

ASSESS – Automated subjective answer evaluation using Semantic Learning

Prof. Era Johri
K J Somaiya College of Engineering
Mumbai, India
erajohri@somaiya.edu

Prem Chandak
K J Somaiya College of Engineering
Mumbai, India
prem.chandak@somaiya.edu

Nidhi Dedhia
K J Somaiya College of Engineering
Mumbai, India
npd@somaiya.edu

Hunain Adhikari
K J Somaiya College of Engineering
Mumbai, India
hunain.a@somaiya.edu

Kunal Bohra
K J Somaiya College of Engineering
Mumbai, India
kunal.bohra@somaiya.edu

Abstract - Drift in the digitization of education is a prime concern at present to enable quality education to every individual. Now, there are no geographical barriers to the availability of education and evaluation. Imparting education is easier through digitization but inconvenient to evaluate. In this paper, we propose 'ASSESS', a system where the evaluation of subjective answers for an examination becomes easier and convenient. We have even catered to the requirements of specially-abled students online. The diversity in educational courses encouraged us to research how we can efficiently auto-evaluate subjective answers and provide feedback for the purpose of self-analysis. During the pandemic of COVID-19, most of the colleges and organizations shifted to the online mode of examinations. These examinations only had MCQs or objective questions which can be easily assessed by the online system. Since such systems can only be used for the evaluation of objective questions, the subjective questions pose a great challenge. In this paper, we directed our research to propose a system that gives features like full-length subjective tests, automated subjective answer evaluation using natural language processing and semantic learning, auto-generated feedback for self-improvement of the students, visual statistics for both teacher and student after each test, text-to-speech & speech-to-text accessibility options and a fully functional hands-free mode for the specially-abled students with disabilities like sluggish typing, poor eyesight, and amputated hands. Since everything will be automated from the evaluation of the answers to providing feedback, there will be minimal stress on the assessors.

Keywords – Automated answer evaluation, education, keyword matching, Semantic Learning, similarity score, subjective answer assessment, Universal Sentence Encoder

I. INTRODUCTION

In the quarantine situation amidst the COVID-19 pandemic, most of the colleges and institutions had shifted to the online mode of examinations. But these examinations mostly consist of MCQs or objective questions which can be easily corrected by the system on which the test is being conducted. Since such systems can only be used for objective type question evaluation, the questions that require more than just a radio or a checkbox click, for example, full-length subjective answers which may need manual intervention for evaluation purpose, pose a great challenge in education. Even the public cloud offering applications like online classrooms have not yet embraced subjective answer evaluations. This limits the scope of examination and evaluation conducted online.

Different examinations that need subjective type evaluation to test the conceptual knowledge of the students cannot use such online systems where only MCQ-based questions are auto-evaluated. Traditionally, students are expected to write the answers on a plain sheet, scan them and upload them for evaluation online. Another scenario is where universities or colleges conduct their exams offline and students attempt them on paper. A dedicated staff is required to scan these sheets and upload them to the system. Then the assessors from the different regions or colleges access these scanned images and manually review them. It is a tiresome and hectic process of evaluation of these answer sheets. Assume, a 3-hour long paper, correcting it would require at least 10-15 minutes offline and approximately 25-30 minutes online because of the several issues like limited network bandwidth, the loading time of answer sheets, the number of concurrent users supported by the system, constant stress to eyes caused by viewing the computer screen at a stretch. This means for 80 students; it would be around 20 hours or even more for that matter. Well, time is not the only hurdle here, vague or improper scanning is oftentimes a barrier in understanding what the student wants to portray and it takes a toll on eyes evaluating the paper. Also, since humans are prone to error, there are chances when almost the same answer could fetch different grades for different students. Thus, developing a system to tackle all these problems arisen by the digitalisation is a much-needed requisite.

In our proposed system, ASSESS, we are going to cater to this need of abolishing the entire process of manual assessment of subjective answers and make the evaluations completely automated. There would no longer be any need to scan the pages of the answer sheets and upload them to the system. The students can themselves submit their answers via typing or making oral submissions on the ASSESS system interface and this record of their submission will be documented in the database. Hence, the amount of pressure reduces to a great extent on all those who are directly or indirectly involved in the process of evaluation.

In the existing systems, both subjective and objective answers can be submitted or uploaded, but it does not incorporate evaluation and feedback. There is no concept of providing feedback on the evaluated answers to the students so that they can improve on the missed points in future. Self-analysis is of the utmost importance when it comes to

learning from your own mistakes and this is a major pitfall in the existing systems.

