DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING | IIT (ISM) DHANBAD

EVOLUTIONARY COMPUTATION

GROUP PROJECT: INTENT CLASSIFICATION

Course Code: CSD405

Professor : Associate Professor ACS Rao

Session : 2023-24

Group : 9

PROBLEM STATEMENT

- Taking data from any popular social media apps like FB/Twitter/Amazon ,where data is in Hinglish format (English and Hindi combined)
- Prepare models for intent classification and train them using feature Selection Algorithms
- Perform intent classification using any of the Evolutionary Algorithm

WHAT IS INTENT CLASSIFICATION ?

Are you using Intent classification?

Answer is yes, if you have ever used chatbots or any virtual assisstants like siri or google assistant.

Let's consider an example scenario: You're interacting with Siri or Google Assistant on your smartphone.
When you ask Siri to "Set an alarm for 7 AM," it recognizes your intent as "Setting Alarm" and schedules the alarm accordingly. So Behind the scenes, the system needs to understand that your intent is to set an Alarm irrespective of language.

So we can conclude that in intent classification, the goal is to categorize a piece of text into predefined
categories or labels that represent the different intentions or purposes behind the text and that is what we will be doing in this project.

PROJECT TIMELINE

01

DATA COLLECTION

- Collect Hinglish data from social media platforms like Facebook, Twitter, or Amazon.
- Manually label or categorize the intents for each query/message.

02

DATA PREPROCESSING

Social media data often contains noise such as special characters, emojis, URLs, and irrelevant information. Preprocessing helps remove this noise to focus on the relevant text content.

03

MODEL DEVELOPMENT

- Develop and train the intent classification model using the collected and preprocessed data.
- Perform cross-validation to evaluate the model's performance and make necessary adjustments.

04

FEATURE SELECTION

- Feature selection using evolutionary algorithms (EAs) for intent classification involves using evolutionary optimization techniques to select the most relevant features from the input data to improve the performance of intent classification models.

05

MODEL EVALUATION AND TESTING

- Evaluate the final model on a separate test dataset to assess its generalization performance.
- Calculate relevant evaluation metrics such as accuracy, precision, recall, and F1-score.
- Conduct error analysis to identify areas for improvement.

DATA COLLECTION

- **We have collected the dataset such that it will be having each query/message be labeled or categorized with its corresponding intent or purpose, such as "Shopping Inquiry," "Customer Support," "Feedback," etc.**
- **Given Below is a Glimpse of our Dataset having two columns :**

Query **Intent**

	A	B
1	Text	intent
2	Mujhe apna account mein kya hoga?	Customer Support
3	Mujhe apna password change karna hai.	Customer Support
4	Mujhe apna order status check karna hai.	Shopping Inquiry
5	Mujhe apna product return karna hai.	Customer Support
6	Mujhe apna product warranty details check karna hai?	Technical Assistance
7	Mujhe apna product feedback dena hai.	Feedback
8	Mujhe apna product price compare karna hai.	Shopping Inquiry
9	Mujhe apna product delivery date check karna hai?	Shopping Inquiry
10	Mujhe apna product return policy check karna hai?	Technical Assistance
11	Mujhe apna product warranty claim karna hai.	Technical Assistance
12	Mujhe apna product quality feedback dena hai.	Feedback
13	Mujhe apna product delivery location change karna hai.	Shopping Inquiry
14	Mujhe apna product return reason check karna hai?	Technical Assistance
15	Mujhe apna product price drop notification check karna hai?	Shopping Inquiry

DATA PREPROCESSING

The following data preprocessing we have applied on our dataset before model training:

1. **Noise Removal**: Social media data often contains noise such as special characters, emojis, URLs, and irrelevant information. Preprocessing helps remove this noise to focus on the relevant text content.

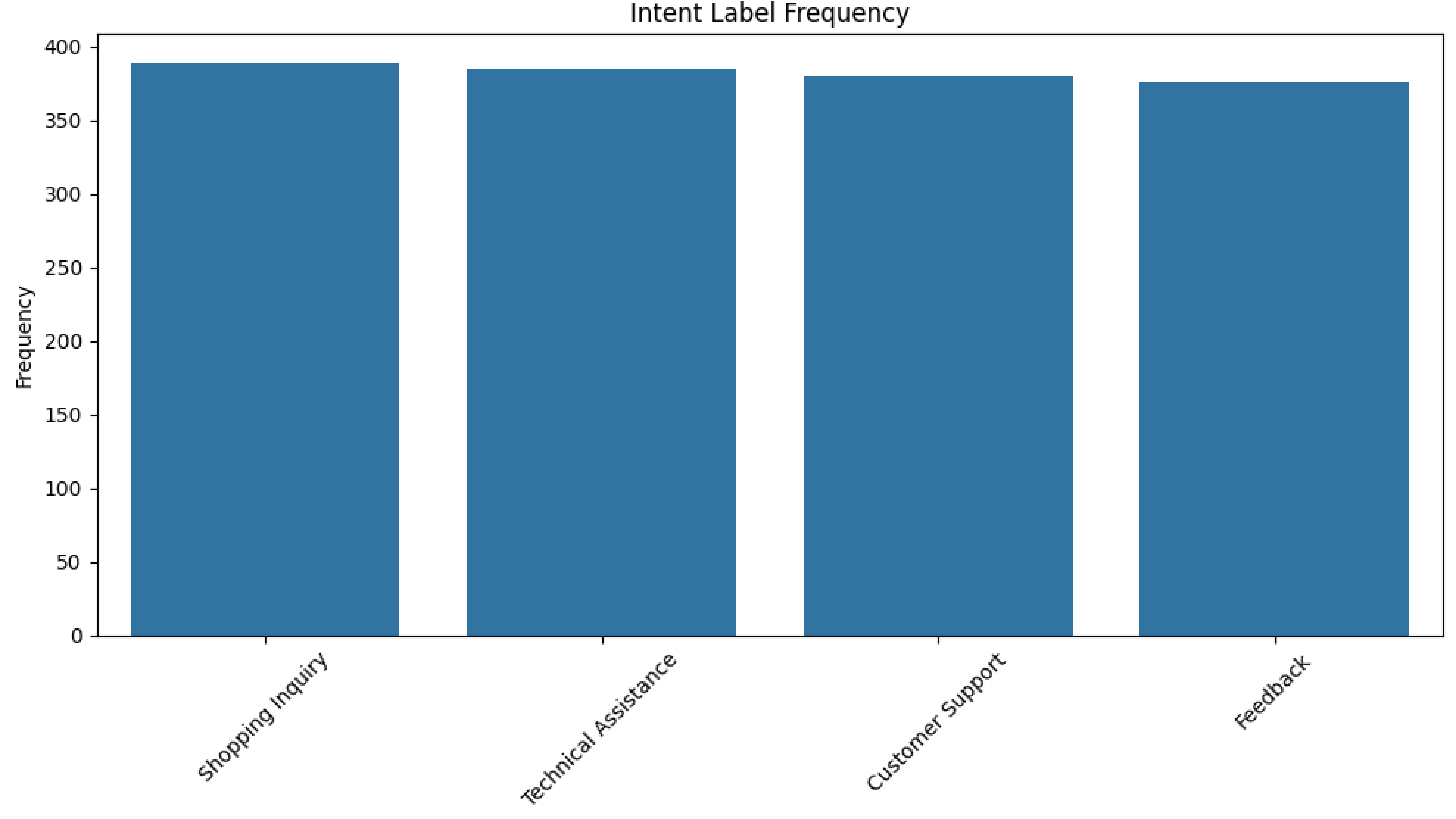
2. **Tokenization:** Tokenization involves breaking down the text into individual words or tokens. This step is necessary for further analysis, as it allows you to examine the text at a more granular level.

3. **Normalization:** Normalizing the text involves converting all text to lowercase, which helps ensure consistency in the data. It prevents the model from treating words like "hello" and "Hello" as different tokens.

4. **Handling Multilingual Text**: Since your data contains Hinglish (a combination of English and Hindi), preprocessing may involve identifying and separating English and Hindi words, as well as handling transliteration and code-switching between languages.

But what benefits are we getting from data preprocessing:

Data preprocessing plays a crucial role in improving the performance, efficiency, and interpretability of models in NLP tasks like intent classification. By cleaning and structuring the text data, preprocessing enables the model to learn effectively from the data and make accurate predictions.


