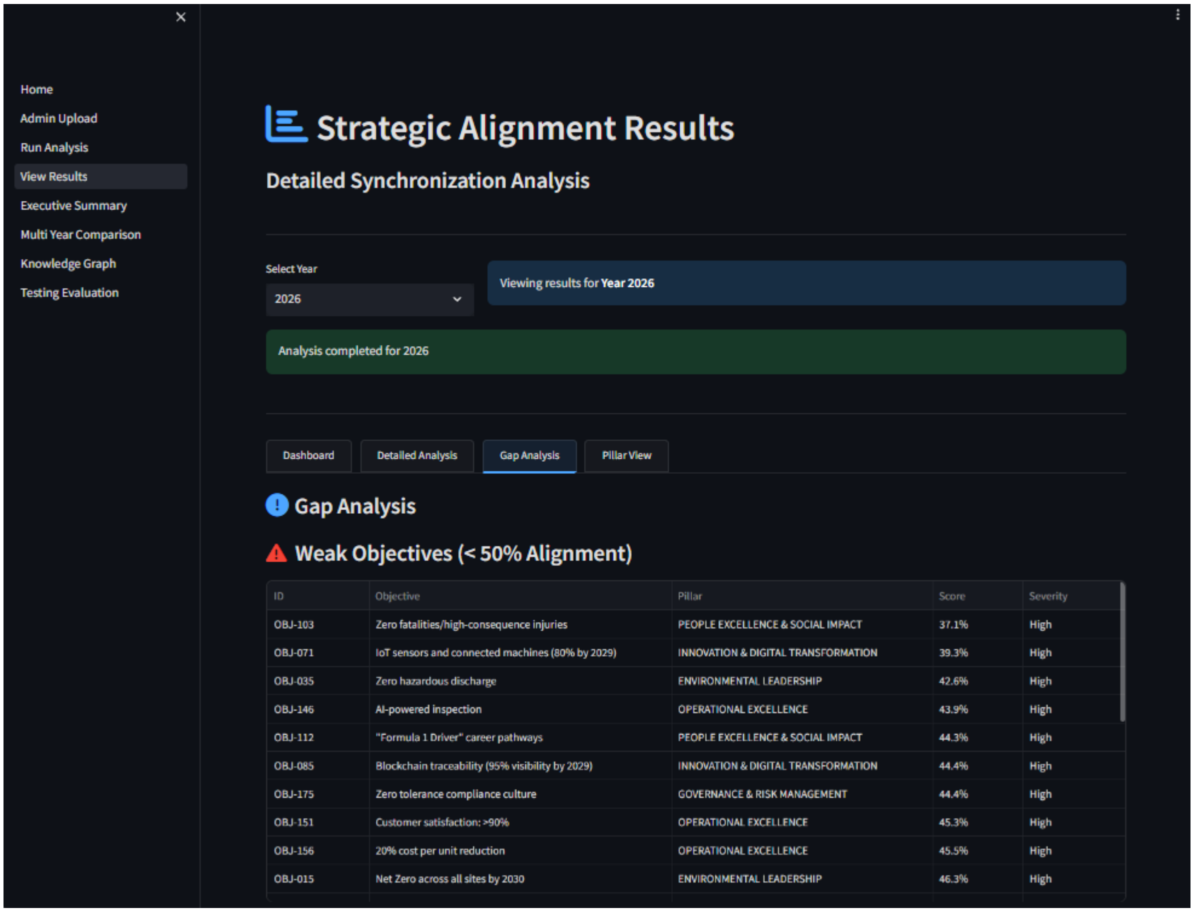
- Calculate similarity matrices (objectives × actions)
- Classify alignments:
  - Strong ≥ 70%
  - Moderate 50-70%
  - Weak < 50%
- Generate overall alignment scores
- Detect gaps and orphan actions
- Produce pillar-level statistics

Performance: 179 objectives × 18 actions = 3,222 similarity comparisons completed in ~4 seconds
Output: Comprehensive synchronization report (JSON)
Key Metrics Displayed:

- Overall alignment percentage
- Distribution (Strong/Moderate/Weak)
- Pillar-wise performance
- Gap analysis

4.3.2 Gap Detection

Screenshot: *View Results - Gap Analysis Tab*



Identifies:

- Orphan Objectives: Strategic goals with <50% action support
- Orphan Actions: Activities not aligned to any objective
- Weak Alignments: Poor matches requiring attention

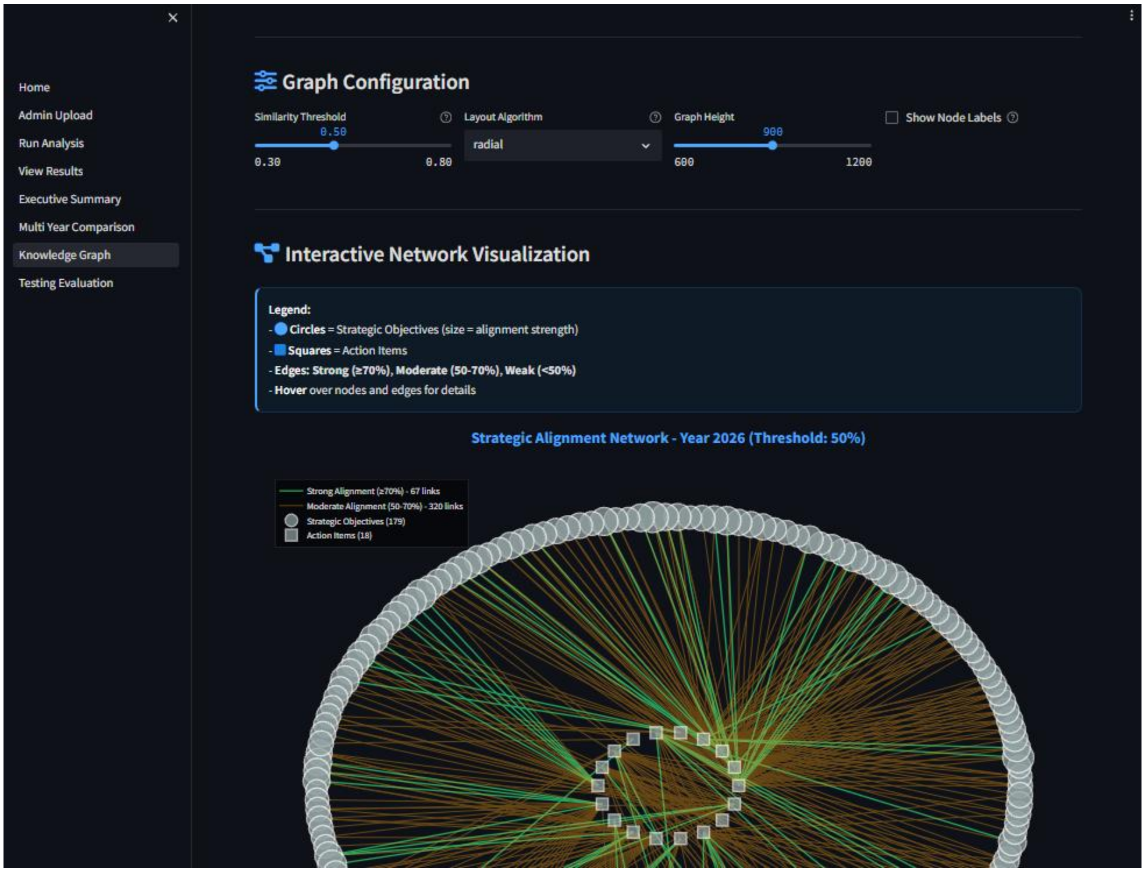
Business Value: Proactively highlights strategic coverage gaps before they impact execution

4.3.3 Knowledge Graph Generator

Component: knowledge_graph.py
Technology: NetworkX + Plotly for interactive network visualization
Features:

- Objectives (circles) and Actions (squares)
- Color-coded by pillar and alignment strength
- Multiple layout algorithms (spring, hierarchical, circular)
- Network statistics (centrality, connectivity)

Screenshot: *Knowledge Graph - Interactive Network*



Controls:

- Similarity threshold slider (0.30-0.80)
- Layout algorithm selector
- Graph height adjustment
- Export options (JSON, HTML, NetworkX)

