electronics

MDPI

Article

Study on Using Machine Learning-Driven Classification for Analysis of the Disparities between Categorized Learning Outcomes

Aleksandra Kowalska [1,*], Robert Banasiak [1], Jacek Stańdo [2], Magdalena Wróbel-Lachowska [1], Adrianna Kozłowska [3] and Andrzej Romanowski [1]

[1] Institute of Applied Computer Science, Lodz University of Technology, 90-537 Lodz, Poland
[2] Centre of Mathematics and Physics, Lodz University of Technology, 90-924 Lodz, Poland
[3] Centre for Teaching and Learning, Lodz University of Technology, 90-924 Lodz, Poland
* Correspondence: akowalska@iis.p.lodz.pl

Abstract: Learning outcomes are measurable statements that articulate educational aims in terms of what knowledge, skills, and other competences students possess after successfully completing a given learning experience. This paper presents an analysis of the disparity between the claimed and formulated learning outcomes categorized in knowledge, skills, and social responsibility competency classes as it is postulated in the European Qualification Framework. We employed machine learning classification algorithms to detect and reveal main errors in their formulation that result in incorrect classification using generally available syllabus data from 22 universities. The proposed method was employed in two stages: preprocessing (creating a Python dataframe structure) and classification (by performing tokenization with the term frequency–inverse document frequency method). The obtained results demonstrated high effectiveness in correct classification for a number of machine learning algorithms. The obtained sensitivity and specificity reached 0.8 for most cases with acceptable positive predictive values for social responsibility competency classes and relatively high negative predictive values greater than 0.8 for all classes. Hence, the presented methodology and results may be a prelude to conducting further studies associated with identifying learning outcomes.

Keywords: learning outcomes; higher education; machine learning; classification; TFIDF

Citation: Kowalska, A.; Banasiak, R.; Stańdo, J.; Wróbel-Lachowska, M.; Kozłowska, A.; Romanowski, A. Study on Using Machine Learning-Driven Classification for Analysis of the Disparities between Categorized Learning Outcomes. *Electronics* **2022**, *11*, 3652. https://doi.org/10.3390/electronics11223652

Academic Editor: Alberto Fernandez Hilario

Received: 31 August 2022
Accepted: 26 October 2022
Published: 8 November 2022

1. Introduction

Over the past 20 years, the process of reconstructing European higher education has brought fundamental changes in the approach to learning process designs. Study programs started to focus not only on graduates' preparation for volatile labor market requirements but also on maximizing their potential impact on democratic society. Moreover, emphasizing what the student should learn rather than what the educator should teach has become one of the biggest challenges for academics, as they should effectively support students in 'acquiring knowledge, skills and competences that best meet their self-development goals and social needs' [1].

Therefore, **learning outcomes** (LO), understood as what the learner should know, understand, and be able to do as a result of the learning process [2], have become a key concept in contemporary education. Additionally, as learning outcomes are always predefined in terms of what is expected at the very end of the learning experience, the adjective 'intended' for learning outcomes has been coined for greater precision [3,4].

The Bologna process introduced the idea of designing learning outcomes across three domains: **knowledge, abilities** (also called skills), and **attitudes** (understood as values or dispositions) [5]. In 2008, upon recommendations for the establishment of the European Qualifications Framework for lifelong learning, similar classifications were used: **knowl-**

edge, skills, and competences [6]. In 2017, the European Commission replaced the word 'competences' with **autonomy and responsibility** [2].

Knowledge is understood as a set of interrelated facts, principles, theories, and experiences that are acquired by the learner. **Skills** are the abilities to apply knowledge, perform tasks, and solve problems. **Competences** are the abilities to autonomously and responsibly perform assigned tasks, demonstrating readiness for lifelong learning, communication skills, and the abilities to interact with others, both as a member and leader of a team. Learning outcomes refer to what the student has achieved, not just the content of what was taught.

The division proposed by the European Commission refers to Bloom's taxonomy of educational objectives, which also identifies three domains: cognitive (knowledge-based), psychomotor (action-based) and affective (emotion-based) [7]. However, this is not a direct reference, and the indicated domains (by EC and Bloom) do not completely overlap. The 'skills' domain can include learning outcomes connected with knowledge but also psychomotor abilities.

Regardless of the terminology used, learning outcomes are the basis for describing programs and courses; moreover, they influence teaching content and methods, as well as educational environments and assessment practices. It is therefore essential that the construction of a curriculum should be based on properly and reasonably formulated, accessible and transparent learning outcomes [2]. They should be defined with appropriate verbs relating directly to the three learning domains, which in turn will be easy for students to understand and for teachers to assess, which will allow for designing student-centered curricula [8].

Despite two decades of practicing the new approach, the proper definition of learning outcomes is not an easy task for academics designing educational paths. As the 'The European Higher Education Area in 2018: Bologna Process Implementation Report' revealed, even though a number of higher education institutions in EU participated in the EUA Trends 2018 survey to design learning-outcomes-based curricula with greater confidence, one fifth still face problems with the formulation of their intended learning outcomes. Furthermore, only a quarter provided systematic training on defining learning outcomes to all teachers in all programs, one-eighth of the institutions questioned provided training in a systematic way for new teachers only, and 5 % provided training for new courses or programs only [1].

The greatest challenge in defining learning outcomes lies in the proper, unambiguous use of verbs, allowing for the operationalization of LOs and their measurement. However, there are verbs typical of LOs relating to skills that are also used with LOs relating to knowledge. Using the same verbs to describe learning outcomes related to knowledge and skills can create difficulties in classification.

The aim of this study was to investigate the distribution of learning outcomes used in the description of study programs, with reference to the three learning domains, i.e., knowledge, skills, and responsibility and autonomy, as well as maintaining consistency in the correct use of action verbs associated with each domain by employing a machine learning recognition system.

2. Related Work

The semi-automated [9] or automatic classification of learning outcomes has already been the subject of scientific research. However, the basis for such analyses has been the cognitive domain of Bloom's taxonomy. There are two main approaches to this task. The first is a keyword-based analysis [10–12], and the second is based on text classification [13–15]. Automatic references to categories from Bloom's taxonomy using an keyword-based approach had low accuracy [16] but also posed problems due to the overlapping of keywords [17]. In text-classification-based approaches, different machine learning- and deep learning-based models were used: CNN, LSTM, K-NN, logistic regression, SVM, ANN, and NLP.

There was also an attempt to create a voting algorithm that combines classifiers, and the combination of SVM, NB, and K-NN appeared as the best option [18].The first efforts to employ a simple deep learning model based on LSTM to classify LOs and assess question items according to Blooms taxonomy brought a classification accuracy of 87% for learning outcomes and 74% for assessment question items [17].

