

JobLens MIT

Introduction

The recruitment process plays a pivotal role in shaping a student's career yet accessing structured and reliable placement information remains a persistent challenge. As students of Manipal Institute of Technology, Bengaluru, we recognize the difficulties in navigating the placement landscape, especially in a relatively new campus where alumni networks and legacy-driven insights are still in their early stages. This often leaves students overwhelmed, resorting to scattered sources for information on company visits, job roles, eligibility criteria, and required skill sets.

To address this issue, we are developing JobLens MIT, that will aggregate and organize placement-related data from publicly available sources, providing a centralized and structured approach to placement research. For now, the project will be limited to MIT campuses across India (specifically the engineering departments), offering insights into common visiting companies, hiring trends, and required qualifications.

While we may not have access to exclusive college-provided placement data, we aim to compile the most relevant and valuable information from credible sources across the web. Additionally, we will incorporate any available institutional insights to enhance the tool's effectiveness. By streamlining this information in one accessible platform, we aspire to bridge the information gap, enabling students to make well-informed career decisions with ease.

Problem Statement

Students at MIT struggle with fragmented and unstructured information regarding company visits, job roles, eligibility criteria, and hiring trends. This lack of centralized data makes it difficult for them to make informed decisions about their careers. Furthermore, manual tracking of placement data is inefficient and errorprone, highlighting the need for an automated solution.

Objectives

- Automate the extraction of placement and internship-related data from official and relevant sources.
- Organize and store data in a structured SQL database for easy access and querying.
- Implement Machine Learning (ML) models to predict student eligibility based on GPA, skills, projects, and resumes.
- Develop an intuitive web interface to allow students to search and filter placement data interactively.
- Enhance decision-making by providing personalized career insights and recommendations.

Methodology

Our project will be developed using a full-stack web application approach, integrating the following components:

- Frontend: React.js (HTML, CSS, JavaScript, Typescript) for an interactive and user-friendly interface.
- Backend: Node.js to manage API requests and handle data processing.
- Database: SQL (SQLite) to store and retrieve structured placement data.
- Web Scraping: Chatgpt and Perplexity.ai to extract placement data from sources.
- Machine Learning: Typescript, for predictive analytics, assessing student eligibility based on past hiring patterns.

Relation Schema & Database Structure

- To ensure efficient data management, we define the following relation schemas:
- Students Table: Students(StudentID, Name, GPA, Skills, Resume, Branch)
- Companies Table: Companies(CompanyID, Name, HiringRoles, EligibilityCriteria, LastVisitYear)

- Job Roles Table: JobRoles(JobID, CompanyID, RoleName, RequiredSkills, SalaryPackage)

- These schemas ensure structured and scalable data storage, enabling seamless querying and analytics.

