

A comparison of various machine learning algorithms and execution of flask deployment on essay grading

Udhika Meghana Kotha, Haveela Gaddam, Deepthi Reddy Siddenki, Sumalatha Saleti

Department of Computer Science and Engineering, SRM University AP, Andhra Pradesh, India

Article Info

Article history:

Received Jul 11, 2022

Revised Jul 25, 2022

Accepted Aug 18, 2022

Keywords:

Automated essay scoring

Count vectorizer

Essay grading

Flask deployment

Machine learning

ABSTRACT

Students' performance can be assessed based on grading the answers written by the students during their examination. Currently, students are assessed manually by the teachers. This is a cumbersome task due to an increase in the student-teacher ratio. Moreover, due to coronavirus disease (COVID-19) pandemic, most of the educational institutions have adopted online teaching and assessment. To measure the learning ability of a student, we need to assess them. The current grading system works well for multiple choice questions, but there is no grading system for evaluating the essays. In this paper, we studied different machine learning and natural language processing techniques for automated essay scoring/grading (AES/G). Data imbalance is an issue which creates the problem in predicting the essay score due to uneven distribution of essay scores in the training data. We handled this issue using random over sampling technique which generates even distribution of essay scores. Also, we built a web application using flask and deployed the machine learning models. Subsequently, all the models have been evaluated using accuracy, precision, recall, and F1-score. It is found that random forest algorithm outperformed the other algorithms with an accuracy of 97.67%, precision of 97.62%, recall of 97.67%, and F1-score of 97.58%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Sumalatha Saleti

Department of Computer Science and Engineering, SRM University AP

Neerukonda, Guntur District, Andhra Pradesh, India

Email: sumalatha.s@srmmap.edu.in

1. INTRODUCTION

Automated essay scoring (AES) [1] is a specialized system used to assign scores to the essays written in the form of various computer programs. It works by extracting various special features such as word count, vocabulary, the density of error, paragraph structure, and length of sentences. It is mainly used for its validity and reliability which has now become a great attraction from schools, colleges, companies, and researchers. Evaluation and grading are considered to play crucial importance in English literature. Where AES was once considered a myth/impossible task/unattainable job is now being developed after many trials and errors and after many upgrades from previous versions are now induced into education systems. As of today, essays written by students from all over the place are graded by not only teachers or examiners but also machines. Currently, most of the AES systems are used for grading English essays in most of the European countries [2].

Many intellectuals argue that the nature of the essay grading leads to the mismatches in the scores given by human raters, which is a major injustice for many of the students. This may be eradicated if the proposed technique can be used to evaluate essays with a scope generated by the human rater. Such artificial intelligent systems attain relative productivity to AES by replicating the human intelligence and the behavior

of electronic systems to soften the burden of educators. The literature field exams which test the student's capability and knowledge in the subject like test of English as a foreign language (TOEFL), international English language testing system (IELTS), and graduate record exam (GRE) are graded by both humans and machine learning based automated essay grading systems where both these grades are taken into consideration and the final score is the average for both. One of the main advantages of this machine is that it provides results with accuracy and precious feedback, using this system has shown a drastic change in the education institutes and also student's careers. Very deliberate research of the previous models had helped us to get the idea of the already existing works which were based on artificial intelligence (AI), machine learning, and natural language processing (NLP). In this process, we also found some loopholes in the existing works which was the reason for the failure of the proposed systems.

We generally look into organizations like schools, colleges in which English speaking, listening, and writing play a crucial role. In higher schooling, we also have to attempt English-based exams such as GRE, TOEFL, IELTS, and scholastic aptitude test (SAT), where literature and essays are evaluated with higher preference. The mandatory scope of testing all the skills is done by evaluating the essays they write which include various aspects like sentence formation, word usage, length of the essay, and it shows the student's capability of English literature. This written work depends on the criteria of several students writing. A single teacher cannot correct all the essays of students written in an English test. If the number of students is more, then the correction time increases. In general, people get tired after doing the same work for a long time. In a few cases, the tiredness may cause loss of interest in correcting and may lead to a difference in marks obtained to actual marks which they would get. So, we can use the automated essay grading system which will evaluate the essays of any number of students. These systems are being adopted by different organizations to reduce the hectic workload from a teacher's point of view. This will not only save the time for evaluation but also give accurate results. The output of the system will be quick such that it could evaluate many papers of essays and get trained. This system benefits both the student and the teacher as well.

