2024 5th International Conference for Emerging Technology (INCET)
Karnataka, India. May 24-26, 2024

A Machine Learning Based Classification of Students' Algebraic Responses Using MathBERT Embeddings

Anees Sajid Injeti, G.Neha Rupsica, G. Praneetha Reddy, Roshni M Balakrishnan, Peeta Basa Pati*
Dept of Computer Science and Engineering
Amrita School of Computing,Bengaluru
Amrita Vishwa Vidhyapeetham, India
m_roshni@blr.amrita.edu
*0000-0003-2376-4591

2024 5th International Conference for Emerging Technology (INCET) | 979-8-3503-6115-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/INCET61516.2024.10593432

***Abstract*—Conduction of assessments and analysing the students' knowledge is the most important part of any teaching learning process. Traditional manual grading of written mathematical solutions can be time-consuming and laborious, especially for a larger group of students. It takes lot of time and effort to correct these answer scripts. In order to evaluate students' open responses for algebraic inquiries automatically, this project explores the use of MathBERT embeddings, an advanced natural language processing technique with different machine learning classifiers. The model tries to classify the student's responses to either correct or wrong. XGBoost Classifier on the MathBERT embeddings exhibited the best accuracy for this model with 75.3% mean accuracy with standard deviation of 0.016.**

***Keywords*—Quadratic Equations, Algebraic Solutions, Auto-grading, Mathematics, MathBERT, TextualEmbeddings, Automatic Evaluation.**

I. INTRODUCTION

One of the most important ways to measure academic development in the field of education is to examine students' comprehension and application of mathematical concepts, especially those related to algebraic equations. Traditional grading practices, particularly in mathematics, frequently entail teachers evaluating students manually over an extended period of time, which puts a heavy load on teaching resources. In order to tackle this problem, we develop a machine learning-based categorization method that makes use of MathBERT embeddings in order to accelerate the assessment of students' algebraic answers.A more efficient and automated method of assessing algebraic answers could have a good effect on the education sector, especially in today's world where there is a growing need for mathematical literacy. Teachers may spend more time improving their lesson plans, giving struggling students individualized attention, and creating a more interesting and customized learning environment by automating the evaluation process. The rapid determination of right or wrong answers via machine learning models not only helps teachers make the most use of their time, but it also advances the larger social objective of improving learning outcomes and encouraging mathematical competency.

ML models are perfect for automating review procedures since they can handle enormous amounts of data effectively. This is especially true when working with big datasets that would be difficult to evaluate manually.Consistent evaluations from ML models are possible since they are not impacted by subjective elements that might skew human judgment. This consistency aids in preserving the uniformity and impartiality of evaluations.When using ML models to automate evaluation, the time savings over manual assessment is substantial. Machine learning algorithms may do tasks that take human workers hours or days to complete in a fraction of the time.

This report is a major step toward long-term sustainable developments in evaluation processes in education and is aligned to sustainable development Goal, SDG-4. Through the integration of MathBERT embeddings' precision and machine learning's strength, we want to improve learner-centeredness, efficiency, and adaptability in the educational landscape while also streamlining the assessment of algebraic responses. With the help of this creative strategy, we see a time when teachers may focus on more important aspects of their work, and students get timely, personalized feedback that helps them on their unique learning paths.

The research on the creation of auto-grading or auto-evaluation systems especially suited for arithmetic questions seems to be noticeably lacking. Although a lot of work has been done in the field of auto-evaluation for different disciplines, there doesn't appear to be a thorough investigation or research article on the automated assessment of arithmetic questions. This points to a chance for educators and researchers to advance the subject by addressing this particular need and creating a framework for automatically evaluation mathematical problems. The rest of the paper contains the following sections Five sections comprise the remaining portion of the paper. The subsequent segment offers an all-encompassing analysis of the extant literature, centering on the present condition of auto-evaluation systems in diverse topics, with a specific emphasis on the disregard for mathematical queries. The research approach is described in the third section, which comes after the literature review. This section describes the methodology, resources, and methods used to create the framework for auto-evaluation of arithmetic questions. The fourth section then goes on to show the outcomes of applying the methodology. The fifth section, which tries to derive significant insights and conclusions from the data acquired during the study, then thoroughly analyzes the findings.

