Shodhshauryam, International Scientific Refereed Research Journal
Available online at : www.shisrrj.com
OPEN ACCESS
ISSN : 2581-6306
© 2024 SHISRRJ | Volume 7 | Issue 2
doi : https://doi.org/10.32628/SHISRRJ

Measuring the Influence of Student Assessments on Learning Outcomes in Machine Learning

B. Rupadevi[1], M. Pandimeena[2]
[1]Associate Professor, Department of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, Andhra Pradesh, India
[2]Post Graduate, Department of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, Andhra Pradesh, India

Article Info

Publication Issue :
March-April-2024
Volume 7, Issue 2

Page Number : 07-13

Article History
Received : 15 March 2024
Published : 30 March 2024

ABSTRACT

A major difficulty in educational research is determining the pedagogical impact of student evaluations on teaching effectiveness and learning outcomes. In this work, we provide a novel method for assessing how student feedback affects pedagogy and learning outcomes by fusing statistical analysis and text mining tools. Using natural language processing methods, we extract pertinent textual data from student assessments and then use it to find important themes, feelings, and patterns in the feedback corpus. We then use statistical techniques to measure the correlation between the themes that have been found and different educational indicators including student performance, contentment, and engagement. Our method helps educators and policy make understand the impact of teaching strategies on student learning experiences and their success by evaluating large-scale datasets of student evaluations collected over numerous semesters. We show the effectiveness and usefulness of our text mining and statistical methodology for evaluating the pedagogical impact of student evaluation in education through empirical validation and case studies. This project adds to the continuing conversation about educational evaluation and offers insightful suggestions for raising instructional standards and boosting student learning.

Keywords : Text Mining, Statistics, Pedagogical Impact, Student Evaluation, Teaching Effectiveness and Learning Outcomes.

I. INTRODUCTION

Within the field of education, pedagogical impact evaluation continues to be a crucial undertaking with the objective of augmenting the efficacy of instruction and elevating student outcomes. A key component of this process is student evaluation of teaching, which offers insightful commentary on both instructional strategies and the overall quality of the educational process. To extract useful insights and evaluate their instructional impact, educators and administrators face tremendous obstacles due to the volume and complexity of student evaluation data. To address these issues, a new method that combines statistical analysis and text mining provides a viable way to assess the pedagogical effects of student evaluation in a methodical manner. Computational linguistics' text mining technique has enormous promise for obtaining important data from vast amounts of unstructured textual data, including evaluation forms filled out by students. Text mining algorithms can uncover important themes, attitudes, and patterns in student feedback by utilizing natural language processing techniques. This allows educators to gain more profound understanding of the advantages and disadvantages of their teaching methods. Statistical analysis provides a rigorous framework for evaluating the correlations between educational results and student evaluation indicators when combined with text mining. Educators can evaluate the effects of their teaching methods on the learning outcomes of their students, their happiness with their courses, and their overall academic achievement by using statistical approaches such as regression modelling and correlation analysis. Through the integration of text mining and statistical analysis, educators can maximize teaching and learning experiences by making data-driven decisions and gaining a thorough understanding of the elements influencing pedagogical efficacy. Furthermore, the suggested method shows promise in resolving long-standing issues with educational assessment, such as the subjectivity of qualitative assessments and the shortcomings of conventional manual analytic techniques. Educators can overcome biases and inconsistencies inherent in traditional evaluation processes by automating the examination of student input using text mining and statistical algorithms. This will result in assessments of educational impact that are more objective and trustworthy. Moreover, the utilization of a text mining and statistics technique has consequences that extend beyond specific educational settings and organizations. Researchers and policymakers can find trends, best practices, and areas for change on a larger scale by compiling and evaluating student evaluation data from various educational settings. Through collaboration, a culture of continual improvement in education is fostered, leading to quality and innovation in teaching and learning methodologies. In conclusion, a potent method for evaluating the pedagogical effects of student evaluation in education is provided by the combination of text mining and statistical analysis. Educators can improve the quality of teaching and learning experiences for students by using computational algorithms to extract meaningful insights from student feedback and support evidence-based decision-making.

