✅ Project Title:

“Extracting Insights From University Data Using AI Powered Visualization”

▯ Theoretical Background:

Universities create a large amount of data like student marks, attendance, faculty workload, and placement records. To understand this data, we need **visualizations** (charts, dashboards). But current tools like **Power BI and Tableau** need technical skills, are costly, and are not made for education data.

AUTOVI is a research system that shows how **AI can make visualization simple and automatic**. Its background is built on these main ideas:

1. **Data Cleaning (Preprocessing):**
   Raw data usually has missing or wrong values. AUTOVI cleans and prepares the data first so that the results are correct.
2. **Automatic Visualization:**
   Instead of asking the user to choose charts, AUTOVI uses AI to **recommend the best chart automatically** based on the data.
3. **AI for Insights:**
   AUTOVI uses a simple and clear AI model (MLP classifier). This helps in **finding patterns and insights** in data without working like a “black box.”
4. **User Feedback:**
   Users can give feedback, and the system learns from it. This makes the tool more **interactive and adaptive** over time.
5. **Easy for Everyone:**
   Because most of the work is automatic, even non-technical users like teachers or administrators can explore data without coding or BI tools.

▯ What You Will Do

- **Collect University Data**
  - Gather data such as student marks, attendance, faculty workload, placements.
- **Automatic Preprocessing**
  - Clean data (handle missing values, errors, and formats) automatically.
- **AI Model for Insights**
  - Use machine learning (MLP/Decision Tree) to detect patterns, trends, and outliers.
- **Generate Visualizations Automatically**
  - Convert insights into charts and dashboards without Power BI or Tableau.

- **Interactive User Feedback**
  - Allow users (teachers, staff, admin) to give feedback (like/dislike visuals).
  - System improves recommendations over time.
- **Build a Simple Interface**
  - A web-based dashboard where users upload data and instantly build result

---

□ Tech Stack (Example, not using Power BI/Tableau)

- **Database:** MySQL / PostgreSQL
- **Preprocessing:** Python (Pandas, NumPy, Scikit-learn)
- **AI Models:** MLP, Decision Tree, Random Forest
- **Visualization:** Plotly, Matplotlib, Seaborn, D3.js
- **Backend:** Flask / Django
- **Frontend:** React.js / Chart.js / HTML-CSS-JS
- **Deployment:** AWS / Heroku / Local Server

□ Expected Output

1. **Cleaned & Processed Data**
   - University data (students, attendance, faculty, placements) automatically cleaned and ready for analysis.
2. **Automatic Visualizations**
   - System generates charts and dashboards (bar, line, pie, scatter, box plots, etc.) without using Power BI or Tableau.
3. **AI-Extracted Insights**
   - Clear insights such as:
     - "Attendance dropped by 15% in Semester 4."
     - "Female students performed better in Subject X."
     - "Placement rate increased after 2022."
4. **Interactive Dashboard**
   - Web interface to upload data and view insights instantly.
5. **User Feedback Integration**
   - Users can like/dislike visuals → system improves future recommendations.
6. **Decision Support for Universities**
   - Helps administrators, faculty, and staff make data-driven decisions quickly and easily.

□ Novelty / What is New

Feature	Existing Papers	Your Project
Dataset / Domain	General tabular/business data (AdaVis, AUTOVI, DashBot, MultiVision)	University data (students, attendance, faculty workload, placements)
Preprocessing	Limited or manual (DashBot, MultiVision, CoInsight)	**Automatic preprocessing** of raw university data (missing values, noise, encoding)
Visualization	Recommended charts/dashboards using ML/DL (AdaVis, MultiVision, DashBot)	**Automatic chart generation** directly from university data
Insight Extraction	Basic patterns (AUTOVI, Waltzboard)	**Detailed insights**: trends, comparisons, outliers, student/department performance
User Feedback	Limited or absent (DashBot, Waltzboard)	**Interactive feedback loop** (system improves with likes/dislikes)
Explainability	Some works provide local/global explanations (AdaVis)	**Clear insight explanations** + justification for chosen visuals
Domain-Specific Focus	Mostly generic or business-oriented dashboards	**Focused on higher education analytics**
Outputs	Visuals only (CoInsight storytelling, MultiVision dashboards)	Visuals **+ AI-driven insights + feedback adaptation**
Ease of Use	Medium (require some technical knowledge or iteration)	**High** (any staff can upload data and get results instantly)

□ Example Flow

Nice — here's a **clear, step-by-step, detailed explanation** of how **AUTOVI** works (so you can use it in your report). I keep language simple and add exact references to the paper where useful.

