Detecting AI-Generated Text in Academic Essays via NLP Classification

FERMIN, Lanz Railey A., LEUNG, Robert Nelson R., & VELAQUE, Francis Ira S.

Introduction

The advent of open-source generative text artificial intelligence (Gen-AI) models like ChatGPT has raised the need for robust automated detection mechanisms. As generative text models continue to improve in their ability to mimic human-like speech and writing patterns, there is a growing concern that humans alone will be less and less able to detect AI works. This is particularly crucial in the field of education, where the unchecked use of Gen-AI poses dangers of widespread plagiarism, violation of academic integrity codes, and, in the worst case, fabrication of source materials.

One potential solution lies in the development and application of natural language processing (NLP) classification models to the automated detection of AI-generated text. However, given the severity of typical punishments for academic dishonesty, such models must be held to a particularly high standard of accuracy so as to avoid the misclassification of genuinely human-written essays.

Cingillioglu (2023) addresses this concern by presenting a language model focused on avoiding the misclassification of human-written essays as AI-generated, a critical consideration in upholding academic integrity. The proposed model employs an n-gram Bag of Words (BOWs) discrepancy language model as input to a machine learning (ML) classifier, with a primary emphasis on achieving high accuracy. While other algorithms and AI-generated text detection software demonstrated higher overall accuracies, including performances by OpenAI's GPTZero and Copyleaks ranging from 95.3% to 96.7%, only the SVM algorithm provided a perfect recall with the highest F2 score (97.2%), ensuring no false negatives and accurate classification of all genuinely human-written essays.

However, the research acknowledges its limitations, specifically in the requirement for essays to contain a sufficient word count (not fewer than 400 words) for the n-gram discrepancy language model to function effectively. The arbitrary nature of this threshold prompts a call for future research to explore word count as a potential classification performance indicator (Cingillioglu, 2023).

In the Philippines, the government's National AI Roadmap and the establishment of the National Centre for AI Research (N-CAIR), demonstrate the country's commitment to embracing AI technology as strategic directions which expected the education to tailor its curricula (Estrellado & Miranda, 2023). As the country navigates the integration of AI into education, this project seeks to build upon these findings, refining detection mechanisms and ensuring practical, accurate identification of AI-generated essays without compromising the recognition of genuinely human contributions. In this project, a text data with a length between 261 to 470 words, with a median length of 350 words will be utilized and an n-gram version which considers up to 8 consecutive words ($n = 8$) will be employed.

The primary objective of this project is to develop an automated natural language detection model for AI generated text in academic essays. Specifically, it aims to:

1. Evaluate and compare the performance of five different types of classifiers, namely: Naive Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and a Multi-layer Perceptron (MLP) neural network.

2. Achieve a minimum classification accuracy of 90% on the training set and 80% on the testing set for each of the models.

Data and Methods

A. Dataset and Exploratory Data Analysis

The models in this study were trained using the *LLM - Detect AI Generated Text Dataset* curated by Sunil Thite, most recently updated in November 2023 and made available on Kaggle under an Apache 2.0 License. The dataset comprises 29,000 English essays below 6000 words in length, roughly 17,000 of which are written by humans while the remaining 12,000 were produced by a variety of Large-Language Models (LLMs). Each sample had only two features: the text itself and a 'generated' label to indicate whether or not it was AI generated which allows for the application of supervised learning techniques. A value of 0 for the generated label indicates human-written while 1 indicates AI-generated.

Initial exploratory data analysis (EDA) revealed that the samples in the dataset were skewed towards texts with a length between 261 to 470 words, with a median length of 350 words. There were no null entries in the dataset, however only around 27,000 of the text sample were unique. Duplicate entries were dropped.

B. Data Preprocessing and Model Development

To extract features from the raw text data, the count vectorizer tool of Sci-Kit Learn was used with the standard English stopwords parameter. Given the length of texts analyzed, the 1-8 n-grams Bag-of-Words (BoW) method was employed to better capture contextual relationships between phrases in the dataset. The data was then randomly shuffled and split into training and testing sets in a 7:3 ratio, resulting in roughly 20,000 train and 8000 test samples. The dataset was further tested on 20, 50, and 100 epochs to accurately measure performance with **`max_iter`** set to 60 to ensure convergence in the lfbgs solver.

Several common and traditional classifiers were trained in this study: Naive Bayes, Logistic Regression, Support Vector Machine, Random Forest, and a Multilayer Perceptron. A brief overview to each model is provided in the succeeding subsections.

