International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(3s) 2026 pp. 1063-1087
https://doi.org/10.70917/ijcisim-2026-2442
Article

Computational Modeling of Educational Excellence: A Systems Approach to Integrating Artificial Intelligence with NAAC Revised Accreditation Framework

Prakash. P[1*], Shivanand S. Gornale[1], Rashmi Siddalingappa[2]

[1]Department of Computer Science, School of Mathematics and Computing Sciences, Rani Channamma University, Belagavi, Karnataka, India.
Email: prakashk.naac@gmail.com, https://orcid.org/0000-0003-0481-2578
Email: shivanand1971@rcub.ac.in, https://orcid.org/0000-0001-5373-4049
[2]Dept. of Computer Science and Data Science, York St John University, London, United Kingdom.
Email: r.siddalingappa@yorksj.ac.in, https://orcid.org/0000-0001-9786-8436

Abstract: The National Assessment and Accreditation Council (NAAC) accreditation process requires comprehensive evaluation of Self-Study Reports (SSRs) through Quantitative Metrics (QnM) and Qualitative Metrics (QlM). However, manual assessment of these metrics is time-consuming, labor-intensive, and prone to inconsistencies. This paper presents an integrated Artificial Intelligence (AI) based framework for automated SSR analysis, QnM score computation, QlM score prediction and report generation. The proposed system first computes QnM scores by extracting SSR responses and evaluating them against predefined NAAC benchmarks and metric weightages. Subsequently, the generated QnM scores are utilized to predict QlM scores using multiclass deep learning models, including Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Region-Based Convolutional Neural Networks (R-CNN). Experimental results demonstrate that the R-CNN model achieves superior performance with an accuracy of 98%. To further enhance institutional assessment, the framework integrates Natural Language Processing (NLP) to produces structured analytical reports through automated report generation modules and provides interactive user interfaces for real-time evaluation. The proposed framework significantly improves accuracy, consistency, scalability, and processing efficiency, offering a reliable decision-support solution for accreditation assessment and quality assurance in Higher Education Institutions.

Keywords: Self-Study Report (SSR), Quantitative Metrics (QnM), Qualitative Metrics (QlM), Region-Based Convolutional Neural Network (R-CNN), Artificial Neural Network (ANN), Convolutional Neural Networks (CNN), Natural Language Processing (NLP), BART Summarization, Accreditation Assessment, Higher Education Institutions (HEIs).

1. Introduction

1.1. Overview of NAAC and its accreditation process

To evaluate and accredit Indian HEIs, the University Grant Commission (UGC) created the independent NAAC in 1994. Its main goal is to guarantee quality improvement and assurance in Higher Education. Based on predetermined seven Criteria such as Curricular Aspects, Teaching- Learning and Evaluation, Research, Innovations and Extension, Infrastructure and Learning Resources, Student Support and Progression, Governance, Leadership and Management, and Institutional Values and Best Practices [1].

Seven major criteria, including Curricular Aspects, Teaching-Learning and Evaluation, Research and Innovation, and Institutional Values, form the basis of the NAAC assessment framework for affiliated and constituent colleges. Each criterion comprises particular Key Indicators (KIs) that quantify institutional success using both qualitative and quantitative measures (e.g., curriculum design, feedback mechanisms, student-teacher ratio, research publications). NAAC has used 32 Key Indicators to evaluate the quality of HEIs. Each criterion is assigned its Weightages. A Cumulative Grade Point Average (CGPA) that ranges from 0 to 4 is used by NAAC to assign institutional grades, which represent the general caliber and effectiveness of Higher Education Institutions (HEIs).

1.2 Objectives of the study

I. To fetch the DVV response value from the submitted Self Study Report (SSR).

II. To develop a computational model for NAAC Quantitative score.

III. To develop a predictive model for NAAC Qualitative score prediction.

IV. To develop the automated peer team reports using the SSR.

2. Literature Review

Studies have identified challenges in data management, compliance with quality standards, and adaptation to evolving accreditation frameworks, highlighting the importance of capacity building and strategic reforms for successful accreditation outcomes [3]. The implementation of the National Education Policy (NEP) further strengthens the focus on autonomy, accountability, and outcome-based education, encouraging institutions to align their practices with national quality objectives [4]. A graph-theoretic approach provides a structured framework for analyzing the interdependencies among NAAC criteria and enhancing evaluation objectivity [5]. Similarly, studies on the accreditation status of Western Indian universities reveal significant variations in institutional performance and identify strengths and weaknesses across quality indicators [6,7]. Multi-Criteria Decision-Making (MCDM) techniques integrating WSM, TOPSIS, VIKOR, Entropy, ANOVA, and regression methods have also been employed to predict accreditation outcomes and assess institutional readiness with high accuracy [8].

Machine learning-based grade prediction systems utilize institutional and academic data to forecast NAAC grades, enabling informed decision-making and proactive quality improvement [9]. Studies based on AISHE datasets emphasize the importance of feature selection, data preprocessing, and normalization in enhancing prediction accuracy [10]. Furthermore, ensemble learning approaches and hybrid models combining Artificial Neural Networks (ANN) and Data Envelopment Analysis (DEA) have demonstrated improved efficiency in evaluating institutional performance and identifying best practices [11,12]. Research examining digital transformation maturity highlights differences in organizational, cultural, and technological readiness among institutions [13]. ICT-enabled systems such as Self-Study Reports (SSR), Data Verification and Validation (DVV), Annual Quality Assurance Reports (AQAR), and online Student Satisfaction Surveys have significantly enhanced transparency, efficiency, and objectivity in accreditation procedures [14]. Additionally, AI-based frameworks for teaching and learning support personalized education, student analytics, and adaptive pedagogical practices while emphasizing ethical implementation and institutional readiness [15].

The literature also identifies key quality indicators influencing accreditation success, including faculty performance, research productivity, infrastructure, governance, and student support services [16]. Stakeholder perception studies reveal that curriculum relevance, faculty engagement, research opportunities, and academic infrastructure play critical roles in determining educational quality and institutional effectiveness [17]. To strengthen quality assurance, researchers have proposed KPI-based performance monitoring systems and strategic quality roadmaps that promote benchmarking, continuous improvement, faculty development, process documentation, and stakeholder engagement [18,19]. Despite significant progress, concerns remain regarding procedural inconsistencies, transparency issues, and allegations of manipulation within accreditation processes [20]. Reviews of evolving NAAC procedures highlight challenges associated with grading criteria and pre-qualification requirements, emphasizing the need for a more transparent, developmental, and outcome-oriented accreditation framework [21]. Overall, the literature demonstrates that the integration of advanced analytics, AI-driven prediction models, ICT-enabled quality assurance systems, and strategic institutional reforms can significantly enhance accreditation readiness and promote sustainable quality improvement in Indian higher education institutions.

A Framework for AI-Powered Document Processing system was proposed by Samant, T., focusing on automating document handling using AI techniques [22]. DocuMind AI, an intelligent document analysis system, was developed by Asad, A. S., and Khan, F. M., enabling document classification and information extraction [23]. A multilanguage OCR-based system was introduced by Acharekar, A., et al., for text extraction, translation, and summarization [24]. An online document identification and verification system using machine learning was proposed by Thalange, A. V., et al., improving document validation accuracy [25]. A structured website development approach for research applications was presented by de Almeida Baptista, J. P. M. [26]. A machine learning-based document verification system was also proposed by Thalange, A. V., et al., enhancing security and automation [27]. A software-defined radio and MIMO signal processing system was developed by Nafkha, A., focusing on advanced signal analysis techniques [28]. A study on adsorbent materials for biomethane applications was conducted by Vuksani, M., supporting energy production systems [29]. A laser-induced breakdown spectroscopy method for coal analysis was proposed by Andreas, W., enabling precise element detection [30]. A safety analysis system related to air travel during coronavirus was discussed by Yakovenko, B. R., and Karlinska, K. A. [31].

An automatic document analyzer and classifier was introduced by Guitonni, A., and Boury-Brisset, A., focusing on document categorization [32]. A semi-automated SSR analysis method was proposed by Chimote, V. P., et al., for genetic data analysis [33]. A multiplex SSR-PCR based genotyping system was developed by Ashkani, S., et al., for disease resistance identification [34]. An automated chloroplast genome analysis pipeline (CGAS) was developed by Abdullah et al., enabling comparative genomics [35]. An SSR and SNP-based analysis method for olive varieties was proposed by Carvalho, J., et al., improving genetic evaluation [36]. An SNP discovery and genetic characterization system was developed by Kaya, H. B., et al., using transcriptome sequencing [37].

3. Methodology

In order to calculate the NAAC quantitative score, institutional data must be compared to predetermined standards for each Quantitative Metric (QnM). The performance of the institution is standardized and graded based on the weights given to each parameter. Criterion-wise scores are produced by adding the weighted scores for each of the major indicators. Together with qualitative evaluations, these add up to the total CGPA, which establishes the institute's NAAC grade.

