

Matching Algorithm Optimization Report

Project: Mentora – University Collaboration & Project Matching Platform

Assignment - Aiproff.ai

1. Introduction

Mentora is a platform designed to match students with faculty members, industry mentors, and projects based on skills and interests. The quality of these matches directly impacts collaboration outcomes and user satisfaction.

This report evaluates an existing **rule-based matching algorithm**, establishes its baseline performance, and demonstrates an **optimized semantic matching approach** using modern embedding-based techniques.

2. Baseline Matching Approach

2.1 Description

The baseline algorithm computes match scores using **exact keyword overlap** between:

- Student skills ↔ Faculty expertise
- Student interests ↔ Faculty research areas

The score is calculated as:

$$\text{Match Percentage} = \frac{\text{Number of common keywords}}{\max(\text{keywords in profile A}, \text{keywords in profile B})} \times 100$$

The final match score is the average of skill match and interest match percentages.

2.2 Strengths

- Fast and deterministic
- Easy to interpret and debug
- No external dependencies
- Suitable for small datasets

2.3 Limitations

- Fails to capture semantic similarity (e.g., “ML” vs “Machine Learning”, “AI” vs “Artificial Intelligence”)
 - Treats all skills equally without importance weighting
 - Produces low scores for strong real-world matches
 - Not robust to variations in terminology
-

3. Baseline Performance Evaluation

3.1 Dummy Data Setup

To evaluate the algorithm, realistic dummy profiles were created for:

- Students (skills + interests)
- Faculty members (expertise + research areas)

The data intentionally included:

- Synonyms and abbreviations
- Domain-related terminology
- Exact and near-exact matches

3.2 Observations

- Strong semantic matches often received low or zero scores
- Exact keyword matches performed reasonably well
- Overall match quality did not align with human intuition

This established the baseline performance and highlighted the need for optimization.

4. Optimized Matching Approach

4.1 Motivation

To overcome the limitations of lexical matching, a **semantic similarity-based approach** was introduced.

This allows the system to understand relationships between related terms rather than relying on exact string equality.

4.2 Methodology

- A pre-trained **sentence embedding model** was used to convert skills and interests into vector representations
- **Cosine similarity** was computed between student and faculty embeddings
- The maximum similarity score was taken as the semantic match score
- Weighted scoring was applied:
 - Skills: 70%
 - Interests / research areas: 30%

This design balances relevance while preserving explainability.

5. Optimized Performance Evaluation

5.1 Results

Compared to the baseline:

- Semantic matches were correctly identified
- Near-synonyms and related concepts scored highly
- Overall match scores increased significantly across most profile pairs

5.2 Quantitative Improvement

Key improvements observed:

- Substantial increase in average match scores
- Large absolute lift for previously under-scored strong matches
- Improved alignment with human judgment

The optimized approach consistently outperformed the baseline on all evaluated scenarios.

6. Comparison Summary

<u>Aspect</u>	<u>Baseline</u>	<u>Optimized</u>
Matching method	Exact keyword overlap	Semantic similarity
Synonym handling	No	Yes
Match quality	Low–Medium	High
Explainability	High	Medium
Real-world robustness	Low	High

7. Conclusion

The baseline matching algorithm provided a solid and interpretable starting point but failed to handle real-world variations in terminology.

By integrating embedding-based semantic matching, the optimized approach achieved a significant improvement in match quality while remaining modular and extensible.

This optimization demonstrates how LLM-assisted techniques can be effectively applied to improve traditional rule-based systems in practical applications like academic and mentorship matching.

8. Future Improvements

- Potential enhancements include:
- Caching embeddings for scalability
 - Hybrid exact + semantic matching
 - Skill importance normalization
 - Threshold-based recommendations
 - Offline batch processing for large datasets