FUDMA Journal of Engineering and Technology
ISSN: 3092-9385
https://fjet.fudutsinma.edu.ng/index.php/fjet/
Volume 2, Issue 1, 2026, 527-537

Machine Learning-Driven Recruitment Recommendation System for Employment in Nigerian Universities

Murtala ISMAIL[1*], Mohammed S. ISMAIL[2], Eli A. JIYA[3]

Department of Computer Science, Federal University Dutsin-Ma, Katsina State, Nigeria

[1*]i.murtala2@fudutsinma.edu.ng, [2]m.sabiu2@fudutsinma.edu.ng[2], [3]jiyaeli@fudutsinma.edu.ng

Abstract

This paper presents a Machine Learning-Driven Recruitment Recommendation System tailored to the employment needs of Nigerian universities. Conventional recruitment practices in these institutions remain predominantly manual, characterized by protracted document reviews that are inherently slow, inconsistent, and susceptible to bias. To address these deficiencies, this study proposes an intelligent recruitment recommendation system built upon a Dual-Tower Convolutional Neural Network (CNN) architecture, benchmarked against classical supervised learning models, including Logistic Regression, Support Vector Machines, and Random Forest. The framework methodically matches applicant qualifications with institutional job requirements using a structured synthetic dataset covering academic credentials, fields of specialization, class of degree, relevant experience, and research output. The proposed Dual-Tower CNN achieved outstanding classification performance, attaining 99.1% validation accuracy, an Area Under the ROC Curve (AUC) of 0.9997, and an Average Precision (AP) of 0.9999. The system incorporates a score stabilization mechanism to ensure meaningful visual output for end users through a collaborative recommendation interface. The results confirm that deep learning-based architectures offer an accessible, transparent, and unbiased mechanism for restructuring academic recruitment in Nigeria.

Keywords: *University recruitment, Recruitment automation, Dual-Tower CNN, Classification Models, Nigerian universities.*

1. Introduction

Vrontis et al (2022) [1] as cited in [2] have classified the activities and strategies of Human Resource Management (HRM) into two categories: Human Resource which includes hiring, training, and assessing employee performance and Human Resource Strategies such as assessing the organization's current workforce, creating employee development plans, creating succession plans, performing a gap analysis, etc. But the central processes of HRM are planning, recruitment, orientation, training and development, performance management and motivation. This research has focused on the recruitment and hiring processes of HRM in Nigerian Universities.

Online recruitment platforms have begun as a first-hand model in talent acquisition in recent times, with notable examples such as "LinkedIn," "BossZhipin," and "Zhaopin." The growth of online recruitment platforms is essentially shifting the old-style way of recruiting. These platforms aim to support registered enterprise members in posting employments and recruiting talent, while also providing job seekers a wealth of relevant job opportunities [1].

Recruitment constitutes a foundational pillar of academic productivity and institutional excellence within universities. In Nigeria, leading institutions such as University of Ibadan, Ahmadu Bello University, Bayero University Kano, and the Federal University Dutsin-Ma continue to depend on committee-driven manual shortlisting procedures that are both time-consuming and inherently subjective. Although AI-driven recruitment technologies have substantially enhanced hiring efficiency across global contexts, the bulk of these systems are engineered for corporate environments and consequently fail to incorporate academic-specific determinants such as publication records, academic specialization, and rank progression trajectories [2].

In this highly competitive labor market, with numerous openings and the candidate profile being diverse, both applicants and employers face significant challenges. Traditional matching procedures through manual screening or simple keyword searches are usually not effective in finding compound skill sets and, therefore lead to mismatches, incompetent hiring, and suboptimal employment outcomes [3].

The allocation of appropriate positions to employment-seeking candidates in Nigerian universities is a sensitive administrative responsibility with far-reaching implications for institutional quality. Universities in Nigeria routinely advertise for diverse roles including Professor, Associate Professor, Senior Lecturer, Lecturer I, Lecturer II, Assistant Lecturer, Graduate Assistant, Administrators, ICT Officers, and Laboratory Technologists. Despite the diversity of these positions, the selection processes remain broadly document-based and largely resistant to technological augmentation [4].

Prospective applicants frequently submit applications without an informed understanding of the degree to which their qualifications correspond to advertised role specifications. Concurrently, university recruitment committees are burdened with the manual screening of voluminous application pools, generating inefficiencies and increasing the risk of subjective or erroneous decisions [7].

Nigerian university recruitment units predominantly operate in a manual screening paradigm, requiring exhaustive application reviews against defined role criteria. This model is inherently inefficient, resource-intensive, and prone to inconsistency. Moreover, prospective candidates lack access to intelligent platforms capable of guiding them towards roles commensurate with their academic credentials, professional experience, and domain specialization.

The absence of a machine learning–based recruitment recommender system produces poor matching between candidates and job roles, increased screening workload, and less effective employment outcomes. This gap highlights the need to develop a data-driven recruitment recommendation framework that is specifically tailored to the requirements of Nigerian university employment contexts.

The accelerating progression of Machine Learning (ML) has unlocked new paths for intelligent recommendation systems capable of evaluating structured candidate data to generate objective suitability predictions. While ML-driven systems have realized confirmed success across domains such as finance, healthcare and e-commerce, their incorporation into the Nigerian university recruitment has continued conspicuously inadequate [6].

