Praval Deep Research - User Manual

Built with Praval Framework

November 2025

Contents

Praval Deep Research - User Manual

Version 1.0
A Local-First, AI-Powered Research Assistant

Built with Praval Agentic Framework

Table of Contents

1. Introduction
2. Getting Started
3. Installation
4. Quick Tour
5. Discovering Papers
6. Chat Interface
7. Knowledge Base Management

Introduction

What is Praval Deep Research?

Praval Deep Research is a **completely local**, privacy-focused research assistant designed to help you discover, analyze, and understand academic papers from ArXiv. Unlike cloud-based solutions where your data is sent to external servers, everything in Praval Deep Research runs on your own machine.

Key Benefits

- **Privacy First**: All your research data stays on your computer
- **No Internet Required**: Once papers are downloaded, work completely offline
- **Intelligent Agents**: 6 specialized AI agents help you research efficiently
- **Semantic Search**: Find relevant information across all your papers
- **Chat History**: Never lose track of your research conversations
- **PDF Access**: Read papers directly in your browser

System Requirements

Minimum: - Docker & Docker Compose installed - 4GB RAM available - 5GB free disk space - OpenAI API key (for embeddings and Q&A)

Recommended: - 8GB+ RAM - 10GB+ free disk space - Modern web browser (Chrome, Firefox, Edge, Safari) - Stable internet connection for initial paper downloads

---

Getting Started

Prerequisites

Before installing Praval Deep Research, ensure you have:

1. **Docker Desktop** installed
   - Download from: https://www.docker.com/products/docker-desktop
   - Verify installation: `docker --version`
2. **Docker Compose** (usually included with Docker Desktop)
   - Verify installation: `docker-compose --version`
3. **OpenAI API Key**
   - Sign up at: https://platform.openai.com
   - Create an API key in your dashboard
   - Keep your key secure - you'll need it during setup

Installation

Step-by-Step Installation Guide

1. Download the Project

```
# Clone the repository
git clone https://github.com/aiexplorations/praval_deep_research.git

# Navigate to the project directory
cd praval_deep_research
```

2. Configure Environment

```
# Copy the environment template
cp .env.example .env

# Open the .env file in your favorite editor
nano .env    # or use: vim .env, code .env, etc.
```

3. Add Your OpenAI API Key   In the `.env` file, find the line:

```
OPENAI_API_KEY=your_key_here
```

Replace `your_key_here` with your actual OpenAI API key:

```
OPENAI_API_KEY=sk-proj-...your-actual-key...
```

Save and close the file.

4. Start the Application

```
# Build and start all services (first time takes 5-10 minutes)
docker-compose up -d

# Wait for all services to become healthy (about 1-2 minutes)
docker-compose ps
```

You should see output showing all services as "Up" or "healthy":

```
research_api        Up (healthy)
research_frontend   Up (healthy)
research_qdrant     Up
research_minio      Up (healthy)
research_rabbitmq   Up (healthy)
research_redis      Up (healthy)
```

5. Access the Application   Open your web browser and navigate to:

```
http://localhost:3000
```

You should see the Praval Deep Research interface with three main sections: - **Discover**: Search and index ArXiv papers - **Chat**: Ask questions about your indexed papers - **Knowledge Base**: Manage your paper collection

Quick Tour

Main Interface

The application has three main pages accessible from the top navigation:

1. **Discover** - Search ArXiv and add papers to your knowledge base
2. **Chat** - Ask questions and have conversations about your papers
3. **Knowledge Base** - View and manage all indexed papers

Top Right Corner: - Praval Logo: Links to pravalagents.com (the framework powering this app)

Your First Research Session

Let's walk through a complete research workflow:

Step 1: Find Papers (2 minutes)

1. Click **"Discover"** in the navigation
2. Enter a research query, e.g., "transformer attention mechanisms"
3. Click **"Search Papers"**
4. Select 2-3 relevant papers by clicking their checkboxes
5. Click **"Index Selected Papers"**
6. Watch the progress as papers are downloaded and processed

Step 2: Ask Questions (1 minute)

1. Click **"Chat"** in the navigation
2. Type a question: "What are the main innovations in transformer architectures?"
3. Press Enter or click **"Ask"**
4. Read the answer with source citations
5. Try the suggested follow-up questions

Step 3: Manage Your Collection (1 minute)

1. Click **"Knowledge Base"** in the navigation
2. See all your indexed papers in a table
3. Click **"View PDF"** on any paper to read it
4. View statistics: total papers, vectors, categories

Congratulations! You've completed your first research session.

