

Reflection Pattern Agentic AI

Reflection Pattern Agentic AI

1. Reflection Pattern
2. Reflection Pattern
3. Reflection Pattern
4. Reflection Pattern
5. Reflection Pattern
6. Reflection Pattern
7. Reflection Pattern
8. Reflection Pattern
9. Reflection Pattern
10. Reflection Pattern

Reflection Pattern

Reflection is a pattern where a model reflects on its own output and iteratively improves it. This is often used in tasks where the model's output is evaluated and then used to refine its response. The process typically involves the model generating an initial response, which is then evaluated against a set of criteria. Based on the evaluation, the model generates a reflection on its performance and uses this to improve its next response. This process can be repeated multiple times to achieve a higher quality output.

Reflection Pattern

Reflection (LLM) is a process where a model reflects on its own output and iteratively improves it. This process involves generating an initial response, evaluating it, and then refining it based on feedback. The reflection process can be implemented in various ways, such as using a separate model for evaluation or incorporating feedback into the main model's training. The goal is to produce more accurate and reliable outputs through iterative refinement.

Reflection Pattern

Reflection Pattern - Generator LLM
Reviewer LLM Generator



■■ ■■■■ ■■■ ■■■■■■■■ Reflection Pattern ■■■■■■■■ ■■■■■■■■ ■■■■■■■■■■:

■■■■■:

```
class ReflectionAgent:
    def __init__(self):
        self.generator_llm = GeneratorLLM()
        self.reviewer_llm = ReviewerLLM()
        self.max_iterations = 3

    # Evaluation criteria for article review
    self.review_criteria = [
        "Grammar and spelling accuracy",
        "Clarity and readability",
        "Factual accuracy",
        "Logical flow and structure",
        "Relevance to the topic"
    ]
```



■■■■■■■■■■ ■■■■■■ ■■■■■■ ■■■■■■ ■■■■■■■■■■ ■■ ■■■■■■■■■■ ■■■■ ■■■■■■■■■■:
 ■■■■■■■■■■ ■■■■■■■■■■ ■■■■■■ ■■ ■■■■■■■■■■ ■■■■■■■■■■ ■■■■■■ ■■ 1
 ■■■■ 10 ■■ ■■ ■■■■■■ ■■■■■■ ■■■■ ■■■■■■ ■■■■■■: ■■■■■■ ■■ ■■■■ ■■■■■■ ■■■■
 ■■■■■■■■■■ ■■■■ ■■■■■■ ■■ ■■■■■■■■■■ ■■■■■■■■■■



■■■■■ ■■■■■ ■■■■ ■■■■■■■■■■ Reflection Pattern ■■ ■■■■■■■■ ■■■■■■■■ ■■■■■■■■:

■■■■■:

```
class CodeReflectionAgent:
    def __init__(self):
        self.code_generator = CodeGeneratorLLM()
        self.code_reviewer = CodeReviewerLLM()

        # Code review criteria
        self.code_criteria = [
            "Correctness and functionality",
            "Code readability and organization",
            "Performance optimization",
            "Security best practices",
            "Error handling",
            "Documentation and comments"
        ]
```

Reflection Pattern

1. **Identifikasi Masalah:** Menentukan masalah yang akan diteliti, seperti "Pengaruh penggunaan media digital terhadap efektivitas pembelajaran."

2. **Penyusunan Hipotesis:** Merumuskan dugaan sementara tentang hasil penelitian, misalnya "Penggunaan media digital meningkatkan efektivitas pembelajaran."

3. **Metode Penelitian:** Menentukan pendekatan, teknik pengumpulan data, dan analisis data yang akan digunakan.

4. **Pengumpulan Data:** Melakukan pengumpulan data sesuai dengan metode yang telah ditentukan.

5. **Analisis Data:** Melakukan analisis data untuk menguji hipotesis dan menjawab pertanyaan penelitian.

6. **Penyusunan Kesimpulan:** Menyimpulkan hasil penelitian dan memberikan rekomendasi berdasarkan temuan.

[illegible]



Reflection Pattern ■■■ ■■■■■ ■■■■■ ■■■■■ ■■■■■ ■■■■■ ■■■■■■■■■■■ ■■■ ■■■■■■
Agentic AI. ■■■■■■ ■■ ■■■■■■ ■■■■■■■ ■■■■■ ■■ ■■■■■ ■■■■■■■ ■■■■■■■■■■.
■■■■■■■■ ■■■■■■■■■■:■■■ ■■ ■■■■■■ ■■■■■■■■■■ ■■■■ ■■■■■ ■■■■ AI ■■■■■
■■■■■■ ■■■■■■ ■■■■■■■■■ ■■■■■■■■■■■. ■■■■■ ← ■■■■■ ← ■■■■ ← ■■■■ ← ■■■■
■■■ ■■■■■ ■■■■■■■ ■■■■■■■■■■■