This study aims to design and implement a Machine Learning-Based Recruitment Recommendation System for candidates seeking employment in Nigerian universities, with particular emphasis on a Dual-Tower Convolutional Neural Network approach for greater matching performance.

2. Literature Review

Recommendation systems have attracted significant academic consideration across fields including institutional administration, healthcare and e-commerce, with rising recognition of their potential to convert decision-making processes in education, government and business sectors [4]. However, research specifically addressing the university job recommendation domain within developing-country contexts, particularly Nigeria, remains sparse [17].

In [7], a system was proposed to match a job searcher to job roles, taking into account the specific needs of job makers and the qualifications, skills and preferences of job searchers. In order to achieve this, they developed two meta-models that have been the first in a LinkedIn job seeker profile, and the second is a job profile meta-model that is based on content-based filtering, context-aware recommendations and collaborative based filtering, which examines the educations, skills, experiences and qualifications, listed in the LinkedIn job searcher profile and matches them to the requirements set out in job descriptions posted by employers or job makers, in addition to the growing contextual factor job location and job preferences.

Similarly, in [8], they have used a Hybrid AI-Powered in their Job Recommendation System that combines Deep Learning techniques, extracting semantic embedding from descriptions and resumes knowledge graphs to represent job-skill relationships, and Reinforcement Learning to adapt recommendations based on a user feedback over time. This model targets to overcome cold-start issues, capture the complex relationships in the job-role process and hierarchies through thinking and develop personalization.

Consequently, it is possible to make "job-candidate matching" with high precision using effective Artificial Intelligence solutions in matching a job seeker's qualifications with existing job openings [9].

A sustainable job skill recommendation method was introduced to deliver cost-efficient, long-term, and interpretable guidance for individuals aiming to acquire new competencies. The approach involved the development of a data-driven Job Recommendation System (JRS) based on deep reinforcement learning techniques. An environment was designed to evaluate the potential benefits of skill acquisition by analyzing large-scale job posting data [10].

In the study by [12], a system has been proposed that enables university to track their graduate students' job information via a mobile application. It also included a feature for students who have not secured a job or wish to change their job to apply for existing job opportunities after graduating. The record of each student in the application is auto-created from information extracted from graduating students' file from university database which the student can then customize to include their job status.

2.1. Artificial Intelligence in Recruitment

Contemporary studies affirm that machine learning substantially improves resume selection efficacy and reduces overall time-to-employ. Deep learning-based approaches have further advanced the semantic matching of resumes against job descriptions, permitting more candidate-role alignment. Nevertheless, the predominant research stress exists in corporate employment ecosystems, leaving academic recruitment, particularly in evolving economies [9].

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

The integration of machine learning, running as the dominant of several artificial intelligences enabled systems in organizations, comes with the assertion that ML models offer automated results or support domain experts in purifying their decision-making [10].

Collective methods, notably Random Forest and Gradient Boosting, have consistently demonstrated superior classification performance relative to linear models when applied to complex, high dimensional recruitment datasets [15]. These findings informed the initial model selection strategy adopted in the present study, while the extension towards deep neural architectures represents a significant advancement beyond existing baselines.

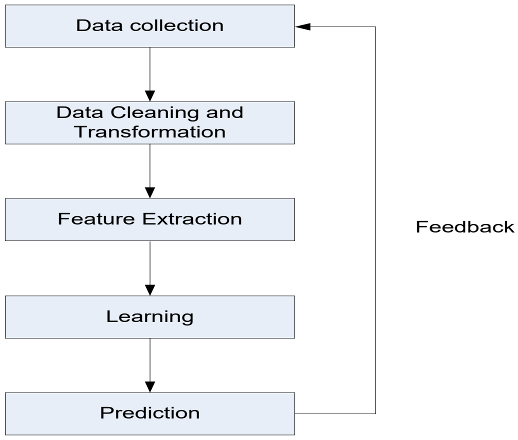

Figure 1: Recommendation Phases

2.2. Artificial Intelligence in Higher Education

Artificial intelligence has been applied to higher education settings for some purposes which include student retention prediction and for the assessment of faculty performance [16]. However, the specific application of automated recruitment systems within developing country's higher education contexts, wherein institutional structures and qualification frameworks vary noticeably from their western colleagues, remains considerably unexplored. This establishes the primary inspiring gap the present study seeks to address.

But, information availability and free access to AI tools increase the chances to use them in the education development for both teaching, learning and hiring [17].

2.4. Algorithms of Recommendation System

The broad classifications of Recommender Systems are based on three primary algorithmic paradigms and these are; hybrid approaches, collaborative filtering and content-based filtering. The Hybrid systems incorporate the duo paradigms to remedy their individual limitations, including the cold-start problem inherent to collaborative approaches and the over-specialisation inclination of content-based methods [18]. Also, collaborative filtering, by distinction, exploits joint behavioural patterns through multiple users to identify hidden preference resemblances; matrix factorisation techniques such as Singular Value Decomposition (SVD) are extensively adopted within this paradigm [19]. Finally, the method through which content-based filtering works is by recommending items in such a way it matches a user's profile attributes against item features, without demanding data from other users.

