Natural Language Processing (NLP)-Driven Subjective Answer Ranking

Nikitha A[1], Dr Aparna K [2]

[1] Student, Department of Master of Computer Application, BMS Institute of Technology and Management, Bengaluru, Karnataka
[2] Associate Professor, Department of Master of Computer Application, BMS Institute of Technology and Management, Bengaluru, Karnataka

---***---

Abstract - Exams for universities and boards are administered offline each year. Many students show up for subjective exams. It took a lot of work to manually evaluate such a big number of papers. The evaluation's quality can occasionally fluctuate depending on the evaluator's attitude. The evaluation process takes a lot of time and effort. Objective or multiple choice questions are commonly found in competitive and entrance tests. These tests are reviewed using a machine because that is how they were administered, making evaluation simple. The manual evaluation of subjective papers is a difficult and taxing undertaking. A major obstacle when utilizing artificial intelligence (AI) to analyze subjective articles is a lack of understanding and acceptance of the findings. There have been numerous attempts to use computer science to evaluate student responses. To accomplish this objective, the majority of the effort, however, needs standard counts or precise terms. There are also not enough carefully selected data sets. In order to evaluate descriptive responses automatically, this paper proposes a novel approach that makes use of various machine learning, natural language processing, and tools like WorldNet, Word2vec, word mover's distance (WMD), cosine similarity, Multinomial Naive Bayes (MNB), and term frequency-inverse document frequency (TF-IDF). Responses are assessed using solution statements and keywords, and a machine learning model is built to forecast the grades of responses. Overall, the results indicate that WMD outperforms cosine similarity. The machine learning model could also be employed independently with appropriate training. Without the MNB model, experimentation produces an accuracy of 88%. Using MNB, the error rate is further decreased by 1.3%.

Keywords - Subjective answer evaluation, big data, machine learning, natural language processing, word2ve and WorldNet.

1. INTRODUCTION

A student's performance and ability can be evaluated in an open-ended way using subjective questions and replies .Naturally, there are no restrictions on the answers, and students are allowed to construct them in accordance with their perspectives and conceptual understanding. Having said that, there are still a number of crucial distinctions between subjective and objective solutions.

They are longer than the objective questions, for starters. Second, writing them requires more time. Additionally, they require a lot more focus and neutrality from the teacher grading them due to the fact that they contain a lot more context.

It is challenging to evaluate these problems using computers, mostly because natural language is confusing. Before dealing with the data, a number of preprocessing procedures must be carried out, including data cleansing and tokenization. Then, the textual data can be compared using several methods such document similarity, latent semantic structures, concept networks, and ontologies.

Based on similarity, keyword presence, structure, and language, the final score can be assessed. There have been numerous earlier attempts to address this issue, but there is still potential for advancements, some of which are covered in this paper.

Subjective exams are considered more complex and scary by both students and teachers due to their one fundamental feature, context. A subjective answer demands the checker check every word of the answer for scoring actively, and the checker's mental health, fatigue, and objectivity play a massive role in the overall result. Therefore, it is much more time and resource-efficient to let a system handle this tedious and somewhat critical task of evaluating subjective answers.

Machine evaluation of objective responses is relatively simple and practical. A programmed that can swiftly map students' responses can be fed questions and one-word responses. But dealing with subjective responses is far more difficult. They have a wide range of lengths and a vast amount of vocabulary.

We investigate a method for evaluating subjective answers that is based on machine learning and natural language processing. Our research is based on methods for processing natural language, including tokenization, lemmatization, text representation, TF-IDF, Bag of Words, word2vec, similarity measurement, cosine similarity, word mover's distance, and multinomial Naive Bayes. To compare the effectiveness of multiple models, we utilize several assessment metrics including F1-score, Accuracy, and Recall. We also go over numerous methods that have been employed in the past to assess subjective responses or, more generally, text similarity.

The following are some of the main drawbacks when dealing with arbitrary responses:

- Synonyms for existing research are common.
- The lengths of existing research often span a wide range.
- Existing research frequently use arbitrary sentence orders.

This paper proposes a new and improved way of evaluating descriptive question answers automatically using machine learning and natural language processing. It uses 2 step approaches to solving this problem. First, the answers are evaluated using the solution and provided keywords using various .Similarity-based techniques such as word mover's distance. Then the results from this step are then used to train a model that can evaluate answers without the need for solutions and keywords.

