**A**

**Summer Internship Report On**

**"American Sign Language Gesture Recognition"**

(AIML306 – Summer Internship - I)

# Prepared by

Yagnik Paghdal(22AIML026)

# Under the Supervision of

Prof. Nishant Koshti

# Submitted to

## Charotar University of Science & Technology (CHARUSAT) for the Partial Fulfillment of the Requirements for the Degree of Bachelor of Technology (B.Tech.)

for Semester 5

### Submitted at



**Accredited with Grade A+ by NAAC Accredited with Grade A by KCG**



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING**

**Chandubhai S. Patel Institute of Technology (CSPIT) Faculty of Technology & Engineering (FTE), CHARUSAT At: Changa, Dist: Anand, Pin: 388421.**

**August, 2023**



22AIML059 Acknowledgment

# ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to all those who have contributed to my internship journey in the Machine Learning domain. This opportunity has been a tremendous learning experience, and I owe my sincere appreciation to the following individuals.

It brings me great pleasure to express my heartfelt appreciation to my excellent mentor, Prof. Dhiraj Shringi, for her unending encouragement and support, which provided me with the morale and self-assurance I needed to continue working on my research. I would like to express my heartfelt gratitude to them for their invaluable and competent supervision and help during the project's implementation.

I extend my gratitude to Mr. Ashish Mishra for your camaraderie, encouragement, and collaborative spirit have made this internship a truly enriching experience. I am thankful for the open environment where ideas were freely exchanged, challenges were collectively addressed, and meaningful discussions took place.

This internship experience would not have been as rewarding without the collective efforts of everyone mentioned above. The knowledge, skills, and insights gained during this internship will undoubtedly shape my future endeavors in the Machine Learning domain and beyond. I am truly grateful for this opportunity and the support that has accompanied it.

With heartfelt thanks,

Yagnik Paghdal

(22AIML026)

# ABSTRACT

Our project focuses on developing a state-of-the-art university recommendation system to assist students in identifying the best Canadian universities that align with their academic goals and preferences. By employing machine learning algorithms and advanced data analysis techniques, we have crafted a robust model capable of accurately recommending universities based on various parameters such as academic performance, program offerings, location, and student preferences. This achievement is a result of comprehensive data collection, meticulous preprocessing, and thorough model optimization, leading to outstanding performance as demonstrated by precision metrics, accuracy scores, and validation results.

Our research extends beyond mere university recommendations, offering profound implications for educational guidance and student success. It lays the groundwork for a future where education accessibility is enhanced, enabling students to make informed decisions about their higher education pathways. By leveraging technology to provide personalized university recommendations, we aim to create a world where every student has the tools to succeed and every educational goal is attainable.

We are deeply committed to advancing technological innovation and educational progress. Our mission is to champion student success and equality, using technology as a tool to enhance educational opportunities. As we embark on this journey of innovation and inclusion, we invite stakeholders and collaborators to join us in shaping a future where technology acts as a catalyst for positive change. Together, we can empower students and ensure that technology serves to uplift and include everyone, fostering a more inclusive and accessible educational environment.

Our project focuses on the development of a cutting-edge generative AI system capable of transforming textual descriptions into highly detailed and accurate images. By leveraging advanced deep learning techniques and generative models, particularly Generative Adversarial Networks (GANs) and Diffusion Models, we have created a robust framework that can interpret natural language input and generate corresponding visual representations.

This achievement is the result of extensive research in text-to-image synthesis, comprehensive data collection, and meticulous model training and optimization. Our system demonstrates outstanding performance, as evidenced by its ability to produce high-fidelity images that closely align with the provided textual descriptions.

