

# **STUDENT EMPLOYABILITY PREDICTION THROUGH INTERNSHIP AND STUDENT CONTEXT FEATURES**

*A*

*Project Report*

*Submitted in partial fulfilment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**1602-19-737-116**

**Gali Suma Sri**

**1602-19-737-107**

**Uvrajana Sneha**

*Under the guidance of*

**Ms. S. Rajyalaxmi**

**Assistant Professor**



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

*ACCREDITED BY NAAC WITH 'A++' GRADE*

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31 2023**

# **Vasavi College of Engineering (Autonomous)**

***ACCREDITED BY NAAC WITH 'A++' GRADE***

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



## **DECLARATION BY THE CANDIDATE**

We, **Gali Suma Sri** and **Uvrajana Sneha** bearing hall ticket number, **1602-19-737-116** and **1602-19-737-107** hereby declare that the project report entitled **Student Employability Prediction through Internship and Student Context Features** under the guidance of **Ms.S.Rajyalaxmi, Assistant Professor**, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering in Information Technology**

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

**Gali Suma Sri**  
**1602-19-737-116**

**Uvrajana Sneha**  
**1602-19-737-107**

# **Vasavi College of Engineering (Autonomous)**

***ACCREDITED BY NAAC WITH 'A++' GRADE***

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



## **BONAFIDE CERTIFICATE**

This is to certify that the project entitled **Student Employability Prediction through Internship and Student Context Features** being submitted by **Gali Suma Sri** and **Uvrajana Sneha** bearing hall ticket number, **1602-19-737-116** and **1602-19-737-107** in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by them under my guidance.

**Ms. S. Rajyalaxmi**  
Associate Professor  
Internal Guide

**Dr. K. Ram Mohan Rao,**  
Professor,  
HOD, IT

## ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of the project seminar would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them.

It is with immense pleasure that we would like to take the opportunity to express our humble gratitude to **Ms.S.Rajyalaxmi, Assistant Professor, Information Technology** under whom we executed this project. We are also grateful to **Ms. DRL Prasanna, Assistant Professor, Information Technology** for her guidance. Their constant guidance and willingness to share their vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

We are very much thankful to **Mr. Ram Mohan Rao, Professor and HOD, Information Technology**, for his kind support and for providing necessary facilities to carry out the work.

We wish to convey our special thanks to **Dr.S.V.Ramana, Principal of Vasavi College of Engineering** for giving the required information in doing my project work. Not to forget, we thank all other faculty and non-teaching staff, and my friends who had directly or indirectly helped and supported me in completing my project in time.

We also express our sincere thanks to the Management for providing excellent facilities. Finally, we wish to convey our gratitude to our family who fostered all the requirements and facilities that we need.

# **ABSTRACT**

Universities around the world are keen to develop study plans that will guide their graduates to success in the job market. The internship course is one of the most significant courses that provides an experiential opportunity for students to apply knowledge and to prepare to start a professional career. However, internships do not guarantee employability, especially when a graduate's internship performance is not satisfactory and the internship requirements are not met.

Many factors contribute to this issue making the prediction of employability an important challenge for researchers in the higher education field. In this paper, our aim is to introduce an effective method to predict student employability based on Internship context and Student context and using Gradient Boosting classifiers. Our contributions consist of harnessing the power of gradient boosting algorithms to perform context-aware employability status prediction processes. Student employability prediction relies on identifying the most predictive features impacting the hiring opportunity of graduates. Hence, we define two context models, which are the student context based on the student features and the internship context based on internship features.

Experiments are conducted using three gradient boosting classifiers: eXtreme Gradient Boosting (XGBoost), Category Boosting (CatBoost) and Light Gradient Boosted Machine (LGBM). The results obtained showed that applying LGBM classifiers over the internship context performs the best compared to student context. Therefore, this study provides strong evidence that the employability of graduates is predictable from the internship context.

# TABLE OF CONTENTS

<b>List of Figures</b>	viii
<b>List of Tables</b>	ix
<b>List of Abbreviations</b>	x
<b>1. INTRODUCTION</b>	11
1.1 Problem Statement	11
1.2 Proposed Method	11
1.3 Scope & Objectives of the Proposed Work	12
1.4 Organization of the Report	12
<b>2. LITERATURE SURVEY</b>	14
<b>3. PROPOSED SYSTEM</b>	20
3.1 System Specifications	21
3.1.1 Software Requirements	21
3.1.2 Hardware Requirements	21
3.1.3 User Requirements	21
3.2 Methodology	22
3.2.1 Training Model	23
3.2.2 Gradient Boosting Algorithms	25
3.2.3 Pseudocode	27
<b>4. EXPERIMENTAL SETUP &amp; RESULTS</b>	32
4.1 Datasets	32
4.1.1 Data Preprocessing	33
4.1.2 Parameter Setting	35
4.2 Results & Test Analysis	36
4.2.1 Outputs from Gradient Boosting Models	36
4.2.2 Metrics of Evaluation	39
4.2.3 Analysis	42
<b>5. SUMMARY AND FUTURE SCOPE</b>	44
5.1 Conclusion	44

5.2 Future Scope	45
5.3 Limitations	46

## **REFERENCES**

## **APPENDIX**

## LIST OF FIGURES

<b>Fig. No</b>	<b>Description</b>	<b>Page No</b>
2.1	Literature Review Framework	16
3.2.1	Architecture	23
3.2.2	Training model selection process	23
3.2.3	Context model for employability prediction	24
4.1.1	Dataset after data elimination	34
4.1.2	Dataset after categorical encoding	35
4.2.1	Performance of Catboost model	36
4.2.2	Output 1 from Catboost model	37
4.2.3	Output 2 from Catboost model	37
4.2.4	Performance of XGBoost model	37
4.2.5	Output 1 from XGBoost model	38
4.2.6	Output 2 from XGBoost model	38
4.2.7	Performance of LGBM model	38
4.2.8	Output 1 from LGBM model	39
4.2.9	Output 2 from LGBM model	39
4.2.10	Graphical representation of evaluation metrics	42



## LIST OF TABLES

<b>Table No.</b>	<b>Table Name</b>	<b>Page No</b>
2.1	Goal, domain and context awareness	17
4.1.1	Dataset	32
4.2.1	Confusion matrix	40
4.2.2	Performance evaluation of gradient boosting models	42

## LIST OF ABBREVIATIONS

Abbreviation	Full Form	Page No
ML	Machine Learning	11
DBN-SR	Deep Belief Network and Softmax Regression	18
LR	Linear Regression	18
CatBoost	Categorical Boosting	24
XGBoost	eXtreme Gradient Boosting	24
LGBM	Light Gradient Boosting Machine	24
Cs	Student Context	24
Ci	Internship Context	24
CGPA	Cumulative Grade Point Average	32
TP	True Positive	39
TN	True Negative	39
FP	False Positive	39
FN	False Negative	39

# **1. INTRODUCTION**

## **1.1 PROBLEM STATEMENT**

Currently in most companies, internship programs are considered a mandatory part of the student curriculum. The internship is defined as ‘‘a short-term practical work experience in which students receive training and gain experience in a specific field or career area of their interest’’. Internship programs ease the transition to the workplace, as they are an effective method to fill in the gap between what is learned in universities and employment demands. An internship program provides a student with the opportunity to practice the content learned during lectures, integrate theoretical knowledge with practical experiences gained through experiential learning, become familiarized with the workplace, clarify career expectations, and develop valuable practical experience and job relevant skills.

Moreover, internship programs have proven to be one of the most important experiential learning activities that enhances graduates’ employability. However, planning, designing and coordinating an effective and successful internship is challenging. As a result, universities continue to produce graduates who may be considered unfit for the job market. Although there are studies that focus on predicting employability, such as, due to the lack of internship data and the difficulty of gathering and analysing these data, there is a shortage of research on the internship factors that affect employability of students after graduation.

## **1.2 PROPOSED METHOD**

In this study, we aim to predict student employability based on their internship program and the context related knowledge; to our knowledge, there is no prior research that has predicted student employability based on contextual knowledge. In the field of Machine Learning (ML), context awareness is a relatively new field of research. Researchers exploring ML and considering the context are very limited. The researchers compared the use of a trained general model that uses all contexts in contrast to a system made of a set of specialized models trained for each specific operating context. Therefore, in this work, we utilize Gradient Boosting Models to predict student employability through student and internship contexts and reveal the most predictive features leading to improved chances in employment.

Our contributions include mainly: (i) a flexible context aware employability prediction process model, (ii) a context model for employability prediction, and (iii) a context-based ML approach

for student employability prediction. The output feature of this approach is the student employability status (i.e., identifying whether the student is most likely to be employed or unemployed).

### **1.3. SCOPE & OBJECTIVES OF THE PROPOSED WORK**

#### **1.3.1 Objective:**

The main objective of this proposed work is to predict the student employability based on their internship program and context-related knowledge. The research aims to identify the most predictive features that contribute to the employability status of a student. The proposed approach utilizes Gradient Boosting Models and a context-based methodology for employability prediction. The output feature of this approach is the student employability status, which identifies whether the student is most likely to be employed, unemployed, continue their studies or be in training. Overall, the objective of this work is to contribute to the field of Machine Learning and context-awareness research, by exploring the use of context models for prediction.

#### **1.3.2 Scope:**

The scope of this proposed work is limited to the prediction of student employability based on their internship program and context-related knowledge. The study will explore the use of context-awareness in employability prediction, which is a relatively new field of research in Machine Learning. The research will utilize Gradient Boosting Models to train the predictive model and reveal the most predictive features. The proposed approach will be tested on a dataset of student and internship contexts, and the output feature will be the employability status of the students.

### **1.4. ORGANIZATION OF THE REPORT**

**Introduction:** This section provides background information on the project, defines the scope of the report, and states the objectives and research questions. It should also provide an overview of the structure of the report.

**Literature review:** This section summarizes and evaluates the existing research and literature related to the project. It should provide a critical analysis of the key theories, concepts, and empirical studies related to the research questions.