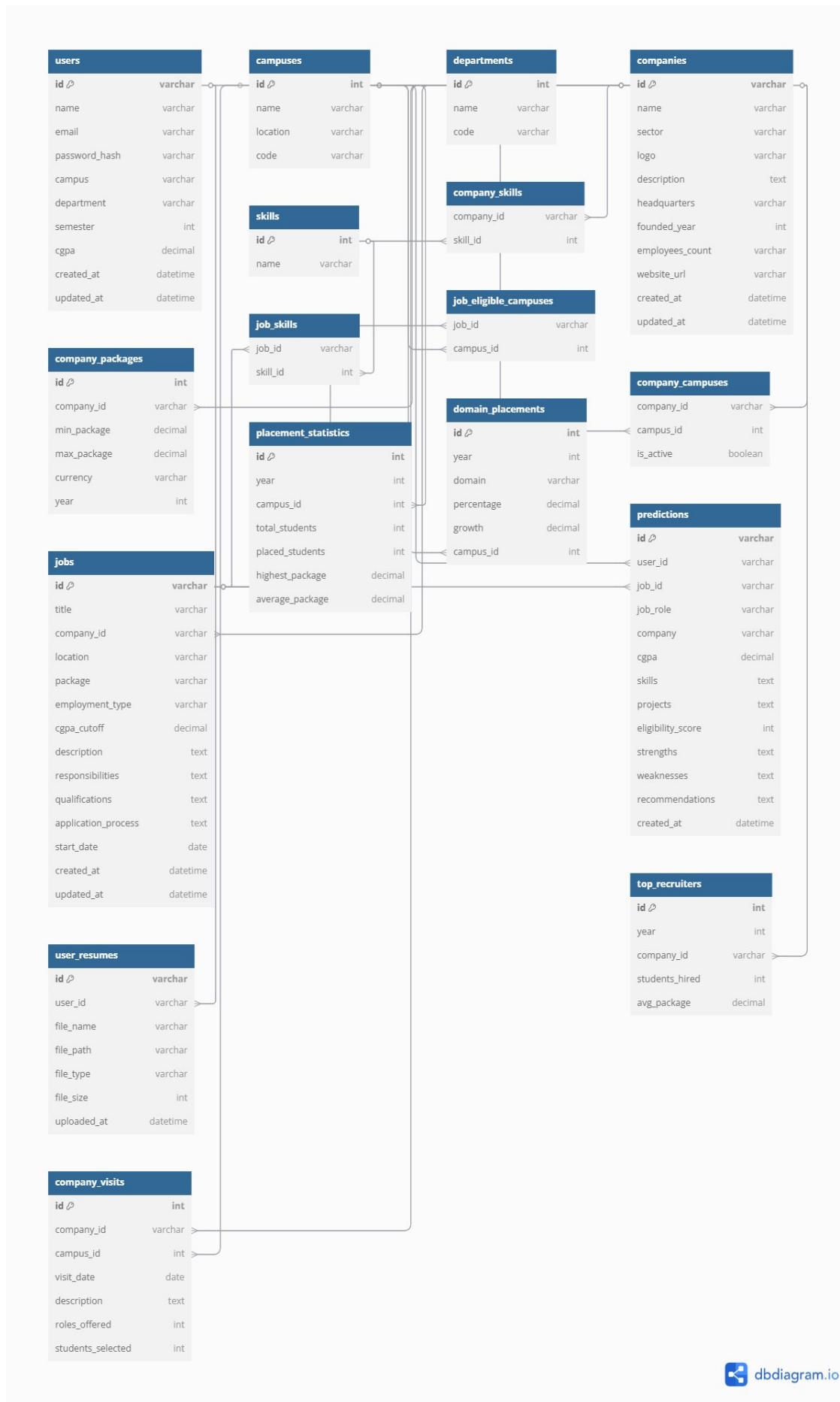
Expected Outcomes

- A centralized database containing accurate and real-time placement-related data.
- An eligibility predictor that helps students assess their chances for specific roles.
- A user-friendly web platform that provides structured placement insights.
- Improved decision-making for students regarding applications and career paths.
- A scalable and modular system that can be expanded for future placement-related analytics.

Conclusion

This project aims to simplify the placement process for MIT students by providing a data-driven platform for accessing placement insights. By integrating web scraping, database management, and ML-based predictions, we seek to enhance career decision-making and improve placement readiness for students at MIT. The system will serve as a scalable and efficient solution that can evolve with the changing recruitment landscape.

ER DIAGRAM:



SQL COMMANDS:

(Users table to store user accounts)

```
CREATE TABLE users (
    id VARCHAR(36) PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) NOT NULL UNIQUE,
    password_hash VARCHAR(255) NOT NULL, -- Store hashed passwords, not plaintext
    campus VARCHAR(50) NOT NULL,
    department VARCHAR(100) NOT NULL,
    semester INT DEFAULT 1,
    cgpa DECIMAL(3,2) DEFAULT 0.00,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
);
```

(Campuses table to store MIT campus information)

```
CREATE TABLE campuses (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL UNIQUE,
    location VARCHAR(100) NOT NULL,
    code VARCHAR(20) NOT NULL UNIQUE
);
```

(Departments table)

```
CREATE TABLE departments (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL UNIQUE,
    code VARCHAR(20) NOT NULL UNIQUE
);
```

(Companies table to store company information)

```
CREATE TABLE companies (
    id VARCHAR(36) PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    sector VARCHAR(50) NOT NULL,
    logo VARCHAR(50),
    description TEXT,
    headquarters VARCHAR(100),
    founded_year INT,
    employees_count VARCHAR(50),
    website_url VARCHAR(255),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
);
```