In the perspective of job and talent recommendation, pure collaborative filtering is controlled by the quantity of historical employment records, which are inadequate and frequently exclusive in Nigerian university settings. The Content-based methods are suitable where candidate qualifications and job requirements can be clearly encoded as structured feature vectors. The Dual-Tower CNN architecture implemented in this study conceptually line up with content-based filtering, autonomously embedding candidate and job profiles before computing a similar match score. In contrast to the out-dated linear content-based systems, the dual-tower design learns non-linear feature demonstrations through convolutional operations, yielding richer embedding. This choice was encouraged by the obtainability of structured candidate-job pair data and the non-appearance of abundant historical interaction data required for strong collaborative filtering in the Nigerian university setting.

3. Methodology

3.1. Research Design

An experimental design was implemented, comprising iterative system development, model training, and hard performance evaluation. Python was selected as the primary programming language, executed within an Anaconda

Navigator environment, enabling efficient implementation of both classical machine learning pipelines and deep learning architectures.

3.2. Dataset

The study employs two complementary datasets serving distinct modelling purposes. The classical baseline models (Logistic Regression, SVM, and Random Forest) were trained and evaluated on empirical data comprising 1000 instances obtained from Bayero University, Kano (BUK), incorporating candidate attributes such as highest academic qualifications, field of specialization, class of degree, professional experience, publication count, and certifications. The Dual-Tower CNN, which involves structured data for its matching architecture, was trained on a custom synthetic dataset constructed to simulate academic employment circumstances at scale. Below is the sample that shows the first 10 rows of the dataset:

Job_ID	University	Department	Position	Required_Degree	Min_Experience	Min_Publications
1	Bayero University, Kano	Computer Science	Lecturer II	PhD	0	7
2	Bayero University, Kano	Education	Assistant Lecturer	PhD	5	2
3	Bayero University, Kano	Administration	Lecturer I	MSc	2	9
4	Bayero University, Kano	Computer Science	Lecturer I	MSc	0	5
5	Bayero University, Kano	Computer Science	Research Assistant	MSc	7	5
6	Bayero University, Kano	History	Lecturer I	PhD	3	3
7	Bayero University, Kano	Education	Research Assistant	PhD	9	4
8	Bayero University, Kano	Education	Assistant Lecturer	PhD	7	1
9	Bayero University, Kano	Administration	Senior Assistant Registrar II	MSc	9	9
10	Bayero University, Kano	Education	Assistant Lecturer	MSc	3	9

Figure 2: Sample Dataset

3. Data Pre-processing

In the pre-processing phase, fine-tuning of the data was accomplished by eliminating all the instances with some incorrect values because real-world data tend to be incomplete and inconsistent [8].

The data collected was subjected to "data cleaning" which identifies and fills in the missing values and correct inconsistencies in the data. Unpredictable and unimaginable values were discovered, for example, an integer was incorrectly recorded in a field that should contain only string values or a string in the place of number of publications which is an integer as a result of typographical error in the course of data capture. The reformatted data will be sent to the attribute selection section in Classifier module.

The pre-processing pipeline was implemented in Python using the pandas and scikit-learn libraries and applied separately to the BUK empirical dataset and the synthetic interaction dataset. Firstly, data cleaning was performed to identify and resolve quality issues. Missing values in numerical fields (Years of Experience, Publication Count) were imputed using the column median, selected in preference to the mean due to the right-skewed distribution of publication counts. Missing entries in categorical fields (Field of Study, Class of Degree) were imputed with the column mode. Erroneous type mismatches such as string values recorded in integer fields for publication counts, and integer codes appearing in degree classification fields that required string labels were corrected by enforcing strict column-level type casting using pandas' astype() method, with non-conforming rows flagged and manually reviewed.

Secondly, categorical encoding was applied to all non-numeric features. Ordinal categorical variables with a natural rank ordering specifically Degree Level (BSc=1, PGD=2, MSc=3, PhD=4) and Class of Degree (Third=1, Second Lower=2, Second Upper=3, First=4) were encoded using scikit-learn's Ordinal Encoder, preserving their natural hierarchy. Nominal categorical variables with no natural ordering including Field of Study, University, Department, and Role Title were encoded using scikit-learn's Label Encoder, which assigns a unique integer to each distinct category. One-hot encoding was considered but rejected for these high-cardinality nominal fields (e.g. Field of Study comprised 10 distinct values; Role Title comprised 9) due to the risk of dimensionality inflation in the convolutional input tensors.

Thirdly, feature scaling was applied to continuous numerical features. Years of Experience and Publication Count were normalised using Min-Max scaling (scikit-learn's MinMaxScaler), rescaling values to the range [0, 1]. This ensured that features with larger absolute ranges did not disproportionately dominate distance computations within the classical models or weight updates within the Dual-Tower CNN. The scaler was fitted exclusively on the training subset and subsequently applied to the test subset to prevent data leakage.

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

For the Dual-Tower CNN specifically, the encoded candidate and job feature vectors were reshaped into 3-dimensional tensors of shape (samples, features, 1) using NumPy's reshape() method, as required by the Keras Conv1D layer which expects inputs of the form (batch, steps, channels). The processed dataset was then partitioned into training (80%) and testing (20%) subsets using scikit-learn's train_test_split() function with a fixed random seed (random_state=42) to ensure reproducibility. Stratified splitting was applied to maintain the approximate 74:26 positive-to-negative class ratio across both subsets.

