Using Handwriting Features to Predict Student Performance in IT and Engineering Domain

May 23, 2022

Candidate: Naira Khachatryan
BS Data Science
American University of Armenia
Yerevan, Armenia
naira_khachatryan@edu.aua.am

Supervisor: Suren Khachatryan
Computer and Information Science
American University of Armenia
Yerevan, Armenia
skhachat@aua.am

Program Chair: Habet Madoyan
BS Data Science
American University of Armenia
Yerevan, Armenia
hmadoyan@aua.am

***Abstract*—This paper represents a research project done in the field of handwriting analysis with the purpose of designing models which could assess student performance in IT and engineering majors. The anonymous data was provided by the American University of Armenia, and the handwriting samples were from midterm exams of two different university subjects. Various machine learning approaches, such as Logistic Regression, Support Vector Machines, Random Forest, Decision Trees, K Nearest Neighbors, and Multilayer Perceptron, and statistical methods were used in order to design a predictive model which could infer student performance from their handwriting characteristics. The main findings were about the impact of the choice of the data, handwriting features and machine learning models. The research was divided into three stages, and each of those had a different dataset and approach. First stage was concerned with grade prediction, and second stage was about general student performance prediction. Many things learned during the first approach were applied during the second one. Some handwriting features were no longer considered, some domain specific handwriting features were added during the seconds stage, and in general directions for defining new features of handwriting in future were discovered. During the last stage of analysis, with more data points involved, it was possible to build a machine learning model with Random Forest Classifier with an accuracy score of 79% in predicting student performance. However, the dataset was limited and contained very small number of samples, which was a huge limitation for the study. Therefore in future stages, more data will need to be collected in order to get better results.**

***Index Terms*—handwriting analysis, machine learning, student performance, statistics**

I. Introduction

Handwriting is one of the most unique things every person owns. Different characteristics and types of handwriting have been observed starting from seventeenth century in order to decode people, their personality and much more. In fact, 400 years ago Camilo Baldi managed to write a book about methods of handwriting analysis. (Joshi et al., 2015)

Identify applicable funding agency here. If none, delete this.

This research project is concerned with finding ways of predicting student grade or performance in the technology and engineering domain with the help of student handwriting samples. Creating a model which could assess one's performance in the field of technologies and engineering based on their handwriting could be useful tool for individuals and institutions. The data was taken from the American University of Armenia, which is an English speaking university, so the handwriting samples were in English. However, the native language of the writers, in majority of the cases, was not English.

The handwriting characteristics used for the first stage of model design were taken from graphology, where they are used for personality detection. The aim with the first project was to design machine learning model which could use a few handwriting features to either predict student's letter grade or whether the students have passed or failed the exam. The choice of machine learning methods was based on the size and specific traits of the data. For this stage Random Forest, Logistic Regression, Support Vector Machine and Multilayer Perceptron classifiers were used. The results of the first stage were promising in terms of numbers and accuracy scores, however were not reliable due to the imbalance in the data and skewed distribution of some features. This was proven using statistical methods such as Receiver operating characteristic (ROC) curves and calculation of the Area Under the Curve (AUC). The findings were summarized and all the knowledge gained was applied during the seconds stage.

The purpose of second stage was to predict general student performance, instead of performance in the scope of one single exam. This was done to avoid bias that could be caused due to semester, subject, exam date and many other factors. New handwriting characteristics were used on top of what was used in the first stage. Some handwriting characteristics which had skewed distributions were not considered for the analysis at this stage. K Nearest Neighbors and Decision Trees were added to the methods already used during the first stage. Also, MLP was trained with batches to improve the learning and increase the

predictive power. For this stage the accuracy scores obtained were smaller for testing datasets compared to training datasets for all models except from MLP, and the assumption was that the models overfit the training data. This was proven with the ROC curve and AUC calculation. The MLP model had higher accuracy score when tested on test dataset, compared to test on the train dataset. However, testing on the training dataset landed a very small accuracy score, which showed that in this case the model underfitted. In addition, Ridge and Lasso regularization models were trained, in order to avoid underfitting or overfitting. The small accuracy scores landed by them also proved that the models did not have much success learning based on the feature set provided. This was important to understand how to organize the future work in terms of feature selection, data collection and model design in order to have better results.

During the last stage of analysis it was possible to construct a binary classification model with the help of Random Forest Classifier which was able to predict the performance of the students based on four main handwriting features with an accuracy of 79%. Some of the four features were newly added to the dataset, and even though the model was not very successful in multiclass prediciton due to insufficient data, it was successful in binary classification problem.

II. Literature Review

At the beginning of the research the main question was about what can handwriting analysis be used for. It was found, that many research groups have put efforts in the problem of identifying personality from handwriting. In fact, there exist software products which solve the problem of reading handwriting with well tuned models. (Digital Scientists, 2022.) Usually, to solve this problem, they use graphology concepts and knowledge, which helps to map the personality from the handwriting with machine learning methods. The process starts with image data preprocessing. Since most of the time handwriting images are on white background with darker ink color, there is an opportunity to use image thresholding in order to transform the data to binary scale. (Joshi et al., 2015) Other methods involve image cropping, noise removal, and Contour Warp affine transformation. (Haridas et al., 2021) A research group used an additional pre-processing technique, called opening the data. This process allows removal of unwanted and helpless characters inside the handwriting such as punctuation marks. Data segmentation, was also used to divide the handwriting based on lines, words and letters. T bar height is an indicator of certain personality, so template matching was used to identify letter "t".

Personality detection from handwriting requires definitions both from technical and graphology perspectives. The handwriting characteristics which can potentially be identified by a model need to be defined, and for supervised learning, the data also needs to be labelled by professionals. However, it is important to note that there is a common pattern in what characteristics of the handwriting are observed in different research papers. Almost all the research works base their models on the following handwriting characteristics:

Baseline

The invisible line along which the letters are written is called baseline. Baseline is the direction of the written lines and can be ascending, descending, or straight, and the personality traits associated with each baseline type correspondingly are optimistic, pessimistic and balanced.

