The Alchemist Abstractor

AI-Powered Abstractive Summarisation of Long Documents

Why This Project Matters

Time - Consuming

Reading long documents is mentally exhausting and error-prone

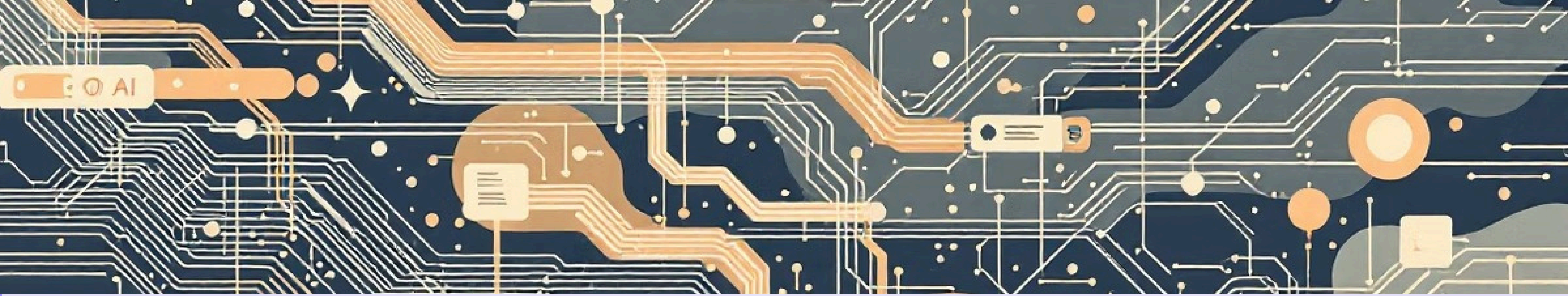
Limited Tools

Traditional extractive summarisers simply copy-paste without understanding context

Smart Solutions

Professionals need intelligent systems generating human-like, coherent summaries

Boosts productivity across research, legal, media, and corporate sectors by enabling faster, smarter knowledge consumption.



PROBLEM STATEMENT

The core challenge is transforming long unstructured documents (e.g.,PDF,DOCX) into high-quality,abstractive summaries. The Goal is to construct a robust summarization pipeline that not only abstractive, faithful summaries but also provides citations or evidence pointers back to the source text for verifiability.