II. LITERATURE REVIEW

Wang et. al. [1] reports the face of changing trends in educational technology, automated grading systems that use machine learning and natural language processing for assessment in the classroom face scalability and ethical issues.

Bicer et. al. [2] reports the intelligent automation, the study suggests a JFLAP-based Automatic Automata Grading System to improve automata-related grading duties and contribute to conversations in automata theory and computer science education.

Muddaluru et. al. [3] reports the integration demonstrates a modern method of automating the grading of C programming assignments using Random Forest Regressor and CodeBERT, in line with the latest developments in the use of machine learning and natural language processing. This new set of methods for automated grading systems improves efficient assessment processes for programming assignments.

Roa et. al. [4] reports the application of autograding in systems courses is investigated in this literature review, which looks at previous studies to identify benefits and drawbacks. The study offers insights into the effects of auto-grading in systems education by exploring practical implications.

Li.X et. al. [5] reports the study covers sophisticated spoken English assessment systems, with a focus on multi-person interaction scenarios and the role that multi-feature fusion plays in improving automated assessment accuracy. Readers are recommended to read the entire work in order to gain deeper insights.

Li.S et. al. [6] reports a fresh method for automated program assessment of C language programs using fuzzy clustering algorithms. This will be helpful information for practitioners and researchers looking to improve program assessment methods. By addressing the difficulties associated with evaluating C programming, this methodology may advance automated assessment systems.

Prabhakar et. al. [11] reports the methods used on Indian court rulings, offering insights into the field's problems and advancements. This work might provide insightful methods for improving the legal context summary procedure.

Mounika et. al. [8] reports the encoder-decoder attention models and automatically corrects speech-recognized mathematical equations. It might examine developments, difficulties, and uses in this field, offering suggestions for enhancing mathematical equation recognition precision.

Narmada et. al. [9] reports the autograding approaches for programming skills is probably found in the literature, which may also examine developments and difficulties in automated evaluation methods. The results of this study could help improve the effectiveness and precision of the assessment of programming skills.

Lan et. al. [10] reports the problems and developments in the field of mathematical language processing approaches for automatic grading and feedback in open-response mathematical questions. This study could provide information on how to increase assessment efficiency and accuracy in math classes.

III. METHODOLOGY

In order to classify the students' open responses to algebraic questions, a series of process was included. Fig 1 explains the complete architecture of the model.

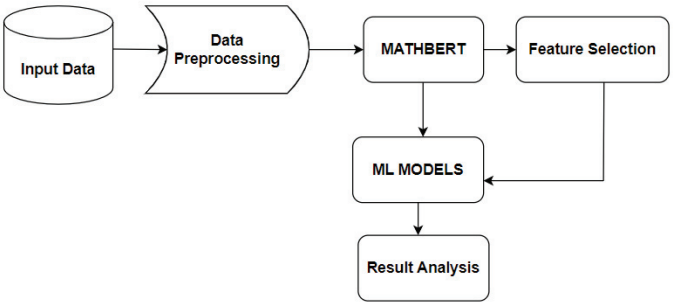

Fig. 1. Architecture diagram explaining the different steps involved in the automatic evaluation process

A. Data Exploration

For any learning based application its obvious that a huge amount of data is required for the training and testing purpose. for automatic evaluation of the mathematical answers, the data used includes answers written by students to 21 random quadratic equation questions. The solutions were collected from 50 students. The answers collected were manually graded by 4 subject experts on a scale of 5 and the average of these marks were calculated as the final mark. This was done because there was a variation in the marks awarded by all the four subject experts. Since the work focuses on classification of answers to either correct or wrong, a threshold was kept and all the marks above the threshold was considered as correct and others as wrong. The threshold for correct wrong separation was kept as 4. The work does not focus on a tri-class or multi-class classification, so all the answers which scored 4 and above on a scale of 5 was correct and others wrong. From the analysis of the marks given by the experts, it was found that, for the answers where the students missed to write certain steps, marks was reduced. That is the reason for choosing the solutions with 4 marks also as correct ones.