Slant

Slant is the direction of letters in the handwriting. It can be extreme left, left, vertical, right, extreme right. Extreme left is associated with fear of the future, tendency towards rejection and defensiveness. Extreme right indicates impulsive and very expressive behavior, with a lack of self-control and low tolerance for frustration. If the slant is left the personality is reflective and independent, and may have difficulty with expression and adaptation of emotions and sympathy. In comparison, the right slant shows expressiveness, freedom in thought and emotions, extrovertedness, and orientation towards the future. When the slant is vertical a person is said to be rational, and very independent emotionally and in work.

T-bar height

T-bar height is associated with confidence and self-esteem level. T-bar can be in the middle of the letter, lower, higher, or not even crossing the stem, and based on how the person writes the letter it is considered that the person has moderate, low, high self esteem respectively, or is a dreamer with very high hopes.

Margin

Handwriting analysis is done on handwriting which is on blank white paper, so the place from where one starts writing matters and it is called the margin. Margin can be wide left in which case one is considered courageous, it can be wide right, showing avoidance of future and reservedness, the absence of margin shows insecurity and devoting self completely, and finally the even margin shows balance and self discipline.

Letter size

Letter size has to do with one's desire to be noticed. If the letters are large and bold, that can be considered as an indication for high desire to be noticed, small writing on the other hand shows that the author prefers not to be noticed. And the middle size letters are associated with a balance and fit in the world.

There are also some less common handwriting characteristics analyzed in some of the research work, including word spacing, pen pressure. Some more unique characteristics used only in particular papers include line spacing and strokes connecting the letters. (Joshi et al., 2015) (Champa and AnandaKumar, 2010)

However, no studies were found, which were concerned with the problem of whether it is possible to predict student grade or performance based on student handwriting and whether that can also be explained by the personality descriptions. One research project was found which tried to analyze whether poor handwriting influences students' score reliability in mathematics. They conducted a survey and as a result found out

that poor handwriting affects their overall achievement in school mathematics hence negatively affects their educational progress. However, this study was done for secondary school students, only for mathematics and the research locale was Nigeria. Their conclusion was that schools should teach handwriting skills and parents should also be concerned in teaching their kids to write properly in order to help them perform better in mathematics. (Oche, 2014)

III. Materials and Methods

A. AI midterm exam grade prediction, AUA data (50 samples)

The first stage of the research project was done based on 50 students data, from Artificial Intelligence midterm exam at the American University of Armenia. The data was anonymous and taken from the university. The handwriting characteristics were manually assessed based on graphology principles learned during the research of handwriting analysis. The data included the handwriting characteristic and its corresponding reference with personality trait based on graphology domain. Four main handwriting characteristics were taken for model design, those included T-bar height, letter size, baseline and slant. The personality traits corresponding to these handwriting characteristics were described in the Literature Review section.

During exploratory data analysis different features distributions were observed in order to find if there is any imbalance in the data (Figure 1).

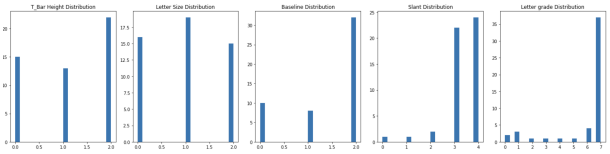

Figure 1. Distribution of features for AI midterm dataset features

In addition, a Pearson correlation matrix was drawn in order to assess what were the correlations between different variables, including the dependant variable (Figure 2).

	T_Bar Height	Letter Size	Baseline	Slant	Grade	Letter grade	Pass no pass 60
T_Bar Height	1.000000	0.153869	-0.031653	0.046832	-0.247520	0.102509	-0.102862
Letter Size	0.153869	1.000000	-0.080903	0.291143	-0.029209	0.061408	-0.097332
Baseline	-0.031653	-0.080903	1.000000	0.046388	0.174230	-0.048097	0.046539
Slant	0.046832	0.291143	0.046388	1.000000	-0.087405	0.260925	-0.150827
Grade	-0.247520	-0.029209	0.174230	-0.087405	1.000000	-0.727223	0.735540
Letter grade	0.102509	0.061408	-0.048097	0.260925	-0.727223	1.000000	-0.775114
Pass no pass 60	-0.102862	-0.097332	0.046539	-0.150827	0.735540	-0.775114	1.000000

Figure 2. Correlation of features for AI midterm dataset features

The dependent variable was the grade, however, considering the sample size was only 50 rows, the grade variable was clustered into classes by two methods. The first approach was to convert the numerical grade into a letter grade based on American system, which made 11 classes from which in the data only 8 were present. And the second approach was to simply mark the grade as pass, if it was higher than 60, or no pass otherwise. It was also important to assess the class imbalance. Figure 3 shows what was the distribution of students who passed or failed to pass the midterm exam.

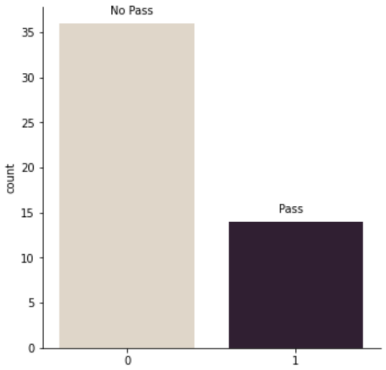

Figure 3. Pass/No Pass results distribution for 50 students

Figure 4 shows what was the amount of students in each letter grade class, where 7 corresponds to the class of Fail (F).

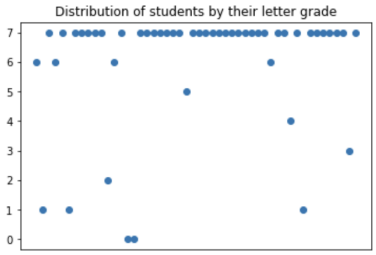

Figure 4. Letter grade distribution for 50 students. Each level on y column corresponds to a letter grade.