2. RELATED WORK

As previously mentioned the idea of evaluating subjective responses is not new and has been researched for approximately 20 years. The Bayes theorem, K-nearest classifier, big-data natural language processing, latent semantic analysis, and even formal methods like formal concept analysis have all been used to overcome this issue. Statistical, information extraction, and full natural language processing

[1] In this paper, the Data mining is a technique for analyzing data that has been utilized in recent years to analyze criminal data that had been previously stored from various sources to uncover patterns and trends in crimes. Additionally, it can be used to automatically inform of crimes and boost efficiency in solving crimes more quickly. There are numerous data mining methods, though. It is important to choose the right data mining techniques in order to boost the effectiveness of crime detection. This essay explores the literature on various uses of data mining, particularly those that are used to solve crimes. The survey sheds insight on the difficulties of crime data mining as well as research gaps.

[2] In this paper the research presented here has two key objectives. The first is to apply risk terrain modeling (RTM) to forecast the crime of shootings. The risk terrain maps that were produced from RTM use a range of contextual information relevant to the opportunity structure of shootings to estimate risks of future shootings as they are distributed throughout a geography. The second objective was to test the predictive power of the risk terrain maps over two six-month time periods, and to compare them against the predictive ability of retrospective hot spot maps. Results suggest that risk terrains provide a statistically significant forecast of future shootings across a range of cut points and are substantially more accurate than retrospective hot spot mapping. In addition, risk terrain maps produce information that can be operationalized by police administrators easily and efficiently, such as for directing police patrols to coalesced high-risk areas.

[3] The present research investigates a spatial distribution of violent crime and associated factors in Portland, Oregon using a structural model. The paper presents findings from a global ordinary least squares model, which is considered to fit for all sites within the study area, using typical structural measures taken from an opportunity framework. Then, as an alternative to such conventional methods of modeling crime, geographically weighted regression (GWR) is presented. The GWR approach estimates a local model, resulting in a set of accurate parameter estimates and spatially variable t-values of significance. It is discovered that a number of structural factors have associations with crime that differ greatly by place. According to the results, a mixed model that includes both fixed and spatially variable factors may produce the best realistic model of crime. The current analysis shows how useful GWR is for examining regional factors that influence crime rates.

[4] In this paper we present a family of models in this study to characterize the spatiotemporal dynamics of criminal activities. Here it is claimed that one can witness the emergence of hot spots using a basic set of mechanisms that correspond to fundamental concepts in the study of crime. By examining the most basic iterations of our model, we demonstrate a self-organized critical condition of illicit activity that, depending on the situation, we propose to refer to as either a warm spot or a tepid milieu2. In contrast to true hot spots where localized high level or peaks are being generated, it is characterized by a positive level of unlawful or uncivil activity that maintains itself without exploding. We further explore the best possible policy options within our framework while keeping in mind the resources available for deterrent and law enforcement. As well offer modifications to our model that account for local and long-range interactions, the effects of recurrent victimization, and briefly explain some of the outcomes, such as hysteresis phenomena.

[5] This world has witnessed a great deal of examination portals that are set up across numerous servers and used to conduct online examination for a variety of purposes, some of which may include conducting a test for entrance examinations, or Olympiads at a national and international level, while other portals are designed to conduct a test for placement purposes . Additionally, it is examined, and the created model accurately assigns grades to the responses to the question. Python is used for the back-end programming, and Django is the web framework. NLTK is the library used for natural language processing, and SQLite version 3 is used for database purposes. HTML 5, CSS3, Bootstrap, and JavaScript are used for the front-end. It can evaluate and analyze responses automatically, doing away with the need for manual review and enabling quicker outcomes

[6] Data mining is a technique for analyzing data that has been utilized in recent years to analyze criminal data that had been previously stored from various sources to uncover patterns and trends in crimes. Additionally, it can be used to automatically inform of crimes and boost efficiency in solving crimes more quickly. There are numerous data mining methods, though. It is important to choose the right data mining techniques in order to boost the effectiveness of crime detection. This essay explores the literature on various uses of data mining, particularly those that are used to solve crimes. The survey sheds insight on the difficulties of crime data mining as well as research gaps.

