

AI BASED E-ASSESSMENT SYSTEM

SALONI KADAM¹, PRIYANKA TARACHANDANI², PRAJAKTA VETAL³, MRS.
CHARUSHEELA NEHETE⁴

1,2,3 Student, Dept. of Information Technology, VESIT, Maharashtra, India.

4 Assistant Professor, Dept. of Information Technology, VESIT, Maharashtra, India.

ABSTRACT

We have seen that a number of students apply for various examinations which may be institutional, non-institutional or even competitive. The competitive exams mostly have objective or multiple-choice questions(mcqs). The automation of scoring of subjective or descriptive answers is a need considered nowadays. Our objective is to design a system that can evaluate subjective answers as good as a human being. The paper presents an approach which focuses on the inference process required for development of such system. The assessment is based on the similarity measures between the answers. Based on the accuracy and importance of the features, weights are assigned to those features which help in calculating the result. Our system is able to assess brief answers efficiently. It aims at faster and flexible evaluation. Implementing such a system would be a step forward in the educational field.

KEYWORDS: E-assessment, Subjective Answer, Cosine Similarity, WMD, Wordnet, Grammar, Jaccard Similarity.

1. INTRODUCTION

Examination is a test of a person's knowledge in a particular area which is either subjective or objective or both. Usually, competitive examinations consist of multiple-choice questions or mcqs. Automatic evaluation of the objective exams is beneficial as it saves time, provides efficiency, reduces usage of resources. However, this automated evaluation technique is only for the objective exams and not for the subjective ones. Subjective answer sheet checking is one of the huge administrative tasks for any education institute. In this examination process, candidates need to write answers, an examiner collects those answer sheets and submits them to authority for further checking process. This process involves 3 levels of paper checking: -

- First Level Paper Checker,
- First Level Moderation,
- Second Level Moderation.

So, the amount of pressure education systems and teachers hold is understandable as the number of answer sheets to evaluate is too large. So, there is a necessity for an approach which will reduce the usage of resources by providing an approach which will automatically evaluate the answers given by students and provide results. Such a system is the goal of this paper. We have developed an E-assessment system that checks the answer sheet of the student and provides marks to the same. The system consists of an algorithm that compares student's answer against the reference answer of faculty using the similarity features. Student gets marks based on the closeness of his answer to the reference answer. Both the answers need not be exactly the same or word to word. This approach can be a quick and easy way for the examiners by reducing their workload.

2. LITERATURE SURVEY

[1] Sheeba Praveen, "An Approach to Evaluate Subjective Questions for Online Examination System", International Journal of Innovative Research in Computer and Communication Engineering. Vol. 2, Issue 11, November 2014. This system solves the problem of deducing knowledge represented by partially or grammatically incorrect sentences, and will translate the meaning conveyed by the student in different forms and sentences, propose a normalized plan of action for grading the answers, ways to interpret the mathematical formulas and expressions however the system is limited to non-mathematical subjects only.

[2] Amarjeet Kaur, M Sasikumar, Shikha Nema, Sanjay Pawar(2013), "Algorithm for Automatic Evaluation of Single Sentence Descriptive Answer", International Journal of Inventive Engineering and Sciences (IJIES). Single sentence descriptive answers which are grammatically correct and have no spelling errors is considered as the text. Their approach is to represent standard answers in the form of graphs and then comparing it by applying similarity measures for the allocation of marks.

[3] Aditi Tulaskar¹, Aishwarya Thengal², Kamlesh Koyande³, "Subjective Answer Evaluation System", Department of Information Technology Vidyankar Institute of Technology, Mumbai, India. The proposed system will allot the marks according to the percentage of accuracy present in the answer. This is a software system in which students will be authenticated by using student login. After the authentication, students will be provided with the questions. The proposed system is designed to evaluate answers for five students providing five different answers. The standard answer is stored in the database with the keywords, meaning and the description of that answer. Then each answer is evaluated by matching the keywords as well as its synonyms with the standard answer. It will also check the grammar and spellings of the words. After the evaluation, the answer is graded depending on the correctness of it.