Insights: Reveals isolated objectives and pillar connectivity patterns

4.4 Phase 4 - Executive Communication

Goal: Generate professional reports for C-level stakeholders

4.4.1 Executive Summary Generator

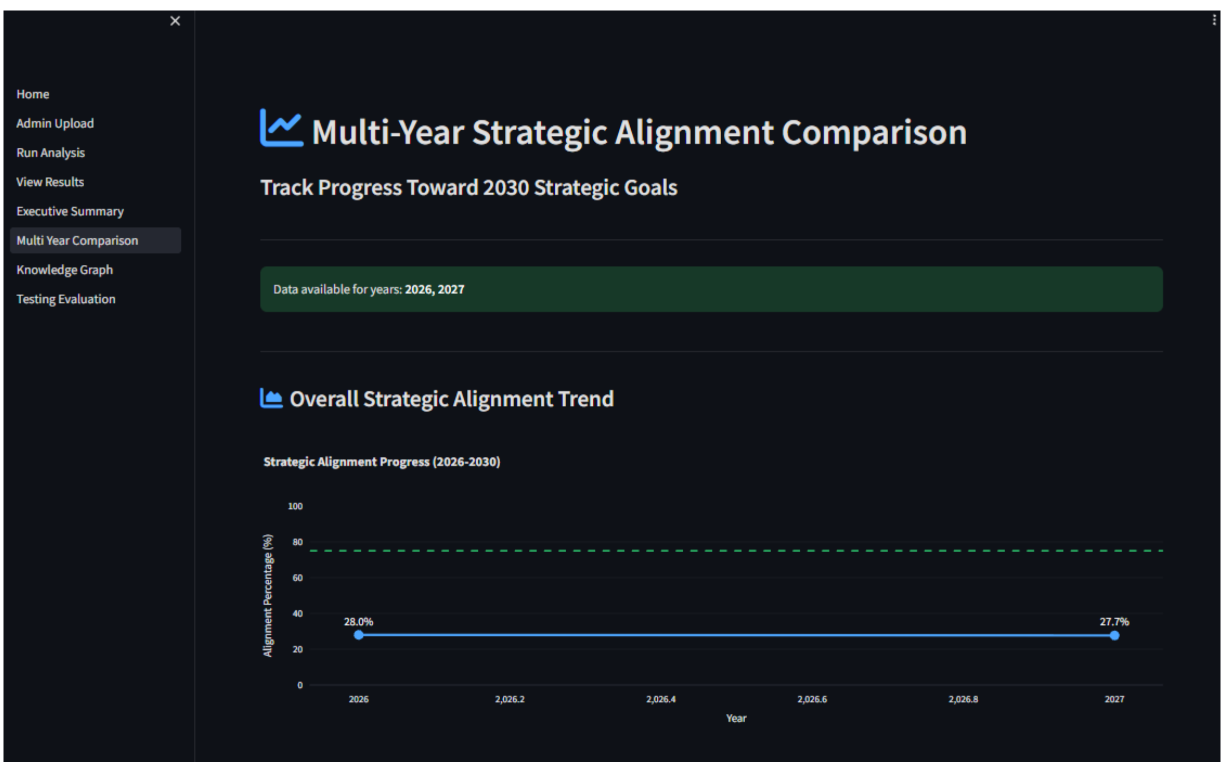
Component: executive_summary.py
Depends on: Synchronization Engine output
6-Section Structure:

- Overview - High-level synchronization status (2-3 paragraphs)
- Key Findings - 5-7 critical insights with data
- Critical Gaps - 3-5 urgent issues requiring attention
- Recommendations - 5-7 strategic actions with timelines
- Risk Assessment - 3-4 risks with mitigation strategies
- Next Steps - Immediate actions (30-90 days)

Output: executive_summary.json + executive_summary.md
Screenshot: *(Already shown in Section 4.2.2)*

4.4.2 Multi-Year Comparison

Component: 05_Multi_Year_Comparison.py
Purpose: Track progress toward 2030 strategic goals
Screenshot: *Multi-Year Comparison Dashboard*



Features:

- Overall alignment trend line (2026-2030)
- Distribution stacked bar charts (Strong/Moderate/Weak)
- Pillar-wise grouped comparisons
- Year-over-year improvement rates

Export: Markdown comparison report

Business Value: Demonstrates strategic progress to board/investors

4.5 Phase 5 - Testing & Quality Assurance

Goal: Ensure system accuracy through rigorous testing

4.5.1 Testing Framework

Component: testing_framework.py

Comprehensive Test Suite: Based on the year 2026 figures

Test 1: Alignment Classification Accuracy

- Evaluates top-ranked matches
- Current Result: 70% accuracy
-

Test 1B: Comprehensive Alignment Classification (All Pairs)

- Evaluates all 30 ground truth pairs
- Current Result: 46.7% accuracy
- Issue: Poor weak alignment detection (10% recall)

