1. Introduction

In modern enterprise environments, digital platforms such as Enterprise Resource Planning (ERP) systems form the core of daily business operations, helping organizations run smoothly and make informed decisions. Despite such advanced systems, a great deal still relies on how accurately the data has been entered by the users. The greatest source of errors, however, lies in the process of human input, especially when unstructured fields require the users to enter their explanations for certain problems. Different individuals may have slightly differing meanings for what they are trying to state when they describe similar problems. Small discrepancies, a lack of clarity, or a certain degree of ambiguity may not be picked up immediately, but eventually, they will begin to creep into the risk control process.

The problem becomes more prominent in the automobile industry, where the processes involved within businesses are complex, and the rules of compliance are quite tough. The problem with ERP Risk Management modules stems from the fact that the risk, control, and potential impact have to be written in detail by the user, which, in turn, depends largely on individual experience, comprehension, and, in some cases, deadlines. Thus, it paves the way for different risks being written differently by different people, or, in some cases, written under incorrect headings. The issue with the quality of the data within these systems has been recognized by the current research, which leverages the use of Natural Language Processing (NLP) to validate the text inputs through the help of a deep learning model based on BERT.

2. Problem statement

ERP systems use data as their foundation for effective risk management. Free text entry as data collection has been a problem since it relies on human input, thus leading to inaccuracies and misclassification. This has an impact on data quality as well as the time taken by various business processes, eventually affecting the objectivity and reliability of reports by the ERP system. This problem becomes even tougher when considering the car industry since there are complexities in the processes as well as the need to maintain quality.

3. Objectives

- **Improve Data Quality and Consistency**: - Ensure risk – related data entered into the ERP system is accurate, clear, and uniform.
- **Implement BERT-based NLP Validation**: -Use a BERT model to automatically classify and validate text – based risk entries.
- **Enable Early Error Detection**: - Identify and validate the data entry mistakes at the time they are made.
- **Enhance Decision-Support Reliability**: - Provide management with trustworthy data for better risk analysis and decision making.

4. Datasets Used

This section outlines the data used to train and validate the proposed NLP model for risk classification.

➢ Data Source and Scope

The study utilizes a corpus of approximately 5,000 rows of financial and operational statements. This dataset is specifically chosen because it mirrors the type of unstructured text found in ERP Risk Management modules, containing professional terminology, financial metrics, and strategic corporate updates

➢ Data Structure

The dataset is structured into two primary columns to facilitate supervised learning:

- **Target Label (Sentiment / Risk Class):** - A categorical variable that classifies the text into one of three categories: Positive (Internal Control), Negative (Potential Risk), or Neutral (Control). In the context of this study, these labels serve as proxies for risk impact levels.
- Textual Input: - Unstructured sentences describing business events (e.g., “The company has no plans to move all production to Russia”).

➢ **Textual Characteristics**

The data presents several challenges that necessitate a deep learning approach:

- **Domain-Specific Vocabulary:** The text includes industry terms such as "operating profit margin," "net sales growth," and "sub-tier supplier insolvency."
- **Syntactic Complexity:** Sentences often contain multiple clauses and numerical data (e.g., "net sales increased by 5.2% to EUR 205.5 mn"), requiring the model to understand the relationship between entities rather than just identifying keywords.
- **Subjectivity:** The "Neutral" class often contains factual statements that must be distinguished from "Negative" risk indicators, a task where the BERT (Bidirectional Encoder Representations from Transformers) model excels due to its context-awareness.