(Company details - package ranges offered)

```
CREATE TABLE company_packages (
    id INT AUTO_INCREMENT PRIMARY KEY,
    company_id VARCHAR(36) NOT NULL,
    min_package DECIMAL(10,2) NOT NULL,
```

```
max_package DECIMAL(10,2) NOT NULL,  
currency VARCHAR(10) DEFAULT '₹',  
year INT NOT NULL,  
FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE  
);
```

(Skills table)

```
CREATE TABLE skills (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100) NOT NULL UNIQUE  
);
```

(Company required skills)

```
CREATE TABLE company_skills (  
    company_id VARCHAR(36) NOT NULL,  
    skill_id INT NOT NULL,  
    PRIMARY KEY (company_id, skill_id),  
    FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE,  
    FOREIGN KEY (skill_id) REFERENCES skills(id) ON DELETE CASCADE  
);
```

(Company campus presence)

```
CREATE TABLE company_campuses (  
    company_id VARCHAR(36) NOT NULL,  
    campus_id INT NOT NULL,  
    is_active BOOLEAN DEFAULT TRUE,  
    PRIMARY KEY (company_id, campus_id),  
    FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE,  
    FOREIGN KEY (campus_id) REFERENCES campuses(id) ON DELETE CASCADE  
);
```

(Jobs table to store job listings)

```
CREATE TABLE jobs (  
    id VARCHAR(36) PRIMARY KEY,  
    title VARCHAR(100) NOT NULL,  
    company_id VARCHAR(36) NOT NULL,  
    location VARCHAR(100) NOT NULL,  
    package VARCHAR(50) NOT NULL,  
    employment_type ENUM('Full-time', 'Internship', 'Part-time') NOT NULL,  
    cgpa_cutoff DECIMAL(3,2) NOT NULL,  
    description TEXT,  
    responsibilities TEXT,  
    qualifications TEXT,  
    application_process TEXT,  
    start_date DATE,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
    FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE);
```

(Job required skills)

```
CREATE TABLE job_skills (  
    job_id VARCHAR(36) NOT NULL,  
    skill_id INT NOT NULL,
```

```
PRIMARY KEY (job_id, skill_id),
FOREIGN KEY (job_id) REFERENCES jobs(id) ON DELETE CASCADE,
FOREIGN KEY (skill_id) REFERENCES skills(id) ON DELETE CASCADE
);
```

(Job eligible campuses)

```
CREATE TABLE job_eligible_campuses (
    job_id VARCHAR(36) NOT NULL,
    campus_id INT NOT NULL,
    PRIMARY KEY (job_id, campus_id),
    FOREIGN KEY (job_id) REFERENCES jobs(id) ON DELETE CASCADE,
    FOREIGN KEY (campus_id) REFERENCES campuses(id) ON DELETE CASCADE
);
```

(store user eligibility predictions)

```
CREATE TABLE predictions (
    id VARCHAR(36) PRIMARY KEY,
    user_id VARCHAR(36) NOT NULL,
    job_id VARCHAR(36),
    job_role VARCHAR(100) NOT NULL,
    company VARCHAR(100) NOT NULL,
    cgpa DECIMAL(3,2) NOT NULL,
    skills TEXT NOT NULL,
    projects TEXT,
    eligibility_score INT NOT NULL,
    strengths TEXT,
    weaknesses TEXT,
    recommendations TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
    FOREIGN KEY (job_id) REFERENCES jobs(id) ON DELETE SET NULL
);
```

(User resume storage)

```
CREATE TABLE user_resumes (
    id VARCHAR(36) PRIMARY KEY,
    user_id VARCHAR(36) NOT NULL,
    file_name VARCHAR(255) NOT NULL,
    file_path VARCHAR(255) NOT NULL,
    file_type VARCHAR(50) NOT NULL,
    file_size INT NOT NULL,
    uploaded_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);
```

(Placement statistics for insights)

```
CREATE TABLE placement_statistics (
    id INT AUTO_INCREMENT PRIMARY KEY,
    year INT NOT NULL,
    campus_id INT NOT NULL,
    total_students INT NOT NULL,
    placed_students INT NOT NULL,
    highest_package DECIMAL(10,2) NOT NULL,
    average_package DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (campus_id) REFERENCES campuses(id) ON DELETE CASCADE,
```

```
    UNIQUE KEY (year, campus_id)
```

```
);
```