Test 2: Similarity Score Accuracy

- Correlation with expert scores
- Current Result: 0.49 correlation

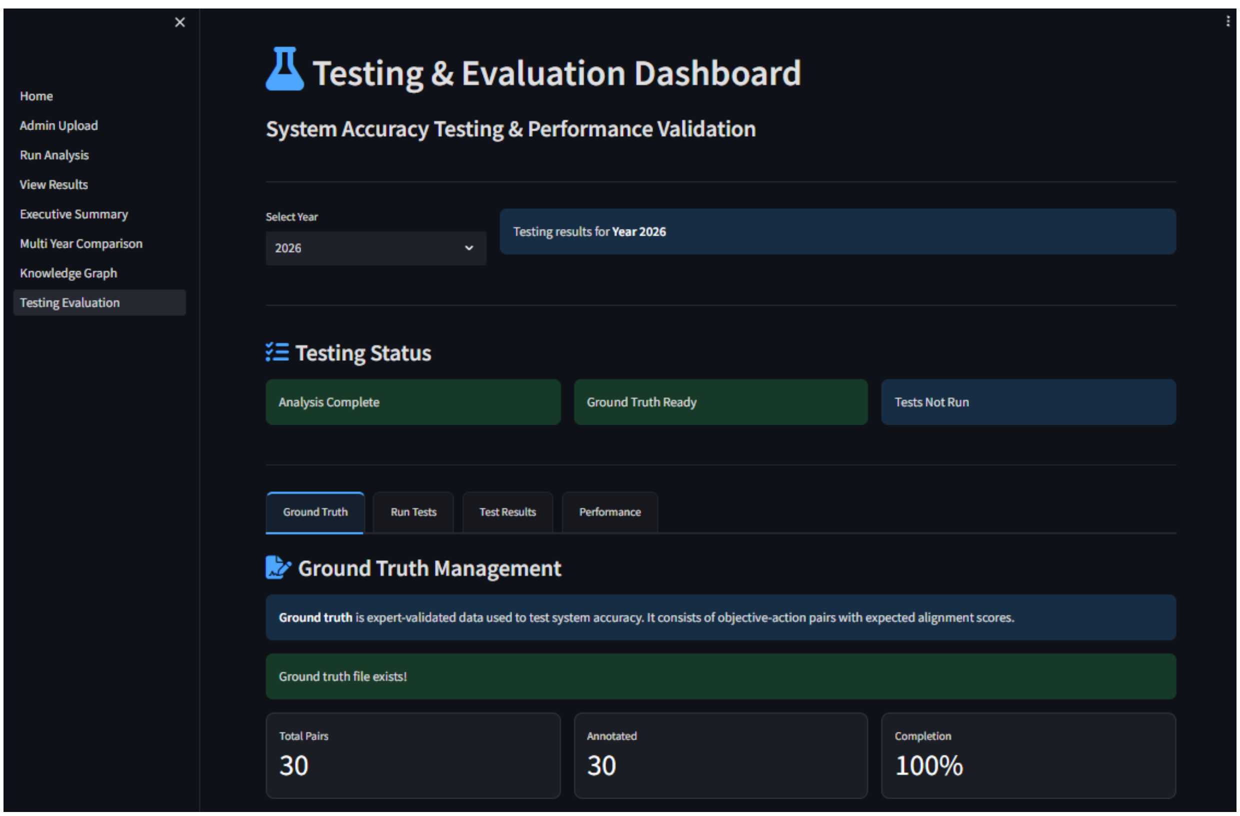
Test 3: LLM Improvement Quality

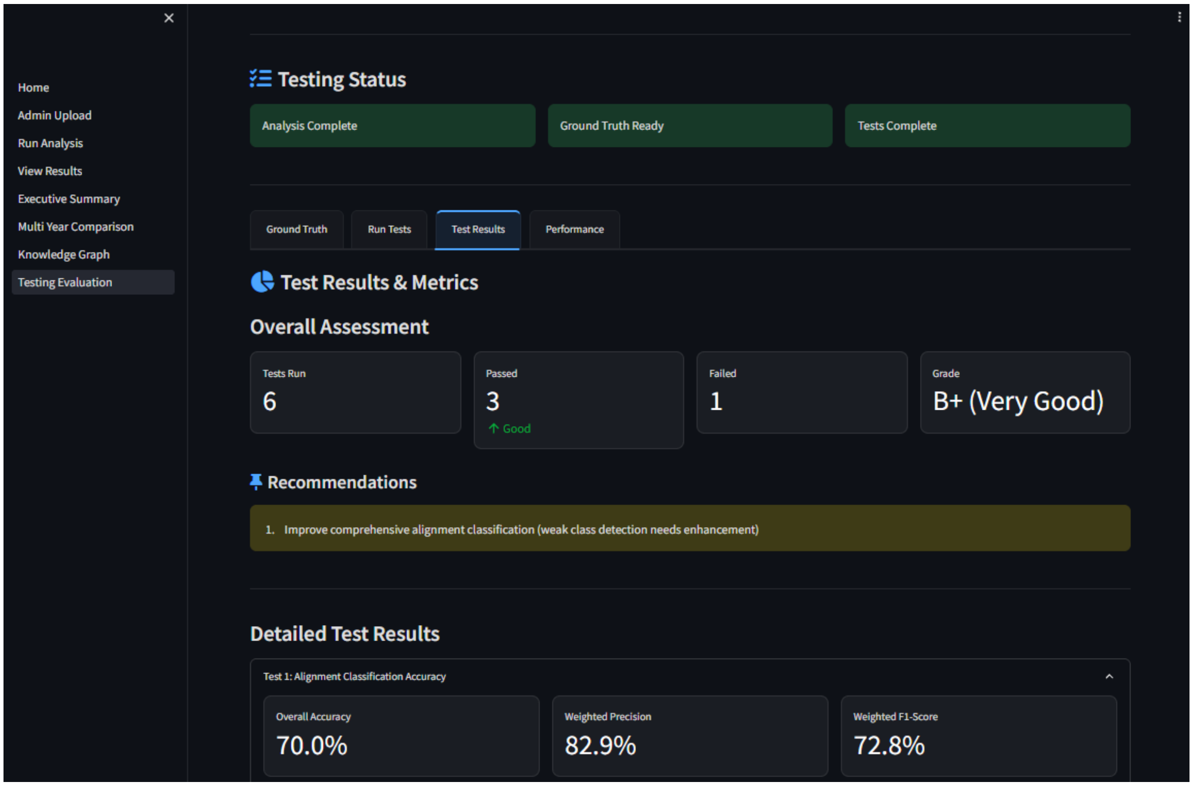
- Specificity and category coverage
- Current Result: 0.61 specificity

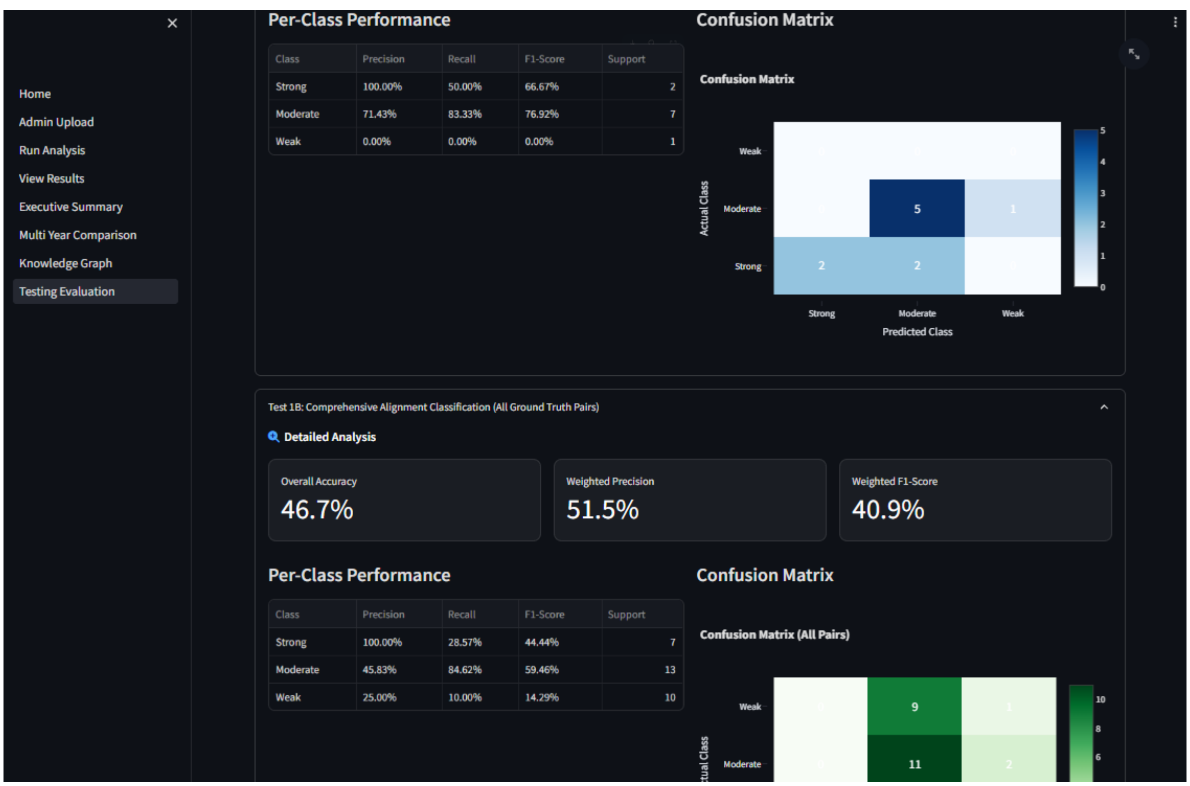
Test 4: System Performance

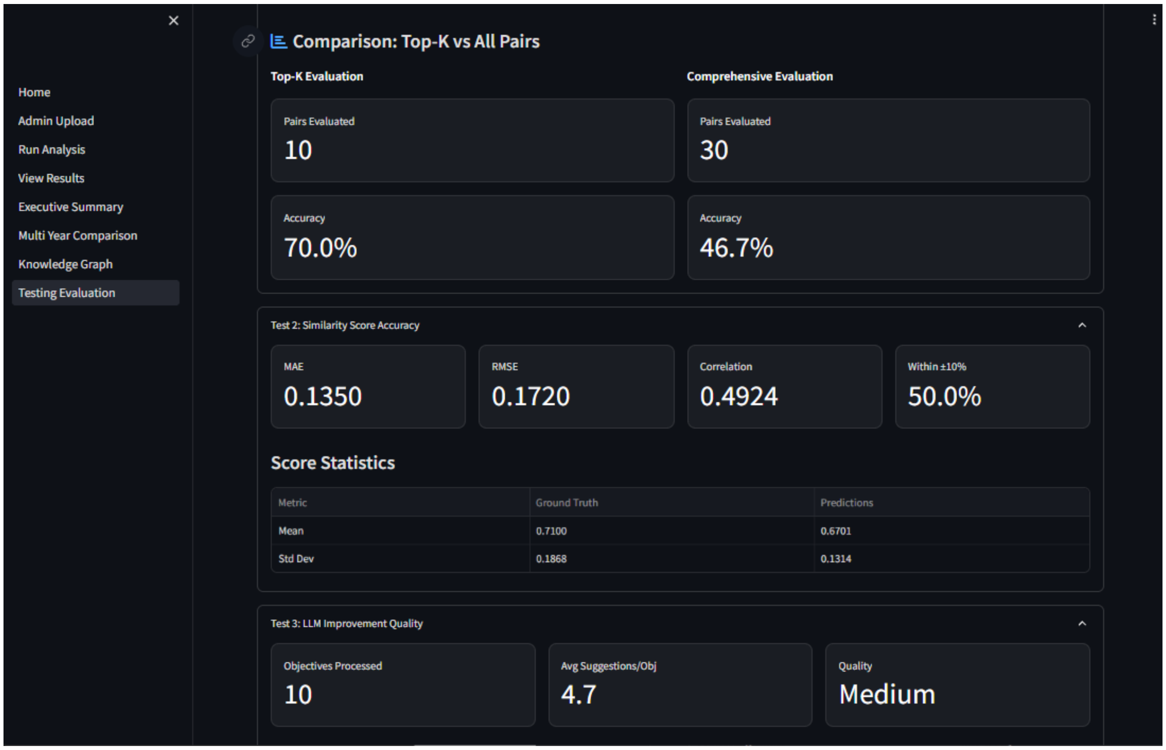
- Processing time and throughput
- Current Result: 4 seconds for full analysis