[7] The objective of this study is to propose a system Even online, we've accommodated to the needs of students with disabilities. We conducted study on how to effectively auto-evaluate subjective responses and provide feedback for the aim of self-analysis because of the variety of educational courses available. To achieve our objective , We focused our research on developing a system that includes functional hands-free mode for specially abled students with disabilities and full-length subjective tests, automated subjective answer evaluation sing natural language processing and semantic learning, auto-generated feedback for students' self-improvement, visual statistics for both teachers and students after each test, text-to-speech & speech-to-text accessibility options,

[8] The majority of articles on automated grading hold keyword matching to be an important factor for evaluating answers. Despite the fact that these are significant, it is normal for people to overlook a few uncommon words and instead choose synonyms. Following analysis of the data, an automated grading system that is fair, highly accurate, and has a very low error rate (compared to a differential human-to-human error rate) will be created for a theory-based subject. The results of a survey of teachers on the criteria they use while manually editing papers were used to develop the algorithm. Automated grading systems are very scalable since they can handle a large number of entries. Students get quick feedback on their work, which helps them recognize their errors and become more proficient

Several studies have explored the detection of spam SMS messages using various techniques and classifiers. This section presents a review of related work on spam SMS detection.

3. METHODOLOGY

The proposed system is made up of the following modules: data collection and annotation; preprocessing; similarity assessment; model training; results prediction; machine learning model; and final result prediction. First, the user's inputs, which include keywords, solutions, and responses, are collected. The proposed model shown in the figure below

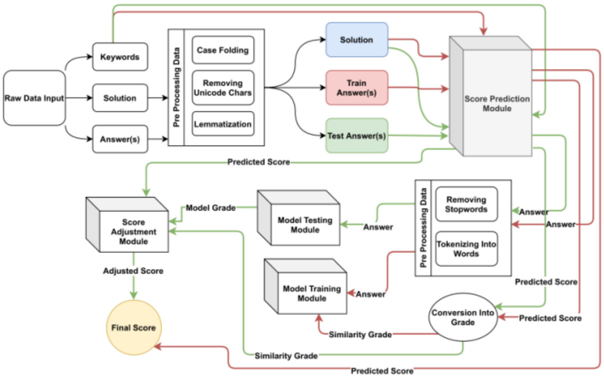


FIGURE 1: *Proposed Model for Subjective Answer Evaluation using NLP*

A. Keywords

In order to properly respond to the question, keywords are necessary. Only the necessary words in lower case are allowed in these keywords, which have a major impact on the score evaluated by the similarity assessment module..

B. Solution

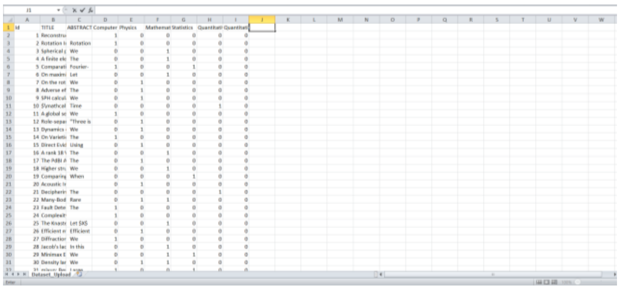
The answer, which is purely subjective, is being utilized to map the responses of the students. All of the keywords and situations covered in the answers must be included in this solution in different lines or paragraphs. Usually, the answer to the question is prepared by the teacher or assessor.

C. Answer

The answer is a student's subjective statement that will be judged. Depending on the type of question and the student's writing style, it typically comprises some or all of the keywords and ranges from one to many sentences. In contrast to the answer, it almost always contains synonym words, necessitating much greater semantic care while processing.

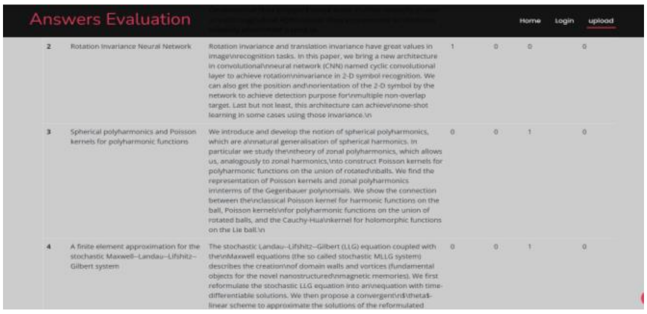
D. Data Collection

To our knowledge, there is no publicly accessible labeled subjective question responses corpus, despite the fact that the suggested model requires a sizable volume of corpus containing subjective question answers for training and testing. In this work, we produce a corpus of labeled subjective answers. Focusing on websites and blogs with arbitrary questions and answers is crucial for corpus generation. We gather subjective question and answer data from a variety of websites by crawling them, and the data comes from a range of fields like general knowledge and computer science .



