GA-NSA

1. **Data Reading**: The code reads data from several files. These files contain instructions, training data, and test data. The instructions include parameters like the radius of self points (R), the number of detectors (M), and the number of epochs for the Genetic Algorithm.
2. **Data Structures**: The code defines several lists to store different types of points (self\_pt, detector, test\_pt, all\_pt) and their properties (detector\_rad, test\_label). These points represent different types of data in the model.
3. **Distance Calculation**: The function distance(x1, x2) calculates the Minkowski distance of order one between two points. This is used to measure the similarity between different topics.
4. **Radius Calculation**: The function calcR(x) calculates the radius of a potential detector point. It checks if the issue lies within the range of any detector point and calculates the radius accordingly.
5. **Genetic Algorithm (GA)**: A class GA is defined for the Genetic Algorithm. It includes an initialisation method that generates a population of potential solutions and a mutation method that performs mutation on the people.
6. **Negative Detectors**: The create\_negative\_detectors function creates negative detectors using the Genetic Algorithm. It generates a population, calculates the cost for each individual in the population, performs crossover operations, and returns the best two individuals and their costs.
7. **Detector Population**: The populate\_detectors function populates the detectors by repeatedly calling create\_negative\_detectors until it has created a specified number of detectors (M). It appends the best detector from each call to create\_negative\_detectors to the list of detectors.
8. **Decision Function**: The calcDecR function calculates whether a given point is within the radius of any detector. It returns False if the point is within any detector’s radius and True otherwise.
9. **Testing**: The test function tests the performance of the detectors on a test set. It first populates the detectors; then, for each point in the test set, it uses calcDecR to predict whether the point is within any detector’s radius. It compares this prediction to the actual label of the point to calculate accuracy.