applied sciences

MDPI

Article

A Recommendation System for Selecting the Appropriate Undergraduate Program at Higher Education Institutions Using Graduate Student Data

Yara Zayed , Yasmeen Salman and Ahmad Hasasneh *

Department of Natural Engineering and Technology Sciences, Data Science & Business Analytics, Arab American University, Ramallah P.O. Box 240, Palestine
* Correspondence: ahmad.hasasneh@aaup.edu

Abstract: Selecting the appropriate undergraduate program is a critical decision for students. Many elements influence this choice for secondary students, including financial, social, demographic, and cultural factors. If a student makes a poor choice, it will have implications for their academic life as well as their professional life. These implications may include having to change their major, which will cause a delay in their graduation, having a low grade-point average (GPA) in their chosen major, which will cause difficulties in finding a job, or even dropping out of university. In this paper, various supervised machine learning techniques, including Decision Tree, Random Forest, and Support Vector Machine, were investigated to predict undergraduate majors. The input features were related to the student's academic history and the job market. We were able to recommend the program that guarantees both a high academic degree and employment, depending on previous data and experience, for Master of Business Administration (MBA) students. This research was conducted based on a published research and using the same dataset and aimed to improve the results by applying hyper-tuning, which was absent in previous research. The obtained results showed that our work outperformed the work of the published research, where the random forest exceeded the other classification techniques and reached an accuracy of 97.70% compared to 75.00% on the published research. The importance of features was also investigated, and it was found that the degree percentage, MBA percentage, and entry test result were the top contributing features to the model.

Keywords: recommendation system; educational data mining; machine learning; undergraduate program forecasting; support vector machine; decision tree; random forest

check for updates

Citation: Zayed, Y.; Salman, Y.; Hasasneh, A. A Recommendation System for Selecting the Appropriate Undergraduate Program at Higher Education Institutions Using Graduate Student Data. *Appl. Sci.* **2022**, *12*, 12525. https://doi.org/10.3390/app122412525

Academic Editor: Kiril Tenekedjiev

Received: 13 November 2022
Accepted: 2 December 2022
Published: 7 December 2022

1. Introduction

Most high school graduates are unsure of which university major to pursue once they complete their studies [1,2]. Determining the appropriate major for high school graduates is a challenging decision. Many aspects affect this decision, including a lack of experience at this age in making such important choices. In addition, an important factor is the lack of an in-depth understanding of the discipline to be studied. Students use internet searches and friends' recommendations in their decisions. Moreover, the socio-economic background of the family is a great influence [3]. Typically, people seek guidance and support from teachers, relatives, and coworkers. However, opinions are subjective and based on personal experience and often do not fully consider students' preferences [1]. It is important to choose a university major that matches the student's capabilities, such as the ability to remember and understand, visual and emotional intelligence, and physical abilities. The future of the student is typically determined and shaped by the specialization he/she chooses upon joining the university. Although it is crucial to identify students' passions and prepare them for their future careers, student specialty selection has not received much attention in educational research. The problem of choosing a university major is a global educational issue. For example, approximately 30% of first-year students in the

United States do not return after their first year due to the wrong major choice, costing taxpayers over USD 9 billion per year [3]. Statistical evidence has revealed that many students continue to fail in their university courses despite receiving family support and having good work ethic; this failure is linked failure to select a suitable faculty/major that is appropriate for their abilities and skills [4].

Choosing a university major is therefore important and affects students' future; current choice mechanisms are random and non-systematic and often ignore student preferences. It is well known that a person's desire for something is insufficient to propel him or her to success. Not everyone who wants to work in commerce, industry, or another field is successful [4]. Many have spent time and money pursuing fields that are incompatible with their skills and abilities. Thus, a smart recommendation system can be a useful tool to guide students in choosing their university majors in relation to qualifications, interests, capabilities, labor market needs, and employment rates. As a result of the influence of AI, we have seen incredible technological development in a short amount of time [5]; computers, robots, and other artifacts now possess human-like intelligence that is defined by cognitive skills, understanding, adaptability, and decision making, thanks to the field of research known as artificial intelligence [6]. There is a branch of artificial intelligence known as machine learning (ML); retraining existing models to improve performance is a common practice among developers, and this often incorporates machine learning. Linear data are ideal for machine learning. Machine learning performs well with less data but not with massive amounts of information. The model is trained using one of three primary machine-learning techniques. In order to learn from data, supervised machine learning requires the assistance of a human supervisor as well as the existing data. Without human oversight, unsupervised machine learning can take place. The use of machine learning with reinforcement is declining. These algorithms learn the best data from the past and use it to make correct decisions [7]. The current growth of artificial intelligence (AI) is a result of developments in machine learning. Rather than relying on extensive human programming, ML employs techniques that enable self-learning machines to explore data and complete tasks. So, we can apply the advantages of AI to help with decision making and to assess students' ability to choose their major and be confident about their choices. Recommendation systems (RSS) have evolved to help pupils determine what it is they want to study. A student's success in their field of study may be significantly enhanced by this method. Students' levels of knowledge, competencies, gender, job experience, and styles of learning all have a role in how RSS are used in education, based on the use of AI and ML techniques. AI is outlined as "the engineering and science of mimicking, extending, and enhancing human intelligence via artificial means and methods for producing intelligent machines [8]. However, we do not trust such important choices to an automated machine; rather, we consider algorithmic knowledge for solving specific types of problems. For example, they can monitor real-time systems, write life insurance, and perform a variety of other tasks that previously required human expertise [4,9]. In addition, such systems are extremely useful to students when deciding on a university specialization because they analyze students' personality and abilities while also introducing them to market demand [10].

This system is a resource to aid the students in making intelligent choices for themselves. Learning success, specialized training, improved student performance, and self-evaluation were all significantly aided by the incorporation of expert systems into educational advice [11].

In this research work, we aim to introduce an intelligent recommendation system to assist students in selecting the most appropriate university major based on prior knowledge and information, including students' past performance, labor market data, student marks, student behavior, expected salary after graduation, student experiences, and the applicant's gender. To achieve this, different ML algorithms were used and investigated, including the decision tree classifier (DTC), support vector machine (SVM), and random forest (RF) classifiers. The main contributions of this paper are twofold: (1) finding the best ML classifier that produces the most accurate prediction of the student major selection using the above-mentioned features, and (2) identifying the most significant features in predicting student majors.