B. Data Preprocessing

To guarantee the quality and integrity of the dataset, a thorough data preprocessing was done. To streamline the dataset and get rid of any duplicate entries, the first step was to do duplicate removal. Then, in order to preserve the dataset's completeness, we thoroughly checked for NULL values and addressed any missing data points. These steps were taken in order to lay a solid foundation for the use of MathBERT embeddings in the future and the operation of various machine learning models. The number of correct and wrong answer instances in the data is shown in Fig. 2

C. Feature Extraction using MATHBERT

The complex mathematical semantics present in students' answers has to be extracted in order to train the model. Jia.T et. al. [7] reports "MathBERT" as an extended version of BERT that aims to improve the applications of natural language

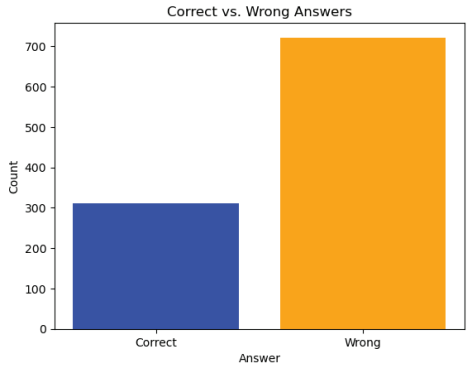

Fig. 2. Distribution of Correct and Wrong answers in the dataset

processing in mathematics education. The MathBERT was used to convert each occurrence in the dataset into a vector of size 384. This transformation into embeddings not only adds context-aware representations to the information but also serves as the foundation for applying machine learning models later on.

D. ML Classifiers

10 machine learning classifiers including traditional and ensembled models were utilized and their performance was analyzed. All the ML classifiers were subjected to hyper parameter tuning with the help of RandomizedSearchCV. Table I shows the different ML models along with their hyper parameters used for the study.

TABLE I. MODELS AND HYPERPARAMETERS

Model	**Hyperparameters**
SVM	{'C':1,'gamma':0.1,'kernel':'rbf'}
KNN	{'K=8 best'}
Decision Tree	{'criterion':'entropy', 'max_depth':'max_features':'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10'}
Naïve Bayes	{'var_smoothing': 1.0'}
AdaBoost	{'learning_rate': 0.1, 'n_estimators: 50'}
XGBoost	{'colsample_bytree': 0.9, 'learning_rate': 0.1,'max_depth': 7, 'n_estimators': 50,'subsample': 1.0'}
CatBoost	{'depth': 3,'iterations': 100, 'l2_leaf _reg': 5,'learning_rate': 0.1'}
MLP	{'activation':'relu','alpha': 0.001, 'hidden_layer_sizes':(100,),'learning_rate': 'adaptive', 'solver':'sgd'}
RF	{'bootstrap': False,'max_depth': 30, 'min_samples_leaf': 2,'min_samples_split': 10, 'n_estimators': 100'}
Logistic Regression	{'C': 1,'penalty': 'l2'}

It was well noted that there was a class imbalance in the data and because of that reason the model performance was effected. So for a proper analysis of which classifiers performed consistently throughout different combinations of train and test, K-Fold cross validation was applied. The number of folds k was fixed to 10.

Like any classification problem, the different metrics like accuracy, precision, recall and F Score was considered to analyse the performance of each ML classifier. For 10 fold cross validation, the accuracy obtained for all the 10 folds were obtained and the mean of that was taken in to consideration. The standard deviation of the 10 accuracy values obtained from the mean accuracy proves how consistent the performance of each model is on different combination of train and test i.e accross all the folds.