Grading an essay is always an issue. It requires more time and effort from teachers, professors, or graders. Grading can be especially distressing for them. Evaluating different student essays is reading different student's minds. The main problem in grading essays is because teachers vary so much in the procedures or materials, they use for determining students' essay grades. The teachers may grade students according to their perspective which may vary from one grade given by a teacher to others. The teachers benefit from a lot of time-consuming tasks such as correcting many scripts. So, this model helps in reducing their hectic work burden and also saves their work burden. Virtual learning is a concept to enlarge educational experiences and it is a productive method of learning technologies based on the Internet [3]. It also helps the students to study independently which means free from educational techniques and becomes an outstanding technology. Irrespective of the physical classroom appearance, students in virtual learning can interactively access more resources, so many of the educators prefer virtual learning which helps in enhancing skills and knowledge. In the English-speaking world, virtual learning environments (VLEs) have been adopted by all higher institutions. It is used in enhancing the computers and systems from both sides [4].

Data imbalance is a major problem in many of the classification models. It is a scenario which shows the uneven distribution of class labels in the given input dataset. This may lead to biased models. In this paper, we detect the scores of essays to automate manual grading using machine learning models. Before training the model, data is balanced using random over sampling (ROS) approach [5]. ROS balances the dataset by increasing the number of minority class samples. It randomly duplicates the samples in minority classes. After training the models using various machine learning algorithms, the models have been evaluated and based on the more accurate model, we have implemented the flask deployment to find the essay scores. To the best of our knowledge, there is no study that provided a detailed comparison between the essay score prediction models considering both the balanced and unbalanced data. The current paper considers the essay grading as a classification task and classifies the essay as poor/average/good grade.

In summary, the following are the contributions of the current research paper: i) explored various machine learning and natural language processing techniques for grading the essays, ii) handled the data imbalance problem using random over sampling technique, iii) compared the efficiency of five machine learning models in evaluating the essays, and iv) built a web application using flask and deployed the machine learning models.

The remaining work is described in the following sections. Section 2 provides the existing works in AES systems. The proposed methodology is explained in section 3. The evaluation of the proposed work is described in section 4. Finally, section 5 concludes the paper.

2. LITERATURE REVIEW

The project essay grade (PEG) by Ajay [6] began the AES research in 1973. Shermis *et al.* [7] developed a modified version of the PEG and it focused on grammatical checking as well as the correlation

between human raters and the automated evaluation. Intelligent essay assessor (IEA) has been introduced in [8]. The authors evaluated the essays using latent semantic analysis to provide the essay score. E-rater [3], [9] is an essay evaluation system that makes use of natural language processing. It not only focused on content of the essay but also style of the essay. Bayesian essay test scoring system [10] use Bayesian approach for scoring the essays. In the early period, the grading was based on the structure and usage of vocabulary in the essays. They assigned weights to each word. For example, words like “Conclusion”, “summary” will have more weightage and the weight for the word “the” will be 0. The feature sets include measures of organization, vocabulary usage, style, development, and the word length is also considered [11]. Further, the use of natural language inference (NLI) and discourse marker prediction (DM) came into the picture and then NLP tasks were performed [12] in grading the essays. AES is an educational application that is based on the criteria of online evaluation of the essays. It is developed based on two mechanisms: i) critical analysis kit which detects the errors in the usage of grammar not related elements in essay and not desired style of elements and ii) evaluation-rater which rates the essays of the students provided. Both of these methods help students to know their line of memory and writing skills and help to improve their vocabulary [13].

An AES system in detail appears in the Wang and Brown in [14]. Ellis Page is the one who proposed the first AES system in 1960. Most of the AES systems depend on training and data testing for grading essays. Basically, essays are of three types: i) argumentation essays, ii) source rely upon essays, and iii) narration essays. The five attributes that are possibly required for writing an essay are content, choices of words, word fluency, organization, conventions. Essays are the most favorable measure of the students nowadays because they assess one’s knowledge and their writing skills and vocabulary. However, AES systems generate meaningful essay scores and also generate possible measures beyond the human rating [3]. Many studies were conducted regarding the AES systems generated several high rates between human rating and AES systems. Latent semantic analysis is also used along with the several machine learning techniques in developing which is the usage of words that allows comparison between the text information, it processes machine language and transforms the words in essay to statistical language [15]. Bayesian approach is also one of the methods in AES which is related to the classification of the text [16], [17] and calibrating features. Larkey [18] proposed text category technique in which several regression techniques are used having several combinational components. Features that indicate the quality of essays are lexical, syntactic, grammar, and content features. Pre-defined features which are taken into consideration for evaluating essays are errors in grammar, errors in word usage, errors in styles, non-related contents, similar vocabulary, the average length of words, and the total number of words [19]. Any kind of the AES systems associated firstly check the language/vocabulary related features and secondly generate the scoring of the essays based on the firstly checked processes [20].