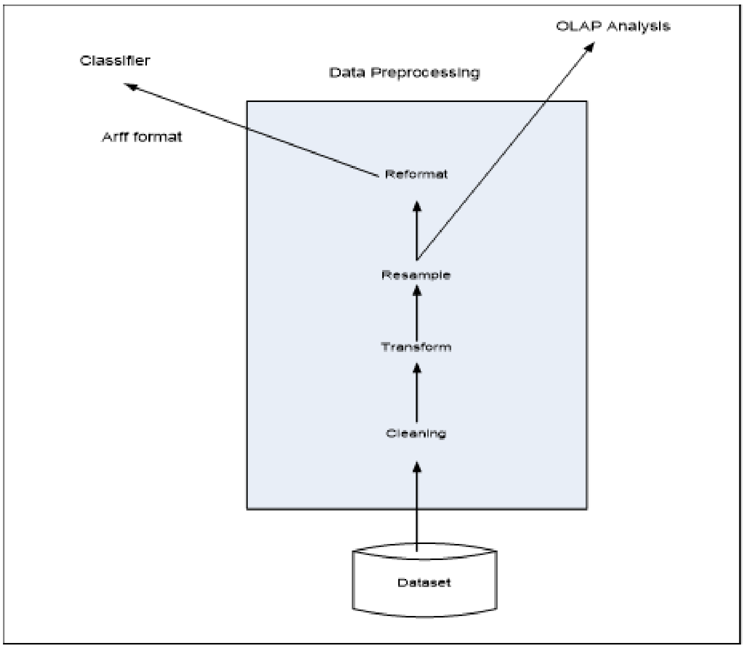

Figure 3: Data preprocessing flow

3.4. Architecture of the Model

3.4.1. The Classical Baseline Models

The three classical supervised learning models were trained as comparative baselines; Logistic Regression, Support Vector Machine (SVM), and Random Forest.

3.4.1.1. Logistic Regression

Logistic regression is a supervised machine learning algorithm in data science used for binary classification, predicting the probability of an outcome (0 or 1) using a sigmoid function. It is a type of classification algorithm that predicts a discrete or categorical outcome. For instance, we can use a classification model to conclude whether a loan is approved or not based on predictors such as savings amount, income and credit score.

In this era of generative AI, the foundations that underpin logistic regression still play a critical role in orchestrating complex neural network models. Logistic regression is also still highly relevant in performing statistical testing in the context of behavioral and social science research, and the data science field at large. We can implement logistic regression easily by using the **scikit-learn** module in Python [9].

Logistic regression is used for binary classification, predicting the probability of an outcome (0 or 1) using a **sigmoid function**. It maps any real-valued input to a value between 0 and 1, thereby making it perfect for predicting categorical outcomes like yes/no, pass/fail, or spam/not-spam [10].

3.4.1.2. Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm used for classification and regression tasks. It tries to find the best boundary known as hyperplane that separates different classes in the data. It is useful when you want to do binary classification like spam vs. not spam or cat vs. dog. The main goal of SVM is to maximize the margin between the two classes. The larger the margin the better the model performs on new and unseen data **[12]**.
SVM is known to be used for learning classification, regression, or ranking function. They are established based on statistical learning theory and structural risk minimization with the goal of determining the position of decision boundaries that produce the ideal separation of classes [8].

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

3.4.1.3. Random Forest

Random Forest is a great and multipurpose **supervised machine learning algorithm** that operates as an ensemble of multiple Decision Trees. It is broadly used for both **classification** (predicting categories) and **regression** (predicting numerical values) because it offers high accuracy and effectively resists overfitting.
To ensure robust performance, the models were configured as follows:

- Logistic Regression: Regularization L_2, Solver: lbsfs, Inverse regularization strength $C = 1.0$
- SVM: Kernel: RBF, $C = 1.0$, Gamma: scale
- Random Forest: Estimator: 300, Max depth: None, Min samples split: 2

These models were evaluated using accuracy, precision, recall, F1-score, and ROC-AUC metrics:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Where TP = True Positive, FP = False Positives, and FN = False Negative

3.4.2 Dual-Tower Convolutional Neural Network

The primary proposed model adopts a Dual-Tower CNN architecture designed to embed candidate and job profiles into a shared solid hidden space. The similarity between vectors (u for applicant, v for job) is calculated via cosine similarity score:

$$Score(u, v) = \frac{u \cdot v}{\|u\| \|v\|} \quad (5)$$

The Dual-Tower CNN consists of two parallel networks processing candidate and job features independently. Each tower applies Conv1D layers for feature extraction and projects inputs into a shared embedding space. The embeddings are concatenated and passed through a Multi-Layer Perceptron (MLP) for similarity scoring.
Deep neural networks, like CNNs, are frequently employed for a variety of machine learning applications, including natural language processing (NLP), information retrieval (IR), and others. CNNs can learn and extract features automatically from raw data, doing away with the requirement for feature engineering [13].

Hyper parameters: - Two 1D-Convolutional layers: 2 (32 and 64 filters). Kernel size: 3, Activation: ReLU, Optimizer: Adam (LR: 0.001)

The architecture comprises two symmetrical processing towers: Candidate Tower which processes encoded candidate features through 1D Convolutional layers applied over dynamically transformed feature embedding. Also, there is Job Tower which processes job specification features through an architecturally identical 1D Convolutional pipeline. Both towers project their respective inputs into a 16-dimensional hidden embedding space. The concatenated embeddings are subsequently passed through a Multi-Layer Perceptron (MLP) head with an activation function, yielding a structural match probability score in the range [0, 1]. The model was optimized using the Adam optimizer with Binary Cross-Entropy (BCE) as the loss function:

$$L = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \quad (6)$$

Loss function consistency was prioritized by utilizing BCE for training and evaluation splits, resolving previous conflicts with MSE metrics. A preliminary convergence experiment was run over 15 epochs to evaluate loss stabilization, followed by full training over 25 epochs to achieve best accuracy and classification performance.