3. Experimental Method & Setup

3.1. Dataset Origin

The need to compare and recognize qualifications across Europe facilitated the publication of outcome-based study program descriptions. Our experimental setup is based on the dataset that includes 1548 learning outcomes classified in three domains (W—Knowledge, U—Skills, and K—Responsibility and Autonomy (Competences)), gathered from 22 university databases/websites in a European Union country. All the LOs in the dataset were translated from the native language to English before further processing.

The experiments were conducted in two stages: preprocessing and classification. The preprocessing phase required the preparation of the dataset by filtering unnecessary data (non-word values), creating a Python dataframe structure. Then, tokenization was performed to replace words with numbers that can be understandable for AI classifiers. In this work, a TFIDF (term frequency–inverse document frequency) method was used to tokenize learning outcomes [19].

3.2. Data Preprocessing and Tokenization

The TFIDF algorithm is a widely used technique in information retrieval and text mining tasks [20] as it helps to quantify words in a set of documents. In this method, a score is computed for each word to signify its importance in the document and corpus.

The term frequency component measures the frequency of a given word in a document. This depends on the length of the document and the generality of the word; for example, a very common word such as 'was' can appear multiple times in a document. If we take two documents with 100 words and 10,000 words, respectively, there is a high probability that the common word 'was' is more frequent in the 10,000-word document. Nevertheless, we cannot say that the longer document is more important than the shorter document. For this reason, the *TF* component performs normalization on the frequency value and divides the frequency of a given word's occurrence by the total number of words in the document. In this work, the *TF* component was used for word vectorization. *TF*-component processing is specific to each document and word; hence, we can formulate *TF* as follows:

$$TF(\mathrm{w}, \mathrm{d}) = \text{count of w in d / number of words in d} \quad (1)$$

To compute *TF*, two data feeds are required: the word count of all words and the length of the document. In the case the given word w does not exist in the studied document d, that particular *TF* value will amount to 0 for this document d. In an extreme case, if all the words in the document are the same, then *TF* will be equal to 1. The final value of the normalized *TF* value will fall in the range of [0 to 1]. The second TFIDF component—document frequency (*IDF*)—measures the importance of documents in a whole set of the corpus. This is very similar to the *TF* component with a small difference lying in the fact that *TF* is the frequency counter for a word w in document d, whereas *DF* counts occurrences of a Word w in the document set N. In other words, *DF* is the number of documents d in which the word w is present. Only a single occurrence is considered, i.e., if the word w is present in the document at least once, we do not need to know the number of times the word w is repeated. The *DF* can be formulated as follows:

$$DF(\mathrm{w}) = \text{occurence of w in N documents} \quad (2)$$

To also keep *DF* value in a range, it needs to be normalized by dividing by the total number of documents. The main goal is to know the informativeness of a word; therefore,

the inverse of *DF* (inverse *DF*) needs to be computed. *IDF* is the inverse of the document frequency that measures the informativeness of a word w. The *IDF* value will be very low for the most frequent words, such as stop words (because they are present in almost all documents, and *N/DF* will give a very low value to that word). This finally gives the relative weight of the given word w.

$$IDF(\mathrm{w}) = N/DF(w) \tag{3}$$

There are a few problems that might arise in *IDF* computations, especially when a large corpus ex. N = 10,000 is processed. In this case, the *IDF* value can be enormously high. To alleviate this effect, the log of *IDF* can be taken as a value. Another difficulty appears when the *DF* is 0 (a few words of the vocabulary might be absent in the document). As we cannot divide by 0, the value can be smoothed by adding 1 to the denominator.

$$IDF(\mathrm{w}) = log[N/DF(w) + 1] \tag{4}$$

Finally, by taking a multiplicative value of *TF* and *IDF*, the *TF-IDF* score is obtained for a given word w.

$$TFIDF(\mathrm{w}, \mathrm{d}) = TF(w, d) * log[N/DF(w) + 1] \tag{5}$$

where: $TF(w, d)$ is the number of occurrences of a word w in a document d. $DF(w)$ is the number of documents containing the word w. N is the total number of documents in the corpus.

In this work, a TFIDF vectorization was performed using TfidfVectorizer class from Scikit-learn Python Toolbox with hyperparameter $max_features$ set to 200. The next step of the experiment was to decide how many learning outcomes will be assigned to a training set and a test set. The research team agreed to an arbitrary breakdown: 40% as a test set and 60% as a training sample.

3.3. Proposed Classification Methods

To perform machine learning-type classification logistic regression (LR), decision tree (DT), random forest (RF), XGBoost (XGB), and hybrid voting classifier (HVC) methods were used. Logistic regression is a simple and efficient procedure for binary and linear classifications of problems. It includes a classification model, which achieves very good performance with linearly separable classes. It is an extensively employed algorithm for classifications in industry. The logistic regression model can also be used as a statistical method for binary classifications that can be generalized to multi-class classifications. In this study, Scikit-learn was applied as it has a highly optimized version of logistic regression implementation that supports multi-class classification tasks that fit well to three-class-based LOs [21].

Decision trees are a non-parametric supervised machine learning method used for establishing classification systems based on multiple covariates or for developing prediction algorithms for a target variable [22]. This technique involves the creation of a model that is able to predict the value of a target variable by simple decision-rules learning inferred from data features. It classifies a set of data into branch-like segments that construct an inverted tree with a root node, internal nodes, and leaf nodes. The algorithm is a non-parametric type and can work with large, complicated datasets without imposing a complicated parametric structure. When the sample size is large enough, classified data can be split into training and test samples. The former sample can be utilized to build a decision tree model while the latter supports the decision on the appropriate tree size needed to achieve the optimal final model.

Random forests are a hybrid of three different predictors, where each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest [22]. The generalization error for forests converges to a limit when the

number of trees in the forest increases. The generalization error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. Typically, a random selection of features used to split each node is more robust with respect to noise. Internal estimates of this method monitor error, strength, and correlation. These parameters are used to show the response to increasing the number of features used in the splitting. Internal estimates are also used to measure variable importance. These ideas are also applicable to regression.

XGBoost is a decision tree-based ensemble machine learning algorithm that uses a gradient boosting framework [23]. In prediction problems involving unstructured data (images, text, etc.), artificial neural networks typically outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree-based algorithms are considered best in class.

