PROVISIONAL PATENT APPLICATION

COVER SHEET

Title of Invention: UltrLLMOrchestrator System (UltrAI)

Inventor(s): Joshua Field 1625 SW Alder St. #510 Portland, OR 97205 Citizenship: United States of America

Correspondence Address: Joshua Field 1625 SW Alder St. #510 Portland, OR 97205

Application Filed By: Inventor

SPECIFICATION

Title: UltrLLMOrchestrator System (UltrAI)

Cross-Reference to Related Applications

[Not Applicable]

Field of the Invention

This invention relates generally to artificial intelligence systems, and more particularly to a system and method for orchestrating multiple Large Language Models (LLMs) through structured analysis patterns to generate enhanced insights.

Background of the Invention

Large Language Models (LLMs) have emerged as powerful tools for natural language processing and generation. However, individual LLMs often have limitations in terms of reliability, depth of analysis, and susceptibility to biases. Current approaches typically utilize a single LLM to respond to user queries, which fails to leverage the potential benefits of combining multiple models with different capabilities and training methodologies. Additionally, existing LLM implementations lack structured analytical frameworks that guide these models through complex reasoning processes.

There exists a need for an advanced system that can effectively orchestrate multiple LLMs through defined analytical patterns, leverage document context intelligently, and synthesize insights in a manner that enhances the quality, depth, and reliability of the generated outputs.

Summary of the Invention

The present invention, UltrAI, is an advanced AI analysis platform designed to enhance the quality, depth, and reliability of insights generated by Large Language Models (LLMs). It functions as a web service that accepts user prompts and optional contextual documents, orchestrating multiple user-selected LLMs through predefined, multi-stage "Analysis Patterns." These patterns guide the LLMs through structured processes like critique, fact-checking, perspective-taking, or confidence scoring, culminating in a synthesized final output generated by a designated "Ultra" LLM.

The system's architecture comprises a frontend interface for user interactions, a sophisticated backend that handles orchestration logic, document processing capabilities, and integrated pricing mechanisms. The core innovation lies in the Pattern-Based Multi-LLM Orchestration methodology, which utilizes structured analysis patterns with sequential processing stages and pattern-specific prompt templates to direct the flow of information across multiple models.

Brief Description of the Drawings

Figure 1: Data Flow Diagram (DFD) illustrating the overall system architecture and information flow of the UltrLLMOrchestrator System (UltrAI).

Figure 2: Codebase structure diagram showing the organization of the application's key components, modules, and their relationships.

Detailed Description of the Invention

1. System Overview

UltrAI is an advanced AI analysis platform designed to enhance the quality, depth, and reliability of insights generated by Large Language Models (LLMs). The system accepts user prompts and optional contextual documents, orchestrating multiple user-selected LLMs through predefined, multi-stage "Analysis Patterns" to generate a synthesized output of superior analytical quality compared to single-LLM approaches.

2. System Architecture

The UltrAI system employs a client-server web architecture consisting of:

2.1 Frontend

- A React-based Single Page Application (SPA) built with TypeScript and Vite
- Styled using Tailwind CSS and UI component libraries
- Provides interfaces for prompt input, document upload/management, model/pattern selection, result viewing, and history management
- Implements responsive design for cross-device compatibility

2.2 Backend

- Python API built with the FastAPI framework

- Serves as the central orchestrator for all system operations
- Handles API requests and manages application state
- Interacts with the database and external LLM providers
- Enforces business/pricing logic and serves data to the frontend
- Modularized using FastAPI routers for different functional areas

2.3 Core Orchestration Engine (PatternOrchestrator)

- Central processing unit that receives requests from the backend API
- Interprets the selected Analysis Pattern
- Manages the sequence of calls to multiple external LLM APIs according to the pattern's stages and templates
- Directs the final synthesis step using the designated "Ultra" LLM

2.4 Document Processing Module (UltraDocumentsOptimized)

- Receives uploaded files and extracts text content
- Performs document chunking when necessary
- Stores and retrieves documents for use as context in analysis tasks

