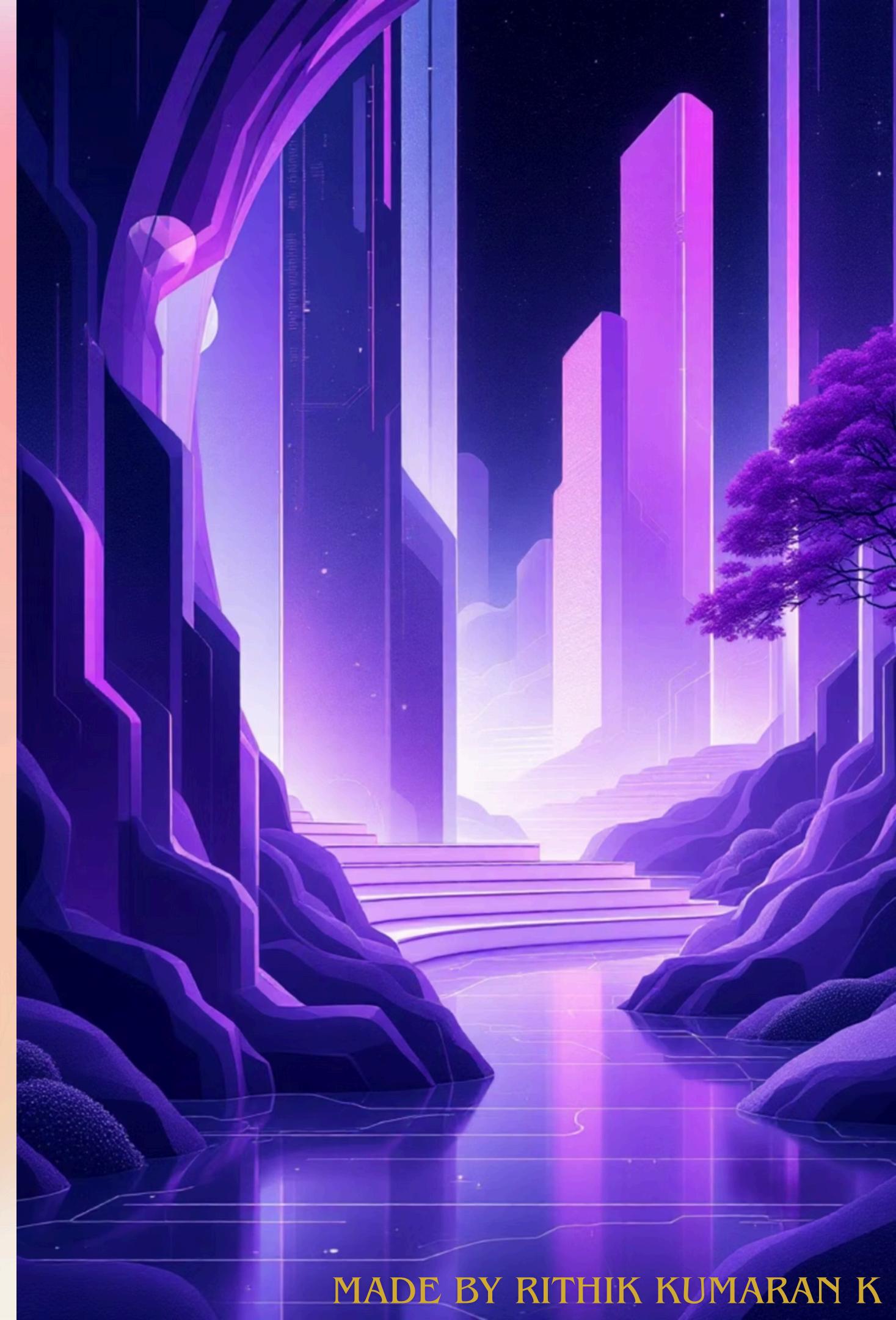


# NeoGlass – Chromatic Summarizer

# Intelligent Summarization & Paraphrasing Suite

A high-performance platform engineered for speed, accuracy, and style in processing information.



## PROBLEM STATEMENT

# The Critical Need for Smarter AI Summarization



### → **Information Overload**

Professionals and students struggle to quickly process massive volumes of text, wasting valuable time and attention.