All these shortcomings in the current systems have been taken care of in our proposed system named 'ASSESS' which fills the gaps and helps in developing a great environment for both students and teachers, to learn, test and improve. It is a single platform for both students and teachers for examination, evaluation and feedback. ASSESS is a well-organised and easy-to-use system where all the test records and feedback can be found in the respective sections. It has a separate user interface for teachers and students with minimal clicks required to complete the tasks. It has a dashboard consisting of charts to analyse the performance of individual students or as a group. All these features in our proposed system not only makes it different and unique from the existing ones but also convenient and accessible to use at the same time.

II. LITERATURE REVIEW

Several applications are developed for the evaluation of subjective answers but consist of certain loopholes which we tried to tackle in our system 'ASSESS'. Systems under consideration are as follows-

The Automated Grading System [1] focuses on a knowledge-oriented approach rather than just a keyword-matching hit ratio. They use ontology to map domains related to a given keyword. LSA and dictionary mapping ensures that relevant answers get marks. Grammar and syntax are also checked however it will not affect the overall score of the response, provided the concept is properly explained. The main drawback of this system is that it does not provide feedback or give any sort of information if the answer of the student is wrong and hence the student is unable to know where he/she goes wrong while writing the answer.

ApTeSa [2] is a system that works in two modes. First is a semi-automated mode which gives the teachers an option to reassess an answer and update the results if they think the system has not evaluated properly. The other mode is a fully automated mode where everything is done by the system solely and there is no involvement of teachers. The semi-automated mode gives better results than the fully automated mode as a manual evaluation is done after the automated evaluation. But in this mode, the speed of evaluating the answers gets hampered if compared to the fully automated one. The hindrance in this system is that it requires the involvement of teachers in a semi-automated mode whereas the fully automated one does not give efficient results.

One of the systems [3] used different machine learning techniques to try and capture the latent relationship between the words. The techniques included were Latent Semantic Analysis, Generalized Latent Semantic Analysis, Maximum Entropy technique and BiLingual Evaluation Understudy. This system measures the relationship between two words, the words and the concepts using Ontology for the evaluation of answers. The techniques discussed and implemented in this paper show a high correlation (up to 90%) with human

performance. It was seen that using ML techniques with Ontology gives satisfactory results due to holistic evaluation. This system might give satisfactory output in terms of the evaluation of the answer, but it lacks a feedback module that can tell students about their mistakes and where to improve.

In the case-based model [4], the framework has the provisions for the reward and penalty schemes. In the case of a reward scheme, extra valid points given by the students earn them bonus marks as rewards. By incremental up-gradation of the question case-base with these extra answer-points, the examiner can incorporate automatic fairness in the checking procedure. The penalty scheme comes into the picture if the neighbouring students adopt some unfair means while answering the questions. This has been detected by using a neighbourhood graph. The degree of penalty is then decided based on the degree of similarity between the adjoining answers. The main question bank, as well as the model answer points, are all maintained using case-based reasoning strategies. This system takes care of unfair means and rewarding bonus marks but fails to make it accessible for specially-abled people or taking answers orally. Also, this system just like the others does not highlight any mistakes of the students or provide feedback if they go wrong.

Another system [5] that we studied gives more attention to the answer's content rather than its grammar, spelling and vocabulary. They considered some deep learning approaches like Character level CNN, Word level CNN, Word level bi-LSTM and BERT for short answer scoring. BERT performed much better than the other models that were in comparison. It also performed well even if the answers were paraphrased. This is because of BERT's deep contextual representations conditioned in both directions which allow the context to be preserved despite the change in structure. The execution of BERT was destitute compared to its performance on the test information. The system is just limited to providing marks to the user entered answer.

The above mentioned are few systems proposed/developed for the automatic subjective answer evaluation which helped the students and teachers with the process but they are yet incomplete. None of the previously mentioned systems provides personalized answer-by-answer feedback for self-evaluation and improvement of the students. These systems are inaccessible by specially-abled people with physical disabilities like partial vision, amputated hands or slow typing speed. There is no single platform for both students and teachers to easily manage online examinations as well as self-evaluation. No system provides any visual representation of the user statistics that help the users to easily visualize and gain valuable insights on their progress. The scope of all these systems is limited till providing marks to the user entered answer. None of them highlights the missing points or errors in the answers.