The rest of this paper is organized as follows. Section 2 discusses the related work. The Materials and Methods are explained in Section 3. Section 4 presents the obtained classification results, followed by a comprehensive discussion. Finally, the findings and future work are summarized in Section 5.

2. Literature Review

Many research studies have been conducted support students in their decision to select the appropriate major at university [3]. This is achieved by introducing recommendations and decision support systems based on different supervised ML techniques and based on student data, such as academic history, absences, and tendencies. Some research has used the K-Nearest-Neighbor (KNN) algorithm as the highest accuracy algorithm for this classification problem [12]. In particular, the authors in [13] developed the King Abdelaziz University Recommendation System (KAURS), which is a recommendation system to predict and suggest a suitable major for students based on their abilities and marks in their preparatory year. In this study, the KNN algorithm was used to predict the appropriate major. The validation for the system was performed using the k-fold cross-validation, which resulted in 74.79% accuracy. In addition, the researchers in [12] proposed a recommendation system that aims to improve student outcomes by suggesting a number of appropriate majors (n) utilizing the KNN approach; the researchers measured the percentage of students who had their major as the n recommended major based on students with similar courses and performance using adjusted cosine distance. However, this could not determine whether the major was suitable for the student; to confirm this, another measurement was used to check if the student's performance was at or above the average performance in this specialty. The system obtained an accuracy of 67%. Another recommendation system was introduced by the authors in [14] using KNN to recommend the branch, followed by collaborative filtering for recommending the college, based on the student's score.

The Naïve Bayes (NB) classifier has been adopted in many recommendations systems. For instance, the authors in [15] tested a number of classifiers; their model mainly relied on additional data along with the student information, such as the number of absences, to determine the students' orientation. After evaluating the classifiers on this data, the NB obtained the best classification result, reaching an accuracy of 92.1% compared to 90.37% for the Neural Networks (NNs) and 88.13% by the SVM. Similarly, Naïve Bayes had a higher accuracy than SVM in [16], at 93%.

Artificial Neural Network (ANN) has been used in this research field. The study [17] evaluated Artificial Neural Network (ANN), Decision Tree Classifier (DTC), Support Vector Machine (SVM), and Naive Bayes using the accuracy, F1-Measure, precision, and recall metrics. The results showed that the ANN algorithm outperformed the other algorithms, with an accuracy of 79%.

Another algorithm that has achieved relatively promising classification rates in major prediction is the RF algorithm. As part of the binary classification, the researchers in [18] implemented an ML model to predict student paths using Logistic Regression (LR), Random Forest, and Decision Tree (DT), where the LR was used to predict students' major paths as a binary outcome for the main two majors. RF and DT were used to categorize students based on study path, demographics, orientation, and goals. The results showed that RF had the highest accuracy, at 94%. The authors in [19] implemented a college major recommendation system. SVM, NB, DTC, Gradient Boosting Decision Tree (GBDT), RF, Convolutional Neural Network (CNN), and Recurrent Neural Network (RNN) as well as collaborative filtering (CF) were trained on the collected data, and RF was able to achieve the highest accuracy of 97%. The authors in [3] introduced a system to predict the undergraduate specialization for students based on academic history and market needs. A few ML algorithms, including

DTC, extra tree classifier (ETC), RF, gradient boosting classifier (GBC), and SVM were trained. This system was able to achieve the highest accuracy of 75% using RF. The researchers in [20] introduced an adaptive system to recommend a suitable engineering department, based on the preparatory year grades and the final grades upon completing the program, by training multiple ML such as SVM, linear regression (LR), and RF. The RF reached an accuracy of 82.57%. Other researchers, such as in [10], introduced an expert system to assist high school students in selecting their university program using the DTC algorithm, which simplifies the decision-making process by breaking it down into a series of simpler decisions, making the solution easy to understand. This system was able to achieve an accuracy of 98%.

Some researchers follow a hybrid or hierarchal model that combines two or more algorithms. The study [21] used a hybrid model that combined the multi-class SVM and KNN algorithms, where the SVM classified the graduate schools that are likely interesting to a candidate, and the KNN algorithm classified universities and colleges, using the same skills and prerequisites. Similarly, [21] adopted a hybrid model that combined both Knowledge Base (KB) and Collaborative Filtering to help students in choosing their university, university majors, and job options; the CF was used to calculate student scores and generate recommendations based on similarities. Then, the outcome of the CF was input into the KB recommendation system to recommend personalized suggestions based on a student's demographic and academic history. The study [22] utilized data on academic results, personality, and intelligence to select the appropriate major using a hierarchal classification approach; the first classifier was responsible for predicting the main streams, and another classifier (for each stream) predicted the subcategory of the major. Each classifier in this hierarchal model was trained using two classification algorithms, Random Forest and Multi-Layer Perceptron, in addition to 10-fold cross-validation to confirm the classification accuracy, which ranged from 89.29% to 96.10% using the RF and confirmed that the hierarchal model outperformed the flat one. Anther hierarchal model was implemented in [23] using Multi-level SVM to categorize a graduate school; a KNN algorithm was used to generate graduate programs with comparable prerequisites and qualifications, with an accuracy of 58%.

Deep Learning Algorithms have also been used in this field. For instance, in [24], the Deep Neural Network algorithm along with five other ML classification algorithms, including LR, SVM, KNN, RF, DT, and Gaussian Naive Bayes, were used for university admission systems. The results show that the Deep Neural Network algorithm was the best, with an accuracy of 95.1%. Other researchers [25] have compared the Neural Network against Nearest Neighbor and Decision Tree; the former was able to achieve 71.30%.

Despite the existence of these related research works, there remain several challenges. For instance, in some research [15], the major prediction is decided based on the student's score at high school, without considering other factors such as the job market, academic history, etc. Many existing studies achieved only a low classification rate, e.g., [1,12,13,26]. Other research works have achieved high classification results; however, they depend on a complex workflow of preprocessing the data. For instance, the authors in [14] initially prepared the data by applying a certain filtration, discretization process, and binarization for the features and even applied data augmentation techniques to increase the size of the data [21]. Furthermore, some researchers have used sophisticated classification algorithms to improve the accuracy rate. For example, in [19], a hybrid model was built, in [15,18,23] a hierarchical model was adopted, and in [17] the implemented system was based on using big data technology concepts such as Hadoop and MapReduce. Finally, not all researchers consider the hyper-parameter optimization for the ML, for instance, [3]. In this research, the proposed model aimed to solve these problems by considering both the student's academic history and data from the job market (e.g., the student's grades in high school and the expected salary of a student). A number of ML algorithms and hyperparameter optimization were used with this data to obtain the best performance.

