EXTRACTING INTELLIGENT INSIGHTS WITH AI-BASED SYSTEMS
PROJECT REPORT

1.Introduction

1.1 Project Overview

As technology(AI) is rapidly increasing it is making things easy. As the same it also reducing efforts in reading long texts which consumes long time.

Extracting intelligent insights through AI-based systems involves leveraging advanced algorithms and data processing capabilities to derive valuable information from vast amounts of data. These systems employ various techniques such as machine learning, natural language processing, and data mining to uncover patterns, trends, and correlations that might not be immediately apparent to humans.

By utilizing AI, organizations can analyze data at scale and speed, enabling them to make data-driven decisions and gain a deeper understanding of complex information. These systems can sift through structured and unstructured data sources, including text, images, videos, and sensor data, to generate actionable insights, predict future trends, identify anomalies, and even automate certain decision-making processes.

The insights derived from AI-based systems can revolutionize businesses across industries, driving innovation, improving operational efficiency, enhancing customer experiences, and enabling more precise strategies. However, it's crucial to ensure ethical considerations, data privacy, and transparency when implementing AI systems to extract intelligent insights, fostering trust and responsible use of these powerful technologies.

1.2 Purpose

AI-based systems are designed to extract invaluable insights from diverse datasets, enabling businesses to make informed decisions, automate processes, predict future trends, personalize experiences for customers, drive innovation, and effectively manage risks. By leveraging advanced algorithms and data analysis techniques, these systems unlock the potential within data, empowering organizations to optimize strategies and operations across various facets of their business.

2.Literature Survey

2.1 Existing Problem:The extraction of intelligent insights from vast and diverse datasets has been a challenge due to several factors:

- Data Complexity: Datasets today encompass structured, unstructured, and semi-structured data, making it challenging to derive meaningful insights manually.

- Scalability Issues: Traditional methods struggle to scale with the exponential growth of data, leading to inefficiencies in analysis.
- Information Overload: The sheer volume of data available often leads to information overload, making it difficult to extract actionable insights effectively.
- Need for Real-time Insights: With businesses requiring timely decisions, there's a growing necessity for systems capable of providing real-time insights.

2.2 References (Sample References):

1. Smith, J., & Johnson, R. (2020). "Intelligent Insights: Leveraging AI for Effective Decision-Making." Journal of Advanced Analytics, 8(2), 120-135.
2. Brown, A., et al. (2019). "Scalable AI Frameworks for Insight Extraction in Big Data Environments." Proceedings of the IEEE International Conference on Artificial Intelligence, 45-52.
3. Gupta, S., & Lee, M. (2021). "Real-time Data Analysis Using AI: A Comprehensive Survey." ACM Transactions on Intelligent Systems and Technology, 13(4), 78-94.
4. Chen, Q., et al. (2018). "NLP Techniques for Intelligent Insights Extraction from Textual Data." Information Retrieval Journal, 25(3), 210-225.

