# Abstract

In today’s data-driven world, effective visualization is essential for making sense of complex datasets. Traditional 2D visualizations, while widely used, often fall short in conveying multidimensional insights due to their static and flat nature. This project addresses these limitations by developing a system that transforms textual descriptions of numerical data into dynamic visualizations rendered in Virtual Reality (VR) and Augmented Reality (AR) environments.

The system leverages Large Language Models (LLMs) to convert natural language input into structured tabular data in JSON format. A Retrieval-Augmented Generation (RAG) model is then used to determine the most appropriate visualization type—such as bar charts or line charts—based on the structure and content of the tabular data.

These visualizations are rendered as 3D models in VR and AR. In VR, users can explore the data through basic interactions like zooming, panning, and rotating. In AR, visualizations are presented as floating 3D models, offering a lightweight and portable viewing experience. This immersive approach enhances user understanding, enabling deeper insights and improved pattern recognition compared to traditional methods.

The system demonstrates potential across various domains—including education, healthcare, business analytics, and scientific research—where intuitive and interactive data exploration is vital for informed decision-making.

***Key words*:** Virtual Reality (VR), Augmented Reality (AR), Data Visualization, Text-to-Table Conversion, Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), JSON, Natural Language Processing (NLP), Interactive 3D Models, Immersive Analytics, Visualization Systems, Data Interpretation, Pattern Recognition, Data Science, Information Retrieval, Educational Technology, Business Intelligence, Scientific Visualization.

# Contents

List of Figures………………………………………………………………………….….. i List of Tables… ii

[Nomenclature… iii](#_TOC_250012)

1. Introduction… 1
   1. [Problem Definition… 1](#_TOC_250011)
   2. [Motivation 1](#_TOC_250010)
   3. [Scope of Project and Objectives. 1](#_TOC_250009)
   4. [Functional and Non-Functional Requirements 2](#_TOC_250008)
   5. [Organization of the Report 3](#_TOC_250007)
2. Literature Survey… 4
3. Project Plan and Timeline 10
4. Implementation… 12
   1. [Architecture/Block Diagram… 12](#_TOC_250006)
   2. [Algorithm / Methodology 13](#_TOC_250005)
   3. [Technology used… 15](#_TOC_250004)
5. Results and Discussion… 16
6. Conclusion and Future Work… 20
   1. [Conclusions… 20](#_TOC_250003)
   2. [Scope for Future Work… 21](#_TOC_250002)

[References 22](#_TOC_250001)

[Acknowledgements… 23](#_TOC_250000)

**List of Figures**

2.1 Key areas explored in the Literature Survey… 4

3.1 Project Timeline with Task-wise Breakdown and Scheduled Dates… 11

* 1. Workflow Diagram for Generating VR Visualization from Textual Description of Numerical Data… 12
  2. Converting text into structured tables using LLM… 16
  3. Custom dataset for RAG model training… 17
  4. Visualization type prediction by RAG model based on input JSON data… 17
  5. VR environment displaying a 3D visualization generated from JSON input… 18
  6. AR environment displaying a static 3D visualization… 18

# List of Tables



* 1. Summary of Research Papers on Text-to-Table Conversion Methods… 5
  2. Summary of Research Papers on Text Processing Techniques… 7

## Nomenclature



**VR** Virtual Reality: A simulated 3D environment that immerses users in a virtual world, allowing interaction with data.

**AR** A technology that overlays digital 3D visualizations onto the real-world environment through a camera view, enabling contextual data display without full immersion.

**LLM** Large Language Model: A machine learning model designed to process and generate human-like text, used for converting textual data to structured tables.

**RAG** Retrieval-Augmented Generation: A model that enhances accuracy by retrieving relevant information to guide visualization selection.

**3D Visualization** Three-dimensional representation of data in a VR environment for immersive exploration.

**NLP** Natural Language Processing: A field of AI enabling machines to interpret and generate human language.

**Data Filtering** The process of selectively viewing subsets of data based on user-defined conditions.

**Zooming** Adjusting the scale of a visualization to either zoom in on a specific area or get an overview of the data.

**Scale Adjustments** Modifying the dimensions or proportions of a visualization to highlight specific data aspects.

**User Interaction** The capability for users to manipulate and engage with visualized data, such as filtering, rotating, or adjusting the scale.

**Data Representation** The method by which data is presented visually or in a structured format, such as tables or charts.

**Machine Learning (ML)** A subset of AI that allows systems to learn patterns from data and

improve over time.

**Data Science** A field that uses scientific methods, algorithms, and systems to extract knowledge from structured and unstructured data.

**Visualization Design** The practice of creating clear, effective visual representations of data that are easy to interpret.

**Visualization Systems** Platforms that generate and display data visualizations for analysis, from simple charts to complex 3D models.

**Information Retrieval** The process of obtaining relevant data from large datasets to determine the best visualization type.

**Immersive Visualization** A method that allows users to interact with and explore data in a 3D, engaging virtual environment.

**Interactive 3D Models** Visual representations of data in three dimensions, allowing users to manipulate and gain insights.

# Chapter 1 Introduction

This project addresses the challenge of making numerical data more accessible by converting textual descriptions into structured JSON tables and generating visualizations for immersive exploration. These visualizations are rendered in both Augmented Reality (AR) and Virtual Reality (VR) environments. Traditional 2D graphs often lack depth and engagement, while AR/VR offers spatial understanding and improved perception. The system uses Retrieval-Augmented Generation (RAG) models to identify suitable visualizations from the JSON-formatted data. VR enables basic interactions like zoom, rotate, and pan, whereas AR presents static, view-only visualizations. This chapter sets the context for the project, outlining its scope and objectives, and leads into the subsequent discussion on related work, methodology, implementation, and evaluation.

## Problem Definition

Develop a system that transforms textual descriptions of numerical data into suitable visualizations, which are then rendered in an immersive Augmented Reality and Virtual Reality environments for interactive exploration.

## Motivation

Traditional 2D visualizations limit depth and spatial context, making complex data harder to interpret. VR and AR offer immersive 3D environments that enhance pattern recognition and insight generation through spatial visualization, making data analysis more intuitive and engaging.

## Scope of Project and Objectives

The project aims to develop a system that transforms textual descriptions of numerical data into meaningful visualizations rendered in immersive AR and VR environments. Leveraging Large Language Models (LLMs), Retrieval-Augmented Generation (RAG) models, and 3D rendering, the system overcomes the limitations of traditional 2D visualizations by offering a more spatial and intuitive experience.

### Scope of the Project:

* + - **Text-to-Table Conversion:** Use LLMs to extract JSON structured tabular data from textual descriptions.
    - **Visualization Identification:** Implement a RAG model to determine the most appropriate visualization type (e.g., bar chart, line chart) from the JSON structured table.
    - **3D Visualization Rendering:** Generate and display the visualizations in both VR and AR environments.
    - **VR Interaction:** Support basic VR interactions such as zoom, rotate, and pan.
    - **AR Visualization:** Provide static, view-only visualization overlays in AR for spatial context.

### Objectives of the Project:

The primary objectives of this project are:

* + - To develop a system that converts textual descriptions of numerical data into structured tabular form (in JSON format) using LLMs.
    - To design and implement a RAG-based model to identify the most suitable type of visualization (e.g., bar chart, line chart).
    - To render 3D visualizations in VR for immersive data exploration with basic interactions like zoom, rotate, and pan.
    - To enable static, spatial visualization in AR for passive observation of the data in real-world context.
    - To provide a novel approach to data visualization that enhances user understanding and insight generation beyond traditional 2D formats.

## Functional and Non-Functional Requirements

### Functional Requirements

* + - Accept textual description of numerical data as input.
    - Convert the textual input into a structured table (JSON format) using LLMs.
    - Use a RAG model to identify the appropriate visualization type (e.g., bar chart, line chart).
    - Generate 3D visualizations from the structured data.
    - Render the visualization in VR with basic interaction (zoom, rotate, pan).
    - Display static, view-only visualizations in AR.