3. Materials and Methods

Generally, developing a recommendation system for choosing the most appropriate major at universities involves several main steps, as shown in Figure 1, including data gathering, data preprocessing and visualization, and machine learning processes.

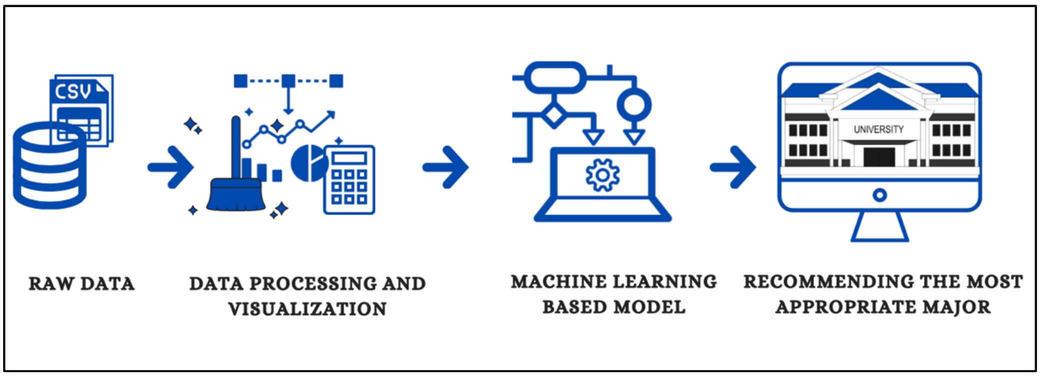

CSV
RAW DATA
DATA PROCESSING AND VISUALIZATION
MACHINE LEARNING BASED MODEL
UNIVERSITY
RECOMMENDING THE MOST APPROPRIATE MAJOR

Figure 1. The workflow of a machine learning–based model for major recommendation systems at higher education institutes.

3.1. Data Collection and Preparation

The data used in this research work are public data that were published on Kaggle [27] and were collected through a study of MBA students at CMS Business School in January 2020. This dataset includes information related to academic history and the job market. The data mainly consist of 216 sample students with 13 input features and the specialization as the target feature. The target feature has two fields, namely the marketing and finance field and the market and human resources field. The first major was labeled as 1, and the latter as 0. Table 1 below specifies the features included in the dataset. The data include the features that are related to the job market, such as work experience, employment status, and salary after graduation.

Table 1. Dataset description.

Feature Name	Description	Data Type
Gender	Student gender	Categorical
Serial Number	Student serial number	Numerical
SSC_P	Secondary school percentage	Numerical
SSC_b	Secondary school board studied (Class 10)	Categorical
HSC_P	Higher secondary school percentage (Class 12)	Categorical
Hsb_p	Higher secondary school board studied (Class 12)	Categorical
Hsb_s	High secondary school (Class 12) specialization	Categorical
Degree_p	Degree percentage	Numerical
workex	Work experience	Categorical
Etest_p	Entry test result	Numerical
specialization	Specialization	Categorical
Mba_p	Student percentage in MBA	Numerical
Status	Student placement status	Categorical
Salary	Student salary	Numerical

This dataset was chosen for this research because it is publicly accessible and differs from other datasets used in this field of study in that it includes both academic performance of the student and labor market status for the chosen specialization, which implies employment and a good salary after graduation. This dataset was also used by the [3]; we conducted a comparison between our results and the outcomes of this study.

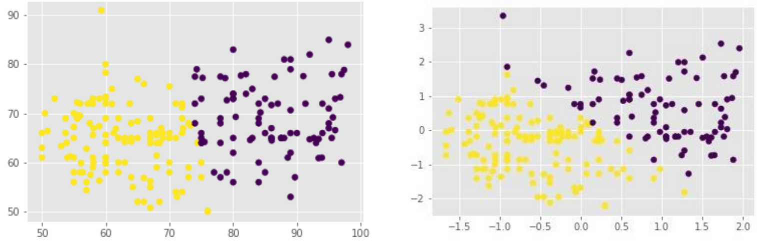
3.2. Data Processing and Visualization

In this stage, the data were cleaned and prepared for the visualization and learning processes. Using the Python module, we then applied multiple preprocessing mechanisms to clean and prepare the data for later tasks. First, we dealt with missing values; 67 missing values were found in this dataset. In particular, in the salary columns, in the case that the student had not been placed at a job after graduation, there was a null salary value; thus, this value was replaced with zero. Then Label Encoder was used to convert categorical feature labels into numeric values, which simplifies the use of ML techniques in later steps; a detailed description of the labeling process is shown in Table 2. As well, the data were normalized to scale all values between 0 and 1 using the MinMaxScaler, as the data contained some outliers (see Equation (1)). Figure 2 shows the data before and after normalization. Finally, the dataset was split into a training and testing dataset, with a ratio of 80:20, i.e., 173 samples and 43 samples for training and testing, respectively. For a fair comparison with the work in [3], the dataset was also split into a training and testing dataset, with a ratio of 70:30.

$$x' = \frac{x_i - \min(x)}{\max(x) - \min(x)} \tag{1}$$

Table 2. Label encoding conversions.