How AUTOVI works — detailed walkthrough

High-level idea (one line)

AUTOVI is a web platform that **automatically cleans tabular data, scores which visuals fit each column, generates visualizations, explains them, and learns from user feedback**.

---

1) Data & dataset preparation

- The authors collected **many real datasets** (25 Kaggle datasets with varied sizes/types) to cover real-world cases. An automated process extracts dataset **metadata** (column names, types, counts, null rates, distributions).
- For training the recommender they **added a "visualization label" column** and populated it (manually, from expert guidance) so the model had examples of which chart is appropriate for which column. This mimics how humans label charts.

---

2) Automatic preprocessing (what it does)

When a user uploads a file, AUTOVI runs an automatic cleaning pipeline that includes:

- **Data type detection** (numeric / categorical / temporal).
- **Null/missing value handling** (reporting counts and applying imputation or exclusion as needed).
- **Standardization / scaling** and **encoding** (one-hot for categoricals, numeric scaling) so features are ready for ML.
- **Outlier / noise detection** and basic corrections so visuals aren't misled by bad rows.
- The system also builds a **profile report** (column distributions, correlation matrix) so users see dataset health and characteristics.

In short: AUTOVI fully prepares the table automatically so downstream models and visuals get clean inputs.

---

3) Feature extraction / metadata used by the model

AUTOVI computes per-column features such as: data type, cardinality, null count, min/max/mean, distribution shape, unique values, etc. These become the input features for the visualization classifier.

---

4) The visualization classifier (the ML core)

- AUTOVI uses a **transparent Multi-Layer Perceptron (MLP)** as its core classifier to predict which visualization suits each column.
- **Architecture (paper):** input layer (column features) → **two hidden layers** with ReLU → output layer with **sigmoid** to give probabilities for visualization types.
- **Training details** reported: binary cross-entropy loss, Adam optimizer, batch size 32, feature scaling and one-hot encoding used in preprocessing. The MLP classifies into types like **histogram, binned histogram, bar chart, or none** (non-informative).
- The paper reports ~**90% accuracy** on their prediction task (on their prepared datasets).

---

5) Mapping model output → actual visualizations + explanations

- The MLP outputs probabilities for visualization types per column. The system maps the top prediction(s) to **visualization templates** (Plot: histogram, bar, line, box, etc.) and creates the chart.
- AUTOVI also **generates explanations**: it shows why a chart was chosen (e.g., "column has numeric values and low cardinality → bar recommended") to build user trust. The explanation module is part of the Identify Visualization module.

---

6) Interactive UI modules (user-facing features)

AUTOVI's platform is modular and user-friendly; main UI modules include:

- **Upload dataset** (auto triggers cleaning).
- **Features Module** (shows column types, distributions).
- **Identify Visualization Module** (automatically recommends + explains).
- **Try Your Own Visualization**: users can pick a column + chart type to try; the system gives hints/feedback (if the selection is suboptimal it explains why). This produces active learning and better user understanding.
- **Profile Report Module**: a summary page (statistics, correlation heatmap) for quick dataset insight.

---

7) User feedback & learning loop

- AUTOVI captures user interactions (likes/dislikes, corrections from "Try Your Own") and uses that feedback to **improve the system** (retrain models, refine recommendations). The paper highlights dynamic feedback mechanisms and that the platform is meant to be iterative and learn from users.

---

8) Model training & offline improvements

- The **Model Training Module** supports retraining (using the labeled examples + user feedback/provenance). The system is designed so model updates can be applied as more labeled data or feedback accumulates.
- Note: initial supervised training needed the manual "visualization label" column (authors created these labels during dataset preparation). After deployment, feedback can reduce the need for manual labeling over time.

---

9) Evaluation & user study (what the authors measured)

AUTOVI was tested vs manual coding; results show strong usability and efficiency:

- **Time:** AUTOVI average time ≈ **3.7 minutes** vs traditional coding ≈ **43.25 minutes**.
- **Usability (SUS):** AUTOVI scored **~82.66** (excellent usability).

- **Engagement** and **workload (TLX)** results also favored AUTOVI strongly; the paper reports statistically significant differences.

---

10) Limitations & future work (from the paper)

- AUTOVI authors note limits: adapting to very diverse datasets, adding more visualization choices per column, and supporting real-time visualizations are areas for future improvement. They emphasise continuing user testing and domain expert collaboration.