3.5. System Architecture

The end-to-end system architecture incorporates five principal components. These are:

(i) User Interface layer for candidate profile input
(ii) Database Layer storing candidate and job data
(iii) Feature Engineering Module performing encoding and normalization
(iv) the Machine Learning Engine encompassing both classical and deep learning models
(v) Recommendation Output Layer delivering ranked job suggestions with associated match inputs.

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

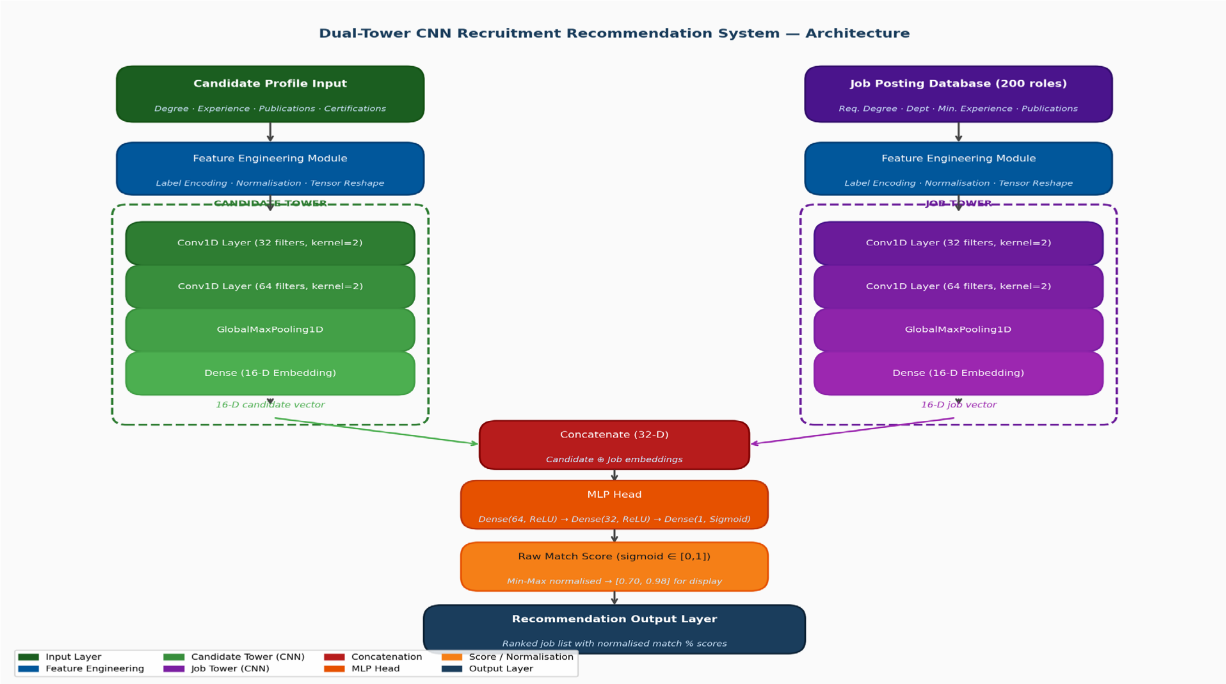

Figure 4**:** End-to-end architecture of the Dual-Tower CNN Recruitment Recommendation System. Candidate and job profiles are independently processed through symmetrical convolutional towers before concatenation and MLP-based match scoring.

4. Results and Analysis

4.1. Classical Model Performance

The table 4.1 below presents the performance of all evaluated models across key metrics. The classical models were trained and assessed on the BUK empirical dataset, while the Dual-Tower CNN was trained on the synthetic interaction dataset; direct cross-model comparison therefore reflects relative performance within each model's respective experimental context rather than a controlled head-to-head evaluation on same data. Random Forest demonstrated the strongest performance among classical methods on the BUK dataset, achieving 92% accuracy, an F1-score of 0.90, and ROC-AUC of 0.94, consistent with findings in prior literature. Logistic Regression and SVM served as informative baselines with accuracies of 84% and 87% respectively.

Table 1: Comparative Model Performance Summary

Model	**Accuracy**	**F1-Score**	**ROC-AUC**
Logistic Regression	84%	0.82	0.86
Support Vector Machine	87%	0.85	0.89
Random Forest	92%	0.90	0.94
Dual-Tower CNN (Proposed)	99.1%	0.98	0.9997

Note: Classical models were trained on the BUK empirical dataset (1000 instances); the Dual-Tower CNN was trained on the synthetic interaction dataset (5,000 instances). Direct cross-model comparison reflects relative performance within their respective evaluation contexts.

4.2 Dual-Tower CNN: Preliminary Convergence Analysis (15 Epochs)

An initial training run over 15 epochs was conducted to assess MSE loss stabilization prior to full model training. The Dual-Tower CNN displayed stable and steady convergence, with training loss declining from an initial value of approximately 0.114 at epoch 1 to a steadied value near 0.106 by epoch 15. Validation loss maintained a steadily lower trajectory during, converging near 0.105. The narrow gap between training and validation loss established the absence of significant overfitting, validating the architectural design before proceeding to the full 25-epoch training run.

