Industry-Specific Placement Prediction System

Nimish Bansal - 2021B5A71179P
Navneet Singla -2021A7PS1450P
Gaurang Karwanyun-2021A4PS1332P

Birla Institute of Technology and Science
Second Semester 2024-25, CS F407 (Artificial Intelligence)

February 20, 2025

1 Abstract

This project aims to predict the industry sector (IT, Finance, Core Engineering, etc.) in which a student is most likely to be placed. By leveraging machine learning models and industry-relevant datasets, the system will provide career guidance and assist students in aligning their skills with industry demands. The study compares multiple ML models based on their predictive performance.

2 Domain

Industry-Specific Placement Prediction

3 Research Papers Summary

3.1 Prediction of Final Result and Placement of Students Using Classification Algorithm (Naik & Purohit, 2012)

- Models: Decision Trees (XLMiner tool) for predicting MCA results and placements, Decision rules derived from classification trees.
- Dataset: 325 MCA students from Mumbai institutes.
- Features: Gender, SSC/HSC scores, graduation marks, MCA results, placement status.
- Key Findings:

- Students from STEM backgrounds had 76.19% placement accuracy.

- Evaluation Metrics: Error rate, classification accuracy.

3.2 Incorporating Features Learned by an Enhanced DKT Model for STEM/Non-STEM Prediction (Yeung & Yeung, 2019)

- Models: DKT+, GBDT, LDA, LR, SVM.
- Dataset: 1,709 students (ASSISTments platform).
- Features: Clickstream interactions, affective states, knowledge states.
- Key Findings:
  - DKT+ knowledge features achieved AUC 0.623.
  - STEM students showed higher math mastery (+5.7%).
- Evaluation Metrics: AUC, RMSE, Average Precision (AP).

3.3 Collaborative Job Prediction Using Naïve Bayes Classifier (Choudhary et al.)

- Models: Na¨ıve Bayes for Bayesian ranking, Skill similarity via Euclidean distance.
- Dataset: 1,500 user profiles from job portals.
- Features: User skills, job history.
- Key Findings:
  - Euclidean distance outperformed Pearson coefficient.
  - Best accuracy: 92.74% with a 0.25 training-test division ratio.
- Evaluation Metrics: Mean Squared Error (MSE), accuracy.

3.4 Model Construction Using ML for Student Placement Prediction (Nutipalli et al., 2022)

- Models: SVM, LR, Na¨ıve Bayes, XGBoost, Decision Tree.
- Dataset: Kaggle dataset of 215 students.
- Features: Academic scores, work experience, MBA specialization.
- Key Findings:
  - SVM had the highest accuracy (91%).
  - Work experience and MBA specialization were critical predictors.
- Evaluation Metrics: Accuracy, F1-score, precision, recall.

3.5 Classification Model of Prediction for Placement of Students (Pal & Pal, 2013)

- Models: Naïve Bayes, MLP, J48 Decision Tree.
- Dataset: 65 MCA students from VBS Purvanchal University, India.
- Features: Seminar performance, lab work, communication skills, graduation background, MCA result.
- Key Findings:
  - Naïve Bayes achieved highest accuracy (86.15%).
  - Top predictors: Seminar performance, communication skills.
- Evaluation Metrics: Accuracy, Kappa Statistic, Precision, Recall.

3.6 Common Themes

- **Academic Performance**: SSC/HSC scores, graduation marks, and technical skills are critical predictors.
- **Algorithm Choice**: SVM and ensemble methods (XGBoost) often outperform simpler models like Naïve Bayes.
- **Data Quality**: Preprocessing (encoding, handling missing values) significantly impacts model performance.
- **Evaluation Metrics**: AUC and F1-score are preferred for imbalanced datasets (e.g., STEM/non-STEM).

4 Problem Statement

The objective of this project is to predict the specific industry (IT, Finance, Core Engineering, etc.) where a student is most likely to be placed. The study compares multiple ML models to determine the most effective approach.

5 ML Models Considered

1. Decision Tree (DT)
2. Naïve Bayes (NB)
3. XGBoost
4. Random Forest
5. Neural Networks (BERT/LSTMs)

6 Theoretical Comparative Study of ML Models

6.1 Decision Tree (DT)

- Simple and interpretable.
- Works well with structured data.
- Prone to overfitting in large datasets.

6.2 Naïve Bayes (NB)

- Effective for categorical data.
- Fast computation.
- Assumes independence of features, which may not always hold true.

6.3 XGBoost

- High accuracy and efficiency.
- Handles complex relationships in data.
- Computationally expensive.

6.4 Random Forest (RF)

- Reduces overfitting compared to Decision Trees.
- Works well with missing data.
- Less interpretable compared to individual decision trees.

6.5 Neural Networks (BERT/LSTMs)

- Best for text-based job descriptions.
- Extracts deep patterns from data.
- Requires large datasets and computational resources.

7 Datasets

- Placement Dataset of College Students (Kaggle)
  https://www.kaggle.com/datasets/firozchowdury/placement-package-ctc-prediction-dataset/data

8 Metrics for Comparison

To evaluate the effectiveness of the models, the following metrics will be used:

- Accuracy: Measures overall correctness of predictions.
- Precision: Evaluates the proportion of correctly predicted industry labels.
- Recall: Measures the model's ability to correctly identify all instances of an industry label.
- F1-Score: Balances Precision and Recall for overall performance.
- Confusion Matrix: Analyzes classification errors.
- ROC-AUC Score: Measures model performance in distinguishing between different industries.

9 Pipeline for Industry-Specific Placement Prediction

9.1 Step 1: Data Collection Preprocessing

Data Sources:

- Scrape LinkedIn job postings using BeautifulSoup/Selenium.
- Extract university placement data from structured databases.
- Use Kaggle datasets for student career transitions.

Preprocessing Steps:

- **Handling Missing Values:** If CGPA, certifications, or internships are missing, fill with median values or create a separate "missing" category.
- **Feature Encoding:** Convert categorical data (e.g., Degree, Major, Certifications) into numerical form using One-Hot Encoding or Label Encoding.
- **Text Processing:** Apply TF-IDF or Word Embeddings (BERT) to extract important features from job descriptions.
- **Feature Scaling:** Normalize numerical values such as CGPA, years of experience, number of certifications.

9.2 Step 2: Feature Engineering

Extract meaningful insights from student data:

- **Academic Features:** CGPA, Major, Degree Level, Relevant Coursework.
- **Skill-Based Features:** Programming Languages, Certifications, Hackathons, Online Courses.
- **Experience-Based Features:** Internships, Past Job Roles, Research Experience.
- **Job Market Trends:** Industry growth rate (from external labor market datasets), Skill demand trends.

9.3 Step 3: Model Training Selection

We need multi-class classification models to predict the industry category.

Models Considered:

- **Decision Tree (DT):** Easy to interpret, works well with structured data. Prone to overfitting on large datasets.
- **Na¨ıve Bayes (NB):** Works well with probability-based categorical data. Assumes feature independence.
- **XGBoost:** Handles imbalanced datasets efficiently but is computationally expensive.
- **Random Forest:** Reduces overfitting but is less interpretable than Decision Trees.
- **Deep Learning (LSTMs BERT):** Effective for textual job descriptions but requires large datasets.

9.4 Step 4: Model Evaluation

Metrics Used:

- Accuracy, Precision, Recall, F1-Score.
- Confusion Matrix to analyze misclassifications.
- Cross-validation using k-fold for better generalization.