**Proposed Work:** This section describes the research methods and procedures used in the project. It should also discuss the basic flow of the project.

**Experimental Study:** This section describes each dataset, pre-processing. It should also focus on result part.

**Summary and Future Scope:** This section summarizes the main findings and conclusions of the project, and identifies any recommendations for future research or action.

**References:** This section provides a list of all sources cited in the report, using a consistent citation style.

**Appendices:** This section includes any additional information or materials that were not included in the main body of the report, but are relevant to the project, such as raw data, interview transcripts, or survey questionnaires.

## 2.LITERATURE SURVEY

There has been a growing interest in predicting student employability through various factors such as student characteristics, academic performance, and internship experience. Based on that some of the projects included are:

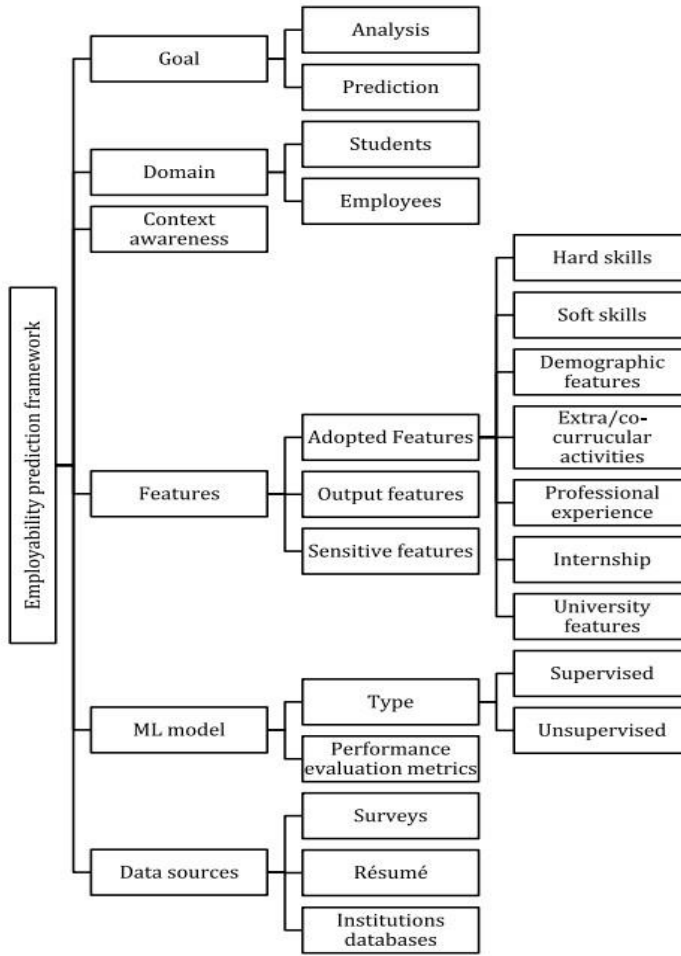
1. "Predicting graduate employability: An exploratory study" by J. Dawkins and S. Martin (2016) This study used logistic regression to predict graduate employability based on four factors: academic performance, work experience, extracurricular activities, and personal characteristics. The results showed that work experience was the strongest predictor of employability.
2. "Predictors of graduate employability: A systematic literature review" by S. Hussain, S. Ali, and S. Akram (2019) This literature review analysed 30 studies on graduate employability and identified several predictors, including academic performance, work experience, internships, extracurricular activities, and soft skills. The review found that internships were consistently identified as a significant predictor of employability.
3. "Employability of business graduates: The role of internships and work-integrated learning" by M. C. Roodt and J. K. De Klerk (2018) This study examined the impact of internships and work-integrated learning on the employability of business graduates. The results showed that internships had a positive effect on employability, particularly for graduates with no work experience.
4. "Predicting student employability in the hospitality industry: The role of work-integrated learning and soft skills" by L. Fan and Y. Liu (2020) This study investigated the impact of work-integrated learning and soft skills on the employability of hospitality students. The results showed that work-integrated learning had a positive effect on employability, and that soft skills such as communication and teamwork were also important predictors.
5. "Predicting graduate employability through career competencies: An empirical study in China" by Y. Yang and Y. Lu (2020) This study examined the relationship between career competencies and graduate employability in China. The results showed that internship experience was a significant predictor of employability, and that career competencies such as adaptability and initiative were also important predictors.

**A. INTERNSHIP AND EMPLOYMENT:** A well planned internship program, jointly coordinated by industry representatives and academic institutions, is expected to increase its effectiveness. An internship program is successful whenever students are satisfied, assigned tasks are relevant and supervisors are qualified. Furthermore, some studies identified the correlation between internship satisfaction and employability, i.e., consultation and treatment provided during internship influence the satisfaction of students with the internship and their decision to remain in the same industry in the future. Eurico et al. claim that the satisfaction of a student with the internship program enhances his or her employability skills.

Furthermore, Hugo assumes that internships, majors and co-curricular activities impact employability. Previous studies have proven that internship programs generally increase employability; however, it is still unclear which internship play a major role in employability.

**B. ML FOR EMPLOYMENT PREDICTION:** Machine learning is widely applied in many research fields. In higher education, ML is used mainly to enhance curriculum outcomes and graduate features. Many researchers are interested in this field and have conducted several studies to discuss the contribution of ML in continuous quality improvement. As recommended by the research community we used Google Scholar, IEEE Xplore, ACM Digital Library, ScienceDirect and Scopus Database to extract relevant published papers between 2012 and 2021. The literature search was based on many fields: employability in general, graduate employability in higher education, prediction, ML and ML algorithms. Twenty relevant studies were retained. Our examination of papers is conducted by the following criteria:

- Goal. The goal of the study.
- Domain. The application domain: who is the subject for which employability is predicted (i.e., employees, students)?
- Context awareness. Indicates whether the context is supported.
- Adopted features. The employability signals adopted in the study, e.g. computational skills, communication skills, number of majors, grades.
- Output features. The predicted outputs.
- Sensitive features. The selected features that truly influence the employability aspects.
- ML model. The ML model used.
- Dataset sources. The dataset source(s) used.
- Performance evaluation metrics. The performance evaluation metrics considered. The following subsections analyse and compare the surveyed studies with respect to the framework proposed in given figure.



**FIGURE 1. Literature review framework.**

**Fig 2.1. Literature review Framework**

**C. GOAL, DOMAIN AND CONTEXT AWARENESS:** This subsection analyses the surveyed studies according to the following criteria: Goal, Domain and Context awareness. The below table gives a summary of the retained studies.

1. Goal: Regarding the goal criterion, most of the studies aim to predict student employability. In the same vein, the study of L. Hugo aimed to predict which students will graduate with a non-employment offer. Kommina et al. and Kalpana and Venkatalakshmi aimed to predict academic performance. Casuat et al. focused on the identification of the most dominant attributes that affect employability. Patel et al. aimed to predict the most suitable job domains.
2. Domain: with respect to the domain criterion, the analysis of the data gathered from the surveyed studies shows that nineteen of the proposed studies focus on



student employability and performance prediction, and only one study worked on candidate employability prediction in general.

3. Context awareness: with respect to the context awareness criterion, none of the surveyed studies addressed contextual aspects or supported employability prediction based on context-related knowledge.

Study and authors	Year	Goal	Domain	Context awareness
Hugo [19]	2018	Predict the students' career outcomes.	Students	Not supported
Othman et al. [20]	2018	Identify the factors that influence graduates employability.	Students	Not supported
Nunley et al. [21]	2016	Estimate the impact of college majors and internship experience on employment prospects.	Students	Not supported
Kommina et al. [43]	2020	Predict the students' academic performance and employability chances.	Students	Not supported
Kalpana and Venkatalakshmi [44]	2014	Analyze the students' performances.	Students	Not supported
Casuat et al. [45]	2020	Predict the students' employability.	Students	Not supported
Patel et al. [46]	2020	Predict the suitable students' job domains.	Students	Not supported
Casuat et al. [47]	2020	Identify the most predictive attributes among employability signals.	Students	Not supported
Bai and Hira. [48]	2021	Predict the students' employability.	Students	Not supported
Dubey and Mani [49]	2019	Predict the employability of high school students with local businesses for part-time jobs.	Students	Not supported
Pinto [50]	2019	Analyze the influence of academic performance and extracurricular activities on the perceived employability of students.	Students	Not supported
Jantawan and Tsai [51]	2013	Predict students' employability	Students	Not supported
Giri et al. [52]	2016	Predict the probability of an undergraduate student getting placed in an IT company.	Students	Not supported
Osmanbegovic and Suljic [53]	2012	Predict the students' success.	Students	Not supported
Laddha [54]	2021	Predict students' employability based on Technical Skills.	Students	Not supported
Kumar and Babu [55]	2019	Predict the students' employability.	Students	Not supported
Reddy et al. [56]	2021	Predict joining efficient candidates.	Employees	Not supported
Aviso et al. [57]	2020	Predict the employability of chemical engineering graduates based on UK university rankings.	Students	Not supported
Maheswari [58]	2020	Predict the student's performance in placement.	Students	Not supported
Almutairi [59]	2018	Predict the suitability of Information Systems' graduates.	Students	Not supported

Table 2.1. Related works: Goal, domain and context awareness.

**D. FEATURES:** This subsection analyses the surveyed studies according to the following criteria: Adopted features, Output features, Sensitive features and Techniques used for determining the sensitive features. Features are identified as factors influencing the success of internship programs. We distinguished three categories of features as follows.

**1. Adopted features:** we classified the different adopted features into six categories: Hard skills, Soft skills, Demographic features, Extracurricular/co-curricular activities, Professional experience and Internship represents the retained works according to these categories. Most of the studies focus mainly on Grade Point Average (GPA) assignments and exam's student performances on technical, computational and analytical courses. Kommina et al., Casuat et al., Bai and Hira, Giri et al. and Kumar and Babu focus on soft skills. Personal and socio-demographic variables are adopted features. University related features are adopted in. Only a minority of studies support extracurricular and curricular activities. In the same vein, a few studies supported internship related features such as experience in internships or the number of

internships. Similarly, professional experience, such as “Number of companies worked previously” is supported.

