Leveraging Data Analytics and Behavioral Analysis for Improved Testimony and Incident Correlation

Dynamic Question Generation and Response Analysis

R25-020

Project Proposal Report

Ediriweera E. R. L. C.

BSc (Hons) In Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

We declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	Student ID	Signature
Ediriweera E. R. L. C.	IT21822230	

Signed by Dr. Lakmini Abeywardhana ..... 03/02/2025 .....

Supervisor Date

(Dr. Lakmini Abeywardhana)

Abstract

The Dynamic Question Generation and Response Analysis component aims to revolutionize criminal interrogations by leveraging advanced Natural Language Processing (NLP) techniques. Traditional interrogation methods often rely on manual questioning, which can be time-consuming, inconsistent, and prone to human error, particularly in identifying areas requiring further probing or detecting inconsistencies in testimonies. This component addresses these challenges by developing an AI-driven system capable of analyzing suspect responses in real-time. The system evaluates the coherence, relevance, and consistency of answers, dynamically generating contextually relevant follow-up questions to deepen the interrogation process.

The innovation lies in the system's ability to adapt to the evolving conversation, ensuring that questioning is tailored to the suspect's responses. By integrating emotional cues and identifying contradictions in testimonies, the system provides a more nuanced and adaptive approach to questioning. Additionally, a feedback mechanism allows investigators to rate the effectiveness of the generated questions, enabling continuous refinement of the model for future interrogations. This real-time, AI-assisted questioning not only reduces the reliance on manual interrogation techniques but also enhances the ability to uncover critical information that might otherwise be overlooked.

The proposed system represents a significant advancement in criminal investigations, offering a more efficient and accurate method for conducting interrogations. By automating the generation of follow-up questions and providing real-time insights into suspect responses, the system streamlines the investigative process, reduces human error, and improves the overall quality of interviews. This component is a key part of a broader AI-driven solution designed to enhance the accuracy, efficiency, and effectiveness of criminal investigations, ultimately contributing to faster case resolution and more reliable outcomes. The integration of NLP with dynamic questioning capabilities marks a novel step forward in the application of AI to criminal justice, offering a powerful tool for investigators to uncover the truth.

Contents

List of Tables 5
List of Figures 5
1. Introduction 6
1.1. Background 6
1.2. Research Gap 6
1.3. Research Problem 8
2. Objective 10
2.1. Main Objectives 10
2.2. Specific Objectives 10
3. Methodology 11
3.1. System Overview 11
3.1.1. Real Time Response Analysis 12
3.1.2. Model Development 12
3.1.3. Interface Development 14
3.2. Programming Languages and Frameworks 14
4. Project Requirements 16
4.1. Functional & Non-Functional Requirements 16
4.2. User Requirements 17
4.3. Software Requirements 17
4.4. Personal Requirements 18
4.5. Work Breakdown Structure 19
4.6. Gannt Chart 20
5. Description of Personal and Facilities 21
6. Budget & Budget Justification 21
7. Commercialization 22
7.1. Market Opportunity 22
7.2. Value Proposition 22
7.3. Revenue Model 23
8. References 24

List of Tables

Table 1 Research Gap 8
Table 2 Functional and Non-functional Requirements 16
Table 3 Budget 21

List of Figures

Figure 1 System Overview 11
Figure 2 Work Breakdown Structure 19
Figure 4 Gannt Chart 20

1. Introduction

1.1. Background

Dynamic question generation has emerged as a critical innovation in enhancing the quality and depth of criminal interrogations. Traditional practices rely on the investigator's ability to generate follow-up questions in real-time based on intuition, which can lead to inconsistencies and missed opportunities for probing critical details [1]. Advances in Natural Language Processing (NLP) have opened new avenues for automating this process, enabling systems to generate contextually relevant questions by analyzing suspect responses in real-time [2].

Studies in this domain demonstrate the transformative potential of NLP techniques in question generation. For instance, Conneau et al. showcased the efficacy of deep neural networks in understanding and generating text, forming the foundation for modern NLP tools [3]. Similarly, Lai et al. explored the application of Recurrent Neural Networks (RNNs) for analyzing text coherence, which helps identify inconsistencies in responses [4]. These foundational studies underscore the capability of NLP to dynamically adapt to context, a crucial requirement in criminal investigations.

To successfully implement dynamic question generation in interrogations, expertise in transformer-based models, such as BERT and GPT, is necessary. Additionally, algorithms for semantic analysis and real-time response evaluation must be integrated to ensure contextual accuracy. Current state-of-the-art systems are largely restricted to structured environments, such as customer service or education, where the context is relatively static [2]. However, their inability to adapt to evolving, high-stakes scenarios like criminal investigations limit their applicability, leaving a critical gap to be addressed.

1.2. Research Gap

The application of artificial intelligence (AI) in criminal investigations has seen significant advancements, particularly in automating processes such as deception detection, evidence analysis, and forensic science integration. However, the interrogation process remains largely reliant on traditional human-driven questioning methods, which are prone to cognitive biases, inconsistencies, and inefficiencies in eliciting truthful and complete testimonies. The existing literature highlights several gaps in leveraging dynamic question generation and response analysis for enhancing interrogation techniques.