➢ **Data Splitting**

	Dataset Split	Number of Samples	Percentage (%)
0	Training	3391	69.99
1	Validation	485	10.01
2	Testing	969	20.00
3	Total	4845	100.00

Fig 4.1: Data Splitting for training, validation, and testing

➢ **Dataset Overview**

positive	With the launch of new 3G handsets , Nokia aims to become the winner in China 's 3G market as it did in the 2G market .
neutral	With this subscription , Fortum 's ownership in TGK-10 has increased to slightly over 76 % of shares and voting rights .
neutral	VNH generates annual net sales of about 5 mln eur and employs 21 people .
positive	Earnings per share (EPS) for the first quarter 2007 amounted to EUR0 .07 , up from EUR0 .04 .
negative	Jan. 6 -- Ford is struggling in the face of slowing truck and SUV sales and a surfeit of up-to-date , gotta-have cars .
neutral	Rautakesko 's business operations in Norway and Russia , acquired in July 2005 , are included in the figures of the comparable period , impacting sales growth starting from August .
positive	Operating profit was EUR 11.07 mn , up from EUR 8.65 mn .
negative	Peer Peugeot fell 0.81 pct as its sales rose only 6.3 pct from the same period last year .
negative	Pharmaceuticals group Orion Corp reported a fall in its third-quarter earnings that were hit by larger expenditures on R&D and marketing .

Fig 4.1: A small section of dataset

5. Tools Used for Implementation

- Python – Used as the main programming language for model development and experimentation.
- Pandas – Used for loading, cleaning, and managing the dataset.
- NumPy – Used for numerical operations and data handling.
- Hugging Face Transformers – Used to implement the BERT tokenizer and BERT-based classification model.
- PyTorch – Used as the deep learning framework for training and fine-tuning the model.
- Hugging Face Datasets – Used for efficient dataset conversion and handling during training.
- Scikit-learn – Used for train–test splitting and evaluation metrics such as accuracy and confusion matrix.
- Matplotlib – Used for visualizing model performance and evaluation results.

6. Github Link of the code folder

https://github.com/VijayKumarBA/AI_AAT_assignment.git

7. Methodology

- Collected historical risk-related textual data from in CSV format.
- Cleaned and pre-processed the data to remove encoding issues and ensure consistent text formatting.
- Converted categorical risk labels into numerical form for supervised learning.
- Divided the dataset into training and testing sets using stratified sampling to preserve class balance.
- Used the BERT tokenizer to transform raw text into token IDs and attention masks while retaining contextual meaning.
- Standardized input sequence lengths through truncation and padding.
- Fine-tuned a pre-trained BERT model for multi-class classification of risk-related text entries.
- Trained the model with optimized hyperparameters to achieve stable convergence and minimize overfitting.
- Evaluated model performance using accuracy, precision, recall, F1-score, confusion matrices, ROC curves, and prediction confidence analysis.

8. Results Obtained

Classification Report:

	precision	recall	f1-score	support
Potential Risk	0.8385	0.9008	0.8685	121
Control	0.9131	0.8941	0.9035	576
Internal Control	0.8218	0.8309	0.8263	272
accuracy			0.8772	969
macro avg	0.8578	0.8753	0.8661	969
weighted avg	0.8782	0.8772	0.8775	969

Fig 8.1: Classification Report of BERT Method

Epoch	Training Loss	Validation Loss	Accuracy	Precision	Recall	F1
1	0.039600	0.988904	0.861713	0.865649	0.861713	0.857832
2	0.028500	0.801938	0.860681	0.861900	0.860681	0.861134
3	0.025000	0.985659	0.871001	0.874561	0.871001	0.872015
4	0.015900	0.988076	0.875129	0.875268	0.875129	0.872968
5	0.027500	0.887598	0.873065	0.872814	0.873065	0.872464
6	0.012500	0.963104	0.867905	0.868972	0.867905	0.868192
7	0.004200	1.037080	0.865841	0.869542	0.865841	0.866733
8	0.005200	0.962442	0.868937	0.869866	0.868937	0.869235
9	0.003200	0.992661	0.874097	0.876626	0.874097	0.874819
10	0.001300	0.982407	0.877193	0.878169	0.877193	0.877475

Fig 8.2: Training Loss, Validation loss, Accuracy, Precision, Recall, F1 after each Epoch

