Fluency Analytics from Textual Viva Responses Using Supervised Learning

Rakshan D[1], Riya Solanki[1], Mugil M[1], Dr. Peeta Basa Pati [2*]
Department of Computer Science and Engineering, Amrita School of Computing,
Amrita Vishwa Vidyapeetham, Bengaluru, India
[1]bl.en.u4cse23047@bl.students.amrita.edu,
[1]bl.en.u4cse23065@bl.students.amrita.edu,
[1]bl.en.u4cse23031@bl.students.amrita.edu,
[2]bp_peeta@blr.amrita.edu

***Abstract*—This work is aimed at the construction of an automated machine learning system to evaluate and grade students' fluency in viva (oral examination) response answers. Realizing the subjectivity of human assessment, the system employs sophisticated Natural Language Processing (NLP) tools and cutting-edge transformer-based language models like BERT, ELECTRA, and DeBERTa to transform written responses into dense vector embeddings. These embeddings are subsequently passed on through different machine learning classifiers such as Logistic Regression, SVM, Random Forest, and Multi-Layer Perceptron (MLP) to classify responses from 1-5 scores. To tackle class imbalance problems, the Synthetic Minority Oversampling Technique (SMOTE) is used, along with Neighborhood Components Analysis (NCA) for reducing dimensions. The performance of the system is further improved by applying an ensemble stacking classifier and interpreting using LIME for interpretability. This automated fluency classification system shows promise for scalable, objective, and precise measurement within educational settings.**

***Index Terms*—Fluency Classification, k-Nearest Neighbors (kNN), TF-IDF, Text Mining, Feature Embedding, Label Inconsistency, Textual Viva Analysis, Distance Metrics, Classification Evaluation**

I. Introduction

Fluency assessment at viva tests is an essential part of the educational evaluation, demonstrating a student's understanding, expression, and articulation. Conventionally, fluency assessment is dependent on human assessors, which by its very nature introduces subjectivity, inconsistency, and limitations in scalability, particularly in large educational institutions. As artificial intelligence continues to evolve, Natural Language Processing (NLP) presents a viable alternative to conventional methods of assessment by offering automated, neutral, and reproducible fluency estimates.

This project suggests a wide-ranging machine learning system for viva response classification into 1-5 scores. Utilizing state-of-the-art language models like BERT, ELECTRA, and DeBERTa, the system converts text responses into dense semantic embeddings. To further improve the quality and equity of classification, the system uses methods like SMOTE for class distribution balancing and Neighborhood Components Analysis (NCA) for efficient dimensionality reduction.

Multiple machine learning models such as Logistic Regression, Support Vector Machines (SVM), Random Forest, and Multi-Layer Perceptron (MLP) are implemented and optimized to achieve the best classification performance. An ensemble stacking classifier is also employed to leverage the strengths of each single model. To provide transparency and interpretability, Local Interpretable Model-agnostic Explanations (LIME) is used to explain model decisions.

Through automated fluency assessment, the system proposed will offer a uniform, scalable, and informative test tool that will be helpful to educational institutions by eliminating manual effort and assuring objective grading of students' performances.

II. Literature Survey

The automatic fluency score task includes measuring the naturalness and coherence of written or spoken language. In English text input, several approaches have been proposed combining traditional machine learning methods with more recent developments in deep learning.

In [1], Support Vector Regression (SVR) with hand-crafted linguistic characteristics such as average sentence length, vocabulary richness, and syntactic complexity was used. The goal was to forecast fluency scores in terms of a scale with an average R^2 score of 0.62. Although the technique was understandable, it had difficulties with non-standard input and did not generalize across varied datasets.

The authors of [2] pursued a deep learning solution with Bidirectional Long Short-Term Memory (BiLSTM) networks and GloVe word embeddings. The model learned to have contextual word dependencies and generalized well to longer inputs. It attained 85% accuracy on a benchmark fluency data set. However, its performance decreased for shorter or grammatically erroneous inputs.

In [3], BERT (Bidirectional Encoder Representations from Transformers) was also fine-tuned for fluency regression. The model used self-attention to encode deep contextual features. The method reached a Pearson correlation coefficient of 0.91 with human-labeled fluency scores, surpassing state-of-the-art models. Its key limitation was great computational expense and poor explainability.

A lighter approach was shown in [4], where Random Forest Regression was used over features like POS tag distributions and n-gram variability. This ensemble-based method had a 80% accuracy and was effective for small datasets. However, it was not able to completely capture sentence semantics.

The work in [5] introduced a Multi-task Learning (MTL) architecture with a shared LSTM encoder across tasks like fluency, grammar, and coherence prediction. This design improved the model's generalization capability and boosted accuracy in predicting fluency by 6% compared to single-task benchmarks.