One of the key challenges in current interrogation practices is the lack of adaptive and real-time questioning mechanisms. Studies emphasize that AI-driven solutions in

criminal justice predominantly focus on automating evidence collection, emotion detection, and deception analysis, rather than dynamically generating questions based on real-time interrogation feedback [1], [2]. Current methodologies for lie detection rely on physiological markers and machine learning models that analyze behavioral indicators such as facial expressions, voice stress, and eye movements [3], [4]. However, these techniques are not integrated with dynamic question-generation models that can adaptively refine interrogation strategies.

Furthermore, AI applications in legal and forensic settings remain fragmented. Existing AI-based interrogation support tools are primarily designed to detect inconsistencies and provide credibility assessments rather than engaging in an interactive dialogue with the suspect [5]. Emotion recognition and speech analysis have been successfully applied to detect stress and deception [6], [7], but there is limited research on utilizing these insights to generate contextually relevant follow-up questions dynamically. This results in a missed opportunity to probe inconsistencies effectively and refine the questioning approach in real time.

Another limitation is the absence of a robust framework that integrates Natural Language Processing (NLP) with real-time response analysis. Studies on short text classification and deep learning in text processing have demonstrated significant advancements in semantic analysis and contextual understanding [8]–[10]. However, these techniques have not been extensively applied in criminal interrogations to analyze suspect responses and generate meaningful, context-driven follow-up questions. The current interrogation models rely on predefined question sets or manually formulated queries, which do not adapt to evolving narratives within an interrogation session.

Moreover, research on AI-driven criminal investigation highlights the challenges of ensuring ethical and legal compliance when using automated systems for interrogations [11]–[13]. Concerns regarding bias in AI models, the reliability of generated questions, and the potential for coercion have been raised in several studies [14]. While existing forensic AI applications demonstrate promise in improving investigative efficiency, a clear gap remains in the development of an AI-powered interrogation assistant that not only assesses suspect credibility but also formulates adaptive questions to maximize information extraction.

Given these limitations, the proposed research aims to address this gap by developing a dynamic NLP-driven system that generates real-time, contextually relevant follow-up questions during interrogations. This approach will bridge the existing gap by integrating automated response analysis with adaptive questioning, ultimately enhancing the quality and effectiveness of criminal investigations.

Feature	Research A [1]	Research B [2]	Research C [3]	Proposed System
Dynamic Question Generation	✗	✗	✗	✓
Response Analysis	✗	✓	✗	✓
Integration with Behavioral Cues	✗	✗	✗	✓
Real-Time Interrogation Feedback	✗	✗	✗	✓
Investigator Feedback Loop	✗	✗	✗	✓
Multi-Modal Integration	✗	✗	✗	✓

Table 1 Research Gap

1.3. Research Problem

1. **Reliance on Static and Predefined Questioning Techniques**: Traditional interrogation methods rely on investigator-driven questioning that follows a linear and often rigid approach. Investigators manually formulate follow-up questions based on their intuition, experience, and available evidence. This static approach limits the ability to explore inconsistencies in testimonies effectively and adapt questioning strategies in real time.
2. **Subjective Human Judgment and Cognitive Bias**: Human-driven interrogations are susceptible to cognitive biases and subjective interpretations. Investigators may unintentionally frame questions based on preconceived notions, leading to suggestive questioning and potential misinterpretation of suspect responses. This can impact the accuracy and fairness of interrogations, reducing their overall effectiveness.
3. **Fatigue-Related Errors and Missed Insights**: Criminal interrogations can be prolonged, increasing the likelihood of investigator fatigue. As fatigue sets in, the ability to detect subtle inconsistencies, formulate relevant follow-up questions, and maintain a structured questioning approach diminishes. This can result in missed opportunities to uncover crucial insights that could strengthen an investigation.
4. **Ineffective Documentation and Misinterpretation**: Traditional interrogation methods heavily rely on manual note-taking and audio/video recordings, which require post-interview analysis. Important details may be overlooked or misinterpreted due to inconsistencies in documentation. The absence of automated analysis further exacerbates the challenge of tracking contradictions in testimonies over multiple interrogation sessions.

5. **Limited Real-Time Response Analysis and Dynamic Questioning**: Existing AI-driven tools in criminal investigations primarily focus on evidence management, deception detection, and behavioral analysis. However, they lack real-time interrogation support that dynamically adapts questioning strategies. Without automated response analysis, investigators rely solely on intuition, increasing the risk of missing crucial contradictions in a suspect's statement.

Research Question

- How can AI-driven NLP techniques be utilized to dynamically analyze suspect responses and generate adaptive follow-up questions in real-time?
- What are the key challenges in integrating behavioral and emotional analysis with dynamic question generation to enhance interrogation effectiveness?
- How can an AI-powered interrogation system minimize investigator bias, fatigue-related errors, and misinterpretation while improving the accuracy and depth of extracted testimonies?

2. Objective

2.1. Main Objectives

The main objectives of this component are centered around developing an AI-driven module that utilizes Natural Language Processing (NLP) techniques to enhance criminal interrogations. The system is designed to generate dynamic, context-aware follow-up questions in real-time and analyze responses to ensure coherence and relevance. By integrating these features, the module aims to improve the quality and efficiency of interrogations, enabling investigators to uncover deeper insights and address inconsistencies effectively.

