

GOVT ARTS AND SCIENCE COLLEGE FOR WOMEN,BARGUR.

BACHELOR OF COMPUTER APPLICATION

*PROJECT TITLE : INTELLIGENT ADMISSIONS - THE FUTURE OF
UNIVERSITY*

TEAM LEADER : SARASWATHI M

TEAM MEMBERS :

ANANDHI C

ASWINI BAI S

KAVIYA S

Intelligent Admissions : The Future of University Decision Making with Machine Learning

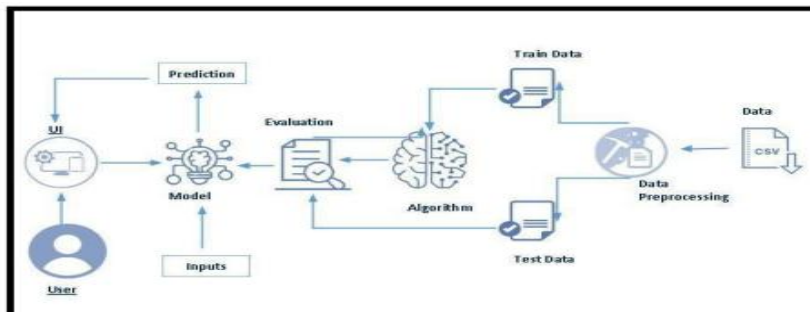
1 INTRODUCTION

1.1 Overview

University admission is the process by which students are selected to attend a college or university. The process typically involves several steps, including submitting an application, taking entrance exams, and participating in interviews or other evaluations.

Students are often worried about their chances of admission in University. the university admission process for students can be demanding, but by being well-informed, prepared, and organized, students can increase their chances of being admitted to the university of their choice.

Technical Architecture:



1.2 Purpose

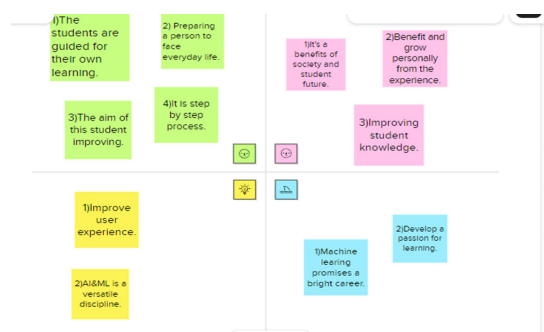
The aim of this project is to help students in short listing universities with their profiles.

Machine learning algorithms are then used to train a model on this data, which can be used to predict the chances of future applicants being admitted. With this project, students can make more informed decisions about which universities to apply to, and universities can make more efficient use of their resources by focusing on the most promising applicants.

The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

2 Problem Definition & Design Thinking

2.1 Empathy Map



2.2 Ideation & Brainstorming Map



3 RESULT

Untitled3.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] 1 import numpy as np
    2 import pandas as pd
    3 import matplotlib.pyplot as plt
    4 import seaborn as sns
    5 %matplotlib inline
    6
    7

[ ] 1 data=pd.read_csv('Admission_predict.csv')
    2 data
    3
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit	
0	1	337	118		4	4.5	4.5	9.65	1	0.92
1	2	324	107		4	4.0	4.5	8.87	1	0.76
2	3	316	104		3	3.0	3.5	8.00	1	0.72
3	4	322	110		3	3.5	2.5	8.67	1	0.80
4	5	314	103		2	2.0	3.0	8.21	0	0.65

0s completed at 7:18 PM

Untitled3.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] 398
    399
    400 rows x 9 columns

[ ] 1 data.info()
    2
    3

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Serial No.            400 non-null   int64
1   GRE Score             400 non-null   int64
2   TOEFL Score           400 non-null   int64
3   University Rating     400 non-null   int64
4   SOP                   400 non-null   float64
5   LOR                   400 non-null   float64
6   CGPA                  400 non-null   float64
7   Research              400 non-null   int64
8   Chance of Admit       400 non-null   float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
```

0s completed at 7:18 PM

Untitled3.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] University Rating  False
    SOP              False
    LOR              False
    CGPA             False
    Research         False
    Chance of Admit  False
    dtype: bool

[ ] 1 #let us rename the column Chance of Admit becuseit has trainling space
    2 data=data.rename(columns = {'Chance of Admit ':' Chance of Admit'})

[ ] 1 data.describe()
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
count	400.000000	400.000000	400.000000	400.000000	400.000000	400.000000	400.000000	400.000000	400.000000
mean	200.500000	316.807500	107.410000	3.087500	3.400000	3.452500	8.598925	0.547500	0.724350
std	115.614301	11.473646	6.069514	1.143728	1.006869	0.898478	0.596317	0.498362	0.142609
min	1.000000	290.000000	92.000000	1.000000	1.000000	1.000000	6.800000	0.000000	0.340000
25%	100.750000	308.000000	103.000000	2.000000	2.500000	3.000000	8.170000	0.000000	0.640000
50%	200.500000	317.000000	107.000000	3.000000	3.500000	3.500000	8.610000	1.000000	0.730000

0s completed at 7:18 PM

4 ADVANTAGES & DISADVANTAGES

Advantages

- 1. Saves time and money*
- 2. Better retention*
- 3. Personalized learning*
- 4. Cost-effective*
- 5. Environment-friendly*

Disadvantages

- 1. Lacks social interaction*
- 2. Inaccessible to others*
- 3. Cheating is unavoidable*
- 4. Requires self-motivation and proper time management skills*
- 5. Focuses more on theory*

5 APPLICATIONS

To start we need a collaborative, industry-wide commitment to more equitable and exclusive admissions policies and practices. At the gathering, teams imagined cross-cutting solutions as key to improving the admission process, hanging on a collective approach action approach to change.

6 CONCLUSION

We conclude by suggested that the future of universities will be taking a crucial and leading role within a governance milieu of many collaborating public and private agencies and organisations to be more inclusive, reflective and restorative organisations.

7 FUTURE SCOPE

Online learning and modern innovations have brought about a major shift in the way society and students perceive education.

Combine that with technology and we have a new pedagogy structure for the Indian education system.

As education practices are being recalibrated to complement technological capacities with the emergence of the Indian education appears bright.

8 APPENDIX

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 %matplotlib inline
6
7
```



```
1 data=pd.read_csv('Admission_predi
2 data
3
```

```
[ ] 1 data.info()
2
3
```

```
[ ] 1 data.isnull().any()
2
```

```
[ ] 1 #let us rename the column Chance of Admit becuseit has trainLing space
2 data=data.rename (columns = {'Chance of Admit ':' Chance of Admit'})
```



```
1 data.describe()
```

```
[ ] 1 sns.distplot(data['GRE Score'])
```

{x}



```
1 sns.scatterplot(x='University Rating',y='CGPA',data=data,color='Red',s=100)
```



```
[ ] 1 x=data.iloc[:,0:7].values
    2 x
    3
```

```
1 y=data.iloc[:,7:].values
2 y
```