II. METHODS AND MATERIAL

A. Data Collection

The first step of the module is gathering data from educational institutions regarding student evaluations, which includes both written comments and numerical ratings. In this project the written comments are taken as the test.csv file in this file it has student comments where n=3912 it has eight attributes that is text_id, full_text, concerned, resentful, uncertain, dissatisfied, confident and contented after training this dataset the train data as stored in another csv file named it as train.csv file. And student_prediction.csv file it has 145 rows and 33 columns.

B. Data Loading and Initial Exploration

The dataset student_prediction.csv and importing the required libraries. The dataset is first explored, with tasks such as locating unique values in the 'COURSE ID' column, presenting basic statistics, and looking for duplicates. The 'STUDENTID' column most likely acts as an identifier, it is removed. To keep informative features, feature selection is done using Mutual Information (MI) scores.

C. Data visualization

A range of plots, including bar plots for categorical features and bar graphs comparing feature values across multiple "SUMM" levels, are created to illustrate the correlations between features and the target variable SUMM.

D. Clustering

The dataset is subjected to K-means clustering, which results in the creation of a new feature called "Cluster," which represents the data points' cluster assignments. By include the 'Cluster' feature in the dataset, the classification models may be able to make use of the data that the clustering method has managed to gather. If the clusters in the data about the target variable "SUMM" show significant patterns or groups, the models might benefit from having this extra feature. By learning to differentiate across various clusters, the models can enhance their classification ability. By spotting patterns or groupings in the data, clustering helps to extract more information that the classification models may use to provide more precise predictions.

E. Model Training and Evaluation

Training the classification models (Decision Tree, Random Forest, KNN, Gaussian Naive Bayes, and Support Vector Classifier) requires the pre-processed data. By using this classification, we can find the Model accuracy, precision, recall, and F1-score are evaluated through the generation of classification reports, and model performance is tested by cross validation.

III. RESULTS AND DISCUSSION

A. Student Feedback Analysis

The first step is to import the required library and modules. This includes NumPy and pandas for data manipulation, Matplotlib and seaborn for data visualization, NLTK for natural language processing tasks, LightGBM for regression, scikit-learn for model evaluation and validation, and Gensim for topic modelling and then load the required datasets and train the data. And then print the test data as test. head ().

	text_id	full_text
0	0000C359D63E	when a person has no experience on a job their...
1	000BAD50D026	Do you think students would benefit from being...
2	00367BB2546B	Thomas Jefferson once states that "it is wonde...

Figure 1. Train dataset of Student Feedback Analysis

The next step is Exploratory Data Analysis (EDA) this examine how the values of different variables are distributed. Check the score_feature of the distribution of 6 different variables this can be shown in the form of histogram.

The next step is Exploratory Data Analysis (EDA) this examine how the values of different variables are distributed. Check the score_feature of the distribution of 6 different variables this can be shown in the form of histogram.

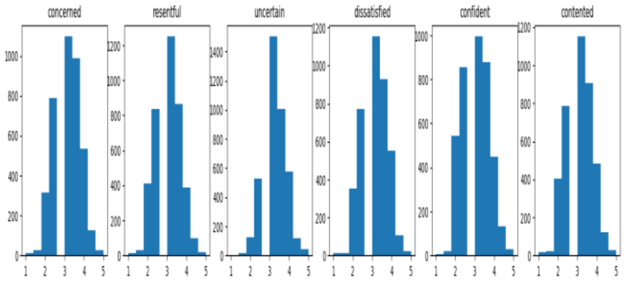

Figure 2. Score Feature of the Variables

This histogram represents the score_feature of the variables. To check standard length, minimum length, maximum length of the text-dataset.csv

Average length: 2333.35

Std length: 1034.51

Min length: 76.00

Max length: 6044.00

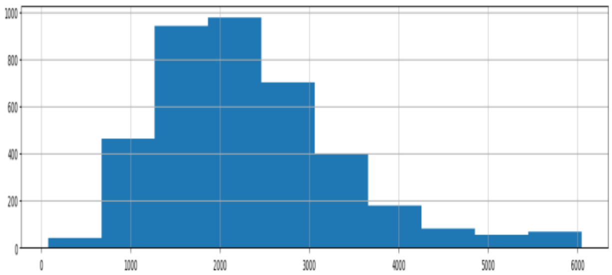

Figure 3. Length of student feedback analysis

This above graph represents the length of the text present in text-data. And then check is any null values or duplicate data is present in the dataset. If that dataset has no null values and duplicates then find the correlation of EDA.