- Design algorithms capable of analyzing suspect responses in real-time for coherence, relevance, and emotional cues.
- Develop NLP-based models trained on real interrogation datasets to generate relevant follow-up questions dynamically.
- Provide investigators with actionable insights based on response patterns and emotional cues detected during interrogations.
- Implement an adaptive questioning strategy tailored to detected behavioral and emotional changes.

2.2. Specific Objectives

The specific objectives focus on the essential tasks required to achieve the main objectives. These include collecting and preprocessing interrogation data, training NLP models for question generation, and implementing user interfaces for seamless investigator interaction. The sub-objectives ensure the creation of a robust, efficient, and investigator-friendly module for dynamic question generation and response analysis.

- Collect and preprocess real or simulated interrogation datasets, annotating them for coherence, relevance, and emotional cues.
- Develop NLP models to generate contextually relevant follow-up questions based on suspect responses and predefined parameters.
- Develop a real-time analysis engine to identify inconsistencies, patterns, and emotional cues in suspect responses.
- Design a user-friendly interface for investigators to interact with the system and provide feedback on question effectiveness.
- Build a feedback mechanism for model refinement, using investigator input to improve the relevance and quality of questions generated.
- Ensure backend development for data storage, model integration, and secure handling of interrogation data.

3. Methodology

3.1. System Overview

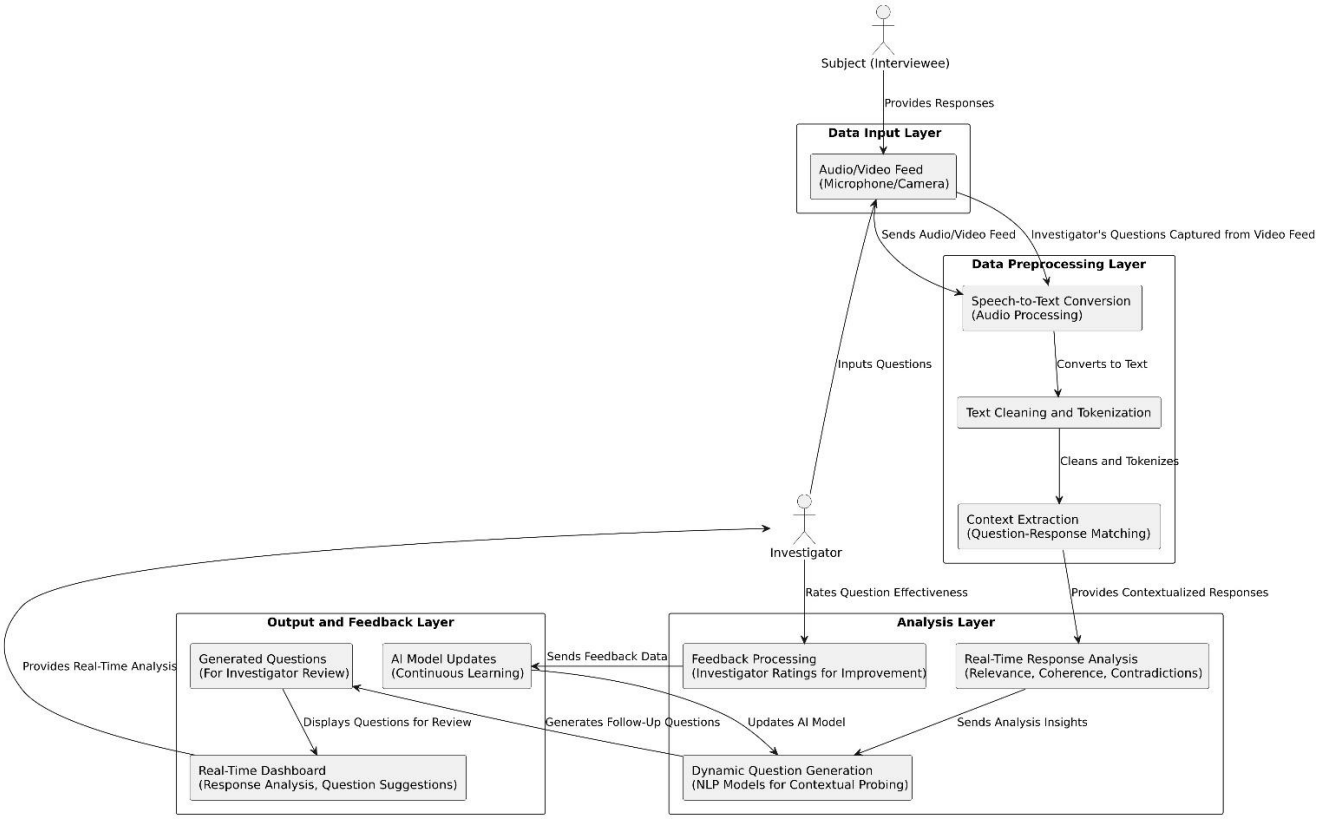


Figure 1 System Overview

The **Dynamic Question Generation and Response Analysis** system is a critical component aimed at enhancing the depth and relevance of criminal interrogations by utilizing Natural Language Processing (NLP) techniques. This system operates through four integrated layers, Data Input Layer, Data Preprocessing Layer, Analysis Layer, and Output & Feedback Layer. These components work together to automate question generation, analyze suspect responses, and minimize investigator biases, ultimately improving interrogation efficiency and testimony accuracy.