Discovering Papers

Searching ArXiv

The **Discover** page is your starting point for finding research papers.

How to Search

1. **Enter your query** in the search box
   - Examples:
     - "neural networks"
     - "reinforcement learning"
     - "attention mechanisms in transformers"
     - "deep learning computer vision"
2. **Set maximum results** (optional)
   - Default: 10 papers
   - Range: 1-50 papers
3. **Click "Search Papers"**
   - Results appear in 1-3 seconds
   - Papers are ranked by relevance

Understanding Search Results   Each paper shows: - **Title**: The paper's full title - **Authors**: First few authors (click to see all) - **Abstract**: Brief summary of the paper - **Published**: Date published on ArXiv - **Category**: Subject classification (e.g., cs.AI, cs.LG) - **ArXiv ID**: Unique identifier (e.g., 2203.14263v1)

Selecting Papers to Index   **Best Practices:** - Start with 3-5 papers for a focused topic - Read abstracts carefully before selecting - Check publication dates (newer papers may reference older ones) - Look for highly cited works in your field

To Select Papers: 1. Click the checkbox next to each paper you want 2. Select multiple papers at once 3. Click **"Index Selected Papers"** when ready

The Indexing Process   When you click "Index Selected Papers", the system:

1. **Downloads PDFs** from ArXiv
2. **Extracts text** from each PDF
3. **Chunks text** into manageable pieces (1000 characters each)
4. **Generates embeddings** using OpenAI
5. **Stores vectors** in your local database

Time Required: - ~30-60 seconds per paper - Processing happens in parallel - You can see progress in real-time

What You See: - "Processing paper X of Y…" - "Downloading PDF…" - "Extracting text…" - "Generating embeddings…" - " Complete"

Troubleshooting Search   **No results found:** - Try simpler queries - Remove quotes from search terms - Try different keywords

Search timeout: - Check your internet connection - ArXiv may be temporarily slow - Try again in a few minutes

Can't select papers: - Make sure papers loaded successfully - Refresh the page if checkboxes are unresponsive

Chat Interface

Having Conversations

The **Chat** page is where you ask questions and explore your research papers through natural conversations.

Creating Conversations

Starting a New Chat: 1. Click **"+ New Chat"** button in the sidebar 2. A fresh conversation begins 3. Type your first question 4. The system automatically creates and names your conversation

Auto-Generated Titles: - After your first question, the system generates a smart title - Titles are descriptive (e.g., "Understanding Transformer Architecture") - Like ChatGPT/Claude, titles help you find conversations later - Titles appear in the sidebar automatically

Asking Questions

Question Box: - Located at the bottom of the chat - Type your question naturally - Press Enter or click "Ask" to submit

Good Questions: - "What is attention mechanism in transformers?" - "How does BERT differ from GPT?" - "What are the main contributions of this paper?" - "Compare the approaches in these papers" - "What are the limitations discussed?"

Tips for Better Answers: - Be specific: "How does self-attention work?" vs "Tell me about attention" - Ask follow-ups: Build on previous answers - Reference specific papers: "According to the transformer paper…"

Understanding Answers

Answer Structure: Each answer includes:

1. **Main Response**: Direct answer to your question
2. **Source Citations**: Papers used to generate the answer
3. **Relevance Scores**: How relevant each source is (0.0-1.0)
4. **Follow-up Questions**: 3 suggested related questions

Reading Citations:

```
Sources (5):
  [1] An Optimal Control View of Adversarial Machine Learning
  [2] YETI: Proactive Interventions by Multimodal AI Agents
  [3] An Optimal Control View of Adversarial Machine Learning
```

Each source shows: - Paper title - Relevance score (higher = more relevant) - Paper details (click to expand)

Managing Conversations

Sidebar Features:

Conversation List: - Shows all your chat threads - Most recent at the top - Click any conversation to load it - See message count for each

Loading Conversations: 1. Click any conversation in the sidebar 2. All messages load instantly 3. Continue where you left off 4. Add more questions to the thread

Deleting Conversations: 1. Hover over a conversation 2. Trash icon appears on the right 3. Click the trash icon 4. Confirm deletion 5. Conversation permanently removed

Current Conversation Indicator: - Active conversation highlighted in blue - Easy to see which chat you're in

Conversation Tips

Organizing Your Research: - Create separate conversations for different topics - Use descriptive first questions (they become the title) - Delete old conversations you no longer need

Effective Research Conversations: 1. Start broad: "What is this paper about?" 2. Go deeper: "How does this specific method work?" 3. Compare: "How does this compare to other approaches?" 4. Critique: "What are the limitations?" 5. Apply: "How could this be used for…?"