Back in 2008 [21], Williams was well known for his experience in building an AEG system. By this system, teachers can grade essays with less effort and more efficiency and which indeed saves a lot of time and energy, further to his build, Nash built an addition to this system called automatic essay writer which generates written essay but the question is if the students use an automatic essay writer and generate an essay then in this situation it does not show their true potential thoughts and capacity. AES system depends on many factors like quality of grammar, punctuation, well-formed sentences, and structure formation of words. The AES is skilled by AI and machine learning and historically this system has been trained by exams like GRE, common admission test (CAT), and graduate management admission test (GMAT). AES’s main scope is to grade essays automatically without the involvement of human effort. In 2008, a Spearman correlation research design was done by Wang and Brown [14] using data analysis. The data were analyzed using Spearman rank correlation coefficient tests. The data analysis revealed that there was no statistically significant relationship among the overall holistic scores awarded by the AES tool and the scores awarded by faculty human raters. The scores awarded by two teams of human raters, on the other hand, had a strong correlation. Item response theory (IRT) based essay scoring has been recently introduced in [22]. This model tries to reduce the effect of rater biases on the performance of essay scoring system. A framework for correcting multiple-choice questions (MCQs), Essay questions and equations has been introduced in [23]. It also presented a similarity checking algorithm for equations. The advantages and disadvantages of automatic scoring and feedback is systematically reviewed in [24].

Burrows *et al.* [19] have done the literature on AES systems from six perspectives: dataset, NLP approaches, model creation, grading models, evaluation, and model effectiveness. Hussein *et al.* [25] looked at two types of AES systems: handcrafted features for AES systems and neural networks approaches [26], [27]. They mentioned a few issues but did not go into detail about feature extraction techniques or AES model performance. A framework for implementing automated scoring was proposed by Williamson [28]. This paper lays out a framework for evaluating and implementing automated scoring for high-stakes assessments, with a focus on the standards and methods employed by educational testing service for their e-rater. It offers some insight into the present status of essay evaluation, with a focus on commercially

available AES systems. Ikram and Castle [29] suggested a machine learning strategy based on semantic analysis. An in-depth analysis of AES systems for the past 50 years has been provided in [30]. The authors identified and classified all critical features that must be taken from essays. However, no comparative analysis of all work was presented, and no challenges were explored.

3. METHOD

This section presents the proposed algorithm for grading the essays. The proposed methodology is composed of six steps, namely, data preprocessing, feature extraction, data balancing, building the predictive model, evaluating the model, and deploying the model. The overall workflow of the process is presented in Figure 1. Data is initially preprocessed. To facilitate the feature extraction, count vectorization has been applied. After that, we check whether the data is balanced or not. In order to balance the data, random over sampling approach is applied. Later, the data is split into training and test set. Subsequently, the machine learning models have been trained. After training each model, the performance has been tested and the model with high performance is deployed using Flask framework.