B.1. Naive Bayes Classification (NB)

The naive Bayes algorithm is considered as the simplest probabilistic classifier, often used in text classification, spam filtering, and sentiment analysis. Along with other types of Bayesian classifiers, it counts the frequencies and value combinations in a given dataset to compute for a set of probability that would best map new pieces of data into a known set of classes or labels (Saritas & Yasar, 2019).

With Bayes' theorem and maximum posteriori hypothesis as its foundations, NB assumes conditional independence – that is, the effect of an attribute value to a given class is independent from the other attributes. This assumption greatly simplifies the computations, although rarely valid in real-world contexts, hence called naive. Nevertheless, its simplicity, fast learning rate, and satisfactory performance in document and text classification enable it to be a viable model (Chakrabarti et al., 2003).

B.2. Logistic Regression (LR)

Extending the notion of linear regression to categorical variables, logistic regression has become one of the most relevant statistical models employed by researchers to analyze binary and proportional datasets. The primary objective of this algorithm is to establish relationships among datapoints by maximizing goodness-of-fit while minimizing variability as much as possible.

Some of the key assumptions in the logistic regression model include: independence between variables, normally-distributed logits, and constant variance (Dreiseitl & Ohno-Machado, 2002). It models the logarithmic odds of an input being classified into a specific class or label, employing the sigmoid function that transforms the input variable into a continuous probability distribution with values ranging from 0 to 1. For binary classification problems, the model only sums the linear combination of input features with weights and bias terms, if applicable. Optimization techniques like gradient descent are then employed to attain the maximum likelihood of correct label classification (Smita, 2021).

B.3. Support Vector Machine (SVM)

Support vector machines, also known as kernel machines, belong to the group of nonparametric supervised learning methods that produce input-output mapping functions from a training set of labeled data. This technique has received significant attention mainly from its promising applications in different fields such as bioinformatics, image recognition, and text mining (Brereton & Lloyd, 2010).

In classification problems, nonlinear kernel functions are often used by SVMs in order to transform input data into an n-dimensional feature space. This allows the input data to be easily separated via maximal-margin hyperplane optimization which, as the name suggests, aim to maximize the margin between examples and the separating hyperplane (Lessmann et al., 2006). This optimization is pivotal to the ability of an SVM model to generalize and make predictions.

One disadvantage of SVMs include its purely dichotomous classification capability and the absence of probability computations for a certain class membership. These are non-issues in the current project, however, since the dataset is intended as a binary classification problem.

B.4. Random Forest (RF)

Interest in the applications of ensemble learning algorithms such as random forest classifiers has been growing over the years due to its enhanced robustness and accuracy over single classifiers like naive Bayes.

Developed by Breiman (2001), a random forest is composed of multiple decision trees, each one in itself acting as a base classifier, that determine the class label of a given unlabeled data. Each tree contributes a vote for the identified class label, and the label with the most votes is considered the generalized prediction of the model (Parmar et al., 2019).

Unlike traditional classification trees, RF strives for diversity by growing trees from subsets of the training dataset created by Breiman's bagging sampling method. Along with bootstrap aggregation and randomization during feature subset selection, the independence among each decision tree is maintained. Hence, its computational efficiency, ability to handle noise, and resistance to overtraining distinguish it from other existing classification methods (Fawagreh et al., 2014).

B.5. Multilayer Perceptron (MP)

To address the limitations of the original perceptron by Rosenblatt (1958), the simple feedforward neural network called multilayer perceptron was developed where non-linear mapping is utilized between inputs and outputs. In between these layers, there exist one or more hidden layers with many neurons stacked together. Every layer transmits its computed results to the subsequent layer, extending continuously through the hidden layers until reaching the output layer.

While perceptrons require the activation function to impose a threshold like ReLU or sigmoid, a multilayer perceptron has the ability to use any bounded and differentiable arbitrary activation function. Hence, this algorithm is considered a feedforward algorithm because inputs are aggregated with the initial weights through a weighted sum and then exposed to the activation function. Backpropagation then happens where the process iterates, adjusting the weights in the network, with the goal to minimize the cost function (Bento, 2021).

C. Model Evaluation

A series of metrics were used to evaluate and compare model performance. First, the raw accuracy scores on both the train and test set were computed for each model. These were compared to the benchmark scores of 80% and 90% respectively as outlined in the objectives. Here, accuracy is simply computed as the ratio between correctly predicted labels to true labels. Afterwards, a detailed confusion matrix was generated for each model to compute the precision (positive predictive value), recall (sensitivity), F1 score, and support. A 10-fold k-Cross Validation was performed as an in-depth comparison method for model performance, with the cross validation score then being computed. Finally, the area under the receiver operating characteristic curve (AUC-ROC) score was computed for each of the models developed and the corresponding plot was generated.