[4] Asmita Dhokrat, Gite Hanumant R., C. Namrata Mahender(2017), "Automated Answering for Subjective Examination", (IJCA). The proposed system has two login facilities. The user login is the login allocated for the students. As soon as you click the student login button you will be asked to enter login id and password. The system

will check for the id and automatically display students' name, email id and phone number for verification. The user login writes answers with respect to the question uploaded. The system will show marks scored as soon as you enter the next button. The admin login will let the teachers login. In admin login each user will have his own password and id through which they can login in to the system. The admin can add/subtract questions, check for students' marks and so on. Just like the teachers can do manually.

[5] Merien Mathew, Ankit Chavan, Siddharth Baikaar, "ONLINE SUBJECTIVE ANSWER CHECKER", International Journal of Scientific & Engineering Research, 2017. The original answer is required to be stored in the system. This is done by the admin. The admin may insert any number of questions and respective subjective answers. The answers are stored as notepad files. When a user takes the test he/she is provided with questions and area to type the answer. Once the user enters his/her answers, these answers are then compared by the system with the original answers written in the database and marks are allocated accordingly. Both the answers need not be exactly the same or word to word. The system consists of built AI sensors that verify answers and allocate marks accordingly as good as a human being.

3. IMPLEMENTATION

We have developed the process of Subjective Answer Evaluation which includes one-word, short answers. System allows teacher to upload questions with reference answer and a unique question paper code. Student can access the questions using the question paper code provided, this way student authentication is also achieved. For Realtime data storage, Firebase is used. All the answers go through preprocessing case normalization, stop words are removed to obtain important terms and keywords. Student answers are checked against reference answer using similarity matching features which are discussed below.