Intent Label Frequency
Frequency
0
50
100
150
200
250
300
350
400
Shopping Inquiry
Technical Assistance
Customer Support
Feedback

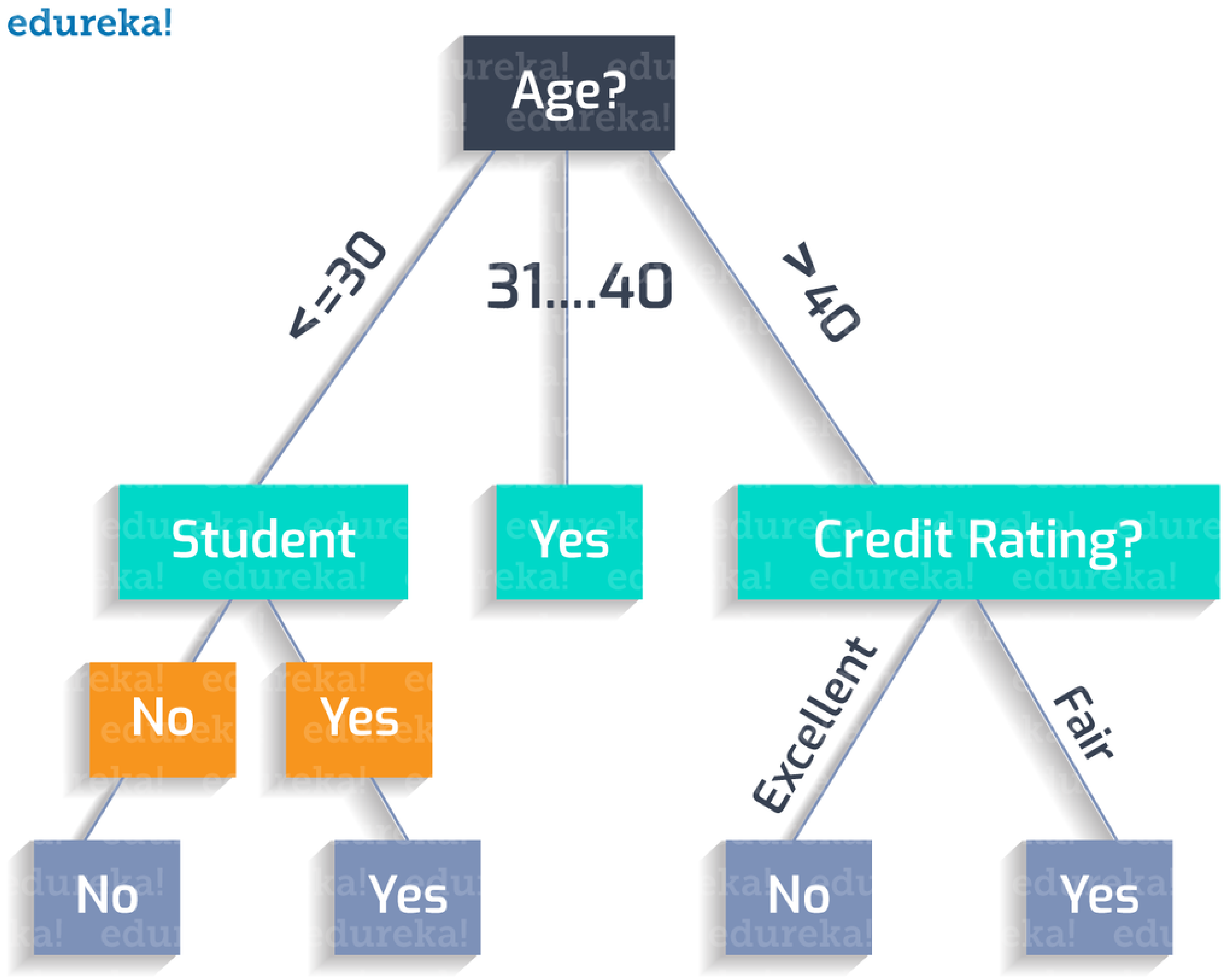
MODELS IN ML

- Multinomial Naive Bayes (MNB)
- Decision Trees
- Neural Networks
- Support Vector Machines (SVM)
- Clustering Algorithms
- and many more ...

Out of these we have choosen Decision Tree, SVM and MNB

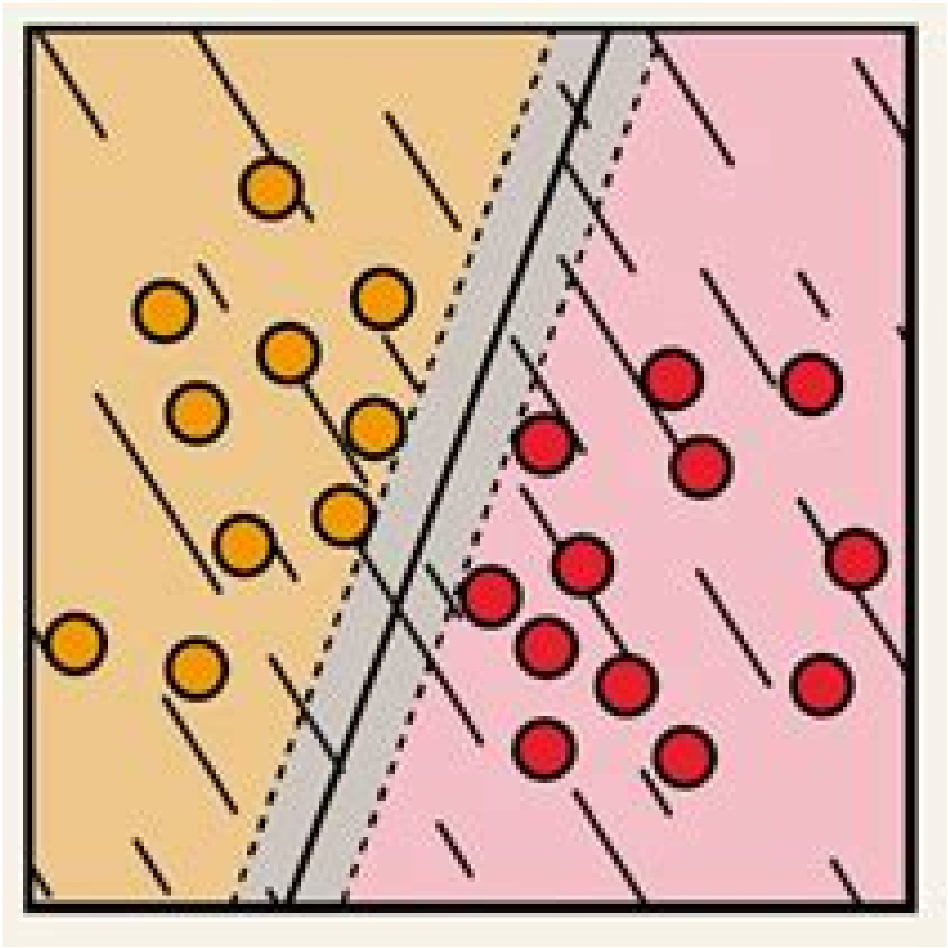
Decision Tree

Decision trees recursively split the data into subsets based on the values of input features, creating a tree-like structure of decisions. Ensemble methods like Random Forests and Gradient Boosted Trees are based on decision trees and are widely used for classification and regression tasks.

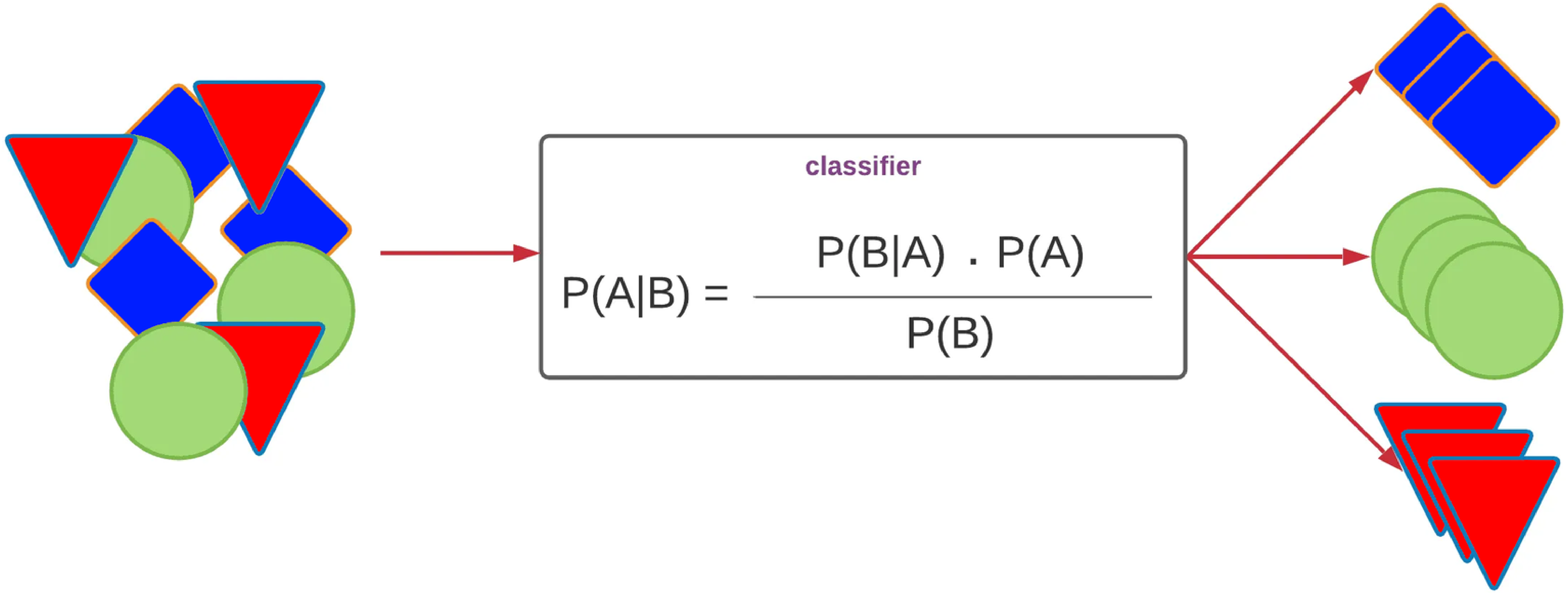


Support Vector Machines

SVMs are supervised learning models used for classification and regression analysis. They find the hyperplane that best separates classes in a high-dimensional space.