The Data Input Layer captures interrogation data through microphones and cameras, collecting both verbal responses and non-verbal cues such as facial expressions and stress indicators. This data is then processed in the Data Preprocessing Layer, where speech-to-text conversion, text cleaning, and question-response matching take place. This layer ensures that raw interrogation data is structured and refined, making it suitable for advanced analysis.

The Analysis Layer is the system's core intelligence, evaluating responses for coherence, contradictions, and relevance while dynamically generating follow-up questions. NLP-based models analyze suspect statements in real time, adjusting questioning strategies based on

inconsistencies and emotional cues. Additionally, behavioral analysis is integrated to detect stress, hesitation, or deception, allowing for adaptive interrogation approaches.

The Output and Feedback Layer serves as the investigator's interface, providing a real-time dashboard with response evaluations, contradiction alerts, and question suggestions. Investigators can review and refine AI-generated questions, and their feedback helps improve the model's performance.

3.1.1. Real Time Response Analysis

The first stage of development involves creating a system that processes and analyzes suspect responses in real time, identifying inconsistencies, behavioral patterns, and emotional cues. Processed data is continuously streamed to the analysis layer, enabling real-time detection of inconsistencies and behavioral cues, improving the depth and accuracy of investigations.

- Developing an audio and video processing module to capture verbal and non-verbal cues from suspects during interrogation.
- Implementing a speech-to-text conversion engine for transcription and real-time analysis of spoken responses.
- Creating a response analysis model that evaluates coherence, contradictions, and linguistic cues to identify potential deception.
- Integrating behavioral analysis algorithms that detect stress, hesitation, and emotional shifts using facial expression recognition and speech sentiment analysis.
- Developing a contradiction and evasion detection module, flagging suspect statements that require further probing.

3.1.2. Model Development

The development of the **Dynamic Question Generation and Response Analysis** model is centered on advanced NLP architecture and reinforcement learning techniques to generate and evaluate follow-up questions effectively. The following stages define the model development process:

1. **Model Architecture**
    i. **Transformer-based Models**: Pre-trained models such as BERT and GPT are fine-tuned on the collected datasets to generate context-aware questions.
    ii. **Dynamic Context Encoding**: The architecture incorporates contextual embeddings to ensure questions are relevant to previous responses.

2. **Transfer Learning**
    i. The models are initialized with pre-trained weights from general-purpose NLP tasks, significantly accelerating the training process and enhancing accuracy.
3. **Training Process**
    i. The combined dataset, including the SQUAD dataset and audio-transcribed data, is used for supervised learning.
    ii. Loss functions such as cross-entropy are optimized over multiple epochs.
    iii. Data augmentation techniques are employed during training to increase model robustness.
4. **Reinforcement Learning for Continuous Improvement**
    i. Investigator feedback on generated questions is used to improve the model dynamically through reinforcement learning.
    ii. Positive feedback enhances the probability of generating similar questions, while negative feedback suppresses less effective patterns.
5. **Hyperparameter Tuning**
    i. Parameters such as learning rate, batch size, and epochs are tuned using grid search and cross-validation techniques.
    ii. A **OneCycleLR** scheduler is used to dynamically adjust the learning rate, optimizing training efficiency.
6. **Evaluation Metrics**
    i. Metrics such as BLEU, ROUGE, and perplexity are used to evaluate the quality of generated questions.
    ii. Human evaluators assess the relevance, coherence, and adaptability of the questions in real-world scenarios.

7. **Model Optimization for Deployment**
    i. Techniques such as pruning and quantization are applied to reduce the model's size for efficient deployment.
    ii. The model is exported to production-ready formats like TensorFlow Lite for compatibility with edge devices.

3.1.3. Interface Development

The final stage involves building an intuitive investigator interface that provides real- time response insights, suggested follow-up questions, and interactive feedback mechanisms. The implementation includes:

- Developing a real-time dashboard where investigators can review suspect response analytics, detected inconsistencies, and AI-generated follow-up questions.
- Creating an interactive question refinement feature, allowing investigators to approve, modify, or reject AI-generated questions before they are asked.
- Implementing an investigator feedback loop, where users rate question effectiveness, helping to refine and improve AI performance over time.
- Ensuring secure storage and encryption of interrogation data, complying with legal and ethical standards for AI-driven law enforcement tools.
- Optimizing system accessibility and usability, ensuring that investigators with minimal technical knowledge can effectively interact with the system.

3.2. Programming Languages and Frameworks

- **Python with TensorFlow and PyTorch**:
  - **Primary Language**: Python is the primary programming language for implementing the dynamic question generation and response analysis system. Python's extensive libraries for Natural Language Processing (NLP) and machine learning make it ideal for this application.