What the System Remembers: - All messages in the current conversation - Source papers used previously - Context from your questions - This helps provide better follow-up answers

---

Knowledge Base Management

Viewing Your Collection

The **Knowledge Base** page is your library of indexed research papers.

Statistics Dashboard

At the top of the page, you'll see four key metrics:

1. **Total Papers**: Number of papers indexed
2. **Total Vectors**: Number of text chunks stored
3. **Avg Chunks/Paper**: Average pieces per paper
4. **Categories**: Different subject areas

What These Mean: - More papers = broader knowledge - More vectors = more detailed coverage - Higher avg chunks = longer papers - More categories = diverse topics

Paper Table

Columns: - **Title**: Paper name (click to sort) - **Authors**: First few authors - **Chunks**: Number of text pieces - **Category**: ArXiv classification - **Actions**: View PDF and Delete buttons

Sorting: - Click any column header to sort - Click again to reverse sort order - Helps find specific papers quickly

Searching: - Use the search box to filter papers - Searches across titles and authors - Results update as you type

Viewing PDFs

Opening Papers: 1. Find the paper in the table 2. Click the **" View PDF"** button 3. PDF opens in a new browser tab 4. Read, scroll, zoom normally

PDF Features: - Opens in your browser's PDF viewer - Full PDF functionality - No download required - Stays on your machine (served locally)

Troubleshooting PDF Viewing: - If PDF doesn't open, paper may not be fully indexed - Try re-indexing the paper from Discover page - Check browser allows pop-ups from localhost

Managing Papers

Deleting Individual Papers: 1. Find the paper in the table 2. Click **"Delete"** button 3. Confirm deletion 4. Paper and all its vectors removed 5. Cannot be undone

Clearing All Papers: 1. Click **"Clear All Papers"** button (top right) 2. First confirmation: "Are you sure?" 3. Second confirmation: "This cannot be undone" 4. All papers and vectors deleted 5. Fresh start for new research

When to Delete Papers: - Paper not relevant after reading - Too many papers (slowing down search) - Starting a new research topic - Cleaning up old work

Refreshing Data: - Click **" Refresh"** button - Reloads current statistics - Updates paper list - Use after indexing new papers from Discover

Knowledge Base Best Practices

Organizing Your Research:

By Topic: - Keep 10-20 papers per research topic - Clear old papers when switching topics - Index papers progressively as needed

By Project: - Index papers for current project only - Export/save important conversations - Clear and start fresh for new projects

Storage Management: - Each paper: ~1-5 MB (PDF) + ~2-5 MB (vectors) - 100 papers: ~300-500 MB total - Monitor disk space in Docker settings - Clear old papers to free space

---

Advanced Features

Conversation Persistence

How It Works: - All conversations automatically saved to Redis - Messages persist across browser sessions - Reload page without losing work - Conversations survive container restarts

Auto-Loading: - Most recent conversation loads automatically - Continue research seamlessly - No manual saving required

Data Location: - Stored in Docker volume `redis_data` - Backed up with Docker volume backups - Deleted only when you delete conversation

Smart Title Generation

LLM-Powered Titles: - System uses GPT-4o-mini to generate titles - Analyzes first question and answer - Creates 5-10 word descriptive title - Updates automatically after first exchange

Title Format: - Specific and descriptive - Easy to scan in sidebar - Helps identify conversations later - Examples: - "Understanding Transformer Architecture Components" - "Comparing RL Approaches in Robotics" - "BERT vs GPT: Key Differences"

PDF Proxy Serving

Technology: - PDFs served through FastAPI backend - No direct MinIO access from browser - Solves signature and CORS issues - Secure and efficient

Benefits: - Always works (no expired URLs) - Fast loading - Browser-native PDF viewer - No external dependencies