Multinomial Naive Bayes



MNB learns the probability distribution of each feature (word) given each class in the training data. It calculates the probability of each class occurring in the training data.
When given a new document (or text), MNB calculates the probability of the document belonging to each class based on the occurrence of its words.
It then selects the class with the highest probability as the predicted class for the document.

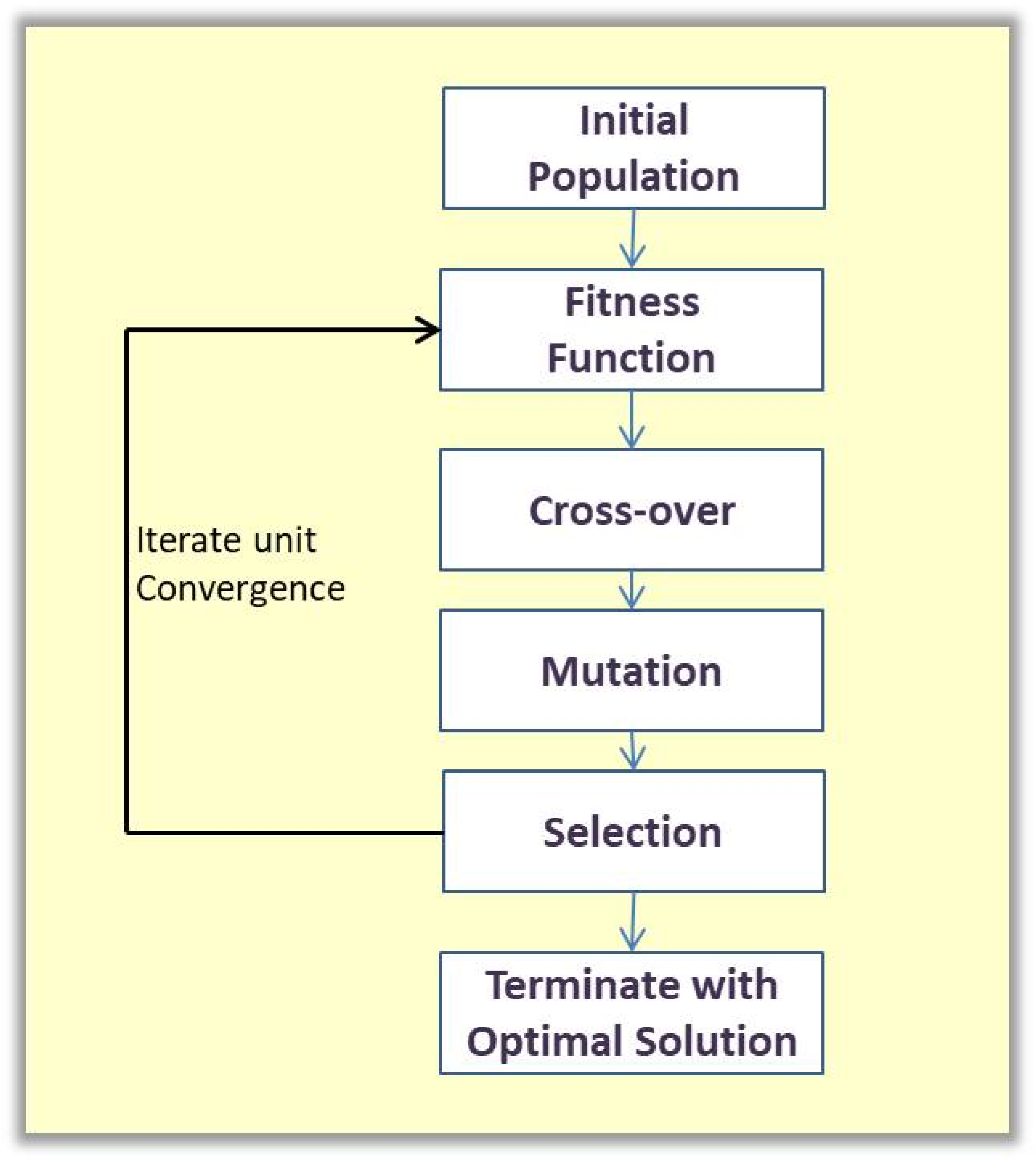
Models in Evolutionary Algorithm

- Genetic Algorithms (GAs)
- Genetic Programming (GP)
- Evolution Strategies (ES)
- Differential Evolution (DE)
- Particle Swarm Optimization (PSO)
- Ant Colony Optimization (ACO)