Results

A. Exploratory Data Analysis

Exploratory data analysis (EDA) is a popular methodology employed in a sample data to extract useful information, identify patterns, detect inconsistencies or mistakes, and provide starting points for selecting appropriate machine learning models. In this section, we characterize some important details about the dataset used in this project.

Figure 1 illustrates the distribution of human-written and AI-generated essays contained in the dataset, wherein from the 17 508 are labelled as the former while 11 637 are classified as the latter. As the goal of this project is to develop a rudimentary model that can distinguish between human- and AI-written essays, it is imperative that the dataset has greater human-made instances to ensure that the model learns diverse patterns and characteristics inherent in human-authored content.

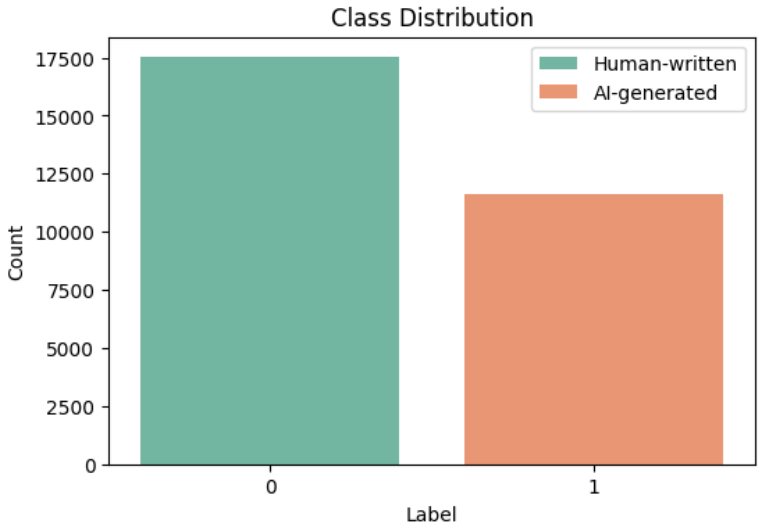


Figure 1. A bar graph showing the distribution of human-written and AI-generated entries contained in the *LLM - Detect AI Generated Text* dataset.

On the other hand, Fig. 2 presents the distribution of text lengths for the two classes in the dataset, along with their respective kernel density estimators (KDE). It can be noted that human-written texts tend to be longer, as its bins and KDE lines are oriented more towards the right-hand side of the graph.

Another interesting feature observed in the histogram is the concentration of AI-generated text lengths in between the third and fourth bin (i.e., around 200–300 words). This may be interpreted with regard to the approach of AIs in generating text, potentially standardized or formulaic.

Text length is a common parameter used in EDA for natural language processing as it contributes to pattern recognition of the model (Liu et al., 2020). Unlike large-language model AIs typically used to create essays, human authors tend to structure their writing with marked variations on lengths of sentences and paragraph structuring, contributing to overall semantic coherence. Further, observed anomalies on text length may also be

indicative whether an essay is written by a human or AI. Unusually short or excessively long passages, which may signal failure or excellence of AI, can be essential for a more effective classification (McCartney, 2018).