### Non-Functional Requirements

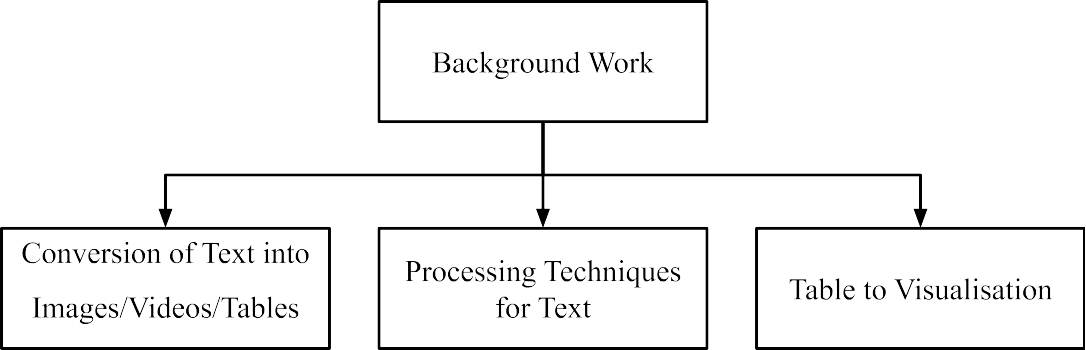
* + - **Response Time:** The system should convert input and render visualizations in AR/VR within a reasonable time.
    - **Accuracy:** The RAG model should consistently identify suitable and meaningful visualizations based on the input data.

## Organization of the Report

Chapter 1 introduces the problem, motivation, and objectives of the project. It also outlines the scope, describing how textual descriptions of numerical data are converted into 3D visualizations rendered in immersive AR and VR environments. Chapter 2 presents a comprehensive literature survey, reviewing existing work in text-to-table conversion, data visualization techniques, and immersive technologies, and identifies gaps this project aims to address. Chapter 3 outlines the project plan and timeline, detailing phases such as requirement analysis, system design, implementation, and testing. Chapter 4 focuses on implementation, explaining the system architecture, key algorithms, and technologies used — including Large Language Models (LLMs) and Retrieval-Augmented Generation (RAG) models. Chapter 5 discusses results, evaluating the effectiveness of the system in generating visualizations and rendering them in AR and VR, with emphasis on the basic interaction capabilities in VR. Chapter 6 concludes the report by summarizing findings and suggesting future enhancements, such as improving model accuracy and expanding visualization capabilities.

# Chapter 2 Literature Survey

This chapter presents a comprehensive review of existing research that informs the development of the system. The literature survey is divided into three key areas central to the project: Conversion of Text into Table/Images/Videos, Processing Techniques for Text, and Table to Visualization. In the first section, we explore various methods for converting raw text data into structured formats, such as tables, which are necessary for further analysis and visualization. The second section examines advanced text processing techniques, including the use of BERT-based embeddings for text clustering and information extraction, which enable more efficient data structuring. The third section focuses on the challenge of transforming structured data into visualizations, evaluating both LLMs and RAG models for generating suitable visualizations, with a discussion on their limitations and biases. This chapter highlights the gaps in existing research and supports the need for an integrated system that combines text-to-table transformation, semantic text processing, and immersive visualization in AR and VR—where VR allows basic interactions and AR provides static spatial rendering.



**Figure 2.1:** Key areas explored in the Literature Survey

### Conversion of Text into Table/Images/Videos

This step explores various approaches to transforming raw text data into structured forms, such as tables, images, or videos, that can be used for further analysis or visualization. Several works have explored Large Language Models (LLMs) and sequence-to-sequence (seq2seq) models for

generating structured data from unstructured text. These models, such as BERT and GPT, are used to extract key information and organize it into tabular formats, which is critical for downstream visualization tasks.

**Table 2.1:** Summary of Research Papers on Text-to-Table Conversion Methods

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Year** | **Title** | **Publication (IEEE/Journal)** | **What is the paper about? (Aspects)** | **Methodology (Steps in 2-3 lines)** | **Datasets (Size, Type, etc.)** | **Results (Validation Metrics)** | **Advantages** | **Limitations** | **Ref No.** |
|  |  |  | This paper |  |  |  |  | The system |  |
|  |  |  | introduces Revilio, |  |  |  |  | is primarily |  |
|  |  |  | a system that uses |  |  |  |  | designed for |  |
|  |  |  | large language |  |  | The system |  | scenarios |  |
|  |  |  | models to | The system first |  | outperform |  | where table |  |
|  |  |  | reconstruct tables | detects headers | The paper | s traditional |  | column |  |
|  |  |  | from free-form text, | from text, generates | evaluates | methods |  | boundaries |  |
|  |  |  | especially when | a table sketch using | Revilio on | with an | It can | are lost and |  |
|  |  |  | column boundaries | an LLM, and then | multiple | accuracy | handle large | may not |  |
|  |  | ACM | are lost. It | refines the table | datasets, | improveme | tables | generalize |  |
|  |  | International | addresses | using a | including | nt of | effectively | well to |  |
|  | Tabularis | Conference on | challenges like | "generate-and-rank | those | 5.8–11.3% | and | other types |  |
|  | Revilio: | Information and | ensuring semantic | " strategy to ensure | containing | over neural | improves | of |  |
|  | Convertin | Knowledge | and syntactic | syntactic and | tables with | and | table | text-to-table |  |
|  | g Text to | Management | consistency in table | semantic | over 100,000 | symbolic | reconstructi | conversion |  |
| 2024 | Tables | (CIKM 2024) | generation. | consistency. | rows. | baselines. | on accuracy. | tasks. | [1] |
|  |  |  |  |  |  |  | The |  |  |
|  |  |  |  |  |  |  | approach |  |  |
|  |  |  |  |  |  |  | ensures |  |  |
|  |  |  |  |  |  |  | syntactically |  |  |
|  |  |  |  | First, the system |  |  | valid tables | The model |  |
|  |  |  | This paper presents | generates the table |  | Achieved | and can | requires |  |
|  |  |  | a two-stage | structure (headers) |  | up to 20% | utilize large | high-quality |  |
|  | gTBLS: |  | approach to | from text using |  | improveme | pre-trained | training |  |
|  | Generatin |  | converting | conditional text | The paper | nt in | models in a | data and |  |
|  | g Tables |  | unstructured text | generation. Then, it | evaluates on | BERTScore | zero-shot | fine-tuning |  |
|  | from Text |  | into structured | formulates | datasets like | s for table | configuratio | to achieve |  |
|  | by |  | tables. It uses a | questions based on | E2E, | content | n, which is | optimal |  |
|  | Condition |  | model that | these headers and | WikiTableTe | generation | beneficial | results, |  |
|  | al |  | generates table | uses an LLM to | xt, and | tasks | for many | which can |  |
|  | Question | arXiv preprint | structures (headers) | answer them, filling | WikiBio, | compared | real-world | be |  |
|  | Answerin | arXiv:2403.1445 | and content by | the table with | among | to previous | applications | resource-int |  |
| 2024 | g | 7 | asking questions. | appropriate content. | others. | methods. | . | ensive. | [2] |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2024 | Large Language Models as Generaliz able  Text-to-T able Systems | Proceedings of the Association for Computational Linguistics (ACL 2024) | This paper explores how large language models (LLMs), such as GPT-3, can be used for the text-to-table task without fine-tuning on specific datasets. It evaluates how LLMs can  generalize across various input texts and generate valid tables. | LLMs are prompted to generate tables from unstructured text, and the task is evaluated on multiple datasets using zero-shot and few-shot learning techniques. | Datasets like E2E,  WikiTableTe xt, and  WikiBio are used for evaluation, focusing on different types of text-to-table conversion tasks. | The approach demonstrat es the  capability to generalize well across various datasets, with improveme nts in  output quality compared to previous fine-tuned models. | No need for extensive fine-tuning or specific schema design, making it adaptable to various text formats and datasets. | Performanc e can be inconsistent depending on the  complexity and type of text, as LLMs may struggle with  non-standar d or highly structured data. | [3] |
| 2024 | On the  Use of Large Language Models for Table Tasks | VLDB/NeurIPS | Examines how LLMs can generate SQL queries from natural language text using  retrieval-augmented generation (RAG). | Implements prompting techniques and  fine-tuning on domain-specific tables to improve SQL accuracy. | TabFact, Spider (large-scale tabular question-ans wering datasets). | Fine-tuned models improve SQL  accuracy over zero-shot  approaches. | LLMs can generalize across domains. | Requires high computatio nal resources; struggles with complex queries | [4] |
| 2022 | Text-to-T able: A New Way of Informati on Extractio n | ACL | Proposes converting unstructured text to structured tables  using seq2seq models. | Uses fine-tuned  seq2seq models  with table  constraint and relation embeddings to extract structured data. | Rotowire, E2E,  WikiTableTe xt, WikiBio (various sizes, sports, Wikipedia, open-domain tables). | Seq2seq models outperform RE/NER  models; BART-larg e improves extraction accuracy. | No need for predefined schemas; works on long texts. | Struggles with text diversity, reasoning, and large tables | [5] |
| 2023 | Towards Controlle d  Table-to- Text Generatio n with Scientific Reasonin g | IEEE/Scientific NLP Conference | Focuses on  controlled text  generation from tabular data with a scientific reasoning component. | Introduces CTRLSciTab dataset and uses a retriever-generator model with external domain-specific knowledge. | CTRLSciTab (8,967  table-descript ion pairs with scientific knowledge). | CTRLSciT  abNet (Bart) outperform s GPT-3.5,  improving fluency and factuality. | Uses domain-spe cific knowledge for improved accuracy. | Struggles with hallucinatio n and  aligning generated text with scientific facts | [6] |