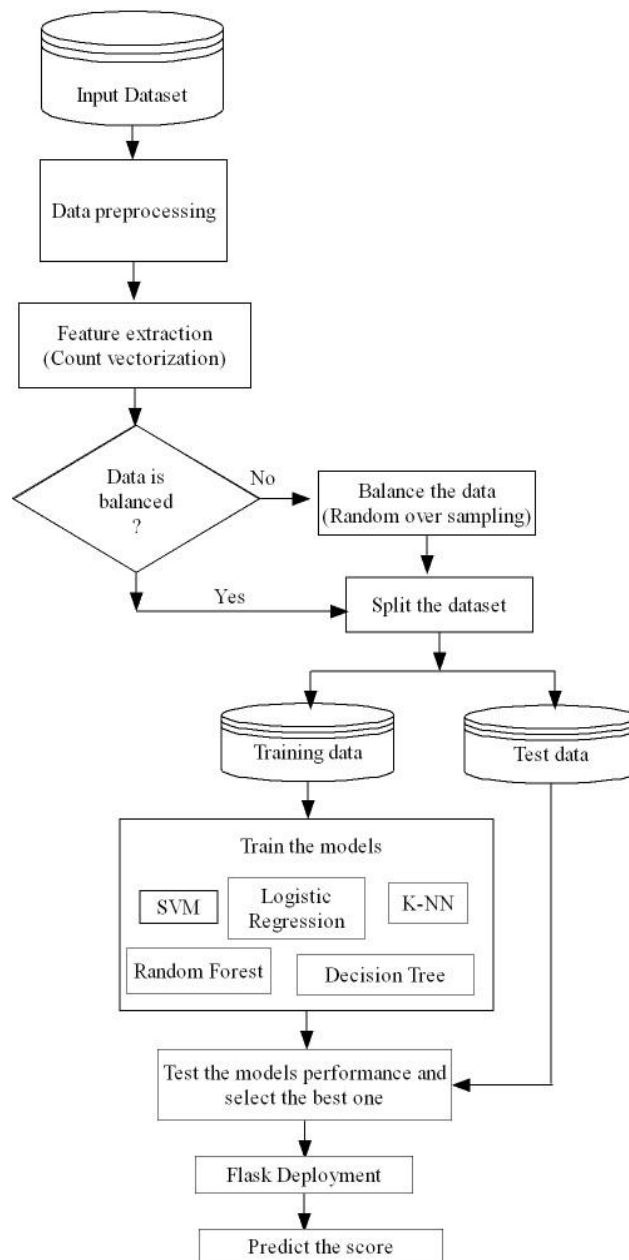


Figure 1. Working of the proposed model

3.1. Data collection

The training data sets consist of eight different essay sets retrieved from Kaggle [31]. Each of these sets was generated from a single prompt. The essays are about an average of 150-600 words per essay. All the essays are written by students studying different levels ranging from primary to secondary school. All the essays were hand graded and were double scored by 2 raters. Each of the data sets adds its own unique features to the essay. The chosen essays were ranging from different levels of complexity to test the capabilities of the essay grades.

3.2. Data preprocessing

Before building a score prediction model, we perform text preprocessing which includes removal of white spaces, converting the uppercase words to lowercase, removal of punctuation marks, tokenization, removing the stop words and stemming. Tokenization refers to dividing each sentence into words. The common words which do not have any importance in distinguishing two essays are called stopwords. The process of converting each word into its root form is called stemming. All the above preprocessing steps have been performed using the NLP library, namely, natural language toolkit (NLTK).

3.3. Feature extraction

Once the data has been cleaned and tokenized, extracting the features from the clean data is critical because the machine does not understand the words but can understand only numbers. Count vectorization aids in the mapping of words to a vector of real numbers, which aids in prediction. This helps in the extraction of key features. The result of count vectorization is a vector whose dimensionality is equal to the size of the vocabulary. A count 1 will be included in the dimension initially, and the count will be incremented by 1 whenever the word is encountered again. The following Pseudocode is used to do count vectorization.

```
- from sklearn.feature_extraction.text import CountVectorizer
- import pandas as pd
- vectorizer=CountVectorizer()
- data=pd.read_csv('data.csv')
- vectorizer.fit(data)
- print(vectorizer.vocabulary_)
- v0=vectorizer.transform(data)
- print(v0.toarray())
```

3.4. Data balancing

The dataset considered in the proposed paper is not balanced. This is because the number of instances in the training dataset for each score is not balanced. This can be illustrated in Figure 2. Figure 2(a) shows the Piechart of the essay score, Figure 2(b) shows the Piechart for the balanced data and Figure 2(c) represents the distribution of essay score against the density distribution. For example, among 12,978 instances of training dataset, 13.38% of the essays have the score 1, 18.84% of the essays have the score 2. Similarly, 21.61% of essays have the score 3 and so on. It is also observed that, we have only one essay with the highest score i.e., 60. This kind of imbalanced dataset will lead to poor predictive performance. Hence, in the proposed methodology, the dataset was balanced using *RandomOver_Sampling* with the *Imblearn* package in Python. In a balanced dataset, each output class denotes the same number of input samples. *RandomOver_Sampling* with *Imblearn* technique in Python balances the dataset by generating the new instances such that all the scores were distributed equally [32]. The balanced dataset after applying the above-mentioned technique is illustrated in Figure 2(b). It is observed that each score has 1.89% of distribution and the number of instances is increased to 1,49,990. So, this balanced dataset helps in finding out the correct accuracy and effective functioning of the model.

3.5. Random forest and decision tree

Random forest (RF) regression is a supervised learning approach that constructs a decision tree (DT) ensemble to perform regression or classification. The training data is divided into random subsets, and each subset is used to build a DT. In a DT regression, data is separated at each node according to a criterion until a continuous score can be predicted. In order to create a forecast, the RF regression model takes the mean of all the decision trees' essay score projections. The data is split according to a condition at each node in DT classification, changing the chances of distinct score classes occurring. The score class with the highest score is traversed through the tree.

3.6. Support vector machine

Support vector machine (SVM) works by mapping all of the essays in an n-dimensional space as a point or vector according to their attributes, then selecting a hyperplane to fit the data and do regression or

categorization. The hyperplane is used to separate the essay vectors in support vector classification into different score categories. The best hyperplane is created by maximizing the distance between multiple types of data points. The model will result in by plotting an essay vector on the same plane and determining the score class it belongs to. The hyperplane is used as a line of greatest fit in support vector regression.