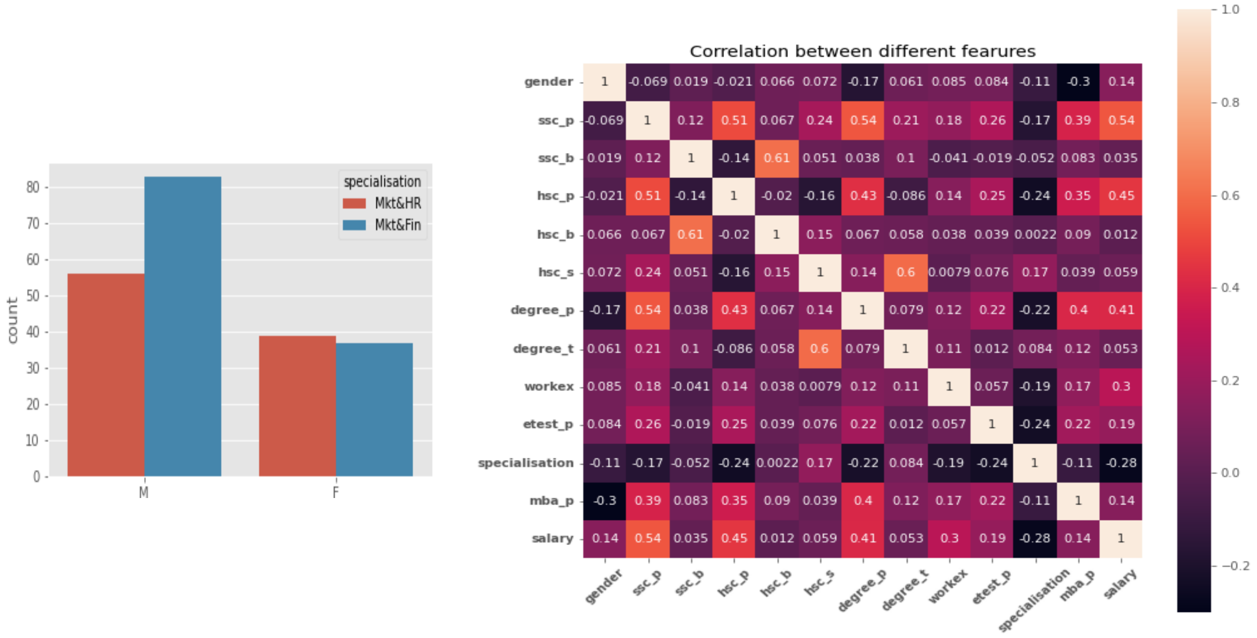
Feature Name	Feature Labels Map
gender	Female (0), Male (1)
ssc_b	Others (0), Central (1)
hsc_b	Others (0), Central (1)
hsc_s	Commerce (0), Science (1), Arts (2)
degree_t	Sci&Tech (0), Comm&Mgmt (1), Others(2)
workex	No (0), Yes (1)
specialization	Mkt&HR (0), Mkt&Fin (1)
status	Placed (0), Not Placed (1)



(**a**) (**b**)

Figure 2. The distribution of the data before and after normalization: (**a**) the distribution before normalization; (**b**) the distribution after normalization.

Data visualization provides a better and more timely understanding of the data; it helps determine the data quality, trends, and relationships, select the proper model, and decide how to proceed with the subject at hand. A bar plot is used to determine the distribution of students in each major by gender in Figure 3a. We can see the number of students (males and females) involved in Marketing and Finance or Marketing and Human Resources majors; the number of females is less than the number of males in the sample and approximately the same for both programs, while the males seem to be more interested in Marketing and Finance. Figure 3b presents the correlation matrix, which shows that there is no collinearity between the input variables, except the status and salary; thus, status column was removed. All the remaining input features can be used in building a machine learning model.



(**a**) (**b**)

Figure 3. Data visualizations: (**a**)the distribution of majors by gender (**b**) correlation matrix.

3.3. *Model Classification and Tuning*

We used three ML algorithms, DT, SVM, and RF, with hyper-parameter tuning to optimize the ML model parameters to achieve the best performance. The GridSearchCV technique was used for hyperparameter optimization; the hyperparameter is the parameter that is used to configure the algorithm or to determine the loss function minimization [28]. It picks up the parameters with the highest performance by searching for given values of the hyperparameters for the training algorithm that guarantee a significant improvement and a performance better than one selected randomly [29]. In addition, the GridSearchCV performs cross-validation during the training process. The data are divided into training data and testing data; the cross-validation divides the training data into k sets. Each time through the iteration, one partition is kept for testing and the other k-1 partitions are used to train the model. In the next iteration, the next partition is used as testing data and the last k-1 is used as training data, and so on. At each step, the model's performance is recorded. Finally, the average of all the results is given. In order to determine the predictive model's capacity to generalize and to avoid overfitting, cross-validation is one of the most popular data resampling techniques. The complete learning set is often subjected to the learning function to construct the final model for the forecasting of real future cases. It is not possible to cross-validate the final model. The goal of cross-validation during model construction is to predict how well the final model will perform when applied to fresh data [30].

3.3.1. Decision Tree (DT)

The DT algorithm is a tree-based supervised algorithm for both classification and regression. Each route from the root node to the leaf node is represented as a series of data separations, until a classification result is achieved at the leaf. The splitting is mainly conducted based on the information gain, which is a measure of how much knowledge is obtained from the variable in the dataset [31]. DT hyperparameters include the criterion, which is a split quality measurement function; max_depth, which is the maximum depth for the tree; min_samples_leaf, which is the minimum sample number to decide a leaf; and min_samples_split, which is minimum sample number to decide a split [32].

3.3.2. Support Vector Machine (SVM)

SVM is one of the supervised learning techniques for classification, regression, and outlier detection. SVM is a classifier that works by creating a hyperplane or multiple hyperplanes for separation, which implies giving the training data labels, based on the optimal hyperplane, that will categorize the new sample [33]. SVN's hyperparameters include the kernel, which has a default value of RBF and which was used to hyper-tune the C parameter, which is the regularization parameter, and Gamma, which is the kernel coefficient [34].

3.3.3. Random Forest (RF)

Random Forest is a supervised classifier that can be useful for regression and classification analysis. RF concept is based on constructing a set of decision trees on the training data and giving predictions based on a high accuracy tree and majority vote. RF provides a high classification rate and can handle both outliers and noise; it is also less exposed to overfitting [35]. The RF algorithm hyperparameters are criterion, which is the split quality measurement function; max_depth, which is the maximum depth for the tree; min_samples_leaf, which is the minimum sample number to decide a leaf; min_samples_split, which is the minimum sample number to decide a split; and n_estimators, which is the number of decision trees to be built on the RF.

After training, many performance metrics were used to evaluate the trained models, and the model with the best performance was chosen based on accuracy, the receiver operating characteristic (ROC), and the confusion matrix, which contains the false positive (FP), false negative (FN), true positive (TB), and true negative (TN). The true positive rate (TPR) and false positive rate (FPR) were based on the following equations:

$$\mathrm{TPR} = \mathrm{TP}/\mathrm{TP} + \mathrm{FN} \tag{2}$$

$$\mathrm{FPR} = \mathrm{FP}/\mathrm{FP} + \mathrm{TN} \tag{3}$$

$$\mathrm{Accuracy} = \mathrm{TP} + \mathrm{TN}/\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN} \tag{4}$$

4. Results

The first experiment in this research used the ML learning algorithm without any hyper-tuning for the hyperparameter. This was applied twice, using the 20:80 and 30:70 testing:training ratio. Table 3 shows the obtained results [3].