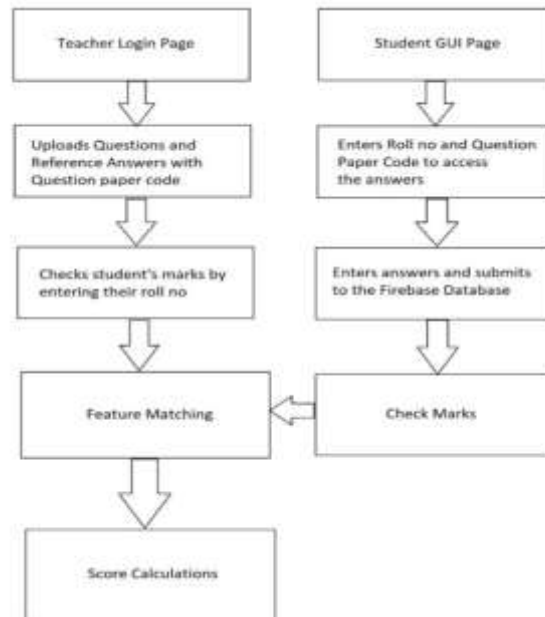


Fig.1. Flow diagram

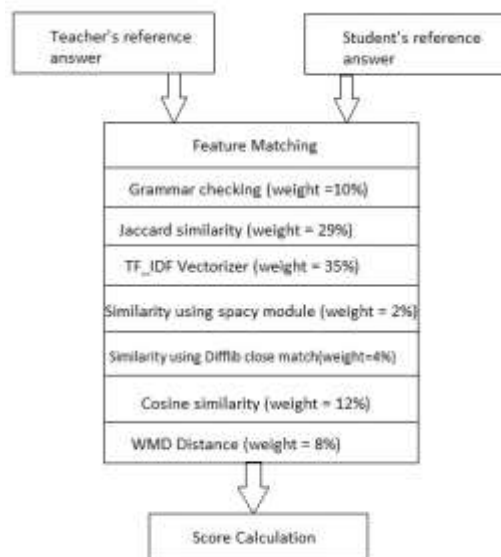


Fig.2. Feature mapping diagram

3.1 FEATURE MATCHING SIMILARITY METHODS:

1. Spacy Similarity
2. Tfidf Vectorizer

3. Difflib Similarity
4. Jaccard Similarity
5. Grammar check
6. Cosine Similarity
7. Word Mover Distance (WMD)

3.1.1 SPACY SIMILARITY

SpaCy is a parameter which ranges between 0 to 1 and tells us semantically, how close two words are. By finding similarity between word vectors, this can be done. It is one of the NLP libraries which provides a simple method for finding similarity. It basically supports two models: - word vector and context-sensitive tensor.

3.1.2 TERM FREQUENCY INVERSE DOCUMENT FREQUENCY

TFIDF or tf-idf, stands for term frequency-inverse document frequency, it is a numerical statistic which reflects how important a word is to a document in a collection or corpus. The significance of a document or the word is directly proportional to the number of times a word appears in the document. tf-idf is one of the most popular schemes for term-weighting today.

It was mainly invented for document search and information retrieval. So, words that are common in every document, such as- this, what, and if, rank low even though they may appear many times, since they don't have much meaning to that document.

For example, if the word *intelligence* appears many times in a document, while not appearing many times in others, it probably means that it's very relevant. For example, what we're doing is trying to find out which topics are more relevant or important in a document so here the word *intelligence* will be of utmost importance.

Term Frequency (tf): It gives us the frequency of the word in each document in the corpus. It is the ratio of the number of times the word appears in a document compared to the total number of words in that document. Tf increases proportionally as the number of occurrences of that word within the document increases. Each and every document has its own term frequency.

$TF(t) = (\text{Number of times } t \text{ appears in a } d) / (\text{Total number of } t \text{ in the } d).$

Inverse Document Frequency (idf): idf is used to calculate the weight of infrequent words across all documents in the corpus. I.e., words which do not appear often in the corpus will have a high IDF score.

$IDF(t) = \log_e(\text{Total number of } d / \text{Number of } d \text{ with } t \text{ in it}).$

where, 't' stands for term and 'd' stands for document.

For example, Sentence 1: SQL injection is a code injection attack.

Sentence 2: SQL injection is a query injection technique.

Here, each sentence is considered as a separate document.

The TF-IDF for the above two documents, which represent our corpus is given below.

Table 1. TF-IDF corpus

Word	TF (A)	TF (B)	IDF	TF*IDF F (A)	TF*IDF F (B)
SQL	1/7	1/7	$\log(2/2) = 0$	0	0
injection	1/7	1/7	$\log(2/2) = 0$	0	0
is	1/7	1/7	$\log(2/2) = 0$	0	0
a	1/7	1/7	$\log(2/2) = 0$	0	0
code	1/7	0	$\log(2/1) = 0.3$	0.043	0
query	0	1/7	$\log(2/1) = 0.3$	0	0.043
injection	1/7	1/7	$\log(2/2) = 0$	0	0
attack	1/7	0	$\log(2/1) = 0.3$	0.043	0
technique	0	1/7	$\log(2/1) = 0.3$	0	0.043

From the above table, we see that TF-IDF of common words is zero, which shows that they are not significant, while others are significant. The TF-IDF of “code”, “query”, “attack” and “technique” are not zero. These words have more significance.

3.1.3 DIFFLIB SIMILARITY

The difflib is a python module whose tools are used for computing and working with differences between sequences. It offers a way to compare multi-line strings and entire lists of words. `get_close_matches(word, possibilities, n, cutoff)` function works in Python which returns the best 'good enough' matches. It accepts four parameters in which n, cutoff is optional. It is used for comparing pairs of sequences which are given as input.