The data was randomized and 25% was chosen to be the test dataset, therefore 75% was the train dataset. Four machine learning methods were used to train and test the model, including Multilayer Perceptron, Logistic Regression, Random Forests and Support Vector Classification. All machine learning approaches were trained with different parameter combinations. From traditional machine learning methods, the ones which are tree-based, land better results for small datasets. (Dwivedi, 2020) Thus, Random Forest Classifier was used. The number of trees was in the range 2, 3, 4, 5, 6, 7 since it is known for theory, that the number of trees should not exceed the square root of the sample size. Both "gini" and "entropy" criterion were used.

Additionally, the more complex is a machine learning model, the higher are the chances it will overfit to the data, especially considering the extremely limited amount of the data. Therefore, more simple approaches such as Support Vector Classifier and Logistic Regression were also used. For SVC the kernel was chosen to be "linear", since as a simple approach it works well for small datasets. For Logistic Regression penalty "l2" was added, in order to have regularization and prevent overfitting or underfitting. In Logistic Regression models 'newton-cg', 'lbfgs', and 'liblinear' were used as solvers, and solvers "sag"

and "saga" were removed, because they are commonly used for big datasets in order to make the training fast. (Pedregosa et al., 2011)

A Multilayer Perceptron can work for any amount of data, however, the number of hidden layers and the number of neurons inside them should be adjusted to the size of the data. The maximum number of hidden layers used was 5, and the maximum number of neurons in a layer was 20, and these two parameter values were not present at the same time in any of the models.

Same approaches were used both for letter grade and pass/no pass predictions. For pass/no pass predictions an ROC curve was plotted and AUC was calculated, since the output was binary, and these methods would help to assess the predictive power of the model. (Pedregosa et al., 2011)

In addition, the training was done based on five groups of features; all features, slant only, baseline only, T-bar height only, and letter size only. Accuracy and F1 scores were calculated for all the models.

B. Student Performance Prediction from Handwritten Midterm exam, AUA data (86 samples)

The handwriting data for the second stage of analysis was collected from Introduction to Object Oriented Programming midterm exam samples taken from the American University of Armenia. Each exam paper was labeled with an ID and made anonymous. The professor who thought the class divided students into three groups, based on their general performance and skills; low, medium, high. Based on the learning from the first stage of the research project, handwriting baseline and slant were no longer taken as features, to avoid imbalance in the data and adding bias to the models. Letter size and T-bar height were the two characteristics which continued to be used. Figure 5,6 and 7 show examples of three levels of the T-bar height.

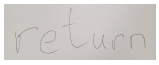

Figure 5. Example of T-bar height "high"

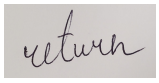

Figure 6. Example of T-bar height "middle"

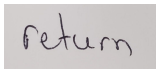

Figure 7. Example of T-bar height "low"

On top of letter size and T-bar height, three new characteristics were taken from handwriting, which did not have any graphology inference, instead they were based on the content of the exam papers. Since the exam was for a course which required programming knowledge, most of the students wrote statement such as "if", "return", and "for" in their solutions, thus letters "t", "f" and "r" were used in all exam samples. One of the characteristics added was "T circle" which indicated whether there was any circle in the "t" or it was written with lines only. It had two values True or False. (Figure 8)

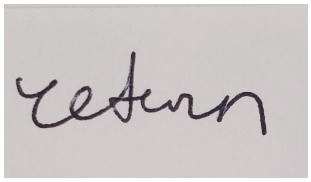

Figure 8. Example of letter "t" with a circle in it

Additionally, the way letters "f" and "r" were written, were also considered as features of handwriting which could potentially explain student performance. For "R type" two classes were considered, handwritten or typed (Figure 9 and 10). For "F type" the description of the letter was provided by three main characteristics. First one was whether the upper part of the letter was linear, circular or oval, second one was whether the bottom part of the letter was linear, circular or oval, and finally the third trait was whether the letter was cut into half in the middle, lower or higher part of it. An example of a letter from class "circle_line_low" is shown in Figure 11.

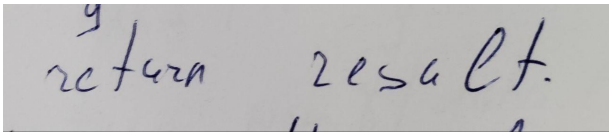

Figure 9. Example of "hand" type of "r"

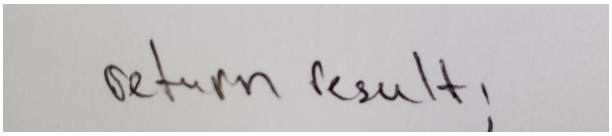

Figure 10. Example of "typed" type of "r"

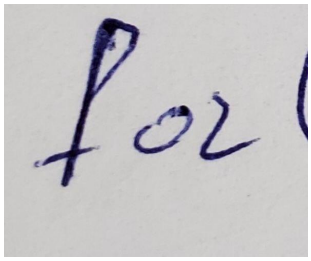

Figure 11. Example of "circle_line_low" type of "f"

In the stage of exploratory data analysis distributions for each of the features were tested in order to remove outliers as well as to exclude features which had skewed distributions

from the training process (Figure 12). The distribution of the dependent variable, student performance, was also checked to be sure there was no class imbalance.