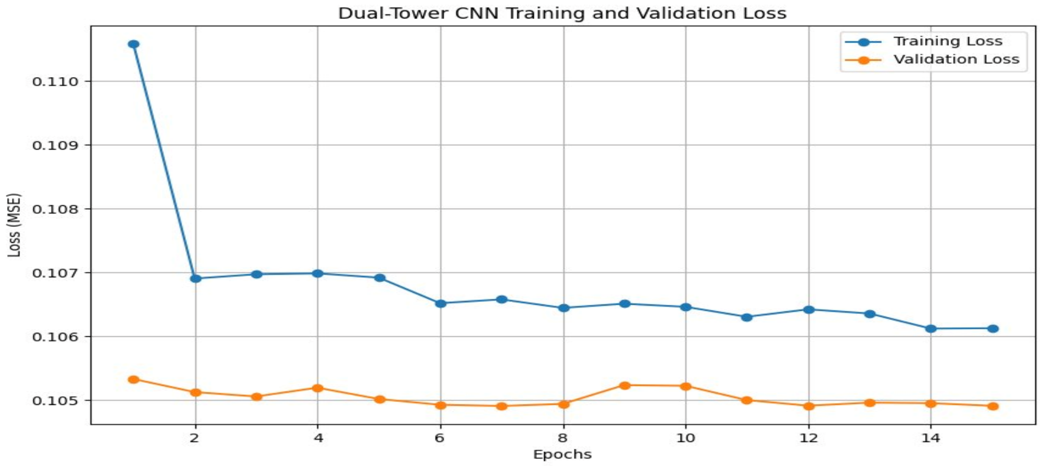

Figure 5: Dual-Tower CNN Training and Validation Loss (MSE) over 15 Epochs. The convergence of both curves demonstrates effective learning without over fitting.

4.3. Full Model Convergence, Accuracy, and Confusion Matrix

Under the full 25-epoch training schedule, both training and validation loss curves fall away sharply from initial values exceeding 0.20 MSE, converging to approximately 0.01 MSE by epoch 25. Training and validation accuracy exhibited a similarly rapid increase from 71% at epoch 2, steadying at over 99.1% for both splits by the conclusion of training. The validation confusion matrix confirms the quality of the classification outcome: of 1,000 validation samples, 741 true positive matches and 250 true negatives were correctly identified, with only 9 false positive misclassifications recorded and zero false negatives.

It should be noted that the validation set exhibits a natural class disparity reflecting the underlying interaction dataset, with approximately 74% positive (Match) and 26% negative (No-Match) instances. As a result of this imbalance, the reported 99.1% accuracy is verified by the near-zero false negative rate and the near-perfect AUC of 0.9997 (Section 4.4), confirming that the model's performance is not an artefact of class distribution but reflects open discriminative ability across both classes.

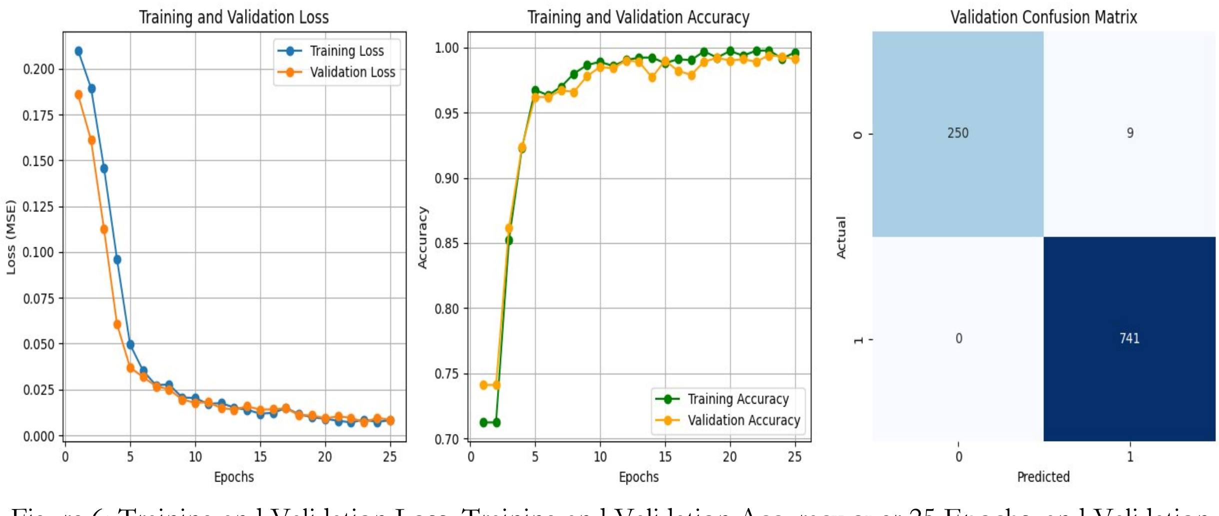

Figure 6: Training and Validation Loss, Training and Validation Accuracy over 25 Epochs, and Validation Confusion Matrix for the Dual-Tower CNN

4.4 ROC and Precision-Recall Curve Analysis

To broadly evaluate the discrimination capacity of the trained recommendation model, Receiver Operating Characteristic (ROC) and Precision-Recall (PR) curves were generated from the final validation epoch predictions. The model accomplished an AUC of 0.9997 and an Average Precision (AP) of 0.9999, both approaching the theoretical maximum of 1.0. These results confirm exceptional binary classification capability, with near-perfect separation between matching and non-matching candidate-job pairs.