III. METHODOLOGY

The system proposed in this paper makes use of semantic learning at its core to find the true meaning of the answers. Semantic learning is a learning technique that builds a knowledge graph that is connected with the semantic network of previously known knowledge. In this way, the context of

the previous sentences is stored and used for the sentences ahead. We have used Google's Universal Sentence Encoder algorithm to generate sentence embeddings. These are high dimensional vectors that represent the meaning of the sentences in the form of numbers. In this system, we use a model answer which is the expected answer and compare it with the user's answer to know how similar both these answers are. This answer is taken from the teacher while creating the test and is stored in the database. At the core, we use the Universal Sentence Encoder to encode both the answers into vectors. Then we calculate the distance between the vectors by using cosine distance which gives us a fair idea of how similar or far both the answers are. The values range between -1 (least similar) to 1 (most similar). Figure 1 shows the flow of the assessment algorithm which starts from the user answering the question. Then both the user answer and the model answer are fed to the evaluation algorithm which starts with pre-processing and gives 'Total score' and 'Feedback Report' as the final output. Each step is discussed in detail in further sub-sections.

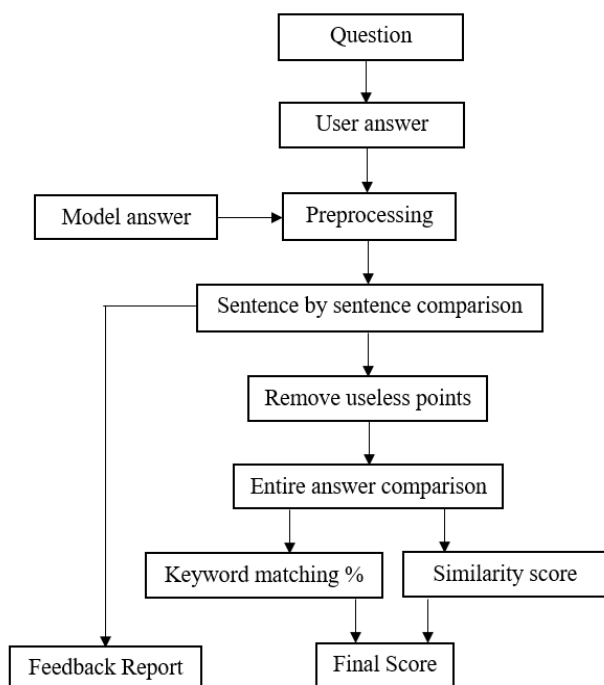


Fig. 1. Assessment Algorithm

A. Pre-processing

Before using the USE algorithm, we clean the answers by trimming the whitespaces and converting the entire sentence to lowercase. No other pre-processing step is required since USE performs a certain amount of pre-processing on its own. Once the pre-processing is done, the answers can be compared with each other by first passing them to the USE algorithm and then calculating the cosine distance of the embeddings.

The problem arises when the user's answer is way larger than the model answer and has some extra points. If we compare these two answers traditionally, then the results might get diluted because the length of the user answer is more i.e., it has some extra points which may be correct but

are not mentioned in the model answer. Since the entire evaluation is based on what the model answer is provided by the teacher, we neglect these redundant points from the user's answer and remove them from that answer so that the final score does not get affected. A sentence-by-sentence comparison approach is to be followed to identify exactly which sentences or points are useless for the calculation of the score.

B. Sentence-by-sentence comparison

The answers are passed through a sentence tokenizer which creates tokens of the sentences and stores them in a list. Every sentence of the model answer is compared with each sentence of the user answer and the similarity scores are stored in the similarity matrix. A 2D matrix is considered because of the uncertainty in the positioning of the sentences. The sentences of the user answer having the maximum similarity score compared to the sentences in the model answer are considered for further evaluation. This is done by taking the maximum score of each row of the matrix by assuming those sentences would be the most similar. This can be called a sentence hit.

		User answer				
Model answer		0.98	0.67	0.56	0.34	0.45
		0.42	0.37	0.92	0.29	0.63
		0.33	0.24	0.38	0.26	0.34

Fig. 2. Similarity Matrix

In figure 2, we can see the similarity matrix of model answer versus the user answer. The rows represent sentences in the model answer while the columns represent sentences in the user answer. The circles highlight that the 1st statement of the model answer is the most similar to the 1st sentence of the user answer and thus can be considered as a sentence hit. Similarly, the 2nd sentence of the model answer is the most similar to the 3rd sentence of the user answer. Now that we have found the most similar sentences, the 3rd sentence of the model answer does not have any corresponding sentence with a decent similarity score. We take the maximum of the 3rd row and check it against a threshold value to be certain that no sentence of the user answer is similar to the 3rd sentence of the model answer. The threshold value taken is 0.40 which is greater than any of the similarity scores in the 3rd row. So, we can conclude that the 3rd sentence is missing in the user's answer. This method gives us the missing points while the method below will give us redundant points.