E. Feature Selection

The forward feature selection approach was employed to improve the effectiveness and performance of the machine learning models. This method helps to determine and keep only the most pertinent aspects in the dataset by evaluating the correlation between various features. It was possible to decrease dimensionality and enhance the interpretability speed, and generalization capacity of the models by eliminating superfluous or strongly correlated elements. The selected features where further subjected to training with all the ML classifiers used on raw MathBERT embeddings. The effects of this feature selection strategy on the overall effectiveness and interpretability of the different ML models were also analysed.

IV. RESULTS AND DISCUSSIONS

In order to analyse the performance of each model in classifying the MathBERT embeddings of the students' open responses to algebraic questions, mean accuracy along with its standard deviation was analysed.

A. Raw Data

From the results obtained it was understood that almost all the ensembled models performed in a similar way for the raw data. Table II shows the mean accuracy and the standard deviation of accuracy obtained in 10 folds with the mean accuracy and it also shows the mean F score and corresponding standard deviation. Similarly the mean precision and mean recall is shown in Table III.

TABLE II. MEAN ACCURACY AND MEAN F1 SCORES SHOWN BY DIFFERENT ML CLASSIFIERS IN 10 FOLDS.

Models	**Accuracy**		**F1-Score**	
	Mean	**STD**	**Mean**	**STD**
SVM	0.74	0.0100	0.69	0.0200
KNN	0.72	0.0200	0.70	0.0200
Decision Tree	0.66	0.0200	0.65	0.0200
Naïve Bayes	0.67	0.0200	0.67	0.0200
Adaboost	0.70	0.0200	0.61	0.0200
XGBoost	0.75	0.0160	0.74	0.0200
Catboost	0.75	0.0190	0.72	0.0200
MLP	0.73	0.0300	0.73	0.0300
RF	0.74	0.0100	0.71	0.0100
Logistic Regression	0.73	0.0200	0.71	0.0100

TABLE III. MEAN PRECISION AND MEAN RECALL SHOWN BY DIFFERENT ML CLASSIFIERS IN 10 FOLDS.

Models	Precision		Recall	
	Mean	STD	Mean	STD
SVM	0.74	0.0100	0.74	0.0300
KNN	0.72	0.0200	0.71	0.0200
Decision Tree	0.66	0.0200	0.65	0.0200
Naïve Bayes	0.67	0.0200	0.67	0.0300
AdaBoost	0.70	0.0200	0.72	0.0600
XGBoost	0.75	0.0100	0.74	0.0100
CatBoost	0.75	0.0100	0.73	0.0200
MLP	0.73	0.0300	0.73	0.0300
RF	0.74	0.0100	0.72	0.0100
Logistic Regression	0.73	0.0100	0.72	0.0300

The examination of the model performance provides important information about how well different machine learning models perform for the particular task, based on mean accuracy and standard deviation in a 10-fold cross-validation. Among all the models examined, XG Boost shows the highest mean accuracy, making it the best-performing model. This indicates that XG Boost has demonstrated its durability and reliability by continuously achieving greater predictive performance across various dataset subsets.

Random Forest, CatBoost, and XGBoost trail closely behind with excellent performance, indicating that they can handle the complexity of the data. Additionally demonstrating competitive performance, SVM and Logistic Regression maintain their ranks in the top tier of models. Conversely, the mean accuracy scores of MLP, KNN, Naïve Bayes, and Decision Tree models are comparatively lower. These models might still offer insightful information, but their performance might be inconsistent across various data divisions or they might have trouble identifying the underlying patterns in the dataset. Fig. 3 shows the graphical representation for a proper comparative analysis of the different models across the folds.

It is noteworthy that the model that behaves consistently should show a lover standard deviation in all the metrics. The in-depth examination carried out by cross-validation provides a thorough grasp of every model's behavior across all the folds.

