***A PROJECT ON***

# “JOB MARKET ANALYSIS AND SALARY PREDICTION”

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE COURSE OF

DIPLOMA IN BIG DATA ANALYSIS



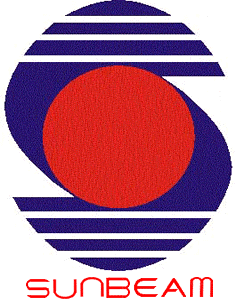
**SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY, PUNE**

Submitted By:

Pimple Aditya (86863)

Keskar Tejas (87031)

**Mr.Nitin Kudale Mrs.Manisha Hingne** Centre Coordinator Course Coordinator



**CERTIFICATE**

This is to certify that the project work under the title ‘Job Market Analysis and Salary Prediction’ is done by Pimple Aditya & Keskar Tejas in partial fulfillment of the requirement for award of Diploma in Big Data Analysis Course.

Mr. Aniket P Mrs. Manisha Hingne

**Project Guide** **Course Coordinator**

Date:

# ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mrs. Manisha Hingne (Course Coordinator, SIIT ,Pune) and Project Guide Mr. Aniket P.

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Pimple Aditya

DBDA Aug 2024 Batch, SIIT Pune

Keskar Tejas

DBDA Aug 2024 Batch, SIIT Pune

.

**TABLE OF CONTENTS**

1. **Introduction**
   1. Introduction And Objectives
   2. Why this problem needs To be Solved?
   3. Dataset Information

## Problem Definition and Algorithm

* 1. Problem Definition
  2. Algorithm Definition

## Experimental Evaluation

* 1. Methodology/Model
  2. Exploratory Data Analysis

## Results And Discussion

1. **GUI**
2. **GitHub link**

## 7.Future Work And Conclusion

* 1. Future Work
  2. Conclusion
     1. **Introduction**
        1. **Introduction And Objectives:**

The job market is highly dynamic, with salaries varying significantly based on factors such as job title, location, experience, skills, and industry. Job seekers often struggle to understand what salary they can expect, while employers face challenges in setting competitive compensation packages. This project aims to address these issues by leveraging machine learning to analyze the job market and predict salaries accurately.

Objectives:

Analyze the job market to identify trends and patterns in salaries.

Build a machine learning model to predict salaries based on key features such as job title, location, experience, and skills. Provide actionable insights for job seekers and employers to make informed decisions.

## Why this problem needs To be Solved?

For Job Seekers:

Lack of transparency in salary expectations can lead to dissatisfaction or

underpayment. Job seekers often rely on anecdotal information or outdated salary data, which may not reflect current market trends. A data-driven salary prediction tool can help job seekers negotiate better offers and make informed career decisions.

For Employers:

Setting competitive salaries is crucial for attracting and retaining top talent.

Employers often struggle to benchmark salaries against industry standards, especially for roles requiring niche skills. A salary prediction model can help employers design fair and competitive compensation packages.

For the Job Market:

Understanding salary trends can highlight skill gaps and emerging job roles. Analyzing the job market can provide insights into regional disparities and industry-specific trends.

## Dataset Information. Stores.csv:

The dataset provided contains information about job postings

Experience: This column specifies the range of experience required for the job role

UG (Undergraduate): This column specifies the undergraduate degree required for the job.

Location: This column lists the location(s) where the job is.

PG (Postgraduate): This column specifies the postgraduate degree required for the job role.

Salary: This column provides the salary range for the job role.

Department: This column specifies the department or industry sector the job belongs to.

Doctorate: This column indicates whether a doctorate degree is required for the job role.

Skills: This column lists the skills required for the job role.

Role Category: This column specifies the category or type of role.

## Problem Definition and Algorithm:

* + - 1. **Problem Definition**

The goal of this project is to analyze job market trends and predict salaries based on various factors such as experience, education, job location, skills, and role category. Many job seekers and employers struggle with estimating fair compensation, and this project aims to bridge that gap by leveraging machine learning models.

The project aims to:

Analyze job trends - Identifying key factors that influence salary.

Build a predictive model - Using machine learning to estimate salaries based on job attributes.

Optimize model accuracy - Improving predictions through data preprocessing, feature engineering, and algorithm selection.

## Algorithm Definition

**XGBoost (Extreme Gradient Boosting)** **:**

XGBoost is a powerful machine learning algorithm based on gradient boosting, designed for both regression and classification tasks. It is known for its high performance, scalability, and efficiency. XGBoost builds an ensemble of decision trees in a sequential manner, where each tree corrects the errors of the previous one.

This method uses boosting, an iterative technique that combines weak learners (shallow trees) to form a strong predictive model. XGBoost also incorporates L1 (Lasso) and L2 (Ridge) regularization to prevent overfitting and improve generalization. Due to its ability to handle missing values, feature importance selection, and fast computation, XGBoost is widely used in predictive modeling tasks, including salary prediction.