2.5 Pricing & Business Logic Module

- Calculates costs based on token usage, model choice, tiers, and features
- Implements business model logic including markups, discounts, token efficiency factors
- Handles user authorization, usage tracking, and billing integration

2.6 Database

- Provides persistence for user accounts, document metadata, document chunks
- Stores usage history, pricing configurations, and cached analysis results
- May store shared analysis links for collaborative features

2.7 External LLM Services Integration

- Connects securely via API keys to various third-party LLM providers
- May include providers such as OpenAI, Anthropic, Google, and Mistral

2.8 Caching Layer

- In-memory TTL cache (cachetools) on the backend
- Stores and retrieves results for identical analysis requests
- Optimizes performance and reduces costs

2.9 Error Monitoring

- Integrates with error tracking services (e.g., Sentry)
- Provides real-time error tracking and performance monitoring

3. Detailed Features

3.1 Core Analysis & Orchestration

3.1.1 Multi-LLM Orchestration

The system executes analysis tasks across a user-selected set of LLMs, leveraging the strengths of different models to enhance the quality of generated insights. The orchestration process involves:

- Distributing prompt processing across multiple LLMs
- Managing parallel or sequential execution of LLM calls
- Collecting and organizing intermediate outputs
- Tracking token usage across multiple API calls

3.1.2 Pattern-Based Analysis

The system guides LLM interaction using predefined, structured patterns, each with unique multi-stage workflows, prompt templates, and instructions. Analysis patterns include:

- Confidence Analysis: Evaluates agreement/disagreement across multiple LLMs to score confidence in outputs
- Critique Analysis: Implements structured feedback loops for critical evaluation
- Fact Check Analysis: Verifies factual claims across multiple models
- Perspective Analysis: Examines topics from multiple viewpoints
- Scenario Analysis: Explores potential outcomes and implications
- Stakeholder Analysis: Maps impacts and considerations across different stakeholder groups
- Systems Mapper: Identifies system components, relationships, and feedback loops
- Time Horizon Analysis: Examines short, medium, and long-term implications
- Innovation Bridge: Connects disparate concepts to generate novel insights
- Gut Analysis: Focuses on intuitive responses with structured evaluation

Each pattern comprises multiple sequential processing stages (e.g., initial, meta, hyper, ultra) with stage-specific templates and instructions.

3.1.3 "Ultra" Model Synthesis

The system uses a user-selected "Ultra" LLM to synthesize intermediate results into a final, enhanced output according to the chosen pattern. This synthesis process:

- Aggregates outputs from multiple base LLMs
- Applies pattern-specific synthesis instructions
- Formats the final output according to user preferences
- May highlight areas of model agreement/disagreement

3.1.4 Model Selection Interface

The frontend allows users to select:

- Multiple base LLMs from available providers
- A single Ultra LLM for final synthesis
- The backend dynamically provides available model information via API endpoint

3.2 Document Handling

3.2.1 Frontend Upload Interface

The system provides a user interface for:

- Selecting and uploading multiple files
- Validating file sizes (e.g., enforcing a 4MB limit)
- Displaying upload progress indicators
- Managing previously uploaded documents

3.2.2 Backend Document Processing

The system handles document processing through:

- File storage in appropriate file systems or databases
- Text extraction from various file formats
- Intelligent document chunking for effective LLM context utilization
- Metadata extraction and storage

3.2.3 Contextual Analysis

The system enables users to:

- Include uploaded documents as context for analysis requests
- Select specific documents to include in the analysis context
- Potentially link documents to user sessions for persistent context

3.3 Pricing, Billing & Business Model

3.3.1 Tiered Pricing Structure

The system implements:

- Multiple user tiers (e.g., basic, pro, enterprise)
- Tier-specific markups, minimum charges, and feature access
- Potentially automatic tier recommendations based on usage patterns

3.3.2 Usage-Based Cost Calculation

The system calculates costs based on:

- Input/output tokens per model
- User tier and associated pricing rules
- Applied discounts for volume or prepaid reserves
- Feature-specific surcharges

3.3.3 Feature Add-on Costs

The system applies specific charges for optional features such as:

- Private processing
- Enhanced sourcing capabilities
- Data encryption
- Additional analysis options

3.3.4 Volume & Reserve Discounts

The system provides cost reductions based on:

- Query volume thresholds
- Pre-paid reserve amounts
- Potentially subscription-based pricing options

3.3.5 Token Efficiency Concept

The system incorporates:

- Model-specific efficiency factors in business simulations
- Value assessment beyond raw token cost
- Potential optimization recommendations

3.3.6 Cost Estimation & Authorization

The API:

- Checks estimated cost against user balance/tier before executing analysis
- Provides cost estimates to users before processing
- Enforces usage limits based on account settings

3.3.7 Usage Tracking & Reporting

The system:

- Records detailed usage per user/request
- Generates summary reports
- Provides usage visualizations and insights

3.3.8 Business Model Simulation

The system includes tools to:

- Analyze profitability of different pricing strategies
- Generate pricing recommendations
- Simulate various business scenarios

3.3.9 User Account & Billing Management

The system supports:

- Account creation and management
- Tier assignment and modification
- Balance viewing and funds addition
- Usage history access

3.4 User Interface & Experience

3.4.1 Step-by-Step Workflow

The frontend guides users through a structured process:

- Introduction to the system capabilities
- Prompt input
- Document selection
- Model selection
- Analysis pattern selection
- Additional options configuration
- Processing status display
- Results presentation

3.4.2 Input Components

The user interface provides:

- Text areas for entering prompts
- Checkboxes/selectors for choosing models
- Pattern selection interfaces with descriptions
- Option configuration controls
- Visual indicators for selections

3.4.3 Visual Feedback

The system includes:

- Progress bars for long-running operations
- Status messages during processing
- Animated elements to indicate system activity
- Clear success/error indications

3.4.4 Result Display

The system presents:

- Clearly formatted final synthesized output
- Automatic scrolling to results when ready
- Potentially expandable sections for intermediate outputs
- Options to copy, share, or save results

3.4.5 History Management

The system provides:

- A panel to view past interactions
- Functionality to load previous analyses
- Options to delete history items
- Local storage of history data

3.4.6 Sharing Functionality

The system enables:

- Generation of unique, persistent shareable links
- Copy functionality for sharing links
- Potential access control for shared analyses

3.4.7 Status Indicators

The interface includes:

- Offline detection and notification
- API connection status
- Processing status indicators
- Error message display

3.4.8 Responsive Design

The interface:

- Adapts to different screen sizes
- Maintains usability across devices
- Optimizes layout for mobile and desktop views

3.4.9 Dynamic Cost Display

The system may include:

- Floating price component
- Real-time cost estimation based on selections
- Tier comparison information

3.5 System Operations & Development

3.5.1 API Architecture

The system implements:

- Well-defined RESTful API endpoints using FastAPI
- Modular router structure for different functional areas
- Standardized request/response formats
- Comprehensive API documentation

3.5.2 Testing & Development Features

The system includes:

- Mock mode for testing without incurring LLM API costs
- Health and metrics endpoints
- Performance monitoring capabilities
- Comprehensive testing infrastructure

3.5.3 Performance Optimization

The system utilizes:

- Backend caching for repeated identical requests
- Asynchronous operations for efficient I/O handling
- Optimized document processing
- Efficient token usage strategies

3.5.4 Configuration & Deployment

The system supports:

- Configuration via environment variables and/or files
- Deployment options including Vercel and Docker/Docker Compose
- Scalable architecture for varying load conditions
- Potential cloud service provider integrations

3.5.5 Security Implementation

The system includes:

- CORS implementation for API security
- Potential rate limiting to prevent abuse
- Authentication mechanisms (potentially OAuth)
- Secure handling of API keys and sensitive data

4. Novel Technical Aspects

The UltrAI system incorporates several novel technical aspects:

4.1 Pattern-Based Multi-LLM Orchestration

A defined methodology involving:

- Structured "Analysis Patterns" with distinct purposes and workflows
- Multiple, sequential processing stages within each pattern
- Pattern-specific prompt templates and instructions
- Orchestration logic that manages data flow across multiple LLMs
- Synthesis by a designated "Ultra" LLM guided by pattern-specific instructions

4.2 Analysis Pattern Implementations

Unique logic and processing steps within each pattern, such as:

- Cross-model agreement evaluation in Confidence Analysis
- Structured feedback loops in Critique Analysis
- Stakeholder mapping in Stakeholder Vision
- System component and feedback loop identification in Systems Mapper
- Temporal reframing in Time Horizon Analysis

4.3 Contextual Document Integration

Methods for:

- Processing uploaded documents through extraction and chunking
- Integrating relevant context into appropriate stages of the analysis workflow
- Optimizing context utilization across multiple LLM calls
- Managing context windows efficiently

4.4 Token Efficiency Normalization

A system for:

- Applying model-specific efficiency factors to normalize cost calculations
- Accounting for varying token requirements across different models
- Incorporating efficiency metrics in business model simulations
- Potentially optimizing model selection based on efficiency factors

4.5 Integrated Pricing Simulation

Computational methods for:

- Simulating business viability with multiple variable factors
- Combining base costs, markups, discounts, and efficiency metrics
- Generating optimal pricing structure recommendations
- Supporting business strategy development

4.6 Dynamic, Tiered Cost Calculation

A system that calculates costs by integrating:

- Base token costs for different LLM providers
- Tiered percentage markups
- Minimum charges per request
- Volume-based discounts
- Feature-specific add-on fees
- Model-type surcharges
- User authorization based on calculated costs

Claims

[While formal claims are not required for a provisional patent application, the applicant reserves the right to include detailed claims in a subsequent non-provisional application covering the novel aspects described above.]