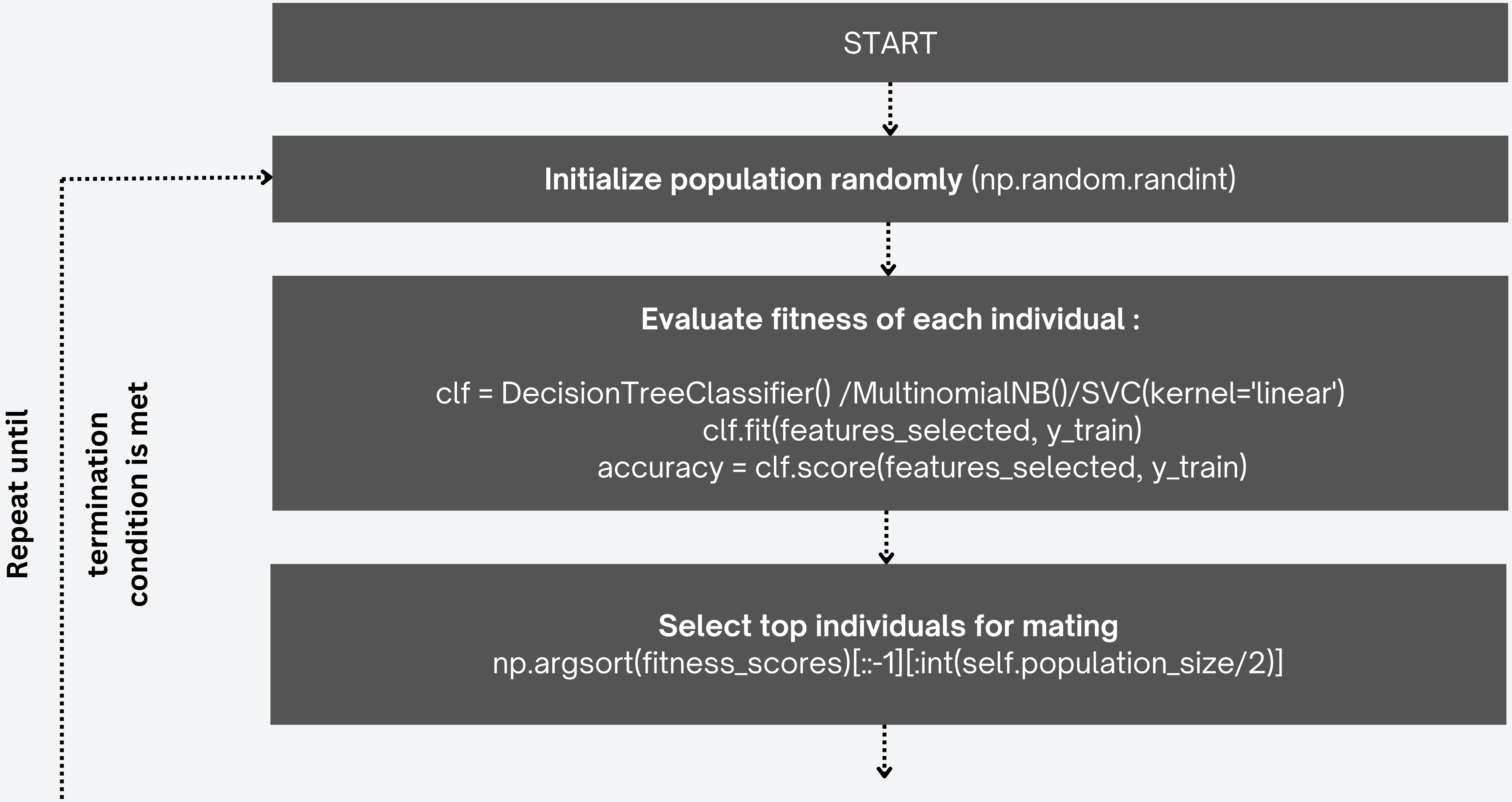
From these we chose GA and PSO

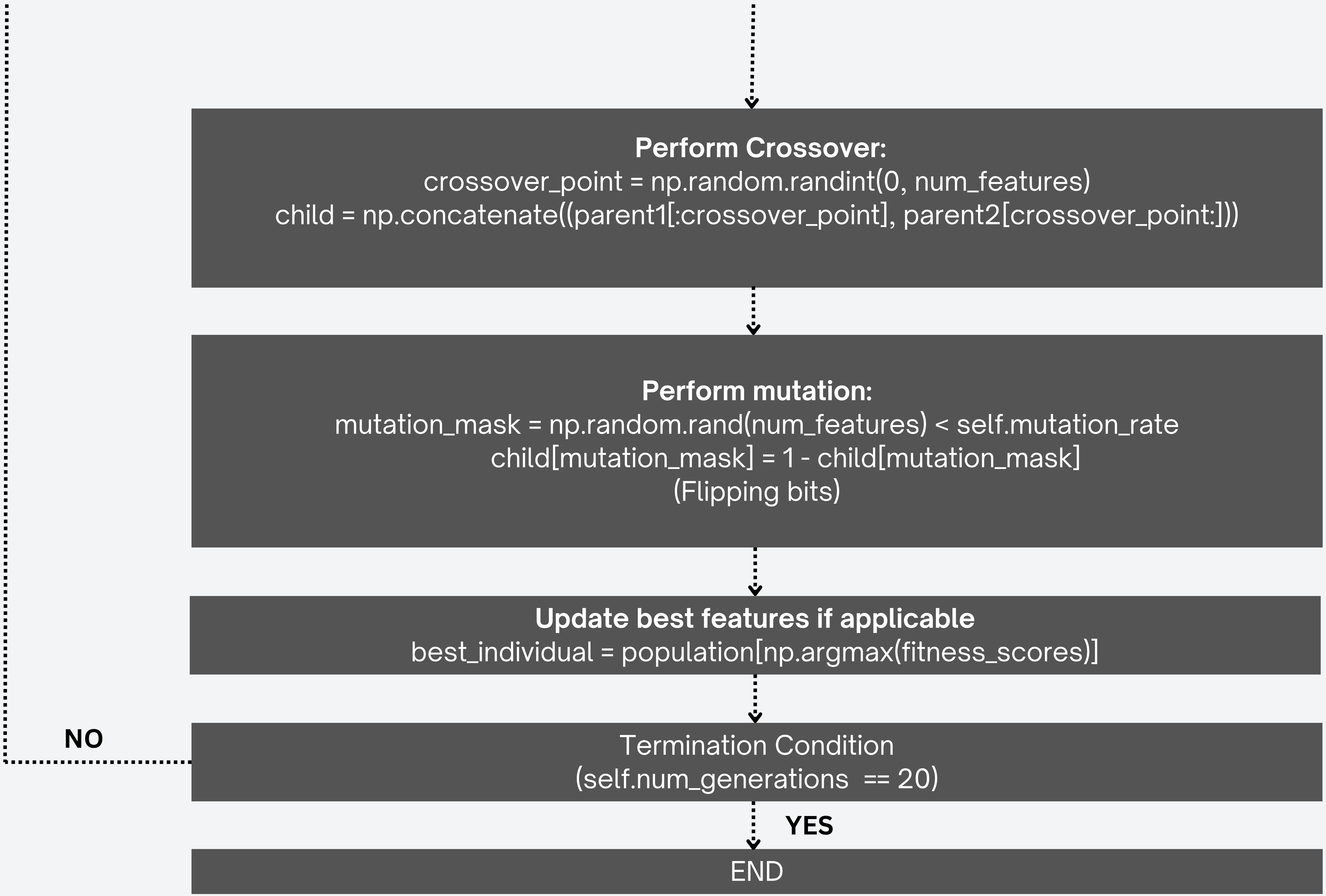
Genetic Algorithm

GA focuses on exploring the search space by maintaining a diverse population of candidate solutions and applying genetic operators like crossover and mutation. This diversity helps in thorough exploration but may take longer to converge.



Flow chart of Genetic Algorithm for feature selection





Perform Crossover:
crossover_point = np.random.randint(0, num_features)
child = np.concatenate((parent1[:crossover_point], parent2[crossover_point:]))
Perform mutation:
mutation_mask = np.random.rand(num_features) < self.mutation_rate
child[mutation_mask] = 1 - child[mutation_mask]
(Flipping bits)
Update best features if applicable
best_individual = population[np.argmax(fitness_scores)]
NO
Termination Condition
(self.num_generations == 20)
YES
END

Flow Chart of Particle Swarm Optimization (PSO) algorithm for feature selection:

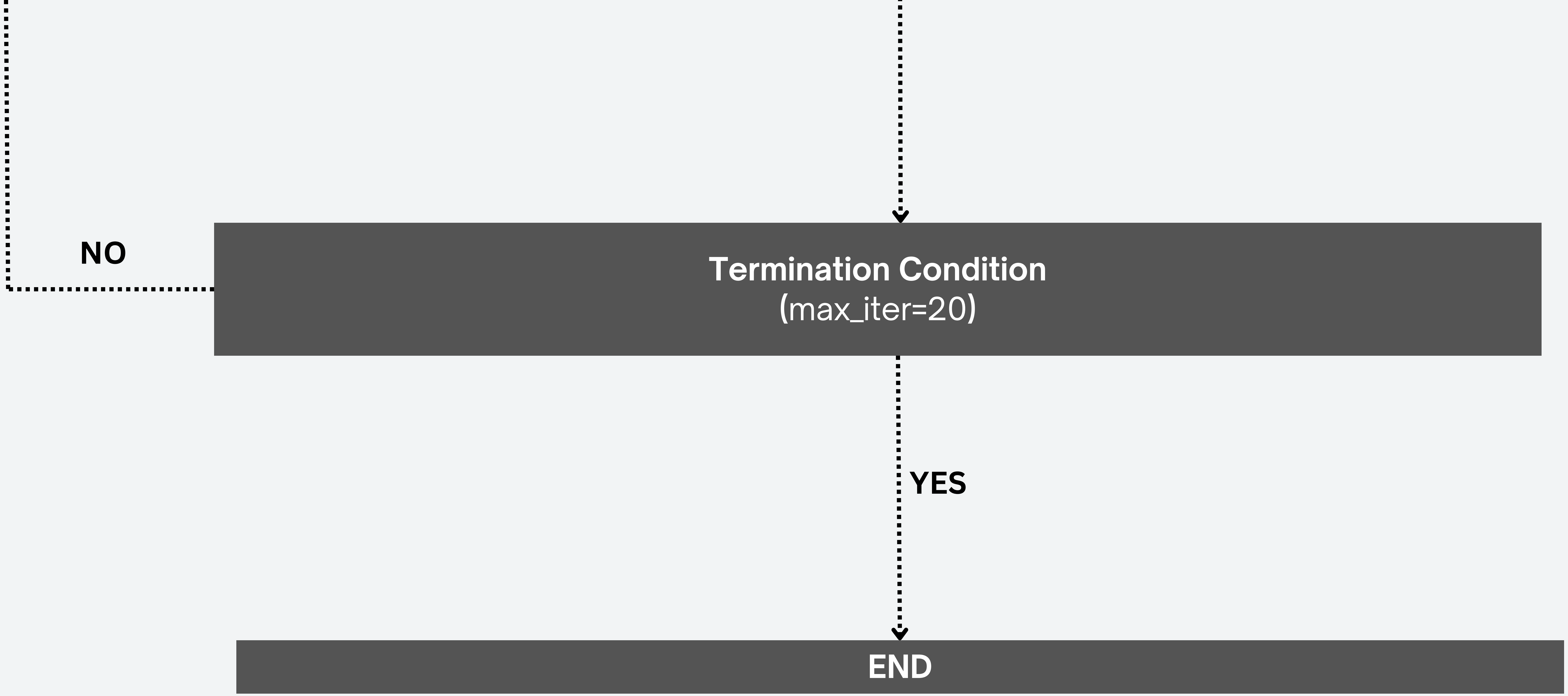
START

Initialize PSOAlgorithm with parameters
(num_particles=50, max_iter=20)

Evaluate fitness:

clf = DecisionTreeClassifier() /MultinomialNB()/SVC(kernel='linear')
clf.fit(features_selected, y_train)
accuracy = clf.score(features_selected, y_train)