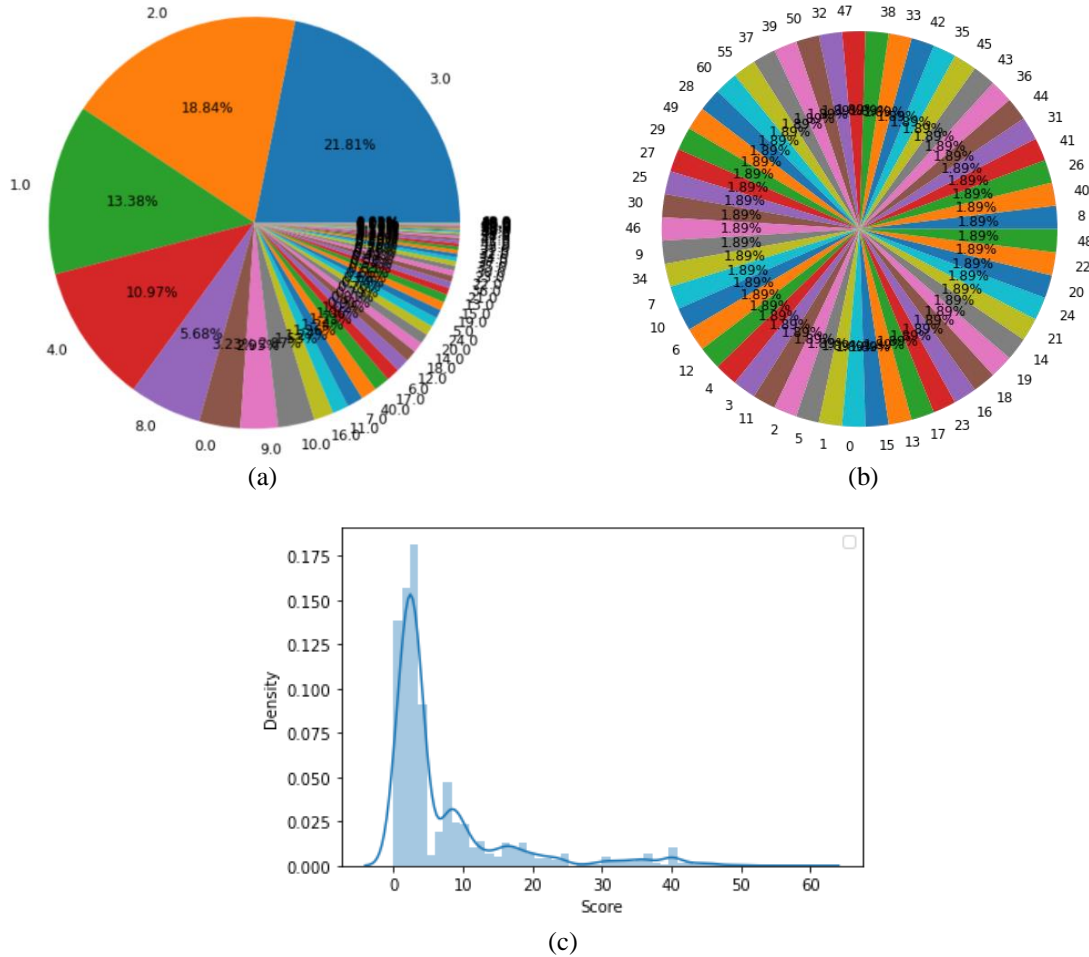


Figure 2. Visualization of input data (a) before balancing, (b) after balancing, and (c) density distribution

3.7. K-Nearest neighbor

The foundation of k-nearest neighbor (KNN) is that a document can be categorized by linking it to other documents using a distance/similarity function. In its most basic version, the intuition behind this strategy is as: given a document *d* (unclassified) and two documents (*c1* and *c2*) that are classified as A and B. If *d* is closer to *c1* than *c2*, it is more likely to belong to class A, according to the distance function. The unclassified document will be allocated to the most common class among its *k* nearest neighbors in KNN classification.

3.8. Logistic regression

Logistic regression (LR) is used to build machine learning models and it is a supervised learning technique. It is used to describe the connection between one dependent variable and one or more independent variables. The concept of utilizing a logistic function inspired the term “logistic regression”. The logistic function is also known as the sigmoid function. A basic S-shaped curve that turns input into a value between 0 and 1 is the logistic function. A logistic regression model can be used only if the outcome is binary. Multinomial logistic regression can predict a dependent variable based on many independent variables. The independent variables might be either a binary variable or continuous variable. It allows multiple categories for a dependent variable. It uses maximum likelihood estimation to find the probability of category membership, just like binary logistic regression.