2. **Output features:** Eleven of twenty studies focused on placement employability hiring recruitment getting a job and working. Rare are those studies that concentrated on employability rate, company or graduation. Moreover, focused on predicting the student matching to skills required by the Saudi industry.

- a. **Sensitive features:** are the selected features. Most of the studies did not identify the most sensitive features. Internship is considered as sensitive in only three studies. In addition, internship is considered in as the most sensitive variable, followed by specific majors and cocurricular activities. Extracurricular activities are considered in as a sensitive feature. Moreover, mental alertness, manner of speaking, ability to express ideas and self-confidence are sensitive features, the most predictive features are as follows: aptitude and reasoning skills, communication skills, family income status, mentor and quality of teaching in the college.
- b. **Techniques for determining sensitive features:** a number of techniques are used to determine the sensitive features such as Univariate Feature Selection technique, Recursive Feature Elimination technique, and Principal component Analysis technique, Logistic regression (LR) - P-values, Pearson correlation method and Kandel correlation method, and WEKA feature selection.

## **D. ML MODEL, DATA SOURCES AND PERFORMANCE**

### **EVALUATION METRICS:**

This subsection analyses the surveyed studies according to the criteria ML model, Data sources and Performance evaluation metrics, in accordance with the research questions posed.

1. **Machine Learning Model:** to predict employability of students or employees, some researchers used traditional approaches (i.e., statistical sampling, surveys) to predict employability rates [19]. In contrast, the majority employed ML algorithms as they show their high prediction performances. Indeed, we notice that approximately 95% of related works implemented or/and sometimes went even further to compare supervised ML models. Limited studies have implemented unsupervised ML models; the authors of proposed a hybrid model of a deep belief network and soft max regression (DBN-SR).

2. **Dataset Sources:** three types of dataset sources are used in the studies examined:
  - Database: most studies exploited databases from registration units or/and institutional departments
3. **Surveys:** surveys of students/employees and employers,
4. **Databases and surveys:** We notice that few studies employed resumes regardless of the difficulty of analyzing résumés and the scarcity of real resumes.
5. **Performance Evaluation Metrics:** most retained articles used one or many performance evaluation metrics to compare ML algorithms (i.e., precision, recall, FI score, accuracy, etc.). Only did not apply any metric.

**E. DISCUSSION ON RESEARCH QUESTIONS:** The previous comparative analysis provides evidence of a number of commonalities and limitations of the above studies. First, with most of the studies, the output features are binary and allow predicting whether the student will have a job. The development of methods that offer multiple output values in predicting employability may prove most useful. However, we observe a lack of research supporting this possibility. With respect to this limitation, our work supports the prediction of employment status, i.e., whether the student will be employed, unemployed, continue studies, or go on training. Second, none of the studies consider context knowledge in the prediction process. We also observe a lack of studies that focus on the contextual information that could have an impact on the prediction of the output features. The studies do not address the question of which contextual information can offer a prediction with better performance with respect to accuracy, precision, recall and F1score. In other words, the relationship between context and prediction has been neglected thus far by previous studies. This observation calls for the development of research that considers contextual information during the prediction process. We propose in this work to model and to include the context related to the student and to the internship in the prediction process.

### **3.PROPOSED WORK**

#### **Introduction:**

The proposed study aims to develop a context-aware approach to predict student employability based on their internship program and contextual knowledge. This is an important area of research as internships are becoming increasingly important for enhancing graduates' employability, and universities need to ensure that their internship programs are effective and successful.

The study will use Gradient Boosting Models to predict student employability, which is a machine learning technique that has been shown to be effective in predictive modelling. This technique will be used to identify the most predictive features of employability based on student and internship contexts, such as the type and duration of the internship, the industry or sector in which the internship was completed, and the skills and knowledge gained during the internship.

It has several contributions, including a flexible context-aware employability prediction process model, a context model for employability prediction, and a context-based approach for student employability prediction. These contributions are expected to provide a better understanding of the factors that influence student employability and help universities to design more effective and successful internship programs.

The output of the study will be the student employability status, which will identify whether the student is most likely to be employed or unemployed based on their internship program and contextual knowledge. This information will be useful for universities, employers, and students themselves, as it will help to identify areas for improvement and guide career development and decision-making.

Finally, we can say that the goal of this approach is to provide a reliable and accurate prediction of a student's employability after completing an internship, which can help both the student and the employer make informed decisions. This approach can also help universities and colleges better understand the impact of their internship programs and make improvements to better prepare their students for the job market.

### **3.1 SYSTEM REQUIREMENTS**

System Requirements are the configuration that a system must have in order for a hardware or software application to run smoothly and efficiently. Failure to meet these requirements can result in installation or performance problems.

The below sections will describe what is required to run our detection system smoothly without any problems.

#### **3.1.1. SOFTWARE REQUIREMENTS**

Software Requirements refer to the software required to run the detection system. Our detection system requires the following software installed:

- Jupyter Notebook/Google Collab
- Python 3 and its libraries (Pandas, NumPy)

#### **3.1.2. HARDWARE REQUIREMENTS**

Hardware Requirements refer to the minimum hardware qualifications needed to run the detection system smoothly. Our detection system requires the following specifications:

- Intel i5 Processor or better version
- 8+ GB RAM
- Windows OS recommended, but can use Linux/MacOS

#### **3.1.3. USER REQUIREMENTS**

The User Requirements refer to the proposal requested by the client or the user and refer to what are the features they require or would prefer. The following are the User Requirements needed:

- A Prediction System for Student Employability
- Must include the features of both student and internship contexts.

### 3.2 METHODOLOGY

The contextual information characterizing the student profile and the internship is of great interest in the prediction process. This fact requires identifying solutions that allow context-based prediction. This study presents a new approach for predicting graduate employability status, based on contextual information related to the student profile and the internship. A large number of attributes and features are used by ML-based algorithms in this regard.

To enhance the functionality and the performance of the latter, a context aware ML-based model is defined. In general, a context-aware system supports the idea of using available information for specific user needs to improve the behaviour of the system itself. We extend this idea to this work: the proposed ML model considers the student profile and the internship conditions. For this purpose, a

“context-aware feature selection” step is added to the standard ML process to decide which relevant and sensitive features (data) to use for learning and testing the model. Even better, we think that representing the different alternatives for achieving the rest of the steps (i.e., data collecting, feature engineering, model training, model evaluating and model employing) is of great help to follow the process fulfillment (i.e., intentions) and allow flexibility by using alternatives.

The distinction between what to achieve (i.e., the goal) and the way to achieve it (i.e., the strategy) to show the different ways allowed to move from one step to another during a standard ML prediction process. For example, we show that going from “Data collection” step to “Data pre-processing” step, many techniques are generally allowed, namely, ‘by eliminating redundant data’, ‘by data imputation’, ‘by dataset balancing’, etc. The same idea is in moving from building the model to evaluating it, where the set of evaluating metrics available for the data analyst are highlighted (i.e., precision metric, recall metric, accuracy metric, etc.).

The study emphasizes the importance of distinguishing between the goal of the prediction process and the strategies used to achieve it, which allows for flexibility in the different steps of the standard machine learning prediction process. The study also highlights the various techniques available for data collection, pre-processing, model building, and evaluation, providing data analysts with a set of evaluating metrics to measure the performance of the model.

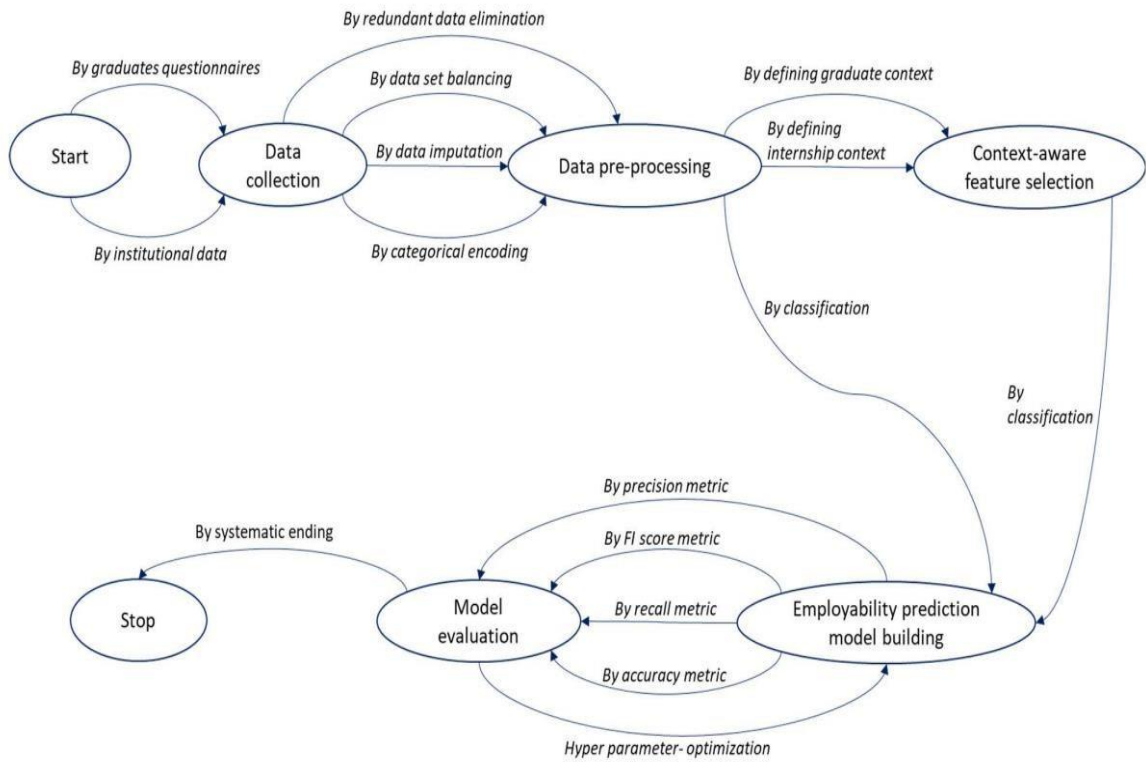


Figure 3.2.1 Context aware employability prediction process (Architecture)

### 3.2.1 TRAINING MODEL

It is worth saying that building a highly reliable prediction model that has the ability to determine the employability status based on an effective feature set, is very challenging. This figure represents the prediction model selection process.