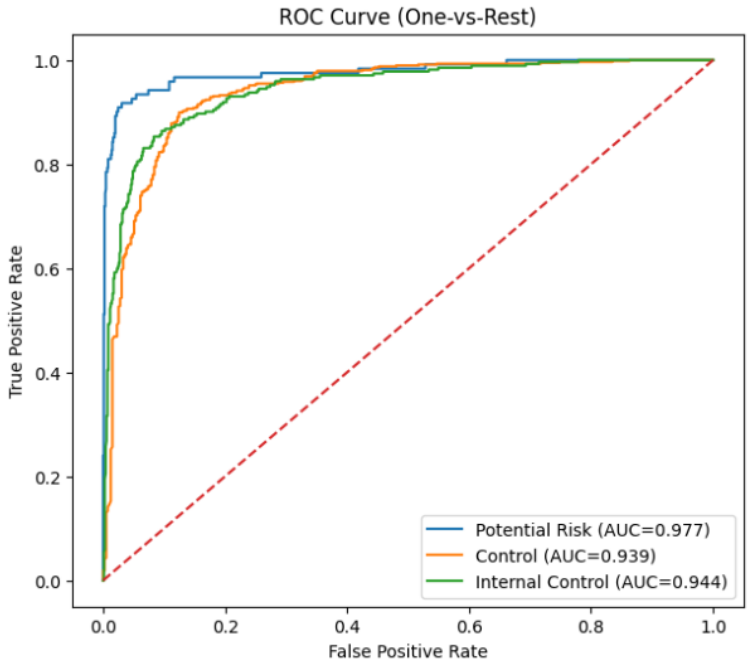


Fig 8.3: ROC of the BERT Model

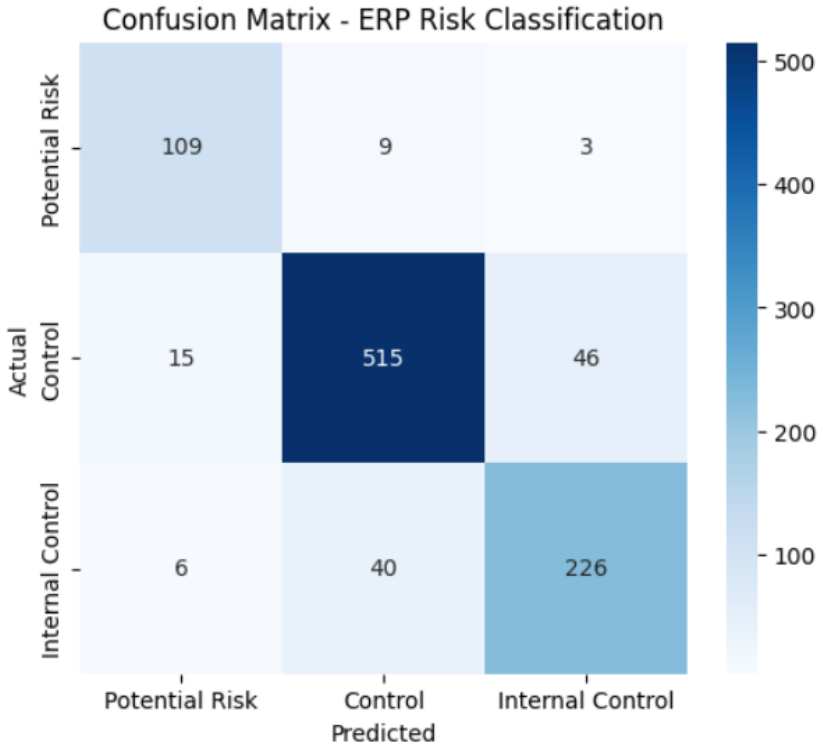


Fig 8.4: Confusion Matrix of BERT Method

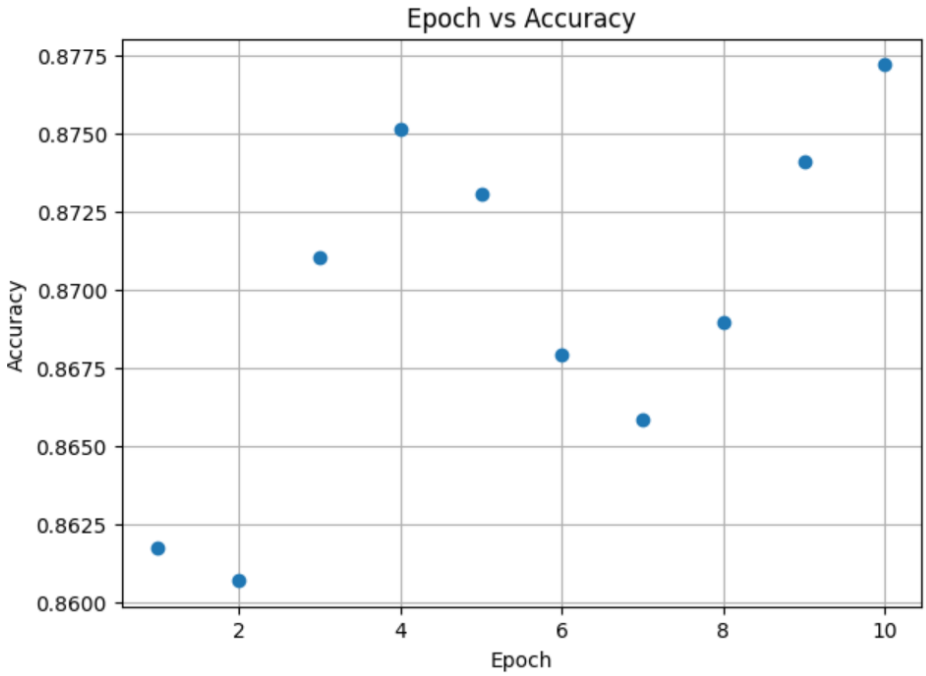


Fig 8.5: Epoch vs Accuracy graph

9. Comparative Study

- TF-IDF+SVM was employed as the traditional baseline to compare the performance of a powerful linear classifier with manual features in the context of risk-related text to establish a basis for comparison with deep learning.
- Embedding + LSTM was used to identify the dependencies in the risk descriptions. The Embedding + LSTM approach was expected to address the shortcomings of traditional models that relied on the TF-IDF technique for processing the data.
- Embedding + GRU was tried as an efficient computation strategy to replace LSTM, as it aims to preserve the context with less number of parameters.
- The same dataset, preprocessing steps, and train–test split were maintained across all models. Any variation in results therefore reflects model capability rather than experimental bias.
- This model evaluation was more than the accuracy. It involved confusion matrices, ROC curves, and the analysis of the confidence level of the predictions
- The best overall performance was given by the BERT-based model. The bidirectional nature of the contextual understanding allowed it to capture various representations of a single issue by a user quite effectively, when compared to traditional and recurrent models.