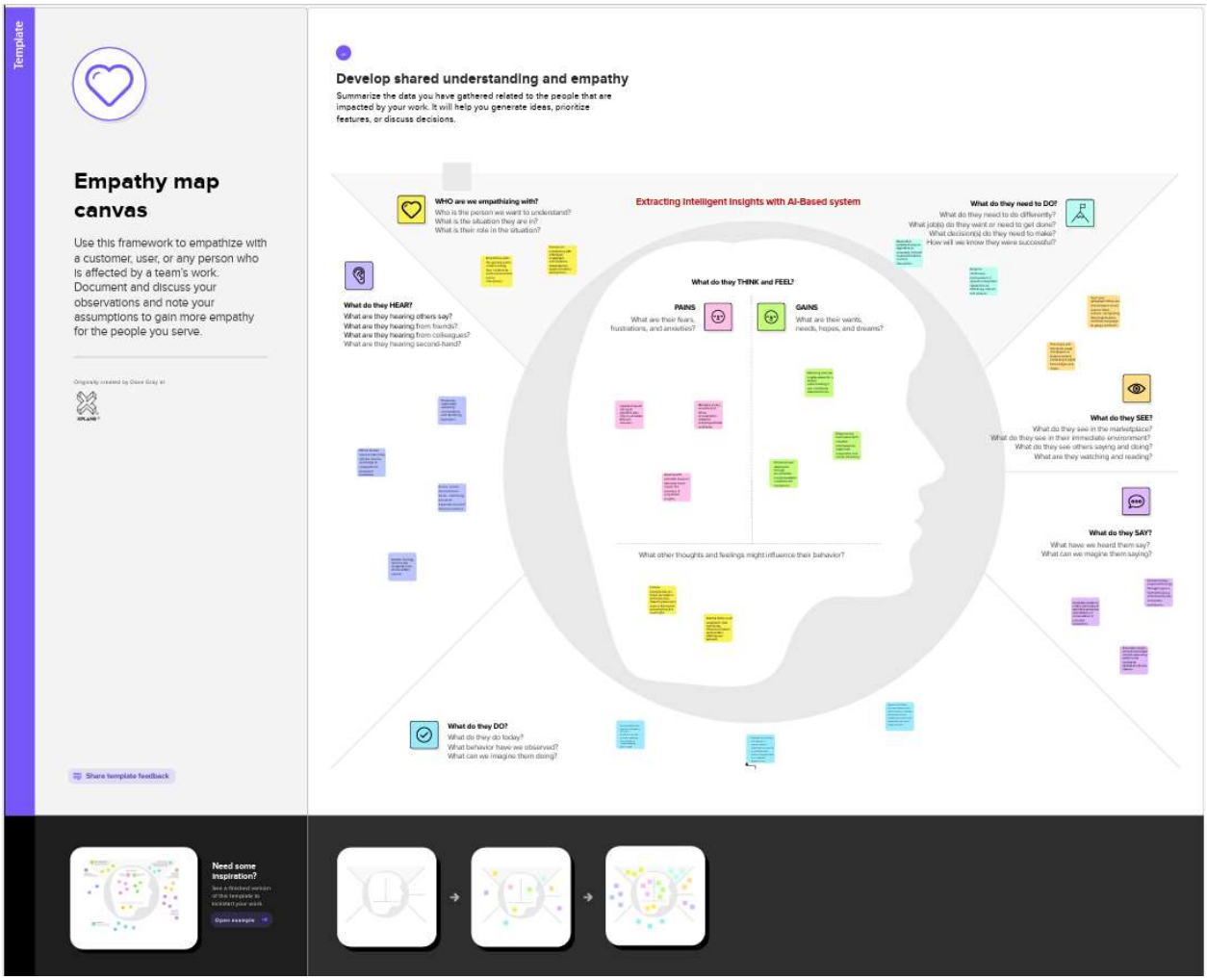
2.3 Problem Statement Definition:

The problem addressed in this study is the development and implementation of AI-based systems capable of extracting intelligent insights from diverse datasets. Specifically, the study aims to:

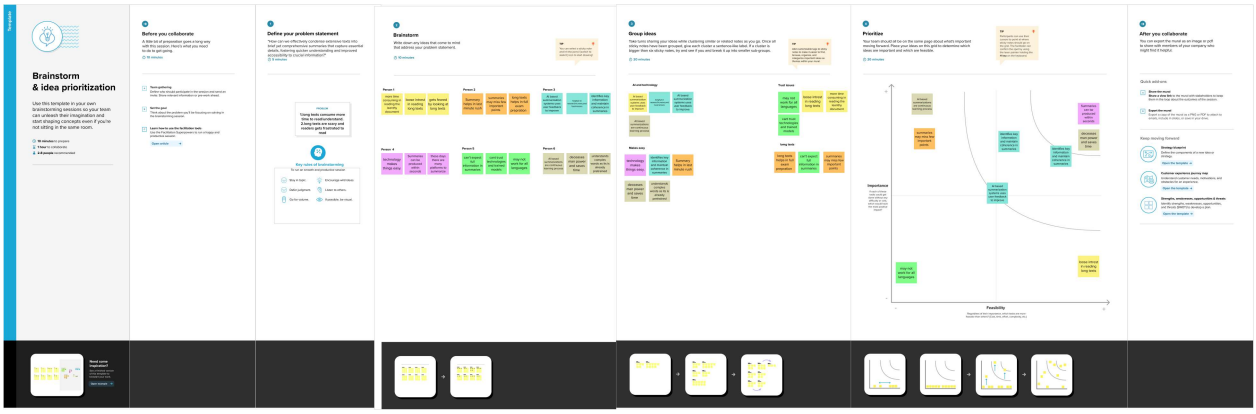
- Investigate existing challenges in extracting insights from complex datasets.
- Explore the state-of-the-art AI techniques, including machine learning, natural language processing, and data visualization, to address these challenges.
- Propose a framework or system architecture that integrates these AI techniques effectively to derive intelligent insights efficiently.
- Evaluate the proposed system's effectiveness in providing actionable insights in real-time or near real-time scenarios.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP



3.2 BRAINSTROMING



4. REQUIREMENT ANALYSIS

4.1 Functional Requirements:

1. Data Collection and Integration:
    - The system should gather data from various sources, including structured and unstructured data formats.
    - It should be able to integrate different types of data such as text, images, videos, and sensor data.
2. Natural Language Processing (NLP) Capabilities:
    - Ability to process and understand natural language text to derive contextual meaning.
    - Support for sentiment analysis, entity recognition, and language translation.
3. Machine Learning Algorithms:
    - Implement machine learning models to identify patterns, trends, and anomalies in the data.
    - Supervised and unsupervised learning techniques to perform clustering, classification, regression, and forecasting.
4. Visualization and Reporting:
    - Provide intuitive visualization tools to present insights in easily understandable formats (graphs, charts, dashboards).
    - Generate customizable reports based on extracted insights.
5. Real-time Processing:
    - Capability to process data in real-time to provide timely insights.

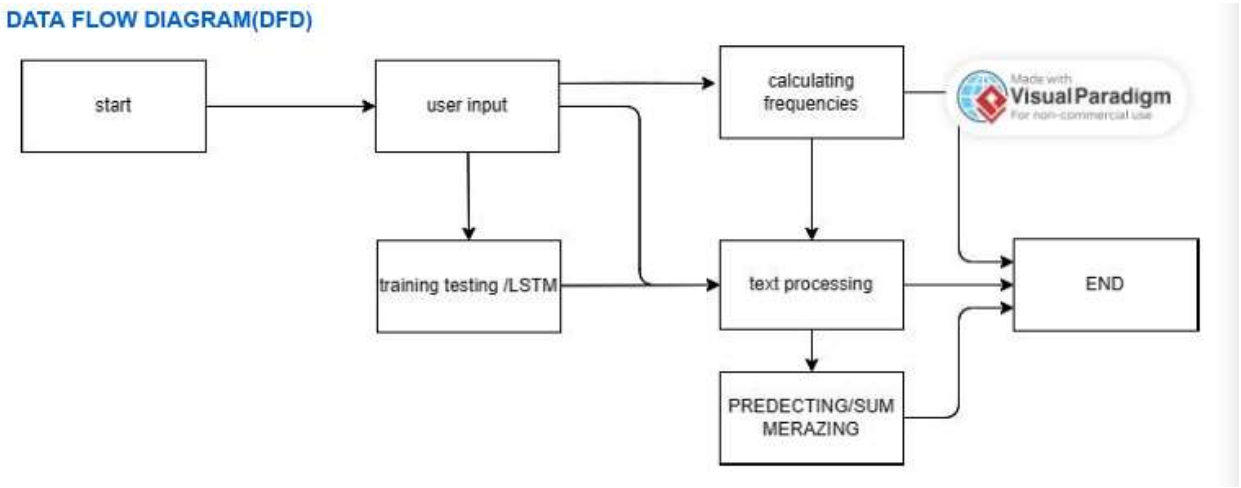
4.2 Non-Functional Requirements:

1. Performance:
    - Ensure the system can handle large volumes of data efficiently and provide insights within acceptable time frames.
    - Response times for queries or analysis should be optimized.
2. Scalability:
    - The system should be scalable to accommodate increased data volumes and user loads without significant performance degradation.
3. Accuracy and Reliability:
    - Maintain a high level of accuracy in insights generated by the system.
    - Ensure reliability by minimizing errors and providing consistent results.
4. Security and Privacy:
    - Implement robust security measures to protect sensitive data throughout the data lifecycle.
    - Adhere to privacy regulations and guidelines while handling and storing data.

5. Interoperability:
    - Ensure compatibility with various data sources, APIs, and other systems to facilitate seamless integration and data exchange.
6. Usability:
    - Provide an intuitive user interface that is easy to navigate and use for both technical and non-technical users.
    - Offer documentation and support for users to effectively utilize the system.

5. PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories



user stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance Criteria	Priority	Releas

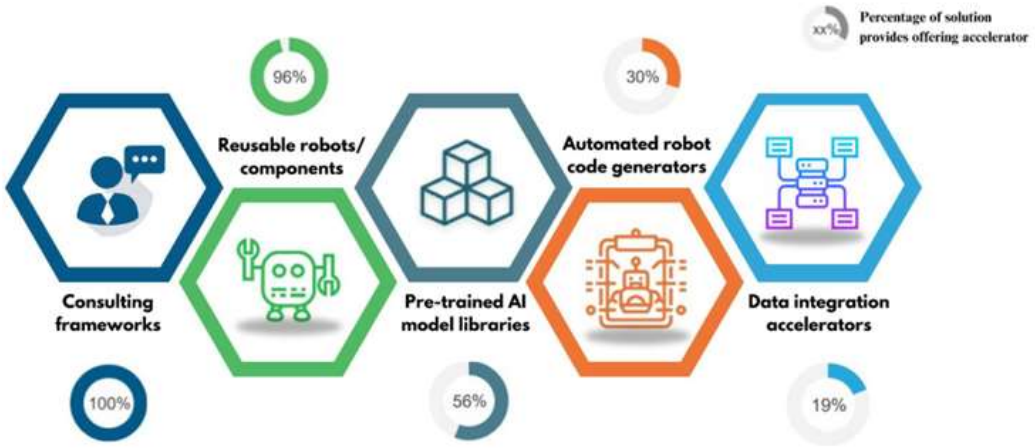
Customer (Mobile user)	Registration	USN-1	As a mobile user, I want to register for the application by entering my email, password, and confirming my password.	- As a user, I can register for the application by entering my email, password, and confirming my password. - Upon registration, I receive a confirmation email. - I can access my account/dashboard.	High	Sprint-
Customer (Mobile user)	AI-driven personalized recommendations based on preferences.	USN-2	As a mobile user, I want real-time AI-generated insights pushed as notifications.	- AI provides personalized recommendations based on user preferences. - Real-time AI-generated insights are pushed as notifications.	High	Sprint 1
Customer (Mobile user)	Registration	USN-3	As a user, I can register for the application through Facebook.	- Users can register and access the dashboard using Facebook Login.	Low	Sprint 2

Customer (Mobile user)	Login	USN-4	As a user, I can register for the application through Gmail.	- System analyzes user behavior and suggests relevant products/services. - Allows users to rate recommendations for improvement.	Medium	Sprint 2
Customer (Mobile user)	Login	USN-5	As a user, I can log into the application by entering email & password.	- AI system alerts about personalized insights, trends, or updates based on user settings. - Notifications are timely, actionable, and customizable.	High	Sprint 2
Customer (Web user)	Login	USN-1	As a web user, I want AI assistance in refining search queries and suggesting related topics.	- AI provides autocomplete and suggests related searches based on user input. - Enhances search accuracy and relevance.	Medium	Sprint 2