## Experimental Evaluation:

* + - 1. **Methodology:**

In this project, XGBoost is used for salary prediction. The methodology follows these key steps:

## Loading in raw data

features\_df = pd.read\_csv("features.csv") stores\_df = pd.read\_csv("stores.csv") walmart\_df = pd.read\_csv("train.csv")

master\_df =walmart\_df.merge(stores\_df, how='left').merge(features\_df, how='left') print(master\_df.shape)

master\_df.head()

## Preprocessing:

The experience is made to the specific levels with use of bins and labels. This data was split to get the experience categorized.

bins = [0, 1, 3, 5, 10, 15, float('inf')]

labels = ['Entry-level', 'Junior', 'Mid-level', 'Experienced', 'Senior', 'Expert']

# Create a new column for categorized experience

df['Experience\_Category'] = pd.cut(df['Processed\_Experience'], bins=bins, labels=labels, right=False)

The data had several missing values and needed to be cleaned. The missing values in ‘PG’ and ‘UG’needed to be cleaned. Since the number of missing values were significant, they were not removed but were replaced with zero.

df['UG'] = df['UG'].fillna('NA')

df['PG'] = df['PG'].fillna('NA')

**Salary Categorization**

Salaries were grouped into different categories for better analysis and prediction. This helps in identifying trends and making classification-based predictions. Handling Missing Salaries. For rows where salary was not disclosed, a predictive model was trained using the available salary data. The trained model was then used to estimate missing salary values, improving data completeness.

**Feature Engineering**

Experience converted to numerical values (e.g., "1-3 years" → 2 years). Education levels encoded (UG, PG, Doctorate). Location, Department, and Role Category encoded for model training.

**Model Training**

XGBoost Regressor was trained on the dataset. Hyper parameter tuning was performed to optimize performance. The model was validated using cross-validation and error metrics

from xgboost import XGBClassifier

model = XGBClassifier(

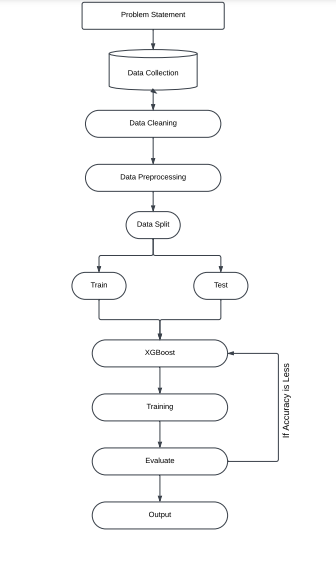
colsample\_bytree= 0.8, learning\_rate= 0.1,

max\_depth= 7, n\_estimators= 100, subsample= 0.8

)

model.fit(x\_train, y\_train)

## Flow Diagram :



* + - 1. **Exploratory Data Analysis**

The salary of the each experience level person can be analyzed with the help of following figure. It shows how salary increases when experience increases and with the help of pie chart we can check the average pay of the each experienced person.

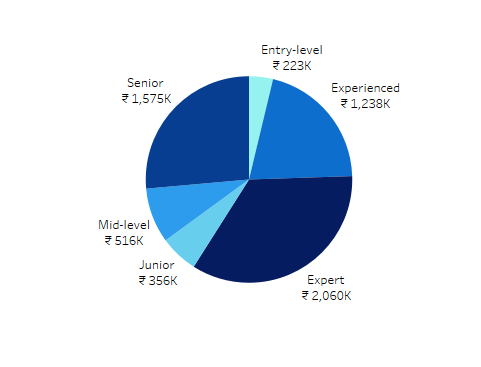


Fig 2: Pie- chart showing experience wise average salary

The average salary of department-Role category can be observed with the help of the give table chart. It clearly shows that the software development Role Category have the highest pay and lowest is Quality Assurance and Testing.

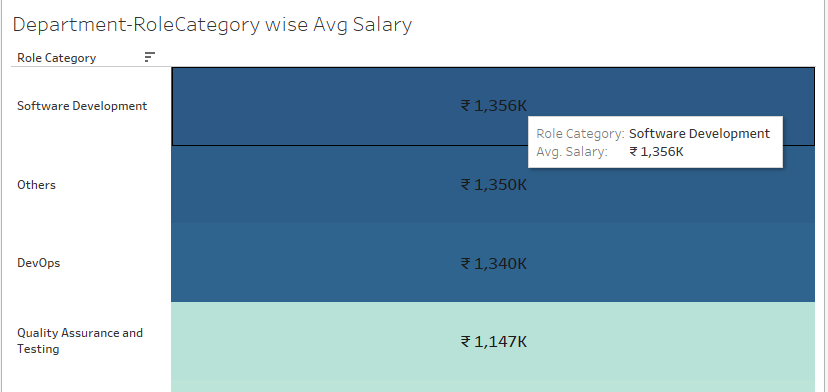


Fig 3: Department-RoleCategory Avg Salary

The educational requirement for the job post is necessary this table chart give the info about department and Role Category Wise Educational Requirement. From this chart we can say that most of the job posting requires the PG course students which will a lot for the EDA purpose.