A voting classifier is a hybrid-type machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of the chosen class as the output [24]. It simply aggregates the findings of each of the included classifiers passed into the voting classifier and predicts the output class based on the highest majority of voting. The idea is that instead of creating separate dedicated models and finding accuracy for each of them, we create a single model that is trained by these models and predicts output based on their combined majority of voting for each output class. There are two types of voting modes included in a voting classifier: hard and soft voting. In hard voting mode, the predicted output class is a class with the highest majority of votes, i.e., the class that had the highest probability of being predicted by each of the classifiers. In soft voting, the output class is the prediction based on the average of probability given to that class. In this analysis, logistic regression, random forest, and XGBoost were used as voting classifier internal classifiers and hard voting was selected as a voting mode.

3.4. Performance Measurement and Evaluation

In order to evaluate the performance of the chosen algorithms, the so-called confusion matrix was selected, as it is a common measure used for classification problems. It can be applied both to binary classification as well as to multi-class classification problems. An example of a multidimensional confusion matrix as proposed by Krüger is illustrated in Figure 1:

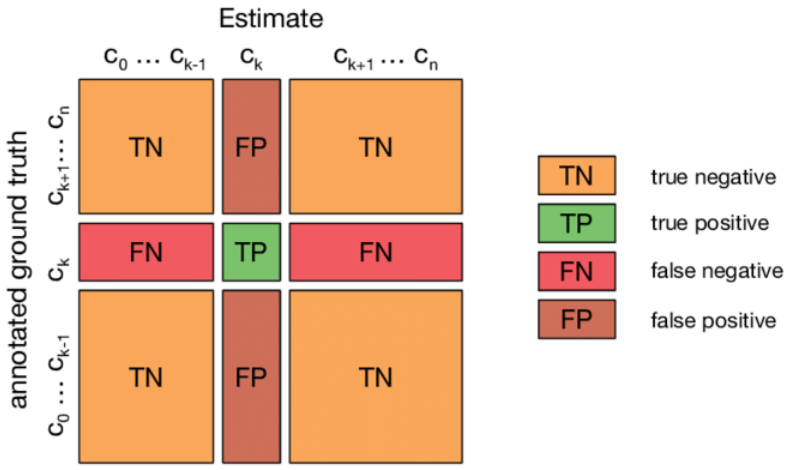

Figure 1. Multidimensional confusion matrix example [25]. Intersection of a column and a row for a given class gives the true positive classification result TP—indicated with green color, while the true negative classification for a given class is represented by TN—orange color, false negative FN—red, and false positive FP—brown.

By definition, a simple 2×2 confusion matrix C is such that $C_{i,j}$ is equal to the number of observations known to be in group i and predicted to be in group j. Thus, in binary

classification, the count of true negatives is $C_{0,0}$, false negatives is $C_{1,0}$, true positives is $C_{1,1}$, and false positives is $C_{0,1}$.

The most frequently used performance metrics for classification according to these values are accuracy (AF), sensitivity (Sn), specificity (Sp), and NPV and PPV values. The accuracy score computes subset accuracy in a multi-label classification task. The set of labels that is predicted to be returned by the classifier for a sample must exactly match the corresponding set of labels in the ground truth (correct) labels subset.

Sensitivity (Sn) is defined as a measure indicating the percentage of the actual positive class covered by a positive prediction. It is calculated as the ratio of correct positive predictions (TP) to all positive results (i.e., TP and FN).

Specificity (Sp) is an analogous measure for negative cases. It determines the ability of the classifier to detect correct negative results (TN) out of all results that could have been negative (i.e., TN and FP). High specificity shows that the classifier is rarely wrong about negative cases. Thus, if it shows that something is positive, we can expect with high probability that it actually is.

Counting the negative predictive value (NPV) measures how many of the negatively predicted examples are actually negative. NPV is the ratio of the number of correctly predicted negative values (TN) by the sum of all negatively predicted ones (including those incorrectly predicted in this way), i.e., TN + FN. The negative predictive value should take values as close to 1 as possible.

The accuracy of a classifier's positive predictions is referred to as precision. It is indicated by the positive predictive value (PPV), which is a measure of how confidently we can trust positive predictions, i.e., what percentage of positive predictions is confirmed by an actually positive state. PPV is defined as a ratio of TP to a sum of TP and all TF. The ideal value of the PPV, with a perfect test, is 1 (100%), and the worst possible value would be zero.

4. Results

The classification results of the LO were calculated using an accuracy factor (AF) and visualized using a graphical representation of a confusion matrix (see Figure 1 example), where a continuous color scale indicates a true or false assignment for one of the given classes: knowledge (W), skills (U), and responsibility & autonomy (K). Five classifiers were run to make decisions about class selection for a given learning outcome from a testing set. Table 1 presents the AF mean score for the five tested algorithms—LR, DT, RF, XGB, and HVC. Detailed results broken down by each algorithms are depicted separately in Figures 2–6.

Table 1. LO classification accuracy.

LR	DT	RF	XGB	HVC
0.94	0.90	0.95	0.95	0.95

The fill colors of individual cells denote the relative range of values indicating the level of classification for each class, while the number values in each cell represent the exact value of classification for a given domain. The records presented diagonally, spanning from the top left to the bottom right, are the K–K, U–U, and W–W true positive values, respectively. Other pairs of the investigated categories, i.e., K–U and K–W (top row), U–K and U–W (central row), and W–K and W–U (bottom row), show results for misclassified LOs.

Figure 2 displays the distribution of the classification results obtained for the logistic regression algorithm. The maximum TP value was produced for the U category (506), with minimal misclassified LOs for K (0) and W (85). W-category classification reached a decent TP value (473), while a substantial number of misclassified LOs fell into U (146); however, none fell into W. The weakest results were achieved for the K category, with only 15 TP values coupled with 255 misclassified LOs for U and 53 for W-claimed labels.

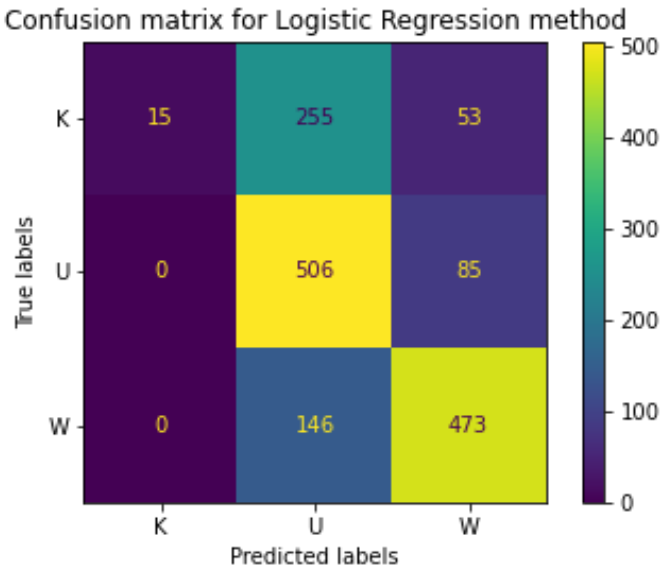

Figure 2. Confusion matrix for logistic regression classifier, where W—knowledge, U—skills, and K—responsibility and autonomy. Color scale ranges from dark purple for 0 matches to light yellow for maximum conformance of prediction with the claimed labels.