  - **Frameworks**: TensorFlow and PyTorch are used for designing, training, and deploying NLP models. These frameworks provide robust tools for building advanced language models, managing training routines, and utilizing pre-trained models like BERT and GPT for question generation and response analysis.

- **Natural Language Toolkit (NLTK) and SpaCy**:
  - **Text Processing**: NLTK and SpaCy are utilized for preprocessing and tokenizing textual data. They handle tasks such as lemmatization, stemming, and part-of-speech tagging, which are essential for analyzing suspect responses during interrogations.

- **Flask and FastAPI**:
    - **Backend Services**: Flask and FastAPI are employed to develop the API for integrating the NLP system with other components. These frameworks enable efficient routing, model hosting, and real-time data exchange between the frontend and backend.

        - **ReactJS**:
            - **Frontend Technology**: ReactJS is used to design the user interface for the application. This framework ensures a dynamic, responsive, and intuitive experience for investigators accessing question generation and response analysis functionalities.

        - **PostgreSQL**:
            - **Data Management**: PostgreSQL is used to store structured data, including interrogations, generated questions, and analysis reports. Its scalability and support for advanced queries ensure efficient handling of large datasets.

- **Google Cloud Platform (GCP)**:
    - **Cloud Infrastructure**: GCP is utilized to host the NLP models, enabling scalable and reliable real-time operations. GCP services such as Cloud Storage and AI Platform streamline deployment and data processing.
- **Docker**:
    - **Containerization**: Docker is used for deploying the system in a microservices architecture, ensuring modularity, consistency, and seamless performance across different environments.

- **Git for Version Control**:
    - **Collaboration and CI/CD**: Git ensures proper version control and collaboration across the development team. GitHub integrates CI/CD pipelines for continuous testing and deployment.

- **HTTPS**:
    - **Secure Communication**: HTTPS is implemented to secure data exchange between client-side applications and the backend server, ensuring data integrity and privacy.

- **Client-Server Architecture**:
    - **Operational Structure**: The system adopts a client-server architecture where the frontend interfaces with the backend services via RESTful APIs, ensuring scalability and maintainability.

- **Microservices Architecture**:
    - **Modular Design**: Independent services for question generation, response analysis, and data storage allow for easy scalability and flexibility in development and deployment.

4. Project Requirements

4.1. Functional & Non-Functional Requirements

Functional Requirements	Non-Functional Requirements
The system shall analyze suspect responses in real-time during interrogations to evaluate coherence, relevance, and contradictions.	The system shall process and analyze responses with a maximum latency of 2 seconds to ensure real-time operation.
The system shall dynamically generate contextually relevant follow-up questions based on analyzed responses.	The system shall scale efficiently to support multiple simultaneous interrogations without significant performance degradation.
The system shall allow investigators to review and approve or modify generated follow-up questions before they are used.	The system shall provide a responsive and intuitive interface for investigators to interact with real-time data and suggestions.
The system shall incorporate investigator feedback to improve the accuracy and relevance of question generation over time.	The system shall achieve an availability rate of 99.9%, ensuring minimal downtime for critical interrogation operations.
The system shall maintain a history of generated questions, responses, and feedback for future reference and analysis.	The system shall comply with data privacy and security standards, ensuring the confidentiality of interrogation data.
The system shall integrate seamlessly with other components, such as emotion analysis and incident correlation modules.	The system shall adhere to design principles for accessibility, ensuring that investigators with disabilities can use it effectively.
The system shall support multilingual capabilities to analyze and generate questions in different languages.	The system shall provide cross-platform compatibility, functioning across desktop, mobile, and tablet devices.
The system shall generate reports summarizing interrogation progress and system-generated question effectiveness.	The system shall use lightweight architecture to minimize resource consumption while maintaining optimal performance.

Table 2 Functional and Non-functional Requirements

4.2. User Requirements

- Investigators shall be able to log in to the system securely using their credentials to access the question generation and response analysis features.
- Investigators shall be able to view and review real-time analysis of suspect responses during interrogations through an intuitive dashboard.
- Investigators shall have the ability to approve, reject, or modify system-generated follow-up questions before they are used in the interrogation.
- Investigators shall be able to provide feedback on the relevance and effectiveness of system-generated questions for continuous system improvement.
- Investigators shall have access to a history of responses, generated questions, and feedback to review interrogation progress and trends.
- Investigators shall be able to generate and download detailed reports summarizing the system's analysis and performance for specific interrogations.
- Investigators shall have the option to switch the system's interface language to accommodate multilingual capabilities.
- Investigators shall receive real-time notifications on the dashboard about inconsistencies or contradictions detected in the suspect's responses.