Short sequence view (step-by-step flow)

1. Upload CSV/Excel → 2. Auto preprocess (type detection, missing values, encoding) → 3. Extract per-column features → 4. MLP predicts best viz type → 5. System builds chart + explanation → 6. User views/tries/feedbacks → 7. Feedback logged → 8. Retrain / refine model offline → repeat.

Quick notes for your project (how to use AUTOVI ideas)

- Reuse AUTOVI's **automatic preprocessing** → **feature extraction** → **MLP classifier** → **mapping to viz templates** → **feedback loop** design for your university data.
- For university domain, extend features with **domain signals** (semester, course type, grade scale) and include specialized visuals (boxplots for grade distributions, heatmaps for attendance). The AUTOVI pipeline makes adding domain features straightforward.

🔄 What Is New in Your Project (Compared to Published Papers):

Paper	Input Modalities	Model Type	Dataset	Accuracy / Eval	Key Contribution
AdaVis (Zhang et al., 2024)	Tabular data (columns & metadata)	Graph Neural Network (GNN) + Explainability Module	OpenML datasets	~83–92% accuracy (on visualization prediction)	Adaptive & explainable visualization recommendation using GNN
AUTOVI (2024)	Tabular datasets (CSV/Excel)	Multi-Layer Perceptron (MLP) classifier + preprocessing pipeline	25 Kaggle datasets	~90% accuracy	Automatic preprocessing + visualization recommendation + user feedback loop
DashBot (2023)	Tabular datasets (structured data)	Deep Reinforcement Learning (A3C Agent)	Synthetic & real datasets	Policy reward scores (not accuracy)	Insight-driven dashboard generation using RL
MultiVision (2022)	Tabular data (query + visualization pairs)	Siamese Neural Network	VizNet dataset	Precision/Recall ~78–85%	Analytical dashboard design using similarity learning
CoInsight (2024)	Hierarchical tables	Graph-based Insight Graph Construction	Real-world hierarchical tables	User study (positive storytelling outcomes)	Converts tables into **visual stories with connected insights**
Waltzboard (2022)	Multi-criteria dashboard inputs	Rule-based + Optimization model	Case studies (Exploratory analysis)	User evaluation	Multi-criteria **automated dashboard design**
VIS+AI (2023)	Tabular datasets	AI-guided Visualization Framework	Multiple (case studies)	Qualitative evaluation	Framework for integrating AI with visualization for **explainability**

Paper	Input Modalities	Model Type	Dataset	Accuracy / Eval	Key Contribution
AdaVis (Zhang et al., 2024)	Tabular data (columns & metadata)	Graph Neural Network (GNN) + Explainability Module	OpenML datasets	~83–92% accuracy (on visualization prediction)	Adaptive & explainable visualization recommendation using GNN
AUTOVI (2024)	Tabular datasets (CSV/Excel)	Multi-Layer Perceptron (MLP) classifier + preprocessing pipeline	25 Kaggle datasets	~90% accuracy	Automatic preprocessing + visualization recommendation + user feedback loop
DashBot (2023)	Tabular datasets (structured data)	Deep Reinforcement Learning (A3C Agent)	Synthetic & real datasets	Policy reward scores (not accuracy)	Insight-driven dashboard generation using RL
MultiVision (2022)	Tabular data (query + visualization pairs)	Siamese Neural Network	VizNet dataset	Precision/Recall ~78–85%	Analytical dashboard design using similarity learning
CoInsight (2024)	Hierarchical tables	Graph-based Insight Graph Construction	Real-world hierarchical tables	User study (positive storytelling outcomes)	Converts tables into **visual stories with connected insights**
Waltzboard (2022)	Multi-criteria dashboard inputs	Rule-based + Optimization model	Case studies (Exploratory analysis)	User evaluation	Multi-criteria **automated dashboard design**
VIS+AI (2023)	Tabular datasets	AI-guided Visualization Framework	Multiple (case studies)	Qualitative evaluation	Framework for integrating AI with visualization for **explainability**

? What You Will Improve

1. **Domain Focus** → AUTOVI is general; yours is **specific to university data**.
2. **Richer Insights** → Not just charts, but **trends, comparisons, outliers**.

3. **Better Explainability** → Explain not only *why this chart*, but also *why the insight matters*.
4. **Stronger Feedback Loop** → System **learns continuously** from user feedback.
5. **More Visual Types** → Add **heatmaps, boxplots, correlation plots** for education data.
6. **Automatic Reports** → Generate **summary reports + visuals** for decision making.