Real-Time Processing Updates

Server-Sent Events (SSE): - Live progress during indexing - See each step as it happens - No page refresh needed - Detailed status messages

What You See: - Current paper being processed - Step in the workflow - Success/error messages - Completion status

Search & Filter

Knowledge Base Search: - Type in search box - Filters papers in real-time - Searches: titles, authors, abstracts - Case-insensitive matching

Results: - Matching papers highlighted - Non-matching papers hidden - Count shown at bottom - Clear search to see all again

---

Tips & Best Practices

Effective Research Workflow

1. Start Focused - Begin with 3-5 key papers - Understand them deeply first - Then expand to related work

2. Index Progressively - Don't index everything at once - Add papers as you need them - Quality over quantity

3. Organize Conversations - One conversation per topic - Use descriptive first questions - Delete old conversations

4. Use Follow-Up Questions - System suggests 3 related questions - Help explore topics deeply - Uncover connections between papers

Optimizing Performance

Faster Indexing: - Index 3-5 papers at a time - Don't index during heavy queries - Good internet connection helps

Faster Q&A: - Fewer papers = faster search - More specific questions = better answers - Use conversation context

Storage Management: - Monitor Docker volume sizes - Clear old papers regularly - Backup important conversations

Data Privacy & Security

What Stays Local: - All PDFs (in MinIO) - All vector embeddings (in Qdrant) - All conversations (in Redis) - All metadata (in PostgreSQL)

What Goes External: - ArXiv API calls (downloading papers) - OpenAI API calls (embeddings & Q&A) - Only paper text and questions sent

Securing Your Data: - Keep OpenAI API key secure - Use Docker volume encryption - Regular backups - Don't expose ports publicly

Best Practices for Questions

Good Question Patterns:

Explanatory: - "What is [concept]?" - "How does [method] work?" - "Explain [technique] in simple terms"

Comparative: - "What's the difference between X and Y?" - "How does this compare to [other paper]?" - "What are the pros/cons of each approach?"

Analytical: - "What are the limitations of [method]?" - "What assumptions does this paper make?" - "What are potential applications?"

Synthesizing: - "What do these papers agree on?" - "What are the conflicting views?" - "What's the current state of research?"

Troubleshooting

Common Issues

Services Won't Start Symptoms: - Error when running `docker-compose up` - Containers exit immediately - Health checks failing

Solutions:

```
# Check Docker is running
docker ps

# Check for port conflicts
lsof -i :3000   # Frontend
lsof -i :8000   # API
```

```
lsof -i :9000   # MinIO


# Stop conflicting services
kill <PID>   # From lsof output


# Clean restart
docker-compose down
docker-compose up -d
```

Frontend Blank Page Symptoms: - Browser shows empty page - No UI elements visible - Console errors in DevTools

Solutions:

```
# Hard refresh browser (clears cache)
# Chrome/Edge: Ctrl+Shift+R (Windows) or Cmd+Shift+R (Mac)
# Firefox: Ctrl+F5


# Check container is running
docker-compose ps research_frontend


# Check nginx logs
docker-compose logs research_frontend


# Rebuild if needed
docker-compose build research_frontend
docker-compose up -d research_frontend
```

Q&A Returns No Results Symptoms: - Questions return "no relevant papers found" - Empty source list - Generic answers

Solutions:

```
# Verify papers are indexed
curl http://localhost:8000/research/knowledge-base/stats


# Should show:
# "total_papers": > 0
# "total_vectors": > 0


# If zero, index papers from Discover page


# Check Qdrant has data
curl http://localhost:6333/collections/research_vectors


# Restart if needed
docker-compose restart research_api
```

Chat History Not Saving   Symptoms: - Conversations disappear on refresh - "New Chat" doesn't create conversation - No conversations in sidebar

Solutions:

```
# Check Redis is running
docker-compose ps research_redis

# Test Redis connection
docker-compose exec research_redis redis-cli ping
# Should return: PONG

# Check conversation API
curl http://localhost:8000/research/conversations
# Should return list of conversations

# Restart if needed
docker-compose restart research_redis research_api
```

PDF Won't Open   Symptoms: - "PDF not found" error - Blank page when clicking View PDF - Download fails