For finding out the useless points in the user answer, we can follow a similar approach and perform column-wise checking in the similarity matrix. The 1st and 3rd sentences have found a hit but for sentences 2, 4 and 5, no sentence has got a hit. But there might be cases when a sentence of the model answer is written as two different sentences by the user. This is taken care of by the threshold value that we set. Considering 0.40 as the threshold value, we can eliminate the 2nd and the 5th sentences as the maximum of these columns,

are greater than the threshold value. The 4th sentence, however, does not have any similar sentence in the model answer because the maximum of this column is less than our threshold value and therefore can be termed as a useless sentence which should be removed before calculating the final score to get undiluted results.

C. Generating Feedback

Feedback is an essential part of this system as every human needs it to improve and grow. Students need feedback so that they do not repeat the same mistakes. ASSESS is a system that not only can conduct full-fledged subjective tests but also can be used as a self-assessment platform. For feedback, we show the missing points that the user should have included to get a higher score and the useless points that the user can skip next time because those sentences are not required to calculate the score. These points are identified using the sentence-by-sentence comparison algorithm mentioned in section 3.2. The useless points are removed from the user answer to only keep the important points for the calculation of the result. The user answer and the model answer are now passed to the comparison algorithm which will give us the similarity score between the two answers.

D. Keyword Matching

Several times, certain keywords are an integral part of the answer and cannot be replaced. Such keywords carry a certain weightage to the overall score of the answer and thus we have considered keyword matching as well. We take the input of keywords from the teacher and check whether the answer contains all the keywords. We find the percentage of keywords that are present in the answer. Keywords are optional and are subject to the answer in question.

E. Score Calculation

The score is calculated considering the two main factors that are similarity score and the keyword matching ratio by the formula given below-

$$\text{Grade} = (a * \text{sim_score} + (1-a) * \text{keyword_match_ratio}) * m$$

where a is a weighted constant, $a = 0.85$ when keywords are mandatory & $a = 1$ when keywords are optional.
 sim_score is the score calculated in section 3.3.
 $\text{keyword_match_ratio}$ is calculated in section 3.4.
 m is the total marks possible for the answer under consideration.

IV. RESULTS AND DISCUSSIONS

While testing the algorithm, we considered a sample of 50 short answers which were manually evaluated and scored by a teacher and automatically evaluated by our system. The manual evaluation of the answers by a teacher was essential to observe how similar the scores given by the teacher were to the automatically generated scores by the system. With this approach, we can analyse how accurate the algorithm is. We

have considered a case where answers are 10 marks each. Now the scores for these answers can be anything ranging from 0-10. For each answer, there were 2 scores i.e., the score given by the teacher and the score given by the system after evaluation. We calculated the difference between the grades for each answer, noted down the observations and plotted a bar graph as shown in figure 3. The idea behind this is lesser the difference, the more accurate the algorithm is. We considered five groups which were as follows: (0-1 -> Excellent), (2-3 -> Good), (4-6 -> Bad), (7-8 -> Very bad) and (9-10 -> Worst). As shown in figure 3, 52% of the total answers considered had a difference of 0 which means the scores were perfect. In total, 76% of grades fell in the excellent region. None of the answers fell in the 'Very bad' or 'Worst' regions. Few answers had a difference of 4 or 5. These were the answers which were tested for contradiction i.e., we tested the answers by negating their actual meaning and this is the part on which we are still working on.