3.1.4 JACCARD SIMILARITY

Jaccard Similarity measures similarity between finite sample sets. It is also called an intersection over union and is defined as the size of intersection divided by the size of union of two sets.

$$\text{Jaccard Similarity} = J(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

The range is between 0 to 1. If the score is 1, then they are identical and if there is no common word between the first sentence and the last sentence then the score is 0.

For example, Teacher's answer : SQL Injection is a code injection attack where an attacker manipulates the data being sent to the server to execute malicious SQL queries.

Student answer : SQL injection is an attack that involves code injection in which the attacker changes the data that is being sent to the server to run SQL queries that are dangerous.

Jaccard Similarity Measure: 0.45

3.1.5 GRAMMAR CHECK

To check and detect grammatical mistakes and spelling errors Grammar Bot API is used in our system. When the text is sent to Grammar Bot's API, it returns a list of potential grammar and spelling errors. System takes count of grammatical mistakes made by the student and if mistakes exceed the acceptable limit, it gives a remark about the mistake with the result to help student improve next time.

3.1.6 COSINE SIMILARITY

Cosine similarity is a standard point of reference to measure how similar the documents are irrespective of their size. It measures the cosine of the angle between two vectors which are projected in multi-dimensional space. To measure similarity between the documents we simply use the approach of counting the maximum number of common words between them. But this approach has a major drawback when the size of a document increases, the number of common words tend to increase even if the documents talk about different topics. In this case cosine similarity is very advantageous. In multi-dimensional space, each dimension corresponds to a word in the document. It captures the orientation of the documents and not the magnitude which means if two similar documents are far apart by Euclidean distance, can still have a smaller angle between them. Smaller the angle, higher the similarity.

For example, Teacher's answer: SQL Injection is a code injection attack where an attacker manipulates the data being sent to the server to execute malicious SQL queries.

Student answer: SQL injection is an attack that involves code injection in which the attacker changes the data that is being sent to the server to run SQL queries that are dangerous.

$$\text{Cosine similarity: } \begin{bmatrix} 1 & 0.71 \\ 0.71 & 1 \end{bmatrix}$$

3.1.7 WORD MOVER DISTANCE

The measure of similarity between two blocks of text can be used as a good measure for evaluation of answers. Ideally statically based algorithms like LSA, BLEU etc. can capture semantic relation between two documents. So when two documents have no word in common their Euclidean distance would be maximum. Word mover's distance (WMD) is used to face this problem. It adapts the earth mover's distance to the space of documents. At a high abstraction, the WMD is the minimum distance required to transport the words from one document to another. We assume that we are given a word embedding matrix (word2vec). We use the Word Mover Distance (WMD) problem on a matrix of pairwise distances between each state vector of the model and student answers. If a word 'wi' appears 'fi' times in a document, its weight is calculated where 'n' is the number of unique words in the document. The higher its weight, the more important the word is. The dissimilarity between word 'wi' in student answer and word 'wj' in model answer can be computed as where 'xi' and 'xj' are the embeddings of the words 'wi' and 'wj', respectively.

3.2 SCORE GENERATION

For every feature we have assigned weights based on their accuracy and importance in the evaluation process. Marks for the answer are reliant on the percentage of keywords match, grammar, synonyms etc. Hence if a student writes an answer missing any of these, marks will be deducted according to their weightage in evaluation.

Table 2. Score Generation Criteria

Features	Weightage allotted
Grammar check	10%
Jaccard Similarity	29%
TF-IDF vectorization	35%
Spacy	2%
difflib	4%
Cosine similarity	12%

WMD distance	8%
--------------	----

In the system each question is considered of 10 marks so the accuracy obtained from the features for a question is scaled out of 10. This is how marks for all questions are calculated and finally added to get the result. Result also includes the remark for grammar mistakes if done beyond an acceptable level.

4. RESULT

The Teacher GUI is as shown below, where the teacher uploads questions and reference answers for the student with unique question paper code. Also, she can click on ‘check marks’ and check marks of students.