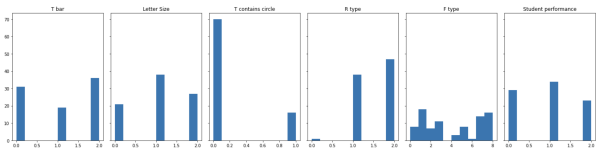

Figure 12. Distribution of features for 2nd stage of the project

Pearson correlation matrix was used in order to identify the correlations between the variables. (Figure 13)

	T bar height	T circle	R type	F type	Letter Size	Student performance
T bar height	1.000000	0.247876	-0.059478	-0.097314	0.266310	-0.150515
T circle	0.247876	1.000000	-0.172342	-0.055833	-0.119415	-0.143891
R type	-0.059478	-0.172342	1.000000	0.407365	0.039432	-0.034698
F type	-0.097314	-0.055833	0.407365	1.000000	-0.115644	-0.078189
Letter Size	0.266310	-0.119415	0.039432	-0.115644	1.000000	-0.012121
Student performance	-0.150515	-0.143891	-0.034698	-0.078189	-0.012121	1.000000

Figure 13. Correlation matrix between variables for the 2nd stage of the project

The data contained 86 samples, with 24 medium, 29 high and 34 low performing students. For this stage of analysis a few machine learning techniques were used. Similar to what was done in the 1st stage of research, tree-based and simple algorithms were used to perform parameter tuning and avoid overfitting. They were tested on all features combined and for each feature separately. First K Nearest Neighbors (KNN), Random Forest classifier (RF), Support Vector classifier (SVC), Logistic Regression (LR), and Decision Tree classifier (DT) were ran on the data.

Furthermore, Random Forest was initialized with number of estimators 3, 4, 5, 7, 8, 9, since it is known from theory that the number of trees should not be more than the square root or log of the sample size. In this case, the threshold was 9, which is largest integer smaller than the square root of 86.

Similar to the first stage of analysis, in case of Logistic Regression "l2" penalty was applied for regularization and "sag" and "saga" solvers were excluded from solvers' list, since they are advantageous for larger datasets. The SVC model was trained with "linear" kernel.

In addition, to validate the interpretation of the results, ROC curves were drawn and AUCs were calculated. Since the classification was multiclass instead of binary, to draw an ROC curve One Versus Rest (OVR) heuristic method was used. (Trevisan, 2022) In addition, the data was trained and tested with Ridge and Lasso regularization models, which prevent unerfitting or overfitting. (Dwivedi, 2020) The alpha parameters of the models were tuned by initializing with different values.

At the end, MLP models were also trained with randomly picked number of hidden layers from range 1-5 and number of neurons from the range 2-20. These numbers were adjusted based on the sample size.

Even though it is a more common practice to use training with batches when the sample size is big, sometimes learning with batches lands higher accuracy on smaller datasets as well. Therefore, the MLP model was also trained with batches and epochs, and the loss of each epoch was calculated to evaluate the learning trend of the model. Number of batches and epochs were set to 20 considering the size of the data.

In addition, at the last step of analysis, the best MLP model was trained on two classes only; high and low performing students. After training the model with the two classes, the predictions of the model for middle class were collected. The professor was also asked to change the evaluation of the students who had performance level "middle" to either "high" or "low". The results were compared in order to understand how well the model was able to classify "middle" class students to either "high" or "low" class compared to the professors assessment.

C. Student Performance Prediction from Handwritten Midterm exam, AUA data (137 samples)

For the last stage of analysis more data was collected from midterm and final exams of students for Introduction to Object Oriented Programming course from years 2015 and 2017. New handwriting characteristics were considered as features, including whether the handwriting was organized and clean (True, False) , whether there was a proper indent in the code (small, normal, absent), and what was the readability level (hard, normal, easy). In addition baseline and slant were considered for analysis, assuming that having more data would help make their distribution less skewed and more normal. The classes remained the same "high", "low" or "medium" performance of the students based on the professor's assessment.

The models and methods used for training and evaluation were the same ones used during stage two. The parameters of Random Forest Classifier were tuned based on data size and number of trees was from range 3-10. Various train and test splittings of the data were tried and the best split was 30% for testing and 70% for training. The best model found was also tested for binary classification on "High" and "Low" classes train data only. Then it was provided the "middle" class for predictions. The professor was also asked to assess middle student performances to be closer to "Low" or "High". The results were gained by comparing model predictions with professors evaluation. In addition, the professor compared each prediction mismatch with his own estimations and commented on whether he agrees with the model prediction or not. Then based on the comments of the professor the reevaluation data for middle class was constructed and the accuracy score was calculated again. In addition, the best model which learned on "High" and "Low" classes only, was also tested on train data set and the accuracy score was calculated.

IV. Results

A. AI midterm exam grade prediction, AUA data (50 samples)

Letter Grade Prediction Results

Each of the models predicting letter grade were assessed based on their accuracy and F1 scores. Interestingly, the results from all the models, with all the parameter combinations and with all the one-feature-based training cycles, landed accuracy score of 0.8 as their best accuracy. The only exception was Random Forest Classifier which had 0.7 as the best accuracy score when trained on all features.

These results were surprisingly positive, so it was interesting to dig deep and understand them more thoroughly. The fact that no matter what was the model, features or the parameters, the final results were the same, could be explained by the imbalance in the data, and its bias towards a certain class, which was 'Fail' in case of Letter Grade prediction and "No pass" in case of Pass/No Pass prediction. The imbalance of classes was also visible in Figure 1.

In addition, the MLP model was also tested on the train dataset, to see if it had predictive power for the dataset on which it learned. The accuracy score landed as a result of testing on train dataset, when trained on all features, was 0.75, which was smaller compared to test accuracy score of 0.8 (Figure 14). This showed that the model didn't learn and had a small predictive power.

