Lao Zi by Pi.Tech

Laozi

Contrainted Reasoning LLM generic plugin README.md

Laozi Go Plugin

A domain-agnostic LLM insight generation engine with hallucination prevention.

Key Features

- **Build-time configuration** - All settings compiled into binary
- **Pre-computed comparisons** - Deterministic severity based on thresholds
- **Validation + Regeneration** - Auto-retry on invalid LLM output
- **Parallel execution** - Configurable concurrent LLM calls
- **Optional RAG** - Augment with domain documents

Quick Start

```
package main

import (
    "context"
    "github.com/Phoenix-Innovation/laozi"
)

func main() {
    // Create engine (all config from config.go)
```

```
    engine := laozi.New(
        laozi.NewDefaultLLMClient(),
        laozi.NewInMemoryRAG(),
    )

    // Add categories with thresholds
    engine.AddCategory(laozi.Category{
        ID:   "glucose",
        Name: "Blood Glucose",
        Thresholds: []laozi.Threshold{{
            Metric:    "glucose",
            Min:       70,
            Max:       99,
            Unit:      "mg/dL",
            Source:    "ADA",
            SourceURL: "https://diabetes.org/guidelines",
        }},
    })

    // Analyze
    metrics := map[string]float64{"glucose": 105}
    insights, _ := engine.AnalyzeAll(context.Background(), metrics)
}
```

Configuration (config.go)

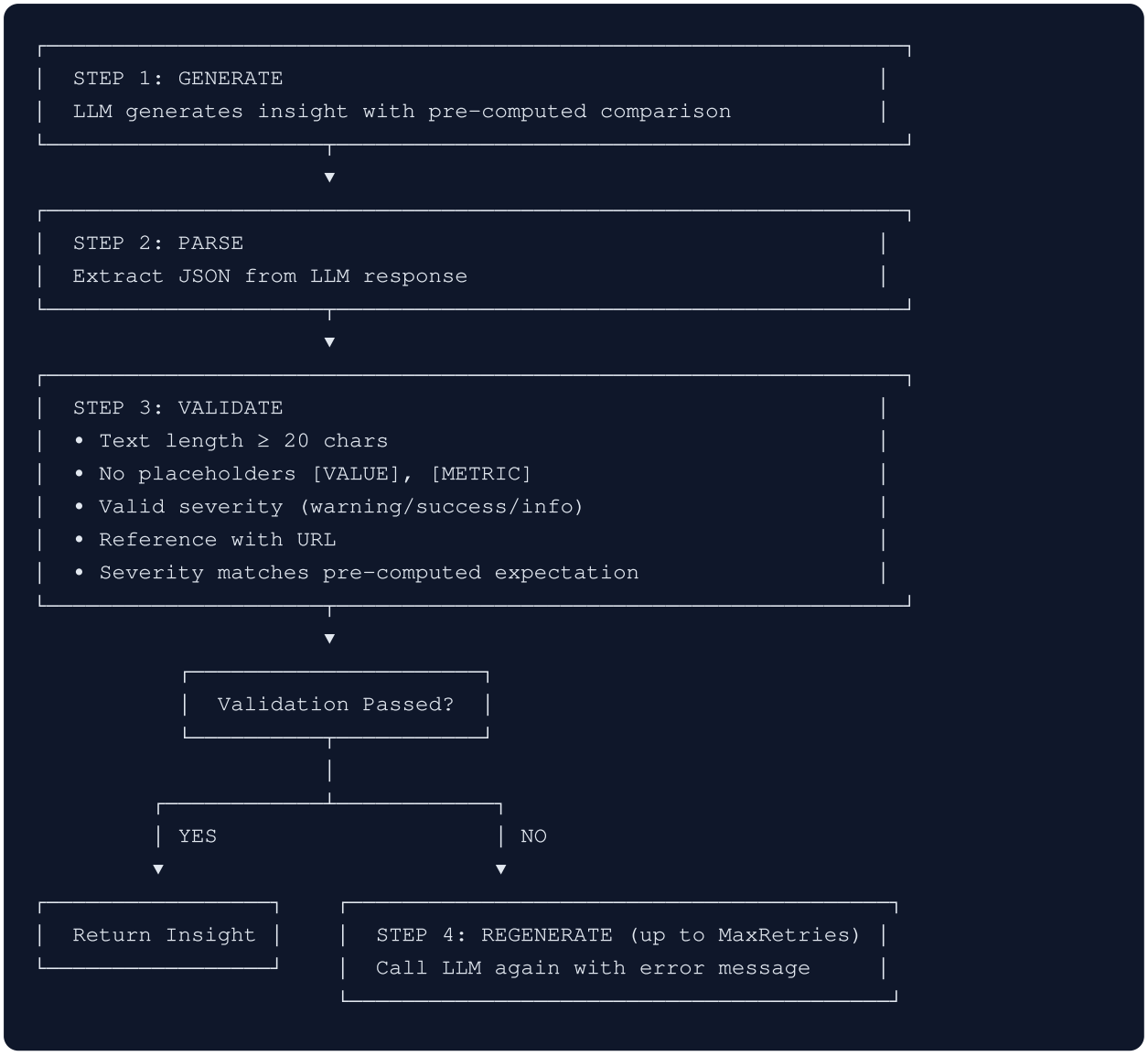
All settings are in `config.go` and compiled into the binary:

```
// LLM Configuration
const (
    LLMModel            = "gpt-4o-mini"
    LLMEndpoint         = "https://api.openai.com/v1/chat/completions"
    LLMTemperature      = 0.3
    LLMMaxTokens        = 500
    MaxParallelLLMCalls = 35
    LLMTimeoutSeconds   = 30
)

// RAG Configuration
const (
    RAGEnabled       = true
    RAGTopK          = 3
    RAGMinSimilarity = 0.7
)

// Validation Configuration
const (
    MaxRetries          = 2
    MinInsightTextLen   = 20
    RequireReference    = true
```

```
    EnforceSeverityMatch = true
)
```