4.3. Software Requirements

- **Programming Languages**
  - Python
  - JavaScript
- **Integrated Development Environments (IDEs)**
  - VS Code
  - PyCharm
- **Libraries and Frameworks**
  - TensorFlow
  - PyTorch
  - Hugging Face Transformers

    - SpaCy
    - NLTK
    - FastAPI
    - PostgreSQL
    - Flask
- **Machine Learning Algorithms**
    - BERT (Bidirectional Encoder Representations from Transformers)
    - GPT (Generative Pre-trained Transformer)
    - Reinforcement Learning Models
- **Visualization and Analysis Tools**
    - Jupyter Notebook

- **Version Control**
    - Git and GitHub
- **Cloud and Hosting Services**
    - Google Cloud Platform (GCP)
- **Database**
    - Relational Database Management System (PostgreSQL)

4.4. Personal Requirements

- **Expertise in Natural Language Processing (NLP)**
    - Proficiency in NLP techniques such as tokenization, semantic analysis, and named entity recognition to build and optimize question generation models.
- **Experience in Machine Learning and Deep Learning**
    - Strong understanding of training, fine-tuning, and deploying models like BERT and GPT for dynamic question generation.
- **Familiarity with Interrogation and Behavioral Analysis Techniques**
    - Knowledge of interrogation practices to understand the context and relevance of follow-up questions.
- **Proficiency in API Development**
    - Ability to develop and maintain APIs for seamless integration between components using tools like FastAPI and Flask.
- **Database Management Skills**
    - Experience in managing structured and unstructured data using PostgreSQL for storing historical responses and feedback.
- **Experience with Cloud Platforms**
    - Knowledge of deploying machine learning models on cloud services such as Google Cloud Platform (GCP).

- **Strong Communication and Collaboration Skills**
  - Ability to work effectively with law enforcement and system development teams to align technical solutions with investigative needs