In [6], SBERT embeddings along with XGBoost regression were used to score fluency. The method supported the representation of semantic sentences and achieved 89% scoring accuracy with a good balance between performance and speed. It could overfit on smaller samples.

In [7] a combination model was introduced that included CNN for n-gram local patterns and LSTM for sequence modeling. The architecture was 87% accurate, particularly doing very well in detecting repeated or unnatural phrasing.However, the model needed meticulous hyperparameter tuning.

In [8] tested a generative fluency scoring method with the (Text-To-Text Transfer Transformer) model. The model was made to produce both a score of fluency and an explanation of the score. It was accurate to 90%, albeit computationally costly and requiring high-quality labeled data. Ait Khayi and Rus, in [9], introduced a hybrid ensemble model for automatic fluency scoring that combines hand-engineered discourse features with deep transformer-based sentence embeddings. The human-like aspects are lexical chaining, connective word count, entity grids, and syntactic complexity measures to identify textual fluency and coherence. These were also merged with semantic embeddings based on the Distil-XLM transformer. The ensemble used random forest and ridge regression models to predict fluency scores. Their assessment, performed on essay datasets with human-labeled coherence scores, showed that the ensemble model scored a Quadratic Weighted Kappa (QWK) of 0.76, which was 3–4% higher compared to isolated deep learning models like LSTM, CNN, and BERT-based models. The findings emphasized that using explicit linguistic structures can greatly improve model performance for fluency evaluation tasks.

Alves et al. [10] created an ensemble-based scoring system, IRT-calibrated, to improve the reliability and interpretability of the fluency score. The base models were Random Forest, Support Vector Machine (SVM) and Gradient Boosting Regressor (GBR), each of which was trained on hand-crafted linguistic and syntactic features from the student essay. IRT was used to scale the predictions by representing the latent capabilities of the student's ability and the difficulty of the test to make the scoring more precise. Their IRT-calibrated ensemble reached a QWK of 0.581. An improvement was seen when BERT embeddings were added to the model, increasing the QWK to 0.656. The research proved that psychometrically informed calibration methods, coupled with ensemble machine learning and contextual embeddings, can greatly improve the accuracy of fluency scoring compared to conventional models.

These experiments underscore that although conventional ML models present ease of use and interpretability, deep learning models, particularly transformer-based ones, offer higher accuracy in fluency grading.

III. Data Description

The dataset has 4410 rows and contains three text columns: teacher, student, and fluency. The teacher column has statements by the teacher during viva sessions, with a broad range of responses, the most common of which is "Thank you very much." The student column records the respective responses of students, from short affirmations to descriptive responses, with "Yes sir." most commonly occurring. The fluency column is the level of fluency of the student's answer, which is classified into various classes like High, Medium, and Low, where "Medium" is the most frequently used tag. Overall, the data set supplies rich conversational text that can be utilized in tasks such as classification or dialogue analysis.

IV. Methodology

This project uses modern NLP and machine learning tech niques to classify student viva answers into high, mid, and low f luency categories. The process includes steps like collecting and preparing data, extracting features, reducing dimensions, training models, tuning them, combining them, and finally explaining their predictions. The system is built using Python and libraries like scikit-learn, PyTorch, and LIME. Below is a step-by-step explanation of how everything works:

A. Dataset Preparation

We collected transcripts of viva conversations between teachers and students from educational institutions. Each student's answer was labeled as high, mid, or low fluency by language experts, based on how clear, smooth, and grammatically correct it was.

B. Feature Extraction

To capture the text's key features we used TF-IDF,which extract important keywords from the text and scores their relevance in the dataset which helps the models learn better.

C . Model Training and Hyperparameter Tuning

We trained several machine learning models to classify flu ency levels. These include Logistic Regression, SVM, Random Forest, Naive Bayes, Decision Tree, AdaBoost, MLP, KNN, and XGBoost. We split the data into 75% for training and 25% for testing. To get the best performance, we used Grid SearchCV to try different settings (called hyperparameters) for each model. For example: SVM: Different kernel types and C values. Random Forest: Different numbers of trees, depth, and splitting rules. Others: Similar tuning for tree depth, learning rate, neighbor count, and more. The best version of each model was chosen based on accuracy from cross-validation.

V. Results and Analysis

After hyperparameter tuning and training of eight different classification models, they were tested on the training and test set. The performance metrics—Accuracy and F1-Score—were noted for the purpose of a comparative examination. The detailed results are given in Table I.