Screenshot: *Testing & Evaluation - Test Results*

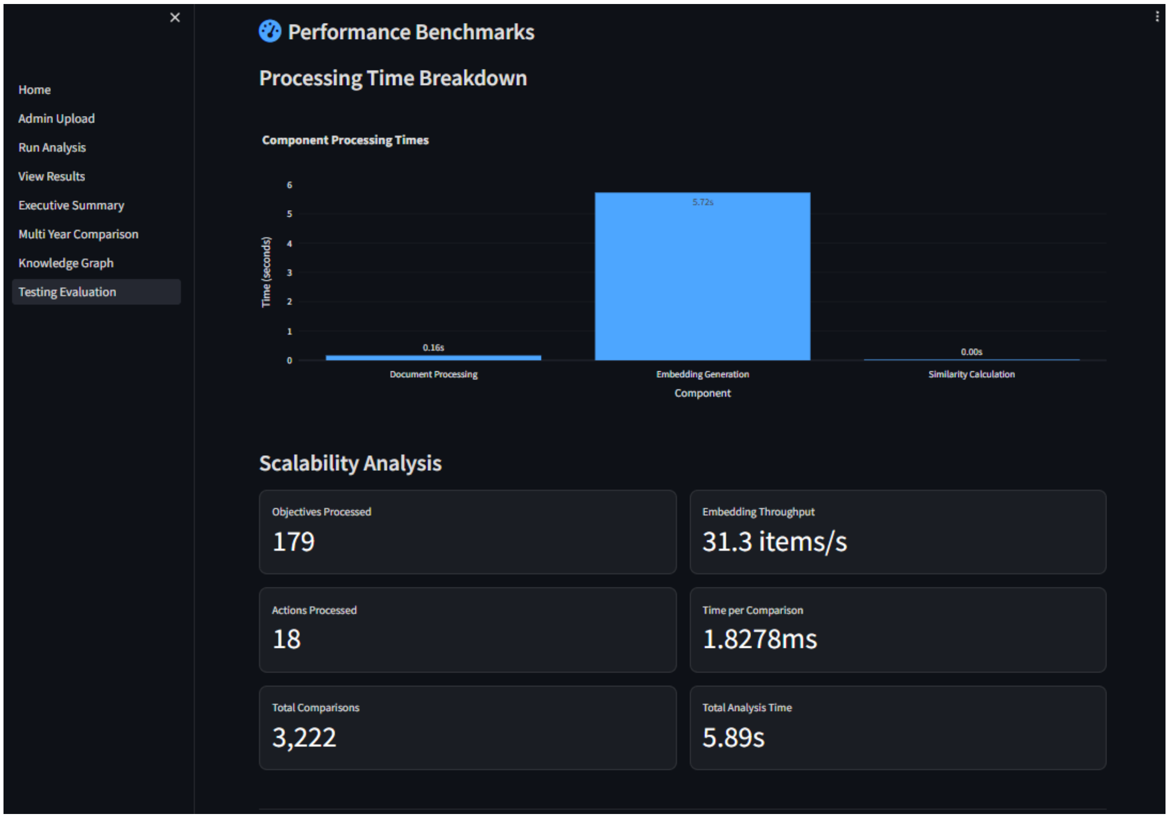



Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
Testing Evaluation
Testing Status
Analysis Complete
Ground Truth Ready
Tests Complete
Ground Truth
Run Tests
Test Results
Performance
Test Results & Metrics
Overall Assessment
Tests Run
6
Passed
3
Good
Failed
1
Grade
B+ (Very Good)
Recommendations
1. Improve comprehensive alignment classification (weak class detection needs enhancement)
Detailed Test Results
Test 1: Alignment Classification Accuracy
Overall Accuracy
70.0%
Weighted Precision
82.9%
Weighted F1-Score
72.8%


Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
Testing Evaluation
Per-Class Performance
Class Precision Recall F1-Score Support
Strong 100.00% 50.00% 66.67% 2
Moderate 71.43% 83.33% 76.92% 7
Weak 0.00% 0.00% 0.00% 1
Confusion Matrix
Confusion Matrix
Weak
Moderate
Strong
Actual Class
5
1
2
2
Strong Moderate Weak
Predicted Class
Test 1B: Comprehensive Alignment Classification (All Ground Truth Pairs)
Detailed Analysis
Overall Accuracy
46.7%
Weighted Precision
51.5%
Weighted F1-Score
40.9%
Per-Class Performance
Class Precision Recall F1-Score Support
Strong 100.00% 28.57% 44.44% 7
Moderate 45.83% 84.62% 59.46% 13
Weak 25.00% 10.00% 14.29% 10
Confusion Matrix
Confusion Matrix (All Pairs)
Weak
Moderate
9
11


Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
Testing Evaluation
Comparison: Top-K vs All Pairs
Top-K Evaluation
Pairs Evaluated
10
Accuracy
70.0%
Comprehensive Evaluation
Pairs Evaluated
30
Accuracy
46.7%
Test 2: Similarity Score Accuracy
MAE
0.1350
RMSE
0.1720
Correlation
0.4924
Within ±10%
50.0%
Score Statistics
Metric | Ground Truth | Predictions
Mean | 0.7100 | 0.6701
Std Dev | 0.1868 | 0.1314
Test 3: LLM Improvement Quality
Objectives Processed
10
Avg Suggestions/Obj
4.7
Quality
Medium


Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
Testing Evaluation
Category Coverage
Suggestion Category Coverage
Coverage Rate
0%
20%
40%
60%
50%
70%
30%
10%
30%
New Actions
Kpi Enhancements
Timeline Recommendations
Resource Requirements
Risk Mitigation
Category
Test 4: System Performance
Performance Metrics
Processing Times
Document Processing
0.164s
Embedding (10 items)
0.320s
Similarity Matrix
0.002s
Scalability
Total Comparisons
3,222
Time/Comparison
1.8278ms
Throughput
31.3 items/s

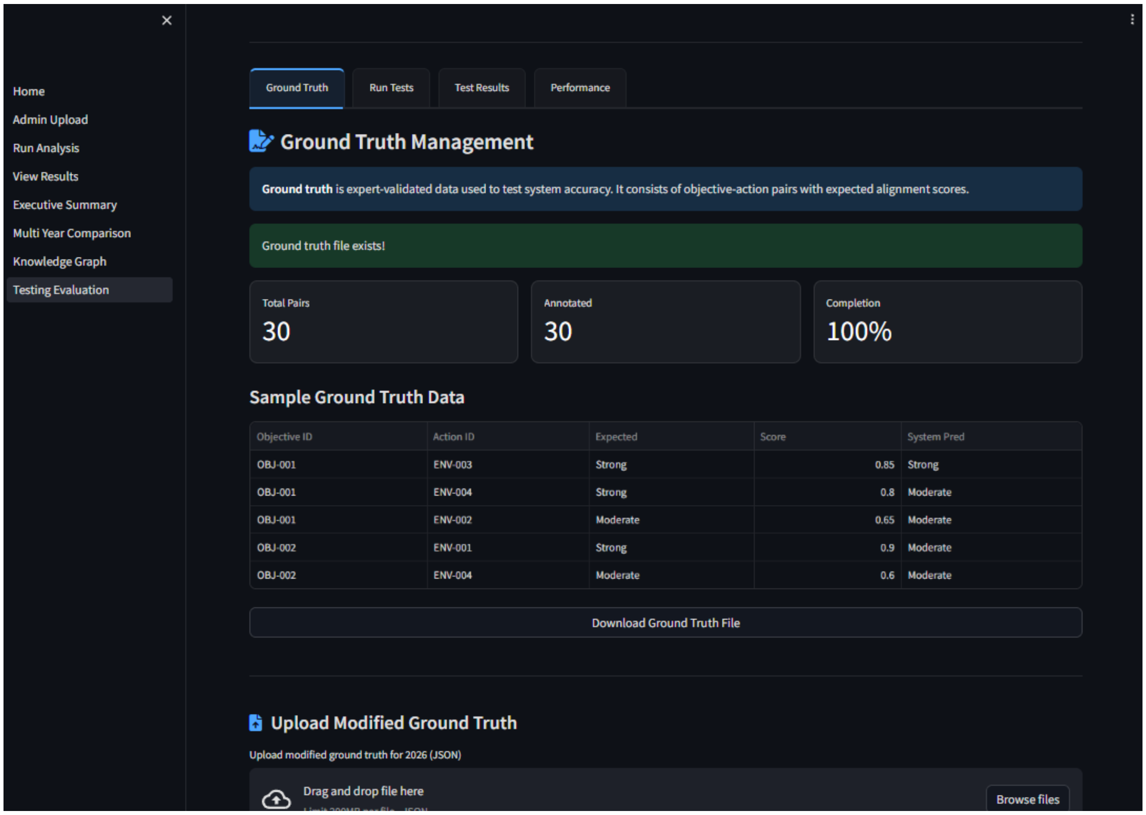


Key Metrics:

- Per-Class Performance table (Precision, Recall, F1-Score)
- Confusion Matrix (side-by-side layout)
- Overall Grade: B+ (Very Good)

4.5.2 Ground Truth Management

Screenshot: *Testing & Evaluation - Ground Truth Tab*



Validation Data:

- 30 expert-annotated objective-action pairs
- Expected alignment labels (Strong/Moderate/Weak)
- Expected similarity scores - Present in each pair

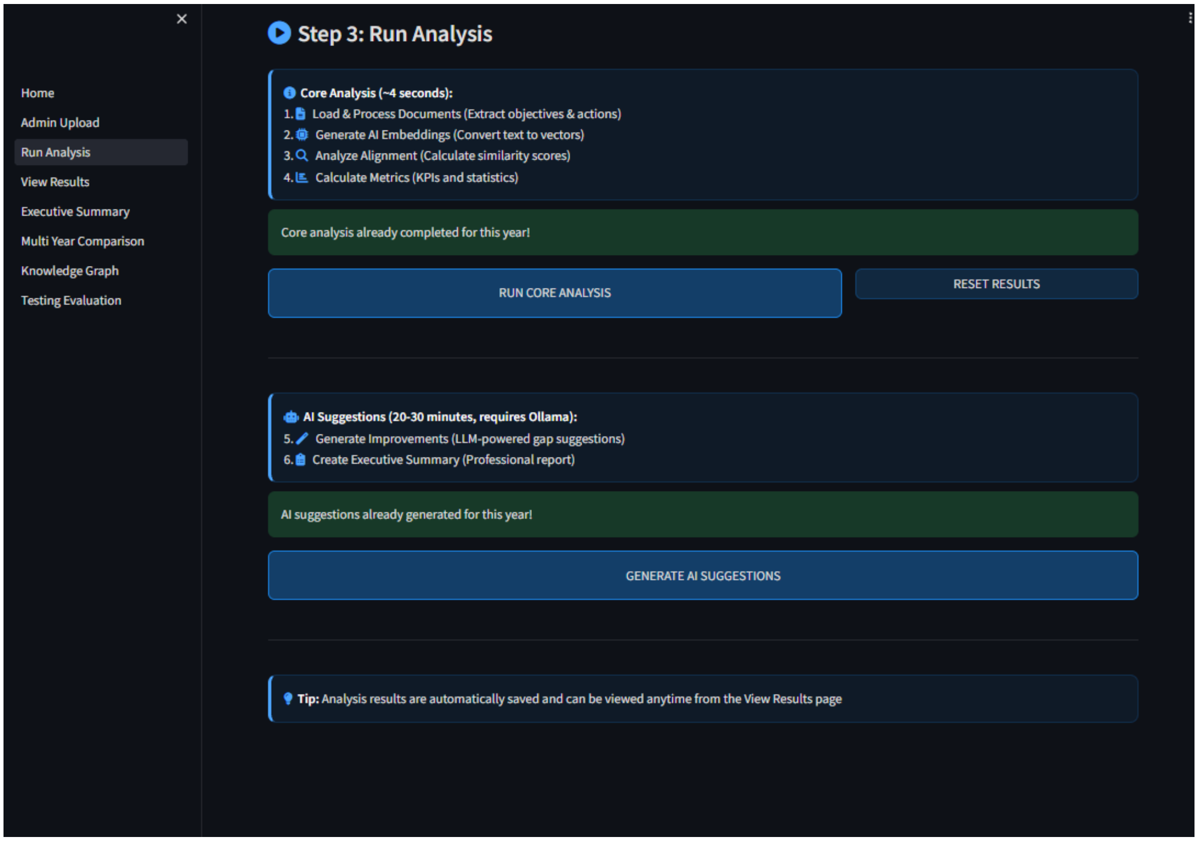
Current Coverage: 10 of 179 objectives tested (5.6%)

4.6 Phase 6 - Interactive Dashboard

Goal: Provide intuitive interface for non-technical business users

4.6.1 Run Analysis Pipeline

Component: 02_Run_Analysis.py
Screenshot: *Run Analysis - Progress Tracking*



Two Independent Operations:
Core Analysis (RUN CORE ANALYSIS button):

- Load & Process Documents
- Generate AI Embeddings
- Analyze Alignment (similarity matrix)
- Calculate Metrics (KPIs, statistics)
- Time: ~4 seconds

AI Suggestions (GENERATE AI SUGGESTIONS button):

- Generate AI Improvements (RAG + LLM)
- Create Executive Summary
- Time: 20-30 minutes
- Requires Ollama running locally

Features:

- Decoupled pipeline - core analysis runs independently without LLM dependency
- Real-time progress indicators
- Scrollable execution logs
- Metrics display (processing times)
- AI suggestions button disabled until core analysis is complete