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

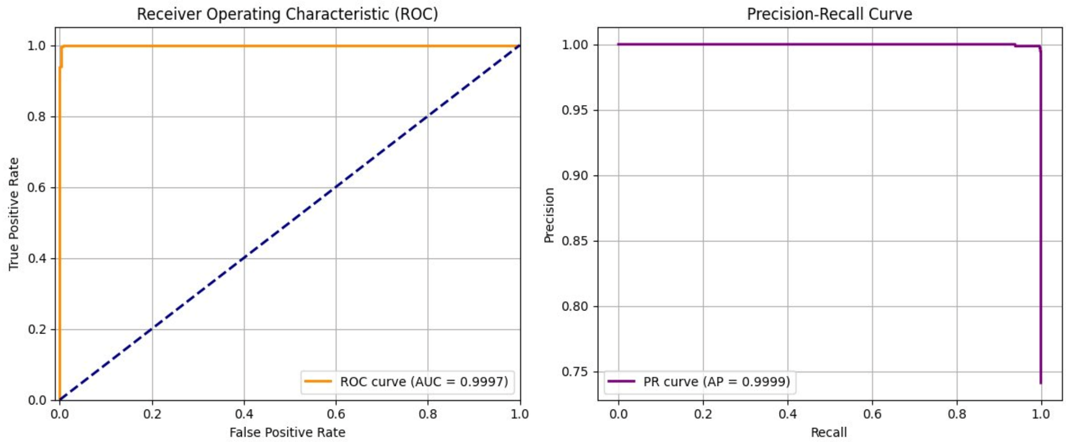

Figure 7: Receiver Operating Characteristic (ROC) Curve (AUC = 0.9997) and Precision-Recall (PR) Curve (AP = 0.9999) validating the superior discrimination capacity of the Dual-Tower CNN architecture

4.5. Recommendation Quality and Score Normalization

In a simulated production environment, the deployed model assesses a candidate profile against all 200 available job postings in under 0.15 seconds on a standard CPU node, with computational complexity scaling as O(C × J) where C represents the candidate tensor batch and J denotes pre-computed static job embedding. This latency profile confirms practical deployment viability.

This linear scaling maps the lowest-ranked candidate-job pair to a displayed score of 70% and the highest-ranked pair to 98%, with the majority of well-matched top recommendations displaying in the 94%-98% range as visible in the system interface (Figure 4.2). The relative ranking integrity of the underlying CNN predictions is fully preserved across this transformation.

5. Discussion

The empirical results establish the Dual-Tower CNN as an effective and scalable framework for mapping asymmetric candidate and job specification vectors into a shared semantic matching space. The architecture's metrics are found to be (99.1% accuracy, AUC = 0.9997) authenticate the core architectural design choices, predominantly the dual-tower embedding strategy, Conv1D feature extraction, and MLP-driven similarity computation.

5.1. Comparison with Manual Screening

The proposed system offers demonstrable efficiency advantages over conventional manual screening processes. The traditional committee-based recruitment in Nigerian universities typically involves recruitment panels to individually review thousands of application documents over epochs spanning several weeks to months, with no standardized scoring framework and significant susceptibility to evaluator-to-evaluator variability. By contrast, the deployed Dual-Tower CNN evaluates a candidate profile against the full 200-job posting database in under 0.15 seconds, generating ranked, score-normalized recommendations immediately. However, beyond speed, the system eradicates subjective bias by encoding all evaluation criteria as fixed quantitative features, ensuring that every candidate is assessed against identical job requirements under identical computational conditions.

6. Conclusion

This study has successfully designed, implemented, and evaluated a Machine Learning-Driven Recruitment Recommendation System tailored to the employment environment of Nigerian universities. The proposed Dual-Tower CNN architecture achieves near-perfect classification performance, with 99.1% validation accuracy, an AUC of 0.9997, and an Average Precision of 0.9999, substantially outperforming classical machine learning baselines including Logistic Regression, SVM, and Random Forest. The system's sub-0.15-second inference latency, coupled with an intuitive score-normalized recommendation interface, confirms practical deployment feasibility within institutional HR contexts. The research contributes a scalable, transparent, and objective technological pathway for modernizing academic recruitment across Nigerian universities, with clear generalizability to other developing-country higher education environments.

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

References

[1] D. Vrontis, M. Christofi, V. Pereira and S. Tarba, "Artificial intelligence, robotics, advanced technologies and human resource management: a systematic review," *The International Journal of Human Resource Management,* vol. 33, no. 6, 2022.

[2] D. Ç. Ertuğrul and S. Bitirim, "Job recommender systems: A Systematic Literature Review, Applications, Open issues and challenges," *Journal of Big Data,* 2025.

[3] G. Xiao, J. Ling, R. Chongzhi and C. Zianggang, "Together is better: Knowledge-aware Model with Resume Fusion for Online Job Recommendation," *ACM,* p. 1, 2025.

[4] P. H. Pol and K. Sanjiv, "The Impact of Artificial Intelligence on Recruitment Efficiency And Onboarding Effectiveness," *IOSR Journal of Multidisciplinary Research (IOSR -JMR),* vol. 2, no. 3, p. 1, 2025.