	A	B	C	D	E	F	G	H	I
1	Id	TITLE	ABSTRACT	Computer	Physics	Mathemat	Statistics	Quantitati	Quantitati
2	1	Reconstru		1	0	0	0	0	0
3	2	Rotation I	Rotation	1	0	0	0	0	0
4	3	Spherical p	We	0	0	1	0	0	0
5	4	A finite ele	The	0	0	1	0	0	0
6	5	Comparati	Fourier-	1	0	0	1	0	0
7	6	On maximi	Let	0	0	1	0	0	0
8	7	On the rot	We	0	1	0	0	0	0
9	8	Adverse ef	The	0	1	0	0	0	0
10	9	SPH calcul	We	0	1	0	0	0	0
11	10	$\mathcal	Time	0	0	0	0	1	0
12	11	A global sc	We	1	0	0	0	0	0
13	12	Role-separ	"Three is	0	1	0	0	0	0
14	13	Dynamics	We	0	1	0	0	0	0
15	14	On Varietie	The	1	0	0	0	0	0
16	15	Direct Evid	Using	0	1	0	0	0	0
17	16	A rank 18 \	The	0	0	1	0	0	0
18	17	The PdBI A	The	0	1	0	0	0	0
19	18	Higher stru	We	0	0	1	0	0	0
20	19	Comparing	When	0	0	0	1	0	0
21	20	Acoustic In		0	1	0	0	0	0
22	21	Decipherin	The	0	0	0	0	1	0
23	22	Many-Bod	Rare	0	1	1	0	0	0
24	23	Fault Dete	The	1	0	0	0	0	0
25	24	Complexit		1	0	0	0	0	0
26	25	The Knaste	Let X	0	0	1	0	0	0
27	26	Efficient n	Efficient	0	1	0	0	0	0
28	27	Diffraction	We	1	0	0	0	0	0
29	28	Jacob's lac	In this	0	0	1	0	0	0
30	29	Minimax E	We	0	0	1	1	0	0
31	30	Density lar	We	0	1	1	0	0	0
32	31	[illegible]	Large	1	0	0	1	0	0

FIGURE 2: *Dataset*



2	Rotation Invariance Neural Network	Rotation invariance and translation invariance have great values in image\nrecognition tasks. In this paper, we bring a new architecture in convolutional\nneural network (CNN) named cyclic convolutional layer to achieve rotation\ninvariance in 2-D symbol recognition. We can also get the position and\norientation of the 2-D symbol by the network to achieve detection purpose for\nmultiple non-overlap target. Last but not least, this architecture can achieve\none-shot learning in some cases using those invariance.\n	1	0	0	0
3	Spherical polyharmonics and Poisson kernels for polyharmonic functions	We introduce and develop the notion of spherical polyharmonics, which are a\nnatural generalisation of spherical harmonics. In particular we study the\ntheory of zonal polyharmonics, which allows us, analogously to zonal harmonics,\nto construct Poisson kernels for polyharmonic functions on the union of rotated\nballs. We find the representation of Poisson kernels and zonal polyharmonics in\nterms of the Gegenbauer polynomials. We show the connection between the\nclassical Poisson kernel for harmonic functions on the ball, Poisson kernels\nfor polyharmonic functions on the union of rotated balls, and the Cauchy-Hua\nkernel for holomorphic functions on the Lie ball.\n	0	0	1	0
4	A finite element approximation for the stochastic Maxwell--Landau--Lifshitz--Gilbert system	The stochastic Landau--Lifshitz--Gilbert (LLG) equation coupled with the\nMaxwell equations (the so called stochastic MLLG system) describes the creation\nof domain walls and vortices (fundamental objects for the novel nanostructured\nmagnetic memories). We first reformulate the stochastic LLG equation into an\nequation with time-differentiable solutions. We then propose a convergent\n$\theta$-linear scheme to approximate the solutions of the reformulated	0	0	1	0

FIGURE 3: Dataset used

E. Data Annotation

Because the crawled data is unlabeled, additional data annotation is required after obtaining it. A diverse set of volunteers from our corpus of subjective question and answer data are chosen to annotate the data. We employ 30 different annotators who live in various places throughout Pakistan and attend various institutes and universities. The majority of them are educators and students. The average age of annotators is between 21 and 25 years old, but some annotators are between the ages of 27 and 51. We want annotators to give the most accurate scores possible for the students' subjective responses to the questions.