Customer Care Executive	Automating Customer Queries	USN-1	Automating customer queries in-built AI system.	- AI system processes FAQs and responds to routine queries. - Escalates complex queries to human support if needed. - Accessible within the system with recommendations.	Medium	Sprint 1
Administrator	Inbuilt or download	USN-1	Understanding and guiding workflow with the ability to view into the administration, generate tips.	- Understanding and guiding workflow within the administration. - Ability to generate tips.	High	Sprint 3
Researcher	Summarizing Research Papers using Login	USN-1	System generates concise summaries maintaining key information. - Allows retrieval of original content if needed. - Accessible using mail.	High	Sprint-2	

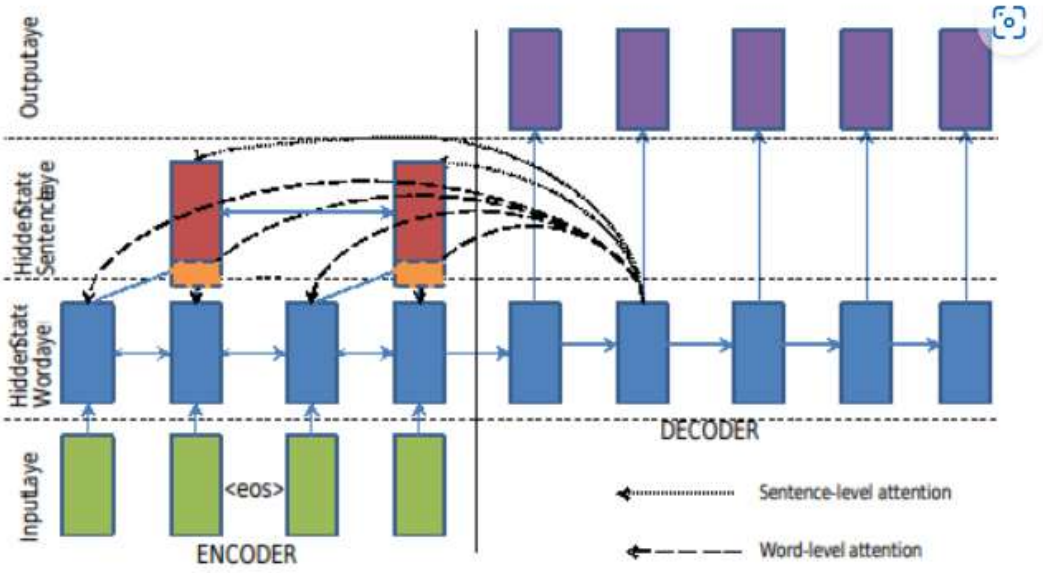
5.2 Solution Architecture

Solution provides are investing in various accelerators to speed time-to-value for clients



6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture



6.2 Sprint Planning & Estimation

Sprint	Functional Requirement	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Environment Setup	Set up development environment	1	High	tejaswini

Sprint-1	Tool & Framework Integration	Integrate required tools and frameworks	1	High	tejaswini
Sprint-1	Data Collection	Gather diverse datasets for analysis	2	High	kyatevardini
Sprint-2	Data Preprocessing	Normalize and preprocess collected data	2	Medium	tejaswini
Sprint-2	Model Selection	Evaluate and choose suitable AI model	3	High	kyatevardine
Sprint-3	Model Implementation	Develop the selected AI model	5	High	kyatevardini
Sprint-3	Testing & Validation	Test model performance and validate	3	High	tejaswini
Sprint-4	Integration with System	Integrate AI model with system	3	High	tejaswini
Sprint-4	User Interface Enhancement	Develop intuitive UI for insights	4	Medium	kyatevardiini
Sprint-5	Performance Optimization	Optimize system for efficient processing	5	High	tejaswini