Figure 4. Correlation of Exploratory Data Analysis, here 1 represent the correlation between each variable

Here all the variables are correlated to each Other. Vectorization and Modelling build 6 models each predict a score to represent an essay, I will use Doc2Vec (D2V) model to encode the essay to an array

```
TaggedDocument(words=['i', 'think', 'that', 'student', 'would', 'benefit', 'from', 'learn', 'at', 'home', 'becaus', 'they', 'wo
nt', 'have', 'to', 'chang', 'and', 'get', 'up', 'earli', 'in', 'the', 'morn', 'to', 'shower', 'and', 'do', 'there', 'hair', 'ta
ke', 'onli', 'class', 'help', 'them', 'becaus', 'at', 'there', 'hous', 'they', 'll', 'be', 'pay', 'more', 'attent', 'they', 'wi
ll', 'be', 'comfort', 'at', 'home', 'the', 'hardest', 'part', 'of', 'school', 'is', 'get', 'readi', 'you', 'wake', 'up', 'go',
'brush', 'your', 'teeth', 'and', 'go', 'to', 'your', 'closet', 'and', 'look', 'at', 'your', 'cloth', 'after', 'you', 'think',
'you', 'pick', 'a', 'outfit', 'u', 'go', 'look', 'in', 'the', 'mirror', 'and', 'youll', 'either', 'not', 'like', 'it', 'or', 'y
ou', 'look', 'and', 'see', 'a', 'stain', 'then', 'you', 'll', 'have', 'to', 'chang', 'with', 'the', 'onlin', 'class', 'you', 'c
an', 'wear', 'anyth', 'and', 'stay', 'home', 'and', 'you', 'wont', 'need', 'to', 'stress', 'about', 'what', 'to', 'wear', 'mos
t', 'student', 'usual', 'take', 'shower', 'befor', 'school', 'they', 'either', 'take', 'it', 'befor', 'they', 'sleep', 'or', 'w
hen', 'they', 'wake', 'up', 'some', 'student', 'do', 'both', 'to', 'smell', 'good', 'that', 'caus', 'them', 'do', 'miss', 'th
e', 'bus', 'and', 'effect', 'on', 'there', 'lesson', 'time', 'caus', 'they', 'come', 'late', 'to', 'school', 'when', 'u', 'hav
e', 'onlin', 'class', 'u', 'wont', 'need', 'to', 'miss', 'lesson', 'caus', 'you', 'can', 'get', 'everyth', 'set', 'up', 'and',
'go', 'take', 'a', 'shower', 'and', 'when', 'u', 'get', 'out', 'your', 'readi', 'to', 'go', 'when', 'your', 'home', 'your', 'co
mfort', 'and', 'you', 'pay', 'attent', 'it', 'give', 'then', 'an', 'advantag', 'to', 'be', 'smarter', 'and', 'even', 'pass', 't
here', 'classmat', 'on', 'class', 'work', 'public', 'school', 'are', 'difficult', 'even', 'if', 'you', 'tri', 'some', 'teache
r', 'dont', 'know', 'how', 'to', 'teach', 'it', 'in', 'then', 'way', 'that', 'student', 'understand', 'it', 'that', 'caus', 'st
udent', 'to', 'fail', 'and', 'they', 'may', 'repeat', 'the', 'class'], tags=[3.5, 3.5, 3.0, 3.0, 4.0, 3.0])
```

In tokenization and stemming which tokenizes text based on a regular expression pattern and then applies stemming using the snowball stemmer for English. This function first tokenized the text using "nltk-regexp-tokenizer ()" and then filters out non-alphabetic tokens and the text can be return in the form of list.

B. Student Evaluation of Teaching

After importing the library load the dataset by using df display the dataset.