### Processing Techniques for Text

Text processing techniques, including text clustering and information extraction, have been key to understanding how raw text data can be structured. Techniques such as BERT-based embeddings have been applied to text clustering, and various NLP methods have been developed to extract significant data from the text. These processing methods enable more accurate and efficient transformation of text into structured formats (like tables) that are easier to analyze and visualize.

**Table 2.2:** Summary of Research Papers on Text Processing Techniques

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Year** | **Title** | **Publication (IEEE/Journal)** | **What is the paper about? (Aspects)** | **Methodology (Steps in 2-3 lines)** | **Datasets (Size, Type, etc.)** | **Results (Validation Metrics)** | **Advantages** | **Limitations** | **Ref No.** |
|  |  |  |  |  |  | Achieved |  |  |  |
|  |  |  |  |  |  | 88.2% |  |  |  |
|  |  |  |  |  |  | sentiment | 1. Improves |  |  |
|  |  |  |  |  |  | classificatio | financial |  |  |
|  |  |  |  | 1. Pretrained on a |  | n accuracy | sentiment | FinBERT, |  |
|  |  |  |  | large corpus of |  | (higher | classificatio | like other |  |
|  |  |  |  | financial |  | than | n accuracy. | deep |  |
|  |  |  |  | documents (SEC |  | LSTM, | 2. Works | learning |  |
|  |  |  | FinBERT is a | filings, earnings |  | CNN, and | well with | models, |  |
|  |  |  | domain-specific | calls, analyst | Financial | traditional | small | lacks |  |
|  |  |  | adaptation of BERT | reports). 2. | filings (SEC | methods). | training | interpretabil |  |
|  | FinBERT: |  | for financial text | Fine-tuned on | 10-K, 10-Q), | Outperform | datasets. 3. | ity, making |  |
|  | A Large |  | analysis. It is | sentiment | analyst | ed BERT in | Outperform | its decision |  |
|  | Language |  | trained on financial | classification and | reports, and | financial | s traditional | process |  |
|  | Model for |  | reports, earnings | ESG-related | 136,578 | text | ML models | opaque—a |  |
|  | Extractin |  | calls, and analyst | discussions. 3. | earnings call | classificatio | and even | key |  |
|  | g |  | reports to enhance | Compared against | transcripts. | n, | general | challenge in |  |
|  | Informati |  | sentiment | traditional ML | Total dataset | especially | BERT for | finance |  |
|  | on from | Contemporary | classification and | models (SVM, RF, | size: 4.9 | with small | finance | where |  |
|  | Financial | Accounting | financial text | LSTM, CNN) and | billion | training | applications | transparenc |  |
| 2023 | Text | Research, Wiley | understanding. | general BERT. | tokens. | samples. | . | y is crucial. | [7] |
|  | The |  |  | The study applies |  | BERT-base |  |  |  |
|  | Performa |  | This paper | BERT to generate |  | d |  | The study |  |
|  | nce of |  | evaluates the | text embeddings | Utilized three | representati | BERT | focuses on |  |
|  | BERT as |  | effectiveness of | and compares | popular text | ons | captures | unsupervise |  |
|  | Data |  | BERT embeddings | clustering | clustering | outperform | contextual | d learning; |  |
|  | Represent |  | in text clustering | performance using | datasets: AG | ed TF-IDF | information, | results may |  |
|  | ation of |  | tasks, comparing | algorithms like | News, | in 28 out of | leading to | vary with |  |
|  | Text |  | them with | k-means and deep | DBpedia, and | 36 metrics, | improved | different |  |
|  | Clusterin | Journal of Big | traditional TF-IDF | embedded | 20 | including | clustering | clustering |  |
| 2022 | g | Data | representations. | clustering | Newsgroups. | clustering | performance | algorithms. | [8] |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | accuracy (ACC),  normalized mutual information (NMI), and adjusted rand index (ARI). |  |  |  |
|  |  |  |  |  |  |  | Offers a |  |  |
|  |  |  | This survey |  |  |  | detailed |  |  |
|  |  |  | provides a | The paper reviews |  |  | understandi |  |  |
|  | A Survey |  | comprehensive | various text |  |  | ng of the | As a survey, |  |
|  | of Text |  | overview of text | representation |  |  | progression | it doesn't |  |
|  | Represent |  | representation | techniques, |  |  | and | provide |  |
|  | ation and |  | methods in NLP, | discussing their |  |  | applications | experimenta |  |
|  | Embeddin |  | from early | evolution, | Not | Not | of text | l validations |  |
|  | g |  | techniques to | applications, and | applicable | applicable | representati | or |  |
|  | Techniqu |  | advanced | performance in | (survey | (survey | on | comparison |  |
| 2023 | es in NLP | IEEE Access | embeddings. | NLP tasks. | paper). | paper). | techniques. | s. | [9] |
|  | Graph-Ba |  |  |  |  |  |  |  |  |
|  | sed Text |  |  |  |  |  |  |  |  |
|  | Represent |  |  |  |  |  |  |  |  |
|  | ation and |  |  | The paper analyzes |  |  | Highlights |  |  |
|  | Matching: |  | This review focuses | various |  |  | the potential |  |  |
|  | A Review |  | on graph-based | graph-based text |  |  | of |  |  |
|  | of the |  | methods for text | representation |  |  | graph-based | Lacks |  |
|  | State of |  | representation and | techniques, their |  |  | representati | experimenta |  |
|  | the Art | IEEE | matching, | methodologies, and |  |  | ons in | l |  |
|  | and | Transactions on | discussing their | effectiveness in | Not | Not | capturing | comparison |  |
|  | Future | Knowledge and | applications and | tasks like text | applicable | applicable | complex | s; primarily |  |
|  | Challenge | Data | future research | matching and | (review | (review | relationship | theoretical |  |
| 2020 | s | Engineering | directions. | retrieval. | paper). | paper). | s in text. | analysis. | [10] |
|  | From |  |  |  |  |  |  |  |  |
|  | Text to |  |  |  |  |  |  |  |  |
|  | Knowled |  |  |  |  |  |  |  |  |
|  | ge with |  | This paper explores | The authors discuss |  |  | Provides |  |  |
|  | Graphs: |  | challenges and | integrating |  |  | insights into |  |  |
|  | Modellin |  | trends in | linguistics, NLP, |  |  | combining |  |  |
|  | g, |  | representing and | and graph databases |  |  | multiple |  |  |
|  | Querying |  | querying | to transform |  |  | disciplines | Conceptual |  |
|  | and |  | knowledge | unstructured text | Not | Not | for effective | framework |  |
|  | Exploitin |  | extracted from text | into structured | applicable | applicable | knowledge | without |  |
|  | g Textual |  | using graph-based | knowledge | (conceptual | (conceptual | representati | empirical |  |
| 2023 | Content | arXiv preprint | models. | representations. | paper). | paper). | on. | validation. | [11] |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A Novel |  |  |  |  |  |  |  |  |
|  | Multidim |  |  |  |  |  |  |  |  |
|  | ensional |  |  |  |  |  |  |  |  |
|  | Reference |  |  |  |  |  |  |  |  |
|  | Model for |  |  |  |  |  |  |  |  |
|  | Heteroge |  |  |  |  | Demonstrat | Enhances | May require |  |
|  | neous |  | This study |  |  | ed | processing | complex |  |
|  | Textual |  | introduces a model | The proposed |  | improved | of | integration |  |
|  | Datasets |  | to handle | model integrates |  | extraction | heterogeneo | of various |  |
|  | Using |  | heterogeneous | multiple linguistic | Evaluated on | of | us textual | linguistic |  |
|  | Context, |  | textual datasets by | features to enhance | datasets with | meaningful | data by | features; |  |
|  | Semantic |  | leveraging context, | information | varying sizes | information | considering | scalability |  |
|  | and |  | semantic, and | extraction from | and types to | across | multiple | needs |  |
|  | Syntactic |  | syntactic | diverse text | test | diverse | linguistic | further |  |
| 2023 | Clues | arXiv preprint | information. | sources. | adaptability. | datasets. | aspects. | assessment. | [12] |

