**Acknowledgements**

The progress of this project would not have been possible without the unconditional support of many people, each of whom in their own capacities have helped us in carrying out this project.   
We would like to take this special opportunity to thank Dr. Vijaya Kumar B.P., Professor and Head, Department of Information Science and Engineering, for permitting us to take up this project.

We also wish to express our gratitude to our Project guide, Dr. Lingaraju G M, who has been abundantly helpful and has oared invaluable assistance and exemplary guidance to us.

We would also like to extend our gratitude to Mr. Shekhar for his assistance and having helped us throughout the project.

We would also like to thank all the teaching and non-teaching staff of the ISE department for their kind cooperation and assistance.

**Abstract**

The current way of checking subjective paper is adverse. Evaluating the Subjective Answers is a critical task to perform. When human being evaluates anything, the quality of evaluation may vary along with the emotions of Person.

In Machine Learning, all result is only based on the input data provided by the user.

Our proposed system uses Machine Learning and NLP to solve this problem. Our Algorithm performs a task like Tokenizing words and sentences, Part of Speech tagging, Chunking, Chinking, Lemmatizing words and Word-netting to evaluate the subjective answer. Along with it, our proposed algorithm provides the semantic meaning of the context.

Our System is divided into two modules:

The ﬁrst one is extracting the data from the scanned images and organizing it in the proper manner.

And the second is applying ML and NLP to the text retrieved from the above step and giving marks to them, classifying them into various categories.

**Keywords**: Naive Bayes Algorithm, Cosine Similarity, Classiﬁer, Semantic Checking, Machine Learning.

**Contents**

[**1 Introduction**](#_gjdgxs) 6

[1.1 Motivation](#_30j0zll) 1

[1.2 Scope](#_1fob9te) 1

[1.3 Objectives](#_3znysh7) 1

[1.4 Proposed Model](#_2et92p0) 1

[1.5 Organisation of Report](#_tyjcwt) 1

[**2 Literature Review**](#_3dy6vkm) 9

[**3 System Analysis and Design**](#_1t3h5sf) 10

[**4 Modelling and Implementation**](#_4d34og8) 14

[**5 Testing, Results and Discussion**](#_2s8eyo1) 18

[**6 Conclusion and Future Work**](#_17dp8vu) 20

**References 7**

**List of Figures**

Fig 1: Workflow diagram

Fig 2: Case diagram

Fig 3: Sequence diagram

Fig: 4- Terminal Output Screenshot

**List of Tables**

Table 1: Adding non numeric values of keywords and key sentences

Table 2: Adding non numeric values of grammar

Table 3: Dataset marking scheme

Table 4: Comparative Evaluation Result table

**Chapter 1**

**Introduction**

The manual system for evaluation of Subjective Answers for technical subjects involves a lot of time and eﬀort of the evaluator. Subjective answers submitted have various parameters upon which they can be evaluated such as the question speciﬁc content and writing style especially individual student’s handwriting being the bottleneck of the whole text retrieval process. Evaluating subjective answers is thus a critical task to perform.

When human being evaluates anything, the quality of evaluation may vary along with the emotions of the person. Performing evaluation through computers using intelligent techniques ensures uniformity in marking as the same inference mechanism is used for all the students. In Machine Learning, all result is only based on the input data provided by the user. Our Proposed System uses machine learning and NLP to solve this problem. Our Algorithm performs a task like Tokenizing words and sentences, Part of Speech tagging, Lemmatizing words and Word-netting to evaluate the subjective answer. Along with it, our proposed algorithm provides the semantic meaning of the context.

The need for online examination aroused mainly to overcome the drawbacks of the existing system. The main aim of the project is to ensure user-friendly and more interactive software to the user. The online evaluation is a much faster and clear method to deﬁne all the relevant marking schemes. It brings much transparency to the present method of answer checking.

The answers to all the questions after the extraction would be stored in a database. The database is designed as such that it is very easily accessible. The work of checking hundreds of answer sheets which more or less contains the same answer can be quite a boring task for the teachers. This system can be used instead in order to reduce their burden. It will save a lot of eﬀort and time on teachers’ part. The obvious human mistakes can be reduced to obtain an unbiased result. The system calculates the score and provides results fairly quickly. This system can be widely used in academic institutions such as schools, colleges, coaching and institutes for checking answer sheets.