Apart from ADBoost all the other Boosting algorithms performed in a similar pattern. When we consider only the Precision values, it can be noted that there is very huge variation across the folds which is depicted by the standard deviation in the graph. This is because the model is making a high number of false positive predictions relative to the total number of positive predictions it made. From the class distribution of the dataset used in this study, it is clear that there is a significant class imbalance. Here the number of correct answer instances is very less when compared to the wrong answer instances which leads to classifying many wrong answers as correct. It can be noted that application of SMOTE or other similar approaches to balance the class might help in reducing the misclassification.

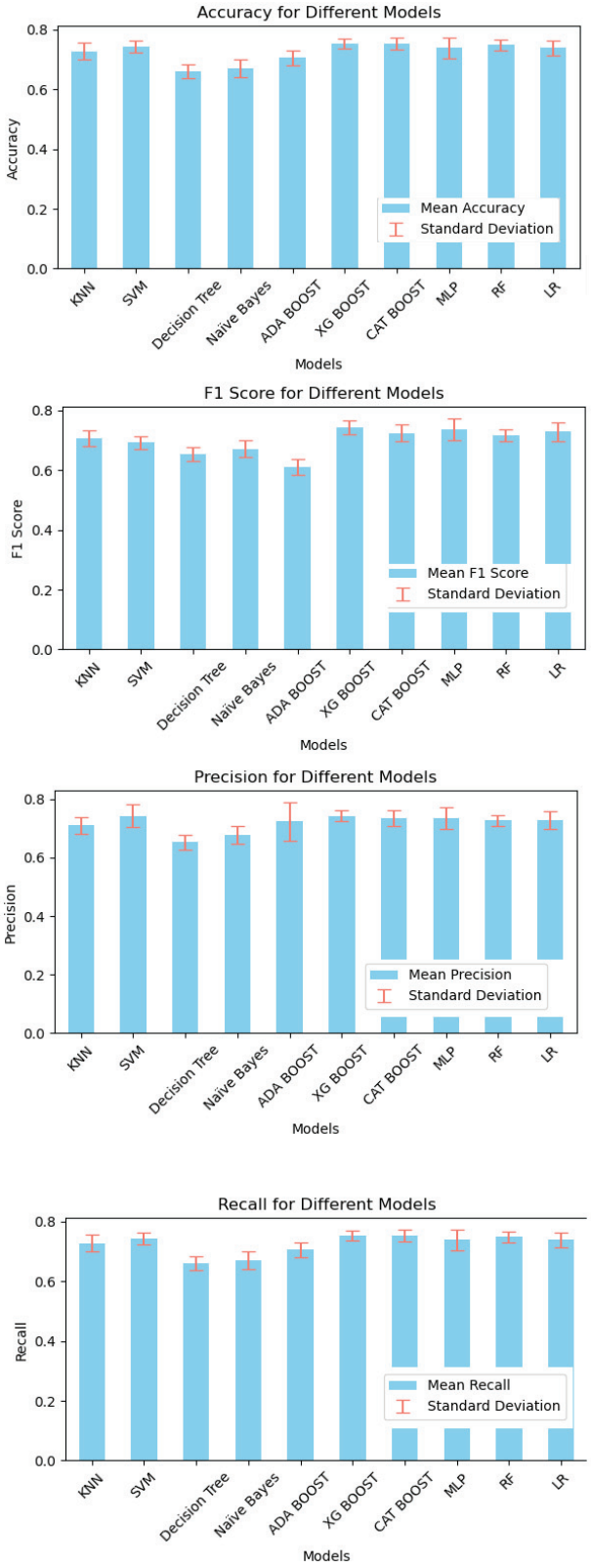

Fig. 3. Graphical Representations of mean accuracy, precision, recall and F1 scores with their corresponding standard deviation across the folds

B. Feature Selection

Feature selection has had a significant impact on the model's performance, especially improving the accuracy and

stability of certain models like SVM, Naive Bayes. The dimension of the raw data after feature extraction was 384 which was further reduced to 20. The top 20 features which contribute more to the result prediction was separately taken and in order to understand the model performance the same ML models with 10 fold cross validation was tried on these. as it was expected these top 20 features alone was also able to give the similar range of accuracy. the F1 scores remained almost the same throughout for all the models. The mean accuracy, Precision, Recall and F1 scores for all the models across the 10 folds are shown in the Table IV and Table V.