	Hidden Layer Sizes	Max Iter.	Activation	Solver	TS Accuracy	TR Accuracy	F1	Loss
308	7	20	logistic	sgd	0.8	0.750	0.711111	1.604336
318	7	30	identity	sgd	0.8	0.750	0.711111	1.171569
4	2	10	logistic	sgd	0.8	0.725	0.711111	1.502498
12	2	15	logistic	sgd	0.8	0.725	0.711111	1.487558
13	2	15	logistic	adam	0.8	0.725	0.711111	1.487923

Figure 14. Best accuracy score MLP models for Letter grade prediction

Pass/No Pass Prediction Results

In case of Pass/No Pass prediction the classification was binary. Support Vector Classifier, Random Forest Classifier, and Logistic Regression had maximum accuracy scores of 0.8 The only exceptions were the best MLP models which managed to get an accuracy score of 0.9 (Figure 15). As it can be observed, even though the accuracy of testing on test dataset was very high, when the accuracy score was estimated based on the testing on train dataset, for the three best MLP models the scores were 0.475, 0.575 or 0.600. These low accuracy scores indicated that the model failed to predict the outputs for the dataset on which it learned. In other words, the model underfitted the data, and the high accuracy score on test dataset was simply a result of imbalance of the classes like in case of Letter grade prediction.

	Hidden Layer Sizes	Max Iter.	Activation	Solver	TS Accuracy	TR Accuracy	F1	Loss
508	15	30	logistic	sgd	0.9	0.600	0.886275	0.682743
411	9	30	tanh	adam	0.9	0.575	0.886275	0.678959
402	9	20	tanh	sgd	0.9	0.475	0.886275	0.697745

Figure 15. Best accuracy score MLP models for Pass/No Pass prediciton

Further, Logistic Regression and SVC results were checked with ROC curves and AUC calculation. The AUC for Logistic Regression was 0.59, and for SVC it was 0.40 (Figures 16 and 17). It is known from theory, that an AUC value near 0.5 shows that the model is unable to make predictions, and that it has not learned. (Mandrekar, 2010)

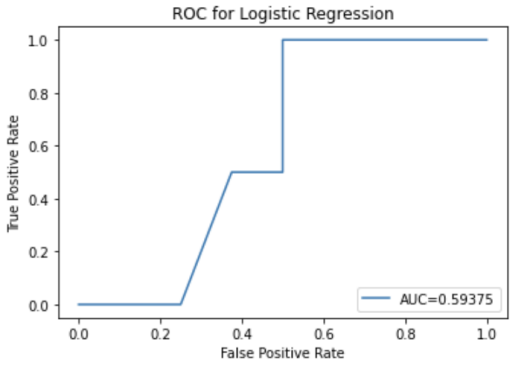

Figure 16. ROC and AUC for Logistic Regression

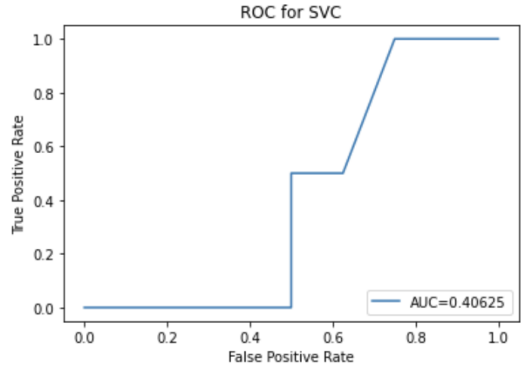

Figure 17. ROC and AUC for Support Vector Classifier

To summarize, both when doing letter grade prediction and pass/no pass prediction based on 50 samples from AUA AI midterm exam, all the models used (RF, MLP, SVC, LR) failed to gain sufficient predicitive power. The exploratory data analysis with statistical methods such as ROC and AUC calculations and performing a test on the train dataset, showed that the accuracy results, even though very high, but were not reliable. There was an imbalance in the data and considering its small size it was impossible to tackle the problem of designing a model which would learn on it. This bias in the data was coming from external factors, such as course content, professor's grading approach, or even outside events which could affect the class performance, including something going on in the world or the country which could affect the learning process overall. This data imbalance made the model overfit towards "Fail" class in case of Letter grade prediction and "No pass" class in case of Pass/No pass prediction. Moreover, it was found that some of the variables had left-skewed distributions, including the baseline and slant, since majority of people tend to write on a right or straight baseline, and the letter slant is usually either vertical or right-oriented. This was one of the important things learned from the study, and was used for the

second stage of data collection and analysis, where baseline and slant were no longer considered. In addition, to reduce the bias which could come from a course content, semester, or professor, for the second stage of research the grade of a specific class was no longer taken as the variable to be predicted. Instead, a more generic indicator of student's performance was considered.

B. Student Performance Prediction from Handwritten Midterm exam, AUA data (86 samples)

For each of the models trained, the testing was done based on train and test datasets. The models used for the problem were KNN, Random Forest, Decision Tree, Logistic regression and Support Vector Machine classifiers. The best accuracy scores landed by each of the models tested both on train and test datasets are shown in the Figure 18. In all cases the accuracy scores of train dataset was multiple times better compared to the accuracy score from testing on test dataset. The accuracy scores on test dataset were small, none of those exceeded 50%. This was a sign that the models overfitted to the train dataset.

Model	Accuracy score on Train dataset	Accuracy score on Test dataset
KNN	0.68	0.13
Random Forest	0.88	0.29
Decision Tree	0.78	0.27
Logistic regression	0.42	0.31
Support Vector Machine	0.45	0.29

Figure 18. ROC curve and AUC calculations for Logistic Regression model using OVR heuristic

To check how much predictive power models had, ROC and AUC were calculated with the help of One Versus Rest (OVR) heuristic method for Logistic Regression and Support Vector Machines. These two models were chosen, since they had the best accuracy scores on the test dataset. As it can be observed in Figure 19, in case of Logistic Regression AUCs for three classes were 0.30, 0.41 and 0.67. These are near 0.5 which indicates that the model did not have much predictive power. And for the SVC model, one of the areas was 0.83 for class 2 (medium performance), which was a positive result, however for classes 1 (high) and 0 (low) the areas were 0.23 and 0.36 correspondingly, which indicated that the SVC model also failed to gain predictive power for based on the given data (Figure 20).