Figure 3 shows the distribution of the classification results obtained for the decision tree algorithm. With the DT classifier, max TP was obtained for the U category (513), with few misclassified LOs that fell into K (42), while as few as 12 fell into W. W-category classification reached max TP value (501) coupled with a small number of misclassified LOs that fell into K (20), while 73 fell into W. The results achieved for K category reached 263 TP while 30 misclassified LOs fell into U and just 17 fell into W-claimed labels.

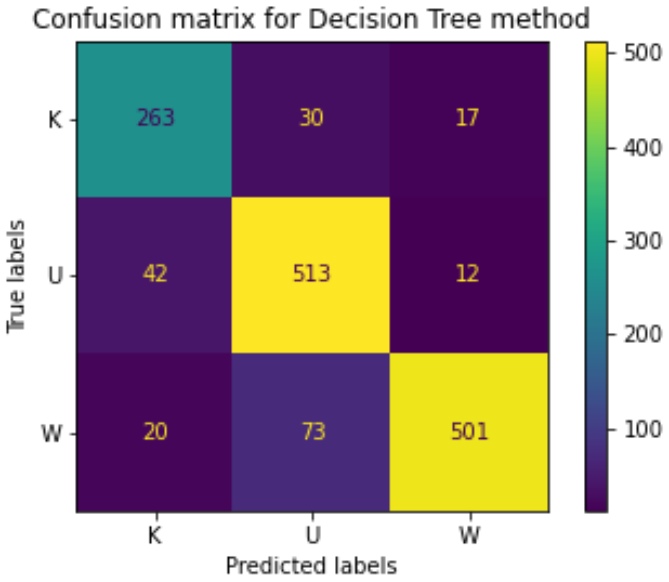

Figure 3. Confusion matrix for decision tree classifier, where W—knowledge, U—skills, and K—responsibility and autonomy. Color scale ranges from dark purple for 0 matches to light yellow for maximum conformance of prediction with the claimed labels.

Figure 4 shows the distribution of the classification results obtained for the random forest algorithm. With the RF classifier, max TP was obtained for the W category (578), with just 1 misclassified LO that fell into K and as few as 15 that fell into U. U-category classification reached max TP value (481) coupled with a small number of misclassified LOs

that fell into K (14), while 72 fell into W. The results achieved for the K category reached 239 while 37 misclassified LOs fell into U, and 34 fell into W-claimed labels.

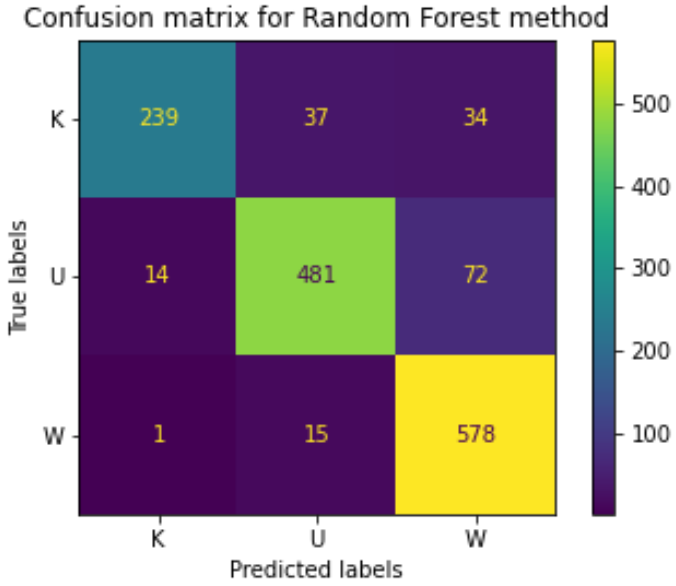

Figure 4. Confusion matrix for random forest classifier, where W—knowledge, U—skills, and K—responsibility and autonomy. Color scale ranges from dark purple for 0 matches to light yellow for maximum conformance of prediction with the claimed labels.

Figure 5 shows the distribution of classification results obtained for the hybrid voting algorithm. With the HV classifier, max TP was obtained for the W category (490), with just one misclassified LO that fell into K and as few as nine that fell into U. U category classification reached the max TP value (462) coupled with a small number (eight) of misclassified LOs that fell into both K and W categories. The results achieved for K category reached 231, coupled with 27 misclassified LOs fell into U, and just 3 that fell into W-claimed labels.

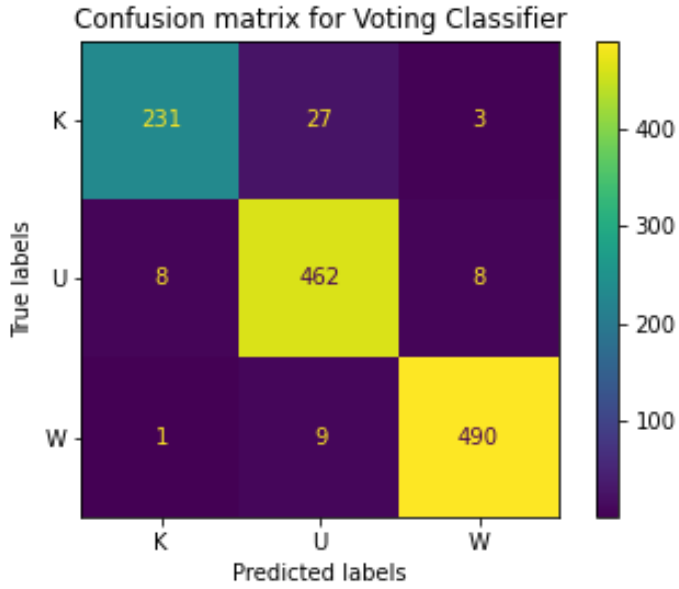

Figure 5. Confusion matrix for hybrid voting classifier, where W—knowledge, U—skills, and K—responsibility and autonomy. Color scale ranges from dark purple for 0 matches to light yellow for maximum conformance of prediction with the claimed labels.

Figure 6 shows the distribution of the classification results obtained for the XGBoost algorithm. With the XGB classifier, max TP was obtained for W category (487), with just four misclassified LOs that fell into K and as few as nine that fell into U. U-category classification reached a max TP value (452) coupled with small number (18) of misclassified LOs that fell into K, and just 8 fell into the W category. The results achieved for the K category reached 222 coupled with 34 misclassified LOs that fell into U, and just 5 fell into W-claimed labels.