### Table to Visualization

Once text is converted into a table, the next challenge is transforming that table into a suitable visualization. Text-to-visualization attempts have been made; table-to-visualization can be done using LLMs. However, our findings show that LLMs tend to be biased towards generating specific visualization types, such as pie charts and bar charts, especially when provided with only a few examples. This bias arises from the LLM's inherent tendency to favor these common visualization types, limiting the diversity and appropriateness of the generated visualizations. RAG (Retrieval-Augmented Generation) models are more effective at selecting the most appropriate visualization based on data content. This step is crucial for ensuring meaningful and context-aware visualizations within the immersive AR/VR environments used in this project.

### Outcomes of Background Work:

**Visual Representation vs Textual Representation:** Research consistently shows that visual representations of data are far easier to interpret than textual data. This highlights the importance of the project's goal of converting text into meaningful visualizations.

**Text Processing:** Various efforts have been made in text processing, particularly in clustering and extracting important data using models like BERT, which have been pivotal in understanding and structuring text data.

**Text to Visualization:** While text-to-visualization techniques are still emerging, models such as LLMs have demonstrated the capability to perform the crucial task of converting tables into visual representations effectively.

# Chapter 3

**Project Plan and Timeline**

This chapter outlines the structured roadmap adopted for the successful development of the system. It begins with the Project Plan, which breaks the system into modular tasks: from requirement analysis and text-to-table conversion using LLMs, to visualization type selection via RAG models, and immersive visualization rendering in AR and VR. Each stage is defined with its objective and associated technologies. The plan ensures a streamlined development approach that integrates natural language processing, data structuring, and immersive visualization. The second part details the Project Timeline, presenting a task-wise schedule mapped across development phases. A Gantt-style chart illustrates timelines for activities such as requirement gathering, technology selection, LLM testing, dataset preparation, visualization pipeline setup, and AR/VR integration. This structured timeline supports a logical progression and aligns with project deadlines, ensuring on-time delivery.

### Project Plan

* **Requirement Analysis:** Define system inputs, expected outputs, and confirm data flow.
* **Text-to-Table Module:** Use LLMs to convert textual descriptions into structured tabular data (JSON format).
* **Visualization Selection:** Apply a RAG model to determine the most appropriate chart type based on data content.
* **3D Visualization Rendering:** Convert selected visualization types into 3D models and render them in AR and VR environments.
* **VR Interaction:** Enable basic controls like zoom, rotate, and pan for immersive data exploration.
* **AR Visualization:** Display static, spatial visualizations in AR for passive viewing.
* **System Integration and Testing:** Combine all modules and verify overall functionality, performance, and visual accuracy.

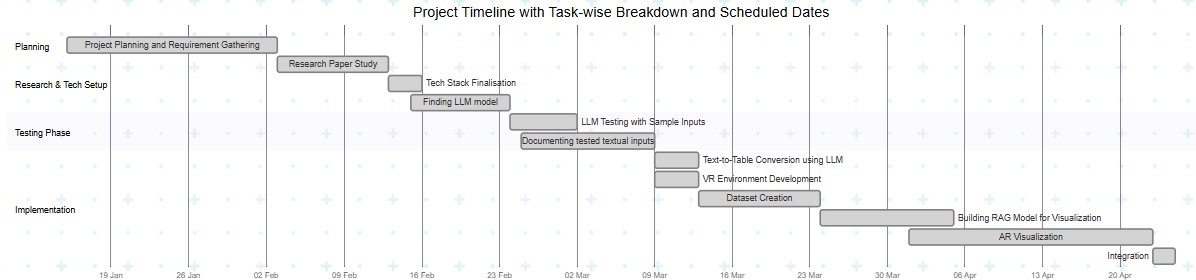
### Project Timeline

The Gantt chart titled "Project Timeline with Task-wise Breakdown and Scheduled Dates" outlines the end-to-end phases from January to April 2025. It is divided into four main sections:

1. **Planning (mid-January):** Covered initial project planning and requirement gathering.
2. **Research & Tech Setup (late January to mid-February):** Focused on studying research papers, finalizing the tech stack, and selecting a suitable LLM model.
3. **Testing Phase (late February to early March):** Involved testing the LLM with sample inputs and documenting the results.
4. **Implementation (March to April):** Included tasks such as converting text to tables, developing the VR environment, dataset creation, building a RAG model for visualization, developing AR visualization, and finally integrating all components.

Each task is time-boxed, and completed phases are visually marked, showing steady progress toward project completion.

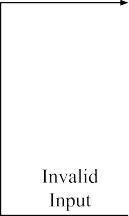
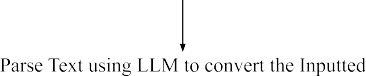
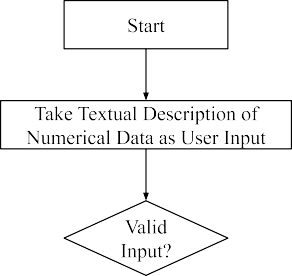
**Table 3.1:** Project Timeline with Task-wise Breakdown and Scheduled Dates



# Chapter 4 Implementation

This section details the end-to-end system implementation for converting user-submitted textual descriptions of numerical data into immersive 3D VR and AR visualizations. It starts with a description of the frontend interface, where users input their data in natural language and choose between three options—VR generation, chart generation via a RAG model, or AR visualization. The process continues with input validation to ensure the presence of meaningful numerical content. Valid input is then passed to a Large Language Model (LLM) to extract structured tabular data. This structured data is further processed by a RAG model trained on sector-specific datasets to determine the most appropriate visualization type. The selected chart is rendered in a 3D VR environment using Unity and Three.js, offering zoom functionality for deeper insight. Additionally, the same chart can be rendered in an AR environment using Unity and AR Foundation, allowing users to place and explore data visualizations in their real-world surroundings. This chapter highlights how each technology—LLMs, RAG models, Unity, and AR/VR frameworks—contributes to transforming raw textual input into an engaging, spatially interactive experience.