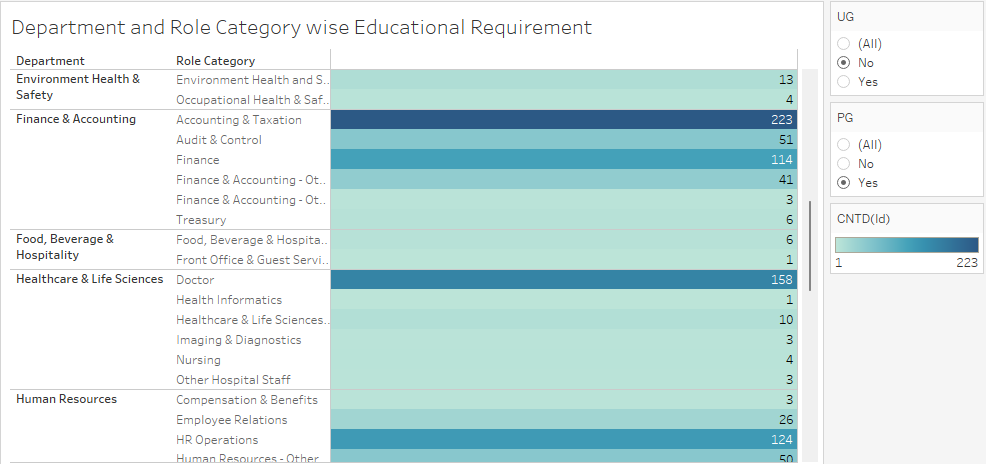


Fig 4: Department and Role Category wise Educational Requirement

## Results and discussion:

XGBoost algorithm is used to predict the salary of a preferred job which will give the value in range of the classes as one of the class of the dependent variable, with the help of Experience, City, UG, PG, Department and Role Category.

from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score, roc\_auc\_score

y\_pred\_train = model.predict(x\_train)

y\_true\_train = y\_train

# calculate training accuracy

training\_accuracy = accuracy\_score(y\_true\_train, y\_pred\_train)

# testing accuracy

y\_pred\_test = model.predict(x\_test)

y\_true\_test = y\_test

# get the performance metrics for testing

testing\_accuracy = accuracy\_score(y\_true\_test, y\_pred\_test)

testing\_precision = precision\_score(y\_true\_test, y\_pred\_test,average='macro')

testing\_recall = recall\_score(y\_true\_test, y\_pred\_test,average='macro')

testing\_f1 = f1\_score(y\_true\_test, y\_pred\_test,average='macro')

print("training\_accuracy", training\_accuracy)

print("testing\_accuracy = ", testing\_accuracy)

training\_accuracy 0.869926873857404

testing\_accuracy = 0.8630838131797824

## GUI:

GUI is made using Flask framework. **Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools

**6.GitHubLink:** <https://github.com/adityapimple0306/CDAC_Project>

## 7.Future work And Conclusion 7.1Future Work:

To further improve the Job Market Analysis and Salary Prediction model, the following enhancements can be considered:

**Feature Engineering Improvements**

Extract more insights from job descriptions using Natural Language Processing (NLP).

Incorporate company size, industry type, and demand trends as additional predictive features.

Use cost of living index for location-based salary adjustments.

**Advanced Model Optimization**

Fine-tune the XGBoost model with Bayesian Optimization for better hyperparameter tuning.

Experiment with neural networks (Deep Learning) for salary prediction if data volume allows.

Try AutoML frameworks to test multiple models efficiently.

**Real-time Data Integration**

Incorporate real-time job postings to analyze market trends dynamically.

Use web scraping techniques to fetch salary insights from job portals.

**Interactive Salary Prediction Tool**

Develop a web application (using Flask, Django, or Streamlit) where users can input their details to predict salaries.

## 7.2 Conclusion:

* + - Salaries increase with experience, but the growth rate varies across industries.
    - Type 'A' stores outclass the 'B' and 'C' types in terms of size and the avergae weekly sales
    - The approach of predicting missing salaries based on disclosed data improved dataset completeness.
    - This helped in maintaining data integrity and ensuring accurate predictions.
    - Experience, PG, UG (Education), Department-Roles, and Location play a crucial role in salary determination.
    - After feature engineering and hyperparameter tuning, XGBoost provided better performance than traditional models.

* + - The model can be integrated into a job portal or salary prediction tool for real-time salary estimates.