Repeat until termination condition is met


NO
Termination Condition
(max_iter=20)
YES
END

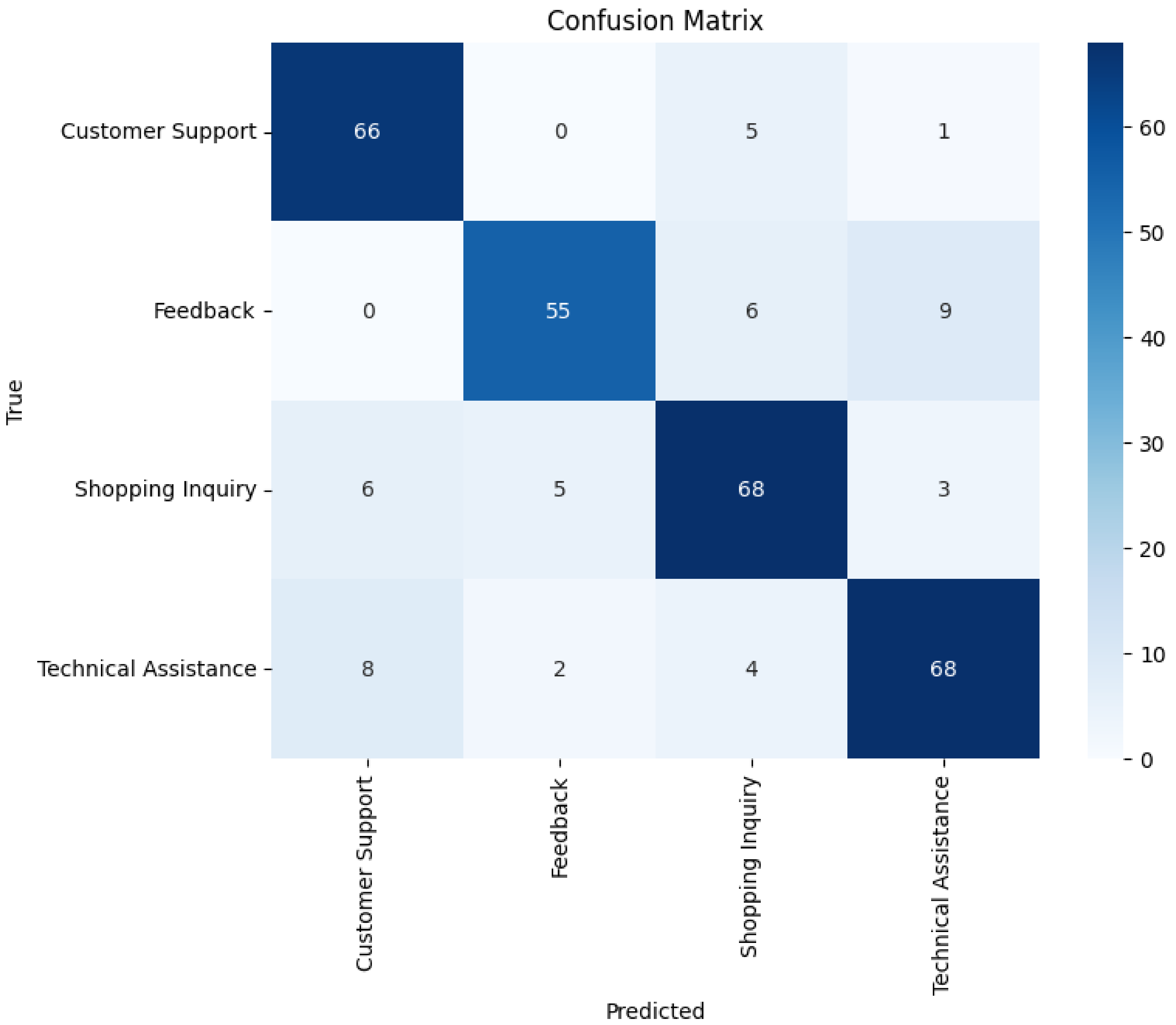
Particle Swarm Optimization

PSO emphasizes exploiting promising regions of the search space by leveraging the collective knowledge of a swarm of particles. This typically leads to faster convergence, but it may sometimes miss out on exploring the entire search space.

Decision Tree with GA

Combining Decision Trees with Genetic Algorithms enhances feature selection in NLP, improving classifier performance. Genetic Algorithms efficiently select relevant features, boosting accuracy and generalization. Decision Trees offer interpretability and scalability, suitable for various domains. This integrated approach enables better intent discernment. Versatile and adaptable, it addresses real-world dataset complexities. Promises to advance NLP system effectiveness.

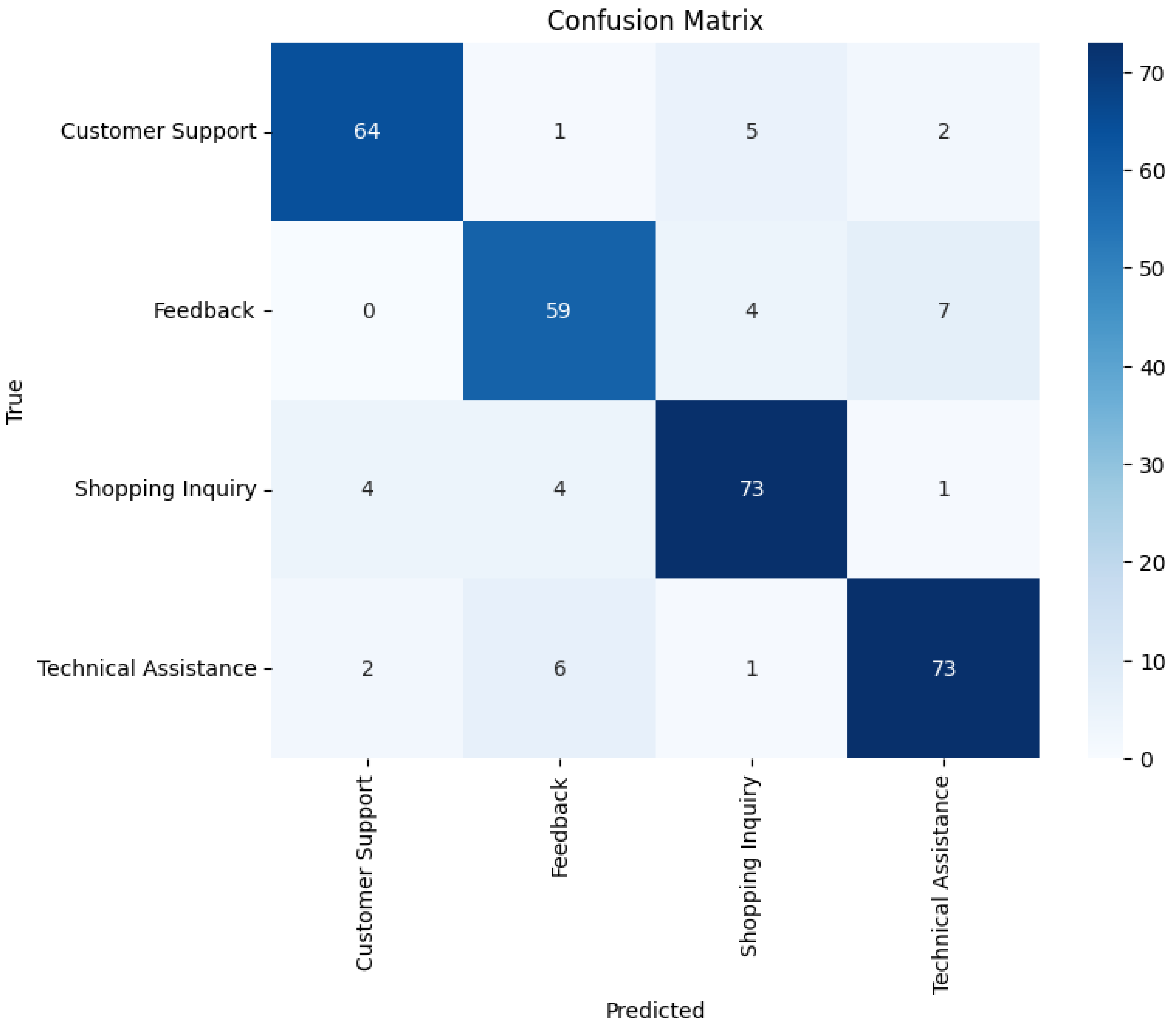
DECISION TREE WITH GA



Decision tree with PSO

Decision Trees with Particle Swarm Optimization (PSO) offer a compelling approach for intent classification. This fusion combines decision trees' interpretability with PSO's optimization capabilities. PSO optimizes decision tree hyperparameters, enhancing classification accuracy and generalization. It simulates birds' behavior to iteratively refine solutions in multidimensional space. The fusion streamlines model optimization and enriches classification outcomes. Promising for intent classification in diverse applications.