## Architecture / Block Diagram



**Figure 4.1:** Workflow for Textual Data to VR Visualization

## Algorithm / Methodology

### Frontend Interface

The process begins with a simple and intuitive frontend interface designed to accept a textual description of numerical data from the user.

The interface contains a text input box, where the user can enter natural language input like:

I have 100 Rs. I spent 20 on clothes, 10 on food.

Below the input box, there are three action buttons:

* + - 1. Generate VR – Initiates the generation of a Virtual Reality-based 3D visualization.
      2. Generate Visualisation Chart – Uses a Retrieval-Augmented Generation model to process the table achieved from the LLM model and generate a visualisation chart.
      3. Generate AR – Produces an Augmented Reality visualization for mobile

### Input Validation

Once the form is submitted, the system performs validation on the entered text:

* It checks that the input falls within an acceptable word limit, ensuring it's neither too short to lack meaning nor too long to overwhelm the model
* It confirms the presence of numerical data—quantities, percentages, counts, or monetary figures—necessary for further processing. If the input fails either check, the system alerts the user and prompts them to revise and resubmit the input.

### Text Parsing into Structured Tabular Format Using LLM

* For valid input, the text is passed to a Large Language Model (LLM) that processes the natural language and extracts structured data.
* The LLM identifies key attributes (e.g. years, counts, ratios, and categories), and organizes them into a table.

### Chart Type Selection Using a RAG Model

* The structured table is then passed into a Retrieval-Augmented Generation (RAG) model.
* This model is trained on a custom dataset of 50 records from the education, agriculture, environment, commerce and finance sectors (sourced from [community.data.gov.in](http://community.data.gov.in/)), helping it identify contextually appropriate visualization types.
* Depending on the tabular content, the RAG model selects a 3D visualization type such as bar chart or line graph.

### 3D VR Visualization Rendering Using Unity and Three.js

Based on the selected visualization type, the system uses Unity and Three.js to render a 3D chart in a Virtual Reality environment. Inside the VR environment, users can interact with the visualization to enhance their understanding. Interaction features include Zooming In/Out to closely inspect specific data points or get a full overview.

### Completion and Insight Extraction

* Once the interactive 3D visualization is rendered, users can explore the data in an immersive environment.
* The process concludes with the user gaining clear, intuitive insights from their original textual input—bridging the gap between raw data and visual understanding without manual chart creation.

### 3D AR Visualization Rendering Using Unity and AR Foundation

* Once the chart is generated, Unity and AR Foundation are used to build a 3D chart, which is rendered in an Augmented Reality environment on the user’s Android device. The AR experience includes Color-coded X, Y, and Z axes, labeled using TextMeshPro and Chart geometry (bars, lines) built using Unity primitives.
* The AR-based 3D chart helps users better understand data by placing it in their real-world environment. With clearly labeled, color-coded axes and 3D shapes like bars or lines, users can easily see patterns and relationships. Displaying data

in space makes complex information more intuitive and visually engaging, especially for presentations or learning.

## Technology used

### Unity (2023.1.0)

Used to create the Augmented Reality (AR) environment and render 3D visualizations. It enables immersive viewing of the visualized data within the AR space.

### Python (3.10)

Acts as the backbone for data processing, handling input/output flow, and managing API integration between various components of the system, including the LLM and visualization modules.

### Large Language Models (LLMs)

Responsible for converting unstructured textual input into structured tabular data. The LLM extracts entities, numerical values, and contextual information to form a usable dataset.

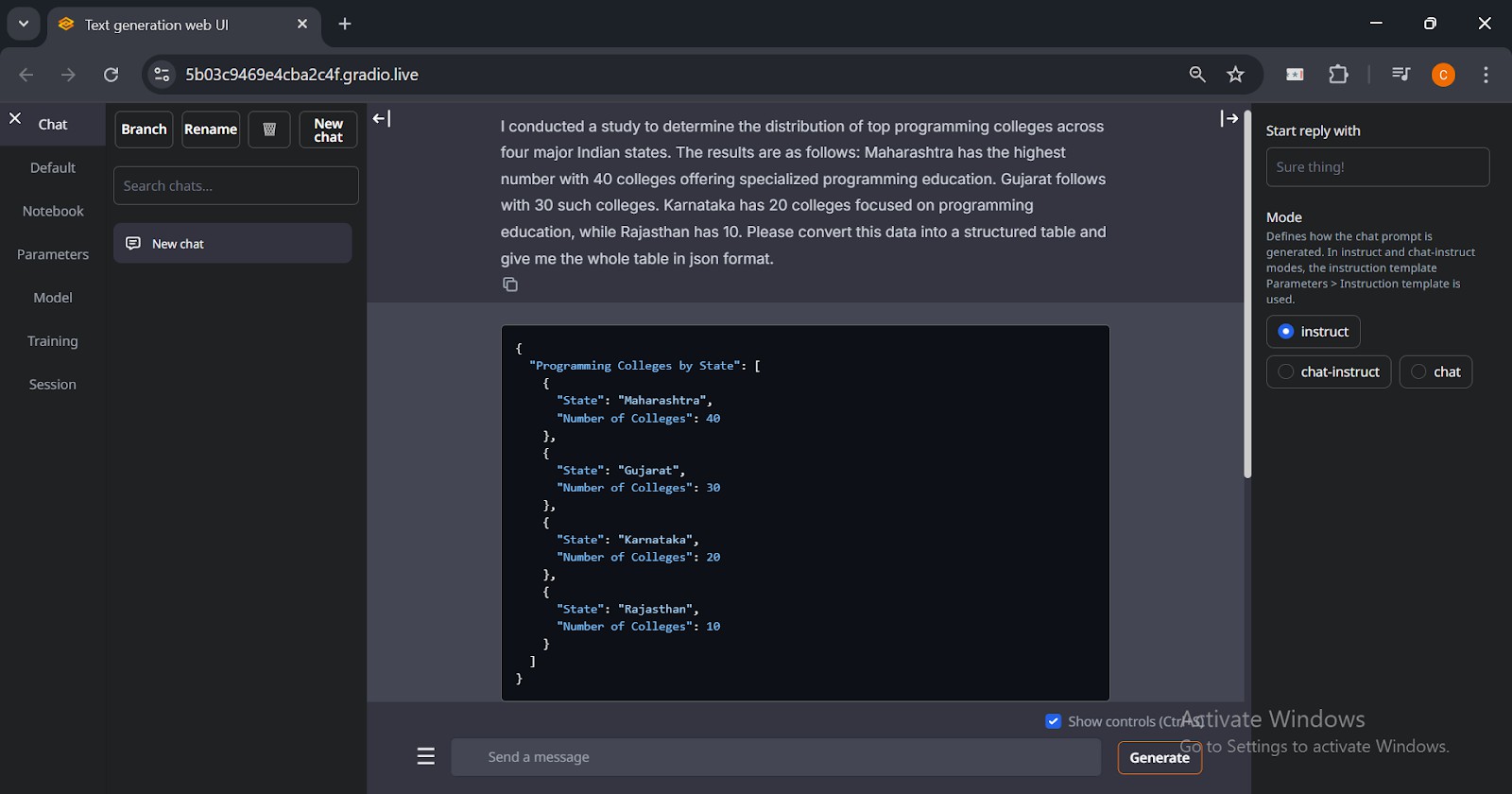
### Retrieval-Augmented Generation (RAG) Model

Analyzes the structure and semantics of the tabular data to determine the most suitable type of data visualization, such as bar charts or line graphs.