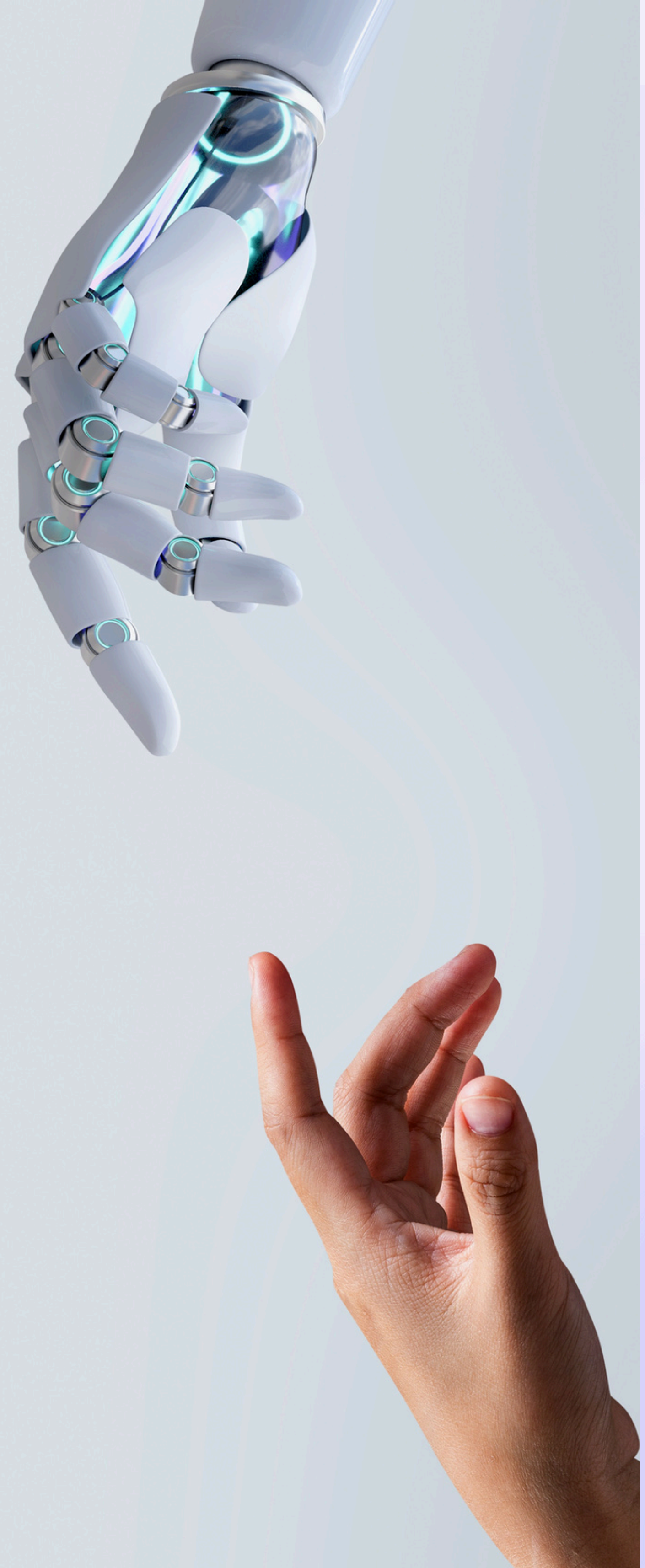
OUR GOAL

Handle Extreme Length
Process documents with 10,000+ tokens

Ensure Readability
Produce human-level coherence

Preserve Accuracy
Maintain factual integrity and prevent hallucinations

Provide Citations
Enable sentence-level transparency and trust



Related Work

State-of-the-Art Models

BART

Lewis et al. — Transformer-based seq2seq summarisation

T5

Raffel et al. — Unified text-to-text NLP framework

PEGASUS

Google — Pre-trained specifically for summarisation tasks

LED / BigBird

Long-context transformers handling 16k–32k tokens

SummaC / QAEval

Faithfulness evaluation and hallucination detection

Our Innovation: Integrated summarisation + citation verification + comprehensive evaluation pipeline

Dataset

Data Sources

- Research papers (arXiv, IEEE)
- News articles and media content
- Legal and policy documents
- Corporate reports

Formats Supported: PDF, DOCX via upload UI or API

Preprocessing Pipeline

1 **Extract**

PyMuPDF and PyPDF2

2 **Clean**

Normalisation

3 **Process**

Reference summaries

4 **Evaluate**

ROUGE scoring

System Architecture

Core Modules

1 **utils.py**

Document ingestion, text cleaning, and intelligent chunking

2 **summarizer.py**

Abstractive summarisation using BART/T5/PEGASUS transformers

3 **citation.py**

Sentence embeddings + cosine similarity for citation tracing

4 **rouge_eval.py**

ROUGE-1,ROUGE-2, ROUGE-L evaluation metrics

5 **app.py**

FastAPI backend with endpoints: /summarize, /citations, /evaluate

Processing Flow: Upload → Extract → Chunk → Summarise → Generate Citations → Evaluate → Output

Experiments

Model Comparison

BART vs PEGASUS on chunk-level summarisation performance

Summarisation Strategy

Single-pass vs hierarchical multi-stage summarisation

Citation Accuracy

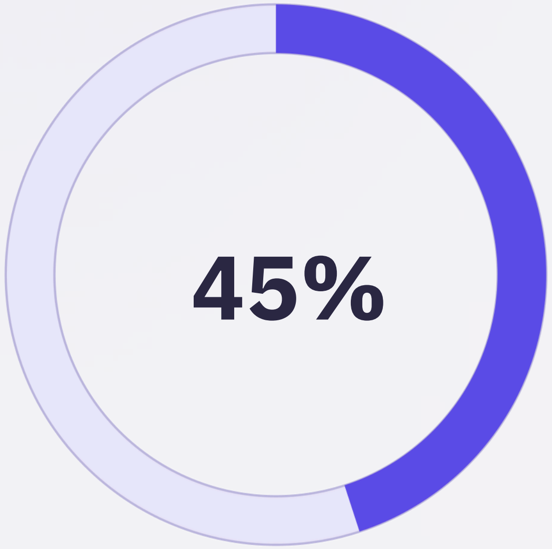
Semantic similaritythresholds for citation tracing