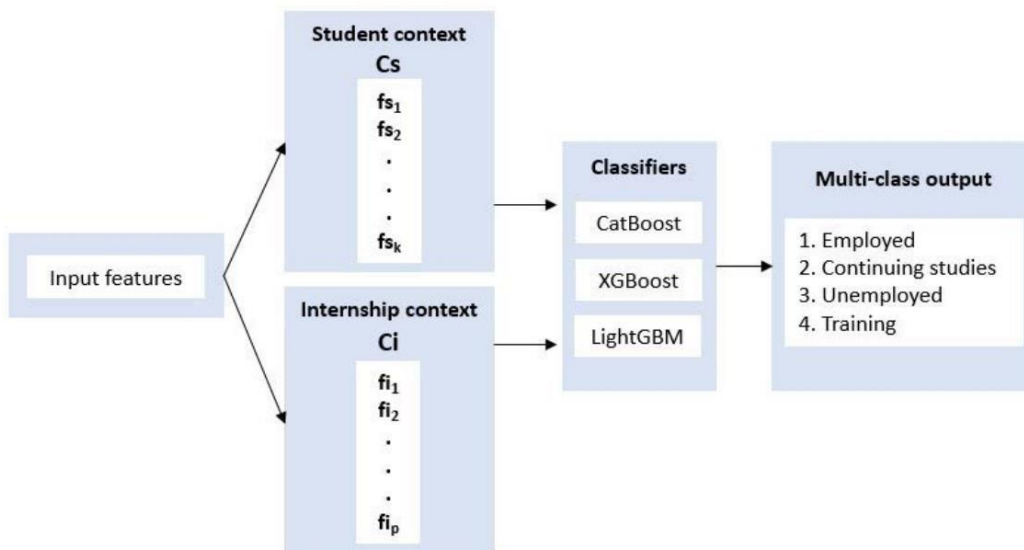


Figure 3.2.2 Training model selection process

This training model represents the exact flow of our project, here firstly we will be taking the input features from our dataset, our dataset is included with student content features and internship context features. After extracting those features we will be combining the student and internship features to get the over all performance of the student and we will be predicting the output with help of the gradient boosting classifiers they are CatBoost, XGBoost, LightGBM. After classification we will be getting a multi-class output with 4 columns, they are employed, continuing studies, unemployed and training.

From the above training model, our main motto is to get the information about the student whether he is getting employed or unemployed. So we are only focusing on employed or unemployed and not on continuing studies or still training.

## CONTEXT MODEL

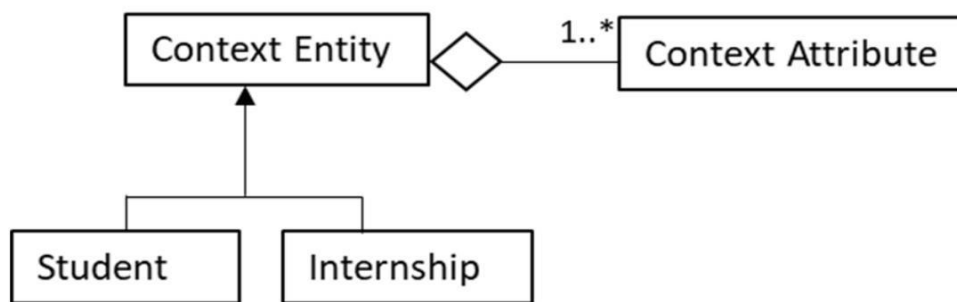


Figure 3.2.3 Context model for employability prediction.

Given a context  $C$  in  $\{C_s, C_i\}$  where:

- $C_s$  is the student context
- $C_i$  is the internship context,
- $FC_s = \{fs_1, \dots, fs_k\}$  is the set of features related to  $C_s$  where  $fs_1$  is the first feature in  $FC_s$ ,  $k$  is the total number of features related to  $FC_s$ , and  $FC_s$  is a part from  $F$  the set of all  $n$  features
- $FC_i = \{fi_1, \dots, fi_p\}$  is the set of features related to  $C_i$  where  $fi_1$  is the first feature in  $FC_i$  and  $fi_p$  is the  $p$ th feature in  $FC_i$ ,  $p$  is the total number of features related to  $FC_i$ , and  $FC_i$  is the set of all  $n$  features from  $F$ .

Our objective is to select the most suitable prediction model that achieves the best performance to our target output: employment status ( $Emp\_Status$ ). This output is a categorical feature that contains 4 classes. For this reason, our task is multiclass classification. Therefore, we use Three



Boosting classifiers - XGBoost, LightGBM and CatBoost to evaluate their performance in predicting employment status.

### **3.2.2 GRADIENT BOOSTING ALGORITHMS:**

Gradient boosting models are a type of machine learning algorithm that are used for both regression and classification tasks. These models are based on the concept of decision trees, where a set of simple decision trees are combined to make a more complex model.

Compared to ordinary models, gradient boosting models have several advantages. Firstly, they are more accurate than traditional models because they can handle non-linear relationships and interactions between features. Secondly, they are more robust to overfitting, which is a common problem with traditional models. Finally, they are able to handle missing data and outliers more effectively than traditional models.

#### **A) CATBOOST ALGORITHM:**

Catboost is an open-source machine learning algorithm which is designed to handle categorical variables in data and is particularly effective in dealing with high-cardinality categorical variables.

Some of the key features of the Catboost algorithm are:

- **Handling categorical variables:** The algorithm can automatically handle categorical variables in the data, without requiring explicit encoding or one-hot encoding. This feature saves time and reduces the risk of data leakage.
- **Fast training and prediction:** Catboost is designed to be highly scalable, making it suitable for large datasets. It also offers fast training and prediction times, thanks to its use of gradient-boosting algorithms.
- **Automatic tuning of hyperparameters:** The algorithm uses a method called Bayesian hyperparameter optimization to automatically tune its hyperparameters, resulting in better performance and reducing the need for manual tuning.
- **Robustness to noisy data:** Catboost can handle noisy data well, thanks to its use of robust loss functions and gradient-based regularization.

## **B) XGBOOST ALGORITHM:**

XGBoost (Extreme Gradient Boosting) is a popular open-source gradient boosting algorithm that was designed to optimize performance and speed in building machine learning models by combining the strengths of gradient boosting algorithms and decision trees.

The algorithm works by iteratively adding decision trees to an ensemble, where each tree attempts to correct the mistakes of the previous tree. XGBoost uses a technique called gradient boosting, where the algorithm calculates the gradient of the loss function for each data point in the training set, and then builds a tree to minimize the loss function based on the gradient.

Some of the key features of XGBoost include:

- **Regularization:** XGBoost supports both L1 and L2 regularization to prevent overfitting.
- **Cross-validation:** XGBoost supports built-in cross-validation to select the optimal number of trees and tree depth.
- **Handling missing values:** XGBoost can handle missing values in the data, by automatically learning the best direction to go when a value is missing.
- **Parallel processing:** XGBoost supports parallel processing on a single machine or across a cluster of machines.

## **C) LGBM ALGORITHM:**

LGBM (Light Gradient Boosting Machine) is a gradient boosting framework that uses tree-based learning algorithms. It is designed to be a fast, efficient, and distributed algorithm that can scale to handle large datasets. LGBM uses the histogram-based algorithm to split data in a tree, which can reduce the computational complexity of finding the best split point.

Some key features of LGBM algorithm include:

- **Speed:** LGBM is designed to be much faster than other gradient boosting algorithms.
- **Memory Efficiency:** LGBM is memory-efficient and can handle large datasets without running out of memory.

- Improved accuracy: LGBM uses histogram-based algorithms, which can result in better accuracy than other gradient boosting algorithms.
- It is used for a wide range of tasks, including classification, regression, and ranking. It is particularly useful for handling large datasets and is often used in applications such as search engines, recommendation systems, and financial modelling.

### 3.2.3 PSEUDOCODE:

>>Importing the required libraries from python and model libraries

```
import pandas as pd
```

```
import numpy as np
```

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.metrics import accuracy_score
```

```
from sklearn.metrics import precision_score, recall_score, f1_score
```

```
from catboost import CatBoostClassifier
```

```
import xgboost as xgb
```

```
import lightgbm as lgbm
```

```
from tabulate import tabulate
```

```
import matplotlib.pyplot as plt
```

>>loading of dataset

```
# Load the dataset
df = pd.read_csv('Book1.csv')
```

## >>Pre-processing of dataset

```
# Drop any rows with missing values
df.dropna(inplace=True)

# Remove any rows where the Internship column has a value less than 0 or greater than 100
df = df[(df['Internship%'] >= 0) & (df['Internship%'] <= 100)]

# Remove any rows where the CGPA column has a value less than 0 or greater than 10
df = df[(df['CGPA'] >= 0) & (df['CGPA'] <= 10)]

# Remove any rows where the LeadershipSkills, CommunicationSkills, or TeamWork columns have a value less than 0 or greater
df = df[(df['LeadershipSkills'] >= 0) & (df['LeadershipSkills'] <= 10)]
df = df[(df['CommunicationSkills'] >= 0) & (df['CommunicationSkills'] <= 10)]
df = df[(df['TeamWork'] >= 0) & (df['TeamWork'] <= 10)]

# Remove any rows where the Overall% column has a value less than 0 or greater than 100
df = df[(df['Overall%'] >= 0) & (df['Overall%'] <= 100)]
```

## >> Categorical Encoding to the dataset

### >> Label Encoding and One Hot Encoding to the Specialized and Domain Columns

```
label_encoder = LabelEncoder()
df['Specialized'] = label_encoder.fit_transform(df['Specialized'])

one_hot_encoder = OneHotEncoder()
one_hot_encoded = one_hot_encoder.fit_transform(df[['Domain']])
df = df.join(pd.DataFrame(one_hot_encoded.toarray(), columns=one_hot_encoder.get_feature_names_out(['Domain'])))

# Drop the original 'Domain' column
df.drop('Domain', axis=1, inplace=True)

# Rename the one-hot encoded columns to remove the prefix 'Domain_'
df.rename(columns=lambda x: x.replace('Domain_', ''), inplace=True)
```

## >>splitting up of datasets to training and testing:

```
X = df[['Internship%', 'CGPA', 'Specialized', 'LeadershipSkills', 'CommunicationSkills', 'TeamWork', 'Overall%']]
y = df['Result']

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

## IMPLEMENTATION OF CATBOOST ALGORITHM

### >> reading of dataset is common for every model

### >>splitting into train and test datasets

### >> Loading pre-processed dataset into new file

```
# Save the preprocessed dataset to a file
df.to_csv('preprocessed_Book1.csv')
df.head()
```

>> The below code trains the CatBoost classifier model on the training data with 100 iterations and with the learning rate of 0.5.