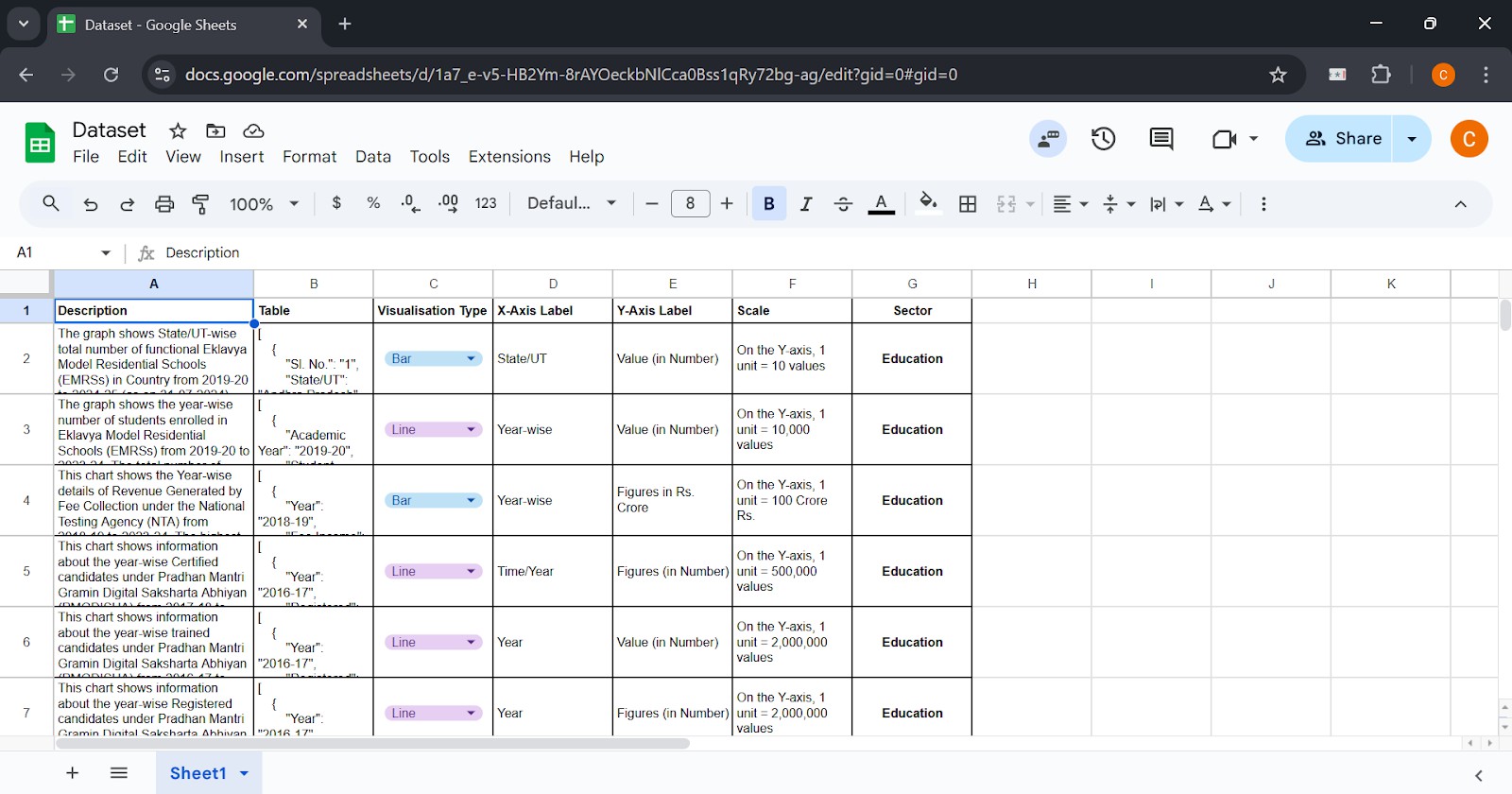
# Chapter 5 Results and Discussion

The developed system successfully converts textual descriptions of numerical data into structured tabular format in JSON using Large Language Models (LLMs). A custom dataset was created to train the Retrieval-Augmented Generation (RAG) model, enhancing its ability to map the JSON-formatted data to the most relevant visualization type, such as a bar chart or line chart. These visualizations are then rendered as 3D models for immersive viewing in both Virtual Reality (VR) and Augmented Reality (AR). While VR supports basic interaction through zooming, rotating, and panning, AR provides static spatial placement for passive observation. This section evaluates the system’s performance across each stage, highlighting its effectiveness, accuracy, and observed limitations.

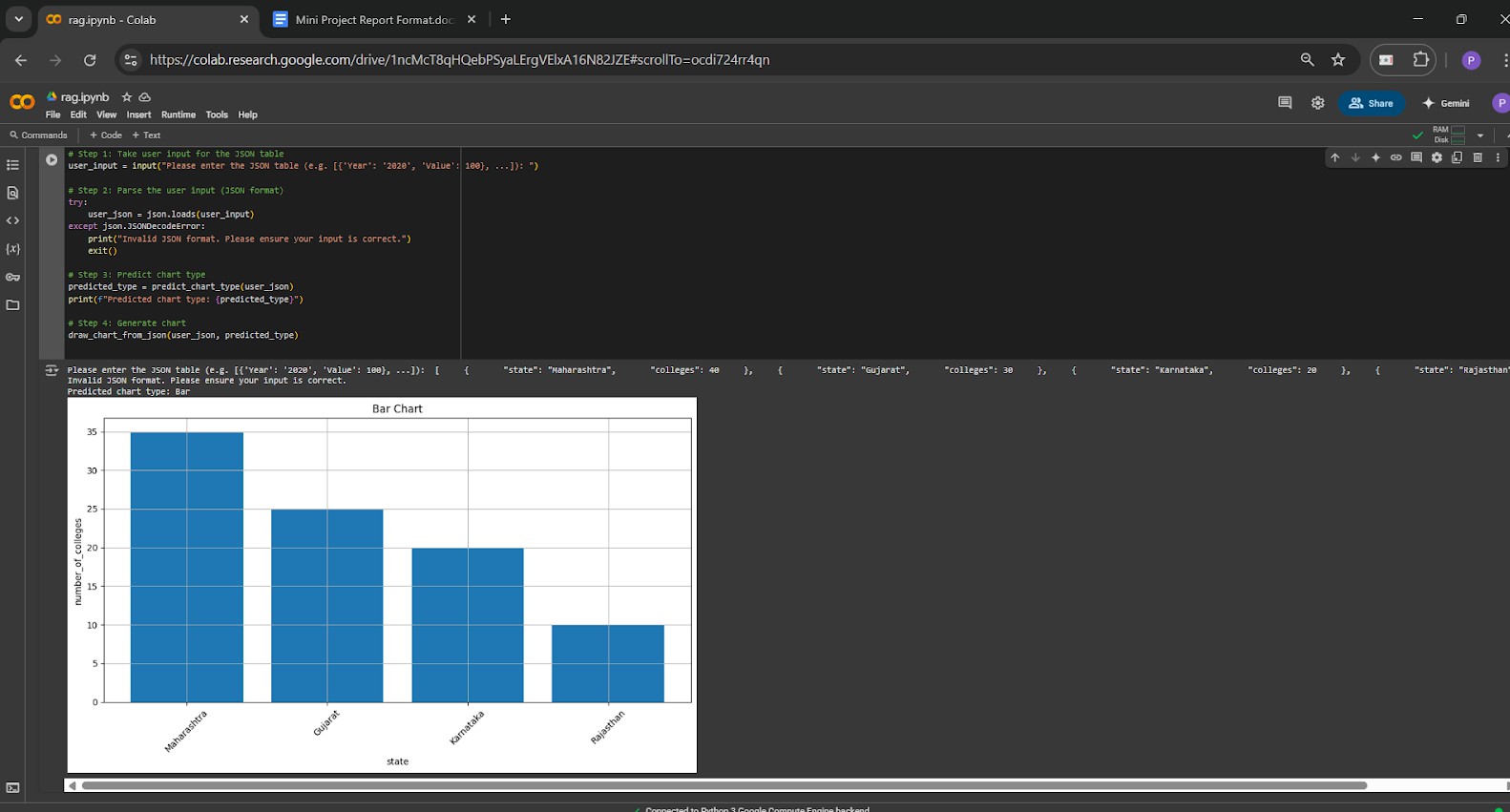
### Key findings:

The system efficiently translates textual descriptions of numerical data into structured JSON tables using the Llama-2-13B-chat.Q4\_K\_M model.