	STUDENTID	AGE	GENDER	HS_TYPE	SCHOLARSHIP	WORK	ACTIVITY	PARTNER	SALARY	TRANSPORT	...	PREP_STUDY	PREP_EXAM	NOTES
0	STUDENT1	2	2	3	3	1	2	2	1	1	...	1	1	3
1	STUDENT2	2	2	3	3	1	2	2	1	1	...	1	1	3
2	STUDENT3	2	2	2	3	2	2	2	2	4	...	1	1	2
3	STUDENT4	1	1	1	3	1	2	1	2	1	...	1	2	3
4	STUDENT5	2	2	1	3	2	2	1	3	1	...	2	1	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
140	STUDENT141	2	1	2	3	1	1	2	1	1	...	1	1	2
141	STUDENT142	1	1	2	4	2	2	2	1	4	...	1	1	3
142	STUDENT143	1	1	1	4	2	2	2	1	1	...	1	1	3
143	STUDENT144	2	1	2	4	1	1	1	5	2	...	2	1	2
144	STUDENT145	1	1	1	5	2	2	2	3	1	...	2	1	3

145 rows × 33 columns

Figure 5. Student Evaluation Dataset

The above results show the attributes of the dataset it has 145rows and 33columns.And then display the head part of the dataset

	STUDENTID	AGE	GENDER	HS_TYPE	SCHOLARSHIP	WORK	ACTIVITY	PARTNER	SALARY	TRANSPORT	LIVING	MOTHER_EDU	FATHER_EDU	#_SIBLI
0	STUDENT1	2	2	3	3	1	2	2	1	1	1	1	2	
1	STUDENT2	2	2	3	3	1	2	2	1	1	1	2	3	
2	STUDENT3	2	2	2	3	2	2	2	2	4	2	2	2	
3	STUDENT4	1	1	1	3	1	2	1	2	1	2	1	2	
4	STUDENT5	2	2	1	3	2	2	1	3	1	4	3	3	

Figure 6. Train data of Student Evaluation Dataset

And display the tail part of the dataset.

```
df.tail()
```

	STUDENTID	AGE	GENDER	HS_TYPE	SCHOLARSHIP	WORK	ACTIVITY	PARTNER	SALARY	TRANSPORT	LIVING	MOTHER_EDU	FATHER_EDU	#_SI
140	STUDENT141	2	1	2	3	1	1	2	1	1	2	1	2	
141	STUDENT142	1	1	2	4	2	2	2	1	4	2	1	1	
142	STUDENT143	1	1	1	4	2	2	2	1	1	1	3	4	
143	STUDENT144	2	1	2	4	1	1	1	5	2	3	4	4	
144	STUDENT145	1	1	1	5	2	2	2	3	1	1	3	1	

Figure 7. Tail part of the Student Evaluation Dataset

Now check the data type of the attributes present in dataset.

```
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   STUDENTID       145 non-null    object
 1   AGE             145 non-null    int64
 2   GENDER          145 non-null    int64
 3   HS_TYPE         145 non-null    int64
 4   SCHOLARSHIP     145 non-null    int64
 5   WORK            145 non-null    int64
 6   ACTIVITY        145 non-null    int64
 7   PARTNER         145 non-null    int64
 8   SALARY          145 non-null    int64
 9   TRANSPORT       145 non-null    int64
 10  LIVING          145 non-null    int64
 11  MOTHER_EDU      145 non-null    int64
 12  FATHER_EDU      145 non-null    int64
 13  #_SIBLINGS      145 non-null    int64
 14  KIDS            145 non-null    int64
 15  MOTHER_JOB      145 non-null    int64
 16  FATHER_JOB      145 non-null    int64
 17  STUDY_HRS       145 non-null    int64
 18  READ_FREQ       145 non-null    int64
 19  READ_FREQ_SCI   145 non-null    int64
 20  ATTEND_DEPT     145 non-null    int64
 21  IMPACT          145 non-null    int64
 22  ATTEND          145 non-null    int64
 23  PREP_STUDY      145 non-null    int64
 24  PREP_EXAM       145 non-null    int64
 25  NOTES           145 non-null    int64
 26  LISTENS         145 non-null    int64
 27  LIKES_DISCUSS   145 non-null    int64
 28  CLASSROOM       145 non-null    int64
 29  CUML_GPA        145 non-null    int64
 30  EXP_GPA         145 non-null    int64
 31  COURSE ID       145 non-null    int64
 32  SUMM            145 non-null    int64
dtypes: int64(32), object(1)
memory usage: 37.5+ KB
```