(Domain-wise placement statistics)

```
CREATE TABLE domain_placements (
    id INT AUTO_INCREMENT PRIMARY KEY,
    year INT NOT NULL,
    domain VARCHAR(100) NOT NULL,
    percentage DECIMAL(5,2) NOT NULL,
    growth DECIMAL(5,2),
    campus_id INT,
    FOREIGN KEY (campus_id) REFERENCES campuses(id) ON DELETE SET NULL
);
```

(Top recruiters statistics)

```
CREATE TABLE top_recruiters (
    id INT AUTO_INCREMENT PRIMARY KEY,
    year INT NOT NULL,
    company_id VARCHAR(36) NOT NULL,
    students_hired INT NOT NULL,
    avg_package DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE
);
```

(Company visits/recruitment timeline)

```
CREATE TABLE company_visits (
    id INT AUTO_INCREMENT PRIMARY KEY,
    company_id VARCHAR(36) NOT NULL,
    campus_id INT NOT NULL,
    visit_date DATE NOT NULL,
    description TEXT,
    roles_offered INT,
    students_selected INT,
    FOREIGN KEY (company_id) REFERENCES companies(id) ON DELETE CASCADE,
    FOREIGN KEY (campus_id) REFERENCES campuses(id) ON DELETE CASCADE
);
```

UI:

Your Campus Placement Companion

Get data-driven insights on placements across all MIT campuses. Find companies, roles, and predict your eligibility with our ML-powered platform.

[Get Started →](#)[Login](#)

Everything you need for placement success

JobLens MIT centralizes placement data across campuses and provides personalized insights to help you prepare better.



Everything you need for placement success

JobLens MIT centralizes placement data across campuses and provides personalized insights to help you prepare better.



Curated Data

Access consolidated placement data from all MIT campuses in one place.



Company Database

Explore detailed information about companies visiting MIT campuses.



Placement Insights

Visualize trends, patterns, and statistics across roles and campuses.



ML Predictions

Get personalized eligibility predictions based on your profile and skills.

Covering All MIT Campuses

Our platform aggregates placement data from all Manipal Institute of Technology campuses across India.

[MIT Manipal](#)[MIT Bengaluru](#)[Manipal University Jaipur](#)[SMTI](#)

Covering All MIT Campuses

Our platform aggregates placement data from all Manipal Institute of Technology campuses across India.

MIT Manipal

8,500+ Students
350+ Companies

[View details →](#)

MIT Bengaluru

3,200+ Students
120+ Companies

[View details →](#)

Manipal University Jaipur

2,800+ Students
95+ Companies

[View details →](#)

SMIT

2,000+ Students
80+ Companies

[View details →](#)

Top Companies at MIT Campuses

Explore opportunities from leading companies that recruit from MIT.

[View All Companies](#)

Microsoft

4 open roles
[Packages from ₹25L+](#)

Google

3 open roles
[Packages from ₹30L+](#)

Amazon

5 open roles
[Packages from ₹22L+](#)

Infosys

7 open roles
[Packages from ₹7L+](#)

Top Companies at MIT Campuses

Explore opportunities from leading companies that recruit from MIT.

[View All Companies](#)

Microsoft

4 open roles
[Packages from ₹25L+](#)

Google

3 open roles
[Packages from ₹30L+](#)

Amazon

5 open roles
[Packages from ₹22L+](#)

Infosys

7 open roles
[Packages from ₹7L+](#)

TCS

8 open roles
[Packages from ₹7L+](#)

Accenture

6 open roles
[Packages from ₹8L+](#)

Wipro

4 open roles
[Packages from ₹6L+](#)

IBM

3 open roles
[Packages from ₹12L+](#)

Ready to boost your placement chances?

Join JobLens MIT today and get access to exclusive placement data, insights, and personalized predictions.

[Sign Up Now](#)

TCS
8 open roles
[Packages from ₹7L+](#)

ACC
6 open roles
[Packages from ₹8L+](#)

WIP
4 open roles
[Packages from ₹6L+](#)

IBM
3 open roles
[Packages from ₹12L+](#)

Ready to boost your placement chances?