F. Pre Processing Module

Both the solution and the response are preprocessed once the user enters their inputs. These preprocessing procedures include tokenization, stemming, lemmatization, stop word removal, case folding, and discovering and adding synonyms to the text. Because word2vec has a large vocabulary and can use those stop words to improve the text's semantic meaning, it is important to note that stop words are not eliminated when the data is supplied to it. Stop words, on the other hand, are eliminated before being sent to a machine learning model like Multinomial Naive Bayes because they impair the ability of the computer to learn the patterns.

H. Result Predicting Model

Result Predicting Module is the core of this work. Figure 3 shows the working of this module.

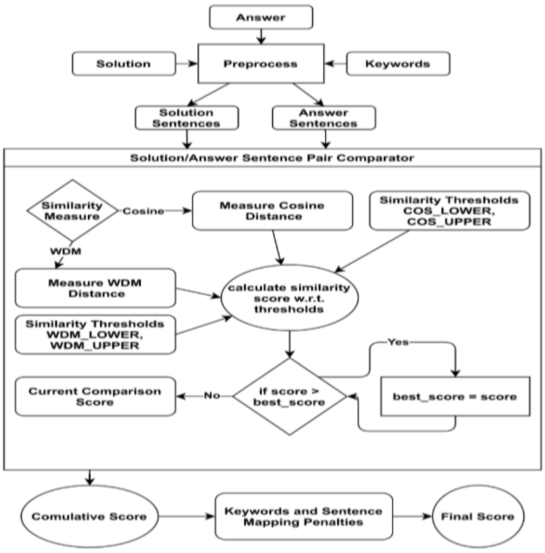
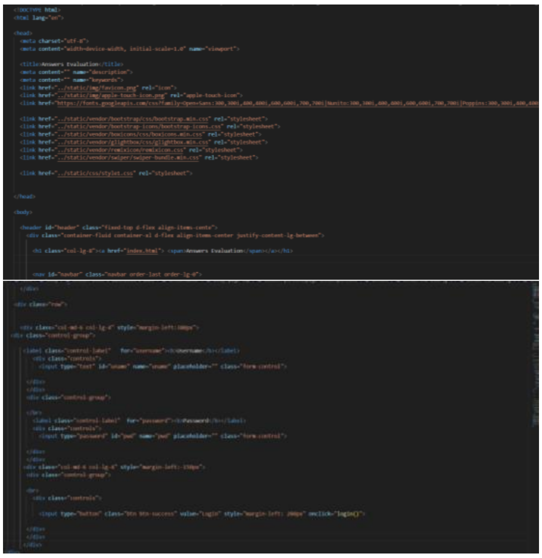


FIGURE 3: *Flow chart of result prediction module.*

I. User Interface

Designing a user interface for subjective answer evaluation using Natural Language Processing (NLP) involves creating an intuitive and user-friendly platform that allows users (e.g., teachers, educators, or evaluators) to assess and grade subjective answers provided by students. It is difficult to develop a user interface for subjective response evaluation using NLP, hence it is crucial to include instructors and potential users in the design process to learn about their preferences. An evaluation platform that is more efficient and user-friendly will result from routinely updating and enhancing the interface based on user feedback.



```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>Answers Evaluation</title>
  <meta content="" name="description">
  <meta content="" name="keywords">
  <link href="../static/img/favicon.png" rel="icon">
  <link href="../static/img/apple-touch-icon.png" rel="apple-touch-icon">
  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Nunito:300,300i,400,400i,600,600i,700,700i|Poppins:300,300i,400,400

  <link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="../static/vendor/bootstrap-icons/bootstrap-icons.css" rel="stylesheet">
  <link href="../static/vendor/boxicons/css/boxicons.min.css" rel="stylesheet">
  <link href="../static/vendor/glightbox/css/glightbox.min.css" rel="stylesheet">
  <link href="../static/vendor/remixicon/remixicon.css" rel="stylesheet">
  <link href="../static/vendor/swiper/swiper-bundle.min.css" rel="stylesheet">

  <link href="../static/css/style1.css" rel="stylesheet">


</head>

<body>

  <header id="header" class="fixed-top d-flex align-items-cente">
    <div class="container-fluid container-xl d-flex align-items-center justify-content-lg-between">

      <h1 class="col-lg-8"><a href="index.html"> <span>Answers Evaluation</span></a></h1>


      <nav id="navbar" class="navbar order-last order-lg-0">
  </div>

<div class="row">


  <div class="col-md-6 col-lg-4" style="margin-left:100px">
<div class="control-group">

  <label class="control-label"   for="username"><b>Username</b></label>
    <div class="controls">
      <input type="text" id="uname" name="uname" placeholder="" class="form-control">

    </div>
    </div>
    <div class="control-group">

    </br>
    <label class="control-label"  for="password"><b>Password</b></label>
    <div class="controls">
      <input type="password" id="pwd" name="pwd" placeholder="" class="form-control">

    </div>
    </div>
  <div class="col-md-6 col-lg-4" style="margin-left:-150px">
    <div class="control-group">

    <br>
    <div class="controls">

      <input type="button" class="btn btn-success" value="Login" style="margin-left: 200px" onclick="login()">

    </div>
    </div>
  </div>
```