Figure 8. Checking the datatypes of the attributes present in the dataset

The gradient is used to draw attention to data variances or trends. The background of the Data Frame can be made gradient-coloured to help you rapidly see the relative values' magnitudes in various rows or columns. For example, adding a gradient can help you identify which values are comparatively greater or lower compared to others in a Data Frame created by df.describe(), which provides summary statistics like mean, min, max, etc. Finding patterns, outliers, or interesting places in the data is made simpler with the help of this visual indication.

	count	mean	std	min	25%	50%	75%	max
AGE	145.000000	1.620690	0.613154	1.000000	1.000000	2.000000	2.000000	3.000000
GENDER	145.000000	1.600000	0.491596	1.000000	1.000000	2.000000	2.000000	2.000000
HS_TYPE	145.000000	1.944828	0.537216	1.000000	2.000000	2.000000	2.000000	3.000000
SCHOLARSHIP	145.000000	3.572414	0.805750	1.000000	3.000000	3.000000	4.000000	5.000000
WORK	145.000000	1.662069	0.474644	1.000000	1.000000	2.000000	2.000000	2.000000
ACTIVITY	145.000000	1.600000	0.491596	1.000000	1.000000	2.000000	2.000000	2.000000
PARTNER	145.000000	1.579310	0.495381	1.000000	1.000000	2.000000	2.000000	2.000000
SALARY	145.000000	1.627586	1.020245	1.000000	1.000000	1.000000	2.000000	5.000000
TRANSPORT	145.000000	1.620690	1.061112	1.000000	1.000000	1.000000	2.000000	4.000000
LIVING	145.000000	1.731034	0.783999	1.000000	1.000000	2.000000	2.000000	4.000000
MOTHER_EDU	145.000000	2.282759	1.223062	1.000000	1.000000	2.000000	3.000000	6.000000
FATHER_EDU	145.000000	2.634483	1.147544	1.000000	2.000000	3.000000	3.000000	6.000000
#_SIBLINGS	145.000000	2.806897	1.360640	1.000000	2.000000	3.000000	4.000000	5.000000
KIDS	145.000000	1.172414	0.490816	1.000000	1.000000	1.000000	1.000000	3.000000
MOTHER_JOB	145.000000	2.358621	0.805156	1.000000	2.000000	2.000000	2.000000	5.000000
FATHER_JOB	145.000000	2.806897	1.329664	1.000000	2.000000	3.000000	4.000000	5.000000
STUDY_HRS	145.000000	2.200000	0.917424	1.000000	2.000000	2.000000	3.000000	5.000000
READ_FREQ	145.000000	1.944828	0.562476	1.000000	2.000000	2.000000	2.000000	3.000000

Figure 9. The Count of every Attribute, mean, standard deviation, min, and max values

In the above table represent the count of every attribute, mean, standard deviation, min, and max values. And then now check whether it has any duplicate values or attributes. Check whether it has any duplicate Rows. In this dataset it has no duplicate rows.

#checking the duplicate

duplicate = df [df. duplicated ()]

print ("Duplicate Rows:")

duplicate

Duplicate Row: 0

To calculate the Mutual Information (MI) scores, we utilized statistical properties such as standard deviation, minimum, maximum, and mean of the features. First, we computed these statistics for each feature in the dataset. The standard deviation represents the dispersion of values around the mean, providing insights into the variability within the feature. The minimum and maximum values denote the range of values present in the feature, offering an understanding of the data spread. Additionally, the mean serves as a measure of central tendency, indicating the average value of the feature. By incorporating these statistics into the MI score calculation, we captured both the spread and central tendency of each feature relative to the target variable. This approach enabled us to assess the information shared between the features and the target, facilitating feature selection and model building processes.

