JobSync Optimization Project

This document provides a comprehensive explanation of the mathematical formulation for the JobSync job matching system optimization models.

Data Overview

The optimization uses three datasets:

1. **Seekers Data**: Information about job seekers (ID, location, desired job type, etc.)
2. **Jobs Data**: Information about job openings (ID, location, requirements, priority weights, etc.)
3. **Location Distances**: Matrix showing distances between locations A through F

Problem Formulation

Sets and Indices

- I: Set of all job seekers, indexed by i
- J: Set of all job openings, indexed by j
- P_j: Number of available positions for job j
- w_j: Priority weight of job j (1-10)

Decision Variables

For both models, we use the binary decision variable:

- $x_{ij} = \begin{cases} 1, & \text{if seeker i is assigned to job j} \ 0, & \text{otherwise} \end{cases}$

For Part 2, we also use the continuous variable:

- z: Maximum dissimilarity score among all matched pairs

Part 1: Maximize Priority-Weighted Matches

Objective Function

Maximize the sum of priority weights of filled positions:

$\text{Maximize} \sum_{i \in I} \sum_{j \in J} w_j \cdot x_{ij}$

Constraints

1. **Each seeker can be assigned to at most one job**: $\sum_{j \in J} x_{ij} \leq 1, \quad \forall i \in I$
2. **Number of seekers assigned to a job cannot exceed its available positions**: $\sum_{i \in I} x_{ij} \leq P_j, \quad \forall j \in J$

3. **Compatibility requirements**: $x_{ij} = 0, \quad \forall(i,j) \text{ where seeker } i \text{ and job } j \text{ are incompatible}$
   A seeker i and job j are considered compatible only if ALL of the following conditions are met:
   - Job Type: Seeker's desired type matches the job's type
   - Salary: Seeker's minimum desired salary is less than or equal to the job's maximum offered salary
   - Skills: Seeker possesses all skills required by the job
   - Experience: Seeker's experience level meets or exceeds the job's requirement
   - Location: The job is remote OR the distance between seeker's location and job's location is within the seeker's maximum commute distance

This compatibility check is implemented in the preprocessing phase, where we only create decision variables for compatible seeker-job pairs.

Part 2: Minimize Maximum Dissimilarity

Dissimilarity Score

The dissimilarity score d_{ij} between a potentially compatible seeker i and job j is calculated as:

$d_{ij} = \frac{\sum_{k=1}^{20} |q_{i,k} - q_{j,k}|}{20}$

where:

- $q_{i,k}$: Seeker i's response to question k (0-5)
- $q_{j,k}$: Job j's value for question k (0-5)

Objective Function

Minimize the maximum dissimilarity score among all matched pairs:

$$\text{Minimize } z$$

where z represents the maximum dissimilarity score.

Constraints

1. **Maximum dissimilarity definition**: $z \geq d_{ij} \cdot x_{ij}, \quad \forall i \in I, j \in J$ This constraint ensures that z is at least as large as the dissimilarity score of any matched pair. When $x_{ij} = 0$ (no match), the constraint becomes $z \geq 0$, which is always satisfied.
2. **Each seeker can be assigned to at most one job** (same as Part 1): $\sum_{j \in J} x_{ij} \leq 1, \quad \forall i \in I$
3. **Number of seekers assigned to a job cannot exceed its available positions** (same as Part 1): $\sum_{i \in I} x_{ij} \leq P_j, \quad \forall j \in J$
4. **Minimum priority weight requirement**: $\sum_{i \in I} \sum_{j \in J} w_j \cdot x_{ij} \geq \omega \cdot M_w / 100$ Where:
   - M_w is the maximum priority weight found in Part 1
   - ω is the percentage parameter (70, 75, 80, 85, 90, 95, or 100)

5. **Compatibility requirements** (same as Part 1): $x_{ij} = 0, \quad \forall (i,j) \text{ where seeker } i \text{ and job } j \text{ are incompatible}$

Implementation Approach

Preprocessing

1. Load and parse the three CSV files
2. Convert string representations of lists (Skills, Questionnaire) to actual lists
3. Identify all compatible seeker-job pairs based on the five compatibility criteria
4. Calculate dissimilarity scores for all compatible pairs

Solving Part 1

1. Create decision variables for each compatible seeker-job pair
2. Set the objective to maximize the sum of priority weights
3. Add constraints for seeker assignment limits and job capacity
4. Solve the model to find the maximum priority weight M_w

Solving Part 2

1. Create decision variables for each compatible seeker-job pair
2. Add a continuous variable z for the maximum dissimilarity
3. Set the objective to minimize z
4. Add constraints:
   - Define z as the maximum dissimilarity
   - Seeker assignment limits
   - Job capacity limits
   - Minimum priority weight (based on ω)
5. Solve the model for each value of ω
6. Analyze the trade-off between priority weight and dissimilarity

Trade-off Analysis and Recommended ω Value

After running the model with different ω values, we can analyze:

1. How the minimum achievable maximum dissimilarity changes with ω
2. The trade-off between priority weight and dissimilarity
3. The point of diminishing returns (where increasing ω leads to a disproportionate increase in dissimilarity)

Based on this analysis, we can recommend an appropriate ω value that balances priority weight achievement and dissimilarity minimization.