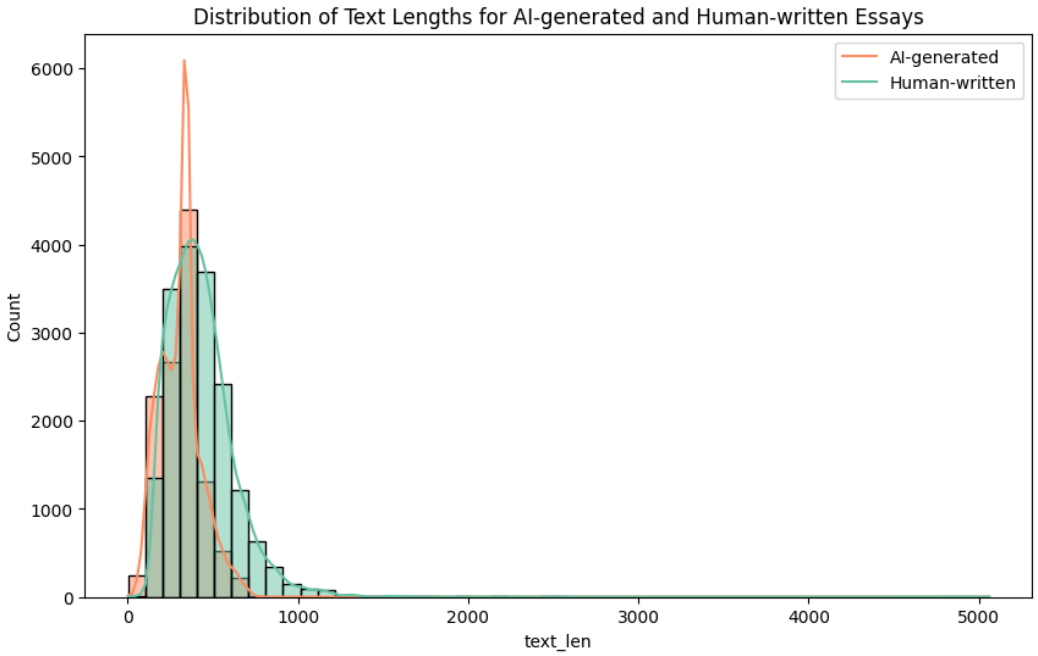


Figure 2. A histogram of the essay text lengths distribution for all instances in the dataset with line plots of kernel density estimators.

Table 1 confirms this text length disparity. We can see that the mean and median values for human-written texts are larger than its AI counterparts, reflective of the nuances embedded within human writings. Hence, it can be inferred that the models used in this project are likely to consider mean text lengths in its classification workflow.

Table 1. Statistical description of text lengths in the dataset using **`df.describe()`** function.

	Entire dataset	**Human-written**	**AI-generated**
count	29 145	17 508	11 637
mean	382.48	430.79	309.80
standard deviation	183.09	200.59	120.93
minimum	1.00	47.00	1.00
25%	261.00	291.00	218.00
50%	350.00	403.00	322.00
75%	470.00	527.00	360.00
max	5061.00	5061.00	764.00

Most NLP projects have included word clouds as a visually-appealing method to analyze text data. By sorting all the existing words in a dataset according to frequency, it provides a starting point for deeper text analysis. Figure 3 shows the most frequently-used words in human-written and AI-generated essays excluding English stopwords, created using the **`WordCloud`** function.

We can notice that some words like "Electoral College", "car", and "student" are at the intersection of these two text corpora. Note that, however, these words do not play a significant role in the classification problem as they are related with the topics each essay in the dataset talks about.

A striking feature of the two word clouds is the disparity between the frequency of "people", despite the fact that the essays in both classes tackle the same topics. Possible explanations behind this finding include LLM limitations, as AI models may not be able to capture nuances and context of certain words or phrases (Su et al., 2023); or training data bias, given that the dataset has a greater number of human-written essays.

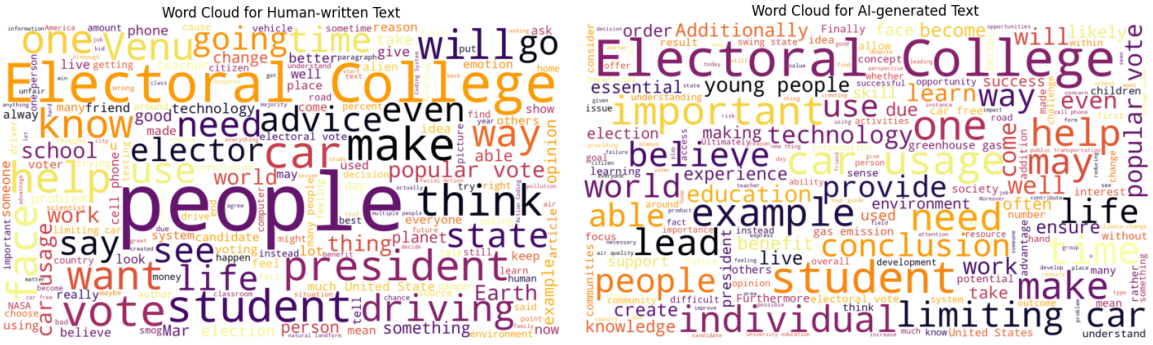


Figure 3. Generated word clouds for human-written text (L) and AI-generated text (R). English stop words were removed from the corpus before generating.

B. Model Evaluation

Table 2. Summary of model performance statistics.

	NB	SVM	LR I	LR II	RF	MLP
Test Accuracy	95. 73	98.27	98.78	98.79	98.43	98.83
Train Accuracy	95.35	99.99	99.95	99.92	98.43	100.00
10-fold Cross Validation	95.31	98.28	98.74	98.74	98.24	98.90
Precision	0.96	0.98	0.98	0.98	0.99	0.99
Recall	0.93	0.98	0.99	0.99	0.97	0.99
F1 Score	0.96	0.98	0.99	0.99	0.98	0.99

Of the six models developed, the MLP classifier consistently performed the best with the highest test accuracy (98.83%), train accuracy (100%), 10-fold Cross Validated Accuracy (98.90%), precision (0.99), recall (0.99), and F1 Score (0.99), while the Naive Bayes classifier performed the worst on each metric.