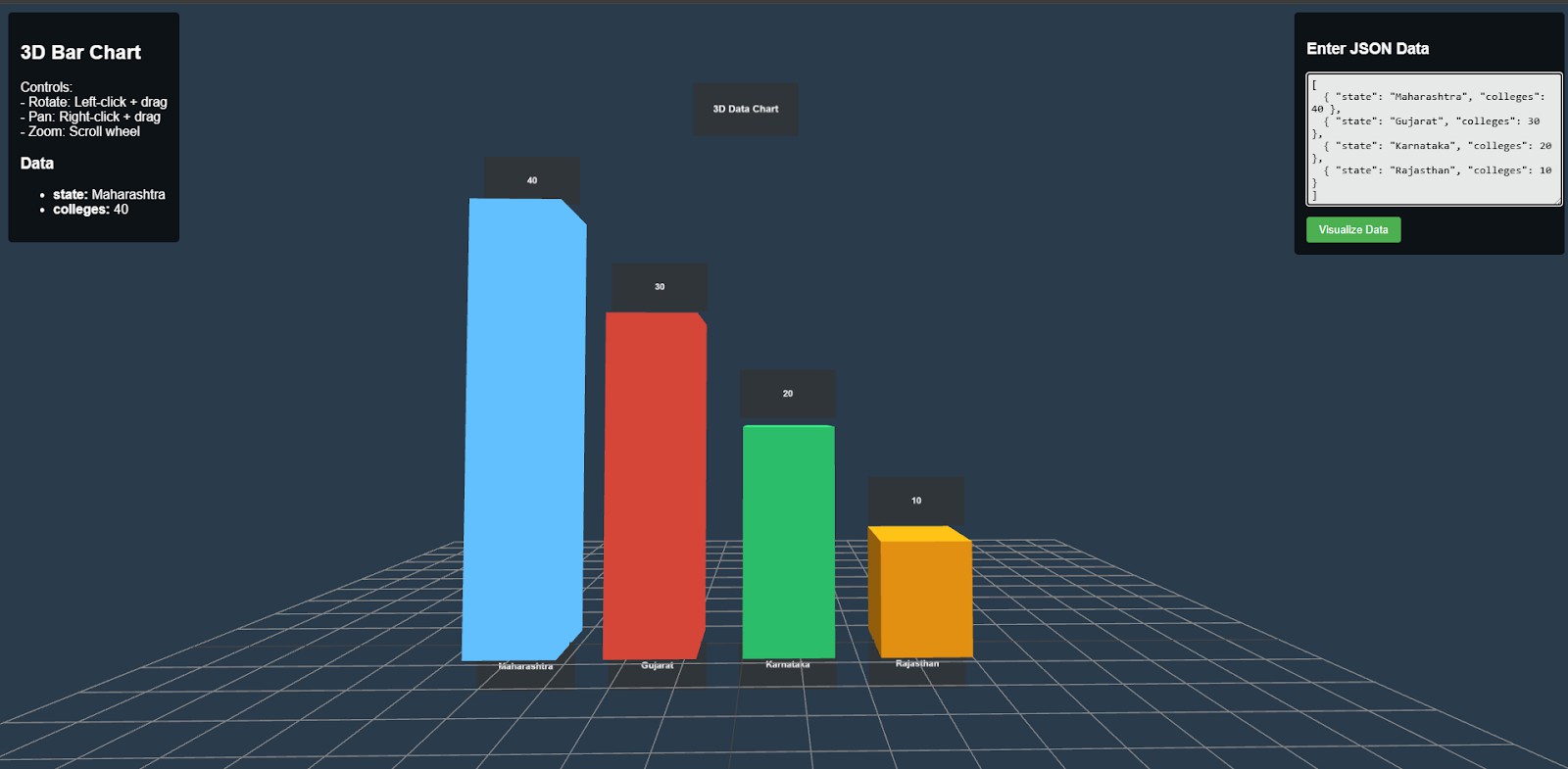
**Figure 5.1:** Converting text into structured tables using LLM (Llama-2-13B-chat.Q4\_K\_M)

A custom dataset of 50 records was created to train the RAG model, improving its ability to map structured tabular data to the most relevant visualization type.

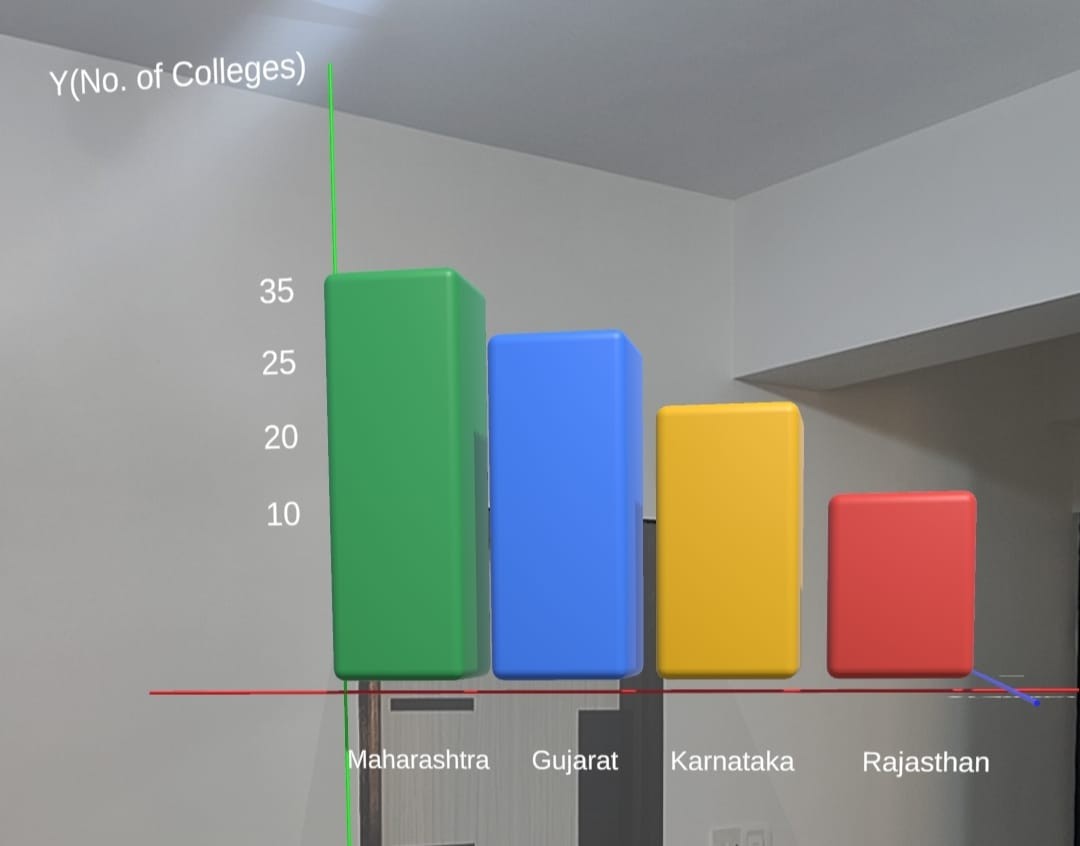
**Figure 5.2:** Custom dataset for RAG model training

The RAG model effectively predicts the most appropriate visualization type based on the nature and distribution of the tabular data.