3.9. Flask deployment

Flask is a software framework developed in 2010 in support of web-based applications. It allows the programmers to create the web applications using a single Python file and makes the life of a programmer easier and flexible. It provides certain tools and features to be used in developing the applications. It does not contain any data abstraction layer and form validation. Flask is called microframework, as its core functionality is simple and also extensible. New functionalities can be added using Flask extensions.

4. RESULTS AND DISCUSSION

4.1. Experimental setup

The hardware used is Intel(R) Core (TM) i5-8250U CPU @ 1.60 GHz 1.80 GHz, RAM-8GB. The proposed methodology is executed on Windows 10 and all the machine learning models are implemented in Python3. During the experiments, we initially tested all the five trained models on both the balanced and unbalanced data. This creates 10 models, among which 5 are built on balanced data and the other 5 are built on unbalanced data. The best model is selected based on the performance metrics such as accuracy, precision, recall and F1-score. In this paper, we compared five algorithms such as LR, RF, K-NN, SVM, and DT. After finding the best model, we deployed it using the essay grading web application which is built using Flask. As of now, we allowed only one essay to be answered and graded the score of the essay, this can be further extended to multiple questions and answers.

4.2. Model evaluation

The trained classification models are evaluated using the performance measures such as accuracy, precision, recall, and F1 Score. The evaluation metrics of the five prediction models before and after balancing the data are given in Table 1 in terms of percentage. The value within the parenthesis is the accuracy obtained after balancing the data. From these tables, it is clear that there is a substantial increase in the performance of all the five models. Also, it is clear that the RF algorithm performs best with an accuracy of 97.67%, precision of 97.62%, recall of 97.67% and F1-score of 97.58% compared to the remaining algorithms. The test data consists of 49,497 essays, the predicted grades and their count by all the models is shown in Table 2. The random forest, support vector machine and K-NN algorithms predict 22,441 essays as poor grade, 22,478 essays as average grade and 4,578 essays as good grade. LR predicts 22,439 essays as poor, 22,480 as average, and 4,578 as good grades. Similarly, DT predicts 22,434 as poor, 22,485 as average, and 4,578 as good grades. Table 3 shows few Essays and their predicted scores and grades using the web-based essay evaluation system. The essay with less than 40% of the maximum marks is assigned a poor grade, the essay with less than 80% of the maximum marks is assigned average grade and more than 80% of the maximum marks is assigned good grade.

Table 1. Evaluation metrics for unbalanced versus balanced data

Algorithm	Accuracy	Precision	Recall	F1-score
Random Forest	47.18 (97.67)	41.84 (97.62)	47.18 (97.67)	41.56 (97.58)
Logistic Regression	46.01 (96.89)	45.25 (96.83)	46.01 (96.89)	45.48 (96.85)
Decision Tree	39.62 (97.01)	38.87 (96.84)	39.62 (97.01)	39.14 (96.9)
Support Vector Machine	53.02 (96.06)	49.58 (96.11)	53.02 (96.06)	48.13 (96.02)
K-NN	25.82 (93.7)	25.42 (93.89)	25.82 (93.7)	23.08 (93.83)

Table 2. Predicted grades

Algorithm	Poor	Average	Good
Random Forest	22,441	22,478	4,578
Logistic Regression	22,439	22,480	4,578
Decision Tree	22,434	22,485	4,578
Support Vector Machine	22,441	22,478	4,578
K-NN	22,441	22,478	4,578

Table 3. Essays and Scores

S. No.	Essay	Score	Grade
1	Bell ring shuffle snap crack	60	Good
2	Laughter everywhere three doors	55	Good
3	Bus flew around tight wound corner	49	Good
4	Well image tell story time	26	Average
5	Dear local newspaper	8	Poor

5. CONCLUSION

This paper provides a detailed comparison between the essay score prediction models considering both the balanced and unbalanced data. To balance the data, we applied random over sampling technique from the Imblearn library. In the proposed model, we made feature extraction using count vectorization and fed these input feature vectors to the score prediction model. Also, a web-based essay scoring system has been introduced using Flask micro web framework. The proposed method has been applied on five machine learning models and observed that Random forest algorithm achieved highest performance than the other four algorithms. Currently, we allowed only one essay to be answered in the proposed web-based essay scoring system. This can be further extended to multiple questions. As a future work, we would like to create datasets of different domains and also work on various feature extraction techniques. We also plan to evaluate different ensemble techniques and stacking classifiers to improve the performance of essay grading system. Also, we will study the methods for providing the feedback to the students' response.