### → **Generic Output**

Traditional summarizers often fail to capture the original text's core meaning or produce contextually weak outputs.

### → **Lack of Fluency**

Existing paraphrasing tools lack natural, human fluency, resulting in robotic and inauthentic rewrites that require manual editing.

# Powering NeoGlass with Advanced AI & Modern Web Tech



## Python (Streamlit)

Serves as the foundation for the interactive web UI and manages all backend AI/API integration logic.



## Hugging Face Transformers

Employed for cutting-edge extractive and abstractive summarization models, providing deep language understanding.



## GROQ LLaMA 3.1 API

Leveraged for ultra-fast paraphrasing, enabling near-instantaneous content rewriting with high LLM precision.



## dotenv & Custom Pipeline

Used for secure key management and a modular pipeline design to seamlessly integrate summarization and paraphrasing backends.



## Glassmorphic UI

A clean, modern design featuring dynamic shadows, gradient layers, and a minimal aesthetic for a superior user experience.



## Streamlit Caching

Optimized for efficiency, using built-in caching mechanisms to reuse the model pipeline and deliver faster inference times.

# NeoGlass: A Dual Engine for Humanized Summarization



## Abstractive Summarization

Utilizes Transformer-based models to intelligently rewrite and rephrase text, generating coherent, human-like summaries.



## Extractive Summarization

Focuses on factual accuracy by extracting the most relevant and important sentences directly from the source material.



## High-Speed Paraphrasing

Powered by [GROQ Cloud](#), this feature provides instant generation of multiple, high-quality paraphrase variations.



## Modern Glassmorphic UI

Designed for immersion, featuring minimal design, dynamic shadows, and gradient layers for a premium aesthetic.



## Configurable Output Length

Users gain precise control by selecting output length: **Short, Medium, or Long** summaries to fit their specific needs.



## Modular Pipeline

A unified class streamlines integration, managing both summarization and paraphrasing backends through a single, cached resource loader.

## CHALLENGES FACED

# Every Innovation Faces its Storms

### API Integration Conflicts

Asynchronous initialization between Hugging Face and GROQ caused unexpected failures within Streamlit's multi-threaded environment.

### UI Rendering Issues

Applying complex glassmorphic HTML/CSS styling was challenging due to compatibility and DOM restrictions imposed by the Streamlit framework.

### Resource Optimization

Switching rapidly between the large extractive and abstractive models caused significant spikes in memory consumption.

### Latency Management

The size and complexity of abstractive Transformer models introduced noticeable inference delay, impacting real-time performance goals.

### Error Handling Complexity

Unmanaged exceptions during the pipeline's initialization phase often led to hard, user-visible application crashes instead of graceful failure.

# Turning Technical Barriers into Breakthroughs

## Pipeline Caching

Implemented @st.cache\_resource to aggressively preserve model states, dramatically reducing reload times and API initialization latency.

## Frontend Optimization

Developed modular 'neo-card' components using direct HTML injection to gain precise, smooth control over the complex glassmorphic layout.

## System Feedback & Trust

Added live 'Pipeline Active / Error' indicators, coupled with real-time API key verification displays, to build user trust and offer immediate feedback.

## Graceful Failure Handling

Critical initializations were wrapped in try-except blocks, ensuring the system presented users with informative fallback messages instead of fatal crashes.

## Error Transparency

Integrated clear, user-facing error notifications (st.error) coupled with status cards featuring real-time API validation badges for accountability.

# Shaping the Future of Human-AI Collaboration

## Current Impact

70%

### Time Reduction

Reduced text processing time for users by 70%, boosting productivity.

1.5X

### Readability Score

Significantly enhanced readability and linguistic fluidity compared to generic tools.

99%

### Pipeline Stability

Achieved near-perfect pipeline stability after implementing advanced caching and error handling.

NeoGlass has established a developer-friendly, reusable summarization interface that significantly improves how information is consumed.

## Future Scope

### Multilingual Support

Expand summarization capabilities to include languages such as Tamil, Hindi, and French.

### Speech-to-Text Integration

Integrate functionality to summarize content directly from recorded lectures, meetings, and calls.

### Offline Caching

Allow for selective offline model caching to improve speed, reliability, and data privacy for sensitive content.