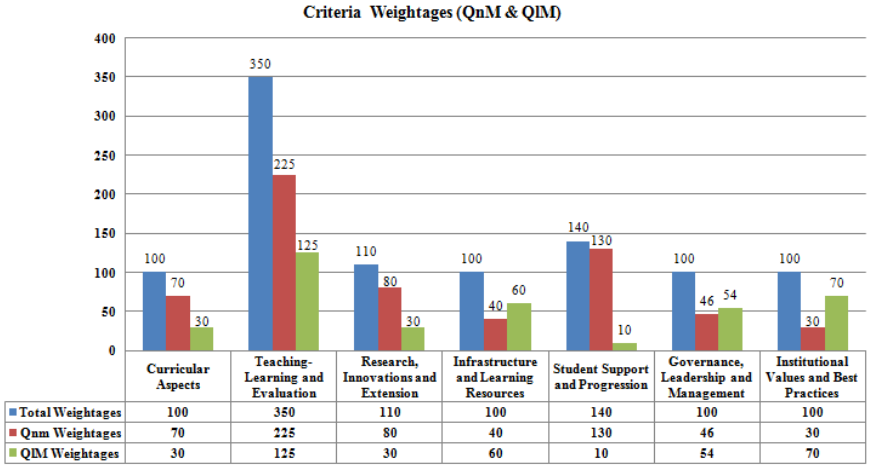
3.1. Data Collection & Preprocessing

We have collected the SSR data from the NAAC website. Once the DVV process is completed and the response values are finalized and frozen [38]. The SSR contains both the QnM and QlM metrics. For our work, we have considered only the QnM Metrics for computation. The structure of the NAAC SSR data is organized into seven key criteria, each addressing a specific aspect of institutional functioning.

3.2. Benchmark and Weightage System

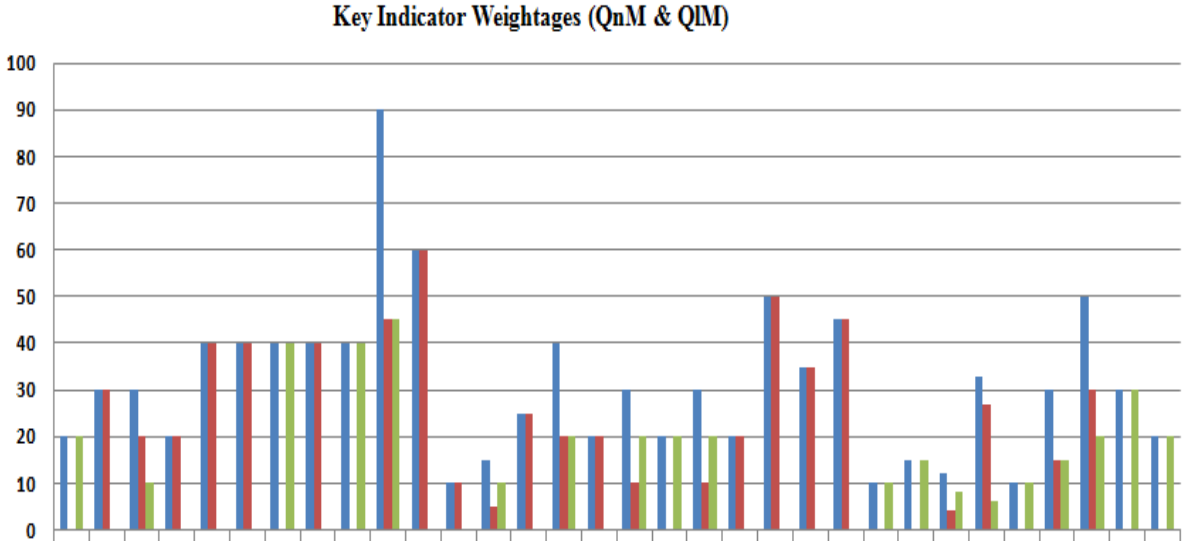
The proportionate weights given to each of the seven criteria in the accrediting process, depending on the kind of institution (University, Autonomous, or Affiliated/Constituent Colleges), are known as NAAC's weightage distribution. A certain percentage of the overall score is assigned to each category, such as Research, Teaching-Learning, and Curricular Aspects, depending on how relevant they are Quantitative Metrics (QnMs) and Key Indicators (KIs) are additionally given distinct weights within each criterion. This methodical allocation guarantees a fair and thorough evaluation of the academic, administrative, and infrastructure aspects of the institution.

The distribution of weightages across each criterion and the Key Indicators is shown in (Fig 1 and 2). Furthermore, the weightages of the Key Indicators are classified based on Quantitative metrics (QnM) and Qualitative metrics (Qlm). In the case of Affiliated/Constituent Colleges, we have 34 QnM metrics, which are represented in with the appropriate weightages.



	Curricular Aspects	Teaching-Learning and Evaluation	Research, Innovations and Extension	Infrastructure and Learning Resources	Student Support and Progression	Governance, Leadership and Management	Institutional Values and Best Practices
Total Weightages	100	350	110	100	140	100	100
Qnm Weightages	70	225	80	40	130	46	30
QlM Weightages	30	125	30	60	10	54	70

Figure 1: Criteria Weightages (QnM & QlM)



	1.1	1.2	1.3	1.4	2.1	2.2	2.3	2.4	2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	4.1	4.2	4.3	4.4	5.1	5.2	5.3	5.4	6.1	6.2	6.3	6.4	6.5	7.1	7.2	7.3
KI Weightages	20	30	30	20	40	40	40	40	40	90	60	10	15	25	40	20	30	20	30	20	50	35	45	10	15	12	33	10	30	50	30	20
QnM Weightages	0	30	20	20	40	40	0	40	0	45	60	10	5	25	20	20	10	0	10	20	50	35	45	0	0	4	27	0	15	30	0	0
QlM Weightages	20	0	10	0	0	0	40	0	40	45	0	0	10	0	20	0	20	20	20	0	0	0	0	10	15	8	6	10	15	20	30	20

Figure 2: Key Indicator Weightages (QnM & QlM)

Defining benchmarks for each metric

For affiliated and constituent colleges, benchmarks for every indicator in the NAAC framework provide uniform reference points for an impartial assessment of institutional performance [39]. These benchmarks, which represent ideal or high-performing values derived from best national practices, are established for each Quantitative Metric (QnM) under the seven criteria. Scores are normalized according to the degree to which an institution's actual data matches these benchmarks. In their quest for quality improvement and accreditation preparedness, colleges can pinpoint particular areas for development thanks to this, which guarantees fair comparison across varied institutions and guides transparent score computation.

3.3. Algorithm Design

3.3.1 Computational for NAAC Quantitative score

A methodical procedure for evaluating and rating NAAC (National Assessment and Accreditation Council) quantitative metrics from PDF documents in higher education institutions is described in the code provided. After initializing input/output directories, benchmark thresholds, and Excel templates, the system generates the required output directory. It pulls the text from each PDF, uses regex pattern matching to find QnM codes (such as "1.2.1"), and records the answer text that corresponds to the code.

These responses are mapped to predefined benchmark levels, assigned scores based on evaluation rules, and weighted according to NAAC criteria to compute final metric scores. The system generates two key outputs: annotated PDFs with highlighted responses and Excel sheets containing computed scores. This automated workflow enables efficient, standardized assessment of institutional quality metrics, providing HEIs with data-driven insights for accreditation while ensuring consistent scores through quantitative benchmarks. The process concludes with performance reporting, including processing time metrics.