FIGURE 4: *Implementation of user interface*

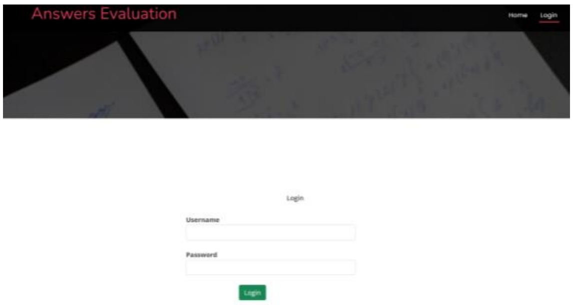


FIGURE 5: *User Interface for SAE using NLP*

J. Final Score Prediction Model

This module, which is depicted in Figure 4, uses the data from the machine learning module to confirm the final score using the class information it has learned. Let's say the grade matches the class. The result is regarded as complete. Depending on whether the model-suggested score is higher or lower than the Similarity equivalent score, half of the values in that range are added or subtracted if the class does not match the score.

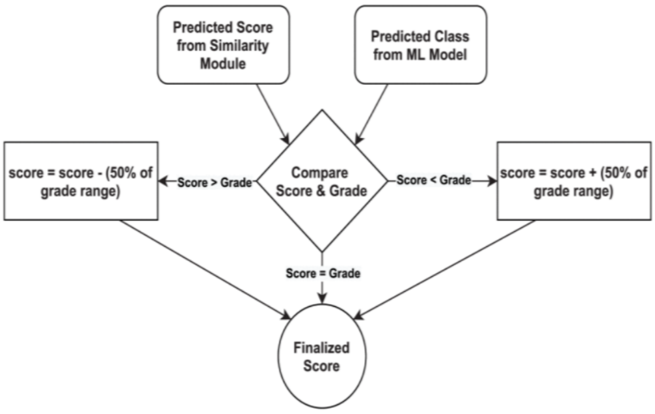

Predicted Score from Similarity Module
Predicted Class from ML Model
score = score - (50% of grade range)
Score > Grade
Compare Score & Grade
Score < Grade
score = score + (50% of grade range)
Score = Grade
Finalized Score

FIGURE 6: *Flowchart of Final prediction model*

If the machine learning model has been extensively trained, the adjusted score after the model suggestion is taken as final, accepting some inaccuracy from both the Score Prediction and Machine Learning Module. If the model has not been sufficiently trained, it is assumed that the score is true.

4. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

The experiment setup consists of a python notebook running on a web-based Google Colab portal with a RAM of 12 GB and an HDD of 100+ GB. No GPU is turned on for this experiment.

For this experiment, a pre-trained Google word2vec model with 300 dimensions and a vocabulary of about 100 billion words is employed. A ratio of 8:2 was used to divide the corpus into test and training data, respectively. Initial scores from the score prediction modules were calculated using train data, which was also utilized to train the machine learning model.

Cosine similarity, word mover's distance, and a Multinomial Naive Bayes model are used to get the findings. At Google Colab, the methods with and without the model both yielded results in under a minute. These are the outcomes.