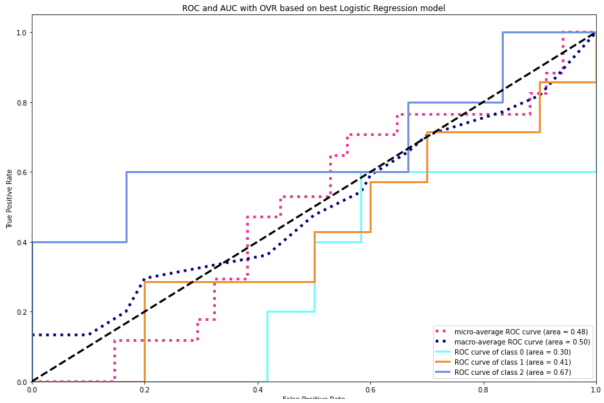

Figure 19. ROC and AUC calculations for Logistic Regression model using OVR heuristic

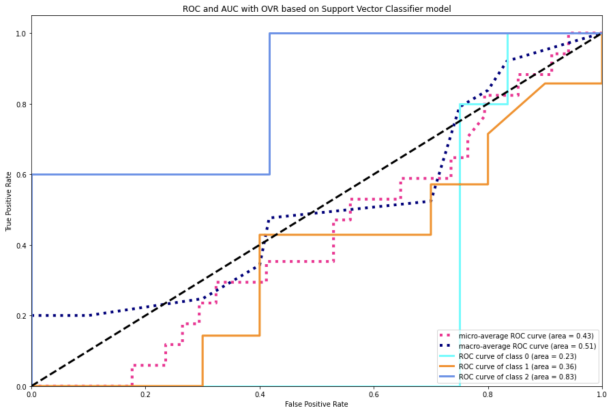

Figure 20. ROC and AUC calculations for Support Vector Classifier model using OVR heuristic

All these results showed that the models overfitted to the train data and failed to predict on the test data. To avoid overfitting Ridge and Lasso models were also trained on the train dataset and tested on both train and test datasets. The Ridge model score for training dataset was 0.039 and for testing it was -0.1 (Figure 21). The Lasso model provided training score of 0.039 and testing score of -0.09 (Figure 22). These results supported the hypothesis that the dataset was such that the models did not manage to gain any predictive power based on it.

```
Ridge model: [-0.08532349 -0.01363351 -0.03725157 -0.10817772]
Ridge Regression Model Training Score:  0.03989509355938847
Ridge Regression Model Testing Score:  -0.1074107529656354
```

Figure 21. Ridge model results

```
Lasso model: [-0.07482212 -0.         -0.03619175 -0.09278225]
Lasso Regression Model Training Score:  0.03935678233191886
Lasso Regression Model Testing Score:  -0.09335022715444374
```

Figure 22. Lasso model results

The best accuracy score, obtained from the models learning on seconds dataset, was 0.6471, and it was two MLP models

with parameters shown in Figure 23. It can be observed that when the models were tested on the training dataset, the accuracy score was 0.35 or 0.30. This showed that there was no overfitting. However, this could be interpreted as underfitting, since the model failed to appropriately predict for the data it was trained on.

	Hidden Layer Sizes	Max Iter.	Activation	Solver	TS Accuracy	TR Accuracy	F1	Loss
4	2	15	relu	sgd	0.6471	0.3529	0.563914	1.220389
56	5	50	relu	sgd	0.6471	0.3088	0.535755	1.276395

Figure 23. MLP best two models parameters and scores

Besides, the MLP was also ran with batches and epochs. The loss calculated represented the sum of losses obtained from the test on training and testing datasets during each epoch. As it is shown in Figure 24 the loss had a fluctuating pattern instead of decreasing one, which showed that the model failed to learn.

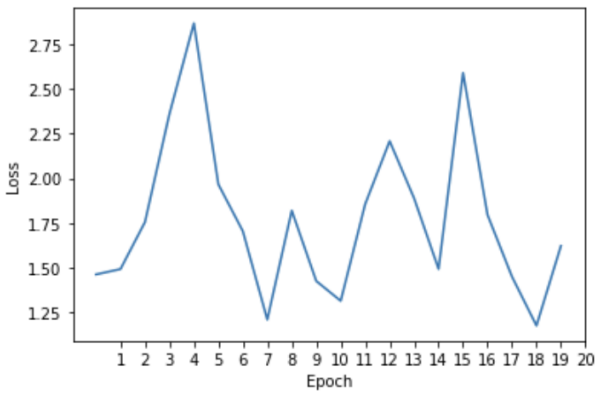

Figure 24. Training and testing Loss per epoch of MLP training with batches

At the last stage of analysis the best MLP model was trained on two classes only; "high" and "low". Then the predictions of the model for middle class were obtained. As a result, the model predicted 0 for all middle class students, which showed that according to the model they were all more similar to the "low" class. The professor estimated 1/3 of the "middle" students" to be rather "high" performing, and 2/3 to be "low" performing. This could indicate that the MLP model had around 66.6% accuracy in prediction. However, this accuracy did not imply that the model had predictive power. This was proved by taking model predictions for "high" and "low" classes as well, and showing that for those too all the predcitions were 0. Overall, the MLP model overtfitted to the class "low" (0) and thus was not very reliable.

These results indicate that all the machine learning models and methods used did not manage to gain predictive power for student performance level based on their handwriting features chosen. For this stage of analysis imbalance in the features distribution and classes to be predicted was taken into account and was not a barrier for the models to learn, however the features selected were not good enough indicators of student performance level.

C. Student Performance Prediction from Handwritten Midterm exam, AUA data (137 samples)

In the last stage of analysis more data was collected, therefore the distribution of the data features and classes was evaluated again. The "Slant" and "Baseline" were added and their distribution was less skewed compared to the initial stage. Also the correlation matrix was drawn and the features "readability", "baseline", "slant", "code indent" had correlation with performance, thus those were chosen for model design (Figure 25).