Precision is a measure of the accuracy of the model's positive predictions, given by the ratio of a model's true positive predictions to its total number of positive predictions. Recall on the other hand is a measure of the completeness or sensitivity of a model, given by the ratio of a model's true positive predictions to the total number of positive cases in the test data. The F1 score is a measure that combines precision and recall to give a more holistic sense of model performance. Each of the six models had test and train accuracies well above the target thresholds of 90 and 80% respectively. Notably, they also all had precision, recall, and F1 scores above 0.9, indicating a high degree of ability to distinguish AI generated texts from human written ones.

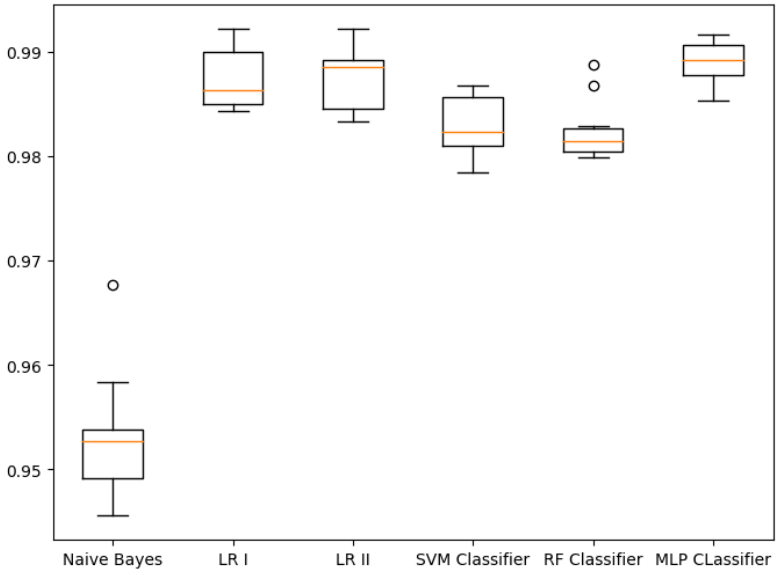


Figure 4. Boxplot of 10-fold cross validation accuracies.

Notably, the performance of the four other models (LR I & II, SVC, and RF) were comparable to that of the more sophisticated MLP classifier, being roughly within 1% of the MLP's test accuracy across each fold. The NB classifier lagged behind all five other models across each epoch. Despite its limitations, the NB classifier's inclusion in the analysis provides valuable insights into the trade-offs associated with model simplicity, as it stands as the least complex model among the six.

The MLP had the smallest variations in test accuracies across folds (as indicated by the visual whisker spread in Fig. 4), suggesting it also had the most consistent accuracy of all the models. Examining the MLP classifier's performance further, it not only achieved the highest accuracy but also showcased remarkable stability across folds. The minimal variations in test accuracies, evident in the visual representation in Figure 4, underscore the MLP's consistent and reliable predictive abilities. This suggests that the MLP model not only

excels in overall accuracy but also offers a dependable performance across diverse subsets of the dataset.