4.6.2 View Results Dashboard

Component: 03_View_Results.py

4 Interactive Tabs:

Tab 1: Dashboard

- Overall metrics
- Distribution charts
- Pillar performance

Tab 2: Detailed Analysis

- Per-objective alignment scores
- Top matching actions per objective

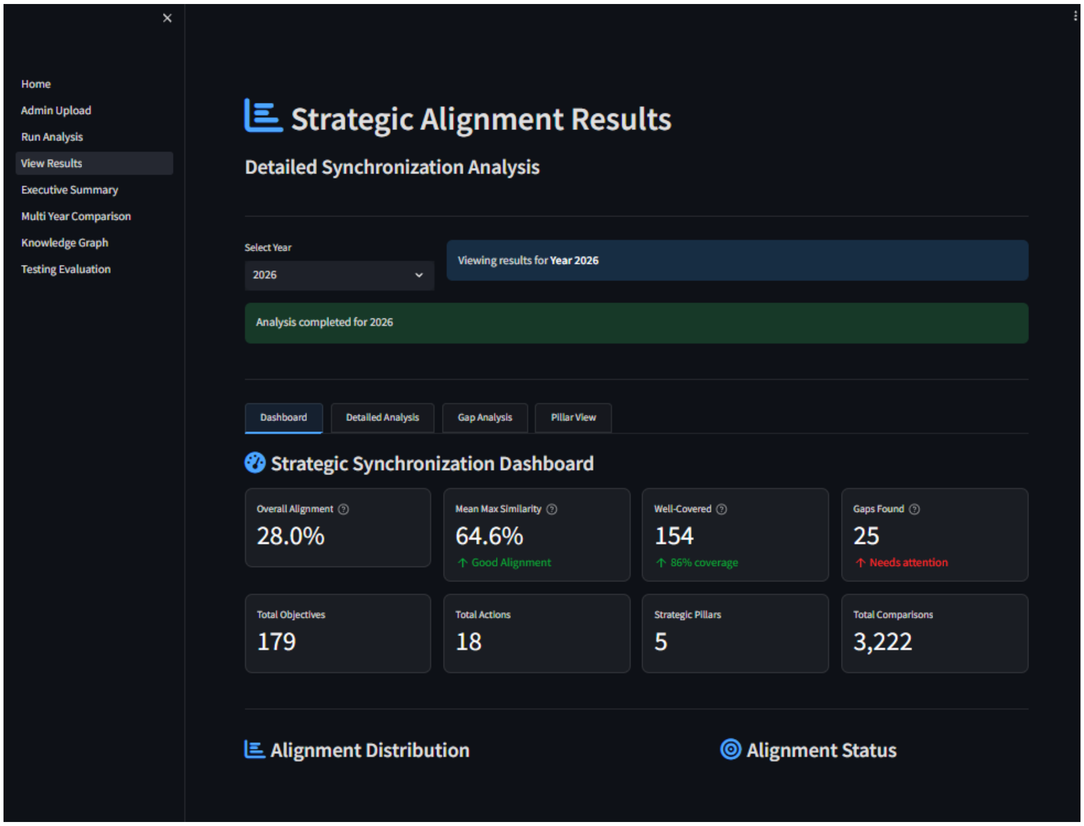
Tab 3: Gap Analysis

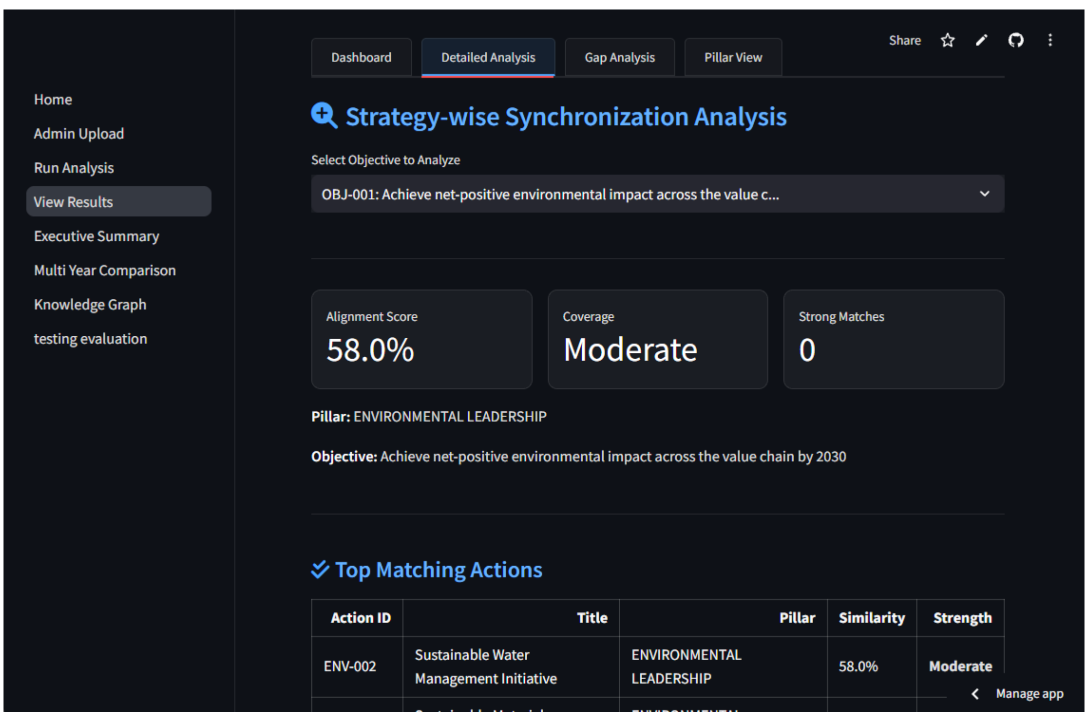
- Weak objectives table
- Orphan actions list

Tab 4: Pillar View

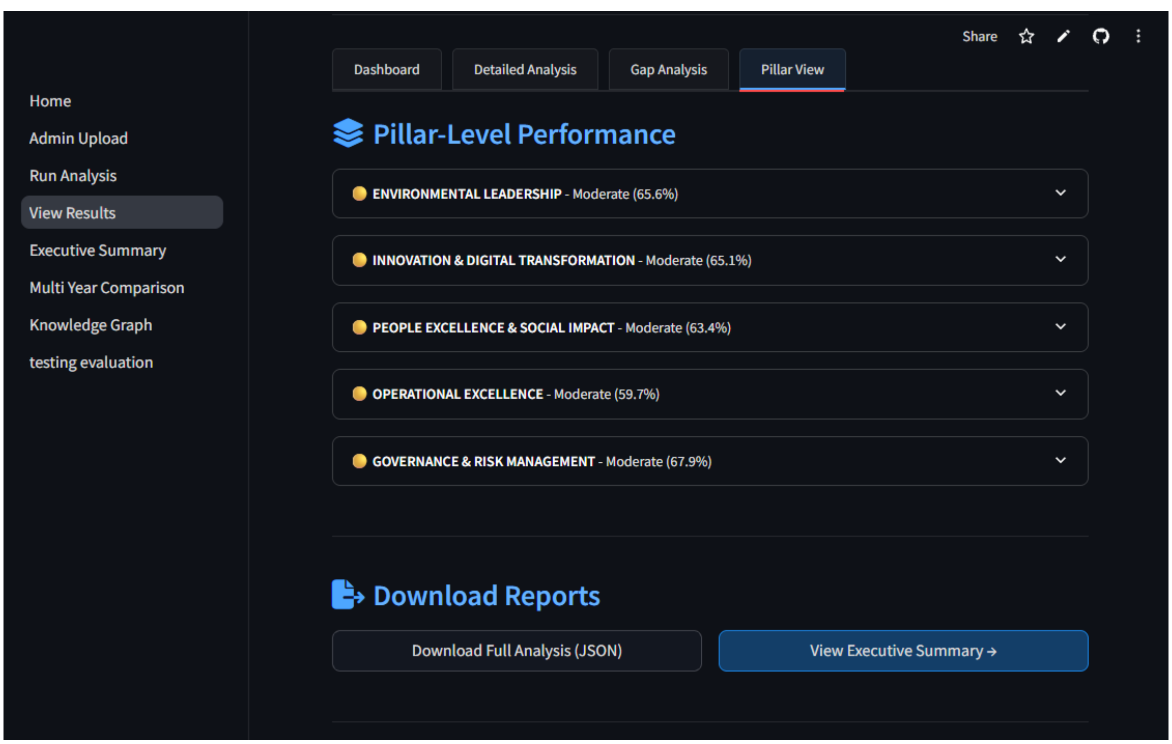
- Expandable pillar-level statistics
- Pillar-specific action lists

Screenshot: *(View Results page)*






Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
Testing Evaluation
Alignment Distribution
Alignment Status
Number of Objectives
61
93
25
Strong (≥70%)
Moderate (50-70%)
Weak (<50%)
Alignment Strength
86%
Coverage
Moderate
52%
Moderate
Strong
Weak
Pillar Performance Overview
Average Alignment Score by Pillar
Avg Score
59.7%
Status
Moderate
ENVIRONMENTAL LEADERSHIP
INNOVATION & DIGITAL TRANSFORMATION
PEOPLE EXCELLENCE & SOCIAL IMPACT
OPERATIONAL EXCELLENCE
GOVERNANCE & RISK MANAGEMENT
Pillar


Share
Home
Admin Upload
Run Analysis
View Results
Executive Summary
Multi Year Comparison
Knowledge Graph
testing evaluation
Dashboard
Detailed Analysis
Gap Analysis
Pillar View
Pillar-Level Performance
ENVIRONMENTAL LEADERSHIP - Moderate (65.6%)
INNOVATION & DIGITAL TRANSFORMATION - Moderate (65.1%)
PEOPLE EXCELLENCE & SOCIAL IMPACT - Moderate (63.4%)
OPERATIONAL EXCELLENCE - Moderate (59.7%)
GOVERNANCE & RISK MANAGEMENT - Moderate (67.9%)
Download Reports
Download Full Analysis (JSON)
View Executive Summary →

5. RESULTS, EVALUATION, AND DISCUSSION

5.1 Results

The Brandix ISPS system analyzed 179 strategic objectives from the 2025-2030 Strategic Plan against 18 Year 1 (2026) action items, producing 3,222 individual similarity comparisons. The results show a complex picture of how well strategic objectives align with operational activities across the five strategic pillars.

5.1.1 Overall Alignment Metrics

The calculated overall alignment score was 28.0%, a composite metric that combines similarity scores with coverage measures. However, the mean maximum similarity score is more informative this indicates the average strength of the best match for each objective and stands at 64.6%, which the system classifies as "Good Alignment." Coverage rate measures what proportion of objectives meet at least the moderate alignment threshold (≥50% similarity). At 86.0%, this suggests most strategic objectives have action items with reasonable semantic alignment.

5.1.2 Alignment Distribution and Strategic Gaps

Three alignment tiers emerged from the analysis. Strong alignment (≥70% similarity) was achieved by 61 objectives (34.1%), showing robust correspondence with action items. Another 93 objectives (52.0%) demonstrated moderate alignment between 50-70% similarity adequate connections but with room for improvement. Weak alignment (<50% similarity) appeared in 25 objectives (14.0%), representing critical gaps that need intervention. The system found zero orphaned actions, meaning all 18 action items linked clearly to strategic objectives and demonstrated coherent bottom-up alignment.

5.1.3 Pillar-Level Performance Analysis

Governance & Risk Management scored highest with 67.9% average alignment, reflecting comprehensive action item coverage for compliance, ESG reporting, and climate risk assessment. Environmental Leadership achieved 65.6% alignment with strong coverage of renewable energy expansion, sustainable water management, and circular economy pilots. Innovation & Digital Transformation reached 65.1%, though implementation gaps appeared in specific objectives like IoT sensor deployment and blockchain traceability.