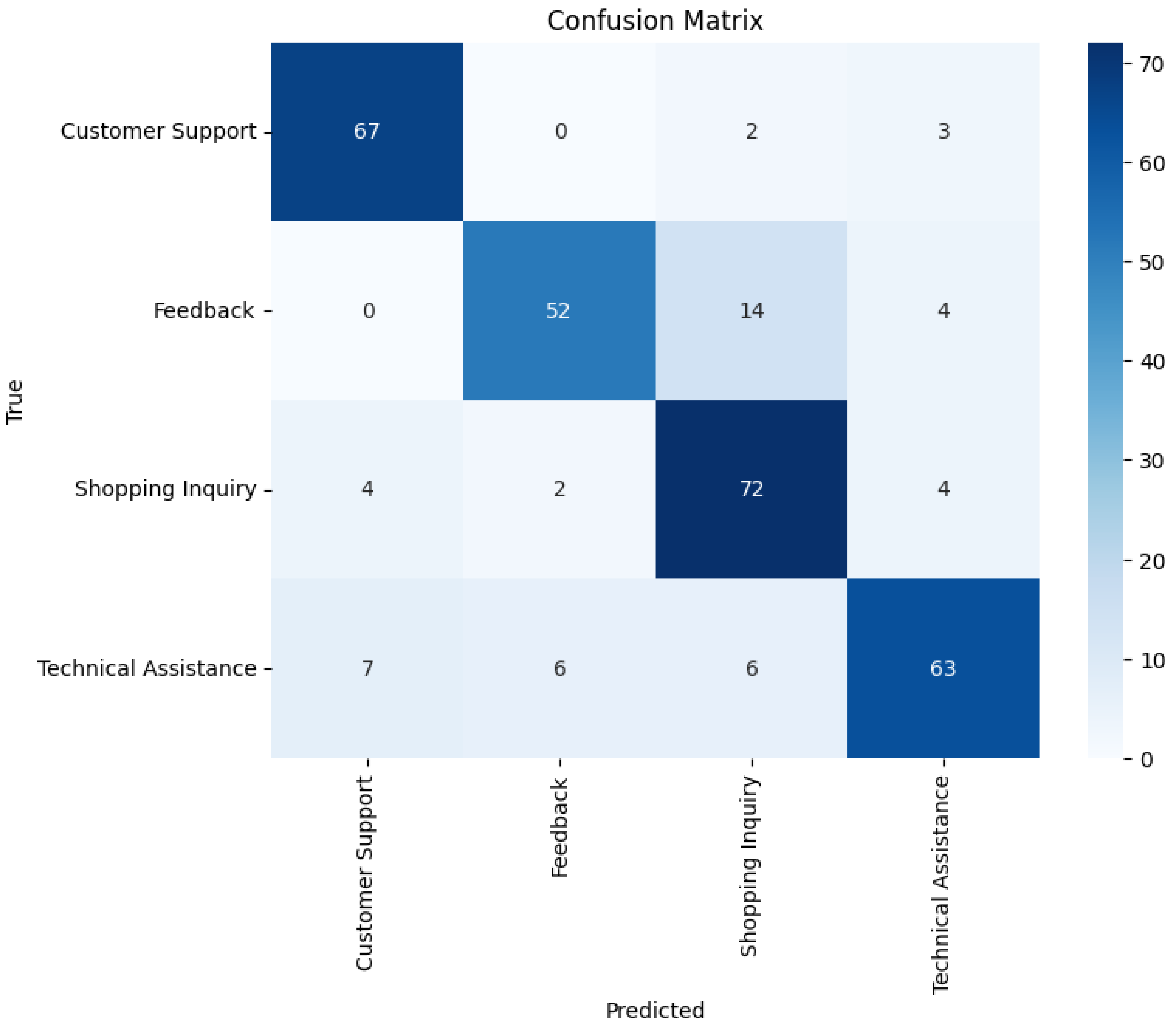
DECISION TREE WITH PSO



MNB with GA

Combining Multinomial Naive Bayes (MNB) with Genetic Algorithm (GA) enhances text-based intent classification. MNB's simplicity and efficiency suit text data well, despite the independence assumption. GA optimizes feature selection, significantly boosting MNB's performance. It navigates high-dimensional feature spaces effectively, selecting discriminative features. This hybrid approach simplifies classification while improving accuracy and robustness. Seamlessly integrating probabilistic modeling with evolutionary computation, it offers a potent solution for real-world applications.

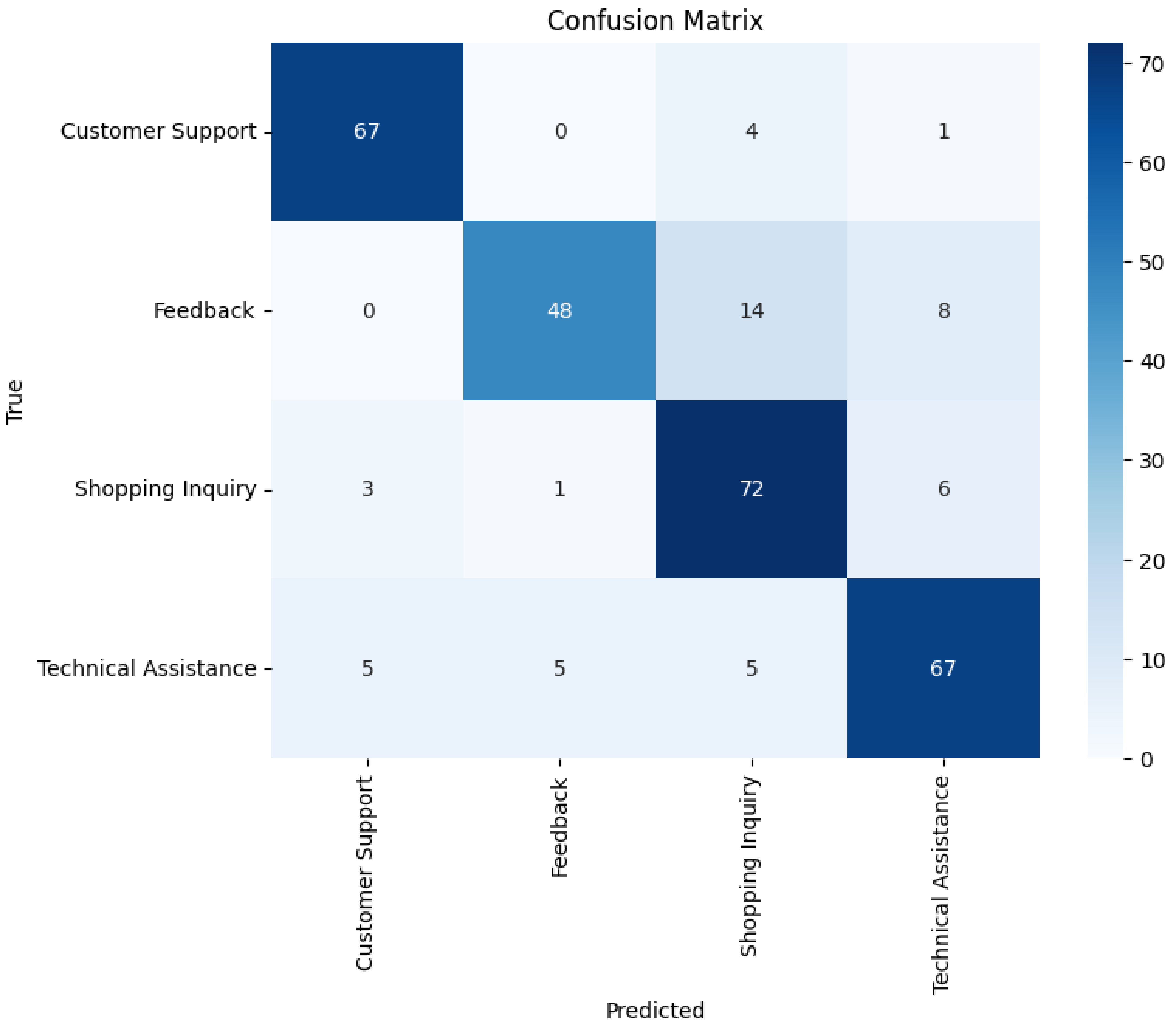
MNB WITH GA


Confusion Matrix
True
Customer Support
Feedback
Shopping Inquiry
Technical Assistance
67 0 2 3
0 52 14 4
4 2 72 4
7 6 6 63
Predicted
Customer Support
Feedback
Shopping Inquiry
Technical Assistance
70
60
50
40
30
20
10
0

MNB with PSO

Multinomial Naive Bayes (MNB) is effective for text classification, including intent classification, by calculating class probabilities from word frequencies. Pairing MNB with Particle Swarm Optimization (PSO) enhances its potency for intent classification, automating feature selection and improving accuracy. PSO identifies relevant words, optimizing classification performance. This combined approach offers a robust solution for categorizing text into intent classes, suitable for diverse NLP applications.