**Table of Contents**

|  |  |
| --- | --- |
| Acknowledgement Abstract  Description of company / organization | i  ii  iii |
| **Chapter 1 Introduction to Uni. Recommendation System** | **1** |
| 1.1 What is Recommendation System | 2 |
| 1.2 Importance of Recommendation System | 2 |
| 1.3 Expected Outcome | 3 |
| **Chapter 2 Methodology** | **4** |
| 2.1 Data collection and preprocessing | 5 |
| 2.2 Dataset Gathering | 5 |
| 2.3 Testing and Cleansing | 5 |
| 2.4 Interface | 5 |
| **Chapter 3 Implementation** | **6** |
| 3.1 Libraries used | 7 |
| 3.2 Code | 8 |
| **Chapter 4 Results** | **10** |
| 4.1Code | 11 |
| 4.2 Prompt | 11 |
| 4.3 Classification report | 12 |
| 4.4 Interface | 13 |
| **Chapter 5 Conclusion** | **14** |
| 5.1 Conclusion | 15 |
| 5.2 Future work | 16 |
| **References** | **18-19** |

# Description of company / organization

**Company Description:** Clumoss is a dynamic and forward-thinking IT services and solutions provider headquartered in Vadodara, India. The company

specializes in delivering a wide range of technological solutions to clients across various industries, focusing on innovation, quality, and customer satisfaction.

**Work Type:** The company engages in multiple facets of the IT industry, including:

* **Software Development:** Custom software solutions tailored to client needs.
* **IT Consulting:** Strategic advice and solutions for IT infrastructure and operations.
* **Web Development:** Creating and maintaining websites and web applications.
* **Mobile app Development:** Designing and developing applications for iOS and Android platform
* **Cloud Services:**Offering cloud computing solutions, including migration, management, and security.
* **Data Analytics:**Providing data analysis, business intelligence, and data management services.
* **Cybersecurity:**Implementing security measures to protect information systems and networks.

**Location and Spread:** Clumoss is based in Vadodara, a city in the state of Gujarat, India. While Vadodara is the primary hub, the company serves clients across various regions, potentially including national and international markets, though its main operations and administrative headquarters are located in Vadodara.

**Group and Divisions:** I am working closely with a team that includes:

* **Team Lead:**Your immediate supervisor who guides your daily tasks and projects.
* **Senior Developers/Analysts:** Experienced professionals who mentor you and provide technical assistance.
* **Fellow Interns:**Other interns who may be working on similar or related projects, offering a collaborative environment

**Number of Employees:** While the exact number of employees can vary, a company of this nature typically employs between 50 to 200 professionals, depending on its size and scope of operations.

### Main functions of the company

* **Delivering Customized IT Solutions:**Providing tailored software, consulting, and development services.
* **Innovation and Development:**Continuously improving and creating new technologies to meet client needs.
* **Customer Support:**Offering comprehensive support to ensure client satisfaction and project success.
* **Training and Development:**Providing ongoing training for employees to stay current with technological advancements.

# Chapter 1

**Introduction to Uni. Recommendation System**

### 1.1 PROJECT OVERVIEW

The project aims to develop two systems: a Canadian University Recommendation system and a Text-to-Image Prompt Generation system. The University Recommendation system will help students identify suitable Canadian universities based on their preferences and qualifications. The Text-to-Image Prompt Generation system will create images from textual descriptions, leveraging generative AI models.

### 1.2 OBJECTIVES

**Canadian University Recommendation System:**

1. **Identify Key Factors:** Determine critical factors influencing university selection.
2. **Develop Recommendation Models:** Create models to provide personalized university recommendations.
3. **Enhance Decision-Making:** Offer actionable insights to help students make informed choices.
4. **Optimize Resource Allocation:** Efficiently match students to universities based on preferences and qualifications.

**Text-to-Image Prompt Generation System:**

1. **Generate Accurate Images:** Develop models to accurately convert textual descriptions into images.
2. **Improve Model Performance:** Enhance the quality and relevance of generated images.
3. **Facilitate Creative Workflows:** Support creative projects by providing high-quality visual content from text prompts.
4. **Expand Use Cases:** Explore various applications for text-to-image generation in different industries.

### 1.3 SCOPE

**Canadian University Recommendation System:**

* **Data Collection:** Gather data on Canadian universities, including courses, admission criteria, and student reviews.
* **Data Preprocessing:** Clean, normalize, and encode data for analysis.
* **Exploratory Data Analysis (EDA):** Explore data to uncover trends and relationships.
* **Feature Engineering:** Create relevant features to improve recommendation accuracy.
* **Model Development:** Build and evaluate machine learning models for university recommendation.
* **Model Evaluation:** Assess model performance using metrics such as precision, recall, and F1 score.