TABLE IV. MEAN ACCURACY AND MEAN F1 SCORES SHOWN BY DIFFERENT ML CLASSIFIERS IN 10 FOLDS AFTER FEATURE SELECTION.

Models	Accuracy		F1-Score	
	Mean	STD	Mean	STD
SVM	0.76	0.03	0.71	0.04
KNN	0.73	0.03	0.70	0.03
Decision Tree	0.68	0.02	0.67	0.02
Naïve Bayes	0.73	0.02	0.71	0.02
Adaboost	0.72	0.03	0.65	0.05
XGBoost	0.72	0.02	0.71	0.02
Catboost	0.75	0.01	0.73	0.02
MLP	0.72	0.02	0.72	0.02
RF	0.74	0.02	0.72	0.03
Logistic Regression	0.76	0.02	0.74	0.03

TABLE V. MEAN PRECISION AND MEAN RECALL SHOWN BY DIFFERENT ML CLASSIFIERS IN 10 FOLDS AFTER FEATURE SELECTION

Models	Precision		Recall	
	Mean	STD	Mean	STD
SVM	0.77	0.03	0.76	0.03
KNN	0.70	0.04	0.73	0.03
Decision Tree	0.67	0.02	0.68	0.02
Naïve Bayes	0.71	0.02	0.73	0.02
AdaBoost	0.72	0.04	0.72	0.03
XGBoost	0.69	0.02	0.71	0.02
CatBoost	0.72	0.03	0.74	0.03
MLP	0.72	0.02	0.72	0.02
RF	0.71	0.03	0.73	0.02
Logistic Regression	0.75	0.02	0.76	0.02

A comparison of the accuracies for the raw data and the data with only 20 features is shown in Fig 4

It was noted that the accuracy for all the ensembled models remains the same after feature selection. So it can be analysed that other features are not contributing to the performance of the model. In order to analyse and make sure about the best contributing features for different ML models SHAP analysis was also performed. SHAP (SHapley Additive exPlanations) analysis is a method used in machine learning for explaining the output of the particular model by telling the importance of input features to the model's predictions. From the results it was observed that ADABoost and CatBOOST performed similarly and a slight variation in the result was shown by XGBoost. So the SHAP analysis of XGBoost and CatBoost was performed.

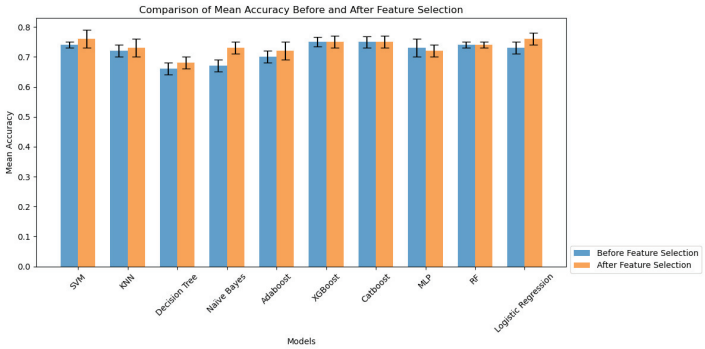

Fig. 4. Comparision of mean accuracy and standrad deviation before and after feature selection

The Fig 5 and Fig 6 shows the SHAP analysis plots of the MathBERT features for the models XGBoost and CatBOOST. It was noted that both the models showed almost same set of features as the high and low contributing. SO it was understood that the features chosen after forward feature selection and the SHAP analysis of different models, all represented almost the same features and the most contributing features.