MNB WITH PSO

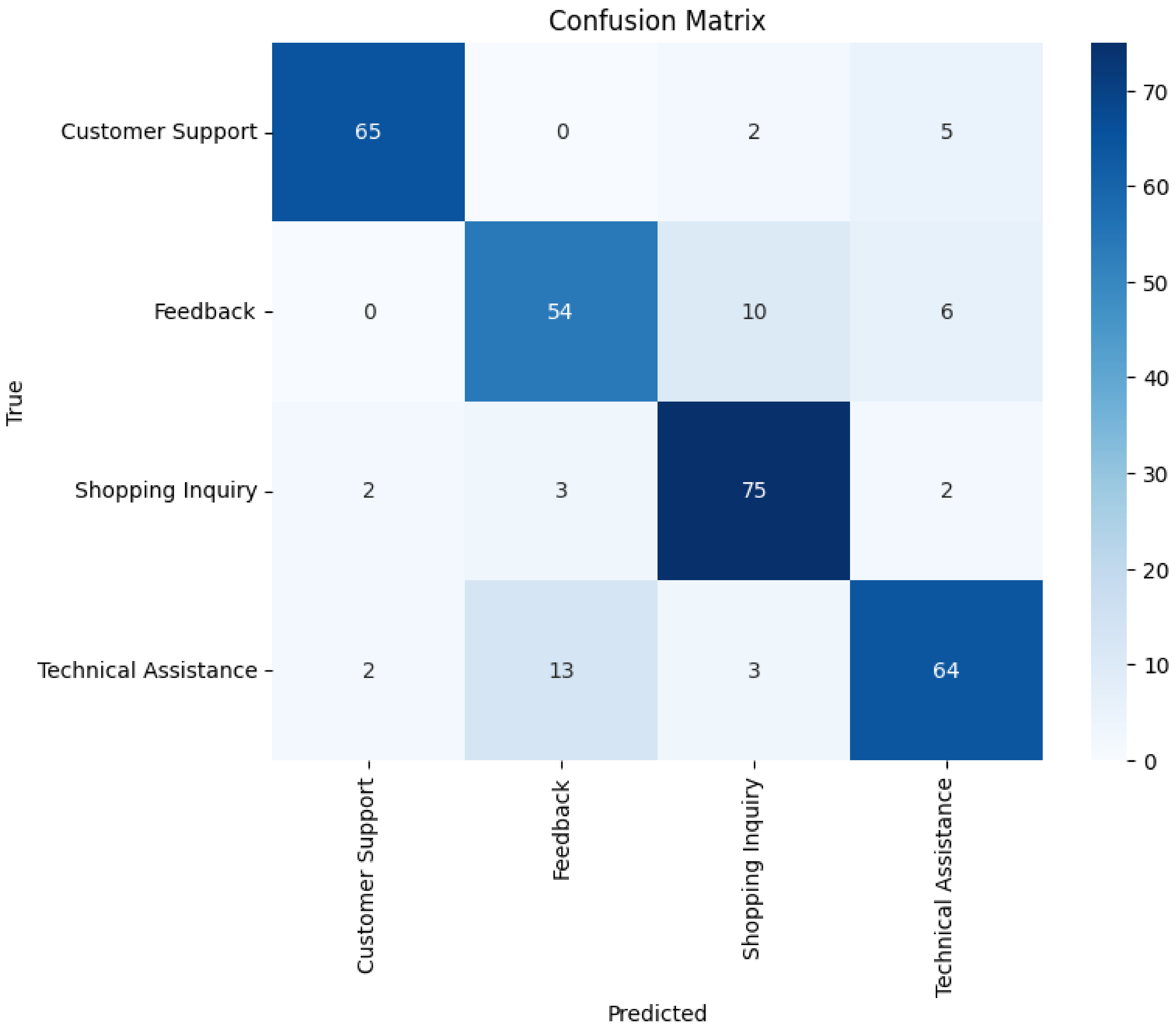


Confusion Matrix
True
Predicted
Customer Support
Feedback
Shopping Inquiry
Technical Assistance
67
0
4
1
0
48
14
8
3
1
72
6
5
5
5
67
70
60
50
40
30
20
10
0

SVM with GA

Support Vector Machines (SVM) coupled with Genetic Algorithms (GA) offer a powerful solution for intent classification in high-dimensional text data. SVMs excel in binary classification, while GAs efficiently navigate feature selection for improved performance. Integrating GA optimizes the feature set, enhancing accuracy and efficiency of SVM-based classification. This combination ensures robust performance across diverse intent categories, empowering systems in various applications. SVM with GA represents a sophisticated yet intuitive framework for accurate intent classification, benefiting virtual assistants and customer service platforms.

SVM WITH GA

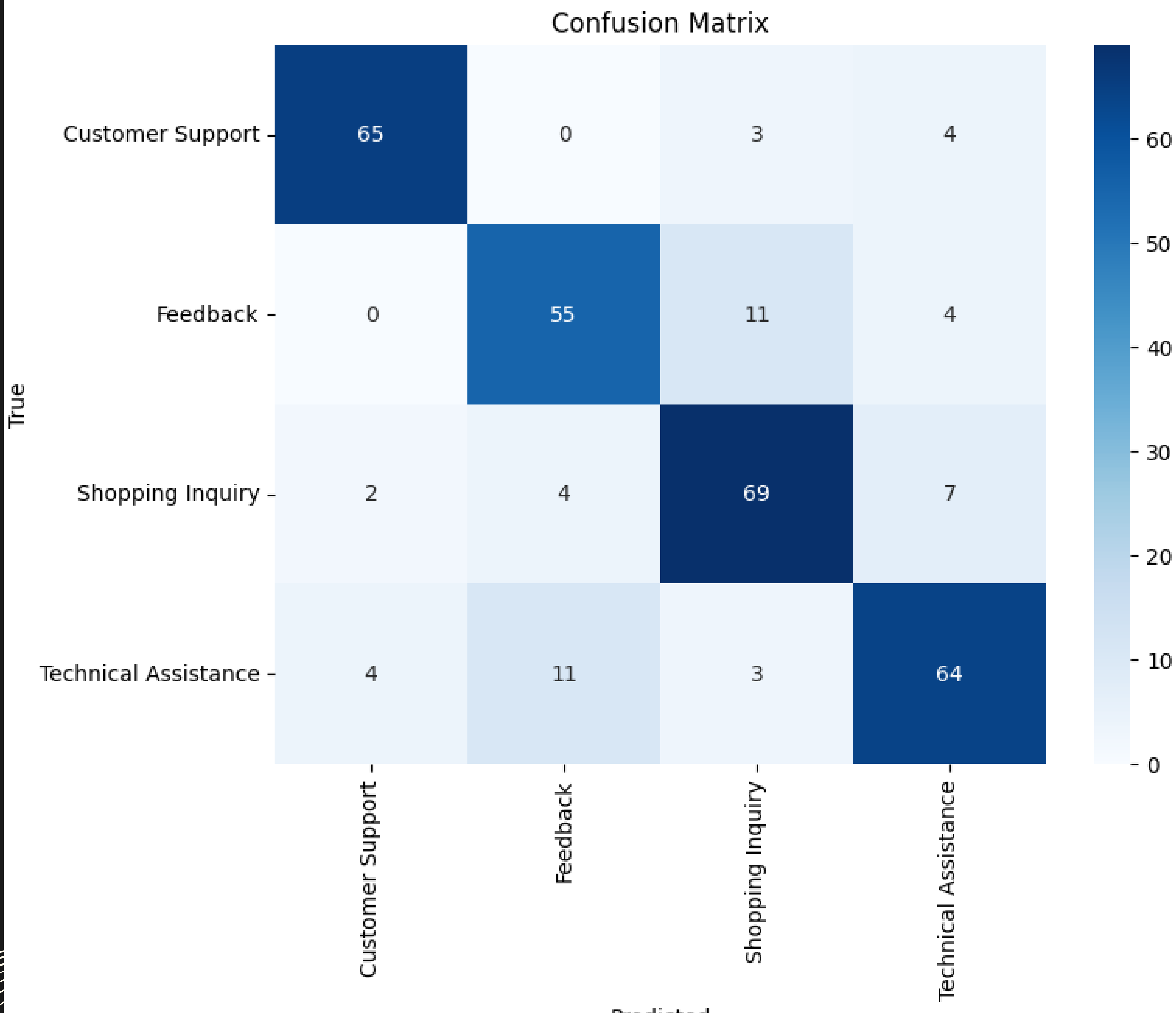


Confusion Matrix
Customer Support
Feedback
Shopping Inquiry
Technical Assistance
True
Predicted
65 0 2 5
0 54 10 6
2 3 75 2
2 13 3 64
70
60
50
40
30
20
10
0

SVM with PSO

Support Vector Machines (SVM) with Particle Swarm Optimization (PSO) combine for efficient intent classification. SVM constructs optimal hyperplanes for class separation, reducing overfitting. PSO enhances SVM's feature selection process iteratively, improving classification accuracy. Inspired by bird and fish behavior, PSO explores and exploits solution spaces effectively. This hybrid approach offers improved feature selection, model performance, and reduced complexity. Suitable for NLP, chatbots, and customer service systems, it provides robust intent categorization.