**Text-to-Image Prompt Generation System:**

* **Data Collection:** Collect datasets containing textual descriptions and corresponding images.
* **Data Preprocessing:** Clean and prepare data for training generative models.
* **Exploratory Data Analysis (EDA):** Analyze data to identify patterns and relationships.
* **Model Development:** Train and optimize generative AI models for text-to-image conversion.
* **Model Evaluation:** Evaluate models using metrics such as image quality and relevance.

### 1.4 METHODOLOGY

**Canadian University Recommendation System:**

1. **Data Collection and Preprocessing:** Collect and clean data to ensure quality and consistency.
2. **Exploratory Data Analysis:** Analyze data to identify patterns and correlations.
3. **Model Development:** Train and optimize machine learning models for university recommendations.
4. **Model Evaluation:** Evaluate models using cross-validation and test set performance metrics.

**Text-to-Image Prompt Generation System:**

1. **Data Collection and Preprocessing:** Collect and clean data for model training.
2. **Exploratory Data Analysis:** Explore data to uncover patterns and trends.
3. **Model Development:** Train and fine-tune generative AI models for text-to-image tasks.
4. **Model Evaluation:** Assess model performance using qualitative and quantitative metrics.

### 1.5 EXPECTED OUTCOME

**Canadian University Recommendation System:**

* Accurate and personalized university recommendations.
* Identification of key factors influencing university choice.
* Insights for students to make informed decisions.
* Efficient matching of students to universities based on preferences and qualifications.

# Chapter 2 Methodology

### 2.1 SOFTWARE TOOLS

**For Both Projects:**

* **Anaconda Navigator:** A graphical interface bundled with the Anaconda distribution for managing environments and packages used in data science and scientific computing.
* **Jupyter Notebook:** An open-source web application that allows you to create and share documents containing live code, equations, visualizations, and narrative text, commonly used for data analysis and research.
* **Jupyter Lab:** The next-generation web-based interface for Project Jupyter, offering a more flexible environment with support for multiple notebooks, text editors, and extensions, ideal for complex data science workflows.
* **Visual Studio Code (VS Code):** A free, open-source code editor by Microsoft, supporting various programming languages with features like syntax highlighting, debugging, and Git integration, widely used for software development and data science projects.

**Additional Tools for Text-to-Image Prompt Generation:**

* **TensorFlow/PyTorch:** Open-source machine learning frameworks that provide comprehensive tools for training and deploying deep learning models, essential for developing generative AI models.
* **GAN (Generative Adversarial Networks) Libraries:** Specialized libraries and frameworks for building and training GANs, crucial for text-to-image generation.

### 2.2 PROGRAMMING LANGUAGE

* **Python:** Python is a high-level, interpreted programming language known for its simplicity and readability. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming styles. Python's extensive standard library and third-party packages make it versatile for a wide range of applications, from web development and scientific computing to data analysis, artificial intelligence, and automation. Its clean syntax and community-driven development have contributed to its popularity among beginners and seasoned developers alike.

### 2.3 DOCUMENTATION TOOLS

* **MS Word:** Microsoft Word (MS Word) is a widely used word processing application developed by Microsoft. It allows users to create, edit, format, and save documents. MS Word offers features such as spell check, grammar check, templates, tables, and advanced formatting options like styles and themes. It is integral for creating professional documents, reports, letters, and manuscripts, and supports collaboration through features like track changes and comments.
* **Markdown:** A lightweight markup language with plain text formatting syntax, commonly used for creating formatted text using a plain text editor. Markdown is widely used for documentation, especially in software development, due to its simplicity and compatibility with various platforms and tools, including GitHub and Jupyter Notebooks.

# Chapter 3 Implementation

**3.1.1 Data Collection and Integration**

* **Feature:** Gather data from multiple sources such as university databases, course catalogs, and student reviews.
* **Importance:** Ensures comprehensive data coverage for accurate university recommendation models.

**3.1.2 Data Preprocessing**

* **Feature:** Cleanse, transform, and integrate data to ensure consistency and quality.
* **Importance:** Improves data accuracy and reliability for effective analysis.