[5] Y. Rui-Ern, H. Su-Cheng and A. Shaymaa, "A Comprehensive Review on Machine Learning-Based Job," *International Journal on Robotics, Automation and Sciences,* vol. 7, pp. 2-4, 10 May 2025.

[6] M. Ismail, U. Haruna, G. Aliyu, I. Abdulmumin and S. Adamu, "An Autonomous Admission Recommender System Using Machine Learning Techniques," in *IEEE*, Lagos, 2020.

[7] H. Aliyu, Interviewee, *Registrar, Bayero University.* [Interview]. 25 February 2026.

[8] O. Olasehinde, A. A. Abiona, O. V. Johnson and D. Omilegan, "Design and Implementation of a Web-Based Internship Placement: A Case Study of Federal Polytechnic, Ile-Oluji, Nigeria," *International Journal of Scientific & Engineering Research,* vol. 13, no. 3, p. 2, 1 March 2022.

[9] K. Hadjer, M. Abdelkrim and H. Slimane, "Toward an Approach for Job Recommender System: Leveraging Hybrid Techniques," in *8th International Conference on Natural Language Processing and Information Retrieval*, Japan, 2024.

[10] S. Soni and S. Shah, "Job Recommendation Systems Through AI and Machine Learning," in *International Journal of Basic and Applied Sciences*, India, 2025.

[11] Y. Sun, F. Zhuang, H. Zhu, Q. He and H. Xiong, "Cost-Effective and Interpretable Job Skill Recommendation with Deep Reinforcement Learning," in *Web Conference*, 2021.

[12] A. Olowolayemo, K. Harun and T. Mantoro, "University Based Job Recommender & Alumni System," in *International Conference on Computing, Engineering, and Design (ICCED)*, 2018.

[13] E. Sholikhah, S. and R. Ilham, "The Application of Artificial Intelligence in Recruitment and Selection," *Jurnal Manajemen Bisnis Kewirausahaan,* vol. 4, no. 1, pp. 1-2, April 2025.

[14] J. C. Li, M. Namvar, G. P. Im and S. Akhlaghpour, "Machine Learning Based Decision-Making," *Saeed,* vol. 28, 2024.

[15] J. Du, J. Chen, X. Wang and Z. Zhao, "Efficient Data Classification and Prediction Using Random Forest (RF) and Gradient Boosting Machine (GBM)," in *International Conference on Cyber Security Intelligence and Analytics*, 2025.

[16] UNESCO, "UNESCO," UNESCO, [Online]. Available: https://www.unesco.org/en/digital-education/artificial-intelligence. [Accessed 12 April 2026].

[17] D. Kalniņa, D. Nīmante and S. Baranova, "Artificial Intelligence for Higher Education: Benefits and Challenges for Pre-service Teachers," *Frotiers in Edication,* vol. 9, p. 1, 2024.

[18] S. SARKAR, "Development of a Hybrid Recommendation System Using Collaborative Filtering and Content-Based Filtering Techniques," *ICONIC RESEARCH AND ENGINEERING JOURNALS,* vol. 8, no. 11, p. 4, 2025.

[19] J. Murel and E. Kavlakoglu, "What is collaborative filtering?," IBM, [Online]. Available: https://www.ibm.com/think/topics/collaborative-filtering. [Accessed 12 April 2026].

[20] F. Jauro, B. B. Alhassan, G. Aliyu and S. M. Toro, "Predicting child delivery mode using data mining by considering maternal," *Science Forum (Journal of Pure and Applied Sciences),* no. 18, pp. 31-35, 2019.

[21] F. Lee, "fangfang-lee," 2026. [Online]. Available: https://www.ibm.com/think/topics/logistic-regression. [Accessed 25 March 2026].

[22] A. C. LEON, "ScienceDirect," 1998. [Online]. Available: https://www.sciencedirect.com/science/chapter/referencework/abs/pii/B0080427073002649. [Accessed 25 March 2026].

[23] Geeksforgeeks, "/support-vector-machine-algorithm," 19 January 2026. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/. [Accessed 25 March 2026].

Received: 26/03/2026; Revised: 12/04/2026; Accepted: 17/04/2026; Published: 25/04/2026

[24] A. A. Basim, A. Mohammed and D. A. R. Shaza, "CNN-CNN: Dual Convolutional Neural Network Approach for Feature Selection and Attack Detection on Internet of Things Networks," *sensors,* p. 1, 2023.

[25] S. J. Russell and P. Norvig, Artificial Intelligence : A Modern Approach, Third ed., Boston: Prentice Hall, 2010.

[26] Y. Wang, "Design and Implementation of Student Job Matching System based on personalized Recommendation Rlgorithm," *ELSEVIER,* vol. 7, p. 1, 2025.

[27] D. Bitirim, E. Çelik and Selin, "Job recommender systems: A Systematic Lliterature Review, Applications, Open issues, and Challenges," *Journal of Big Data,* vol. xii, no. 140, p. 1, 2025.

[28] J. Chen, "neuralnetwork," 22 September 2019. [Online]. Available: https://www.investopedia.com/terms/n/neuralnetwork.asp.

[29] C. L. Jingqi, N. Morteza, P. I. Ghiyoung and A. Saeed, "Machine Learning Based Decision-Making: A Sensemaking Perspective," *Australasian Journal of Information Systems,* pp. 1-2, 2024.