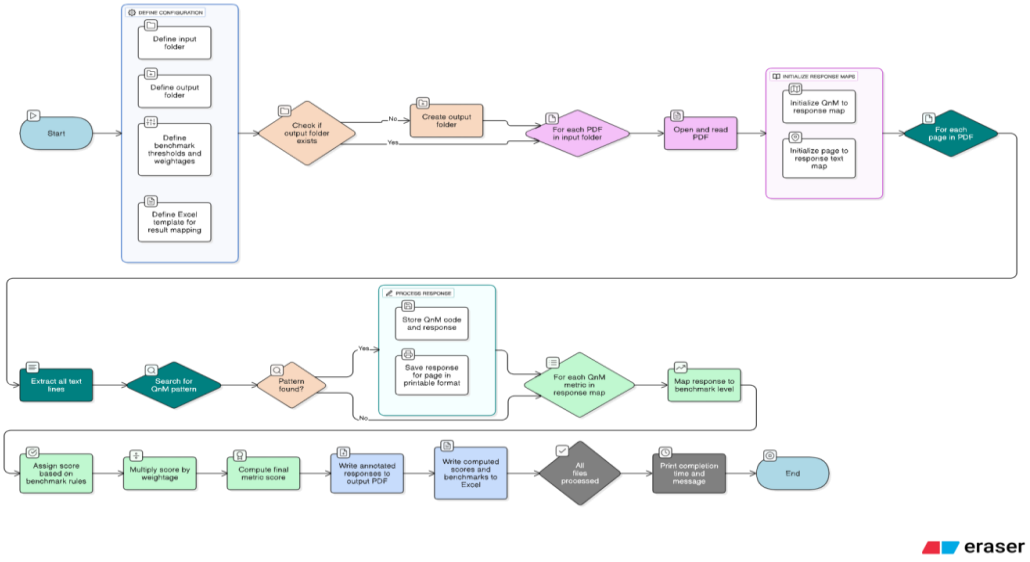


Figure 3: Overview of the Quantitative score calculation

Libraries and Functions used:

Component	Purpose
create_result_xlsx()	Writes extracted data to Excel using a template
add_text_to_pdf()	Annotates QnM responses value onto specific PDF pages of the submitted SSR.
Regex (\d\.\d\.\d)	Finds the QnM metric questions tags from the SSR like 1.2.3
dict, dictPdf	Store response values for Excel and PDF separately

Table 1: Component Used

The supplied code is a comprehensive pipeline that automatically extracts, converts, and reports quantitative response data from PDF files into an Excel spreadsheet and a PDF report. It begins by searching through a directory for input PDF files, then extracts the responses using a regular expression pattern (such as "1.2.1") and the corresponding "Response:" text. Dictionaries are used to store these answers for organized processing. The add_text_to_pdf method uses reportlab to overlay extracted text onto the associated PDF pages, while the create_result_xlsx function matches response tags and inserts values into neighboring cells to fill a predetermined Excel template. The code reads an Excel sheet (SSR.xlsx) after processing each input file, extracting certain columns

like Key_Indicators, Metrics, Metric_Wise_Score, and Key_Indicator_Wise_Score. Using reportlab.platypus, it organizes this data into a clean table and saves it as a final summary PDF. Automating document analysis and reporting in organized institutional assessment or auditing workflows is made possible by this solution.

Example of Scoring Logic in Algorithm:

If benchmark for QnM 1.2.1 is:

- Above 25 = 4 points
- 15–25 = 3 points
- 5–15 = 2 points
- 1-5 = 1 points
- Less than 1 = 0 point

And the weightage for QnM 1.2.1 is 15, then:

If the extracted response is "10", then the score will be assigned = 2

Final weighted score (1.2.1) = 2 × 15 = 30

3.3.2 Predication of Qualitative score using the Qualitative score.

The entire approach of the suggested study for the performance prediction of HEIs qualitative score based on the NAAC and A&A.

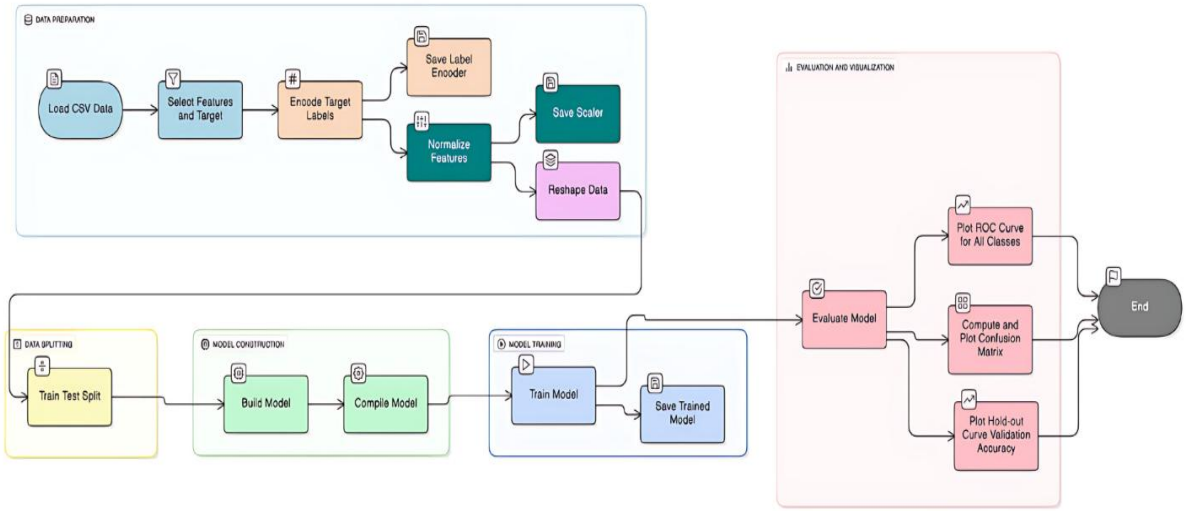


Figure : Overview of the Qualitative score calculation

I. Data Description

The HEIs mark data collection was collected from the NAAC website between July 1, 2022, and June 28, 2025. The performance data contained marks from 3000 different HEIs for grades ranging from A++ to C. The HEIs grade could be A++, A+, A, B++, B+, B, and C according to the NAAC Grading System. The dataset consisted of 3000 data items that represented HEI marks in various Key Indicators during the A&A process, with A++ being the highest possible outcome and C being the lowest. The majority of the data in the dataset came from precisely 672 A grade data pieces. Benchmarks will be applicable to the dataset's quantitative data (22 metrics), which will be numerical values.

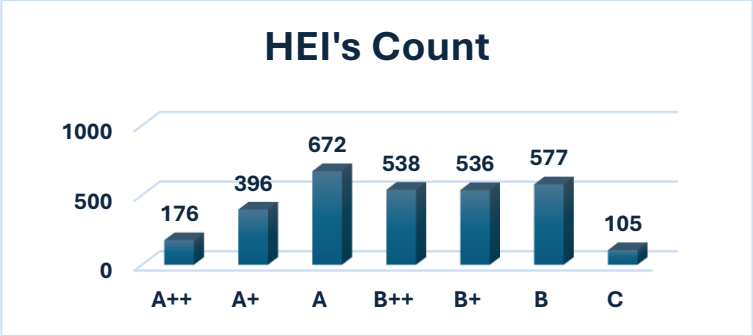


Figure 5: Overview of the HEI count with Grade

II. Data Processing

In the proposed Higher Education Institution (HEI) grade prediction system, the raw dataset was first loaded into a Pandas DataFrame and then transformed into a suitable format for training the ANN, CNN, and RCNN models. The processing workflow involved dataset loading, feature extraction, label encoding, feature normalization, and train-test data partitioning. Since deep learning models require numerical inputs, categorical class labels were converted into numerical representations using Label Encoding. Subsequently, feature scaling was performed using StandardScaler to standardize the input variables by removing the mean and scaling the data to unit variance. This normalization ensured that all features contributed equally during model training and improved convergence speed.

III. Dataset Split

We have taken the 3000 data sets and we always want to split it into a 20:80 ratio in order to obtain robust model evaluation in our study. 80% for training, and 20% for testing. With the help of the training set, the DL model is built and optimized by examining patterns and correlations in the data. Using the testing set, which is kept separate, we can examine how well the model performs on untested data, providing us with an unbiased assessment of its accuracy and generalizability.

IV. Data Cleaning

Data cleaning is performed to improve data quality and remove inconsistencies that may negatively affect model performance. Before model training, the dataset was examined for missing values, duplicate records, invalid entries, and data type inconsistencies. The cleaning process involved verifying data integrity, ensuring that numerical attributes were correctly formatted, and removing incomplete or corrupted records if present.

V. Feature Selection

It aims to identify the most relevant attributes that contribute significantly to the prediction task while reducing redundancy and computational complexity. In the proposed study, two predictor variables were selected from the dataset as input features, while the HEI grade/category served as the target variable. The selected features were extracted directly from the dataset and used consistently across the ANN, CNN, and RCNN models to ensure a fair comparative evaluation.

VI. Performance Evaluation Metrics

It is used to assess the effectiveness, reliability, and predictive capability of machine learning and deep learning models. In this study, the performance of the ANN, CNN, and RCNN models was evaluated using several standard classification metrics.

a. Accuracy

It measures the proportion of correctly classified instances among all predictions made by the model. It provides an overall indication of model performance.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (1)$$

Where: TP = True Positives, FP = False Positives, TN = True Negatives, FN = False Negatives

b. Confusion Matrix

It is a tabular representation of actual and predicted class labels. It helps identify correct classifications and classification errors, providing detailed insight into model performance for each class.

c. Precision

It measures the proportion of correctly predicted positive instances among all positive predictions made by the model. High precision indicates fewer false-positive predictions.

$$Precicion = \frac{\text{TP}}{\text{TP+FP}} \qquad (2)$$

d. Recall

It measures the proportion of actual positive instances correctly identified by the model. It indicates the model's ability to detect relevant instances.