4.5. Work Breakdown Structure

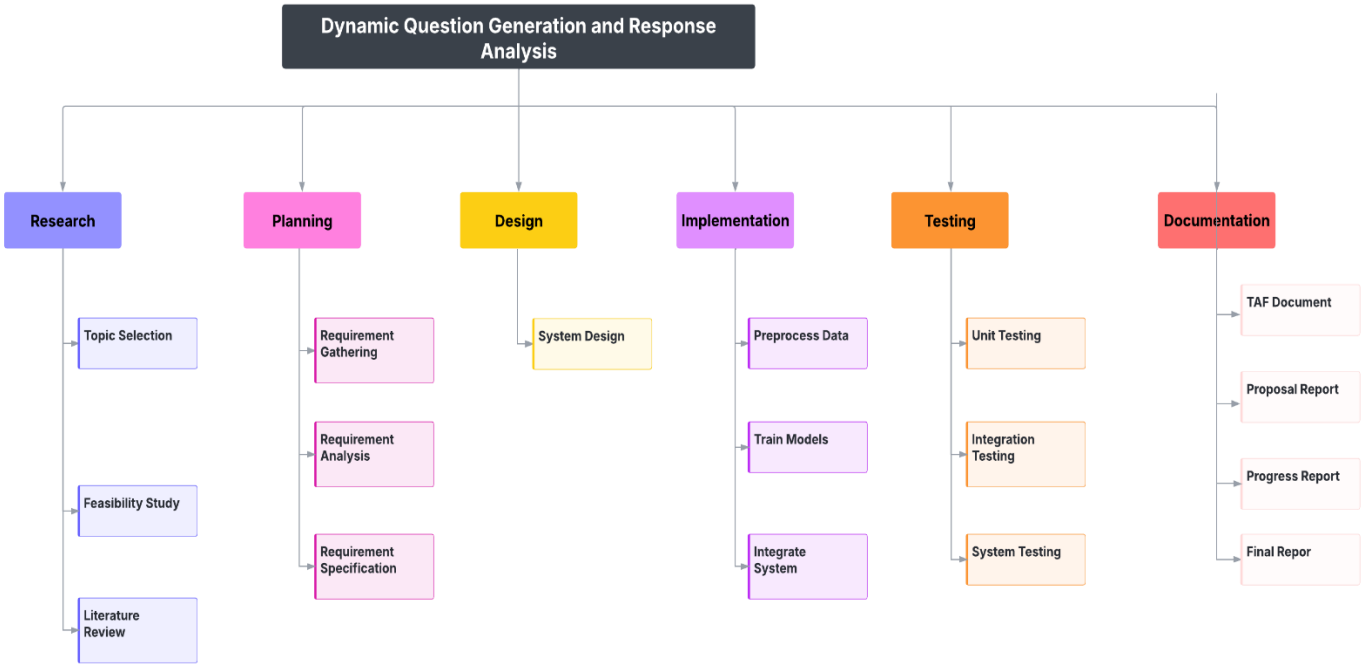


Figure 2 Work Breakdown Structure

4.6. Gannt Chart

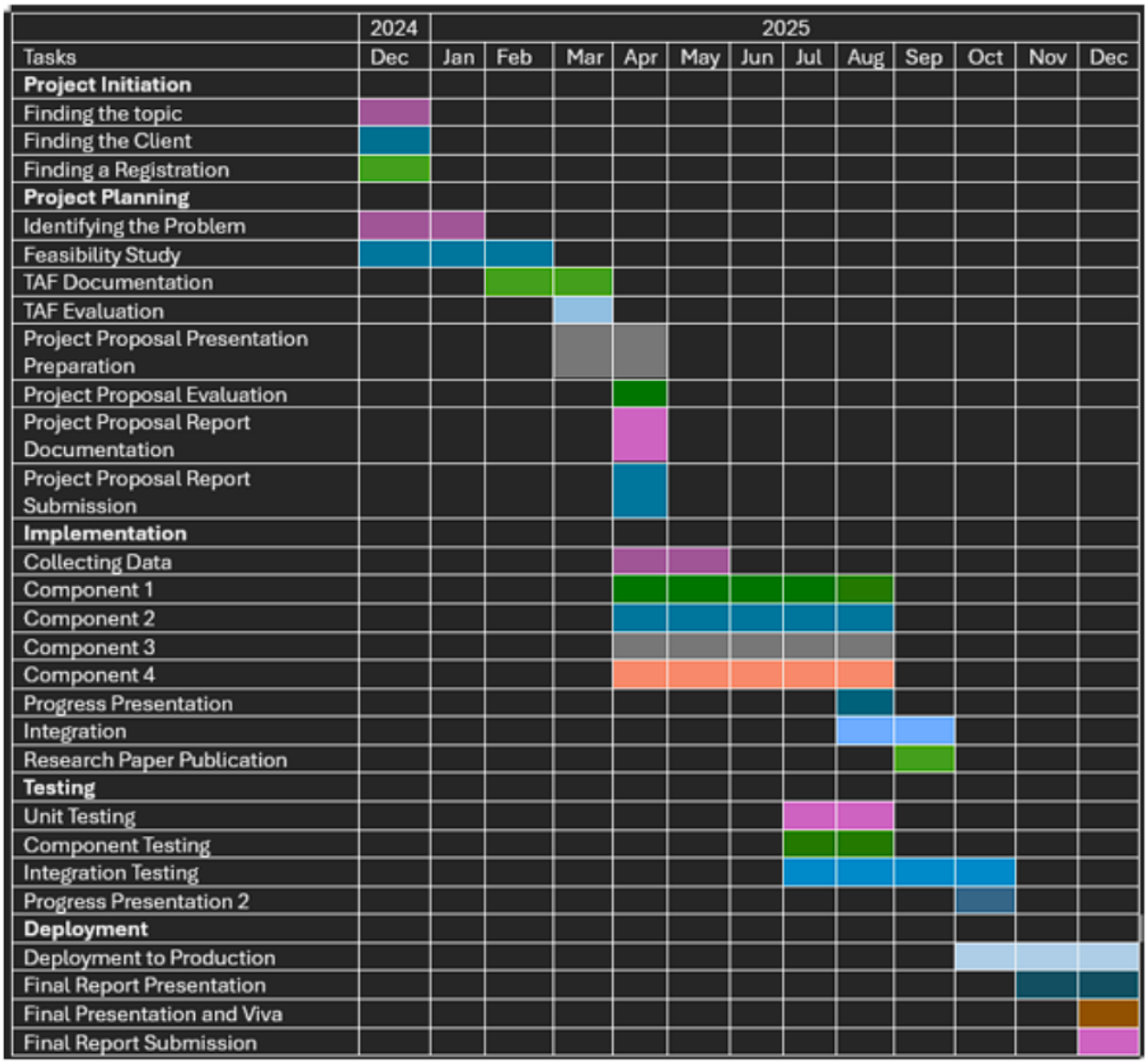


Figure 3 Gannt Chart

5. Description of Personal and Facilities

- Supervisor : Dr. Lakmini Abeywardhana
- Co-supervisor : Ms. Pipuni Wijesiri

6. Budget & Budget Justification

Category	Item Description	Unit Cost (USD)	Quantity
Cloud Services	Google Cloud Platform (Free Trial) for model training and hosting	$0	1
Software & Tools	Python, TensorFlow, PyTorch, NLTK, SpaCy, Flask, FastAPI, ReactJS, PostgreSQL, Docker (Free Versions)	$0	1
Hardware	Existing laptop used for development	already owned	1
Data Acquisition	Datasets from Kaggle (Free)	$0	1
Online Courses	Enhance skills in NLP	$50 - $100	
Google Colab (Cloud Computing)	Provides computational power for running experiments and reducing local hardware dependency.	Free / $120 per year	3
Miscellaneous & Contingency	Unforeseen expenses, research materials, and other minor costs	$50	1

Table 3 Budget

7. Commercialization

7.1. Market Opportunity

Market Gap

- Existing interrogation tools lack AI-driven dynamic questioning, real-time behavioral analysis, and multi-modal data processing (audio, video, text).
- Current systems focus on static analysis (e.g., lie detection, speech transcription) but fail to generate adaptive, context-aware follow-up questions for investigators.
- Manual interrogation methods lead to investigator bias, fatigue-related errors, and missed insights, creating inefficiencies in legal and forensic investigations.

Target Users

- Law enforcement agencies – Police departments, national security agencies, forensic teams.
- Private investigators – Individual and agency-level investigators handling fraud, corporate investigations, and legal cases.
- Judicial bodies – Courts, public defenders, prosecution teams using AI-assisted testimony evaluation.

7.2. Value Proposition

- Automates data handling and incident correlation, significantly reducing human error and time spent on manual interrogation analysis.
- Provides real-time behavioral and emotional insights, helping investigators detect stress, deception, and inconsistencies in suspect testimonies.
- Generates dynamic, AI-driven follow-up questions based on suspect responses, contradictions, and detected emotional cues.
- Reduces investigator workload by providing automated response analysis, contradiction detection, and question optimization.
- Predictive analytics for case resolution – Helps law enforcement identify patterns, cross-reference historical case data, and improve investigative efficiency.