Drawings

Figure 1: Data Flow Diagram (DFD) of the UltrLLMOrchestrator System

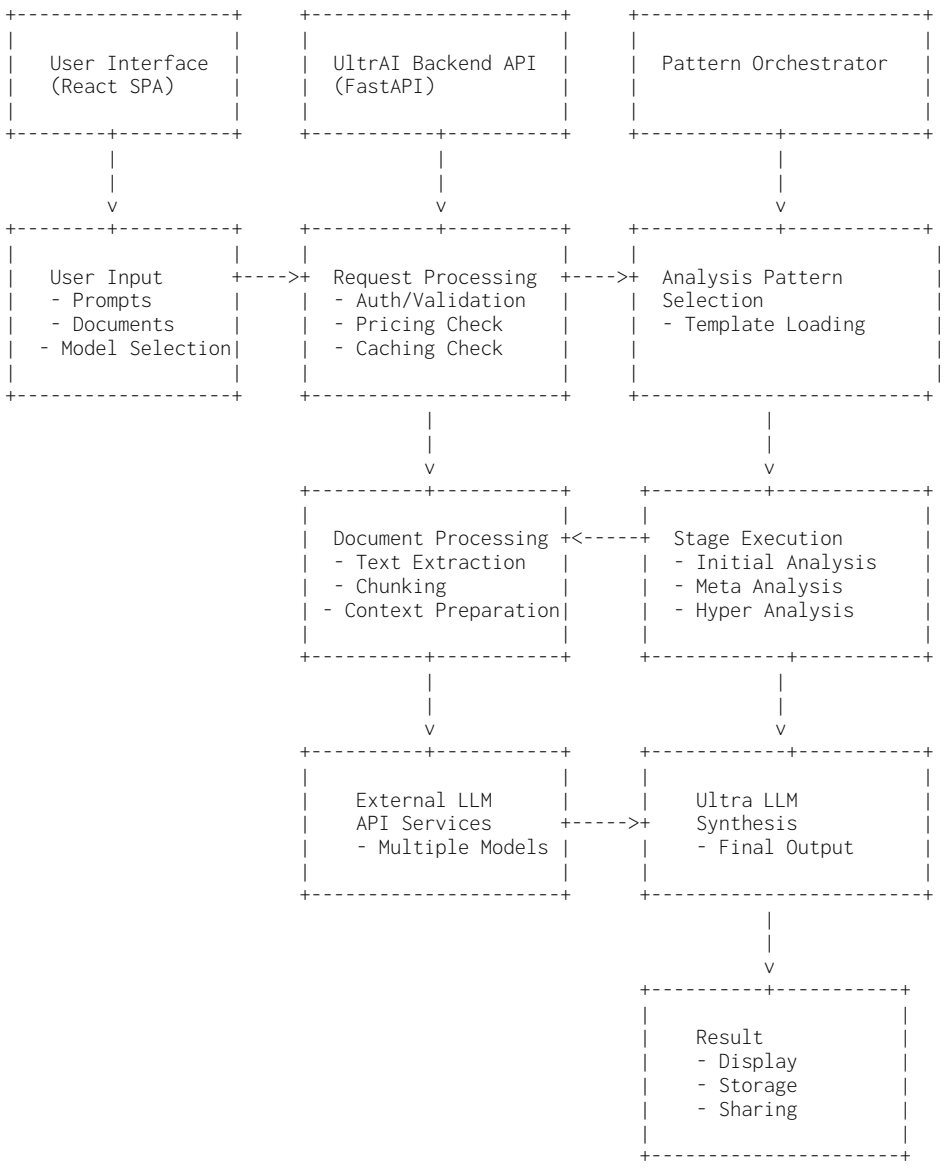
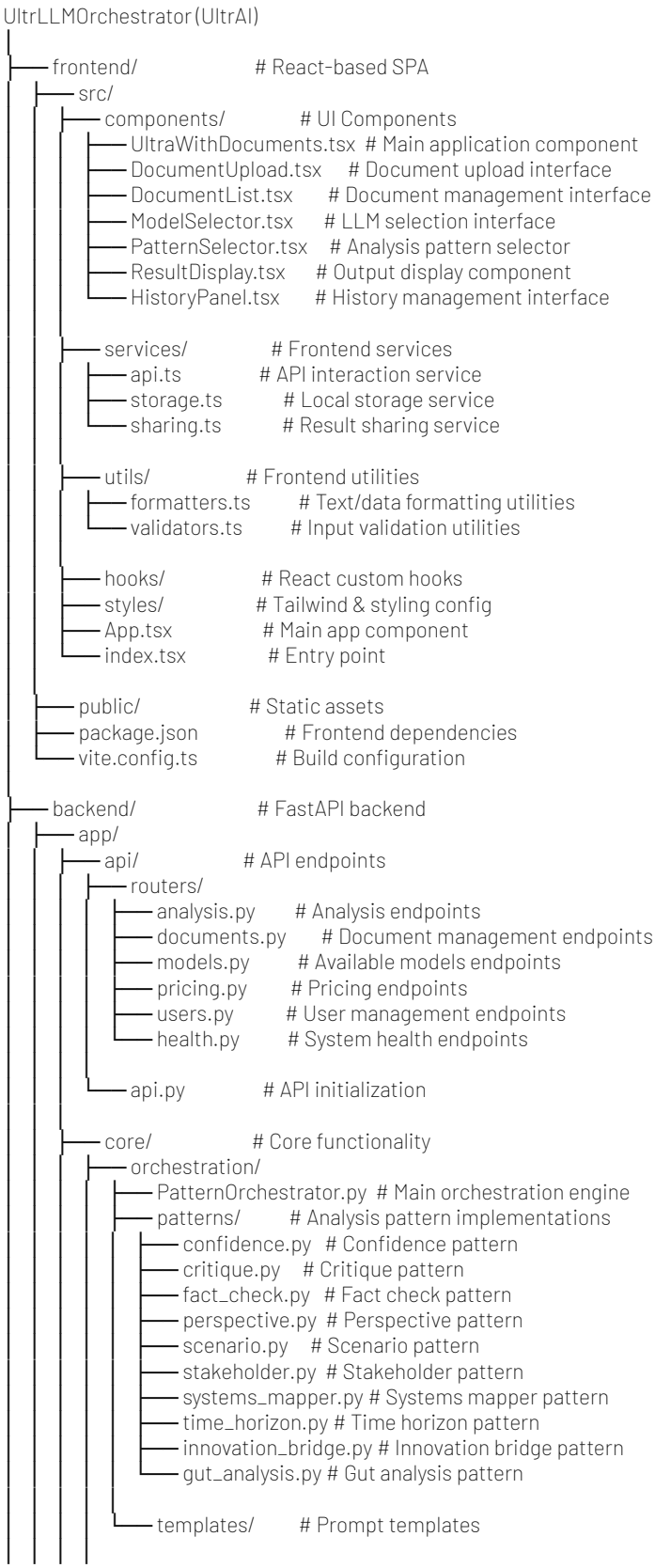
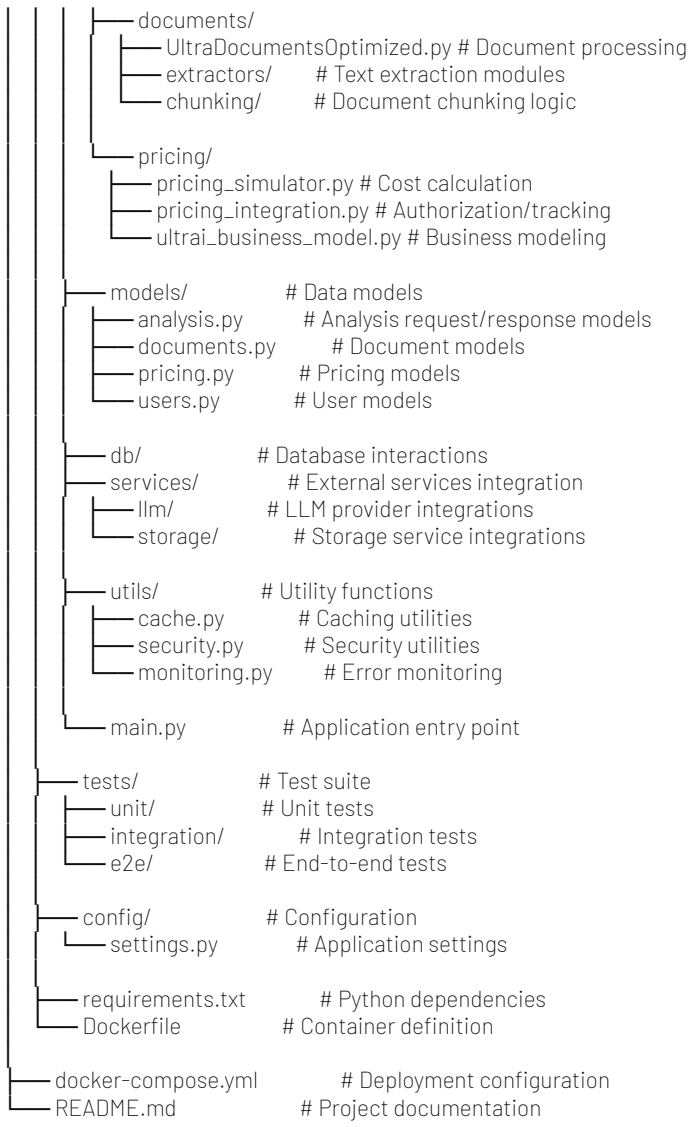


Figure 2: Codebase Structure Diagram





Abstract

The UltrLLMOrchestrator System (UltrAI) is an advanced AI analysis platform that enhances the quality, depth, and reliability of insights generated by Large Language Models (LLMs). The system orchestrates multiple user-selected LLMs through predefined, multi-stage "Analysis Patterns" to guide structured analytical processes. Each pattern implements unique workflows with sequential processing stages and pattern-specific prompt templates. The system includes document processing capabilities for contextual analysis, an "Ultra" LLM synthesis stage, and integrated pricing mechanisms with tiered structure and token efficiency considerations. The web-based implementation features a React frontend and FastAPI backend with comprehensive user interface elements for prompt input, document management, model selection, and result visualization.