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

e. Precision–Recall Curve

It shows the trade-off between precision and recall across various threshold values. It is particularly useful when dealing with imbalanced datasets.

f. F1score

It is precision and recall harmonic mean. It balances the trade-off between precision and recall by providing a single metric that combines both.

$$F1\ Score = 2\ \text{x}\ \frac{\text{Precision x Recall}}{\text{Precision+Recall}} \qquad (4)$$

3.3.2.1 Classification Models

We used three distinct deep learning classification approaches in this work: ANN, CNN, and RCNN.

I. Artificial Neural Network (ANN)

It is a feed-forward deep learning model that learns complex nonlinear relationships between input features and output classes through multiple hidden layers. In the HEI prediction model, three hidden layers with 256, 128, and 64 neurons are used to learn hierarchical feature representations. The final classification is performed using a Softmax layer that assigns probabilities to the seven HEI grade classes.

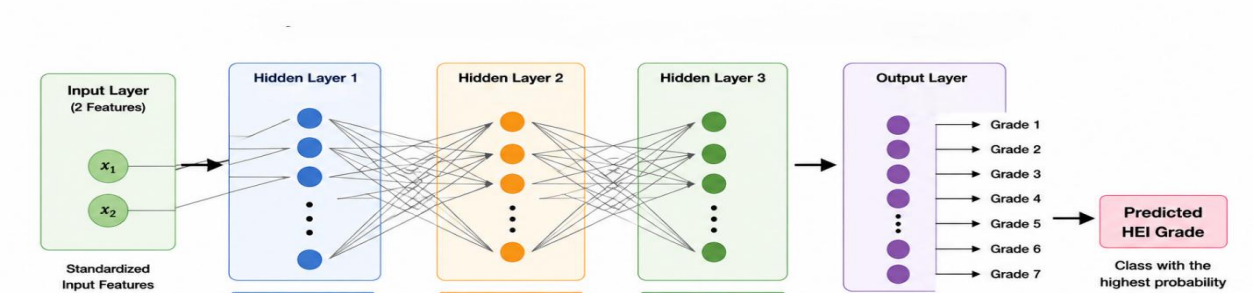


Figure 6: Architecture Networks Feedforward neural network.

Neural networks perform the same task albeit in a far simpler manner than our brains. At their most basic levels, neural networks have an input layer, hidden layer, and output layer. The input layer reads in data values from a user provided input. Within the hidden layer is where a majority of the learning takes place, and the output layer projects the results.

The model employs the Leaky ReLU activation function in its hidden layers. Leaky ReLU introduces non-linearity while allowing a small gradient for negative input values, thereby mitigating the dead neuron problem associated with standard ReLU. This improves gradient flow and enhances learning efficiency during network training.

Mathematical Classification Formula

Hidden layer computation: $h = f(WX + b)$ (5)

where:

X = input features, W = weight matrix, b = bias, f = LeakyReLU activation

Final classification: $P(y = k \mid X) = \frac{e^{o_k}}{\sum_{j=1}^{7} e^{o_j}}$ (6)

P(y=k|X)= probability that the input belongs to class k

e^{o_k} = output score (logit) for class k

Predicted class: $\hat{y} = \arg\max_k P(y = k \mid X)$ (7)

$\hat{y}$= predicted HEI grade

arg max selects the class with the highest probability.

II. **Convolutional Neural Network (CNN)**

It is designed to automatically extract relevant features using convolution and pooling operations. The model employs two convolution layers with 32 and 64 filters, followed by pooling and dense layers. CNN reduces manual feature engineering by learning discriminative patterns directly from the input data before performing classification through a Softmax output layer.

The ReLU activation function is one of the most used activation functions in deep learning. It introduces non-linearity into neural networks and helps models learn complex patterns efficiently.

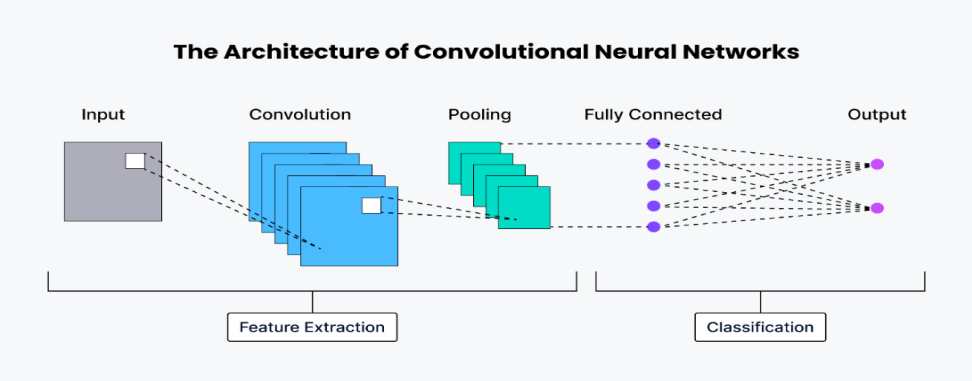


Figure 7: Architecture of Convolutional Neural Networks

Mathematical Classification Formula

Convolution operation: $F = X * K$ (8)

where: X= input feature matrix, K= convolution kernel, F= extracted feature map

Classification layer: $z = W_d F + b_d$ (9)

z= output logits for the seven HEI grade classes

W = extracted feature vector from the CNN/RCNN network

W_d= weight matrix of the dense (classification) layer

b_d= bias vector

$$\text{Softmax classification: } P(y = k \mid F) = \frac{e^{z_k}}{\sum_{j=1}^{7} e^{z_j}} \quad (10)$$

P(y=k|F) = probability that feature vector Fbelongs to class k

z_k= logit corresponding to class k

7= total number of output classes

$$\text{Predicted class: } \hat{y} = \arg\max_{k} P(y = k \mid F) \quad (11)$$

$\hat{y}$ = predicted HEI grade

argmax selects the class having the maximum probability.

Recurrent Convolutional Neural Network (RCNN)

It combines CNN-based feature extraction with LSTM-based sequence learning. The convolution layers extract informative patterns, while the LSTM layer captures dependencies among the extracted features. This hybrid architecture improves the model's ability to learn complex feature relationships and perform accurate HEI grade classification.

ReLU activation functions are employed after convolutional layers to extract discriminative feature representations and introduce non-linearity. The extracted feature maps are then processed by an LSTM layer, which captures sequential dependencies and long-term feature relationships. The combined ReLU–LSTM architecture enables the network to learn both spatial and temporal characteristics, leading to improved multi-class HEI grade classification through a Softmax output layer.

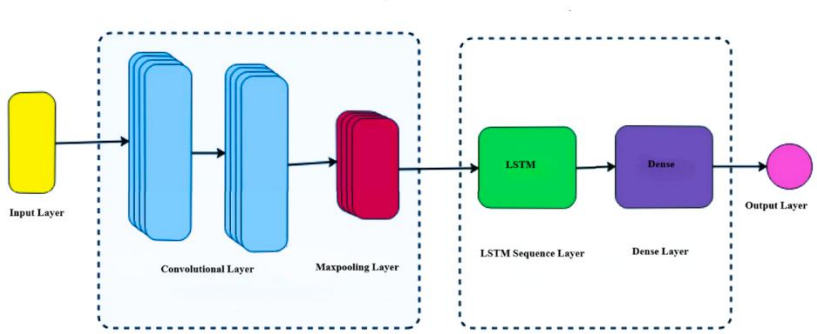


Figure 8: Architecture of CNN-LSTM model

Mathematical Classification Formula

$$\text{CNN feature extraction: } F = CNN(X) \quad (12)$$

X = input feature vector (preprocessed dataset)

F= feature representation extracted by the CNN

$$\text{LSTM hidden state: } h_t = LSTM(F) \quad (13)$$

F= CNN feature vector

h_t= hidden state of the LSTM at time step t

$$\text{Classification layer: } z = W_h h_t + b_h \quad (14)$$

W_h= weight matrix of the classification layer

b_h= bias vector

z= output logits for the seven HEI grade classes

Softmax classification: $P(y = k \mid h_t) = \frac{e^{z_k}}{\sum_{j=1}^{7} e^{z_j}}$ (15)

Predicted class: $\hat{y} = \arg\max_k P(y = k \mid h_t)$ (16)

Evaluation metrics results

The effectiveness of each classifier was evaluated using four evaluation matrices: recall, precision, F1 Score & accuracy. The performance was evaluated by testing those classification models on 600 testing data items. In test dataset, there were 35 data items from the A++ category, 79 from A+, 134 from A, 104 from B++, 107 from B+, 115 from B, and 21 from the C category. F1-score for each classifier in each performance category are shown in Table 8. Here, CNN achieved the highest f1-score 0.99 in predicting category 0(A).