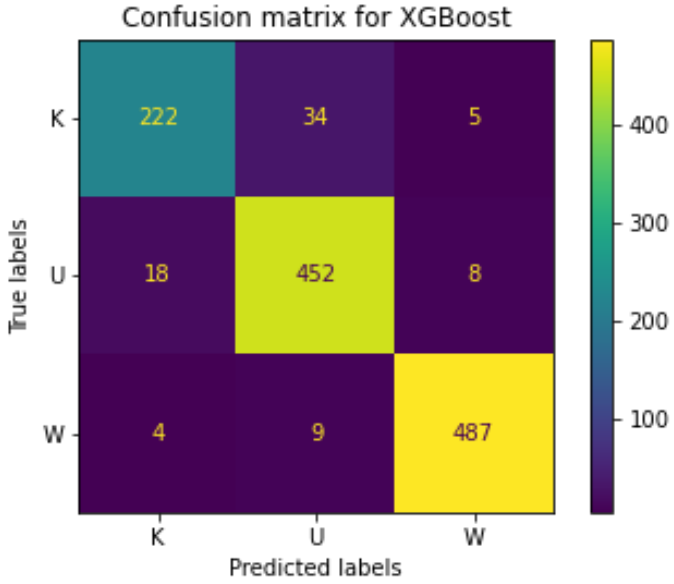

Figure 6. Confusion matrix for XGBoost classifier, where W—knowledge, U—skills, and K—responsibility and autonomy. Color scale ranges from dark purple for 0 matches to light yellow for maximum conformance of prediction with the claimed labels.

To evaluate the classification results, sensitivity (Sn), specificity (Sp), and NPV and PPV values were calculated. Table 2 shows the sensitivity results obtained for the all the tested algorithms (LR, DT, RF, HVC, and XGB) broken down by K, U, and W classes. Most of algorithms performed satisfactorily regarding claimed LO labels, with 84–90% for DT and 86–97% for XGB. The best performance, on average for all the classifiers, was achieved for class U (in the range of 85–97%), with the highest sensitivity achieved by the HVC and XGB, and for class W (76–98%), with the highest sensitivity achieved by the RF, XGB, and HVC classifiers. The sensitivity results for class K were good for DT (85%) and XGB (86%), a bit worse for RF (77%), and notably low for LR (5%) and HVC (1%).

Table 2. LO classification sensitivity values calculated for K, U, and W classes (columns from left to right) broken down for logistic regression—LR, decision tree—DT, random forest—RF, hybrid voting—HVC, and XGBoost—XGB classifiers (top-down rows, respectively).

	Class K	**Class U**	**Class W**
LR	0.05	0.86	0.76
DT	0.85	0.90	0.84
RF	0.77	0.85	0.97
HVC	0.01	0.97	0.98
XGB	0.86	0.95	0.97

Table 3 shows the specificity results obtained for the tested algorithms (LR, DT, RF, HVC, and XGB) broken down by K, U, and W classes. Most of the algorithms performed

satisfactorily regarding true negative classification cases, which were as high as 95–99% for HVC and 94–98% for the XGB algorithms, with up to 8–12% dispersion of results for DT and RF; however, LR achieved only 55% for class U while scoring 100% for class K.

Table 3. LO classification specificity values calculated for K, U, and W classes (columns from left to right) broken down for logistic regression—LR, decision tree—DT, random forest—RF, hybrid voting—HVC, and XGBoost—XGB classifiers (top-down rows, respectively).

	Class K	**Class U**	**Class W**
LR	1.00	0.55	0.79
DT	0.94	0.88	0.96
RF	0.99	0.94	0.87
HVC	0.99	0.95	0.98
XGB	0.98	0.94	0.98

Table 4 shows the negative predictive value results obtained for the tested algorithms (LR, DT, RF, HVC, and XGB) broken down by K, U, and W classes. The highest NPV values were achieved by the HVC and XBG algorithms (96–99% range), followed by DT and RF (90–98%) and LR (80–86%).

Table 4. LO classification NPV values calculated for K, U, and W classes (columns from left to right) broken down for logistic regression—LR, decision tree—DT, random forest—RF, hybrid voting—HVC, and XGBoost—XGB classifiers (top-down rows, respectively).

	Class K	**Class U**	**Class W**
LR	0.80	0.86	0.84
DT	0.96	0.94	0.90
RF	0.94	0.91	0.98
HVC	0.97	0.98	0.99
XGB	0.96	0.97	0.98

Table 5 displays the PPV results obtained for the tested algorithms (LR, DT, RF, HVC, and XGB) broken down by K, U, and W classes. The lowest positive predictive values were achieved for the U and W classes (1–28%), while class K was associated with much higher values (81% for DT and up to 100% for LR).

Table 5. LO classification PPV values calculated for K, U, and W classes (columns from left to right) broken down for logistic regression—LR, decision tree—DT, random forest—RF, hybrid voting—HVC, and XGBoost—XGB classifiers (top-down rows, respectively).

	Class K	**Class U**	**Class W**
LR	1.00	0.28	0.09
DT	0.81	0.05	0.03
RF	0.94	0.07	0.05
HVC	0.96	0.05	0.01
XGB	0.91	0.07	0.01

Figure 7 presents the distribution of the keywords' importance for the decision tree classifier. The word 'able' featured as the most impactful. Notably, most of the featured 15 keywords are either not heavy weighted (top half of the diagram) or not very meaningful (prepositions and short words: 'to', 'for', 'is').

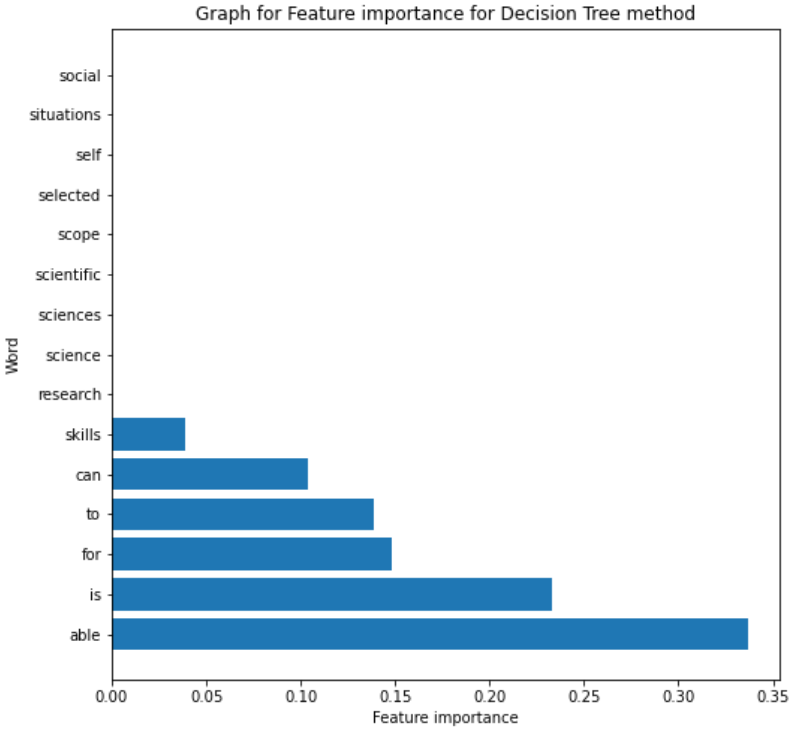

Figure 7. Graph for feature importance for decision tree method.