Join JobLens MIT today and get access to exclusive placement data, insights, and personalized predictions.

[Sign Up Now](#)

Centralizing placement data for MIT campus students with predictive insights.



RESOURCES

[Companies](#)
[Job Roles](#)
[Placement Insights](#)
[Eligibility Predictor](#)

MIT CAMPUSES

[MIT Manipal](#)
[MIT Bengaluru](#)
[Manipal University Jaipur](#)
[SMMIT](#)

LEGAL

[Privacy Policy](#)
[Terms of Service](#)
[Data Usage Policy](#)

© 2025 JobLens MIT. All rights reserved.

DISCUSSIONS:

Log in

Enter your email and password to access your account

Email

hahahoho@gmail.com

Password

[Forgot password?](#)

.....

[Log in](#)

[Quick Demo Login](#)

Don't have an account? [Sign up](#)

Welcome, hahahoho!

Here's what's happening with campus placements

 Companies
250+

 Job Roles
560+

 Insights
140+

 Eligibility Predictions
98%

Recent Companies

Latest companies visiting campus

Microsoft
4 roles - ₹25.5 LPA

April 3, 2025

Infosys
7 roles - ₹7.2 LPA

April 2, 2025

Amazon
3 roles - ₹22 LPA

Mar 29, 2025

TCS
5 roles - ₹7 LPA

Mar 27, 2025

Trending Job Roles

Popular roles across campuses

Software Development Engineer
Microsoft - ₹25 LPA

Python, Azure, Algorithms

Machine Learning Engineer
Google - ₹28 LPA

TensorFlow, Python, NLP

System Engineer
Infosys - ₹7.5 LPA

Java, SQL, Cloud

Data Analyst
Amazon - ₹15 LPA

SQL, Power BI, Python

Predict Your Eligibility

Use our ML model to check your chances for different roles

[Try Prediction Tool](#)

Predict Your Eligibility

Our ML-powered tool will analyze your profile and predict your eligibility for specific job roles

Eligibility Prediction Form

Fill in your details to get personalized eligibility predictions

Job Role
Company

Machine Learning Engineer

Oracle

CGPA

8.58

Skills (comma separated)

Machine learning, tensorflow, python,

Projects (briefly describe)

data sorting

Resume (optional)

PDF, DOC or DOCX (5MB max)

[Reset](#)
[Predict Eligibility](#)

Predict Your Eligibility

Our ML-powered tool will analyze your profile and predict your eligibility for specific job roles

Prediction Result

Eligibility prediction for Machine Learning Engineer at Oracle

Edit inputs

Eligibility Score

80%

Strong Match

✓ Strengths

- Your CGPA of 8.58 meets or exceeds the typical threshold for Oracle
- You have 4 relevant skills for this role: machine learning, tensorflow, python...

✗ Areas for Improvement

- No significant weaknesses identified

ⓘ Recommendations

- Your profile is strong! Focus on interview preparation.

[Start New Prediction](#)

[View Related Jobs >](#)

Job Roles

Explore job opportunities available at MIT campuses

Q Machine Learning Engineer All Campuses All Types ⌂ Reset filters

CGPA Cutoff: 0 - 8.58

G Machine Learning Engineer
Google - Bengaluru

Full-time | ₹35 LPA

Required Skills:
Python, TensorFlow, Machine Learning, NLP

CGPA Cutoff: 8

MIT Manipal

[Details >](#)

Machine Learning Engineer

Google - Bengaluru

Full-time

Job Description

Google is looking for a talented Machine Learning Engineer to join our team in Bengaluru. This is an exciting opportunity to work on challenging projects and grow your career in a supportive environment.

Key Information

📍 Location: Bengaluru

🏢 Company: Google

💻 Job Type: Full-time

✉️ Min CGPA: 8

⌚ Start Date: July 2025

₹ Package: ₹35 LPA

Required Skills

Python, TensorFlow, Machine Learning, NLP

Responsibilities

- Design and implement machine learning models
- Process, clean, and verify the integrity of data
- Conduct model training, evaluation, and deployment
- Research and implement appropriate ML algorithms
- Extend existing ML libraries and frameworks

Qualifications

- Bachelor's degree in Computer Science, Engineering or related field
- Minimum CGPA of 8