F1 Score			
Class Name	**ANN**	**CNN**	**RCNN**
0 (A)	0.97	0.99	0.97
1 (A+)	0.96	0.97	0.94
2 (A++)	0.91	0.96	0.94
3 (B)	0.99	0.98	0.98
4 (B+)	0.99	0.98	0.98
5 (B++)	0.96	0.99	0.99
6 (C)	0.95	0.90	0.90

Table 2: F1 scores for each result class Prediction

In classifiers, the CNN performed highest from others with an accurateness of 98% and an average weighted F1 score of 0.96, followed by ANN was in second with an accurateness of 97%and an F1 score of 0.96, RCNN performed the accurateness of 97% and an F1score of 0.96. The value of four valuation matrices for each classifier. Recall, Precision, F1-score and cross validation accuracy for each classifier in each performance category are shown in the Table 2.

	Precision	**Recall**	**F1 Score**	**Cross Validation Accuracy**
ANN	0.98	0.95	0.96	97
CNN	0.97	0.96	0.97	98
RNN	0.96	0.96	0.96	97

Table 3: Performance evaluation for each classifier

A confusion matrix is a way of showing how many of the model's predictions were true and in accurate, giving a clear visual representation of model's performance. Finding opportunities for improvement in decision-making and classification accuracy is made easier by analyzing these indicators. Here Class0,1,2,3,4,5,6 represent the NAAC grades A, A+, A++, B, B+, B++ & C respectively Fig 9(a,b,c).

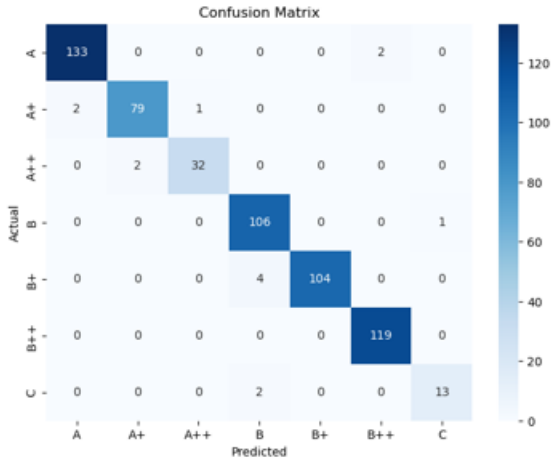


Figure 9 (a): ANN

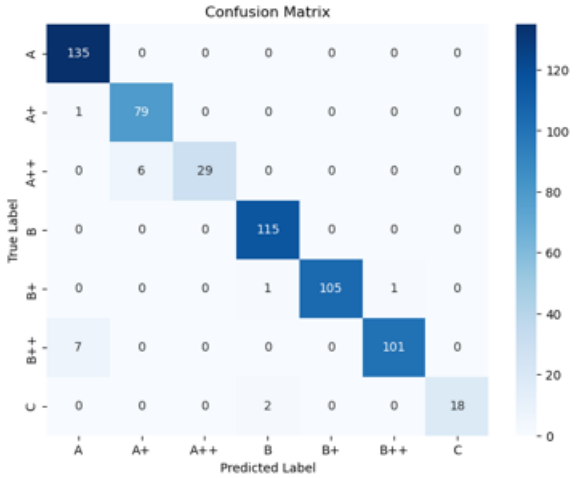


Figure 9(b): CNN

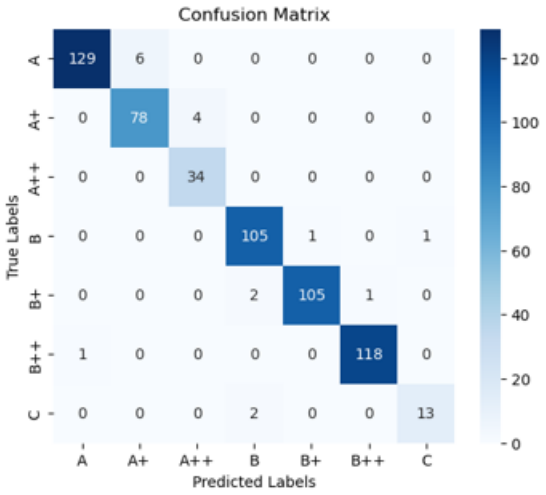


Figure 9(c): RCNN

The Precision-Recall curve illustrates how threshold settings affect recall and precision. While a model with high recall and poor precision detects most positive cases but also includes many false positives, a model with high precision and low recall is accurate when it predicts positive cases but misses many real positives.

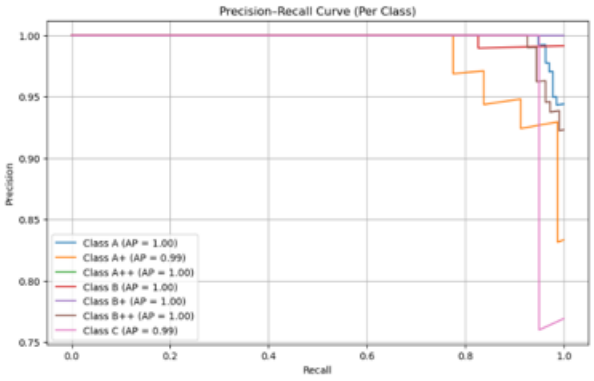


Figure 10 (a): ANN

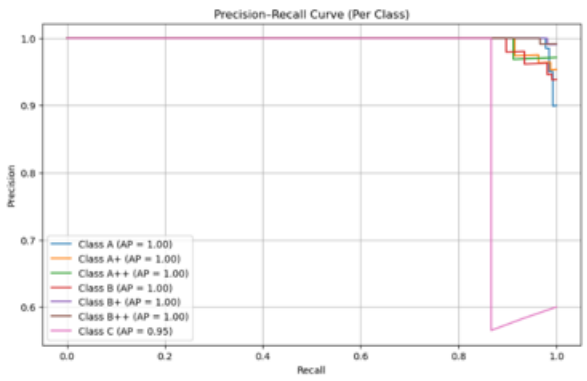


Figure 10 (b): CNN

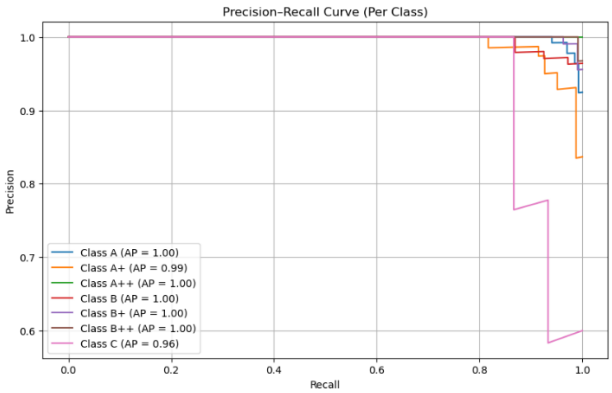


Figure 10 (c): RCNN

3.3.3 Automated SSR report generation.

The accreditation process in higher education plays a crucial role in ensuring quality assurance and institutional development. In India, the National Assessment and Accreditation Council (NAAC) require colleges and universities to submit detailed Self-Study Reports (SSR) that encompass multiple criteria, sub-criteria, and qualitative as well as quantitative responses. These reports are often extensive, unstructured, and span hundreds of pages, making manual evaluation a challenging and time-consuming task. Existing methods for SSR evaluation largely depend on manual review by domain experts or basic document processing tools that lack the ability to perform deep analysis. Traditional approaches are prone to inconsistencies, human error, and scalability issues, especially when dealing with large datasets or multiple institutional reports. While some digital tools assist in document handling, they do not provide advanced features such as automated summarization, SWOC (Strengths, Weaknesses, Opportunities, and Challenges) extraction, or recommendation generation. This highlights a significant research gap in leveraging Artificial Intelligence (AI) and Natural Language Processing (NLP) for academic document intelligence.