Table 3. Comparison of accuracy results between the related work [3] and this research without applying hyper-tuning.

ML Model	Related Work Results *	30:70 Testing:Training Ratio	20:80 Testing:Training Ratio
Decision Tree Classifier	55.38%	71.00%	74.00%
Support Vector Machine	52.31%	74.00%	79.00%
Random Forest Classifier	75.38%	77.00%	86.00%

* Related work from [3], where a 30:70 testing:training ratio was used with no hyper-tuning.

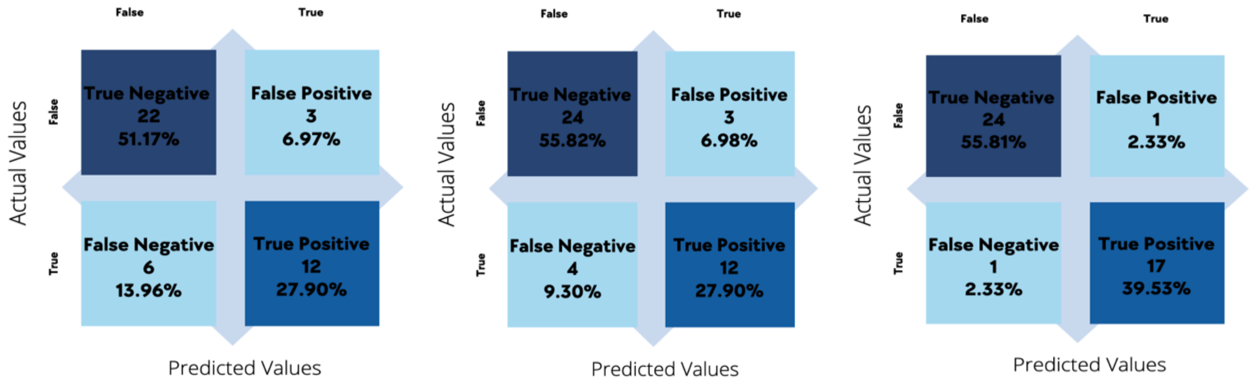
The DT hyperparameters were hyper-tuned, including the criterion, max_depth, min_samples_leaf, and min_samples_split. Table 4 shows the values used for each parameter and the combination of the hyperparameters that was selected as the best estimator. The decision tree–trained model achieved an accuracy of 79% using five-fold cross-validation. The confusion matrix, true positive rate, and false positive rate were calculated for the SVM model, as shown on Figure 4a and Table 5.

Table 4. Decision Tree best hyperparameter combination after hyper-tuning.

Hyperparameter Name	Hyperparameter Values	Hyperparameter Optimal Value
criterion	gini, entropy	entropy
max_depth	100, 200	100
min_samples_leaf	2, 3	2
min_samples_split	1, 2, 3, 4	3

Table 5. Accuracy, TPR, and FPR of machine learning model experiments.

ML Model	True Positive Rate	False Positive Rate	Accuracy
Decision Tree Classifier	66.67%	12.00%	79.00%
Support Vector Machine	75.00%	11.11%	84.00%
Random Forest Classifier	94.00%	4.00%	95.00%



(**a**) (**b**) (**c**)

Figure 4. Confusion matrix for ML algorithms. (**a**) DT confusion matrix; (**b**) SVM confusion matrix; (**c**) RF confusion matrix.

The hyper-tuning was applied to the SVM for the C and Gamma hyperparameters. Table 6 shows the hyperparameter values and the combination that was selected as the best estimator parameters that trained the model to give the best accuracy of 84.00% using five-fold cross-validation. Figure 4b and Table 5 shows the confusion matrix for the SVM model with TPR and FPR values.

Table 6. Support vector machine best hyperparameter combination after hyper-tuning.

Hyperparameter Name	Hyperparameter Values	Hyperparameter Optimal Value
C	0.001, 0.01, 0.1, 1, 10	10
Gamma	0.001, 0.01, 0.1, 1	0.1

The same hyperparameters applied to the RF, including the criterion, max_depth, min_samples_leaf, min_samples_split, and n_estimators. The parameter combination shown in Table 7 was selected as the best estimator and was used in training the algorithm. The algorithm with these parameters was able to outperform both the DT and SVM, with an accuracy of 95% using five-fold cross-validation. Figure 4c shows the confusion matrix for the RF with TPR and FPR.

Table 7. Random Forest best hyperparameter combination after hyper-tuning.

Hyperparameter Name	Hyperparameter Values	Hyperparameter Optimal Value
criterion	gini, entropy	gini
max_depth	100, 200, 300, 400, 500	100
min_samples_leaf	1, 3	1
min_samples_split	2, 3	3
n_estimators	100, 200, 250	200

In our experiment, we found that the accuracy and TPR of the RF were the highest, whereas the FPR was the lowest among the other alternatives, as shown in Figure 5. An essential method for measuring the efficacy of an ML model is the receiver operating characteristic curve (ROC). This is a 2D ROC curve representing the relation between FPR and TPR and showing the prediction ability of the generated model.

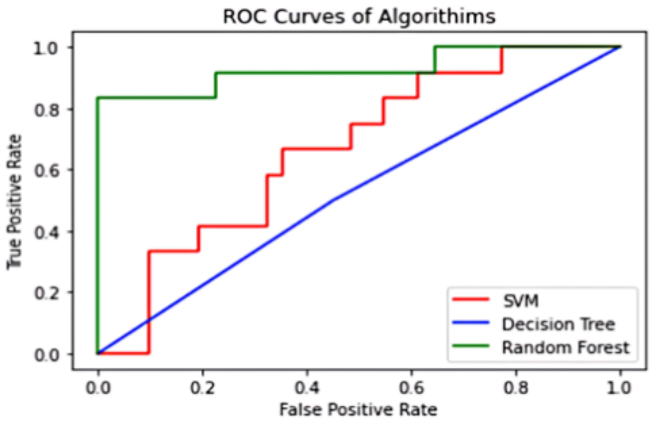


Figure 5. ROC curves for ML algorithms.