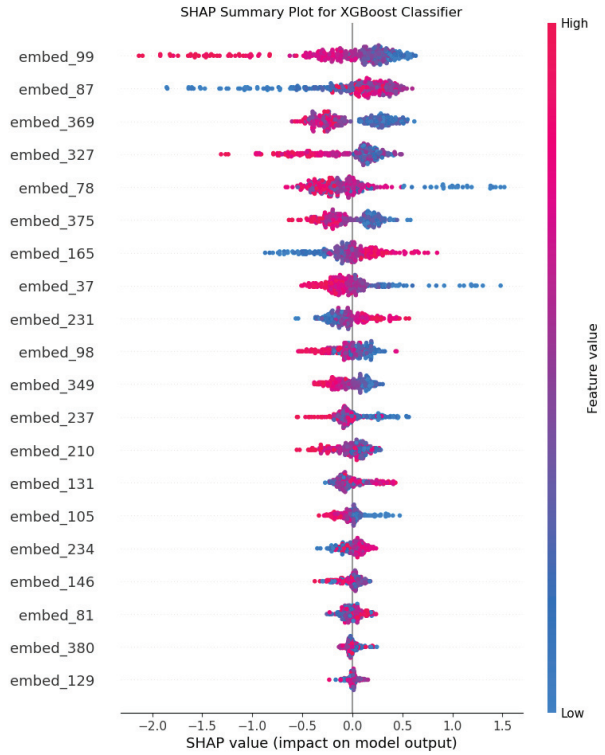

Fig. 5. SHAP analysis of the features on XGBoost

V. CONCLUSION

The primary goal of the study was to categorize students' algebraic equation solutions as either correct or incorrect. MathBERT embeddings was generated and this served as the features for the training and testing of different traditional and ensembled ML models. Using feature selection more especially, the forward feature selection technique was a key

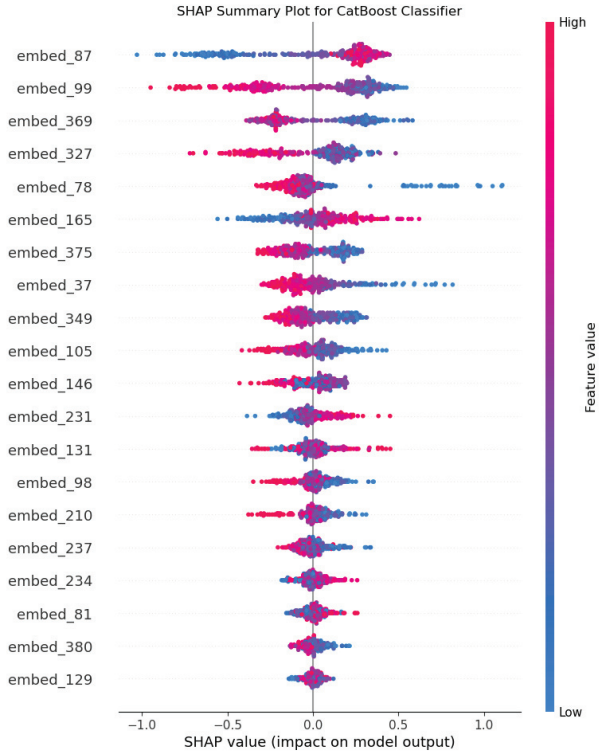

Fig. 6. SHAP analysis of the features on CatBOOST

component of the methodology. The model showed a variation in the recall mainly because of the class imbalance. The most contributing 20 features were selected and these features when used for training and testing also showed a similar behaviour for almost all the ensembled models. The work can be further extended by other pretrained models for generating the embeddings and fine tuning these models particularly for this dataset. Application of SMOTE to make balance the class can also be tried.

ACKNOWLEDGEMENT

Sincere gratitude is given to the mentors from Amrita School of Computing for their invaluable counsel and assistance, which were essential in finishing this work. Their expertise and support were very helpful in forming the research and honing the writing. QuillBot is also thanked for its invaluable assistance. This acknowledgement, which expresses gratitude for the cooperative efforts of both human mentors and cutting-edge AI technology, highlights the connection between traditional academic mentorship and the breakthroughs made possible by technologies like QuillBot and ChatGPT.