>> The evaluation of the model is done by the metrics accuracy, precision, recall and F1-score.

```
from catboost import CatBoostClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
catboost_model = CatBoostClassifier(iterations=100, learning_rate=0.5)
catboost_model.fit(X_train, y_train, cat_features=[2, 3])
y_pred = catboost_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1score = f1_score(y_test, y_pred)
```

>> Create Data frame as user input

```
user_df = pd.DataFrame({
    'Internship%': [internship_percent],
    'CGPA': [cgpa],
    'Specialized': [specialization_encoded],
    'LeadershipSkills': [leadership_skills],
    'CommunicationSkills': [communication_skills],
    'TeamWork': [teamwork],
    'Overall%':[overall]
})
```

>> Prediction through the CatBoost model

```
catboost_model.load_model('catboost_model.cbm')

# Make the prediction using the CatBoost classifier
prediction = catboost_model.predict(user_df)

if prediction[0] == 1:
    print("Congratulations! You are likely to be hired.")
else:
    print("Sorry, you are not likely to be hired.")
```

## IMPLEMENTATION OF XGBOOST ALGORITHM:

>> reading of dataset is common for every model

>> splitting into train and test datasets

>> Create XGBoost Classifier and fit into the training set

```
xgboost = xgb.XGBClassifier(objective='binary:logistic', max_depth=5, learning_rate=0.5)
xgboost.fit(X_train, y_train)
```

>> Make predictions on the testing set and calculate evaluation metrics

```
y_pred = xgboost.predict(X_test)
accuracy_xgb = accuracy_score(y_test, y_pred)
precision_xgb = precision_score(y_test, y_pred)
recall_xgb = recall_score(y_test, y_pred)
f1score_xgb = f1_score(y_test, y_pred)
```

>> Make the prediction using XGBoost Classifier

```
prediction = xgboost.predict(user_df)

if prediction[0] == 1:
    print("Congratulations! You are likely to be hired.")
else:
    print("Sorry, you are not likely to be hired.")
```

## IMPLEMENTATION OF LGBM CLASSIFIER

>> reading of dataset is common for every model

>> splitting into train and test datasets

>> Create LGBM Classifier and fit into the training set

```
lgbm_classifier = lgbm.LGBMClassifier(objective='binary', max_depth=5, learning_rate=0.2)
lgbm_classifier.fit(X_train, y_train)
```

>> Make predictions on the testing set and calculate evaluation metrics

```
y_pred = lgbm_classifier.predict(X_test)
accuracy_lgbm = accuracy_score(y_test, y_pred)
precision_lgbm = precision_score(y_test, y_pred)
recall_lgbm = recall_score(y_test, y_pred)
f1score_lgbm = f1_score(y_test, y_pred)
```

>> Make the prediction using LGBM Classifier

```
prediction = lgbm_classifier.predict(user_df)

if prediction[0] == 1:
    print("Congratulations! You are likely to be hired.")
else:
    print("Sorry, you are not likely to be hired.")
```

## 4. EXPERIMENTAL STUDY

### 4.1 DATASET:

We have collected the dataset from our college website for this project and we have added some of our additional features which is necessary to include the student context details and internship context details after adding some of these details like CGPA, Communication skills, Leadership skills, Team work etc. We have summed up all the student's performance details and internship performance details into one and we named it as overall performance of the student. This is how our dataset is done.

For the additional features, we also included Specialized in and Domain given in

Internship such that we can predict the student's ability of how fast they can learn in different domains and how productive they are.

1	Organization	Days	Internship	CGPA	Specialized	Domain	Leadership	Communication	TeamWork	Overall%	Result
2	5 GENIES	75	78	8	Java	Python	6	7	7	71.8	1
3	5 GENIES	75	81	9.3	C++	Python	7	6	7	74.8	1
4	AAASPAAS	47	78	8.5	Python	SQL	6	9	10	82.6	1
5	ACCOLITE	110	84	8.9	Python	Python	7	9	10	86.6	1
6	ACCOLITE	110	78	9.1	Java	Python	6	6	6	69.8	0
7	ADOBE	67	73	8.9	Java	Java	7	7	8	76.4	1
8	Adrin	74	67	6.8	C++	Python	6	5	8	65	0
9	Adrin	74	74	6.6	Python	Python	5	6	8	68.2	0
10	Adrin	74	56	6.7	C	Python	6	8	4	60.6	0
11	Adrin	74	56	6.4	C	Python	5	9	8	68	0
12	Adrin	74	65	6.9	C	Python	5	7	7	64.8	0
13	Adrin	74	67	7.1	Python	Python	6	6	8	67.6	0
14	Adrin	74	93	7.6	C++	Python	7	7	7	75.8	1
15	Adrin	74	86	7.2	Python	Python	6	9	8	77.6	1
16	ADRIN	74	95	9.3	Java	Python	6	6	5	71.6	1
17	Adrin	74	97	8.1	Python	Python	10	6	10	87.6	1
18	ADTRAN	79	80	8.9	C++	C++	9	10	7	85.8	1
19	ADTRAN	79	87	8.2	Python	C++	7	7	7	75.8	1
20	ADTRAN	79	69	8.5	Python	C++	6	6	7	68.8	0
21	AIR	67	91	9.1	Java	Web Deve	5	8	9	80.4	1
22	AMAZON	149	92	9.7	Java	Java	9	8	10	91.8	1
23	AMAZON	149	95	9.3	Python	Java	9	10	5	85.6	1
24	AT&T	47	78	8.9	Java	Java	4	5	6	65.2	0
25	ATLASSIAN	78	67	7.7	Python	Java	8	9	8	78.8	0
26	CALLIDUS	43	84	8.7	Java	SQL	5	8	9	78.2	1
27	CAPGEMIN	54	59	6.8	C	Python	5	8	8	67.4	0
28	CAPITAL IC	55	70	8.1	C	Python	6	9	9	78.2	1
29	Cisco	68	78	8.8	SQL	Networkin	5	7	5	67.2	0
30	CISCO	68	98	9.1	Python	Networkin	6	7	10	83.8	1
31	CISCO	68	76	9.4	SQL	Networkin	5	7	8	74	0
32	CISCO	68	75	8.9	Python	Networkin	6	8	6	72.8	0
33	CISCO	68	98	9.8	Java	Networkin	10	9	10	97.2	1

Table 4.1.1 Dataset



The above is the dataset model, which we have been using for our project, which includes the information about the student's internship and the student's context features or co-curricular activities. The student's context features included are CGPA, Specialized In, Leadership Skills, Communication Skills, Team Work.

After combining the internship details and additional features which we mentioned before, we finally get the overall performance of the student, based on that we will be predicting the student's employability through the gradient boosting models.

## 4.1.1 DATA PREPROCESSING

### A) DATA ELIMINATION:

Data elimination in datasets refers to the process of removing certain data points or variables from a dataset. The reason for data elimination can vary based on the specific goals and requirements of the analysis.

Some common reasons for data elimination in datasets include:

1. Missing data: If a dataset contains missing data points, they may need to be removed in order to prevent the missing values from biasing the results.
2. Irrelevant variables: Variables that are not relevant to the analysis or do not have a significant impact on the outcome may be removed to simplify the analysis and improve accuracy.
3. Duplicate data: Duplicates data points may be removed to avoid bias and improve the accuracy of the analysis.
4. Low-quality data: Data that is of low quality or contains errors may need to be removed to prevent it from biasing the results.

```
# Load the dataset
df = pd.read_csv('Book1.csv')

# Drop any rows with missing values
df.dropna(inplace=True)

# Convert the Days column to integer type
df['Days'] = df['Days'].astype(int)
# Remove any rows where the Internship column has a value less than 0 or greater than 100
df = df[(df['Internship%'] >= 0) & (df['Internship%'] <= 100)]

# Remove any rows where the CGPA column has a value less than 0 or greater than 10
df = df[(df['CGPA'] >= 0) & (df['CGPA'] <= 10)]

# Remove any rows where the LeadershipSkills, CommunicationSkills, or TeamWork columns have a value less than 0 or greater than 100
df = df[(df['LeadershipSkills'] >= 0) & (df['LeadershipSkills'] <= 100)]
df = df[(df['CommunicationSkills'] >= 0) & (df['CommunicationSkills'] <= 100)]
df = df[(df['TeamWork'] >= 0) & (df['TeamWork'] <= 100)]
|
# Remove any rows where the Overall% column has a value less than 0 or greater than 100
df = df[(df['Overall%'] >= 0) & (df['Overall%'] <= 100)]
df.head(35)
```

	Organization	Days	Internship%	CGPA	Specialized	Domain	LeadershipSkills	CommunicationSkills	TeamWork	Overall%	Result
0	5 GENIES	75	78	8.0	Java	Python	6	7	7	71.8	1
1	5 GENIES	75	81	9.3	C++	Python	7	6	7	74.8	1
2	AAASPAAS ONLINE SERVICES	47	78	8.5	Python	SQL	6	9	10	82.6	1
3	ACCOLITE	110	84	8.9	Python	Python	7	9	10	86.6	1
4	ACCOLITE	110	78	9.1	Java	Python	6	6	6	69.8	0
5	ADOBE	67	73	8.9	Java	Java	7	7	8	76.4	1
6	Adrin	74	67	6.8	C++	Python	6	5	8	65.0	0
7	Adrin	74	74	6.6	Python	Python	5	6	8	68.2	0
8	Adrin	74	56	6.7	C	Python	6	8	4	60.6	0
9	Adrin	74	56	6.4	C	Python	5	9	8	68.0	0
10	Adrin	74	65	6.9	C	Python	5	7	7	64.8	0

Figure 4.1.1 Dataset after data elimination

## B) CATEGORICAL ENCODING:

Categorical encoding is an important step in data pre-processing because many machine learning algorithms cannot handle categorical variables directly. By converting categorical variables into numerical representations, we can use them in our models and extract valuable insights from our data.