Solutions:

```
# Check MinIO is accessible
curl http://localhost:9000/minio/health/live

# Test PDF endpoint
curl -I http://localhost:8000/research/knowledge-base/papers/[paper_id]/pdf

# Should return: HTTP 200 OK

# Check MinIO logs
docker-compose logs research_minio

# Restart if needed
docker-compose restart research_minio research_api
```

Indexing Fails   Symptoms: - Papers stuck at "Processing…" - Error messages during indexing - Papers not appearing in Knowledge Base

Solutions:

```
# Check all services healthy
docker-compose ps

# View API logs for errors
docker-compose logs -f research_api

# Common issues:
```

```
# - OpenAI API key invalid
# - OpenAI rate limit hit
# - ArXiv temporarily unavailable
# - Disk space full

# Check OpenAI key
grep OPENAI_API_KEY .env

# Restart services
docker-compose restart research_api
```

Getting Help

Log Files:

```
# View all logs
docker-compose logs

# Specific service
docker-compose logs research_api
docker-compose logs research_frontend

# Follow logs (live)
docker-compose logs -f research_api

# Last 100 lines
docker-compose logs --tail=100 research_api
```

System Status:

```
# Container status
docker-compose ps

# Resource usage
docker stats

# Disk usage
docker system df

# Volume sizes
docker volume ls
docker system df -v
```

Health Checks:

```
# API health
curl http://localhost:8000/health

# Frontend
curl http://localhost:3000
```

```
# Qdrant
curl http://localhost:6333/health

# MinIO
curl http://localhost:9000/minio/health/live
```

FAQ

General Questions

Q: Do I need an internet connection?

A: You need internet for: - Downloading papers from ArXiv - OpenAI API calls (embeddings & Q&A)

Once papers are indexed, you can work offline for reading PDFs and browsing your knowledge base. Q&A requires OpenAI API access.

Q: How much does it cost to run?

A: Costs are only for OpenAI API usage: - Embeddings: ~$0.0001 per paper - Q&A: ~$0.001-0.005 per question - Example: 100 papers + 100 questions = ~$0.50-1.00

The software itself is free and open-source.

Q: Can I use a different LLM?

A: Currently requires OpenAI. Future versions will support: - Anthropic Claude - Local models (Ollama) - Custom endpoints

Q: How private is my data?

A: Very private: - All papers stored locally - Vectors never leave your machine - Only text sent to OpenAI for processing - No telemetry or analytics - No cloud backups unless you create them

Technical Questions

Q: Can I run this on a server?

A: Yes! The system is designed for local use but can run on any machine with Docker. For remote access: - Use SSH tunnel: `ssh -L 3000:localhost:3000 user@server` - Set up reverse proxy (nginx/Caddy) - Configure authentication

Q: How do I backup my data?

A: Backup Docker volumes:

```
# Backup all data
docker run --rm \
  -v praval_deep_research_qdrant_data:/data \
  -v $(pwd):/backup \
  alpine tar czf /backup/backup.tar.gz /data
```

```
# Restore
docker run --rm \
  -v praval_deep_research_qdrant_data:/data \
  -v $(pwd):/backup \
  alpine tar xzf /backup/backup.tar.gz -C /
```

Q: Can I index my own PDFs?

A: Not directly in v1.0. Future versions will support: - Custom PDF upload - Local PDF folder monitoring - Non-ArXiv sources

Q: What happens if I lose my API key?

A: If you lose your OpenAI API key: 1. Generate new key at platform.openai.com 2. Update `.env` file 3. Restart containers: `docker-compose restart`

Your indexed papers remain, but you can't: - Index new papers - Ask questions (need embeddings)

Q: How do I update to a new version?

A:

```
# Pull latest code
git pull origin main

# Rebuild containers
docker-compose build

# Restart with new version
docker-compose down
docker-compose up -d

# Your data persists in volumes
```

Q: Can multiple people use the same instance?

A: Not in v1.0. The system is single-user. Multi-user support planned for future versions with: - User authentication - Private knowledge bases - Shared conversations

Troubleshooting Questions

Q: Why is indexing so slow?

A: Indexing speed depends on: - Internet connection (downloading PDFs) - OpenAI API rate limits - System resources (RAM/CPU) - Number of papers

Average: 30-60 seconds per paper.

Q: Why are my questions returning generic answers?

A: Common causes: - No papers indexed (check Knowledge Base) - Question too broad (be more specific) - Papers don't cover the topic - Wrong embedding model (check config)

Q: Can I change the embedding model?