B1. Confusion Matrices and Epoch Consistency

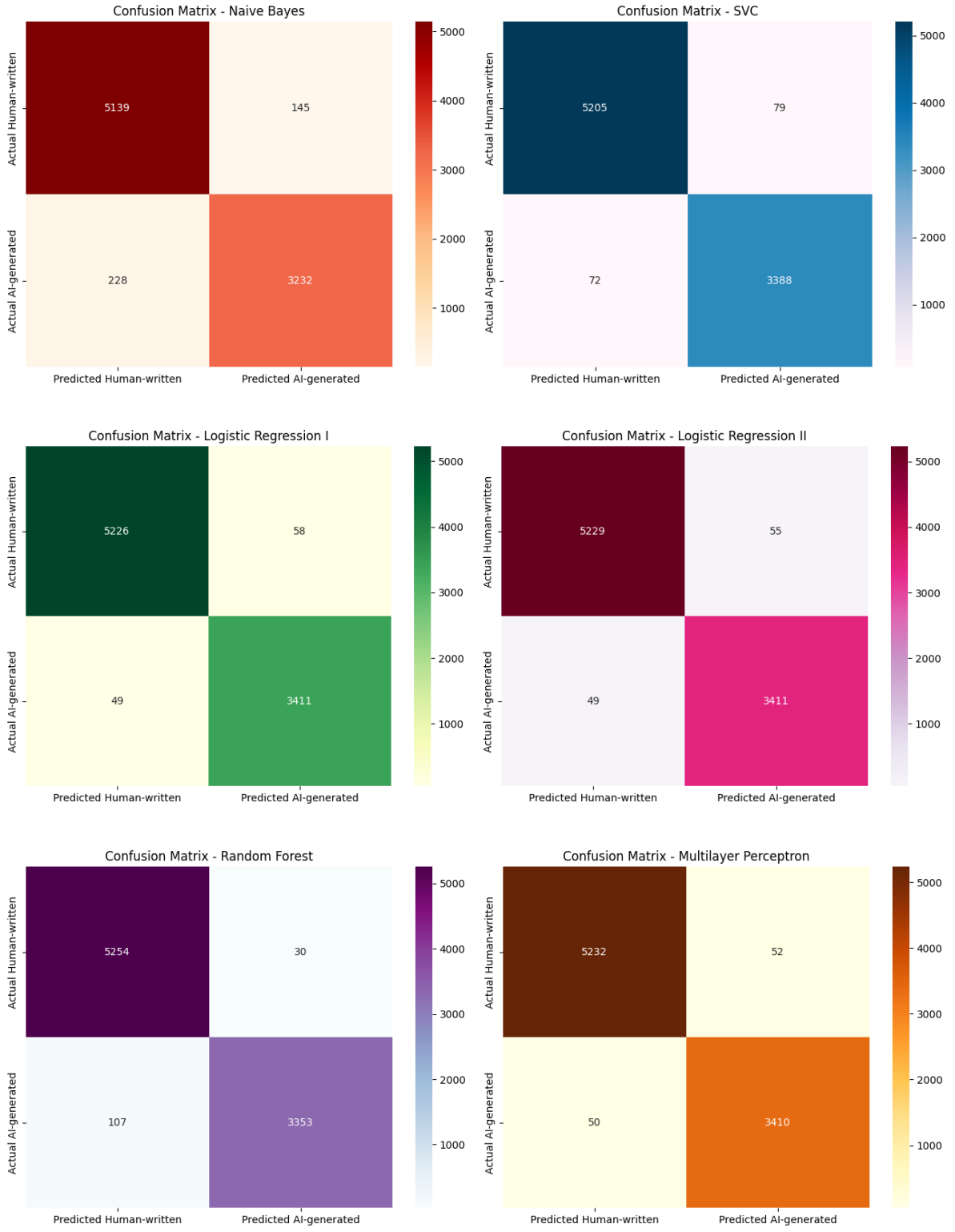


Figure 5. Confusion matrices for the six models (arranged from top-bottom, L-R): Naive Bayes, SVC, Logistic Regression I (Lasso), Logistic Regression II (Ridge), Random Forest, Multilayer Perceptron.

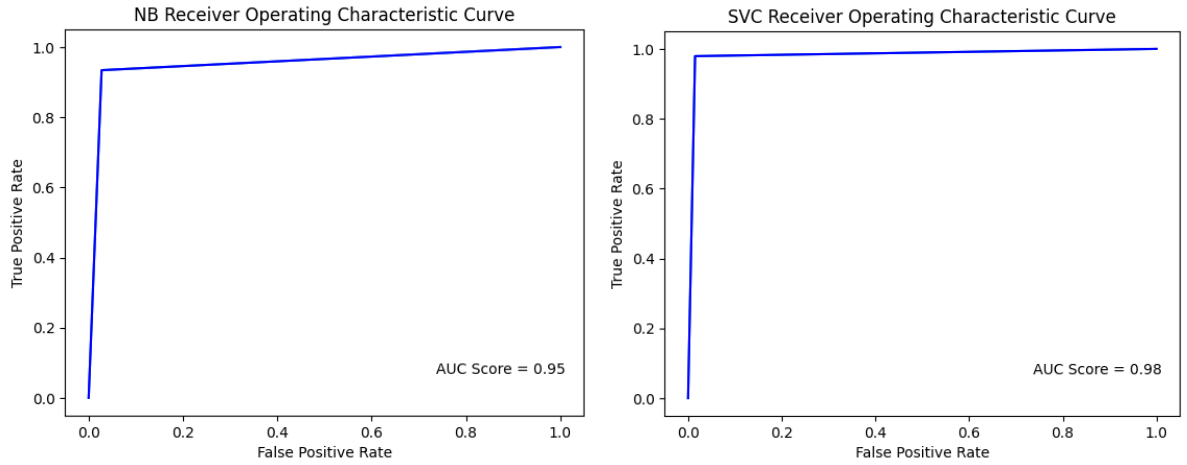
The high accuracy of these models is derived from the confusion matrices in Fig. 5. While our models exhibit commendable performance in discerning between human-written and AI-generated essays, it's crucial to acknowledge that there are intricacies and challenges in achieving absolute precision in distinguishing between these categories with Naive Bayes performing worst having exceeded the one-hundred mark for both false negatives and false positives. This limitation is evident even with accuracy ranging from 98-100% in all metrics. Delving into the specifics of the Multilayer Perceptron (MLP), despite its remarkable performance across different epochs—20, 50, and 100—maintaining an accuracy of 98% and a minimal loss of 1.19 x 10^-5, it still remains prone to misclassification. MLP presents 52 false positives and 50 false negatives, revealing the persistence of misclassification challenges even with increased training iterations.