Figure 8 illustrates the distribution of the keywords' importance for the random forest classifier. The 15-keyword set shown here reveals gradual differences in importance among the features rising top-down; however, the difference is not as significant as for the decision tree algorithm.

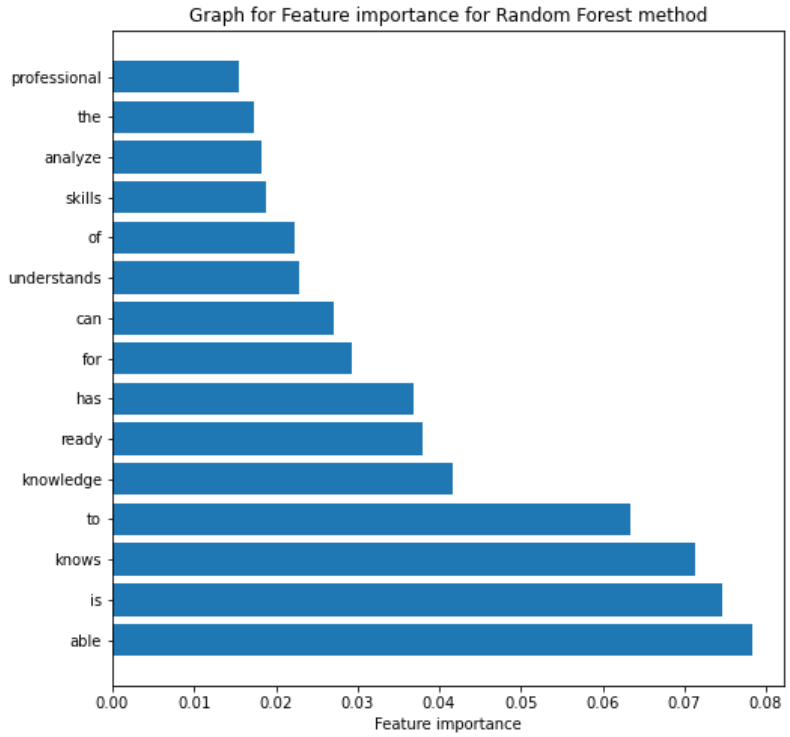

Figure 8. Graph for feature importance for random forest method.

Figure 9 presents the distribution of the keywords' importance for the XGB algorithm. The presented keyword set reveals exponential differences in importance among the features, with the word 'ready' as the most influential keyword for the XGB classifier.

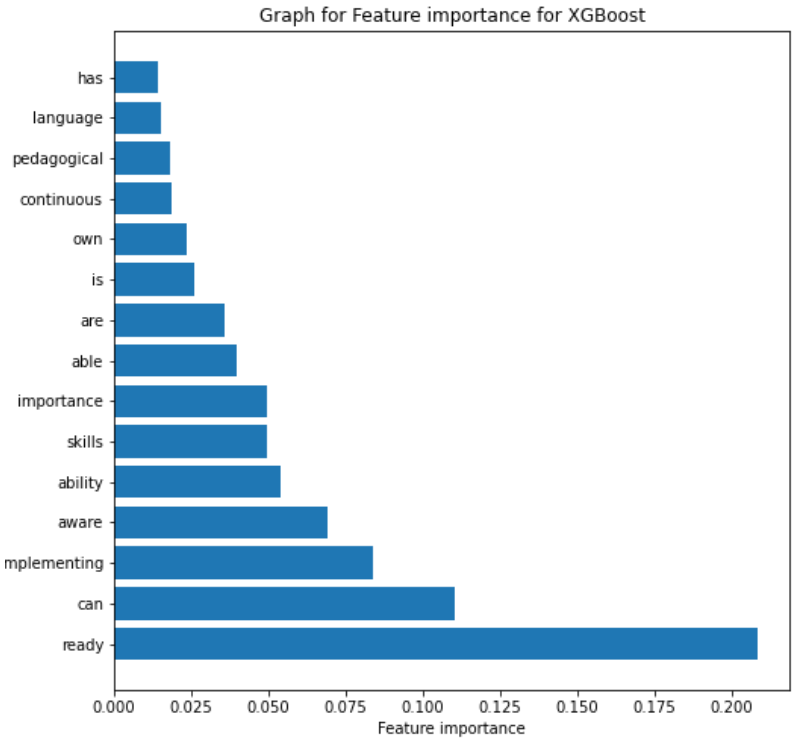

Figure 9. Graph for feature importance for XGBoost method.

5. Discussion

The obtained results have demonstrated how selected machine learning algorithms can predict and categorize LOs using TFIDF vectorization. The first step of the evaluation was to compute the prediction accuracy factor (see Table 1). All the classifiers achieved high AF values, which means a high ratio of the number of correct classifications to the total number of classifications: RF, XGB, and HVC—0.95, LR—0.94, and DT—0.90. The second step was to analyze detailed results of the predictions using confusion matrices. As can be seen inside all of them, the 'skills' and the 'knowledge' categories were identified with the 'best' predicted-to-known scores: LR (skills) = 506, DT (skills) = 513, RF (knowledge) = 578, HVC (knowledge) = 490, and XGB (knowledge) = 487.

On the one hand, these results demonstrate a high potential of using Machine learning-driven prediction for LO categorization. On the other hand, it exposes undesirable tendencies in the selection of proper keywords for definitions of LO contents. By analysis of the 'skills' category case, it can be noticed that the importance of 'able' and/or 'can' terms can cause significantly high 'skills' LO predictions. At the same time, we can observe for most classifiers a relatively high number of false assignments of 'skills' to the 'knowledge' category. This might indicate that too many LO definitions from the 'knowledge' category overuse 'skills'-type words, such as 'can' or 'able'. To some extent, this phenomenon also applies to 'knowledge' category prediction, where almost all ML classifiers scored an importance of knowledge-based words at a very high level; however, there is a significant number of incorrect assignments to the 'skills' category.