7.3. Revenue Model

Subscription-Based Model (SaaS - Monthly/Yearly)

- Suitable for law enforcement agencies and investigative firms requiring continuous access.
- Pricing tiers based on number of users, AI processing capacity, and integrated behavioral analysis features.

Per-Case Usage Pricing

- Designed for independent investigators, legal consultants, and forensic analysts who require on-demand AI interrogation assistance.
- Charges based on number of interrogations, AI-generated questions, and behavioral analysis reports per case.

Enterprise Licensing for Large-Scale Deployment

- For national security agencies, forensic research institutions, and high-profile investigation units.
- Custom contracts for on-premise deployment, advanced AI customization, and dedicated support.

Additional Monetization Opportunities

- API Integration – Licensing AI interrogation capabilities for third-party forensic and legal tech platforms.
- Partnerships with law enforcement bodies for custom AI model training and integration.
- Expansion into multilingual markets to offer global AI-driven investigative tools.

8. References

1. A. T. A. Sadiqur Rahman and A. Kaur, "The Application of Artificial Intelligence in the Administration of Criminal Law and the Procedures Involved in Criminal Cases," *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)*, Indore, India, 2024, pp. 1-5. doi: 10.1109/ACROSET62108.2024.10743336.
2. V. Singh, A. Dixit, S. Pandey, B. V. Kumar, V. Pachouri, and M. Sahu, "Role of Artificial Intelligence in Criminal Investigation," *2023 International Conference on Quantum Technologies, Communications, Computing, Hardware and Embedded Systems Security (iQ-CCHESS)*, Kottayam, India, 2023, pp. 1-4. doi: 10.1109/iQ-CCHESS56596.2023.10391288.
3. J. Baltrūnienė, "Place of artificial intelligence in the detection and investigation of crime: the present state and future perspectives," *Current Problems of Forensic Science*, vol. 26, pp. 43-58, 2023. doi: 10.52097/pwk.5431.
4. I. Lakshan, L. Wickramasinghe, S. Disala, S. Chandrasegar, and P. S. Haddela, "Real-Time Deception Detection for Criminal Investigation," *2019 National Information Technology Conference (NITC)*, Colombo, Sri Lanka, 2019, pp. 90-96. doi: 10.1109/NITC48475.2019.9114422.
5. R. Ejjami, "AI-driven Justice: Evaluating the Impact of Artificial Intelligence on Legal Systems," *IJFMR*, vol. 6, no. 3, pp. 1-6, May-June 2024. doi: 10.36948/ijfmr.2024.v06i03.23969.
6. S. S. Das, S. Patnaik, P. Pattnayak, and A. Mohanty, "An Advancement in Forensic and Criminal Investigation Through Artificial Intelligence," *2023 OITS International Conference on Information Technology (OCIT)*, Raipur, India, 2023, pp. 920-926. doi: 10.1109/OCIT59427.2023.10431257.
7. J. K. Sabherwal and K. Kaur, "The Role of Artificial Intelligence in Improving Criminal Justice System," *International Journal of Science and Research (IJSR)*, vol. 13, no. 4, pp. 999-1002, April 2024. doi: 10.21275/SR24412153835.
8. Y. Hu, Y. Li, T. Yang, and Q. Pan, "Short Text Classification with A Convolutional Neural Networks Based Method," *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Singapore, 2018, pp. 1432-1435. doi: 10.1109/ICARCV.2018.8581332.
9. S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent Convolutional Neural Networks for Text Classification," *AAAI*, vol. 29, no. 1, Feb. 2015.
10. A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very Deep Convolutional Networks for Text Classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, Valencia, Spain, 2017, pp. 1107-1116.
11. A. Higginson, E. Eggins, and L. Mazerolle, "Police Techniques for Investigating Serious Violent Crime: A Systematic Review," *Trends and Issues in Crime and Criminal Justice*, vol. 539, 2017.

12. C. Rigano, "Using Artificial Intelligence to Address Criminal Justice Needs," *National Institute of Justice (NIJ)*, Oct. 8, 2018. [Online]. Available: https://nij.ojp.gov/topics/articles/using-artificial-intelligence-address-criminal-justice-needs. [Accessed: Dec. 4, 2024].
13. "Investigation and Evidence Gathering," *Criminal Justice Research*, [Online]. Available: https://criminal-justice.iresearchnet.com/criminal-justice-process/investigation-and-evidence-gathering. [Accessed: Dec. 4, 2024].
14. EclipseForensics, "Comparing Traditional vs. Modern Forensic Methods: A Brief Guide," *EclipseForensics*, Oct. 2, 2024. [Online]. Available: https://eclipseforensics.com/comparing-traditional-vs-modern-forensic-methods-a-brief-guide. [Accessed: Dec. 4, 2024].
15. "Sri Lanka Law, Code of Criminal Procedure Act," [Online]. Available: https://www.srilankalaw.lk/c/217-code-of-criminal-procedure-act.html. [Accessed: Dec. 5, 2024].