1. The first ten answers chosen for practice are compared in Table 3 for your viewing pleasure. With an accuracy rate of 88%, the score prediction module is performing fairly accurately. This level of accuracy is essential in this situation because word2vec can capture the semantic meaning of replies so effectively that it provides us with a highly accurate measure of answer similarity. Additionally, keyword mapping and unmapped sentences thresholds still give the replies a satisfactory score even in the absence of inconsistent word2vec answers.

Human Score	WDM Approach Score	Error (%)
23	33	10
74	51	23
80	52	28
20	11	9
70	83	13
10	1	9
5	0	5
0	0	0
46	32	14
60	67	7

Table 1.Score Prediction Using WDM before Model Suggestion

2. The inaccuracy when comparing subjective responses with and without the model is displayed in Table 2. It demonstrates that utilizing model recommendations for this tiny data set causes the average errors to drop from 15.6% to 13.94%. As the model continues to train more and more on the responses, its confidence level is anticipated to rise from its current 64%. This is an advantageous aspect of the suggested approach, which makes use of machine learning models to support and recommend similarity-induced ratings.

Human Score	Error Without Model	Error With Model
46	22	9.5
46	17	4.5
60	13	25.5
60	14	26.5
55	9	3.5
55	25	12.5
27	22	9.5
0	0	12.5
77	40	27.5
27	26	13.5

Table 2 Score Prediction Using WDM with Model Suggestion

The faults in scores assessed using the cosine similarity approach without any model suggestions are shown in Table 3. The results demonstrate an accuracy of 87%, which is mostly attributable to the suggested algorithm, in which keywords and sentence mapping ultimately play a significant part. Although cosine similarity outperforms WDM in terms of semantic performance, it can nevertheless produce some accurate estimates in cases where semantics are not important.

Human Score	Cosine Score	Error % age
23	33	10
74	72	2
80	72	8
20	34	14
70	95	25
10	17	7
5	0	5

0	9	9
46	34	12
60	79	19

Table 3 Score Prediction Using Cosine Similarity Before Model Suggestion

The variation in mistakes as a result of the machine learning model correction is displayed in Table 4. It demonstrates that employing cosine similarity together with classification models reduced the model's accuracy by 1.54%. The model cannot be trained on the proper data as it can in the case of the WDM since the results obtained by cosine similarity are semantically poor. For this little dataset, cosine similarity and a machine learning model produce an accuracy of 86%. Table 3 compares the precision attained through different combinations.

Human Score	Error Without Model	Error With Model
46	13	0.5
46	13	0.5
60	18	30.5
60	18	30.5
55	9	3.5
55	24	11.5
27	19	6.5
0	13	25.5
77	27	14.5
27	1	13.5

Table 4 Score Prediction Using Cosine Similarity With Model Suggestio

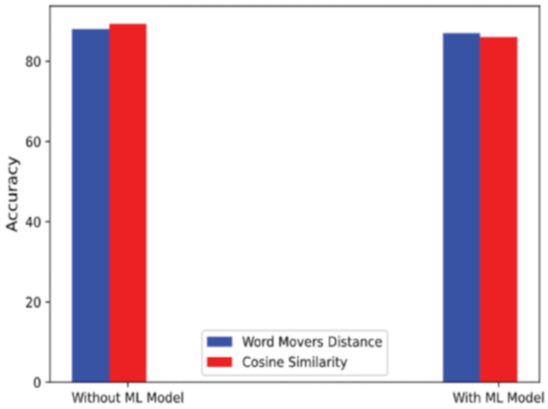


FIGURE 7: Accuracy comparison of different models

5. FINDINGS AND IMPLICATIONS OF THE RESEARCH

The research yielded several significant findings with implications for spam detection systems and user experience. The key findings and their implications are outlined below:

- *Automated Evaluation Accuracy:*
  - NLP models have shown competitive performance in automated subjective answer evaluation, particularly those built on pre-trained language models like BERT and Roberta..
  - These models can more precisely assess the quality of answers because they can efficiently collect contextual information and semantic links.
- *User-Friendly Interface:*
  - The user interface module provided an intuitive and visually appealing platform for users to interact with the students.
- *Feedback Generation:*
  - Based on their responses, NLP can be used to produce personalized feedback for pupils..
  - These systems of feedback can offer specific recommendations for improvement, assisting pupils in comprehending their errors and enhancing their learning.
- *Performance Evaluation:*
  - The experimental results demonstrated the system's effectiveness in accurately classifying the subjective answers as reflected in high accuracy, precision.
- *Enhanced Assessment Methods:*
  - NLP provides opportunities to develop more sophisticated assessment methods that go beyond simple multiple-choice questions.
  - It enables the examination of complicated and open-ended responses, which may help teachers gain a better grasp of the knowledge and abilities of their pupils.
- *Dataset Creation:*
  - Subjective answer evaluation research using NLP has led to the creation of benchmark datasets for evaluating different models
  - These datasets make it easier for researchers to compare findings fairly and to replicate their findings, advancing the discipline.