False negatives imply instances where the model mistakenly identifies AI-generated essays as human-written. This misclassification can have implications, particularly in contexts such as academic integrity, where the consequences of overlooking AI-generated content are significant.

Conversely, false positives highlight the model's tendency to misjudge human-written essays as AI-generated, potentially leading to unwarranted accusations of plagiarism or academic misconduct. This nuanced landscape underscores the importance of not solely relying on global accuracy metrics but rather understanding the specific types of errors each model may encounter. It prompts a more thorough consideration of the practical implications of model misclassifications and the necessity for ongoing refinement to address these intricacies in real-world applications.

B2. AUC–ROC

The receiver operating characteristics curve (ROC) is a visual representation of a model's True Positive and False Positive Rates at varying classification thresholds. The area under the ROC, known as the area-under-curve (AUC) score provides an aggregate measure of model performance across all such thresholds. AUC ranges from 0 to 1, with an AUC of 1 representing perfect performance while an AUC of 0.5 represents random guessing between categories.



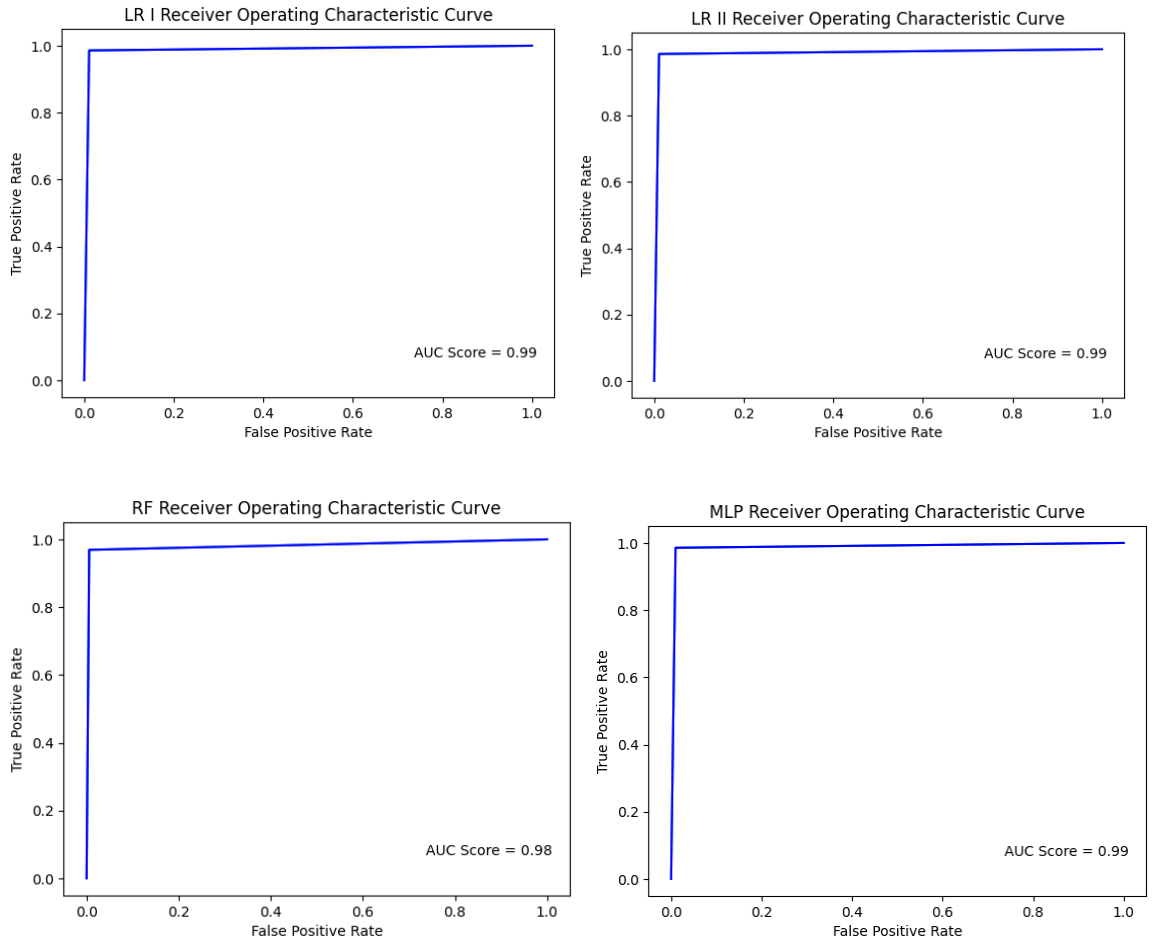


Figure 6. AUC–ROC Plots with respective scores for each of the six models.

Each of the six models in Fig. 6 achieved high AUC scores of above 0.9, with the MLP, LR I, and LR II models all achieving a score of 0.99 which represents almost perfect class discrimination. The performance of these models was closely followed by the SVC and RF classifiers at 0.98. The NB classifier again had the lowest score of 0.95.

Conclusion

The aim of this paper was to develop a machine learning model capable of distinguishing between human- and AI-generated academic essays to a high degree of accuracy. Six different types of models were developed, all of which performed well above the target thresholds of 90% training accuracy and 80% test accuracy. Of them, the Multi-layer Perceptron Classifier performed the best across all different evaluation metrics used, with a test accuracy of above 98%.

However, it is important to note that all models developed in this project remain susceptible to misclassification, as evidenced by the non-zero values in false positives and false negatives in their respective confusion matrices. It encourages a more in-depth examination of the real-world implications stemming from model misclassifications and underscores the imperative for continuous refinement to navigate these subtleties effectively in practical applications.