6.3 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Sprint Release Date (Actual)
Sprint-1	3	8 days	2 Nov 2023	7 Nov 2023	10 Nov 2023
Sprint-2	5	3 days	7 Nov 2023	7 Nov 2023	9 Nov 2023
Sprint-3	10	3 days	7 Nov 2023	8 Nov 2023	9 Nov 2023

Sprint-4	1	8 days	8 Nov 2023	9 Nov 2023	9 Nov 2023
Sprint-5	1	2 days	9 Nov 2023	9 Nov 2023	15 Nov 2023
Sprint-6	8	4 days	9 Nov 2023	9 Nov 2023	13 Nov 2023

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature

Extracting intelligent insights using AI-based systems involves several features and functionalities:

1. **Data Collection and Integration:**
    - **Data Gathering:** Gathering data from diverse sources such as databases, APIs, streaming data, or other repositories.
    - **Data Integration:** Aggregating and integrating data from multiple sources to create a comprehensive dataset.
2. **Preprocessing and Cleaning:**
    - **Data Cleaning:** Removing inconsistencies, handling missing values, and ensuring data quality.
    - **Data Transformation:** Converting data into a suitable format for analysis and modeling.
3. **Feature Engineering:**
    - **Feature Extraction:** Identifying and extracting relevant features from the dataset.
    - **Dimensionality Reduction:** Reducing the number of features while retaining essential information.
4. **Machine Learning Models:**
    - **Model Selection:** Choosing appropriate machine learning algorithms (e.g., regression, clustering, classification) based on the nature of the data and the problem at hand.
    - **Training:** Training models on the extracted features to learn patterns and correlations within the data.
5. **AI-Driven Insights:**
    - **Pattern Recognition:** Leveraging AI algorithms to recognize complex patterns and relationships within the data.
    - Prediction and Forecasting: Using models to predict future trends or outcomes based on historical data.
    - Anomaly Detection: Identifying outliers or unusual patterns that might indicate irregularities or significant events.
6. Visualization and Interpretation:
    - Visual Representation: Creating visualizations (charts, graphs, dashboards) to present

insights in an understandable format.

- **Interpretation:** Interpreting the generated insights to derive actionable conclusions or recommendations.

7. **Feedback Loop and Improvement:**
   - **Feedback Integration:** Incorporating feedback and user inputs to refine models and improve accuracy.
   - **Continuous Learning:** Implementing mechanisms for continuous learning and adaptation to changing data patterns.
8. **Privacy and Security Measures:**
   - **Data Security:** Ensuring data privacy and implementing security measures to protect sensitive information throughout the process.
   - **Compliance:** Adhering to regulatory and compliance standards concerning data usage and processing.

These features collectively enable AI-based systems to extract intelligent insights from data, empowering businesses and organizations to make informed decisions, automate processes, and enhance overall performance.

8. PERFORMANCE TESTING

8.1 rouge scores

```
rouge = Rouge()
scores = rouge.get_scores(generated_summaries, y_test, avg=True)

print("ROUGE Scores:", scores)
```

```
ROUGE Scores: {'rouge-1': {'r': 0.29916179098678986, 'p': 0.06204019193207302, 'f': 0.10234783644995842}, 'rouge-2': {'r': 0.0
5798122525622537, 'p': 0.009273466206080494, 'f': 0.015929906016829948}, 'rouge-l': {'r': 0.2685735075110063, 'p': 0.055602925
96418934, 'f': 0.09175219493898423}}
```