REFERENCES

[1] Z. Wang, J. Liu and R. Dong, "Intelligent Auto-grading System," 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), Nanjing, China, 2018, pp. 430-435, doi: 10.1109/CCIS.2018.8691244.

[2] M. Biçer, F. Albayrak and U. Orhan, "Automatic Automata Grading System Using JFLAP," 2023 Innovations in Intelligent Systems and Applications Conference (ASYU), Sivas, Turkiye, 2023, pp. 1-4, doi: 10.1109/ASYU58738.2023.10296744.

[3] R. V. Muddaluru, S. R. Thoguluva, S. Prabha, P. B. Pati and R. M. Balakrishnan, "Auto-grading C programming assignments with CodeBERT and Random Forest Regressor," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-6, doi: 10.1109/ICCCNT56998.2023.10308341.

[4] D. M. Rao, "Experiences With Auto-Grading in a Systems Course," 2019 IEEE Frontiers in Education Conference (FIE), Covington, KY, USA, 2019, pp. 1-8, doi: 10.1109/FIE43999.2019.9028450.

[5] X. Li, "Automatic Evaluation System of Spoken English for Multi Person Dialogue in English Teaching based on Multi Feature Fusion," 2021 International Conference on Education, Information Management and Service Science (EIMSS), Xi'an, China, 2021, pp. 269-272, doi: 10.1109/EIMSS53851.2021.00065.

[6] S. Li, "Design of automatic evaluation system of C language program based on fuzzy clustering algorithm," 2023 IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE), Jinzhou, China, 2023, pp. 1519-1524, doi: 10.1109/ICSECE58870.2023.10263375.

[7] Jia.T ,MathBERT: "A Pre-trained Language Model for GeneralNLP Tasks in Mathematics Education," 2023.

[8] Mounika, Y., Tarakaram, Y., Lakshmi Prasanna, Y., Gupta, D., Basa Pati, P., 2022. Automatic correction of speech recognized mathematical equations using encoder-decoder attention model, in: 2022 IEEE 19th India Council International Conference (INDICON).

[9] Narmada, N., Pati, P.B., 2023. Autograding of programming skills, in: 2023 IEEE 8th International Conference for Convergence in Technology (I2CT).

[10] Lan, A.S., Vats, D., Waters, A.E., Baraniuk, R.G., 2015. Mathematical language processing: Automatic grading and feedback for open response mathematical questions.

[11] Prabhakar, P., Gupta, D. and Pati, P.B., 2022, December. Abstractive Summarization of Indian Legal Judgments. In 2022 OITS International Conference on Information Technology (OCIT) (pp. 256-261). IEEE.

[12] Kushman, N., Artzi, Y., Zettlemoyer, L., Barzilay, R., 2014. Learning to automatically solve algebra word problems, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics,Baltimore, Maryland.

[13] R. Gao, N. Thomas and A. Srinivasa, "Work in Progress: Large Language Model Based Automatic Grading Study," 2023 IEEE Frontiers in Education Conference (FIE), College Station, TX, USA, 2023, pp. 1-4, doi: 10.1109/FIE58773.2023.10343006.

[14] M. Biçer, F. Albayrak and U. Orhan, "Automatic Automata Grading System Using JFLAP," 2023 Innovations in Intelligent Systems and Applications Conference (ASYU), Sivas, Turkiye, 2023, pp. 1-4, doi: 10.1109/ASYU58738.2023.10296744.

[15] K. R. Arvind, P. B. Pati and A. G. Ramakrishnan, "Automatic text block separation in document images," 2006 Fourth International Conference on Intelligent Sensing and Information Processing, Bangalore, India, 2006, pp. 53-58, doi: 10.1109/ICISIP.2006.4286061.