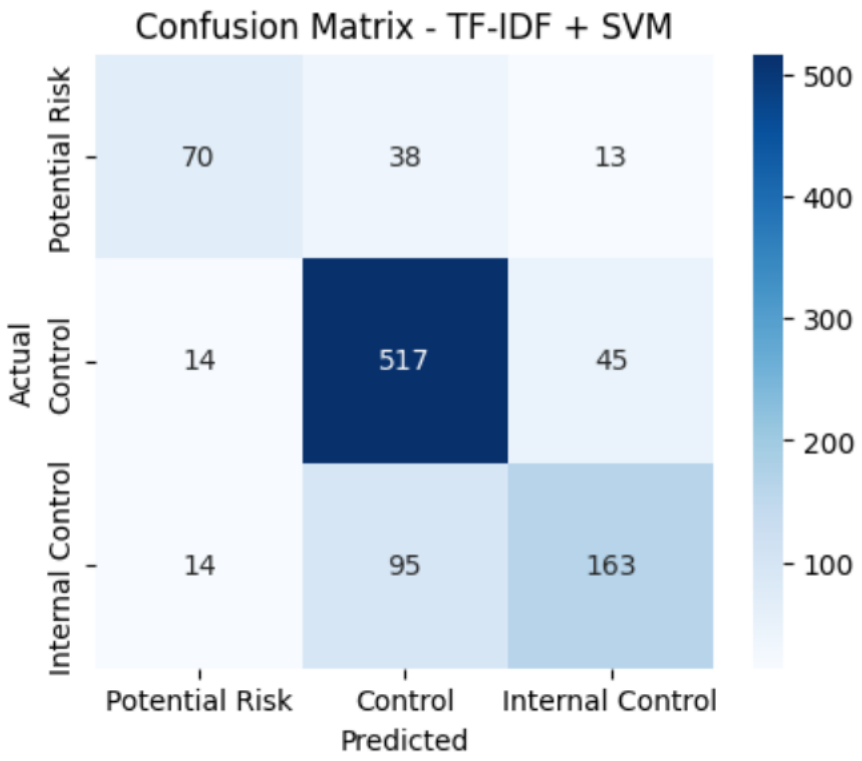


Fig 9.1: Confusion Matrix of TF-IDF + SVM

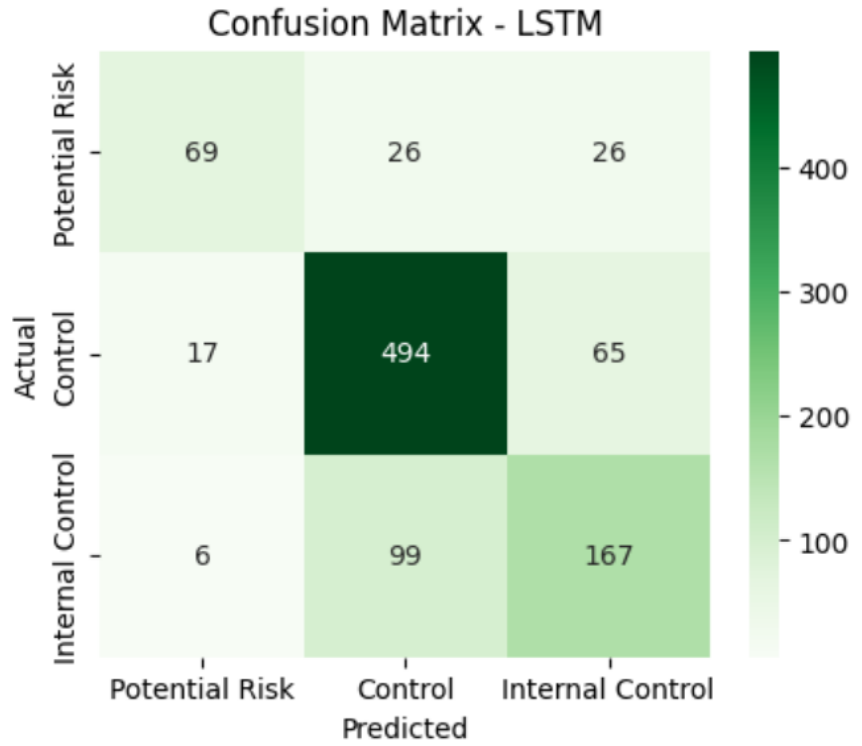


Fig 9.2: Confusion Matrix of Embedding LSTM

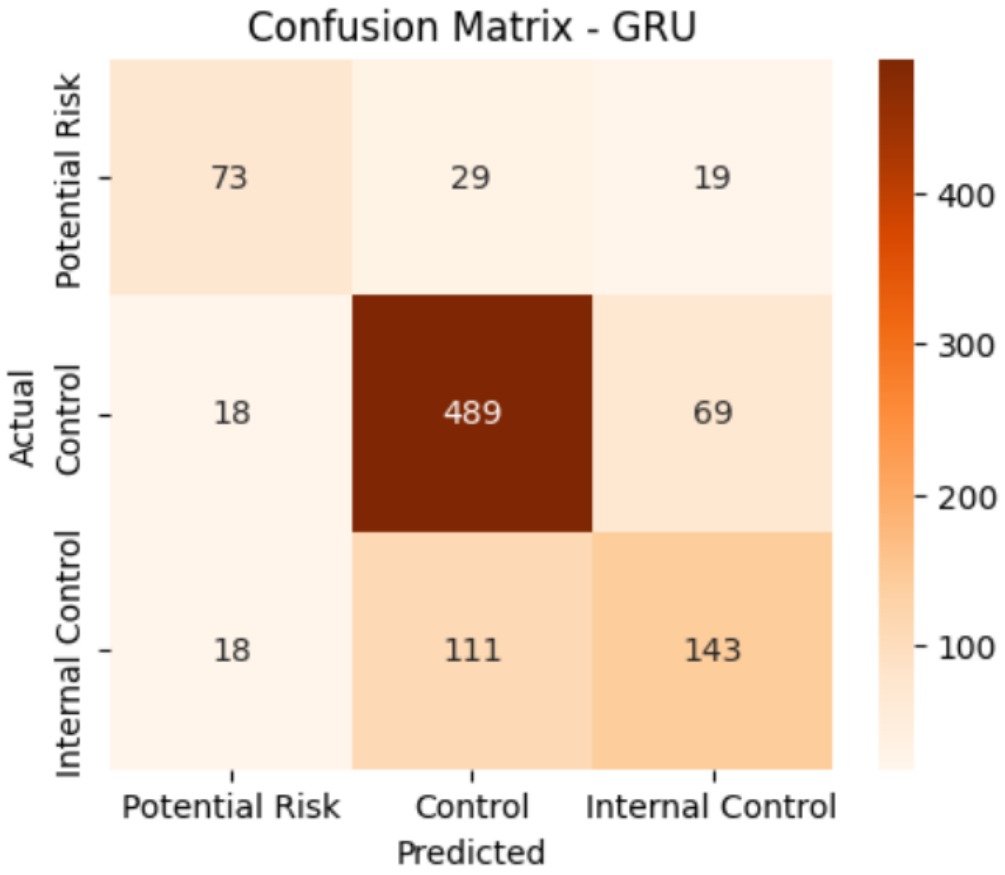


Fig 9.3: Confusion Matrix of Embedding GRU

10. Key Observations

- **TF-IDF + SVM:** It turns in very decent baseline performance but suffers from its sparse features representation as well as a lack of semantic understanding.
- **Embedding + LSTM:** Demonstrates high performance in capturing sequential context but generally tends to overfit or generalize poorly on unseen data.
- **Embedding + GRU:** Although GRU enjoys better computational efficiency than that of LSTM, it still failed to model complex semantic variations in text,
- **BERT:** It achieves superior performance because of its deep and bidirectional understanding of language and strong generalization across diverse textual patterns.

Comparison Outcome

Model	Accuracy	Precision	Recall	F1 - Score
BERT	0.8782	0.8786	0.8782	0.8783
TF – IDF + SVM	0.7739	0.7690	0.7740	0.7668
Embedding + LSTM	0.7554	0.7497	0.7534	0.7493
Embedding + GRU	0.7275	0.7195	0.7276	0.7213

Table 9.1: Comparision of BERT vs TF – IDF + SVM, Embedding + LSTM, Embedding + GRU

11. Learning Outcomes

- This work made it very clear how data quality problems start in ERP systems, especially when users freely type descriptions in text fields without any standard format.
- While preparing the dataset, it became obvious that cleaning and preprocessing real-world text takes far more effort than expected and cannot be skipped if meaningful results are needed.
- Working with the BERT model involved a learning curve, particularly during fine-tuning, but it helped in understanding how modern NLP models handle context much better than basic approaches.
- Comparing Naive Bayes, CNN, XGBoost, and BERT helped in clearly seeing where traditional models struggle and why transformer-based models perform better for subjective text.
- Model evaluation went beyond accuracy alone, and tools like confusion matrices and ROC curves were useful in understanding where the model was actually getting confused.
- Ultimately, the project demonstrated how NLP-based automation can reduce inconsistencies in ERP risk data and support more reliable decision-making in enterprise risk management.