The remaining 'responsibility and autonomy' category was also classified with a satisfactory score higher than 200 apart from the LR classifier, which identified 'responsibility and autonomy' LOs as 'skills'—over 250 predictions. For selected classifiers (LR and RF), we can also observe a few incorrect assignments to both 'skills' and 'knowledge'. It means

that LOs in the 'responsibility and autonomy' category can also be oversaturated with words: 'can', 'is able', and knowledge-related terms.

Tendencies found in confusion matrices can also be observed in relevant feature importance graphs (especially for DT, RF, and XGB classifiers)—see Figures 6–8—where, noticeably, some words occur much more frequently than others and will therefore significantly influence prediction efficiency. The widest range of words of importance was recorded for both RT and XGB classifiers, as can be seen in Figures 7 and 8. Additionally, these two classifiers achieved the most uniform confusion matrices with the smallest number of false-positive assignments.

The performance of LO classification was carried out using four measures: specificity (Sp), sensitivity (Sn), PPV, and NPV. The results of this analysis were stored in Tables 2–5.

For all the classes and classifiers, the specificity value varies from 0.55 (class U with logistic regression) to 1.00 (class K predicted by LR). The high specificity we achieved shows that all the classifiers are rarely wrong about negative cases. Differently, in the case of sensitivity, we can observe a relatively low percentage of the actual positive class covered by a positive prediction for class K and two classifiers LR and HVC. For the remaining classifiers, the sensitivity value is relatively high and varies from 0.76 to 0.98, which proves there is a high number of correct predictions for the true class.

The accuracy of a classifier's positive predictions is shown in Table 5. As can be noted, there is a relatively high percentage of positive predictions (LO from given class classified to its class), which are confirmed by an actually positive state for class K, and a low percentage of confirmed positive predictions for the remaining classes U and W. The highest relative precision for all the classes was achieved by the LR classifier (1.00, 0.28, and 0.09), and the lowest was achieved by the DT classifier (0.81, 0.05, and 0.09).

By computing NPV, we could measure how many of the negatively predicted LOs come from different classes. These results are presented in Table 4. All the classifiers achieved a high ratio of NPV—close to 1.00 (0.80–0.99)—which also confirms the high efficiency of LOs classification.

The obtained classification results and classifiers performance analysis generally indicate that most learning outcome sets applied to university teaching processes are built using incorrect word sets that do not correspond explicitly to the three learning domains, making LOs ambiguous, immeasurable, and difficult to interpret.

6. Conclusions

Properly addressing and constructing learning outcomes is one of the most important aspects of the university teaching process. Well-defined learning outcomes can serve as measurable statements that articulate what students should know, be able to do, or value as a result of taking a course or completing a program. Teachers can use learning outcomes as a tool to efficiently inform students about teaching strategies, course activities, and assessments. Clearly identified learning outcomes allow teachers to make correct decisions about selecting course content, design knowledge, and skills-oriented assessments, help them to develop teaching strategies or learning activities that will help students develop their knowledge and skills, and accurately and effectively measure students' learning levels. On the other hand, having access to articulated learning outcomes helps students to evaluate if a course is a good fit for their academic trajectory, identify what they need to do to be successful in the course, take responsibility for their progress, and be mindful of what they are actually learning. Atkinson claims in his work that there should be greater transparency in the relationship between competencies and intended learning outcomes [26]. This educational area includes a wide range of teaching aspects that can be efficiently supported by computational intelligence. In this paper, the authors demonstrated an example of automatic LO classification based on selected machine learning techniques.

The developed solution allows for:

1. Automatic verification of the category assignment to a given learning outcome;
2. Automatic quantitative verification of the categorization of learning outcomes;

3. Measuring performance of learning outcomes' automatic classification;
4. Opening new directions for further research in the area of formal educational frameworks;

In this research work, the English language was tested as a common language, which potentially provides the possibility of monitoring and analyzing the directional systems of learning outcomes at the level of the European Qualification Framework. As preliminary classification results obtained in this study seem to be promising, the authors intend to conduct further research. It would be interesting to employ big data methods for inspection of larger datasets [27] as well as explore specific approaches such as the multistage ML approach as shown in [28] where initial regularization and PC analysis preceded linear regression methods. Moreover, further studies for more languages and educational fields using natural language-processing techniques to help identify incorrectly defined learning outcomes would be aimed. Implementing LOs in higher education is still a challenge, especially due to the fact that it is uncertain as to what their impacts on teaching and learning are [29].

Author Contributions: Conceptualization, A.K. (Aleksandra Kowalska), R.B., J.S., M.W.-L., A.K. (Adrianna Kozłowska) and A.R.; methodology, J.S., A.R., A.K. (Aleksandra Kowalska), A.K. (Adrianna Kozłowska), M.W.-L. and R.B.; software, A.K. (Aleksandra Kowalska) and R.B.; validation, A.K. (Aleksandra Kowalska), R.B. and M.W.-L.; formal analysis, A.K. (Aleksandra Kowalska), R.B., J.S. and M.W.-L.; investigation, J.S.; resources, M.W.-L.; data curation, A.K. (Aleksandra Kowalska) and R.B.; writing—original draft preparation, A.K. (Aleksandra Kowalska), R.B., J.S. and M.W.-L.; writing—review and editing, A.R. and A.K. (Adrianna Kozłowska).; visualization, R.B.; supervision, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financed by the Lodz University of Technology, Poland as a part of statutory activity.

Data Availability Statement: Datasets on learning outcomes supporting reported results were taken from the selected major polish universities websites: Malopolska School of Economics in Tarnow, UMCS University in Lublin, Nicolaus Copernicus University in Torun, AFMKU in Cracow, Faculty Of Education-University Of Bialystok, University of Social Sciences in Lodz, The Mazovian State University in Plock, University of Warsaw, Unversity of Wroclaw, University of Opole, The Faculty of Educational Studies at Adam Mickiewicz University, University of Silesia in Katowice, Academy of Applied Medical and Social Sciences in Elblag, University of Gdansk and Warsaw University of Life Sciences. Each university in Poland is obliged to provide the information about learning outcomes.