	T bar height	T circle	R type	F type	Letter Size	baseline	slant	Organized	readability	code indent	Performance
T bar height	1.000000	0.143428	0.045934	-0.020548	0.238624	-0.187133	0.002166	0.145001	-0.025353	0.025833	-0.002534
T circle	0.143428	1.000000	-0.241597	-0.181319	-0.013582	0.037719	0.151018	0.086617	-0.053919	0.129190	-0.142809
R type	0.045934	-0.241597	1.000000	0.440168	0.064374	0.046844	-0.047501	-0.055775	-0.052357	-0.006911	0.028127
F type	-0.020548	-0.181319	0.440168	1.000000	-0.053933	0.062380	0.028304	-0.127802	0.033619	0.034843	0.017413
Letter Size	0.238624	-0.013582	0.064374	-0.053933	1.000000	-0.098502	-0.081713	0.095323	0.151495	-0.012209	0.005504
baseline	-0.187133	0.037719	0.046844	0.062380	-0.098502	1.000000	0.086547	0.127045	-0.064274	0.013551	-0.118090
slant	0.002166	0.151018	-0.047501	0.028304	-0.081713	0.086547	1.000000	0.054171	0.011181	0.102927	-0.181057
Organized	0.145001	0.086617	-0.055775	-0.127802	0.095323	0.127045	0.054171	1.000000	-0.155007	-0.033947	-0.059240
readability	-0.025353	-0.053919	-0.052357	0.033619	0.151495	-0.064274	0.011181	-0.155007	1.000000	-0.039415	0.147734
code indent	0.025833	0.129190	-0.006911	0.034843	-0.012209	0.013551	0.102927	-0.033947	-0.039415	1.000000	-0.087651
Performance	-0.002534	-0.142809	0.028127	0.017413	0.005504	-0.118090	-0.181057	-0.059240	0.147734	-0.087651	1.000000

Figure 25. Correlation matrix for the last stage of analysis

The results landed by the machine learning models indicated that most of the models had a tendency to overfit to the train dataset, since all of them had much higher accuracy scores when tested on the train dataset (Figure 26). The best model was Random Forest Classifier with a 0.41% accuracy score in test dataset. The model used 5 trees and criterion "gini".

Model	Accuracy Score on Train Dataset	Accuracy Score on Test Dataset
KNN	0.62	0.32
Random Forest	0.76	0.41
Decision Tree	0.77	0.27
Logistic Regression	0.55	0.29
Support Vectore Machines	0.33	0.41

Figure 26. Results for model accuracy scores from the 3rd stage of analysis

The ROC curves and AUC calculations for Logistic Regression and Support Vector Machines landed very similar results compared to the second stage of analysis, so those models were not considered for further analysis. The MLP model also failed to gain much predictive power, it had highest accuracy score of 0.4146 on the test dataset, however the same model had accuracy score of 0.3368 on the train dataset.

In the last stage of analysis, the Random Forest model was trained on "Low" and "High" classes only and the predictions for the middle class were obtained. During his first evaluation, the professor estimated 42% of the students in the middle class to be closer to "Low" performance" and 58% of the students to be closer to "High" performance. The accuracy score for model predictions and professor evaluations was calculated and it was equal to 0.526. Then the professor was asked to assess one by one all the predictions of the model which did not match his initial evaluations, and he commented on whether he agrees with the prediction or not. After the reevaluation the professor assessed 37% of the middle performing students to be closer to "Low" performance and 63% to be closer to "High" performance. The accuracy score was calculated again,

and this time it reached 0.79. The model was also tested on the train dataset and landed an accuracy of 0.79.

It was possible to detect certain handwriting features, including "readability", "baseline", "slant", and "code indent", based on which a Random Forest model with an accuracy score of 79% was designed.

V. Conclusion

The research project was divided into three stages. During the first stage a limited handwriting dataset of Artifical Intelligence course midterm exam was taken from the American University of Armenia (AUA). The dataset was for a specific course and the problem was to predict the student letter grade (based on American system), or Pass/No Pass status. The handwriting characteristics were selected based on the graphology knowledge gained during the literature review stage. Machine learning models and their parameters chosen were based on the data size. Exploratory data analysis showed that there was an imbalance of classes, since most of the students failed the exam. In addition, some of the features of the handwriting had skewed distributions, such as handwriting baseline and slant. These factors and the limited sample size made the models to overfit to a certain class, and even though they landed high accuracy scores, they had small predictive power and tended to output the dominating class, which was "F" in case of letter grade prediction problem and "No pass" in case of pass/no pass prediction. The main learning from this stage of research project was that many external factors can make a dataset of one specific course very imbalanced and biased. Factors which could play a role include course content and difficulty level, professors grading methods, and many others including external events in the world or in the country which could impact university, education, emotional and mental states of the students. Moreover, for the next stage of research baseline and slant were no longer considered in the list of handwriting characteristics, considering that in a small sample size they have a skewed distribution and can deviate model prediction.

Considering all the knowledge gained from the first dataset analysis, for the second stage of the research 86 new handwriting samples were taken from AUA, from the class of 2017 Spring semester Introduction to Object Oriented Programming course. The professor teaching the class provided general assessment of student abilities and performance. This evaluation was not based on the specific course grade only, so it removed some of the bias that could come from course specific aspects. New handwriting characteristics were added to the analysis, considering the domain was coding and most of the papers contained handwritten code and similar coding expressions. Different machine learning algorithms were trained, with parameter tuning and adjustments based on data size, however all of them, except from MLP, tended to overfit to the data and failed to land a good accuracy on the test dataset. ROC and AUC calculations also supported the hypothesis that models failed to learn. Ridge and Lasso regularization had very small scores, holding up to the fact that the features had a little correlation with the student performance, and therefore, models did not manage to learn based on them. MLP was the only method which landed an accuracy of 0.6472 for test dataset, and the accuracy score for training was smaller. In case of MLP the model underfitted, since it failed to predict on the dataset it was learning on. Besides, MLP training with batches showed fluctuating trend in loss, and the general pattern was not decreasing, so the loss was not getting smaller, implying that the model was not learning.