Categorical variables are those that take on a limited, predefined set of values, such as colours, categories, or types.

There are several methods for categorical encoding, mostly used are:

1. One-hot encoding: This method creates a binary column for each category, where a value of 1 indicates the presence of the category and 0 indicates its absence.
2. Label encoding: This method assigns a unique numerical label to each category. For example, if we have three categories "red", "blue", and "green", they could be assigned the labels 0, 1, and 2.

In our project, we have been done the data elimination, categorical encoding and balanced dataset in the data pre-processing step. While the categorical encoding is done, label encoding is applied towards the “Specialized In” column and one-hot encoding is applied towards the “Domain” column.

```

from sklearn.preprocessing import LabelEncoder, OneHotEncoder
label_encoder = LabelEncoder()
df['Specialized'] = label_encoder.fit_transform(df['Specialized'])
# One-hot encode the 'Domain' column
one_hot_encoder = OneHotEncoder()
one_hot_encoded = one_hot_encoder.fit_transform(df[['Domain']])
df = df.join(pd.DataFrame(one_hot_encoded.toarray(), columns=one_hot_encoder.get_feature_names_out(['Domain'])))

# Drop the original 'Domain' column
df.drop('Domain', axis=1, inplace=True)

# Rename the one-hot encoded columns to remove the prefix 'Domain_'
df.rename(columns=lambda x: x.replace('Domain_', ''), inplace=True)

# Display the preprocessed dataset
print(df.head())

```

	Organization	Days	Internship%	CGPA	Specialized	\
0	5 GENIES	75	78	8.0	2	
1	5 GENIES	75	81	9.3	1	
2	AAASPAAS ONLINE SERVICES	47	78	8.5	3	
3	ACCOLITE	110	84	8.9	3	
4	ACCOLITE	110	78	9.1	2	

	LeadershipSkills	CommunicationSkills	TeamWork	Overall%	Result	C++	\
0	6	7	7	71.8	1	0.0	
1	7	6	7	74.8	1	0.0	
2	6	9	10	82.6	1	0.0	
3	7	9	10	86.6	1	0.0	
4	6	6	6	69.8	0	0.0	

	Java	Java, C++	Java, SQL	Networking	Python	Python, SQL	SQL	\
0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	1.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
3	0.0	0.0	0.0	0.0	1.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	1.0	0.0	0.0	

	SQL, Java	Web Development
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

Figure 4.1.2 Dataset after categorical Encoding

## 4.1.2 PARAMETER SETTING

Setting appropriate parameters is essential for achieving good performance and accurate predictions. Parameters can be set for different components of a machine learning pipeline, including data preprocessing, feature selection, model selection, and hyperparameter tuning.

Hyperparameter tuning involves searching for the optimal combination of hyperparameters for a given model. Hyperparameters control the behavior of the model during training and can include learning rates, regularization parameters, and the number of iterations.

Parameter settings for the CatBoost, XGBoost, and LightGBM algorithms:

- `learning_rate`: The learning rate determines the step size at each iteration while moving toward a minimum of a loss function.
- `depth`: The depth parameter determines the depth of the trees in the ensemble.
- `iterations`: The number of iterations to run the training process.
- `max_depth`: The maximum depth of a tree.
- `Objective`: This parameter is used to define the loss function to be minimized during the training process. In Binary Classification problems, the objective parameter should be set to 'binary:logistic' which indicates that the binary cross-entropy loss function is being minimized.

In our project, we have used all these parameters for getting the better evaluation, accuracy and good performance scores in predicting our valid output. In the CatBoost algorithm the parameters we used are `learning_rate`, `iterations`. And in the XGBoost and LightGBM algorithms the parameters we used are `max_depth`, `objective` and `learning_rate`.

Based on the above parameters setting we have got the accurate predictions and the valid outputs for the project.

## 4.2 RESULT & TEST ANALYSIS:

### 4.2.1 OUTPUTS FROM THE GRADIENT BOOSTING MODELS

**Outputs from the CatBoost Algorithm:**

```
Accuracy: 0.7941176470588235
Precision: 0.8571428571428571
Recall: 0.8181818181818182
F1-score: 0.8372093023255814
```

Figure 4.2.1 Performance of Catboost model

```
File Edit View Insert Cell Kernel Widgets Help
Enter your internship percentage: 78
Enter your CGPA: 8.9
Enter your specialization: 5
Enter your leadership skills score (out of 10): 8
Enter your communication skills score (out of 10): 9
Enter your teamwork score (out of 10): 7
Congratulations! You are likely to be hired.
```

Figure 4.2.2 Output 1 from Catboost model

```
File Edit View Insert Cell Kernel Widgets Help
Enter your internship percentage: 67
Enter your CGPA: 7.8
Enter your specialization: 5
Enter your leadership skills score (out of 10): 6
Enter your communication skills score (out of 10): 7
Enter your teamwork score (out of 10): 6
Sorry, you are not likely to be hired.
```

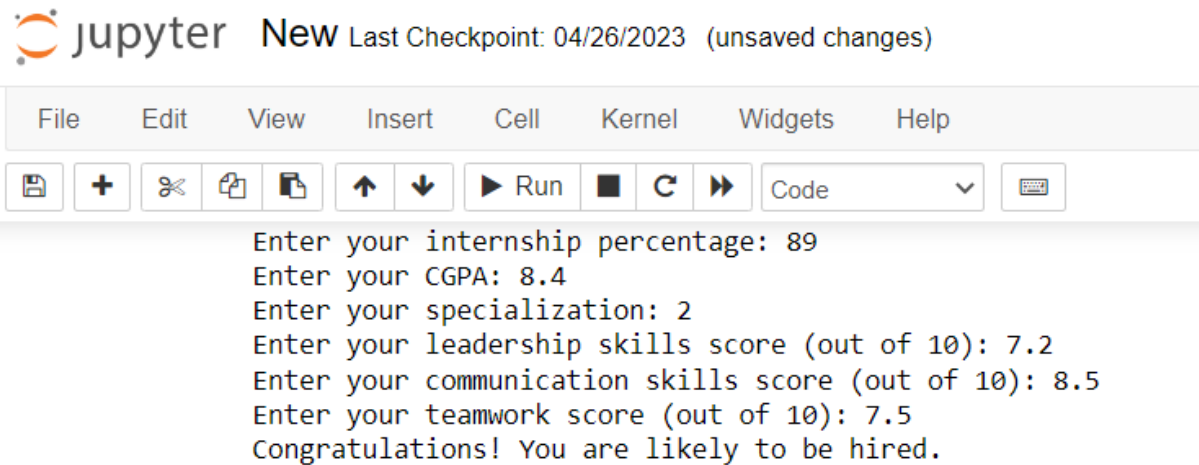
Figure 4.2.3 Output 2 from Catboost model

From above results, we acquired a good accuracy from the catboost algorithm and also better scores for precision, recall and f1-scores. As we discussed before, we took the iterations about 100 and learning rate about 0.5 in calculating the scores from the catboost algorithm.

#### Outputs from the XGBoost Algorithm:

```
Accuracy: 0.7647058823529411
Precision: 0.8181818181818182
Recall: 0.8181818181818182
F1-score: 0.8181818181818182
```

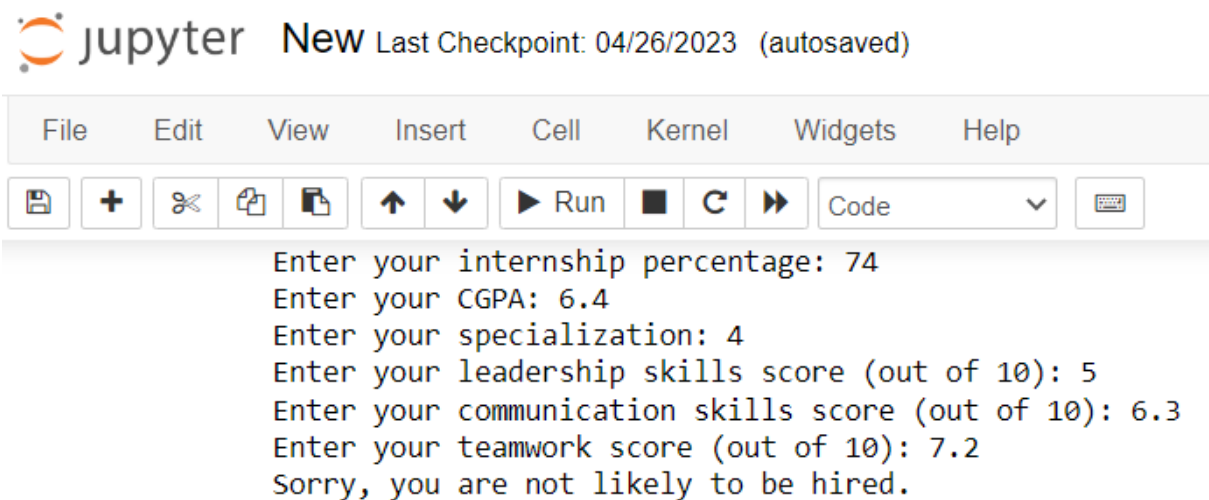
Figure 4.2.4 Performance of XGBoost model