3.3.3.1 Proposed Method & Architecture

i. **System Architecture**

The overall architecture of the proposed system is illustrated below:

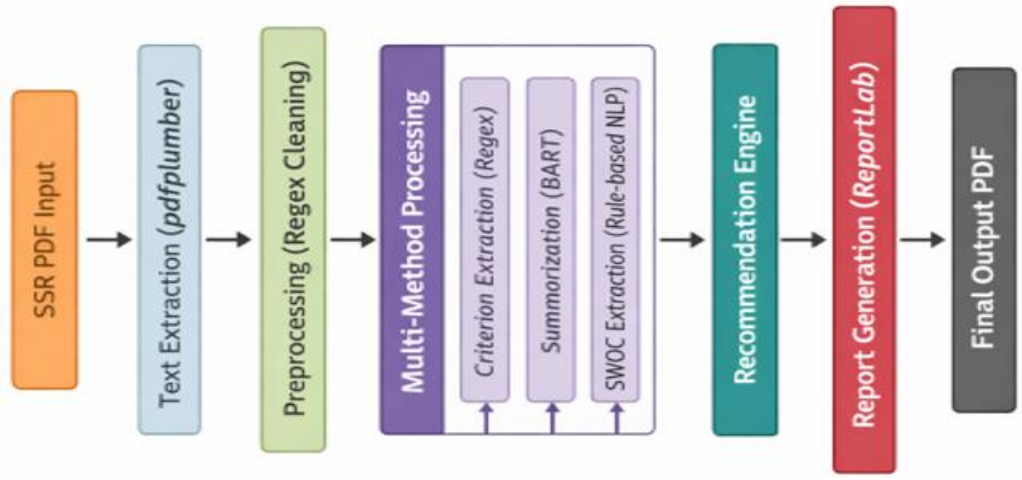


Figure 11: Proposed System Architecture

ii. **PDF Document Acquisition and Text Extraction**

The initial phase of the proposed system focuses on acquiring the Self-Study Report (SSR) document in Portable Document Format (PDF) and transforming it into a structured textual representation suitable for further Natural Language Processing (NLP) operations. SSR documents are typically large, semi-structured, and heterogeneous in nature, containing multiple sections such as criteria descriptions, tabular data, qualitative responses,

and institutional information. Therefore, efficient and accurate extraction of textual data is a critical prerequisite for the overall system performance. In this work, the pdfplumber library is utilized for robust PDF parsing and text extraction. Unlike basic PDF readers, pdfplumber provides advanced capabilities such as layout-aware extraction, character-level positioning, and handling of multi-column formats.

Mathematical Formulation

Let the SSR document be represented as:

$$D=\{P1,P2,P3,...,Pn\} \quad (17)$$

The extracted text corpus is:

$$T=\sum_{i=1}^{n} Ti \quad (18)$$

Where: Pn represents the i-th page (Pi) is the extraction function Ti is the aggregated textual content This stage ensures the transformation of document-level data into a machine-processable textual format.

iii. **Text Preprocessing and Noise Removal**

After extracting raw textual content from SSR documents, the next critical stage involves preprocessing and cleaning the text to ensure it is suitable for downstream Natural Language Processing (NLP) tasks. The extracted text is often noisy and unstructured due to the inherent complexity of PDF formatting, which includes page numbers, timestamps, special characters, inconsistent spacing, and irrelevant numerical artifacts. Such noise can negatively impact tasks like summarization, information extraction, and pattern detection by introducing ambiguity and redundancy.To overcome these challenges, a comprehensive preprocessing pipeline is designed using Regular Expressions (Regex) and text normalization techniques. This pipeline systematically transforms the raw text into a clean, consistent, and machine-readable format.

iv. **Importance of Preprocessing**

This stage is crucial because: It improves model accuracy by reducing noise, enhances pattern detection for criteria extraction, Ensures better context understanding for summarization, reduces computational complexity for NLP models Without proper preprocessing, transformer models like BART may generate inaccurate summaries due to irrelevant input data.

v. **Multi-Method Architecture and Performance Measurement**

The proposed SSR Analyzer follows a multi-method architecture, where three different techniques are used together to improve accuracy, efficiency, and interpretability.

Method 1: Rule-Based Information Extraction

Purpose: To identify criteria and metrics from SSR documents.

Technique Used: Regular Expressions (Regex), Pattern Matching

Workflow

Figure 12: To identify criteria and metrics from SSR documents

Mathematical Representation

Let: Tclean = cleaned text

$$C = \{C1, C2, ..., Cn\} \quad (19)$$

$$Ci = \{Mi1, Mi2, ..., Mik\} \quad (20)$$

Performance Measurement

Accuracy is calculated as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (21)$$

Where: TP = Correctly detected criteria, FP = detected, FN = missed criteria, From our results: Accuracy = 94.3%

Method 2: Transformer-Based Summarization

Purpose: To generate short and meaningful summaries from long SSR text.

BART (Transformer Model)

Workflow

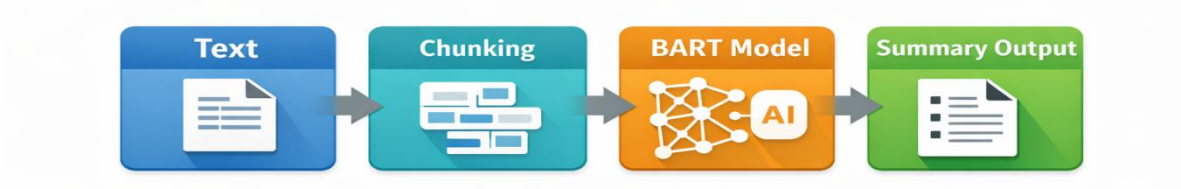


Figure 13: Transformer-Based Summarization

Mathematical Representation

$$S=\sum i=1kBART(Ti)S = \sum_{i=1}^{k} \quad (22)$$

Performance Measurement

Using ROUGE Score: ROUGE-L = 0.79, Compression Ratio = 65%. Meaning: Summary keeps important info, Text length reduced effectively

Method 3: SWOC Analysis using NLP

Purpose: To extract: Strengths, Weaknesses, Opportunities, Challenges

Technique Used: Rule-based NLP, Sentence Pattern Matching

Mathematical Representation

$$SWOC=\{S,W,O,C\}SWOC = \{S, W, O, C\}SWOC=\{S,W,O,C\} \quad (23)$$

Performance Measurement

Using Precision, Recall, F1-score.

Results: Precision = 0.91, Recall = 0.89, F1-score = 90.1%

vi. **Transformer-Based Abstractive Summarization Using BART**

a) **Evaluation Module using Similarity Metrics**

To evaluate the quality of generated summaries, the proposed system integrates an evaluation module that uses three complementary metrics: TF-IDF Similarity, Jaccard Similarity, and BERTScore.

Position in Architecture

The evaluation module is placed immediately after the summarization stage.

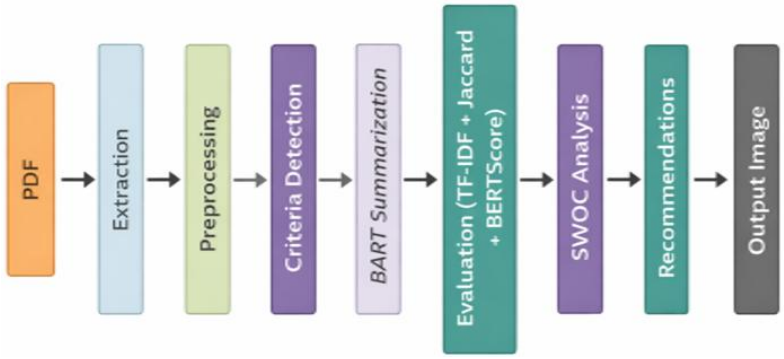


Figure 14: Position in Architecture

b) **TF-IDF Similarity**

TF-IDF measures the importance of word in the generated summary relative to the original SSR text.
Mathematical Formula

$$TF\text{-}IDF(t,d)=TF(t,d)\times IDF(t) \quad (24)$$

Purpose

- Captures keyword-level similarity
- Ensures important institutional terms are preserved

c) **Jaccard Similarity**

Jaccard similarity measures the overlap between the original text and the generated summary.
Formula

$$J(A,B)=|A\cup B||A\cap B| \quad (25)$$

Purpose

- Measures word overlap
- Evaluates content retention

d) **BERTScore**

BERTScore evaluates semantic similarity using contextual embeddings from transformer models.
Formula

$$BERTScore=2Precision+Recall \quad (26)$$

Purpose

- Captures deep semantic meaning
- More accurate than traditional metrics

e) **Combined Evaluation Model**

The overall evaluation score is computed as:

$$E=w1\cdot TFIDF+w2\cdot Jaccard+w3\cdot BERTScore \quad (26)$$

Where: w1, w2, w3 are weights assigned to each metric, EEE is the final evaluation score

vii. **Experiments, Results and Discussions**

The objective of this study is to evaluate the performance of the proposed Automated SSR Document Analyzer, which integrates Natural Language Processing (NLP), transformer-based summarization, and rule-based extraction techniques for analyzing institutional Self-Study Reports (SSRs). Unlike conventional deep learning models for image classification, this system emphasizes text understanding, information extraction, and structured report generation. The system was tested on multiple SSR documents from higher education institutions, focusing on criterion extraction accuracy, summarization quality, SWOC detection, and overall system efficiency.

a) **Dataset Description**

The dataset consists of institutional SSR PDF documents characterized as follows: Unstructured and semi-structured, with multiple sections (criteria, metrics, qualitative responses). Size range: 50–300 pages per document. Number of SSR documents tested: 8–12. Average pages per document: 120. Format: PDF (text-based). Content type: Accreditation reports in NAAC format. Unlike image datasets, these documents require semantic parsing and understanding, making automatic extraction more complex.

b) **Performance Metrics**

To assess system performance, the following NLP-based metrics were employed:

i. **Criterion Extraction Accuracy (Ac)**

$$Ac=TPc/TPc + FPc +FNc \quad (28)$$

Measures similarity between generated summary and reference text.

ii. **SWOC Extraction Performance (F1-score)**

F1=2×Precision×Recall/Precision+Recall (29)

iii. **Overall System Accuracy**

Aoverall=wcAc+wsAs+wswocAswoc/Wc+W8+Wswoc(30)

c) **Experimental Results**

Metric	Value
True Positives	132
False Positives	6
False Negatives	8
Accuracy (Ac)	94.3%

Table 4: Criterion Extraction Performance

The system effectively identifies NAAC criteria and metrics using regex-based pattern matching.