In the last stage of analysis, it was possible to design a machine learning model with the help of Random Forest Classifier, which landed an accuracy score of 79% when trained on four features for a binary classification problem.

All these results will play a significant role in the dataset domain choice, collection principles, and in the design of future models.

VI. Future work

The next steps of this research project will include new feature selection from handwriting. These characteristics can either be domain specific, for example how people write the same coding words ("if", "else", "for", "return") in the scope of the same problem solution, or it can be about how they align and indent the code, or lines in general, what is the distance between letters, words, and lines. There is a lot of opportunity to continue work on feature selection and perfect the set of features trying to find the ones that have some correlation with student performance.

Another important thing is to increase the data size and involve more diversity. In case of first two stages, the datasets were coming from specific courses and exams, however as a next steps more subjects/courses, semesters, and exam types can be involved to remove any bias that the day of the exam, the course difficulty, professor grading methods or any other factors could add to the study.

In addition, image processing and pattern recognition methods can be built which will enable the labeling of the data, which will significantly increase the precision and efficiency of data collection process, save a lot of time resources and increase the speed of labeling.

If it will be possible to increase the accuracy score obtained and prove that special handwriting characteristics correlate with student performance in the field of Engineering and IT, then it will be possible to design models which would help predict or evaluate student performance, and this could possibly be a tool used by individuals and institutions.

REFERENCES

[1] Abhijit, P., Mayuresh, R., Archana, S. and Saheel, T., 2022. Personality Prediction Using Handwritten Characters. [online] Ijitjournal.org. Available at: <http://www.ijitjournal.org/volume-6/issue-4/IJIT-V6I4P5.pdf> [Accessed 17 May 2022].

[2] Achinthu Haridas, Sravan S, Arjun R, Mruthula N R, Rohith Muralidharan, 2021, Personality Prediction based on Handwriting using CNN MLP, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH TECHNOLOGY (IJERT) ICCIDT – 2021 (Volume 09 – Issue 07)

[3] Champa, H. and AnandaKumar, D., 2010. Artificial Neural Network for Human Behavior Prediction through Handwriting Analysis. International Journal of Computer Applications, 2(2), pp.36-41.

[4] Digital Scientists. 2022. Handwriting Analysis | AI + Machine Learning | Digital Scientists. [online] Available at: <https://digitalscientists.com/case-studies/handwriting-analysis/> [Accessed 17 May 2022].

[5] Dwivedi, R., 2020. Hands-On-Implementation of Lasso and Ridge Regression. DEVELOPERS CORNER, [online] Available at: <https://analyticsindiamag.com/hands-on-implementation-of-lasso-and-ridge-regression/> [Accessed 17 May 2022].

[6] Dwivedi, R., 2020. How To Implement ML Models With Small Datasets. DEVELOPERS CORNER, [online] Available at: <https://analyticsindiamag.com/how-to-implement-ml-models-with-small-datasets/: :text=This%20means%20tree%2Dbased %20algorithms,with%20missing%20values%20and%20outliers> [Accessed 17 May 2022].

[7] Joshi, P., Agarwal, A., Dhavale, A., Suryavanshi, R. and Kodolikar, S., 2015. Handwriting Analysis for Detection of Personality Traits using Machine Learning Approach. International Journal of Computer Applications, 130(15), pp.40-45.

[8] Lemos, N., Shah, K., Rade, R., Shah, D. (2018). Personality Prediction based on Handwriting using Machine Learning. 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), 110-113.

[9] Maheswari, J., 2019. Breaking the curse of small data sets in Machine Learning: Part 2. Towards Data Science, [online] Available at: <https://towardsdatascience.com/breaking-the-curse-of-small-data-sets-in-machine-learning-part-2-894aa45277f4> [Accessed 17 May 2022].

[10] N. Mandrekar, J., 2010. Receiver Operating Characteristic Curve in Diagnostic Test Assessment. Journal of Thoracic Oncology, [online] 5(9), pp.1315-1316. Available at: <https://reader.elsevier.com/reader/sd/pii/S1556086415306043?token=30F61810F6A03413D9E9F3CEB6608B79F8092AC741297B92DAD0BE2198BB7767979C48D21DEBD51A3A96DA52B5A1EAAForiginRegion=eu-west-1 originCreation=2 0220517174427> [Accessed 17 May 2022].

[11] Sajeevan, S., Wickramaarachchi, W.U. (2022). Detection of Personality Traits Through Handwriting Analysis Using Machine Learning Approach. In: Saeed, F., Al-Hadhrami, T., Mohammed, E., Al-Sarem, M. (eds) Advances on Smart and Soft Computing. Advances in Intelligent Systems and Computing, vol 1399. Springer, Singapore.

[12] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[13] SHARMA, S., 2017. Epoch vs Batch Size vs Iterations. Towards Data Science, [online] Available at: <https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9> [Accessed 17 May 2022].

[14] Sunday Oche, E., 2014. The Influence of Poor Handwriting on Students' Score Reliability in Mathematics. Mathematics Education Trends and Research, [online] Available at: <https://www.researchgate.net/publication/284265735_The_Influence_of_Poor_Handwriting_on_Students%27_Score_Reliability_in_Mathematics> [Accessed 17 May 2022].

[15] Trevisan, V., 2022. Multiclass classification evaluation with ROC Curves and ROC AUC. Towards Data Science, [online] Available at: <https://towardsdatascience.com/multiclass-classification-evaluation-with-roc-curves-and-roc-auc-294fd4617e3a> [Accessed 17 May 2022].

[16] Versloot, C., 2022. How to see if your model is underfitting or overfitting?. GitHub, [online] Available at: <https://github.com/christianversloot/machine-learning-articles/blob/main/how-to-check-if-your-deep-learning-model-is-underfitting-or-overfitting.md> [Accessed 17 May 2022].