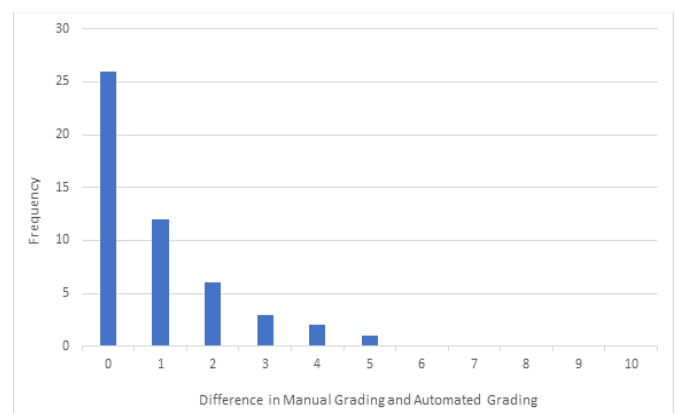


Fig. 3. Frequency v/s Difference in Manual & Automated Grading

Figure 3 is a frequency vs 'difference in grading' bar graph that helped in understanding the accuracy of the algorithm. Once the answer is evaluated by the system and feedback is generated, the way it displays it to the student is shown in figure 4. Along with the student's final grades, a detailed question by question feedback report is generated for each test which shows marks allotted per question and the pass/fail result. In case if the student misses any important key point from the model answer provided, then that sentence is highlighted in the expected answer section so that the student can realize which points did he/she miss out on. Also, when the student's answer has some extra points (which may be correct but are not present in the model answer) or wrong points which are not required for that particular question, then such points are struck-through. Since we are comparing the user answer with the model answer only, so if at all the extra point written by the student is correct and not in the model answer, it will be struck through. But this would not affect the final answer score as these points will be removed before final scoring as described in section 3.2. Figure 4 shows the student feedback report for a particular question of a test attempted by the student.

Assess logo	Test3	Total Score: 84.08%	Attempted: 2/2	PASS
Question 1	✓	Q 1. What are data structures?		
Question 2	✓	Score: 6/10		
<p>Your answer</p> <p>Data Structure can be defined as the group of data elements which provides an efficient way of storing and organising data in the computer so that it can be used efficiently. Data structures can be classified as linear and non-linear.</p>		<p>Expected answer</p> <p>In computer science, a data structure is a data organization, management, and storage format that enables efficient access and modification. More precisely, a data structure is a collection of data values, the relationships among them, and the functions or operations that can be applied to the data.</p>		
<p>◀ PREV</p>		<p>NEXT ▶</p>		

Fig. 4. Student-side test feedback report

V. LIMITATIONS AND FUTURE SCOPE

The future of education is going to be online, and our system tries to bridge the gap between the online assessment from offline to online. Our system currently does not provide accurate results for contradictory answers compared to the model answer. We are working on improving the efficiency of the system when it comes to such contradictory statements. These sentences convey opposite ideas compared to the model answer. We wish to tackle this problem in the future to provide higher accuracy. Security features like automated proctoring and behavioural analysis of the candidate can be collected for better use and also to avoid malpractices that are probable in the online examination system.

VI. CONCLUSION

We developed a system using a semantic learning algorithm, USE, for the analysis of the subjective answers. Necessary feedback is also generated for the students to improve their answers by highlighting the contents of the model answers provided. This automated approach is beneficial when students need to be assessed online for self-improvement. This system gives special emphasis to the specially-abled by providing various speech-based usability features, where the gaps are filled by providing audio facilities like listening to the questions and answering them verbally. The advantage of this system is that it is near completion, has improved performance and caters to a very large audience.

REFERENCES

- [1] A. Rokade, B. Patil, S. Rajani, S. Revandkar and R. Shedje, "Automated Grading System using Natural Language Processing", Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies, 2018.
- [2] D.R. Tetali, G. Kiran Kumar and L. Ramana, "A Python Tool for Evaluation of Subjective Answers (APTESA)", International Journal of Mechanical Engineering and Technology (IJMET), vol. 8, pp. 247–255, July 2017.
- [3] M. Syamala Devi and H. Mittal, "Machine Learning techniques with Ontology for subjective answer evaluation", International Journal on Natural Language Computing (IJNLC), vol. 5, April 2016.
- [4] C. Roy and C. Chaudhari, "Case-Based Modeling of answer points to expedite semi-automated evaluation of subjective papers", IEEE 8th International Advance Computing Conference (IACC), pp. 85–90, 2018.
- [5] K. Surya, E. Gayakwad, Nallakaruppan and M.K., "Deep learning for Short Answer Scoring", International Journal of Recent Technology and Engineering (IJRTE), vol. 7, issue 6, 2277–3878, 2019.
- [6] D. Cer, Y. Yang, S.Y. Kong, N. Hua, N. Limtiaco, R.S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, c. Tar, Y.H. Sung, B. Strope and R. Kurzweil, "Universal Sentence Encoder", 2018.
- [7] A. Dhokrat, R.G. Hanumant and C. Mahender, "Automated Answering for Subjective Examination", International Journal of Computer Applications, vol. 56, pp. 14–17, 2012.
- [8] S.A. Saany, A. Mamat, A. Mustapha, L.S. Affendey and M.N.A. Rahman, "Semantics question analysis model for question answering system", Applied Mathematical Sciences, vol. 9, pp. 6491–6505, 2015.
- [9] M.K. Siva Prasad & P. Sharma, "Similarity of sentences with contradiction using semantic similarity measures", The Computer Journal, 2020.