References

Bento, C. (2021, September 21). *Multilayer Perceptron Explained with a Real-Life Example and Python Code: Sentiment Analysis*. Towards Data Science. https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141

Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5-32.

Brereton, R. G., & Lloyd, G. R. (2010). Support Vector Machines for classification and regression. *Analyst*, *135*(2), 230–267. https://doi.org/10.1039/B918972F

Chakrabarti, S., Roy, S., & Soundalgekar, M. V. (2003). Fast and accurate text classification via multiple linear discriminant projections. *The VLDB Journal*, *12*(2), 170–185. https://doi.org/10.1007/s00778-003-0098-9

Cingillioglu, I. (2023). Detecting AI-generated essays: the ChatGPT challenge. *International Journal of Information and Learning Technology*, *40*(3), 259-268. https://doi.org/10.1108/IJILT-03-2023-0043

Dreiseitl, S., & Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: A methodology review. *Journal of Biomedical Informatics*, *35*(5), 352–359. https://doi.org/10.1016/S1532-0464(03)00034-0

Estrellado, C. J. P., & Miranda, J. C. (2023, May 06). Artificial Intelligence in the Philippine Educational Context: Circumspection and Future Inquiries. *International Journal of Scientific and Research Publications*, *13*(5). http://dx.doi.org/10.29322/IJSRP.13.05.2023.p13704

Fawagreh, K., Gaber, M. M., & Elyan, E. (2014). Random forests: From early developments to recent advancements. *Systems Science & Control Engineering*, *2*(1), 602–609. https://doi.org/10.1080/21642583.2014.956265

Lessmann, S., Stahlbock, R., & Crone, S. F. (2006). Genetic Algorithms for Support Vector Machine Model Selection. *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 3063–3069. https://doi.org/10.1109/IJCNN.2006.247266

Liu, T., Liang, Y., & Yu, Z. (2020). The Influence of Text Length on Text Classification Model. In J. Wang, L. Chen, L. Tang, & Y. Liang (Eds.), *Green, Pervasive, and Cloud Computing – GPC 2020 Workshops* (pp. 79–90). Springer. https://doi.org/10.1007/978-981-33-4532-4_8

McCartney, A. (2017). "How Short is a Piece of String?": An Investigation into the Impact of Text Length on Short-Text Classification Accuracy. *Dissertations*. https://arrow.tudublin.ie/scschcomdis/116

Saritas, M. M., & Yasar, A. (2019). Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification. *International Journal of Intelligent*

Systems and Applications in Engineering, *7*(2), Article 2. https://doi.org/10.18201//ijisae.2019252786

Su, J., Zhuo, T. Y., Mansurov, J., Wang, D., & Nakov, P. (2023). *Fake News Detectors are Biased against Texts Generated by Large Language Models* (arXiv:2309.08674). arXiv. https://doi.org/10.48550/arXiv.2309.08674