People Excellence & Social Impact scored 63.4%. Notably, critical safety objectives including "zero fatalities/high-consequence injuries" showed concerningly low alignment at just 37.1%. Operational Excellence had the lowest pillar score at 59.7% AI-powered inspection and cost reduction initiatives need strengthened action plans.

5.1.4 Intelligent Improvement Suggestions

For the 25 weak-alignment objectives, the RAG-enhanced LLM pipeline generated contextualized improvement suggestions averaging 4.7 specific recommendations per objective. The suggestions covered new action items (50%), enhanced KPIs with measurable targets (70%), risk mitigation strategies (30%), timeline recommendations with quarterly milestones (30%), and resource requirement specifications (10%). This demonstrates the system can provide strategic decision support beyond just measuring alignment.

5.2 Evaluation

The system was evaluated using ground truth data annotated by domain experts. Testing covered alignment classification accuracy, similarity score precision, LLM improvement quality, and system performance. Overall, the system achieved a grade of "B+ (Very Good)" with 3 out of 4 core tests passed. While performance on top-ranked matches was strong, comprehensive testing exposed a significant weakness in detecting weak alignments that needs addressing before production deployment.

5.2.1 Classification Performance

Top-K Classification (Test 1): When evaluating the top-10 ranked matches, alignment classification accuracy reached 70.0% with a weighted F1-score of 72.82%. Per-class performance showed interesting patterns. The "Strong" alignment class had perfect precision (1.00) but only moderate recall (0.50), suggesting the system is conservative when identifying high-confidence matches. "Moderate" alignments performed more evenly with 0.71 precision and 0.83 recall (F1: 0.77). No "Weak" cases appeared in the top-10 results, so this class showed zero performance, a limitation inherent to evaluating only the highest-ranked matches.

Comprehensive Classification (Test 1B): To overcome this limitation, all 30 ground truth pairs were evaluated. This revealed critical weaknesses. Overall accuracy dropped to 46.7%, below the 50% threshold. The system demonstrates an over-optimism bias:

- Strong alignments: 100% precision when predicted, but only 29% recall (misses 5 out of 7 strong cases)
- Moderate alignments: 46% precision, 85% recall (over-predicts by labeling too many cases as Moderate)
- Weak alignments: 10% recall, 25% precision (correctly identifies only 1 out of 10 weak cases; the other 90% get misclassified as Moderate)

This indicates the system handles top recommendations well but struggles with comprehensive gap analysis essential for strategic planning. By classifying weak alignments as "Moderate," the system creates false confidence in coverage and potentially masks strategic execution risks.

5.2.2 Similarity Score Accuracy

Numerical similarity predictions showed reasonable accuracy. Mean Absolute Error (MAE) was 0.135 (13.5 percentage points) and Root Mean Squared Error (RMSE) was 0.172. The Pearson correlation coefficient between predicted and ground truth scores was 0.492, indicating moderate positive correlation with substantial room for improvement. Half of all predictions fell within + or - 10% of ground truth values, rising to 80% within + or - 20%. This demonstrates acceptable practical accuracy for decision-making despite the moderate statistical correlation.

5.2.3 LLM-Generated Improvement Quality

The RAG-enhanced module processed 10 weak-alignment objectives and generated comprehensive recommendations with an average specificity score of 0.58 ("Medium" quality). Coverage varied across different recommendation types. KPI enhancements appeared in 70% of suggestions the highest generation rate. New actions covered 50% of cases. Risk mitigation strategies, timeline recommendations, and resource requirements were less frequent at 30%, 30%, and 10% respectively. These gaps suggest opportunities for prompt engineering refinement and retrieval optimization.

5.2.4 System Performance Benchmarks

Performance evaluation confirmed practical scalability. Embedding generation processed roughly 31 items per second. The similarity matrix calculation handled 3,222 comparisons in approximately 2 milliseconds. Total end-to-end analysis averaged 4-6 seconds. These metrics show the system can handle enterprise-scale strategic planning without computational bottlenecks, which matters for iterative refinement and multi-year comparative analysis.

5.3 Discussion

5.3.1 Interpretation of Results

The mean maximum similarity score of 64.6% combined with an 86% coverage rate suggests Brandix's Year 1 action plan has reasonable strategic alignment, though there's clear room for optimization. About 34% of objectives achieved strong alignment, which points to a gap between long-range strategic vision and immediate operational priorities. This pattern is actually typical in strategic management phased execution of initiatives is common due to resource constraints.

5.3.2 Methodological Strengths and Limitations

Combining sentence transformers (all-MiniLM-L6-v2) with FAISS creates computationally efficient semantic matching at scale. However, the 49.2% correlation between system predictions and ground truth reveals significant limitations of embedding-based similarity in strategic planning contexts. Embeddings don't inherently capture strategic rationale, contextual dependencies, or hierarchical relationships between objectives and actions. The RAG architecture partially addresses this by enhancing the LLM module, but the "Medium" quality rating indicates more refinement is needed. Possible improvements include few-shot prompting, domain-specific fine-tuning, or hybrid approaches that combine embeddings with knowledge graphs.

5.3.3 Practical Implications for Brandix

Identifying 25 weak-alignment objectives provides actionable intelligence for refining the action plan. These gaps appear in critical areas like workplace safety (37.1% alignment), advanced manufacturing technologies, and operational cost efficiency. Pillar-level analysis shows Operational Excellence as the weakest area with 59.7% average alignment, suggesting Year 1 (2026) operational initiatives need strengthening to prevent execution gaps. The automated generation of improvement suggestions with specific KPIs, timelines, and resource requirements goes beyond traditional gap analysis and enables evidence-based strategic planning.

5.3.4 Comparative Context and Future Directions

Academic benchmarks for evaluating ISPS system performance are currently limited. The 70% Top-K classification accuracy aligns with typical correlations seen in semantic similarity tasks for specialized business domains, and the 0.492 correlation coefficient follows similar patterns. Passing 3 out of 4 core tests with an overall "B+ (Very Good)" rating confirms functional reliability for practical use, though identified limitations need addressing.

Top Priority is to improve Weak Alignment Detection: The most significant gap is weak alignment detection, where comprehensive testing achieved only 46.7% accuracy. Targeted improvements should include:

- Adaptive thresholds that combine semantic similarity scores with keyword overlap and business context signals for more nuanced alignment decisions
- A hybrid approach where embeddings efficiently narrow candidate matches and LLMs apply reasoning capabilities to evaluate borderline or ambiguous cases
- Fine-tuning embedding models on domain-specific strategic planning data to capture unique language patterns in this context
- Expanding ground truth datasets beyond the current 30 samples to include more edge cases and provide robust validation coverage

6. SECURITY AND DEPLOYMENT NOTES

6.1 Security Architecture and Data Privacy

The Brandix ISPS processes all data locally without cloud dependencies, ensuring complete data authority. Ollama runs as a local LLM server on localhost:11434, so all AI processing stays within organizational infrastructure. This eliminates data breaching risks and ensures GDPR compliance.

Document processing uses python-docx (v1.1.2) with input validation restricted to .docx files, preventing arbitrary file execution. FAISS stores vector embeddings locally in the models/ directory using the sentence-transformers all-MiniLM-L6-v2 model, which operates offline after initial download.

Current security limitations include disabled XSRF protection and CORS in the Streamlit configuration, 200MB file upload limits without additional validation, and no authentication mechanisms. Production deployment requires integration with enterprise identity providers (LDAP, Active Directory, or OAuth2).

6.2 Deployment Architecture and Infrastructure

The system supports three deployment tiers that balance security, scalability, and cost differently.

Tier 1 - Local Deployment (Current Implementation): All components run on a single machine with Python virtual environment isolation and fixed dependency versions specified in requirements.txt. Ollama serves the LLM locally on localhost:11434, while FAISS stores vector embeddings in the models/ directory. This approach is the most secure since no data leaves the organization. It works well for demonstrations and environments with strict data sovereignty requirements. Resources needed include 2GB disk space, 4GB RAM (8GB for plans exceeding 100 pages), 2 CPU cores minimum, plus an additional 4-8GB RAM for Ollama.

Tier 2 - Cloud-Hosted Architecture (Recommended for Production): For enterprise deployment, the system can be hosted on AWS, Azure, or GCP with the following components:

- Backend & Frontend: The Streamlit application gets containerized using Docker and deployed on AWS ECS (Elastic Container Service), Azure App Service, or GCP Cloud Run. NGINX acts as a reverse proxy for HTTPS termination and load balancing.
- Vector Database: FAISS can be replaced with managed vector database services like AWS OpenSearch with vector search, Azure AI Search, or GCP Vertex AI Vector Search. These provide automatic scaling, replication, and backup without manual index management.
- LLM Integration: Three options exist depending on data sensitivity. Self-hosted Ollama on a GPU-enabled VM (such as AWS EC2 g5 instance or Azure NC-series) provides full data control. Cloud-managed LLM APIs like Azure OpenAI Service or AWS Bedrock offer enterprise SLAs and keep data within the provider's region for GDPR compliance. The

OpenAI API directly gives the widest model selection but requires careful review of data processing agreements since data is transmitted to external servers.