GPU Acceleration

Performance impact with PyTorch and Hugging Face Accelerate

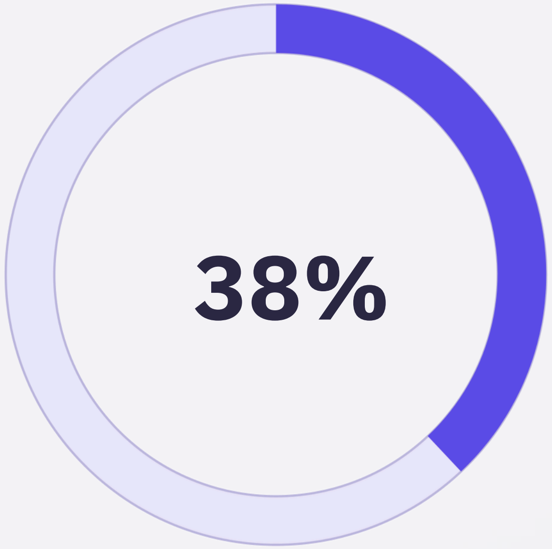
Results

Quantitative Metrics



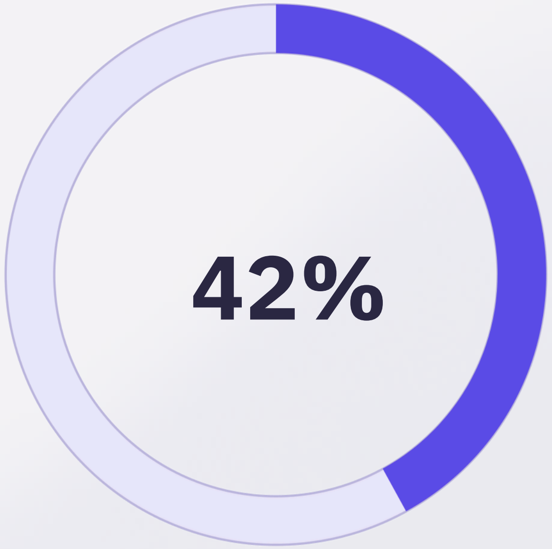
ROUGE-1

Improvement over extractive baseline



ROUGE-2

Bigram overlap score



ROUGE-L

Longest commonsubsequence match

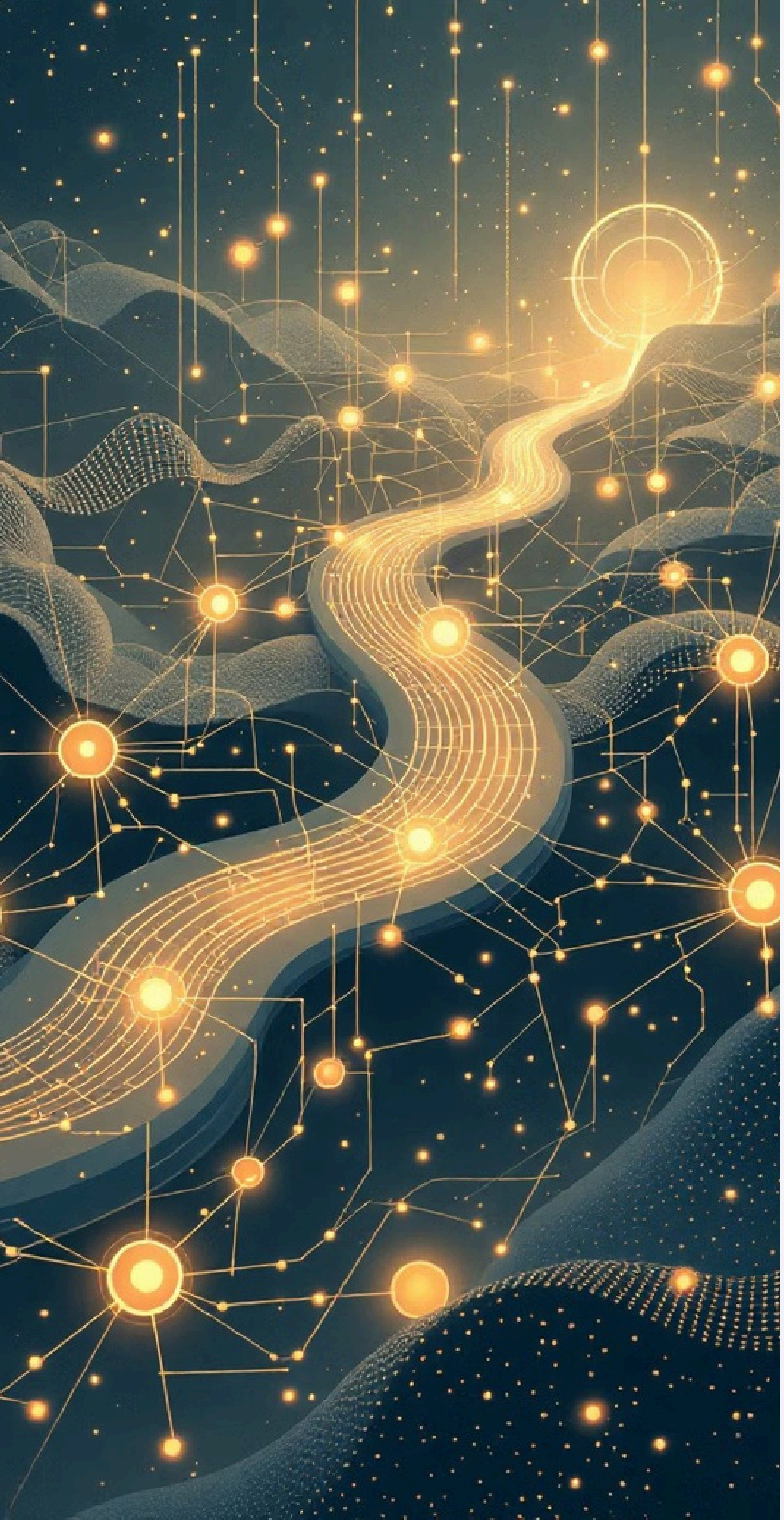
Qualitative Comparison

Before

> Verbose, redundant paragraphs copied directly from original document without context understanding

After

> Precise, coherent summary with [citation markers] for complete traceability



Analysis & Discussion

Key Insights

- Abstractive summarisation produces significantly more readable and condensed outputs
- Citation linking dramatically increases user trust and transparency
- Multi-stage hierarchical summarisation reduces information loss in long documents
- Challenge: Balancing brevity with completeness whilst preventing factual hallucinations