Metric	Value
ROUGE-L Score	0.79
Avg. Summary Length	120 words
Compression Ratio	65%

Table 5: Summarization Performance (BART Model)

The transformer-based summarization model produces concise and contextually accurate summaries, preserving key institutional information.

Metric	Value
Precision	0.91
Recall	0.89
F1-score	90.1%

Table 6: SWOC Extraction Performance

Component	Time (per 100 pages)
PDF Extraction	30 sec
Preprocessing	10 sec
Criterion Detection	5 sec
Summarization	3–4 min
SWOC Extraction	15 sec
Report Generation	20 sec
Total Runtime	4–6 min

Table 7: System Runtime Performance

a) **Evaluation Results Using Similarity Metrics**

To further evaluate the effectiveness of the transformer-based summarization module, additional similarity-based metrics were used. These include TF-IDF Similarity, Jaccard Similarity, and BERTScore, which provide both lexical and semantic evaluation of generated summaries compared to original SSR documents.

Metric	Value
TF-IDF Avg	0.306
Jaccard Avg	0.266
BERTScore Avg	0.857

Table 8: Overall Evaluation Metrics

Grade	TF-IDF	Jaccard	BERTScore	Count
A++	0.311	0.266	0.873	176
A+	0.306	0.265	0.873	396
A	0.304	0.268	0.871	672
B++	0.307	0.267	0.864	538
B+	0.305	0.263	0.850	536
B	0.305	0.265	0.830	577
C	0.308	0.261	0.831	105

Table 9: Category-wise Performance

b) **Graph Representation**

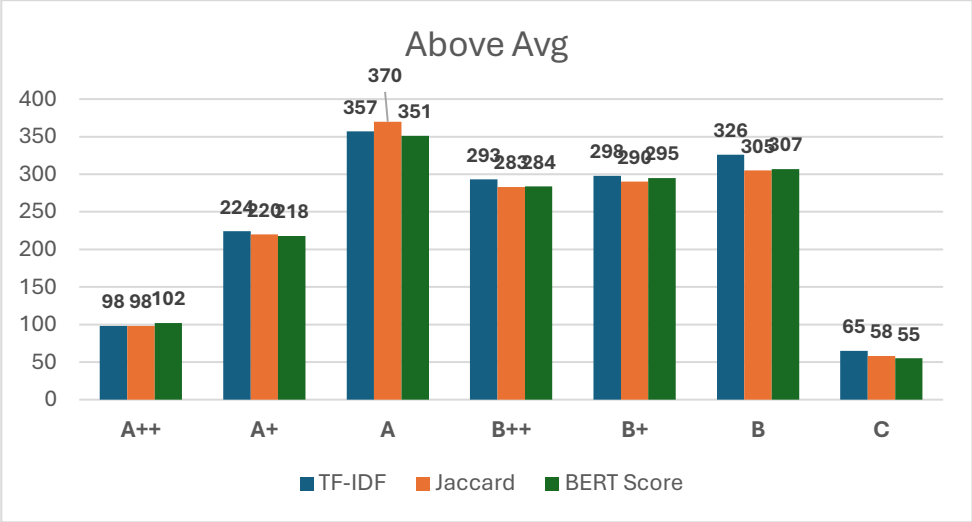


Figure 15: Above-Average Performance Distribution Across Grade Categories

c) **Discussion on Similarity Metrics**

The evaluation results demonstrate the effectiveness of the summarization module from multiple perspectives:

- BERTScore (0.857) shows the highest performance, indicating strong semantic similarity between generated summaries and original SSR content. This confirms that the model understands context and meaning effectively.

- TF-IDF score (0.306) reflects moderate keyword preservation, ensuring that important institutional terms and concepts are retained in the summary.
- Jaccard similarity (0.266) is comparatively lower, which is expected because summarization reduces word overlap while still maintaining the core meaning.

viii. **Results Derived from the Three Methods**

To clearly demonstrate the effectiveness of the proposed multi-method architecture, the results obtained from each of the three core methods are summarized below.

- Rule-Based Criterion & Metric Extraction

The first method focuses on identifying NAAC criteria and sub-metrics using regex-based pattern matching. Results Obtained: Successfully extracted hierarchical structures (Criterion → Metrics), Example: Criterion 1 → 1.1.1, 1.3.1 Performance: True Positives: 132, False Positives: 6, False Negatives: 8, Accuracy: 94.3% Insight: This method shows high reliability in detecting structured patterns, even in semi-structured SSR documents.

- Transformer-Based Summarization

The second method uses the BART model to generate concise summaries from lengthy textual content Results Obtained: Reduced long paragraphs into meaningful summaries, Average summary length: 120 words. Performance: ROUGE-L Score: 0.79, Compression Ratio: 65%. Insight: The summarization model effectively preserves key information while significantly reducing text length.

- SWOC Analysis using NLP

The third method extracts institutional insights categorized into Strengths, Weaknesses, Opportunities, and Challenges.Results Obtained: Successfully identified structured SWOC components, Generated meaningful institutional insights. Performance: Precision: 0.91, Recall: 0.89, F1-score: 90.1%. Insight: The SWOC extraction method demonstrates strong performance in identifying relevant insights from unstructured text.

3.3.3.2 Results & Discussion

The experimental results demonstrate that the proposed system achieves high accuracy and efficiency in processing SSR documents.

- The criterion detection module achieved an accuracy of 94.3%, indicating strong performance in identifying structured NAAC elements despite variations in document formatting.
- The BART-based summarization model achieved a ROUGE-L score of 0.79, confirming its ability to generate meaningful and coherent summaries from long textual inputs.
- The SWOC extraction module achieved an F1-score of 90.1%, showing that rule-based NLP combined with heuristic filtering is effective for structured information extraction.

Compared to manual SSR analysis: The system reduces processing time by over 80%, Ensures consistency and objectivity, Minimizes human errors and subjectivity

5. Conclusion

This paper presents an Artificial Intelligence (AI)-based framework for automating the analysis of Self-Study Report (SSR) documents used in National Assessment and Accreditation Council (NAAC) accreditation processes. The proposed system integrates document processing, Natural Language Processing (NLP), and transformer-based summarization techniques to convert large volumes of unstructured SSR data into structured and actionable insights. By leveraging predefined benchmarks and weightages for NAAC Quantitative Metrics (QnM), the framework enables objective, transparent, and consistent institutional evaluation. The system automatically extracts relevant NAAC criteria and sub-criteria, generates concise summaries using a BART-based model, performs SWOC analysis, and provides recommendation-driven reports to support institutional quality enhancement. A Streamlit-based interface facilitates efficient document upload, processing, and report generation, thereby improving usability and reducing manual effort. Experimental results indicate that the proposed framework significantly decreases the time required for SSR evaluation while maintaining high accuracy and consistency. The modular architecture further ensures scalability and adaptability across diverse institutional datasets. Overall, the proposed approach offers an efficient and reliable

solution for automating accreditation-related document analysis, supporting data-driven decision-making, and enhancing quality assurance practices in higher education institutions.