**3.1.3 Exploratory Data Analysis (EDA)**

* **Feature:** Analyze data to identify patterns, correlations, and outliers.
* **Importance:** Provides insights into factors influencing university selection, guiding feature selection and engineering.

**3.1.4 Feature Engineering**

* **Feature:** Create new features from existing data to enhance model performance.
* **Importance:** Improves predictive capabilities by capturing relevant aspects of student preferences and qualifications.

**3.1.5 Model Development**

* **Feature:** Implement machine learning algorithms (e.g., collaborative filtering, decision trees) for university recommendation.
* **Importance:** Generates predictive models to provide personalized university recommendations accurately.

**3.1.6 Model Evaluation and Validation**

* **Feature:** Assess model performance using metrics such as precision, recall, and F1 score.
* **Importance:** Validates model effectiveness and reliability for deployment.

**3.1.7 Deployment and Integration**

* **Feature:** Implement models into production systems for real-time university recommendations.
* **Importance:** Enables students to receive up-to-date and relevant university recommendations based on their profiles.

**3.1.8 Reporting and Visualization**

* **Feature:** Generate dashboards and reports to communicate insights and recommendations.
* **Importance:** Facilitates decision-making for students and educational advisors.

**For Text-to-Image Prompt Generation System:**

**3.1.1 Data Collection and Integration**

* **Feature:** Gather data from multiple sources such as image datasets and corresponding textual descriptions.
* **Importance:** Ensures comprehensive data coverage for accurate text-to-image generation models.

**3.1.2 Data Preprocessing**

* **Feature:** Cleanse, transform, and integrate data to ensure consistency and quality.
* **Importance:** Improves data accuracy and reliability for effective analysis.

**3.1.3 Exploratory Data Analysis (EDA)**

* **Feature:** Analyze data to identify patterns, correlations, and outliers.
* **Importance:** Provides insights into factors influencing image quality and relevance, guiding feature selection and engineering.

**3.1.4 Feature Engineering**

* **Feature:** Create new features from existing data to enhance model performance.
* **Importance:** Improves generative capabilities by capturing relevant aspects of textual descriptions and visual attributes.

**3.1.5 Model Development**

* **Feature:** Implement generative AI models (e.g., GANs, transformers) for text-to-image prompt generation.
* **Importance:** Generates models to accurately convert textual descriptions into images.

**3.1.6 Model Evaluation and Validation**

* **Feature:** Assess model performance using metrics such as image quality, relevance, and user satisfaction.
* **Importance:** Validates model effectiveness and reliability for deployment.

**3.1.7 Deployment and Integration**

* **Feature:** Implement models into production systems for real-time text-to-image generation.
* **Importance:** Enables users to generate high-quality images from text prompts instantly.

**3.1.8 Reporting and Visualization**

* **Feature:** Generate dashboards and reports to communicate insights and model outputs.
* **Importance:** Facilitates decision-making and creative workflows for users.

### 3.2 FEASIBILITY STUDY

**3.2.1 Technical Feasibility**

* **Assessment:** Determine if technology and infrastructure (e.g., computing resources, software tools) are sufficient for data processing, modeling, and deployment.
* **Outcome:** Ensure the projects can be implemented within technical constraints and performance requirements.

**3.2.2 Operational Feasibility**

* **Assessment:** Evaluate organizational readiness and capability to integrate models into existing operations.
* **Outcome:** Identify potential challenges and ensure adequate support for deployment and maintenance.

**3.2.3 Economic Feasibility**

* **Assessment:** Calculate costs associated with data acquisition, software licenses, infrastructure, and manpower.
* **Outcome:** Determine if the projects provide a favorable return on investment (ROI) through improved university recommendations and efficient text-to-image generation.

**3.2.4 Legal and Ethical Feasibility**

* **Assessment:** Ensure compliance with data privacy regulations (e.g., GDPR, CCPA) and ethical considerations regarding data usage.
* **Outcome:** Mitigate risks associated with data handling and ensure responsible use of information.

**3.2.5 Schedule Feasibility**

* **Assessment:** Develop a timeline and milestones for each phase of the projects (e.g., data collection, model development, deployment).
* **Outcome:** Ensure the projects can be completed within the specified timeframe and align with objectives.