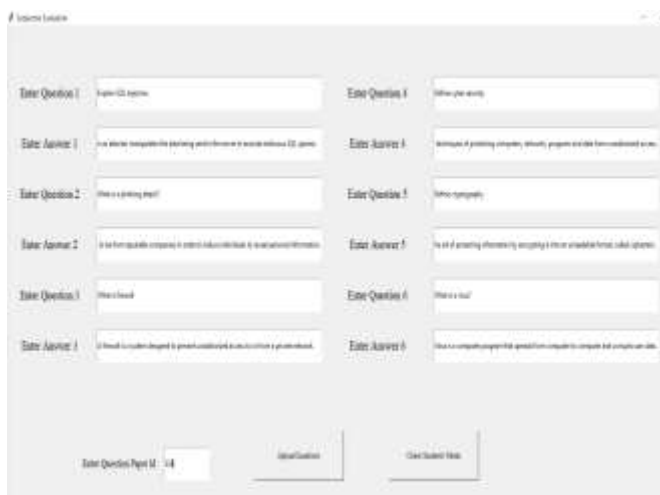


Fig.2. Teacher GUI-1

After clicking on the ‘check marks’ button, the page below appears.



Fig.3. Teacher GUI-2

In Students GUI, when the student enters Roll No, Question paper ID and clicks on ‘Get question’, the question and the answer space will be displayed. The student then Submits the answers and Checks marks.



Fig.4. Student GUI

When the student clicks on ‘Check marks’, a result page is displayed wherein they can see the marks of each question and also the Total marks scored.

Question 1 marks	7
Question 2 marks	5
Question 3 marks	7
Question 4 marks	6
Question 5 marks	5
Question 6 marks	7
Total Marks	37

Fig.5. Result Page

5. CONCLUSION

The E-Assessment System would be beneficial for the universities, schools and colleges for academic purpose by providing ease to faculties and the examination evaluation cell. Many educational institutes conduct their examinations online, but these exams only contain multiple choice questions which only tests the student's aptitude, and fail to test the conceptual knowledge a student or learner must possess. Therefore, descriptive answers must be included in online examinations. Our proposed system evaluates the answer based on some similarity features. By judging against the reference answer marks are allocated. Highest marks are gained if the student achieves maximum closeness to the reference answer. Hence the proposed system could be of great utility to the educators whenever they need to take a quick test for revision purposes, as it saves time and the trouble of evaluating the bundle of papers.

ACKNOWLEDGEMENT

We are thankful and would like to express our sincere gratitude towards Prof Mrs. Charusheela Nehete for her continuous guidance, support and encouragement in making this project possible. Our sincere thanks to all the Lecturers who helped us in our endeavor, gave valuable advice and thus made our journey possible. Last but not the least we would also like to thank our family and friends for their constant support, motivation and belief in us.

6. REFERENCE

[1] Sheeba Praveen, “An Approach to Evaluate Subjective Questions for Online Examination System”, International Journal of Innovative Research in Computer and Communication Engineering. Vol. 2, Issue 11, November 2014

[2] Amarjeet Kaur, M Sasikumar, Shikha Nema, Sanjay Pawar(2013), “Algorithm for Automatic Evaluation of Single Sentence Descriptive Answer”, International Journal of Inventive Engineering and Sciences (IJIES).

[3] Aditi Tulaskar¹, Aishwarya Thengal², Kamlesh Koyande³, “Subjective Answer Evaluation System”, Department of Information Technology Vidyalkar Institute of Technology, Mumbai, India.

[4] Asmita Dhokrat, Gite Hanumant R., C. Namrata Mahender(2017), “Automated Answering for Subjective Examination”, (IJCA).

[5] Merien Mathew, Ankit Chavan, Siddharth Baikar, “ONLINE SUBJECTIVE ANSWER CHECKER”, International Journal of Scientific & Engineering Research, 2017.