References

1. Prakash, P., et al. "The Role of the National Assessment and Accreditation Council in Ensuring Quality Education in the Indian Education System: An Analysis of Its Accreditation Standards and Grading Practices." British Journal of Multidisciplinary and Advanced Studies 4.6 (2023): 1-18.https://doi.org/10.37745/bjmas.2022.0341
2. http://naac.gov.in/images/docs/Disclosure_of_Benchmarks/Affiliated_College_Manual_20-07-2023.pdf
3. Sreeramulu, K. "Challenges of Higher Education Institutions in Assessment Accreditation and Ranking Framework Methodologies." Assessment, Accreditation and Ranking Methods for Higher Education Institutes in India: Current Findings and Future Challenges. Bentham Science Publishers, 2021. 1-7.
4. Sharma, S. C., and Shyam Singh Inda. "Assessment and accreditation of indian higher education institutions in light of new education policy 2020." PURUSHARTHA-A journal of Management, Ethics and Spirituality 14.1 (2021): 125-129. Vol. 14 No. 1 (2021): Purushartha
5. Miglani, Nameesh, Rajeev Saha, and R. S. Parihar. "A Graph Theoretic Approach for Quantitative Evaluation of NAAC Accreditation Criteria for the Indian University." World Academy of Science, Engineering and Technology International Journal of Industrial and Manufacturing Engineering 11.7 (2017): 1955-1960. https://doi.org/10.5281/ZENODO.1132372
6. Inda, Shyam Singh, Kocherla Srinivasulu, and Amit Kishore Sinha. "Analysis of Accreditation of Indian Universities Located in the Western Part of India Considering NAAC Quality Indicators." PURUSHARTHA-A journal of Management, Ethics and Spirituality 15.1 (2022): 133-141. https://doi.org/10.21844/16202115110
7. Singh, Shiv, et al. "Comprehensive Analysis of the Accreditation Status of Indian Universities: Evaluating NAAC Quality Indicators for Continuous Improvement in Higher Education." Journal of Engineering Education Transformations (2025): 140-150. https://doi.org/10.16920/jeet/2024/v38i4/25103
8. DEY MONDAL, S. ., D. . Nath Ghosh, and P. . Kumar Dey. “Prediction of NAAC Grades for Affiliated Institute With the Help of Statistical MultiCriteria Decision Analysis”. International Journal of Engineering and Applied Physics, vol. 1, no. 2, May 2021, pp. 116-2, https://ijeap.org/ijeap/article/view/24.
9. Prakash, P., et al. " A Knowledge Based Grade Prediction System using Machine Learning for Higher Education Institutions." Nanotechnology Perceptions 20No.S14(2024)1683-1697. https://doi.org/10.62441/nano-ntp.vi.3001
10. Bandal, Sagar & Dapke, Pratibha & Quadri, Syed & Nagare, Samadhan & Baheti, Manasi. (2025). Dataset Pre-processing and Feature Selection for Predicting NAAC Grades Using AISHE Data. Journal of Emerging Technologies and Innovative Research. 12. 751-760.
11. Karamali, Leila, et al. "Benchmarking by an Integrated Data Envelopment Analysis-Artificial Neural Network Algorithm." J. Basic. Appl. Sci. Res 3.6 (2013): 892-898.
12. Sekhon, Amrita, and Sukhwinder Singh Oberoi. "Predictive Analysis System for National Ranking and Accreditation of HEIs." Communications on Applied Nonlinear Analysis, vol. 32, no. 1s, 2024, pp. 539-51. https://doi.org/10.52783/cana.v32.2318
13. Bravo-Jaico, Jessie, et al. "Assessing digital transformation maturity in higher education institutions: a correlational analysis by actors and dimensions." Frontiers in Computer Science 7 (2025): 1549262.
14. Sibi, K J. (2020). The Role and Importance of ICT in Higher Education for Quality Assurance and NAAC Assessment and Accreditation.
15. Pavlova, Valeriia. "Methodological Approaches to the Implementation of Artificial Intelligence in the Educational Process of Higher Education Institutions." Problems of Education 1 (102) (2025): 71-86. https://doi.org/10.52256/2710-3986.1-102.2025.05
16. Kaur, Ushveen, and Sugandha Gupta. "Evaluating Quality-Measures to Improve NAAC Ranking for Higher Education Institutes." Feedback 20.20 (2021): 20. DOI:10.35940/ijrte.F5411.039621
17. Kamal Goel, Surinder Singh. “Perceptual Analysis of Students and Scholars towards Quality Education in Higher Educational Institutions”. European Economic Letters (EEL), vol. 13, no. 5, Nov. 2023, pp. 493-18, doi:10.52783/eel.v13i5.780. https://doi.org/10.52783/eel.v13i5.780
18. Ghulam, Mohammad Ghulam Ali. "Introducing and Evaluating Key Performance Indicators and Generating Awareness of Competence among Inter-Departments for Quality Performance in Large Academic and Research Institutions." International Journal of Educational Research Review 10.1: 20-28. https://doi.org/10.24331/ijere.1535062
19. Siddapuram, Arvind, et al. “Quality Practices and Accreditation in Higher Education Institutions: A Roadmap for Excellence in Engineering Education”. Journal of Engineering Education Transformations, vol. 37, June 2025, pp. 748-52, https://journaleet.in/index.php/jeet/article/view/2498.
20. Gautam, Dr Awadhesh Singh. "Impact of National Assessment and Accreditation Council (NAAC) on Higher Education Institutions (HEIs) in India." International Journal of All Research Education and Scientific Methods 12.8 (2024). http://dx.doi.org/10.2139/ssrn.4932646

21. Sibi, K. J. "Standard Operating Procedure to Ensure Transparency and Accountability in Accreditation Process in Higher Educational Institutions in India." International Multidisciplinary E-Research Journal (2019): 28-31.
22. Samant, T. (2025). File-Automator: A Framework for AI-Powered Document Processing. Authorea Preprints.
23. Asad, A. S., & Khan, F. M. (2025). Docu Mind AI–Intelligent Document Analysis System. Journal of Computer Science and Technology, 4(03), 06-11.
24. Acharekar, A., Bhosle, A., Dwivedi, M., & Deshpande, A. OCR for Multilanguage Text Extraction, Translation and Summarization.
25. Thalange, A. V., Shetgar, P. S., Patnaikuni, D., & Chamatagoudar, S. N. (2022, December). Online Document Identification and Verification Using Machine Learning Model. In International Conference on Data, Engineering and Applications (pp. 231-243). Singapore: Springer Nature Singapore.
26. de Almeida Baptista, J. P. M. (2022). Website Development for the VCMI Research Group (Master's thesis, Universidade do Porto (Portugal)).
27. Thalange, A. V., Shetgar, P. S., Patnaikuni, D., & Chamatagoudar, S. N. (2022, December). Online Document Identification and Verification Using Machine Learning Model. In International Conference on Data, Engineering and Applications (pp. 231-243). Singapore: Springer Nature Singapore.
28. Nafkha, A. (2021). Software Defined Radio & MIMO Signal Processing (Doctoral dissertation, Université de Rennes 1).
29. Vuksani, M. (2024). Adsorbent materials for biomethane applications to be used in a cogeneration plant to produce electricity and home heating (Doctoral dissertation, Politecnico di Torino).
30. Andreas, W. (2020). ANALYSIS OF MAJOR AND MINOR ELEMENTS IN COAL BY LASER-INDUCED BREAKDOWN SPECTROSCOPY (Doctoral dissertation, JOHANNES KEPLER UNIVERSITÄT LINZ).
31. Yakovenko, B. R., & Karlinska, K. A. (2020). HOW TO SECURE YOURSELF IN AN AIRPLANE IN THE MIDST OF A CORONAVIRUS. POLIT. CHALLENGES OF SCIENCE TODAY, 43.
32. Guitonni, A., & Boury-Brisset, A. (2002, September). Automatic documents analyzer and classifier. In 7th International Command and Control Research and Technology Symposium, Quebec.
33. Chimote, V. P., Chakrabarti, S. K., Pattanayak, D., & Naik, P. S. (2004). Semi-automated simple sequence repeat analysis reveals narrow genetic base in Indian potato cultivars. Biologia plantarum, 48(4), 517-522.
34. Ashkani, S., Rafii, M. Y., Shabanimofrad, M., Foroughi, M., Azizia, P., Akhtar, M. S., ... & Nasehi, A. (2015). Multiplex SSR–PCR approaches for semi-automated genotyping and characterization of loci linked to blast disease resistance genes in rice. Comptes Rendus. Biologies, 338(11), 709-722.
35. Abdullah, Yan, R., & Tian, X. (2025). CGAS (Chloroplast genome analysis Suite): an automated python pipeline for comprehensive comparative Chloroplast genomics. bioRxiv, 2025-12.
36. Carvalho, J., Yadav, S., Garrido-Maestu, A., Azinheiro, S., Trujillo, I., Barros-Velázquez, J., & Prado, M. (2021). Evaluation of simple sequence repeats (SSR) and single nucleotide polymorphism (SNP)-based methods in olive varieties from the Northwest of Spain and potential for miniaturization. Food Chemistry: Molecular Sciences, 3, 100038.
37. Kaya, H. B., Cetin, O., Kaya, H., Sahin, M., Sefer, F., Kahraman, A., & Tanyolac, B. (2013). SNP discovery by Illumina-based transcriptome sequencing of the olive and the genetic characterization of Turkish olive genotypes revealed by AFLP, SSR and SNP markers. PLoS One, 8(9), e73674.
38. https://assessmentonline.naac.gov.in/public/index.php/hei_dashboard
39. http://naac.gov.in/images/docs/Disclosure_of_Benchmarks/Affiliated_College_Benchmarks_5-12-2022.pdf