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

References

1. European Education and Culture Executive Agency, Eurydice. *The European Higher Education Area in 2018: Bologna Process Implementation Report*; Publications Office: Luxemburg, 2018. [CrossRef]
2. Council Recommendation of 22 May 2017 on the European Qualifications Framework for Lifelong Learning and Repealing the Recommendation of the European Parliament and of the Council of 23 April 2008 on the Establishment of the European Qualifications Framework for Lifelong Learning (2017/C 189/03). 2017. Available online: https://tinyurl.com/4e2rhw68 (accessed on 6 July 2022).
3. Biggs, J. Enhancing teaching through constructive alignment. *High. Educ.* **1996**, *32*, 347–364. [CrossRef]
4. Tang, J.B.C. *Teaching for Quality Learning at University*; Open University Press: Maidenhead, UK, 2011.
5. Battersby, M. *So, What's a Learning Outcome Anyway?* Technical Report; British Columbia Ministry of Advanced Education, Centre for Curriculum, Transfer and Technology: Vancouver, BC, Canada, 1999.
6. Recommendation of the European Parliament and of the Council of 23 April 2008 on the Establishment of the European Qualifications Framework for Lifelong Learning (2008/C 111/01). 2008. Available online: https://tinyurl.com/yy2bfabr (accessed on 6 July 2022).
7. Bloom, B. *Taxonomy of Educational Objectives: The Classification of Educational Goals*; Bloom, B.S., Ed.; David McKay Company Inc.: New York, NY, USA, 1956.
8. European Commission and Directorate-General for Education, Youth, Sport and Culture: The EU in Support of the Bologna Process (2017/C 189/01). 2018. Available online: https://tinyurl.com/4476evnk (accessed on 13 June 2022). [CrossRef]
9. Pasterk, S.; Kesselbacher, M.; Bollin, A. A Semi-automated Approach to Categorise Learning Outcomes into Digital Literacy or Computer Science. In *Empowering Learners for Life in the Digital Age*; Part 2: Programming and Computer Science Education; Open Conference on Computers in Education (OCCE); Passey, D., Bottino, R., Lewin, C., Sanchez, E., Eds.; Springer International Publishing: Linz, Austria, 2018; Volume AICT-524, pp. 77–87. [CrossRef]

10. Chang, W.C.; Chung, M.S. Automatic applying Bloom's taxonomy to classify and analysis the cognition level of English question items. In Proceedings of the 2009 Joint Conferences on Pervasive Computing (JCPC), Taipei, Taiwan, 3–5 December 2009; pp. 727–734. [CrossRef]
11. Omar, N.; Haris, S.S.; Hassan, R.; Arshad, H.; Rahmat, M.; Zainal, N.F.A.; Zulkifli, R. Automated Analysis of Exam Questions According to Bloom's Taxonomy. *Procedia-Soc. Behav. Sci.* **2012**, *59*, 297–303. [CrossRef]
12. Harrison, J.; Dikken, O.; van Peer, D. Question Classification According to Bloom's Revised Taxonomy. 2017. Available online: https://tinyurl.com/5n953xx7 (accessed on 14 June 2022).
13. Osman, A.; Yahya, A.A. Classifications of Exam Questions Using Linguistically-Motivated Features: A Case Study Based on Bloom's Taxonomy. 2016. Available online: https://tinyurl.com/4y89bua2 (accessed on 16 June 2022).
14. Zhang, J.; Wong, C.; Giacaman, N.; Luxton-Reilly, A. Automated Classification of Computing Education Questions Using Bloom's Taxonomy. In *ACE '21: Australasian Computing Education Conference, Online, 2–4 February 2021*; Association for Computing Machinery: New York, NY, USA, 2021; pp. 58–65. [CrossRef]
15. van Hoeij, M.J.; Haarhuis, J.C.; Wierstra, R.F.; van Beukelen, P. Developing a Classification Tool Based on Bloom's Taxonomy to Assess the Cognitive Level of Short Essay Questions. *J. Vet. Med. Educ.* **2004**, *31*, 261–267. [CrossRef] [PubMed]
16. Bengio, Y.; Senecal, J.S. Adaptive Importance Sampling to Accelerate Training of a Neural Probabilistic Language Model. *IEEE Trans. Neural Netw.* **2008**, *19*, 713–722. [CrossRef] [PubMed]
17. Shaikh, S.; Daudpotta, S.M.; Imran, A.S. Bloom's Learning Outcomes' Automatic Classification Using LSTM and Pretrained Word Embeddings. *IEEE Access* **2021**, *9*, 117887–117909. [CrossRef]
18. Abduljabbar, D.A.; Omar, N. Exam questions classification based on Bloom's taxonomy cognitive level using classifiers combination. *J. Theor. Appl. Inf. Technol.* **2015**, *78*, 447–455.
19. Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
20. Wu, H.; Luk, R.; Wong, K.; Kwok, K. Interpreting TF-IDF term weights as making relevance decisions. *ACM Trans. Inf. Syst.* **2008**, *26*, 1–37. [CrossRef]
21. Raschka, S. *Python Machine Learning*; Packt Publishing Ltd.: Birmingham, UK, 2015.
22. Song, Y.Y.; Ying, L.U. Decision tree methods: Applications for classification and prediction. *Shanghai Arch. Psychiatry* **2015**, *27*, 130–135. [CrossRef] [PubMed]
23. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–7945. [CrossRef]
24. Zhang, Y.; Zhang, H.; Cai, J.; Yang, B. Weighted Voting Classifier Based on Differential Evolution. In *Abstract and Applied Analysis*; Hindawi: London, UK, 2014; pp. 1–6. [CrossRef]
25. Krüger, F. Activity, Context, and Plan Recognition with Computational Causal Behaviour Models. Ph.D. Thesis, Hoschschule Wismar-University of Applied Sciences, Technology, Business and Design, Wismar, Germany, 2016.
26. Atkinson, S. Graduate Competencies, Employability and Educational Taxonomies: Critique of Intended Learning Outcomes. *Pract. Evid. Scholarsh. Teach. Learn. High. Educ.* **2015**, *10*, 154–177.
27. Romanowski, A. Big Data-Driven Contextual Processing Methods for Electrical Capacitance Tomography. *IEEE Trans. Ind. Inform.* **2016**, *15*, 1609–1618. [CrossRef]
28. Rymarczyk, T.; Krol, K.; Kozlowski, E.; Wolowiec, T.; Cholewa-Wiktor, M.; Bednarczuk, P. Application of Electrical Tomography Imaging Using Machine Learning Methods for the Monitoring of Flood Embankments Leaks. *Energies* **2021**, *14*, 8081. [CrossRef]
29. Havnes, A.; Prøitz, T.S. Why use Learning Outcomes in Higher Education? Exploring the Grounds for Academic Resistance and Reclaiming the Value of Unexpected Learning. *Educ. Assess. Eval. Account.* **2016**, *28*, 205–223. [CrossRef]