## You

I just built an AI search system for training guides, what stages of testing can I do and what possible tests can I do for it?

## Copilot

Congratulations on building your AI search system! Testing is a crucial step to ensure its reliability and effectiveness. Here are some stages of testing and possible tests you can perform:

1. **Training, Validation, and Testing**:
   * **Training**: During this stage, your AI model learns patterns and relationships from a dataset.
   * **Validation**: Fine-tune and optimize the model using validation data.
   * **Testing**: Assess the model's performance on new data [2].
2. **AI-Specific Testing Considerations**:
   * **Data Quality**: Validate the quality of your training data.
   * **Model Accuracy**: Evaluate how well the model predicts outcomes.
   * **Bias Detection**: Check for biases in predictions.
   * **Robustness**: Test the model's behavior under different conditions.
   * **Exploratory Testing**: Experiment with various scenarios [1].
3. **Tools and Techniques**:
   * **Visual Testing**: Assess the application's look and feel using image-based learning and screen comparisons.
   * **Declarative Testing**: Specify test intent in natural or domain-specific language, letting the system decide how to perform the test [4].

Remember that AI testing is iterative, and continuous improvements are essential. Good luck with your testing process! 😊🚀

Source: Conversation with Copilot, 9/30/2024

1. [The 4 Stages of AI: Understanding the Journey to Intelligent Machines](https://gcelt.org/the-4-stages-of-ai-understanding-the-journey-to-intelligent-machines/)
2. [Chapter 6 - Testing and Iteration | AI in Production Guide](https://azure.github.io/AI-in-Production-Guide/chapters/chapter_06_testing_waters_testing_iteration)
3. [AI in Software Testing | Best Approaches to Look in 2024 - TestingXperts](https://www.testingxperts.com/blog/AI-in-Software-Testing)

## You

How can I experiment with various scenarios for exploratory testing?

## Copilot

Certainly! Exploratory testing involves actively exploring the application to uncover issues. Here are some techniques you can use:

1. **Ad Hoc Testing**:
   * Freestyle exploration without predefined test cases.
   * Investigate different features, inputs, and edge cases.
2. **Scenario-Based Testing**:
   * Create realistic user scenarios.
   * Test how the system responds to specific actions or sequences.
3. **Boundary Testing**:
   * Push system boundaries (e.g., maximum input length, extreme values).
   * Observe how the application handles these situations.
4. **Negative Testing**:
   * Purposefully input incorrect data.
   * Verify error handling and graceful degradation.
5. **Exploring Data Variability**:
   * Use different data sets (valid, invalid, edge cases).
   * Observe how the system processes them.
6. **Time-Related Scenarios**:
   * Test during peak load times or low-resource conditions.
   * Check for performance issues.

Remember, exploratory testing is creative and flexible. Trust your instincts and document your findings. Happy testing! 😊🔍

## You

What boundaries can I test for this AI search?

## Copilot

Certainly! When testing your AI search system, consider exploring the following boundaries:

1. **Input Length**:
   * Test with very short and very long queries.
   * Observe how the system handles different input lengths.
2. **Special Characters and Symbols**:
   * Include special characters (e.g., punctuation, emojis).
   * Verify if the system processes them correctly.
3. **Ambiguous Queries**:
   * Test queries with multiple interpretations.
   * Check if the system provides relevant results.
4. **Synonyms and Variations**:
   * Use synonyms or similar terms.
   * Ensure the system captures context and variations.
5. **Out-of-Vocabulary Terms**:
   * Introduce uncommon or domain-specific words.
   * Evaluate how the system handles unfamiliar terms.
6. **Edge Cases**:
   * Test extreme values (e.g., empty query, very long query).
   * Verify robustness and error handling.

Remember to document your findings and iterate on improvements. Happy testing! 😊🔍