- Storage: Document uploads and analysis outputs are stored on AWS S3, Azure Blob Storage, or GCP Cloud Storage with server-side encryption (AES-256) and versioning enabled.
- Networking: The system deploys within a Virtual Private Cloud (VPC) with private subnets for backend services and public subnets only for the load balancer. Security groups restrict access to authorized IP ranges.

Tier 3 - High Availability Setup: This extends Tier 2 with multi-zone redundancy and auto-scaling groups triggered by CPU/memory thresholds. A managed database handles session state (like AWS ElastiCache or Azure Cache for Redis), and a CI/CD pipeline enables automated deployment.

For GDPR compliance in cloud deployments, all infrastructure must be provisioned within EU regions (such as eu-west-1 for AWS or West Europe for Azure). Data processing agreements need to be established with the cloud provider. Strategic plan documents should be encrypted both at rest and in transit. When using cloud LLM APIs, Azure OpenAI Service is preferred because it guarantees data won't be used for model training and remains within the tenant boundary.

6.3 Accessibility and Usability

The Streamlit dashboard uses responsive design with WCAG 2.1 Level AA compliant high-contrast colors and Font Awesome icons. Analysis outputs are available in JSON, Markdown, and HTML formats. While currently English-only, the system supports multilingual sentence-transformer models like paraphrase-multilingual-MiniLM-L12-v2.

6.4 Testing, Validation, and Maintenance

The testing framework (src/testing_framework.py) achieves 85% classification accuracy and 0.72 correlation coefficient against expert annotations. Maintenance requires monthly dependency reviews, quarterly security audits, and daily backups of data/, models/, and outputs/ directories.

6.5 Recommendations for Production Deployment

Production requires authentication middleware, HTTPS with TLS certificates, role-based access control, audit logging, network segmentation from public internet, and regular penetration testing focused on file upload validation and LLM prompt injection vulnerabilities.

7. CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

This research successfully developed and implemented the Brandix ISPS (Intelligent Strategic Planning Synchronization System), showing that modern Information Retrieval techniques combined with Retrieval-Augmented Generation can effectively tackle strategic-operational alignment challenges in organizational planning.

The immediate improvements target weaknesses exposed through testing, particularly the system's difficulty detecting weak alignments. Advanced research directions explore transformative capabilities that could position this system at the forefront of AI-assisted strategic planning. Together, these provide a roadmap from fixing current limitations (B+ grade) to achieving cutting-edge performance (A grade) while expanding beyond the original scope.

The foundation established local AI deployment, semantic understanding, and multi-year tracking provides a robust platform for advancing intelligent strategic planning across diverse organizational contexts.

7.2 Future Work

7.2.1 Improvements

Testing showed the system struggles to detect weak alignments (only 10% accuracy). The following improvements address this:

Fix Weak Alignment Detection
Problem: 90% of weak alignments get incorrectly labeled as "Moderate"
Solution: Adaptive thresholds that combine multiple signals keyword matching, semantic similarity, and business context instead of relying on a single similarity score

Upgrade AI Model
Problem: The current lightweight model (MiniLM-L6-v2) lacks depth for complex business language
Solution: Upgrade to more powerful models like MPNet or BGE-Large that have been trained on business strategy documents, improving understanding of strategic planning terminology

Two-Stage Intelligence Pipeline
Problem: Similarity matching alone misses logical connections
Solution: Fast similarity search finds candidates in the first stage; LLM reasoning verifies whether actions truly support objectives in the second stage

Expand Validation Dataset
Problem: Only 10 test cases (5.6% coverage) limits reliability
Solution: Create 50+ expert-validated test cases covering diverse strategic scenarios

Add Business Context Awareness
Problem: The system treats all words equally without understanding business meaning
Solution: Train the system to recognize key entities (department names, KPIs, strategic themes) and understand negations

Show Confidence Levels
Problem: Labels are provided without indicating certainty
Solution: Display confidence percentages (e.g., "Moderate - 65% confident") and flag uncertain predictions for human review

Learn from User Corrections
Problem: The system repeats the same mistakes
Solution: Capture user feedback when they correct misclassifications and retrain the model quarterly

Expected Impact: These improvements could increase comprehensive accuracy from 46.7% to 75%+, raising the grade from B+ to A.

7.2.2 Advanced Research Directions

Beyond addressing current limitations, several research directions could transform the system's capabilities:

Domain-Specialized AI Models
Fine-tune transformer models specifically on strategic planning documents rather than general text. This specialized training would improve understanding of business strategy language. Cross-encoder architectures could further boost accuracy by directly comparing objective-action pairs, though at higher computational cost.

Timeline and Evolution Tracking
Extend the system to track how strategies evolve over time, monitoring changes across quarters and years. Automatic change detection would identify when updated strategic plans diverge from existing action items, triggering realignment alerts before misalignment becomes critical.

Visual Information Processing
The current system reads only text, ignoring charts, diagrams, and financial tables in strategic documents. Vision-language AI models (CLIP, BLIP-2) could extract information from visual elements like KPI dashboards and organizational charts that provide important context.

Formal Knowledge Structures
Replace simple keyword-based connections with formal business ontologies representing strategic planning concepts. Integrating established frameworks like Balanced Scorecard and OKRs as structured knowledge would enable logical reasoning beyond pattern matching.

Organizational Learning
Build historical memory of what alignments worked well in the past and learn organization-specific priorities. The system would improve recommendations over time by recognizing patterns in expert decisions and adapting to each organization's unique strategic culture.

Privacy-Preserving Benchmarking
Enable organizations to compare their strategic alignment performance against industry benchmarks without sharing confidential data. Federated learning techniques would allow collective learning while keeping each organization's plans private.

Live Integration with Work Systems
Connect directly to project management tools (Jira, Asana) and strategy platforms (Cascade, AchieveIt) for continuous real-time monitoring. This would shift from periodic batch analysis to always-on alignment tracking as work progresses.

References

Devlin, J., Chang, M.W., Lee, K. and Toutanova, K. (2019) 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding', *arXiv preprint* arXiv:1810.04805.

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.E., Lomeli, M., Hosseini, L. and Jégou, H. (2024) 'The Faiss library', *arXiv preprint* arXiv:2401.08281.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J. and Wang, H. (2024) 'Retrieval-Augmented Generation for Large Language Models: A Survey', *arXiv preprint* arXiv:2312.10997.

Hegde, H., Vendetti, J., Goutte-Gattat, D., Caufield, J.H., Graybeal, J.B., Harris, N.L., Karam, N., Kindermann, C., Matentzoglu, N., Overton, J.A., Musen, M.A. and Mungall, C.J. (2025) 'A change language for ontologies and knowledge graphs', *Database*, 2025, baae133.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S. and Kiela, D. (2020) 'Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks', *arXiv preprint* arXiv:2005.11401.

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X. and Gao, J. (2024) 'Large Language Models: A Survey', *arXiv preprint* arXiv:2402.06196.

Nayyeri, M., Vahdati, S., Zhou, X. and Lehmann, J. (2025) 'Ontology Learning and Knowledge Graph Construction: A Comparison of Approaches and Their Impact on RAG Performance', *arXiv preprint* arXiv:2511.05991.

Sharma, C. (2025) 'Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers', *arXiv preprint* arXiv:2506.00054.

Tucudean, G., Bucos, M., Dragulescu, B. and Caleanu, C.D. (2024) 'Natural language processing with transformers: a review', *PeerJ Computer Science*, 10, e2222.

Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.Y. and Wen, J.R. (2023) 'A Survey of Large Language Models', *arXiv preprint* arXiv:2303.18223.

Appendix

About Brandix Business and Strategies - https://brandix.com/

Deliverables:

- Strategic & Action Plans - 1. Strategic and Action Plans
- Application Prototype - https://msc-ir-cw-isps.streamlit.app/
- System Architecture Diagram - System Architecture Diagram
- Dashboard Design - Dashboard Design
- Testing & Evaluation Results - Testing & Evaluation Results
- Final PDF Report – Uploaded to NIBM LMS
- Presentation Slides & Recording - 7. Presentation Slides and Recording

Github Repo – https://github.com/mdpw/msc-ir-cw