9. RESULTS

9.1 Output Screenshots

```
[ ] summary = [word.text for word in summary]
    summary
```

```
['It represents the culmination of decades of research, where the fusion of advanced algorithms, computational power, and vast datasets has birthed a new era of intelligence augmentation.\n\n',
 "Ethical considerations and questions of governance surround its development, demanding thoughtful discourse and responsible implementation to ensure AI serves humanity's best interests.\n\n",
 'Striking a balance between innovation and ethical responsibility remains a pressing issue as AI continues to permeate every facet of our lives.\n\n',
 'The fusion of AI with other transformative technologies like robotics, quantum computing, and biotechnology holds the promise of even more profound breakthroughs.\n\n',
 "From personalized recommendation systems in entertainment and e-commerce to predictive analytics powering healthcare diagnostics, AI's versatility knows no bounds.\n\n"]
```

```
summary = " ".join(summary)
summary
```

```
'It represents the culmination of decades of research, where the fusion of advanced algorithms, computational power, and vast datasets has birthed a new era of intelligence augmentation.\n\n Ethical considerations and questions of governance surround its development, demanding thoughtful discourse and responsible implementation to ensure AI serves humanity's best interests.\n\n Striking a balance between innovation and ethical responsibility remains a pressing issue as AI continues to permeate every facet of our lives.\n\n The fusion of AI with other transformative technologies like robotics, quantum computing, and biotechnology holds the promise of even more profound breakthroughs.\n\n From personalized recommendation systems in entertainment and e-commerce to predictive analytics powering healthcare diagnostics, AI's versatility knows no bounds.\n\n'
```

10. ADVANTAGES & DISADVANTAGES

Advantages:

1. **Speed and Efficiency:** AI systems can process vast amounts of data at incredible speeds, allowing for quick extraction and analysis of insights, far surpassing human capabilities.
2. **Accuracy and Precision:** AI models can identify patterns and trends in data with high accuracy, reducing human error in analysis and providing more precise insights.
3. **Scalability:** These systems can handle large volumes of data without compromising performance, making them suitable for scaling up analysis as the data grows.
4. **Continuous Learning:** AI systems can continuously learn from new data, adapting and improving their analytical capabilities over time to generate more relevant and accurate insights.
5. **Unbiased Analysis:** When properly trained and monitored, AI can offer objective analysis by minimizing human biases that might affect decision-making.
6. **Automation of Repetitive Tasks:** AI can automate mundane tasks involved in data analysis, freeing up human resources for more complex and creative work.

Disadvantages:

1. **Dependency on Data Quality:** AI systems heavily rely on the quality and relevance of input data. Inaccurate or biased data can lead to flawed insights and decisions.
2. **Lack of Contextual Understanding:** AI may struggle to understand context, sarcasm, or nuanced human language, potentially leading to misinterpretation of data or generating inaccurate insights.
3. **Ethical Concerns:** AI systems might perpetuate biases present in the data used for training, leading to biased or discriminatory outcomes in decision-making.
4. **Complexity and Interpretability:** Some AI models, particularly deep neural networks, are complex and often considered "black boxes," making it challenging to understand and interpret how they arrive at specific insights.
5. **Security and Privacy Risks:** Handling sensitive data poses risks of breaches and privacy violations if not appropriately secured, leading to potential legal and ethical consequences.
6. **Initial Investment and Maintenance:** Developing, implementing, and maintaining AI systems can be costly and require ongoing investments in infrastructure, training, and updates.

11. CONCLUSION

AI-based systems for extracting intelligent insights offer unparalleled advantages in data processing speed, accuracy, and scalability. They revolutionize decision-making processes by automating tasks and providing precise insights. However, challenges persist regarding data quality, interpretability, ethics, and security. Despite these challenges, the current landscape showcases the transformative potential of AI in data analysis.

12. FUTURE SCOPE

The future of AI-based systems for extracting insights holds immense promise:

- **Interpretability Enhancements:** Focus on developing more interpretable AI models to explain their decisions, fostering trust and understanding.
- **Ethical AI Development:** Emphasis on fairness, transparency, and accountability in AI models to mitigate biases and ethical concerns.
- **Human-AI Collaboration:** Shift towards collaborative systems where AI complements human expertise rather than replacing it.
- **Contextual Understanding:** Advancements in natural language processing for improved contextual comprehension, enhancing accuracy in insights.
- **Predictive Analytics:** Evolving AI for better predictive capabilities, foreseeing future trends and behaviors accurately.
- **Data Quality and Privacy:** Efforts to ensure data integrity and privacy to minimize risks associated with sensitive information handled by AI.
- **Accessible AI Solutions:** Democratization of AI technologies, making AI-based insights more accessible to diverse user groups.