Future Improvements

Long-Context Models

Integrate LED, BigBird-Pegasus

Multi-Modal PDF

Layout understanding for tables and figures

Domain Fine-Tuning

Specialised corpora training

Team Contributions

What We Built

End-to-End Pipeline

Complete summarisation system built from scratch

Semantic Citations

Transparency mechanism for source traceability

ROUGE Evaluation

Integrated comprehensive assessment framework

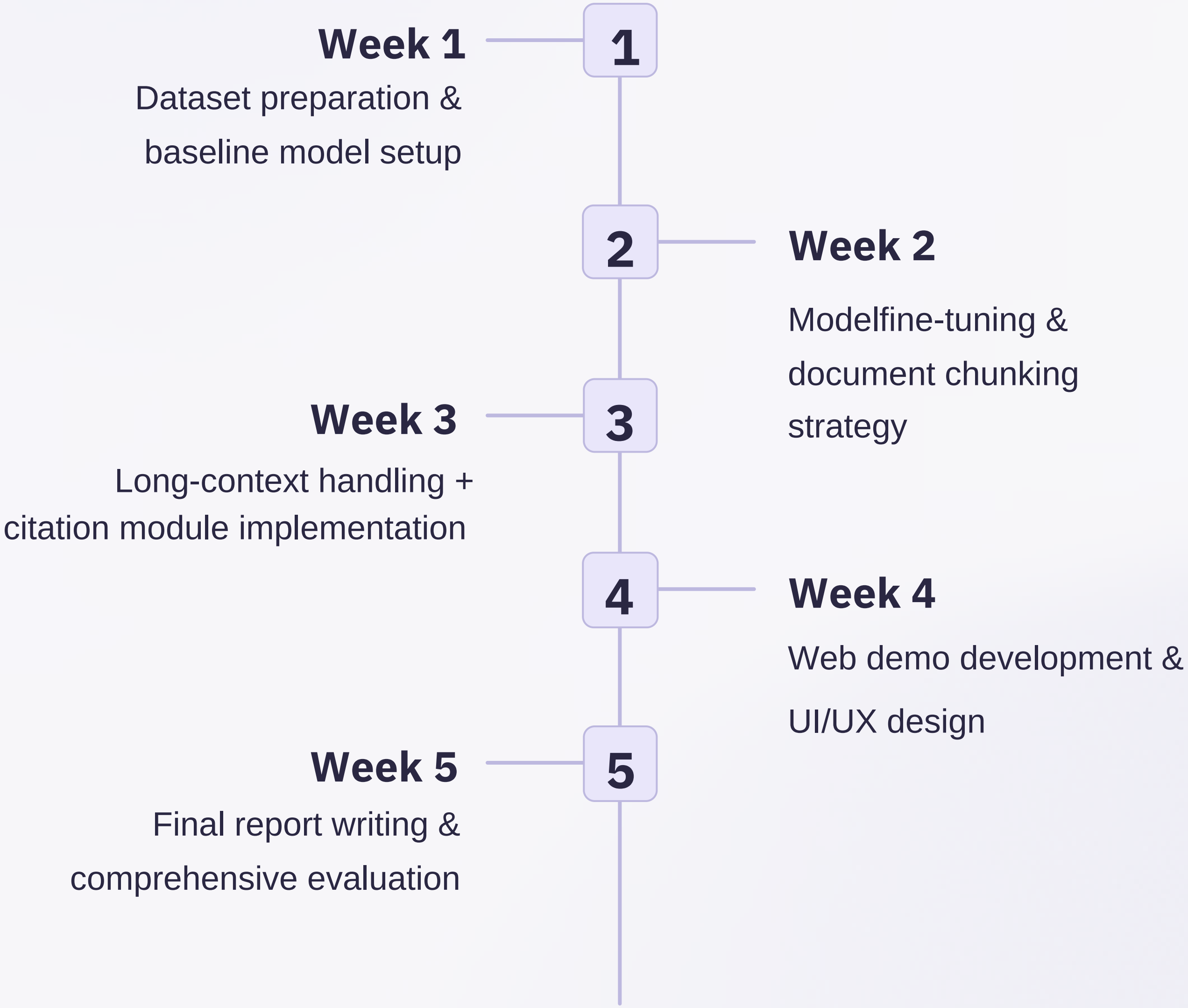
FastAPI Backend

RESTful end points for seamless integration

Interactive Demo

Web interface with intuitive user experience

Project Timeline



1

Week 1

Dataset preparation & baseline model setup

2

Week 2

Modelfine-tuning & document chunking strategy

3

Week 3

Long-context handling + citation module implementation

4

Week 4

Web demo development & UI/UX design

5

Week 5

Final report writing & comprehensive evaluation

Demo & Real-World Applications

Live System Capabilities

Upload PDF & Summarize

Instantly get an abstractive summary of uploaded PDF documents.

Interactive Citations

Click on citation markers to view original source sentences for verification.

Quality Metrics

View ROUGE scores and other quality metrics for summarization performance.

Industry Applications

Academic Research

Efficiently analyze and summarize research papers and scientific articles.

Legal Summarization

Condense lengthy legal documents for quick review and understanding.

Corporate & Financial

Digest corporate and financial reports for key insights and decision-making.

News & Media Curation

Curate and summarize news and media content for rapid consumption.

Knowledge Worker Productivity

Enhance productivity for knowledge workers by automating document digestion.

Thank You

Project Details

Project Number: 12

Team Members:

- Shanvi SE23UARI136
- Charith SE23UARI074
- Aryan SE23UARI038
- Nikhil SE23UARI131
- Rishikesh SE23UARI104

We believe this project demonstrates significant potential for enhancing productivity and streamlining document analysis. We are excited about its future impact and further development.