IMPACT	ATTEND	PREP_STUDY	PREP_EXAM	NOTES	LISTENS	LIKES_DISCUSS	CLASSROOM	CUML_GPA	EXP_GPA	COURSE ID	SUMM
45.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000	145.000000
1.206897	1.241379	1.337931	1.165517	2.544828	2.055172	2.393103	1.806897	3.124138	2.724138	4.131034	3.227586
0.588035	0.429403	0.614870	0.408483	0.564940	0.674736	0.604343	0.810492	1.301083	0.916536	3.260145	2.197678
1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
1.000000	1.000000	1.000000	1.000000	2.000000	2.000000	2.000000	1.000000	2.000000	2.000000	1.000000	1.000000
1.000000	1.000000	1.000000	1.000000	3.000000	2.000000	2.000000	2.000000	3.000000	3.000000	3.000000	3.000000
1.000000	1.000000	2.000000	1.000000	3.000000	3.000000	3.000000	2.000000	4.000000	3.000000	7.000000	5.000000
3.000000	2.000000	3.000000	3.000000	3.000000	3.000000	3.000000	3.000000	5.000000	4.000000	9.000000	7.000000

Figure 10. Predication values of datasets

Based on these values we can apply the classification models to predict the accuracy of the model. The cross_val_predict () method is used to carry out cross-validation. The dataset is divided into a given number of folds (cv=5 in this example), the model is trained on a subset of the folds, and the learned model is then used to predict the target variable for the remaining folds. The predict variable contains the predictions produced by cross-validation.

IV. CONCLUSION

The project concludes by showing the need of combining text mining and statistical methods to evaluate the pedagogical influence of student assessments of instruction on educational learning outcomes. Educators and institutions can obtain actionable insights to support student performance and evidence-based teaching practices by utilizing quantitative analysis and textual feedback. To further improve methods, solve issues, and deepen our grasp of the intricate relationship between instructional quality, student feedback, and learning outcomes in educational settings, further study and collaboration in this field are imperative.

V. REFERENCE

[1]. Y. J. Lin and H. Wang, "Using virtual reality to facilitate learner's creative self-efficacy and intrinsic motivation in an EFL classroom", Educ. Inf. Technol., vol. 26, pp. 4487-4505, Mar. 2021.

[2]. G. Yusufu and N. Nathan, "A novel model of smart education for the development of smart university system", Proc. Int. Conf. Math. Comput. Eng. Comput. Sci. (ICMCECS), pp. 1-5, Mar. 2020.

[3]. F. Pettersson, "Understanding digitalization and educational change in school by means of activity theory and the levels of learning concept", Educ. Inf. Technol., vol. 26, no. 1, pp. 187-204, Jun. 2020.

[4]. M. S. Chiu, "Exploring models for increasing the effects of school information and communication technology use on learning outcomes through outside-school use and socioeconomic status mediation: The ecological techno-process", Educ. Technol. Res. Develop., vol. 68, no. 1, pp. 413-436, Feb. 2020.

[5]. E. K. Zashchitina and P. V. Pavlov, "Higher education today: Mass or individual approach", Proc. Int. Conf. Quality Manage. Transp. Inf. Secur. Inf. Technol. (IT IS), pp. 653-656, Sep. 2019.

[6]. L. Widger, M. Denny, M. Benke, M. Pajnkihar, C. Bruen and C. Madden, "The strategic implementation and augmentation of technology enhanced learning (TEL) in third level education: A critical lens", Proc. 10th Int. Technol. Educ. Develop. Conf. (INTED), pp. 1289-1297, Mar. 2016.
[7]. Z. Xu, H. Yuan, and Q. Liu, "Student performance prediction based on blended learning", IEEE Trans. Educ., vol. 64, no. 1, pp. 66-73, Feb. 2021.
[8]. Y. Dong, J. Ang and Z. Sun, "Designing path of SPOC blended teaching and learning mode in post-MOOC era", Proc. 10th Int. Conf. Educ. Inf. Technol. (ICEIT), pp. 24-28, Jan. 2021.
[9]. S. S. Urmi and K. A. Taher, "Integrating ICT in teaching and learning at university level focusing education 4.0", Proc. Int. Conf. Inf. Commun. Technol. Sustain. Develop. (ICICT4SD), pp. 300-304, Feb. 2021.
[10]. A. Sen and C. K. C. Leong, "Technology-enhanced learning" in Encyclopedia of Education and Information Technologies, Cham, Switzerland: Springer, pp. 1-8, 2020.