To customize, edit `config.go` and rebuild.

Architecture



Pre-computed Comparison

The key anti-hallucination mechanism:

```
GUIDELINES:
📊 CURRENT_RATIO: Range 1.50 - 3.00 ratio
   Source: CFA Institute
   URL: https://cfainstitute.org/...
```

```
PATIENT VALUES:
- current_ratio: 1.20 ratio

PRE-COMPUTED COMPARISON:
current_ratio: 1.20 < 1.50 (BELOW minimum) → severity = warning

REQUIRED SEVERITY: warning
```

The LLM's only job is to write prose around the deterministic result.

Example Log Output

```
15:04:05 [BATCH] all: Starting 3 categories with 35 parallel
15:04:05 [GENERATE] liquidity: Calling LLM {expected=warning}
15:04:06 [PARSE] liquidity: OK {severity=success}
15:04:06 [VALIDATE] liquidity: ✗ FAIL {error=severity mismatch: got success, expecte
15:04:06 [REGENERATE] liquidity: Attempt 1/2
15:04:07 [REGENERATE] liquidity: ✓ SUCCESS
15:04:05 [GENERATE] profitability: Calling LLM {expected=success}
15:04:06 [PARSE] profitability: OK {severity=success}
15:04:06 [VALIDATE] profitability: ✓ PASS
```

API Reference

Engine

```go
// Create engine
engine := laozi.New(llmClient, ragStore)

// Add context (optional)
engine.AddContext("company", map[string]interface{}{...})

// Add categories
engine.AddCategory(category)
engine.AddCategories([]Category{...})

// Analyze
insight, err := engine.Analyze(ctx, "categoryID", metrics)
insights, err := engine.AnalyzeAll(ctx, metrics)

// Custom logger
engine.SetLogger(myLogger)
```

Types

```
type Category struct {
    ID          string
    Name        string
    Thresholds  []Threshold
    Educational bool   // For "Did You Know" tips
    RAGQuery    string // Optional RAG query
}

type Threshold struct {
    Metric    string  // Metric name
    Min       float64 // Minimum acceptable
    Max       float64 // Maximum acceptable
    Unit      string  // Unit label
    Source    string  // Source name
    SourceURL string  // Source URL
}

type Insight struct {
    CategoryID string
    Text       string
    Severity   Severity // "warning", "success", "info"
    Reference  string
    Metadata   map[string]string
}
```

Environment Variables

Variable	Description
`LAOZI_API_KEY`	LLM API key (overrides config.go)

License

MIT License