SVM WITH PSO



Confusion Matrix
True
Predicted
Customer Support
Feedback
Shopping Inquiry
Technical Assistance
65 0 3 4
0 55 11 4
2 4 69 7
4 11 3 64
0
10
20
30
40
50
60

Why GA is better than PSO

GA, even if it is slow, is explorative; it keeps all possible solutions in mind when searching for the best solution

While PSO follows the best solution, that is why it's fast, but it sometimes misses the most optimal one.

Classification Report:

	precision	recall	f1-score	support
Customer Support	0.89	0.89	0.89	72
Feedback	0.82	0.73	0.77	70
Shopping Inquiry	0.82	0.83	0.82	82
Technical Assistance	0.78	0.84	0.81	82
accuracy			0.82	306
macro avg	0.83	0.82	0.82	306
weighted avg	0.82	0.82	0.82	306

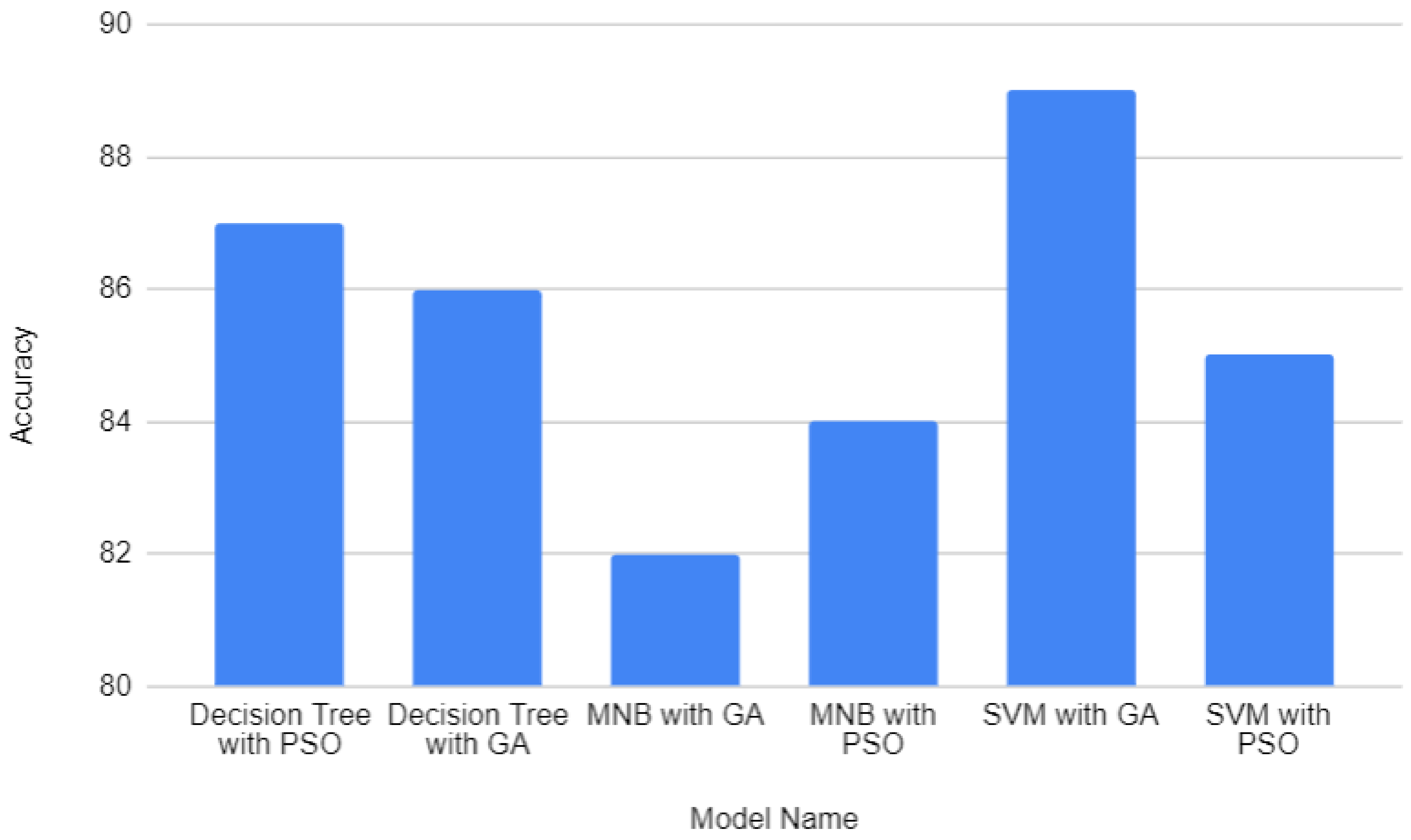
PSO

Classification Report:

	precision	recall	f1-score	support
Customer Support	0.86	0.94	0.90	72
Feedback	0.91	0.71	0.80	70
Shopping Inquiry	0.80	0.87	0.83	82
Technical Assistance	0.82	0.83	0.82	82
accuracy			0.84	306
macro avg	0.85	0.84	0.84	306
weighted avg	0.84	0.84	0.84	306

GA

Overall Accuracy



FEATURE SELECTION

ok that’s fine you are using ML Algorithm for Intent classifier, But what is the use of Evolutionary Algorithm in your project.

In our project, the Evolutionary Algorithm (EA) plays a crucial role in optimizing the performance of intent classifier model by doing feature selection.

Now How exactly you are performing feature selection ?

In our project, feature selection plays a crucial role in improving the performance of your intent classification model by selecting the most relevant features (or attributes) from your dataset.

Here's how feature selection benefits your project:

1. Dimensionality Reduction:
   - Feature selection helps in reducing the dimensionality of your dataset by eliminating irrelevant or redundant features.
2. Improved Model Performance:
   - By selecting only the most relevant features, you provide your machine learning model with more focused and informative input.
3. Faster Training and Inference:
   - With fewer features to process, both training and inference times of your model are reduced.
4. Reduced Overfitting:

- Removing irrelevant features reduces the model's tendency to memorize noise in the training data, which in turn reduces overfitting.

Ok which Evolutionary Algorithm are you using?

Genetic Algorithm and PSO(Particle Swarm Optimization) Evolutionary Algorithm are used for intent classification task .

MODEL EVALUATION AND TESTING

Splitting the dataset into three subsets: training, validation, and test sets. The training set is used to train the model, the validation set helps to perform feature selection and the test set evaluates the final model performance.

SPLIT DATA

Once you've trained and tuned your model, evaluate its performance on the test set. Pass the test set through the trained model and compare the predicted labels with the true labels. Calculate the chosen evaluation metrics to assess the model's performance.

EVALUATE ON TEST SET

Analyze the results of the evaluation metrics to understand how well your model is performing. Pay attention to metrics like accuracy, precision, recall, and F1-score to gauge different aspects of performance.

ANALYZE RESULTS

Based on the evaluation results, identify areas where the model may be underperforming and iterate on your approach. This could involve refining the preprocessing steps, adjusting model parameters, or exploring different algorithms.

ITERATE AND IMPROVE

OUR TEAM AND CONTRIBUTIONS

1. **Dataset Collection:** Siddharth Dhiman (20JE0948), Soutrik Das (20JE0971) , Varun Parihar (20JE1062), Ayush Ranjan (21JE0215)
2. **DataSet Preprocessing:** Chirag Gajana (20JE0296), Siddharth Dhiman (20JE0948)
3. **Feature Selection:** Avinash Kumar (20JE0217), Aviral Kumar Singh (20JE0220), Varun Parihar (20JE1062), Ayush Ranjan (21JE0215)
4. **Model Development:** Avinash Kumar (20JE0217), Anurag Kumar(20JE0169), Soutrik Das (20JE0971), Piyush Mishra (21JE0654)
5. **Documentation:** Soutrik Das (20JE0971) , Piyush Mishra (21JE0654)

Source Code

https://github.com/Avinash19042002/Intent-Classification-Using-EA/tree/main

Avinash19042002 Multinomial Naive Bayes with GA · defabc1 · 12 hours ago · 12 Commits

Decision Tree with GA.ipynb	Decision tree with GA	4 days ago
Decision_Tree_with_PSO.ipynb	Decision tree with PSO	4 days ago
MNB with GA.ipynb	Multinomial Naive Bayes with GA	12 hours ago
MNB_PSO_With_Pyswarm_Library.ipynb	MNB_PSO_with_Pyswarm	4 days ago
MNB_without_pyswarm.ipynb	MNB_without_pyswarm	4 days ago
SVM_with_GA.ipynb	SVM with GA	4 days ago
SVM_with_PSO.ipynb	SVM with PSO	4 days ago
dataset.csv	Added Dataset	12 hours ago

THANK YOU