To examine the importance of input variables to the model, the variable importance plot, as shown in Figure 6, was used; it provides the most significant features in descending order based on a mean decrease in Gini. This result implies that the top variable has greater effect on the model results than the bottom one and has classification power. The plot shows that degree percentage, MBA percentage (mba_p), entry test result (etest_p), salary and student percentage, and field in secondary school (hsc_p, ssc_p) play important roles in predicting the appropriate major for students. The model can be built based on these features; removing any of these features will cause a drop in the accuracy. Predicting the student specialty is weakly influenced by other variables such as gender, work experience, etc., which shows that dropping these features may improve the model accuracy.

As part of testing the features' importance in the major selection, we dropped the features, starting from the smallest importance, and monitored the accuracy, which increased after dropping the variables gender, hsc_b, ssc_b, workex, and degree_t; when the hsc_s was dropped, the accuracy decreased again, which showed that this feature was important to the classification process, while the other dropped variables did not contribute greatly to the classification process. Table 8 demonstrates the results of this experiment.

Table 8. Results of accuracy of machine learning models after dropping low-importance features.

ML Model	Accuracy before Dropping	Accuracy after Dropping
Decision Tree Classifier	79.00%	83.70%
Support Vector Machine	84.00%	86.00%
Random Forest Classifier	95.00%	97.70%

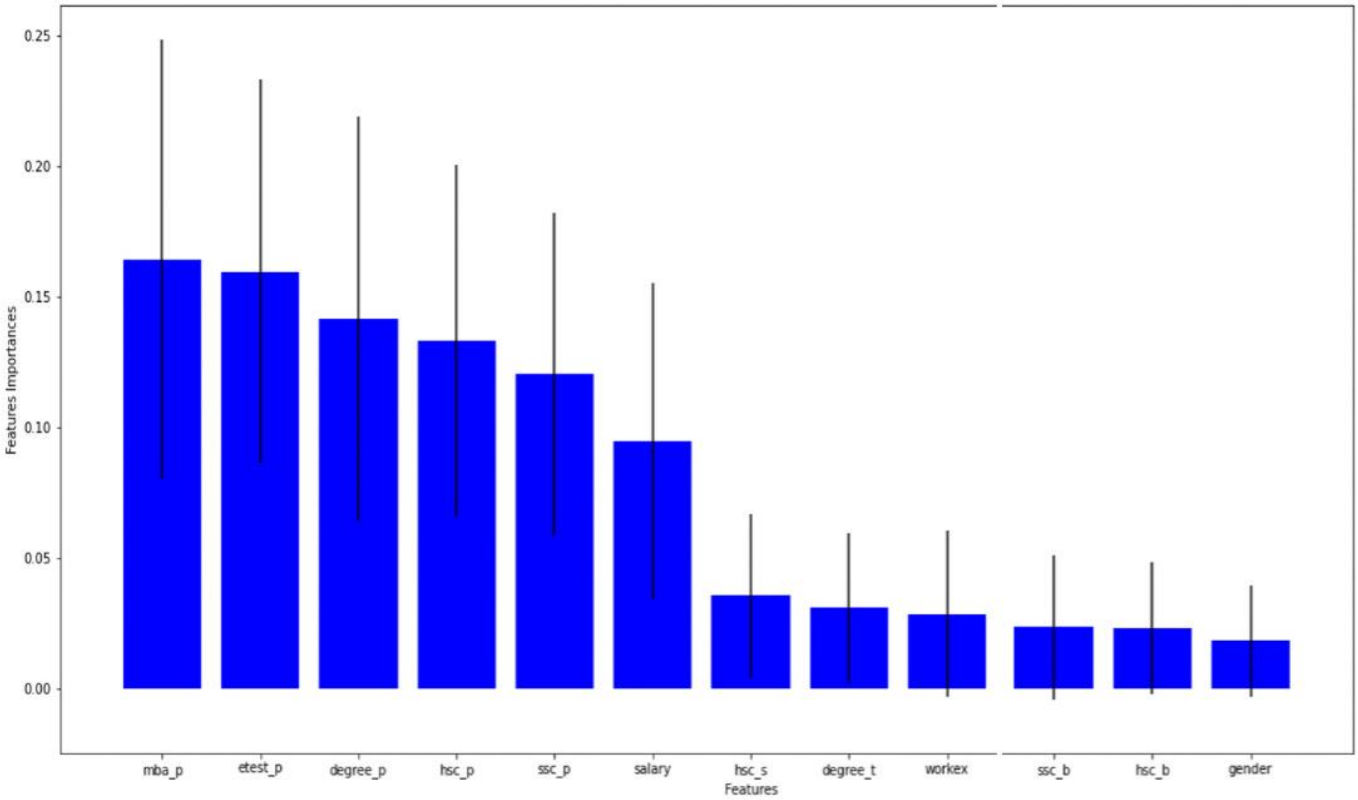


Figure 6. Input features importance chart.

5. Discussion

After applying the first experiment, without hyper-tuning on both data split ratios 30:70 and 20:80, the obtained result, as shown in Table 3, demonstrates that our model outperforms the work presented in [3], even without hyper-tuning and using the same testing:training ratio. This is related to the preprocessing performed before training the models, including the normalization and hot encoding, which transformed the data into a format suitable for ML algorithms, leading to higher performance. Moreover, increasing the training dataset size against the testing dataset size (80:20) had a positive impact on the classification rates, especially the RF algorithm, with an increase of 9%. This underlines that the model did not see all the samples when the 70:30 ratio was used, as some characteristics existed on the testing and not the training dataset. This implies the need to use k-fold cross-validation, where the dataset is split into K sets, and the test is applied K times, which can make the models more generalizable and guarantee that the model behavior is optimal. The result is comparable with the results in [3]; however, it can be improved by applying hyper-tuning for the hyperparameters.

For the next experiment, hyper-tuning was applied on all the ML models. Figure 4a and Table 5 show the confusion matrix for the DT model, which shows the classification rate, where the TPR = 66.67% and FPR = 12% with an accuracy of 79%. This means that the model was able to predict 66.67% the choice of the Marketing and Finance major correctly, while 12.00% of this major was classified erroneously, with the actual choice being Marketing and Human Resources. The SVM, as shown in Figure 4b and Table 5, shows the confusion matrix for the SVM model, with TPR = 75% and FPR = 11.11%. These values show that the SVM model classified 75% of the Marketing and Finance major correctly, while 11.11% was labeled incorrectly with this major. The RF algorithm with the selected parameters was able to outperform both the DT and SVM, with an accuracy of 95% using the five-fold cross-validation. Figure 4c shows the confusion matrix for the RF, with TPR = 94.90% and FPR = 4%. For the prediction of studying Marketing and Finance, 94.90% matched this major, and only 4% was labeled erroneously. Moreover, the constructed ROC in Figure 5 implies that the RF has the best performance in implementing our classifier, as its curve is confirmed closest to the upper left corner, which is the optimal point of TPR = 100.00% and FPR = 0.00%; RF had a very close value of TPR = 94.90% and FPR = 4.00%.