REFERENCES

- [1] K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1882–1891. doi: 10.18653/v1/D16-1193.
- [2] Benjamin S. Bloom, *Taxonomy of educational objectives: The classification of educational goals*, Longmans, Green, 1956.
- [3] Y. Attali and J. Burstein, "Automated essay scoring with E-Rater® V.2.0," *ETS Research Report Series*, vol. 2004, no. 2, Dec. 2004, doi: 10.1002/j.2333-8504.2004.tb01972.x.
- [4] T.-H. Wang, "Developing an assessment-centered e-learning system for improving student learning effectiveness," *Computers & Education*, vol. 73, pp. 189–203, Apr. 2014, doi: 10.1016/j.compedu.2013.12.002.
- [5] A. H. Filho, F. Concatto, J. Nau, H. A. do Prado, D. O. Imhof, and E. Ferneda, "Imbalanced learning techniques for improving the performance of statistical models in automated essay scoring," *Procedia Computer Science*, vol. 159, pp. 764–773, 2019, doi: 10.1016/j.procs.2019.09.235.
- [6] H. B. Ajay, "Strategies for content analysis of essays by computer," Ph.D. dissertation, University of Connecticut, 1973.
- [7] M. D. Shermis, H. R. Mzumara, J. Olson, and S. Harrington, "On-line grading of student essays: PEG goes on the world wide web," *Assessment and Evaluation in Higher Education*, vol. 26, no. 3, pp. 247–259, Jul. 2001, doi: 10.1080/02602930120052404.
- [8] P. W. Foltz, D. Laham, and T. K. Landauer, "The intelligent essay assessor: applications to educational technology," *Interactive Multimedia Electronic Journal of Computer - Enhanced Learning*, vol. 1, 1999.
- [9] D. E. Powers, J. C. Burstein, M. Chodorow, M. E. Fowles, and K. Kukich, "Stumping e-rater: challenging the validity of automated essay scoring," *Computers in Human Behavior*, vol. 18, no. 2, pp. 103–134, Mar. 2002, doi: 10.1016/S0747-5632(01)00052-8.
- [10] L. M. Rudner and T. Liang, "Automated essay scoring using Bayes' theorem," *Journal of Technology, Learning, and Assessment*, vol. 1, no. 2, pp. 1–22, 2002.
- [11] M. Mahana, M. Johns, and A. Apte, "Automated essay grading using machine learning," *Machine Learning Session Stanford University*, pp. 3–7, 2012.
- [12] F. Nadeem, H. Nguyen, Y. Liu, and M. Ostendorf, "Automated essay scoring with discourse-aware neural models," in *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 2019, pp. 484–493. doi: 10.18653/v1/W19-4450.
- [13] S. Mathias and P. Bhattacharyya, "ASAP++: enriching the ASAP automated essay grading dataset with essay attribute scores," in *11th International Conference on Language Resources and Evaluation*, 2019, pp. 1169–1173.
- [14] J. Wang and M. S. Brown, "Automated essay scoring versus human scoring: a comparative study," *The Journal of Technology, Learning, and Assessment*, vol. 6, no. 2, 2007.
- [15] H. Chen and B. He, "Automated essay scoring by maximizing human-machine agreement," *2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pp. 1741–1752, 2013.
- [16] Y. Su, Y. Huang, and C.-C. J. Kuo, "Efficient text classification using tree-structured multi-linear principal component analysis," in *2018 24th International Conference on Pattern Recognition (ICPR)*, Aug. 2018, pp. 585–590. doi: 10.1109/ICPR.2018.8545832.
- [17] Y. Su, Y. Huang, and C.-C. J. Kuo, "Efficient text classification using tree-structured multi-linear principal component analysis," *Expert Systems with Applications*, vol. 118, pp. 355–364, Jan. 2018, doi: 10.1016/j.eswa.2018.10.020.
- [18] L. S. Larkey, "Automatic essay grading using text categorization techniques," in *Proceedings of the 21st Annual International ACM SIGIR Conference On Research and Development in Information Retrieval*, 1998, pp. 90–95. doi: 10.1145/290941.290965.
- [19] S. Burrows, I. Gurevych, and B. Stein, "The eras and trends of automatic short answer grading," *International Journal of Artificial Intelligence in Education*, vol. 25, no. 1, pp. 60–117, Mar. 2015, doi: 10.1007/s40593-014-0026-8.
- [20] M. D. Shermis, J. Burstein, D. Higgins, and K. Zechner, "Automated essay scoring: writing assessment and instruction," in *International Encyclopedia of Education*, Elsevier, 2010, pp. 20–26. doi: 10.1016/B978-0-08-044894-7.00233-5.
- [21] R. Williams and J. Nash, "Computer-based assessment: From objective tests to automated essay grading. now for automated essay writing?," 2009, pp. 214–221. doi: 10.1007/978-3-642-01112-2_22.
- [22] M. Uto and M. Okano, "Learning automated essay scoring models using item-response-theory-based scores to decrease effects of rater biases," *IEEE Transactions on Learning Technologies*, vol. 14, no. 6, pp. 763–776, Dec. 2021, doi: 10.1109/TLT.2022.3145352.
- [23] H. M. Balaha and M. M. Saafan, "Automatic exam correction framework (AECF) for the MCQs, essays, and equations matching," *IEEE Access*, vol. 9, no. 8, pp. 32368–32389, Aug. 2021, doi: 10.1109/ACCESS.2021.3060940.
- [24] M. G. Hahn, S. M. B. Navarro, L. De La Fuente Valentin, and D. Burgos, "A systematic review of the effects of automatic scoring and automatic feedback in educational settings," *IEEE Access*, vol. 9, pp. 108190–108198, 2021, doi: 10.1109/ACCESS.2021.3100890.
- [25] M. A. Hussein, H. Hassan, and M. Nassef, "Automated language essay scoring systems: a literature review," *PeerJ Computer Science*, vol. 5, p. e208, Aug. 2019, doi: 10.7717/peerj-cs.208.
- [26] Y. Su and C.-C. J. Kuo, "On extended long short-term memory and dependent bidirectional recurrent neural network," *Neurocomputing*, vol. 356, pp. 151–161, Sep. 2019, doi: 10.1016/j.neucom.2019.04.044.