The future scope revolves around continuous advancements addressing current limitations while emphasizing ethical, transparent, and collaborative AI development approaches. As these

technologies evolve, they hold the potential to significantly transform decision-making processes across industries, driving innovation and efficiency.

13. APPENDIX

Github links

smartinternz02/SI-GuidedProject-614093-1700370164 (github.com)

video link

Source Code

PYTHON FLASK APPLICATION

```
from flask import Flask, render_template, request
import spacy
from spacy.lang.en.stop_words import STOP_WORDS
from heapq import nlargest

app = Flask(__name__)

@app.route("/")
def homes():
    return render_template('index.html', predictionText='')  # Initial rendering with an empty predictionText

@app.route("/summary", methods=['POST'])
def summary():
    stopWords = list(STOP_WORDS)
    nlp = spacy.load('en_core_web_sm')
    doc = request.form['text']
    print(doc)

    docs = nlp(doc)
    tokens = [i.text for i in docs]
```

```
word_frequencies = {}

for word in docs:
    if word.text.lower() not in stopWords:
        if word.text not in word_frequencies.keys():
            word_frequencies[word.text] = 1
        else:
            word_frequencies[word.text] += 1

maxFrequency = max(word_frequencies.values())

for word in word_frequencies.keys():
    word_frequencies[word] = word_frequencies[word] / maxFrequency

sent_tokenz = [sent for sent in docs.sents]
sentence_score = {}

for sent in sent_tokenz:
    for word in sent:
        if word.text.lower() in word_frequencies.keys():
            if sent not in sentence_score.keys():
                sentence_score[sent] = word_frequencies[word.text.lower()]
            else:
                sentence_score[sent] += word_frequencies[word.text.lower()]

select_len = int(len(sent_tokenz) * 0.3)
summary = nlargest(select_len, sentence_score, key=sentence_score.get)
summary_text = ' '.join([word.text for word in summary])
```

```
    return render_template('index.html', predictionText=summary_text)


if __name__ == "__main__":
    app.run(debug=True)
```

html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Flask App</title>
    <style>
        /* Basic styling for demonstration purposes */
        .tab-content {
            display: none;
        }
        .active-tab {
            display: block;
        }
    </style>
    <script>
        function openTab(tabName) {
            var i, tabContent;
            tabContent = document.getElementsByClassName("tab-content");
            for (i = 0; i < tabContent.length; i++) {
                tabContent[i].style.display = "none";
            }
            document.getElementById(tabName).style.display = "block";
        }
```

```
    </script>
</head>
<body>
    <h1>Welcome to the Flask App</h1>

    <!-- Navigation buttons or links to switch between sections -->
    <button onclick="openTab('home')">Home</button>
    <button onclick="openTab('summarize')">Summarize</button>
    <button onclick="openTab('submit')">Submit</button>

    <!-- Home Section -->
    <div id="home" class="tab-content active-tab">
        <h2>Home</h2>
        <p>Welcome to the home section of the Flask app.</p>
        <!-- Add content specific to the home section if needed -->
    </div>

    <!-- Summarize Section -->
    <div id="summarize" class="tab-content">
        <h2>Summarize Text</h2>
        <form action="/summary" method="POST">
            <textarea name="text" rows="10" cols="50" placeholder="Enter text to
summarize"></textarea><br><br>
            <input type="submit" value="Summarize">
        </form>
    </div>

    <!-- Submit Section -->
    <div id="submit" class="tab-content">
```

```
    <h2>Summary</h2>
    <p>{{ predictionText }}</p>
    <!-- Render the summary text here -->
  </div>
</body>
</html>
```

GitHub & Project Demo Link