The final experiment was the feature importance test and the dropping of insignificant features. The results in Table 8 illustrate the accuracy after dropping the aforementioned features. It can be seen that the classification rate increases after dropping the low-importance features mentioned above from the dataset.

In this research, we used the same approach as in [3] as well as the same dataset. However, in our proposed model, we hyper-tuned the ML methods, which led to real improvements in the prediction rates, as shown in Table 9. As part of the evaluation of our machine learning models, a comparison was conducted with this article's outcome. The following table summarizes the accuracy of each model for the related work, our model without/with hyper-tuning using a 30:70 training:testing data ratio and without/with hyper-tuning using a 20:80 training:testing data ratio.

Table 9. Comparison of accuracy results between the related work [3] and this research.

ML Model	Related Work Results *	Without Hyper-Tuning on 30:70 Testing: Training Ratio	With Hyper-Tuning on a 30:70 Testing: Training Ratio	Without Hyper-Tuning On 20:80 Testing: Training Ratio	With Hyper-Tuning on 20:80 Testing: Training Ratio
Decision Tree Classifier	55.38%	71.00%	74.00%	74.00%	79.00%
Support Vector Machine	52.31%	74.00%	80.00%	79.00%	84.00%
Random Forest Classifier	75.38%	77.00%	92.30%	86.00%	95.00%

* The work conducted in ref. [11], where a 30:70 testing:training ratio was used with no hyper-tuning.

The results in Table 9 demonstrate that our model outperforms the work presented in [3]; these results confirmed the significance of the correct preprocessing of the data, increasing the training data set size, the use of cross-validation, which helps to make the model more generalizable and perform better on the unseen data, and the hyper-tuning of the machine learning algorithms, which finds the optimal combination of hyperparameters, leading to an improvement in the classification rate and a more accurate model.

6. Conclusions

This research proposed an enhanced intelligent recommendation system, as compared to the published work in [3], to predict the appropriate undergraduate specialty based on data that include both student academic history and the job market status. In this study, we trained a set of machine learning algorithms to develop a recommendation system for predicting suitable university majors. To improve the performance, ML hyper-tuning was applied, which had a significant role in increasing the performance of the ML algorithms and helped to introduce a much more accurate system compared to the work in [3]. In addition, the study focused on the importance of input features, which had a large effect on simplifying the overall classification process by reducing the number of input features, consequently leading to a better result. The resulting findings confirmed the importance of hyper-tuning and of input features in such a high-accuracy model; the RF algorithm was the best, with an accuracy of 97.70% compared to an accuracy of 75.00% in the previously published work. The model was able to achieve this accuracy considering both the historical academic data and the job market and without adopting an overly complex model structure. Based on these results, we can conclude a few points and recommendations that should be taken into consideration in implementing a machine learning model. (1) Hyperparameter tuning is the main step in ML modeling, minimizing the error and ensuring predictions are as close as possible to actual values. (2) The data size in general and the training size more specifically have a great effect in making the model more accurate. (3) Using cross-validation is important, as it helps to assess whether a model is accurate in a real-world environment with new and dynamic data. (4) Data visualization plays an important role in exploratory data analysis before applying machine learning models. (5) The significance of features is an integral component of model development. It identifies which features passing into a model have a greater impact on prediction generation than others. The results of identifying significant characteristics can directly inform model testing and explain ability. It has recently been shown that deep learning approaches have played an important role in improving the linear separability through constructing the apropos feature set [36]. Thus, investigating these new approaches on our classification problem would be worth to be considered in the future. In addition, other recommendation systems could be explored, including those that can predict whether the student will gain employment and the likely salary after graduation, using the same data.

Author Contributions: Conceptualization, Y.Z. and Y.S.; Methodology, Y.Z. and Y.S.; Software, Y.Z. and Y.S.; Validation, Y.Z. and Y.S; Formal Analysis, Y.Z. and Y.S.; Investigation, Y.Z. and Y.S.; Resources, A.H; Data Curation, Y.Z., and Y.S.; Writing—Original Draft Preparation, Y.Z. and Y.S; Writing—Review and Editing, Y.Z., Y.S. and A.H; Visualization, Y.Z. and Y.S.; Supervision, A.H.; Project Administration, A.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Data were retrieved from an open-source site (Kaggle).

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this study is available online (https://www.kaggle.com/benroshan/factors-affecting-campus-placement accessed on 24 June 2022) for research purposes. No personal information is included.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alghamdi, S.; Alzhrani, N.; Algethami, H. Fuzzy-based recommendation system for university major selection. In Proceedings of the 11th International Joint Conference on Computational Intelligence (IJCCI 2019), Dhaka, Bangladesh, 25–26 October 2019; pp. 317–324.
2. Ayman, M.; Ahmar, A. *A Prototype Rule-based Expert System with an Object-Oriented Database for University Undergraduate Major Selection*; International Journal of Applied Information Systems (IJAIS): New York, NY, USA, 2012.
3. Alsayed, A.O.; Rahim, M.S.M.; AlBidewi, I.; Hussain, M.; Jabeen, S.H.; Alromema, N.; Hussain, S.; Jibril, M.L. Selection of the right undergraduate major by students using supervised learning techniques. *Appl. Sci.* **2021**, *11*, 10639. [CrossRef]
4. Naser, S.S.A.; Baraka, M.H.; Baraka, A. A Proposed Expert System for Guiding Freshman Students in Selecting a Major in Al-Azhar University, Gaza. Available online: http://dstore.alazhar.edu.ps/xmlui/handle/123456789/387 (accessed on 12 November 2022).
5. Nadikattu, R. The Supremacy of Artificial intelligence and Neural Networks. *Int. J. Creat. Res. Thoughts* **2017**, *5*. Available online: https://ssrn.com/abstract=3655849 (accessed on 12 November 2022).
6. Chen, L.; Chen, P.; Lin, Z. Artificial Intelligence in Education: A Review. *IEEE Access* **2020**, *8*, 75264–75278. [CrossRef]
7. Priyanka; Sanjeev, K. A review paper on breast cancer detection using deep learning. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1022*, 012071. [CrossRef]
8. Zhongzhi, S. Advanced Artificial Intelligence. Available online: https://books.google.ps/books?id=wNbMOoTuGU0C (accessed on 29 June 2022).
9. Baral, C. Knowledge Representation, Reasoning and Declarative Problem Solving with Answer Sets 1. Cambridge University Press: Cambridge, UK, 2003.
10. Alnajjar, A.; Hasasneh, N.; Macido, M. A Novel Expert System to Assist High School Students in Selecting Their Appropriate University Program: A Case Study of Hebron University. 2021. Available online: https://digitalcommons.aaru.edu.jo/hujr_a/vol10/iss1/3 (accessed on 12 November 2022).
11. Supriyanto, G.; Widiaty, I.; Abdullah, A.G.; Yustiana, Y.R. Application expert system career guidance for students. *J. Phys. Conf. Ser.* **2019**, *1402*, 066031. [CrossRef]
12. Stein, S.A.; Weiss, G.M.; Chen, Y.; Leeds, D.D. A College Major Recommendation System. In Proceedings of the 14th ACM Conference on Recommender Systems, Virtual, 22–26 September 2020; pp. 640–644.