A: Yes, in `.env` file:

```
# Default
OPENAI_EMBEDDING_MODEL=text-embedding-3-small
EMBEDDING_DIMENSIONS=1536

# Alternatives
# text-embedding-3-large (better quality, higher cost)
# text-embedding-ada-002 (legacy)
```

After changing, you need to re-index all papers.

Q: How do I completely reset the system?

A:

```
# Stop and remove everything
docker-compose down -v

# This deletes:
# - All papers
# - All embeddings
# - All conversations
# - All containers

# Start fresh
docker-compose up -d
```

Keyboard Shortcuts

Global

- `Ctrl/Cmd + K`: Focus search (Knowledge Base)
- `Ctrl/Cmd + N`: New chat (Chat page)
- `Ctrl/Cmd + /`: Show keyboard shortcuts
- `Esc`: Close modals/dialogs

Chat Page

- `Enter`: Send message
- `Shift + Enter`: New line in message
- `↑`: Edit last message (future)
- `Ctrl/Cmd + L`: Clear current chat (future)

Knowledge Base

- `Delete`: Delete selected paper (future)
- `Ctrl/Cmd + A`: Select all papers (future)

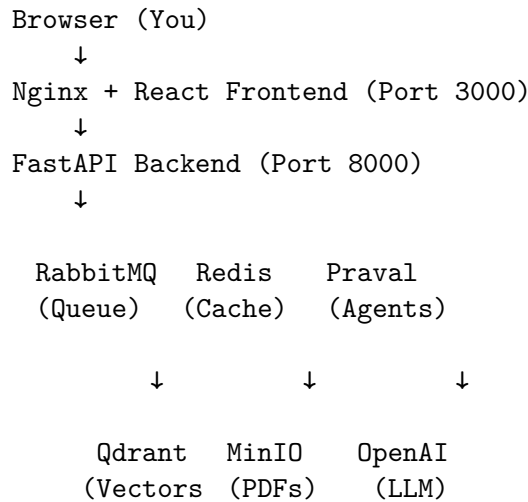### Navigation

- `1`: Go to Discover

- 2: Go to Chat
- 3: Go to Knowledge Base

Note: Some shortcuts marked (future) are planned for upcoming versions.

Appendix

System Architecture

Praval Deep Research uses a microservices architecture:

```
Browser (You)
    ↓
Nginx + React Frontend (Port 3000)
    ↓
FastAPI Backend (Port 8000)
    ↓

  RabbitMQ   Redis    Praval
  (Queue)   (Cache)   (Agents)


        ↓          ↓           ↓


    Qdrant   MinIO    OpenAI
   (Vectors  (PDFs)    (LLM)
```

Data Storage

Docker Volumes: - `qdrant_data`: Vector embeddings (~200MB/1000 chunks) - `minio_data`: PDF files (~1-5MB per paper) - `redis_data`: Conversations (~10KB per conversation) - `rabbitmq_data`: Message queue (~1MB)

Total Storage: - 100 papers   300-500 MB - 1000 papers   3-5 GB

API Endpoints

Paper Management: - `POST /research/search` - Search ArXiv - `POST /research/index` - Index papers - `GET /research/knowledge-base/papers` - List papers - `GET /research/knowledge-base/papers/{` - Get PDF

Q&A: - `POST /research/ask` - Ask question - `GET /research/conversations` - List conversations - `POST /research/conversations` - Create conversation - `DELETE /research/conversations/{id}` - Delete conversation

Full API documentation: http://localhost:8000/docs

Support & Resources

Documentation

- **README**: Project overview and quick start
- **DESIGN.md**: System architecture details
- **CLAUDE.md**: Development guidelines

Online Resources

- **Praval Framework**: https://pravalagents.com
- **GitHub Repository**: https://github.com/aiexplorations/praval_deep_research
- **Issue Tracker**: https://github.com/aiexplorations/praval_deep_research/issues

Getting Help

1. **Check this manual** - Most questions answered here
2. **Check README** - Installation and setup issues
3. **Search GitHub Issues** - Known problems and solutions
4. **Create New Issue** - For bugs and feature requests

Contributing

Contributions welcome! See CONTRIBUTING.md for guidelines.

Thank you for using Praval Deep Research!

Built with   using [Praval Framework](#)

Version 1.0 | Updated November 2025