The image shows a Jupyter Notebook interface. At the top, it says "jupyter New" and "Last Checkpoint: 04/26/2023 (unsaved changes)". Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Under the menu bar is a toolbar with icons for saving, adding cells, deleting, copying, pasting, undo, redo, and running code. The code cell contains the following text:

```
Enter your internship percentage: 89
Enter your CGPA: 8.4
Enter your specialization: 2
Enter your leadership skills score (out of 10): 7.2
Enter your communication skills score (out of 10): 8.5
Enter your teamwork score (out of 10): 7.5
Congratulations! You are likely to be hired.
```

Figure 4.2.5 Output 1 from XGBoost model



The image shows a Jupyter Notebook interface. At the top, it says "jupyter New" and "Last Checkpoint: 04/26/2023 (autosaved)". Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Under the menu bar is a toolbar with icons for saving, adding cells, deleting, copying, pasting, undo, redo, and running code. The code cell contains the following text:

```
Enter your internship percentage: 74
Enter your CGPA: 6.4
Enter your specialization: 4
Enter your leadership skills score (out of 10): 5
Enter your communication skills score (out of 10): 6.3
Enter your teamwork score (out of 10): 7.2
Sorry, you are not likely to be hired.
```

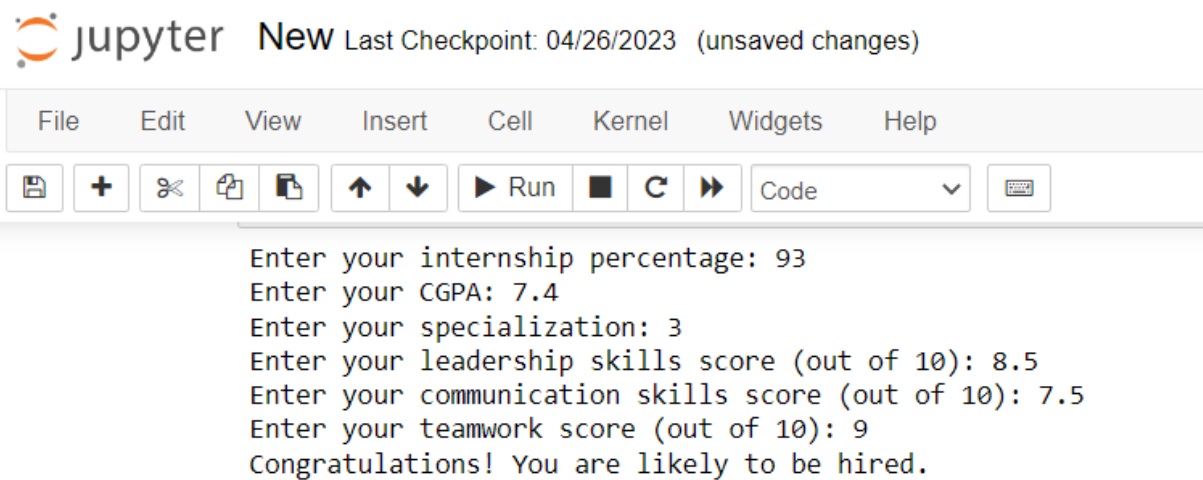
Figure 4.2.6 Output 2 from XGBoost model

In the above results, we can say that the accuracy score from the xgboost algorithm is less compared to catboost and other metrics like precision, recall and f1-score are equal with each other.

#### **Outputs from the LighGBM Algorithm**

```
Accuracy: 0.7647058823529411
Precision: 0.85
Recall: 0.7727272727272727
F1-score: 0.8095238095238095
```

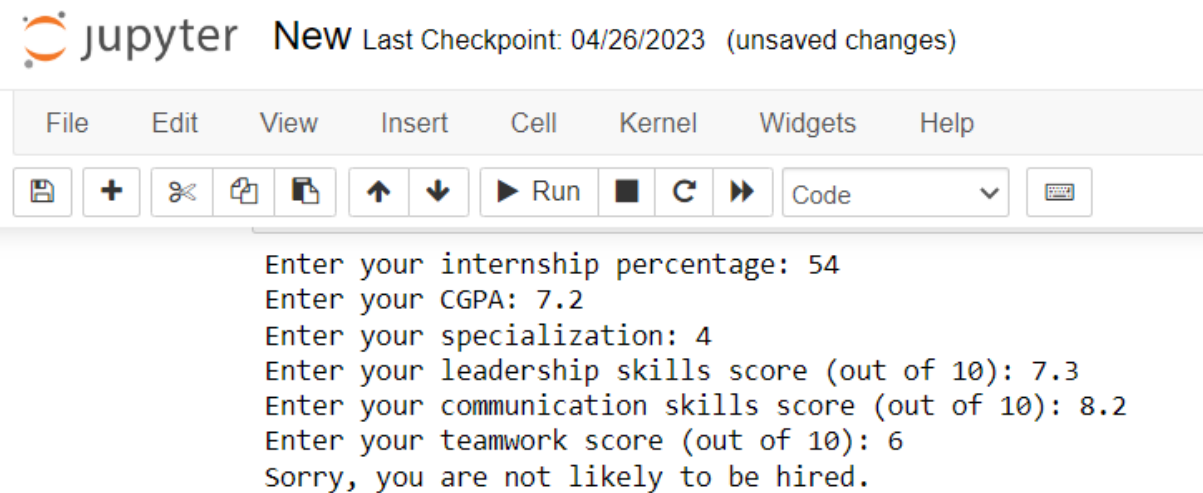
Figure 4.2.7 Performance of LightGBM model



The image shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for saving, adding, deleting, copying, pasting, and running code. The code cell contains the following text:

```
Enter your internship percentage: 93
Enter your CGPA: 7.4
Enter your specialization: 3
Enter your leadership skills score (out of 10): 8.5
Enter your communication skills score (out of 10): 7.5
Enter your teamwork score (out of 10): 9
Congratulations! You are likely to be hired.
```

Figure 4.2.8 Output 1 from LightGBM model



The image shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for saving, adding, deleting, copying, pasting, and running code. The code cell contains the following text:

```
Enter your internship percentage: 54
Enter your CGPA: 7.2
Enter your specialization: 4
Enter your leadership skills score (out of 10): 7.3
Enter your communication skills score (out of 10): 8.2
Enter your teamwork score (out of 10): 6
Sorry, you are not likely to be hired.
```

Figure 4.2.9 Output 2 from LightGBM model

From the above results, we got the accuracy score from lgbm algorithm almost similar to xgboost algorithm and other metrics like precision, recall and f1-score also acquired some good scores.

#### **4.2.2 METRICS OF EVALUATION:**

As mentioned in previous results, we intend to use four metrics for evaluation: Accuracy, Recall, Precision, F1Score.

## A) ACCURACY

The Accuracy score is calculated by dividing the number of correct predictions by the total prediction number. It is very useful for binary classification problems. However, it cannot be applied for imbalanced datasets.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

- TP = True Positive: It represents the values which were predicted positive and are truly positive.
- TN = True Negative: It represents the values which were predicted negative and are truly negative.
- FP = False Positive: It represents the values which were predicted positive but are negative. It is an incorrect prediction.
- FN = False Negative: It represents the values which were predicted negative but are positive. It is a missed prediction.

		Actual Values	
		True	False
Predicted Values	True	TP	FP
	False	FN	TN

Confusion Matrix

Table 4.2.1 Dataset

## B) PRECISION

Precision is defined as the ratio of correctly classified positive samples (True Positive) to a total number of classified positive samples (either correctly or incorrectly).

Precision = True Positive / True Positive + False Positive

The precision of a machine learning model will be low when the value of;

TP+FP (denominator) > TP (Numerator)

The precision of the machine learning model will be high when Value of;



$TP \text{ (Numerator)} > TP+FP \text{ (denominator)}$

### **C) RECALL**

The recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect positive samples. The higher the recall, the more positive samples detected.

$\text{Recall} = \text{True Positive} / \text{True Positive} + \text{False Negative}$

$\text{Recall} = TP / TP+FN$  o Recall of a machine learning model will be low when the value of;

$TP+FN \text{ (denominator)} > TP \text{ (Numerator)}$  o Recall of machine learning model will be high when Value of;  $TP \text{ (Numerator)} > TP+FN \text{ (denominator)}$ .

### **D) F1-Score**

F-score or F1 Score is a metric to evaluate a binary classification model on the basis of predictions that are made for the positive class. It is calculated with the help of Precision and Recall. It is a type of single score that represents both Precision and Recall. So, *the F1 Score can be calculated as the harmonic mean of both precision and Recall, assigning equal weight to each of them.*

The formula for calculating the F1 score is given below:

$$F1 - score = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

As F1-score make use of both precision and recall, so it should be used if both of them are important for evaluation, but one (precision or recall) is slightly more important to consider than the other. For example, when False negatives are comparatively more important than false positives, or vice versa.

In Summary we can say that, all the metrics (accuracy, precision, recall, and F1-score) have their own significance in evaluating a machine learning model.

The choice of the metric to use depends on the problem being solved and the importance of each type of error. In practice, it is common to use a combination of these metrics to evaluate the performance of a machine learning model.

However, accuracy is still the most commonly used metric in machine learning projects, especially for binary classification problems with a balanced dataset, due to its simplicity and ease of interpretation.