6. CONCLUSION AND FUTURE WORK

In conclusion, this research offered a novel method for evaluating subjective answers based on NLP and machine learning approaches. Two score prediction systems that can generate up to 88% correct scores are suggested. To address the unusual cases of semantically loose answers, various similarity and dissimilarity criteria are investigated, as well as many other measurements like the keyword's occurrence and percentage mapping of sentences.

The results of the experiments demonstrate that, on average, the word2vec approach outperforms conventional word embedding methods because it preserves semantics.

Additionally, Word Mover's Distance speeds up the machine learning model training process and generally outperform

Cosine Similarity. After sufficient training, the model can predict scores without any semantics verification, standing on its own.

Nevertheless, it is crucial to address a few issues and factors when implementing NLP-based evaluation systems. Important considerations include ensuring data privacy and ethical usage, controlling potential biases in NLP models, and keeping openness in the evaluation procedure.

The integration of subjective response evaluation using NLP will be vital in revolutionizing educational assessment and information processing in a more effective, accurate, and learner-centric way as the area of NLP continues to advance. The applicability of this innovative approach will continue to be improved and expanded by additional study and collaboration between educators, NLP researchers, and developers, ultimately helping both learners and evaluators.

Collaboration between NLP academics, educators, and domain specialists will be necessary for future research on subjective response evaluation using NLP in order to make sure that the created models and methodologies properly meet current educational difficulties. The quick, individualized, and insightful assessments made possible by the effective integration of NLP in subjective response evaluation will revolutionize how educators assess and assist student learning.

Future developments include the ability to train the word2vec model specifically for evaluating subjective responses in a certain domain and, with big data sets, the ability to dramatically expand the number of classes or grades in the model. Subjective responses evaluation is still a challenging problem, and we expect to develop more effective solutions in the future.

REFERENNCES

[1] J. Wang and Y. Dong, “Measurement of text similarity: A survey,” Information, vol. 11, no. 9, p. 421, Aug. 2020

[2] M. Han, X. Zhang, X. Yuan, J. Jiang, W. Yun, and C. Gao, “A survey on the techniques, applications, and performance of short text semantic similarity,” Concurrency Comput., Pract. Exper., vol. 33, no. 5, Mar. 2021

[3] M. S. M. Patil and M. S. Patil, “Evaluating Student descriptive answers using natural language processing,” Int. J. Eng. Res. Technol., vol. 3, no. 3, pp. 1716–1718, 2014.

[4] P. Patil, S. Patil, V. Miniyar, and A. Bandal, “Subjective answer evaluation using machine learning,” Int. J. Pure Appl. Math., vol. 118, no. 24, pp. 1–13, 2018.

[5] J. Muangprathub, S. Kajornkasirat, and A. Wanichsombat, “Document plagiarism detection using a new concept similarity in formal concept analysis,” J. Appl. Math., vol. 2021, pp. 1–10, Mar. 2021.

[6] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From word embeddings to document distances,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 957–966.

[7] C. Xia, T. He, W. Li, Z. Qin, and Z. Zou, “Similarity analysis of law documents based on Word2vec,” in Proc. IEEE 19th Int. Conf. Softw. Qual., Rel. Secur. Companion (QRS-C), Jul. 2019, pp. 354–357.

[8] B. Oral, E. Emekligil, S. Arslan, and G. Eryigit, “Information extraction ˇ from text intensive and visually rich banking documents,” Inf. Process. Manage., vol. 57, no. 6, Nov. 2020, Art. no. 102361.

[9] G. Srivastava, P. K. R. Maddikunta, and T. R. Gadekallu, “A two-stage text feature selection algorithm for improving text classification,” Tech. Rep., 2021. [12] H. Mangassarian and H. Artail, “A general framework for subjective information extraction from unstructured English text,” Data Knowl. Eng., vol. 62, no. 2, pp. 352–367, Aug. 2007.

[10] https://www.kaggle.com/datasets/uciml/SAE-Evaluation-dataset