12. References

[1] H. Canli, “Improving Data Entry Quality in Enterprise Applications With NLP Methods: A Model Proposal Based on BERT and Deep Learning,” IEEE Access, vol. 13, pp. 128592–128602, 2025, doi: 10.1109/ACCESS.2025.3590983.

[2] M.M. Dagli, Y. Ghenbot, H. S. Ahmad, D. Chauhan, R. Turlip, P. Wang, W. C. Welch, A. K. Ozturk, and J. W. Yoon, ‘‘Development and validation of a novel AI framework using NLP with LLM integration for relevant clinical data extraction through automated chart review,’’ Sci. Rep., vol. 14, no. 1, Nov. 2024, doi: 10.1038/s41598-024-77535-y.

[3] A. Rajbhoj, P. Nistala, A. Pathan, P. Kulkarni, and V. Kulkarni, ‘‘RClassify: Combining NLP and ML to classify rules from require ments specifications documents,’’ in Proc. IEEE 31st Int. Requirements Eng. Conf. (RE), Hannover, Germany, Sep. 2023, pp. 180–189, doi: 10.1109/re57278.2023.00026.

[4] S. Mohanty, A. Behera, S. Mishra, A. Alkhayyat, D. Gupta, and V. Sharma, ‘‘Resumate: A prototype to enhance recruitment process with NLP based resume parsing,’’ in Proc. 4th Int. Conf. Intell. Eng. Manage. (ICIEM), London, U.K., May 2023, pp. 1–6, doi: 10.1109/ICIEM59379.2023.10166169.

13. Screenshot of Similarity & AI Generated report

DrillBit

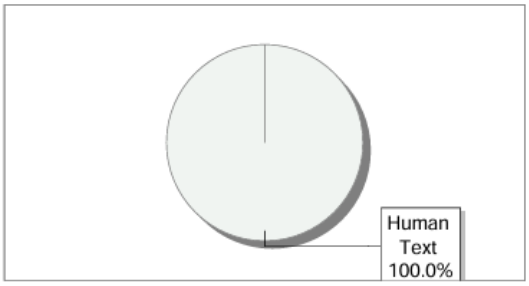
The Report is Generated by DrillBit AI Content Detection Software

Submission Information

Author Name	VijayKumar B A
Title	AI AAT
Paper/Submission ID	5150097
Submitted By	library@bmsce.ac.in
Submission Date	2026-01-14 14:53:31
Total Pages	14
Document type	Assignment

Result Information

AI Text: **0 %**



Disclaimer:

* The content detection system employed here is powered by artificial intelligence (AI) technology.
* Its not always accurate and only help to author identify text that might be prepared by a AI tool.
* It is designed to assist in identifying & moderating content that may violate community guidelines/legal regulations, it may not be perfect.

DrillBit

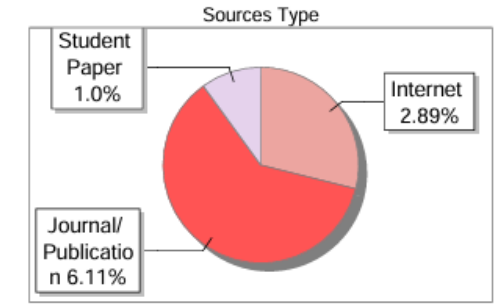
The Report is Generated by DrillBit Plagiarism Detection Software

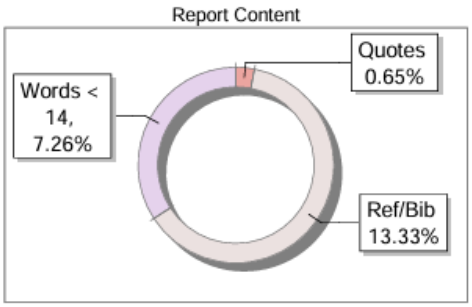
Submission Information

Author Name	VijayKumar B A
Title	AI AAT
Paper/Submission ID	5150097
Submitted by	library@bmsce.ac.in
Submission Date	2026-01-14 14:53:31
Total Pages, Total Words	14, 1695
Document type	Assignment

Result Information

Similarity **10 %**





Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File