# Chapter 4 Results

## **4.1 DATASET**

### CODE

1)

import tkinter as tk

import customtkinter as ctk

from PIL import ImageTk

from authtoken import auth\_token

import torch

from torch import autocast

from diffusers import StableDiffusionPipeline

# Create the app

app = tk.Tk()

app.geometry("532x632")

app.title("Stable Bud")

ctk.set\_appearance\_mode("dark")

prompt = ctk.CTkEntry(height=40, width=512, text\_font=("Arial", 20), text\_color="black", fg\_color="white")

prompt.place(x=10, y=10)

lmain = ctk.CTkLabel(height=512, width=512)

lmain.place(x=10, y=110)

modelid = "CompVis/stable-diffusion-v1-4"

device = "cuda"

pipe = StableDiffusionPipeline.from\_pretrained(modelid, revision="fp16", torch\_dtype=torch.float16, use\_auth\_token=auth\_token)

pipe.to(device)

def generate():

with autocast(device):

image = pipe(prompt.get(), guidance\_scale=8.5)["sample"][0]

image.save('generatedimage.png')

img = ImageTk.PhotoImage(image)

lmain.configure(image=img)

trigger = ctk.CTkButton(height=40, width=120, text\_font=("Arial", 20), text\_color="white", fg\_color="blue", command=generate)

trigger.configure(text="Generate")

trigger.place(x=206, y=60)

app.mainloop()

2)

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

data = pd.read\_excel("d.xlsx")

X = data.drop(columns=['Uni\_name', 'Status'])

y = data['Uni\_name']

X = pd.get\_dummies(X)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

clf = RandomForestClassifier()

clf.fit(X\_train, y\_train)

# predictions = clf.predict(X\_test)

#

# accuracy = accuracy\_score(y\_test, predictions)

# print("Accuracy:", accuracy)

#

# print("Predicted universities:")

# predicted\_universities = set(predictions)

# for uni in predicted\_universities:

# print(uni)

def recommend\_universities():

student\_details = {}

student\_details['Subject'] = input("Subject: ")

student\_details['Prev\_Uni'] = input("Previous University: ")

student\_details['CGPA'] = float(input("CGPA: "))

student\_details['12th %'] = float(input("12th Percentage: "))

student\_details['10th %'] = float(input("10th Percentage: "))

student\_details['IELTS Listening'] = float(input("IELTS Listening: "))

student\_details['IELTS Reading'] = float(input("IELTS Reading: "))

student\_details['IELTS Writing'] = float(input("IELTS Writing: "))

student\_details['IELTS Speaking'] = float(input("IELTS Speaking: "))

student\_details['Overall Bands'] = float(input("Overall Bands: "))

student\_details['Uni\_fees'] = float(input("University Fees: "))

student\_details['Location'] = input("Location: ")

student\_details['Course'] = input("Course: ")

student\_df = pd.DataFrame([student\_details])

student\_encoded = pd.get\_dummies(student\_df)

student\_encoded = student\_encoded.reindex(columns=X\_train.columns, fill\_value=0)

prediction = clf.predict(student\_encoded)

return prediction

recommended\_universities = recommend\_universities()

print("Recommended Universities:", recommended\_universities)

# Chapter 5 Conclusion

* 1. **Conclusion**

In Conclusion, university recommendation system for Canadian universities signifies a major step forward in educational technology and student support services. By employing sophisticated machine learning algorithms and data analysis techniques, we have developed a system that offers personalized university recommendations, helping students make informed decisions about their higher education paths.

The success of our project highlights the importance of integrating technology into the educational sector to enhance accessibility and guidance. This system not only simplifies the decision-making process for students but also ensures that they find the best fit for their academic and personal preferences. The comprehensive data-driven approach ensures accuracy and relevance in the recommendations, providing valuable insights for prospective students.

Secondly, our generative AI text-to-image project represents a significant advancement in the field of artificial intelligence and computer vision. By successfully developing a system that can accurately convert textual descriptions into detailed visual representations, we have demonstrated the powerful potential of generative models. This work not only showcases the capabilities of advanced deep learning techniques but also opens up numerous possibilities for practical applications across various domains.