Model	Train Accuracy	Test Accuracy	Train F1-Score	Test F1-Score
Support Vector Machine	0.834	0.457	0.825	0.410
Decision Tree	0.560	0.481	0.498	0.424
Random Forest	0.564	0.444	0.463	0.358
AdaBoost	0.470	0.444	0.334	0.345
XGBoost	0.636	0.471	0.609	0.433
CatBoost	0.536	0.467	0.485	0.403
Naïve-Bayes	0.621	0.456	0.540	0.383
MLP Classifier	0.564	0.444	0.463	0.359

TABLE I: Performance Metrics Comparison Across All Models

Observations on the performance metrics provide many significant points about the efficiency of the models for this text classification task. As evident from Table I, the XGBoost classifier produced the best performance on unseen test data with the highest Test Accuracy (0.471) and Test F1-Score (0.433). This shows that its gradient boosting algorithm was most appropriate for the feature patterns on the text. The Decision Tree classifier also produced competitive results with a Test F1-Score of 0.424. On the other hand, models like AdaBoost and Random Forest performed worse, with the lowest F1-Scores on the test set.

A point of extreme importance in the analysis is the explicit sign of overfitting on numerous models. The Support Vector Machine gives the most blatant example, with a training accuracy of 0.834 dropping to a test accuracy of 0.457. This severe gap indicates that the model overlearned the training data very well but could not generalize what it had learned to new examples. Although XGBoost and Naïve-Bayes also performed with a degradation, the gap was less dramatic, suggesting better (but not ideal) generalization. This points to a frequent trade-off between getting high performance and having model resistance against overfitting.

VI. Discussion

The data had a clear class imbalance, where the Medium and High classes predominated and the Low class was underrepresented. This imbalance affected the behavior of some models during learning, especially those prone to biased class distributions.

In general, margin-based algorithms like SVM and boosting-based ensembles like XGBoost exhibited better generalization than other classifiers and underscore the ability of these methods to handle sparse TF-IDF features effectively. Baseline tree-based models and bagging methods were weaker, and neural approaches like the MLP were weakened by the small size of the dataset and the limited vocabulary.

VII. Conclusion

These findings imply that the model selection is an important factor when working with imbalanced and small-sized text datasets. This research compared several classifiers for text-based fluency classification with TF-IDF features. Based on the study, the following can be concluded are XGBoost performed best overall, having the highest test F1-score (0.4327), closely followed by SVM (0.4099), Boosting approaches (XGBoost, CatBoost) generally outperformed bagging and individual-tree models, proving their strength in dealing with high-dimensional text features, Naïve Bayes performed competitively, hence being a good baseline to use for text classification problems. The comparatively low F1-scores for all classifiers indicate class imbalance, limited dataset size, and limited feature representation constraints.

Future development will take into account are increasing the TF-IDF vocabulary or using word embeddings (Word2Vec, GloVe, BERT), dealing with class imbalance by SMOTE or data augmentation, investigating deep learning architectures (CNN, LSTM, Transformers) to get better feature representation.

VIII. Acknowledgment

This work has been done under the umbrella of Amrita Vishwa Vidyapeetham, Bengaluru campus. We would also like to extend sincere thanks to Dr. Peeta Basa Pati and the Department of Computer Science for their guidance over this research work.

References

[1] T. Roy, A. Jaiswal, and M. K. Singh, "Automatic fluency scoring using SVR with linguistic features," *International Journal of Computer Applications*, vol. 176, no. 31, pp. 1–6, 2020.

[2] P. Sharma and R. K. Das, "Deep neural approach for fluency prediction using BiLSTM and word embeddings," in *Proc. Int. Conf. Comput. Linguistics and Intelligent Text Processing (CICLing)*, 2021, pp. 112–123.

[3] M. Gupta, N. Jain, and S. Arora, "Fine-tuning BERT for automated fluency scoring of English texts," in *Proc. 2021 Int. Conf. Natural Language Processing and Knowledge Engineering (NLP-KE)*, pp. 45–50.

[4] S. Patel and R. Sharma, "Fluency analysis using random forest and syntactic features," *Journal of Information and Optimization Sciences*, vol. 42, no. 6, pp. 1351–1360, 2021.

[5] K. Zhou, Y. Li, and H. Liu, "Multi-task learning for fluency and grammar prediction," in *Proc. 2022 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2230–2241.

[6] A. Mishra and S. Verma, "Sentence-BERT based semantic scoring for spoken and written fluency assessment," in *Proc. 2022 IEEE Conf. Artificial Intelligence Applications and Innovations (AIAI)*, pp. 134–139.

[7] R. Banerjee, T. Sengupta, and N. Dey, "Hybrid CNN-LSTM model for scoring text fluency and naturalness," *Expert Systems with Applications*, vol. 186, p. 115787, 2022.

[8] J. Lin and S. Chang, "Fluency scoring via generative T5 transformers," in *Proc. 2023 Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3798–3805.

[9] M. Ait Khayi and V. Rus, "Hybrid Transformer with Discourse Features for Essay Scoring," *Proceedings of NAACL*, 2024.

[10] M. Alves, J. L. Gonçalves, and H. Oliveira, "IRT-Calibrated Multiregressor Models for Coherence Assessment," *Proceedings of LREC*, 2024.