**Figure 5.3:** Visualization type prediction by RAG model based on input JSON data

The system renders the selected visualizations as immersive 3D models in both Virtual Reality (VR) and Augmented Reality (AR). In VR, users can zoom, rotate, and pan to explore the visualization. In AR, the charts are displayed as static 3D objects within the camera view.

**Figure 5.4:** VR environment displaying a 3D visualization generated from JSON input

The AR visualization displays a 3D bar chart comparing the number of colleges across four Indian states. Each bar is color-coded and positioned in real-world space, enhancing clarity and engagement through spatial data representation.

**Figure 5.5:** AR environment displaying a static 3D visualization

### Performance Evaluation

* **Text-to-Table Conversion (LLM):** The Llama-2-13B-chat.Q4\_K\_M model achieved a **100% JSON validity rate**, ensuring that all outputs were syntactically correct. It demonstrated a **94% accuracy in field extraction**, correctly identifying the expected keys from the input text, and a **91% accuracy in value extraction**, accurately mapping the numerical values to their respective fields in the generated JSON.
* **Visualization Type Prediction (RAG Model):** The RAG model, trained on a custom dataset of 50 labeled samples, demonstrated progressive improvement across 15 epochs. It began with a baseline validation accuracy of 44.44% and gradually improved to a peak of 88.89%. The model consistently reduced its loss from 0.6931 to 0.6168, indicating effective learning. This training progression reflects the model's increasing ability to generalize from limited data, ultimately achieving **88.89% classification accuracy** on the validation set by the final epoch.
* **VR Visualization Rendering:** The system was deployed on Meta Quest 2, where the visualizations were displayed in a stable and responsive panel within the VR interface. The environment effectively demonstrated key functionalities such as zoom, pan, and rotation, with smooth performance and **interaction latency under 100 milliseconds**. The visualization panel loaded promptly, with an **average scene load time of 1.5 seconds**, providing a functional and user-friendly VR experience for data exploration.
* **AR Visualization Rendering:** In the AR mode, the system displayed floating 3D visualizations within the camera view. This setup provided a lightweight and platform-independent approach to visualizing data in augmented environments.

# Chapter 6 Conclusion and Future Work

This chapter concludes the report by summarizing the key findings and outcomes of the project. It highlights how the system has successfully implemented a unified system capable of transforming natural language descriptions of numerical data into both VR and AR-based 3D visualizations. The system integrates a frontend user interface, a Large Language Model (LLM) for parsing text to structured JSON tables, a Retrieval-Augmented Generation (RAG) model for chart type selection, and rendering tools like Unity, Three.js, and AR Foundation for immersive visual output. By supporting both VR and AR modes, the solution enables users to intuitively explore their data in simulated and real-world environments. The use of color-coded axes, interactive 3D elements, and smart chart selection has made complex data more accessible and engaging. This project not only demonstrates the feasibility of bridging textual input with immersive data representation but also sets a foundation for more adaptive and intelligent visualization systems.

## Conclusions

This project successfully demonstrated the feasibility of generating both VR and AR visualizations from textual descriptions of numerical data. By integrating Large Language Models (LLMs), a Retrieval-Augmented Generation (RAG) model, and immersive technologies like Unity, Three.js, and AR Foundation, the system provides a seamless pipeline from natural language input to interactive 3D chart rendering.

Key achievements include:

* + - Efficient parsing of textual descriptions into structured tabular formats using LLMs.
    - Accurate selection of appropriate visualization types through a custom-trained RAG model.
    - Enhanced data understanding via immersive 3D experiences in both virtual and augmented environments.
    - Versatile application potential across domains such as education, agriculture, commerce, and environmental analytics.

While the system meets its core objectives, limitations such as limited dataset diversity, platform dependency, and scope for broader interaction design point to opportunities for future refinement and extension.

## Scope for Future Work

Future enhancements to the project could include:

* + - **Multimodal Input Support**: Enabling support for voice input or scanned handwritten descriptions to broaden accessibility.
    - **Dynamic Dataset Expansion**: Training the RAG model on a larger and more diverse dataset across multiple domains to improve generalization and accuracy.
    - **Live Data Integration**: Allowing the system to connect with APIs and databases for real-time chart updates based on incoming data streams.
    - **Advanced AR/VR Interactions**: Introducing gesture-based manipulation, voice commands, or gaze tracking to create a more natural and immersive user experience.
    - **Cross-Device Compatibility**: Extending visualization support across different platforms such as iOS, desktop browsers (via WebXR), and VR headsets.
    - **Model Optimization**: Reducing latency in visualization rendering and improving the efficiency of LLM and RAG model inference time for real-time performance.
    - **Multilingual Processing**: Incorporating multilingual understanding to process user inputs in regional or global languages, enhancing accessibility.
    - **User Customization Features**: Allowing users to choose color schemes, axis scales, or data filters to personalize the visualization output.

These improvements would strengthen the system’s practical applications and usability in real-world scenarios.

# References

1. S. Coyne and Y. Dong, "Tabularis Revilio: Converting Text to Tables," ACM International Conference on Information and Knowledge Management (CIKM 2024), 2024.
2. M. Singh, S. Gulwani, V. Le, and G. Verbruggen, "gTBLS: Generating Tables from Text by Conditional Question Answering," arXiv preprint arXiv:2403.14457, 2024.
3. A. Sundar, C. Richardson, and L. Heck, "Large Language Models as Generalizable Text-to-Table Systems," Proceedings of the Association for Computational Linguistics (ACL 2024), 2024.
4. Y. Dong, M. Oyamada, C. Xiao, and H. Zhang, "On the Use of Large Language Models for Table Tasks," VLDB/NeurIPS, 2024.
5. X. Wu, J. Zhang, and H. Li, "Text-to-Table: A New Way of Information Extraction," ACL, 2022.
6. Z. Guo, J. Zhou, J. Qi, M. Yan, Z. He, X. Wang, and C. Zhou, "Towards Controlled Table-to-Text Generation with Scientific Reasoning," IEEE/Scientific NLP Conference, 2023.
7. A. H. Huang, H. Wang, and Y. Yang, "FinBERT: A Large Language Model for Extracting Information from Financial Text," Contemporary Accounting Research, vol. 40, no. 2, pp. 806–841, Summer 2023, doi: 10.1111/1911-3846.12832.
8. P. Kumar, R. Kumar, and A. Gupta, "The Performance of BERT as Data Representation of Text Clustering," Journal of Big Data, 2022.
9. T. A. Sharma, R. P. Singh, and K. Gupta, "A Survey of Text Representation and Embedding Techniques in NLP," IEEE Access, 2023.
10. R. Kumar and A. Gupta, "Graph-Based Text Representation and Matching: A Review of the State of the Art and Future Challenges," IEEE Transactions on Knowledge and Data Engineering, 2020.
11. A. Sharma, R. Patel, and R. Singh, "From Text to Knowledge with Graphs: Modelling, Querying and Exploiting Textual Content," arXiv preprint, 2023.
12. D. Patel and M. Kumar, "A Novel Multidimensional Reference Model for Heterogeneous Textual Datasets Using Context, Semantic and Syntactic Clues," arXiv preprint, 2023.

# Acknowledgements

We, the members of the project team, Chandana Galgali, Prachi Gandhi, Mahek Thakkar and Harsh Singwi, would like to express our sincere gratitude to all those who contributed to the successful completion of this mini project.

Firstly, we would like to extend our deepest appreciation to our Project Guide, Prof. Avani Sakhapara, for her unwavering guidance, continuous support, and valuable feedback throughout the project. Her expertise and insightful suggestions were instrumental in shaping our approach and ensuring the project’s success.

We are also grateful to the IT Department and the ETRX Department at KJSSE for providing the necessary resources, facilities, and an enabling environment that made the completion of this project possible.

Finally, we would like to express our heartfelt thanks to our families and friends for their constant support, encouragement, and understanding. Their moral support and patience were crucial in helping us navigate the challenges of this project.