The success of our project highlights the importance of integrating technology into the educational sector to enhance accessibility and guidance. This system not only simplifies the decision-making process for students but also ensures that they find the best fit for their academic and personal preferences. The comprehensive data-driven approach ensures accuracy and relevance in the recommendations, providing valuable insights for prospective students.

Our commitment to advancing educational technology is reflected in this project, which aims to democratize access to information and support for all students. As we continue to improve and expand our recommendation system, we look forward to collaborating with educational institutions, stakeholders, and the wider community. Together, we can create a future where every student is empowered to achieve their educational goals through informed and personalized guidance, fostering a more inclusive and supportive educational environment.

### Future work for University Recommendation System

While the current university recommendation system for Canadian universities has demonstrated promising results, several avenues for future work can enhance its capabilities and expand its applications. Future efforts can focus on improving the system's accuracy, robustness, and usability in real-world scenarios.

**Dataset Expansion:** Expanding the dataset to include more diverse student profiles, academic records, and preferences can help the system generalize better. Incorporating data from different regions, backgrounds, and educational systems can make the recommendations more inclusive and comprehensive.

**Real-time Updates:** Enhancing the system to provide real-time updates on university information, such as changes in admission criteria, new programs, and campus events, would significantly improve its utility. This involves integrating with university databases and using web scraping techniques to keep the information current.

**Personalized Feedback and Guidance:** Developing personalized feedback mechanisms that offer students detailed insights into why specific universities are recommended can enhance the system's transparency and trustworthiness. Providing guidance on application strategies, scholarship opportunities, and career prospects can add value to the recommendations.

**Advanced Machine Learning Techniques:** Utilizing advanced machine learning techniques, such as ensemble learning and reinforcement learning, can improve the system's accuracy and adaptability. These techniques can help in better handling the complex and dynamic nature of student preferences and university offerings.

**User Interface Enhancements:** Improving the user interface to include features like interactive dashboards, visualizations of university comparisons, and personalized recommendation paths can make the system more engaging and user-friendly. Enhancing accessibility options within the interface can ensure it caters to a broader range of users.

**Deployment and Scalability:** Future work should focus on deploying the system in various environments, such as mobile applications, web services, and educational platforms. Ensuring the scalability and robustness of the system in different deployment scenarios will make it more versatile and widely applicable.

### 5.3 Future work for University Recommendation System

While the current generative AI text-to-image system has shown promising results, there are several avenues for future work to enhance its capabilities and expand its applications. Future efforts can focus on improving the model’s accuracy, diversity, and usability in real-world scenarios.

**Dataset Expansion:** Expanding the dataset to include more diverse and complex textual descriptions and corresponding images can help the model generalize better. Incorporating different styles, contexts, and subject matters can broaden the model’s applicability and improve its creative output.

**Improving Image Quality:** Enhancing the model to generate higher-resolution images with finer details would be a significant advancement. This involves optimizing the model architecture and training process to focus on image quality and realism.

**Contextual Understanding:** Developing the model’s ability to understand and incorporate contextual information from the text can lead to more accurate and meaningful image generation. This includes understanding nuances, emotions, and specific details described in the text.

**Multi-modal Inputs:** Integrating other modalities, such as audio descriptions and user sketches, can provide a more comprehensive generative AI system. This multi-modal approach can lead to richer and more accurate visual representations.

**User Interface Enhancements:** Improving the user interface to include features like interactive text-to-image customization, feedback mechanisms, and support for iterative refinement can make the system more interactive and user-friendly. Enhancing accessibility options within the interface itself can ensure it caters to a broader range of users.

**Deployment and Scalability:** Future work should also focus on deploying the model in various environments, such as mobile applications, web services, and creative software. Ensuring the scalability and robustness of the system in different deployment scenarios will make it more versatile and widely applicable.

# References

**References**

Web References:

1. <https://youtu.be/YSuWejNaSfE?si=13ysKYZcIfBBqhdL>
2. <https://youtu.be/7xc0Fs3fpCg?si=6fIe2EzqPvOycN2o>
3. <https://chatgpt.com/c/43d3e400-8036-4ce1-a9d4-d3a91e7fd33b>
4. <https://en.wikipedia.org/wiki/Generative_artificial_intelligence>
5. <https://en.wikipedia.org/wiki/Random_forest>