Question 02

Imagine you are tasked with building a simple Natural Language Understanding (NLU) system to classify intents from text input. Your goal is to train a model using given text examples for different intents and then test it with random texts, providing an intent classification along with a confidence score. You are also required to implement a fallback mechanism in case the confidence level does not meet a certain threshold.

Scenario:

You are provided with a dataset containing examples of different intents along with their corresponding labels. For example:

- Intent: Greet
  - Examples: "Hi", "How are you?", "Hello"
- Intent: Farewell
  - Examples: "Goodbye", "See you later", "Take care"
- Intent: Inquiry
  - Examples: "What's the weather like today?", "Can you tell me the time?", "Where is the nearest restaurant?"

You are required to train a model using this dataset to classify intents from new text inputs. Once trained, your model should be able to classify intents with a confidence score. If the confidence score for a classification exceeds a predefined threshold (e.g., 0.7), your system should return the predicted intent along with the confidence score. If the confidence score does not meet this threshold, your system should return a fallback response indicating that the intent could not be confidently determined.

Tasks:

1. Implement the training process for your intent classification model using a suitable machine learning or natural language processing library (e.g., scikit-learn, TensorFlow, PyTorch).
2. Develop a function to classify intents from new text inputs using the trained model. This function should return the predicted intent along with a confidence score.
3. Implement a fallback mechanism in your classification function. If the confidence score for the predicted intent is below the predefined threshold, return a fallback response (e.g., "NLU fallback: Intent could not be confidently determined").
4. Test your model and classification function with random text inputs to ensure they provide the expected intent classifications along with confidence scores or fallback responses.

Submission Requirements:

Submit your Python code implementing the training process, intent classification function, and fallback mechanism.

Include a brief explanation of your approach, detailing the choice of model, any preprocessing steps, and the rationale behind the confidence threshold selected for fallback.

Note: You may use any programming language and libraries of your choice to complete this task, but provide clear instructions for setting up and running your code