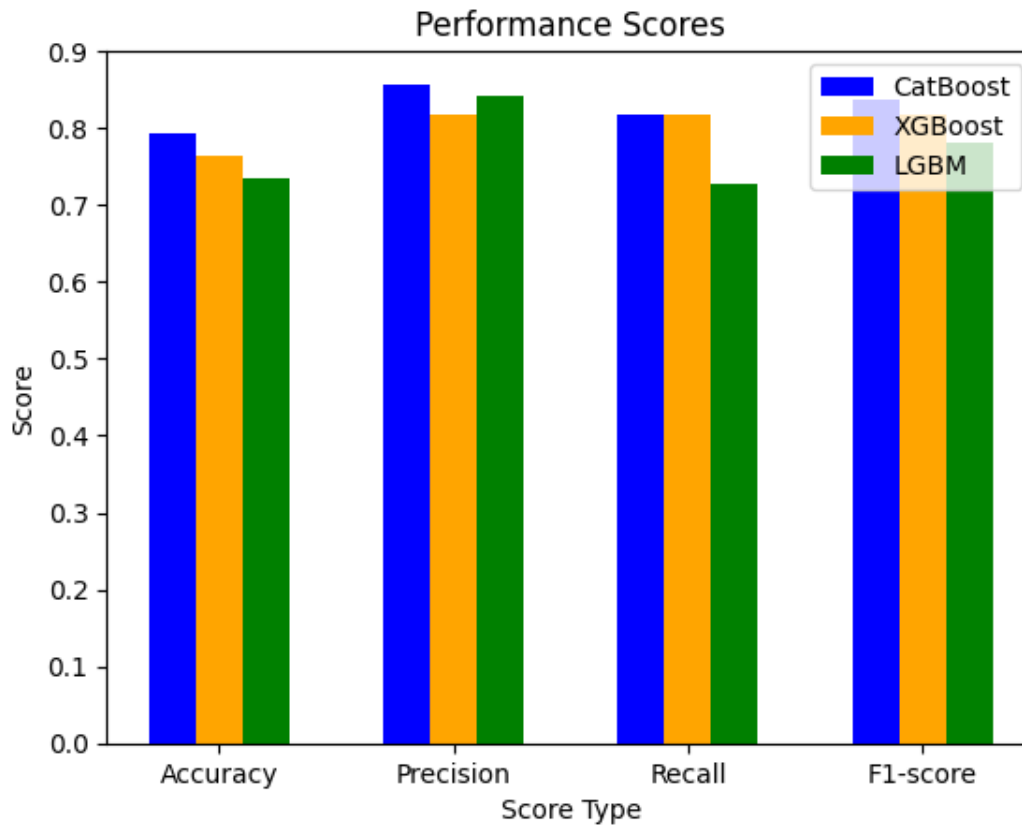


Figure 4.2.10 Graphical representation of the evaluation metrics

Classifier	Accuracy	Precision	Recall	F1-score
CatBoost	0.794118	0.857143	0.818182	0.837209
XGBoost	0.764706	0.818182	0.818182	0.818182
LGBM	0.764706	0.85	0.772727	0.809524

Table 4.2.2 Performance evaluation of gradient boosting models using all features.

Scores of the Accuracy, Precision, Recall and F1-Score for the three gradient boosting models which we used in our project (i.e. CatBoost, XGBoost, LightGBM).

Based on the dataset and context features we used, we obtained the results as shown in above table.

### 4.2.3 ANALYSIS:

Based on the evaluation metrics, we can say that CatBoost has the highest accuracy and F1-score, which suggests that it is the best algorithm for student employability prediction among the three models.

CatBoost has a higher precision score than XGBoost and LGBM, which indicates that it has a lower false positive rate, i.e., it is better at correctly identifying students who are employable. Furthermore, it is clear that internship context is the key factor determining the future of graduates in terms of employability.

Catboost algorithm acquired with good accuracy because we have totally preprocessed the dataset into new training model and loaded it into the catboost model to get the efficient predictions.

However, it is important to note that the performance of these models may be affected by various factors, such as the quality and quantity of the data used for training and evaluation, as well as the specific features and hyperparameters used for each model. So, we can say that based on the dataset we have used and the context features we used, they will be varying from one to another and gives the results in that way. Different models give different scores and the predictions will be based on those hyper parameter values and data values provided.

## **5.SUMMARY AND FUTURE SCOPE**

### **5.1 CONCLUSION**

In conclusion, predicting student employability through internship and student context features is a complex task that requires careful consideration of a variety of factors. With the advent of machine learning and data analysis techniques, it is now possible to leverage large amounts of data to make accurate predictions about student employability. By using features such as internship experience, CGPA, major, and other contextual information, we can build predictive models that can provide valuable insights into which students are most likely to succeed in the job market.

Finally, we can say that all of these models can be used for predicting student employability through internship and student context features with varying degrees of accuracy.

However, it is important to note that these predictive models should not be used in isolation, but rather as part of a larger strategy to support student success and career readiness. Factors such as networking, adaptability, and soft skills are also important determinants of employability, and should be considered when designing programs and interventions to support student success.

Among the three models, CatBoost appears to perform the best, having the highest accuracy and F1-score.

Overall, these results suggest that gradient boosting models can be effective for predicting student employability using relevant features such as internship and student context. So, compared to other models, gradient boosting models are more effective and gives good results.

## 5.2 FUTURE SCOPE:

**Improving data quality:** The accuracy and reliability of the data used for training and evaluation can have a significant impact on the performance of the models. Future work could focus on improving data quality through better data collection methods and data cleaning techniques.

**Developing personalized recommendations:** Predictive models could be used to develop personalized recommendations for students based on their internship and student context features. These recommendations could include specific courses, skill-building activities, and networking opportunities that would improve their employability in their chosen field.

**Generalizing to other domains:** The techniques used for predicting student employability could be adapted and applied to other domains where similar datasets are available, such as predicting job performance or career success.

**Evaluating the effectiveness of internship programs:** Predictive models could be used to evaluate the effectiveness of internship programs in improving student employability. This would provide valuable insights for educators and employers and help to ensure that internship programs are providing meaningful experiences for students.

**Soft Skills:** Soft skills such as problem solving skills, effective presentation skills, will be important for employability. So future research could explore ways to measure and incorporate soft skills into predictive models.

### **5.3 LIMITATIONS:**

1. Limited data availability: The availability of data on internships and student context features may be limited, which could affect the accuracy of predictive models. For example, some students may not have participated in internships or may not have provided complete information on their context features.
2. Lack of standardized data: The data on internships and student context features may not be standardized across different institutions or regions, which could make it difficult to compare and generalize predictive models.
3. Dynamic job market: The job market is constantly evolving, and it may be difficult to keep predictive models up-to-date with the latest trends and requirements of employers.

## REFERENCES:

1. D. A. Sapp and Q. Zhang, “Trends in industry supervisors’ feedback on business communication internships,” *Bus. Commun. Quart.*, vol. 72, no. 3, pp. 274–288, Sep. 2009.
2. P. Stansbie, R. Nash, and S. Chang, “Linking internships and classroom learning: A case study examination of hospitality and tourism management students,” *J. Hospitality, Leisure, Sport Tourism Educ.*, vol. 19, pp. 19–29, Nov. 2016.
3. H. Yang, C. Cheung, and H. Song, “Enhancing the learning and employability of hospitality graduates in China,” *J. Hospitality, Leisure, Sport Tourism Educ.*, vol. 19, pp. 85–96, Nov. 2016.
4. A. Giri, M. V. V. Bhagavath, B. Pruthvi, and N. Dubey, “A placement prediction system using k-nearest neighbors classifier,” in *Proc. 2nd Int. Conf. Cognit. Comput. Inf. Process. (CCIP)*, Aug. 2016.
5. N. Mezhoudi, R. Alghamdi, R. Aljunaid, G. Krichna, and D. Düşteğör, “Employability prediction: A survey of current approaches, research challenges and applications,” *J. Ambient Intell. Humanized Comput.*, pp. 1–17, Jun. 2021.
6. N. Nascimento, P. Alencar, C. Lucena, and D. Cowan, “A context-aware machine learning-based approach,” in *Proc. 28th Annu. Int. Conf. Comput. Sci. Softw. Eng. (CASCON)*, Oct. 2018.
7. L. H. Pinto and D. C. Ramalheira, “Perceived employability of business graduates: The effect of academic performance and extracurricular activities,” *J. Vocational Behav.*, vol. 99, pp. 165–178, Apr. 2017.
8. S. Kumar and G. P. Babu, “Comparative study of various supervised machine learning algorithms for an early effective prediction of the employability of students,” *J. Eng. Sci.*, vol. 10, pp. 240–251, Oct. 2019.
9. S. Maheswari, “A review on predicting student performance using deep learning technique,” *Tierärztliche Praxis, Tech. Rep.*, 2020.
10. L. K. Smirani, H. A. Yamani, L. J. Menzli, and J. A. Boulahia, “Using ensemble learning algorithms to predict Student failure and enabling customized educational paths,” *Sci. Program.*, vol. 2022, pp. 1–15, Apr. 2022.

11. K. S. Bhagavan, J. Thangakumar, and D. V. Subramanian, “Predictive analysis of student academic performance and employability chances using HLVQ algorithm,” *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 3, pp. 3789–3797, Mar. 2021.
12. A. Dubey and M. Mani, “Using machine learning to predict high school student employability—A case study,” in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2019.
13. L. Jamel, O. Saidani, and S. Nurcan, “Flexibility in business process modeling to deal with context-awareness in business process reengineering projects,” in *Proc. 20th Enterprise, Bus.-Process Inf. Syst. Modeling*, Tallinn, Estonia, Jun. 2018.
14. Z. Othman, S. W. Shan, I. Yusoff, and C. P. Kee, “Classification techniques for predicting graduate employability,” *Int. J. Adv. Sci., Eng. Inf. Technol.*, vol. 8, nos. 4–2, p. 1712, Sep. 2018.
15. L. S. Hugo, “Predicting employment through machine learning,” *NACE J.*, May 2019.