13. Alshaikh, K.; Bahurmuz, N.; Torabah, O.; Alzahrani, S.; Alshingiti, Z.; Meccawy, M. Using Recommender Systems for Matching Students with Suitable Specialization: An Exploratory Study at King Abdulaziz University. *Int. J. Emerg. Technol. Learn.* **2021**, *16*, 316–324. [CrossRef]
14. Roshan, M.; Bhanuse, S.; Yenurkar, G. Recommendation of Branch of Engineering using machine learning. *Int. Res. J. Eng. Technol.* **2020**.
15. Ouatik, F.; Erritali, M.; Ouatik, F.; Jourhmane, M. Students' Orientation Using Machine Learning and Big Data. *Int. J. Online Biomed. Eng.* **2021**, *17*, 111–119. [CrossRef]
16. Mostafa, L.; Beshir, S. University Selection Model Using Machine Learning Techniques. In *The International Conference on Artificial Intelligence and Computer Vision*; Springer: Cham, Switzerland, 2021; pp. 680–688.
17. Mengash, H.A. Using data mining techniques to predict student performance to support decision making in university admission systems. *IEEE Access* **2020**, *8*, 55462–55470. [CrossRef]
18. Dirin, A.; Saballe, C.A. Machine Learning Models to Predict Students' Study Path Selection. *Int. J. Interact. Mob. Technol.* **2022**, *16*, 158–183. [CrossRef]
19. Meng, Y.; Fun, M. *CMRS: Towards Intelligent Recommendation for Choosing College Majors*; ACM: New York, NY, USA, 2020; Volume 6.
20. Ezz, M.; Elshenawy, A. *Adaptive Recommendation System Using Machine Learning Algorithms for Predicting Student's Best Academic Program*; Springer Science Business Media: Berlin, Germany, 2020; Volume 25, pp. 2733–2746.
21. Baskota, A.; Ng, Y.K. A graduate school recommendation system using the multi-class support vector machine and KNN approaches. In Proceedings of the 2018 IEEE 19th International Conference on Information Reuse and Integration for Data Science, Salt Lake City, UT, USA, 6–9 July 2018; pp. 277–284.
22. Obeid, C.; Lahoud, C.; El Khoury, H.; Champin, P.A. A Novel Hybrid Recommender System Approach for Student Academic Advising Named COHRS, Supported by Case-based Reasoning and Ontology. *Comput. Sci. Inf. Syst.* **2022**, *19*, 979–1005. [CrossRef]
23. Kamal, N.; Sarker, F.; Mamun, K.A. A Comparative Study of Machine Learning Approaches for Recommending University Faculty. In Proceedings of the 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, 19–20 December 2020.
24. El Guabassi, I.; Bousalem, Z.; Marah, R.; Qazdar, A. A Recommender System for Predicting Students' Admission to a Graduate Program using Machine Learning Algorithms. *Int. J. Online Biomed. Eng.* **2021**, *17*, 135–147. [CrossRef]
25. Balaji, P.; Alelyani, S.; Qahmash, A.; Mohana, M. Contributions of machine learning models towards student academic performance prediction: A systematic review. *Appl. Sci.* **2021**, *11*, 10007. [CrossRef]
26. Sethi, K.; Jaiswal, V.; Ansari, M.D. Machine Learning Based Support System for Students to Select Stream. *Recent Adv. Comput. Sci. Commun.* **2020**, *13*, 336–344. [CrossRef]
27. Placement_Data_Full_Class. csv. Available online: https://www.kaggle.com/datasets/benroshan/factors-affecting-campus-placement (accessed on 16 June 2022).
28. Charbuty, B.; Abdulazeez, A. Classification Based on Decision Tree Algorithm for Machine Learning. *J. Appl. Sci. Technol. Trends* **2021**, *2*, 20–28. [CrossRef]
29. Deris, A.M.; Zain, A.M.; Sallehuddin, R. Overview of support vector machine in modeling machining performances. *Procedia Eng.* **2011**, *24*, 308–312. [CrossRef]
30. Ahmad, I.; Basheri, M.; Iqbal, M.J.; Rahim, A. Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection. *IEEE Access* **2018**, *6*, 33789–33795. [CrossRef]
31. Yang, L.; Shami, A. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing* **2020**, *415*, 295–316. [CrossRef]
32. Liashchynskyi, P.; Liashchynskyi, P. Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS. *arXiv* **2019**, arXiv:1912.06059.
33. Berrar, D. *Cross-Validation. Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*; Elsevier: Amsterdam, The Netherlands, 2018; Volume 1–3, pp. 542–545.
34. Scikit-Learn 1.1.1 Documentation—Decision Tree Classifier. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html?highlight=decision%20tree#sklearn.tree.DecisionTreeClassifier (accessed on 22 July 2022).
35. Scikit-Learn 1.1.1 Documentation—Support Vector Classification. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html (accessed on 22 July 2022).
36. Hasasneh, A.; Kampel, N.; Sripad, P.; Shah, N.J.; Dammers, J. Deep learning approach for automatic classification of ocular and cardiac artifacts in meg data. *J. Eng.* **2018**, *2018*, 1350692. [CrossRef]