- [27] P.-Y. Huang, J. Hu, X. Chang, and A. Hauptmann, "Unsupervised multimodal neural machine translation with pseudo visual pivoting," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8226–8237. doi: 10.18653/v1/2020.acl-main.731.
- [28] D. Williamson, "A framework for implementing automated scoring," *Language*, pp. 1–23, 2008.
- [29] A. Ikram and B. Castle, "Automated essay scoring (AES): a semantic analysis inspired machine learning approach," in *2020 12th International Conference on Education Technology and Computers*, Oct. 2020, pp. 147–151. doi: 10.1145/3436756.3437036.
- [30] B. Beigman Klebanov and N. Madnani, "Automated evaluation of writing – 50 years and counting," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7796–7810. doi: 10.18653/v1/2020.acl-main.697.
- [31] J. Morgan, Lynnvandev, M. Shermis, and T. Vander Ark, "The Hewlett foundation: automated essay scoring," *Kaggle*. 2012. Accessed: Aug. 01, 2022. [Online]. Available: <https://www.kaggle.com/c/asap-aes>
- [32] A. Moreo, A. Esuli, and F. Sebastiani, "Distributional random oversampling for imbalanced text classification," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, Jul. 2016, pp. 805–808. doi: 10.1145/2911451.2914722.

BIOGRAPHIES OF AUTHORS






Udhika Meghana Kotha    received the B.Tech degree in Computer Science and Engineering from SRM University, Guntur, Andhra Pradesh, India. She attended workshops such as Data analysis using Python, Big Data and Hadoop, also she participated in Life skills training program conducted by Barclays company. She did an internship in Web development using Django in Python for 3 months during her graduation from Andhra Pradesh State Skill Development Corporation. She presented a paper titled Essay grading system on research day conducted at SRM AP. She published a patent on Essay grading system during her graduation. Her research areas of interest include data science, machine learning and full stack development. She can be contacted at email: kumeghana08@gmail.com.






Haveela Gaddam    received the B.Tech degree in Computer Science and Engineering from SRM University, Guntur, Andhra Pradesh, India. Currently she is working as a software developer at MX Player, Bangalore, India. She attended workshops such as Data analysis using Python, Big data and Hadoop, also she participated in Life skills training program conducted by Barclays company. She did an internship for 3 months during her graduation from Andhra Pradesh State Skill Development Corporation. She presented a paper titled Essay grading system on research day conducted at SRM AP. She published a patent on essay grading system during her graduation. Her research areas of interest include data science, machine learning and full stack development. She can be contacted at email: haveelagaddam06@gmail.com.



Deepthi Reddy Siddenki    received the B.Tech degree in Computer Science and Engineering from SRM University, Guntur, Andhra Pradesh, India. Currently, she is pursuing her Masters at Illinois Institute of Technology, Chicago. She presented a paper titled Essay grading system on research day conducted at SRM AP. She published a patent on essay grading system during her graduation. Her research areas of interest include machine learning, data structure's and algorithms, data base management systems. She can be contacted at email: deepthireddysiddenni2610@gmail.com.



Sumalatha Saleti    is currently working as an Assistant Professor in the department of Computer Science and Engineering at SRM University, Guntur, Andhra Pradesh, India. She holds Ph.D. from National Institute of Technology, Warangal, India. Her research interests include, data mining, big data analytics, and machine learning. She can be contacted at email: sumalatha.s@srmmap.edu.in.