## Motivation

* Answers are crucial testing tools for assessing academic achievement, integration of ideas

and ability to recall, but are expensive and time consuming to grade manually.

* Manual grading of answers takes up a significant amount of instructor’s valuable time, hence is an expensive process. Automated grading, if proven to match or exceed the reliability of human grades, will significantly reduce costs.
* Automating repetitive tasks has been the main aim of the industrial and technological revolution.
* The main motivation behind this project is to ensure user-friendly and more interactive software to the user.
* The human eﬀorts applied in this repetitive task can be saved and spent more in other academic endeavors. The obvious human mistakes can be reduced to obtain an unbiased result.
  1. **Scope**

The product has scope in departments of education where developing new forms of testing

and grading methods, to assess the new common core standards. For example, we know that answers are an important expression of academic achievement, but it is expensive and time consuming for states to grade them by hand. So, we are frequently limited to multiple-choice

standardized tests. We believe that automated evaluation systems can yield fast, effective and affordable solutions that would allow states to introduce answers and other sophisticated

testing tools.

Benefits:

* Human time and effort are saved.
* Coherence in evaluation of all the scripts present

## Objectives

* The first and the main objective is to save Time of the evaluators and the amount of budget any institute will have to spend.
* Sometimes, Teachers evaluating answers, evaluate the same answer differently for different papers. The marks awarded sometimes might depend on the emotions of the teacher evaluating. To overcome this, to grade everyone equally, we can make use of this software.
* It is possible that each teacher evaluates the same answer differently. Students might or might not be satisfied with the evaluation of their paper. For fair evaluation, this software is very useful.
* The goal also includes dealing with students’ academic data encountered in the evaluation system without any professional bias.
* To overcome a cold start problem in an evaluation system
  + - Building a robust algorithm for correctly awarding the marks to students.
    - Building an efficient algorithm that takes wide variations between answers into account.

## Proposed Model

One of the key roadblocks to teaching and evaluating critical thinking and analytical skills is the expense associated with scoring tests to measure those abilities. For example, tests that require “constructed /responses” (i.e., written answers, written essays) are useful tools, but they typically are hand scored, commanding considerable time and expense from public agencies. So, because of those costs, standardized examinations have increasingly been limited to using “bubble tests” that deny us opportunities to challenge our students with more sophisticated measures of ability.

Thus, an opportunity for the people with technical background and designer potential to turn it into business or an advancement and betterment of mankind.

Our Proposed System uses machine learning and NLP to solve this problem. Our Algorithm performs a task like Tokenizing words and sentences, Part of Speech tagging, Lemmatizing words and Word-netting to evaluate the subjective answer. Along with it, our proposed algorithm provides the semantic meaning of the context.

## Organization of Report

In order to explain the developed system, the following sections are covered:

* **Literature Review** describes the study of the existing systems and techniques taken into account prior to development of the proposed system.
* **System Analysis and Design** provides a detailed walk through of the software engineering methodology adopted to implement the model, an overview of the system and the modules incorporated into the system
* **Modelling and Implementation** provides a deeper insight into the working of the model. The various modules and their interactions are depicted using relevant descriptive diagrams.
* **Testing** the model to ensure bug/error free model along with the **Results** obtained. **Discussion** then provides detailed analysis on quality assurance measures.
* **Conclusion** about the Results obtained after successfully running the model and **Future Scope** of the model is highlighted.

**Chapter 2**

# Literature Review

1. Dharma Reddy Tetali, Dr. Kiran Kumar G, Lakshmi Ramana “Evaluation of Subjective Answers (APTESA)” IJMET, Volume 8, Issue 7, July 2017, Article ID: IJMET\_08\_07\_029.
2. Piyush Patil, Sachin Patil, Vaibhav Miniyar, Amol Bandal

“Subjective Answer Evaluation Using Machine Learning” IJPAM, International Journal of Pure and Applied Mathematics, Volume 118 No. 24 2018, ISSN: 1314-3395.

**Chapter 3**

# System Analysis and Design

* Response times: How long should the Application take to load and how fast it can perform the functional requirements.
* Good User Interface: The User Interface be such that even a new user can easily access all the functions of the application.
* Should Not Crash: If an Application crashes then the chances that a user will come is very less. So, we should make sure that the Application crashes are kept to a minimum.
* Less File Size: Size of the application matters a lot as users wants to allocate as less disk space as possible for an Application. We should be able to compress the Application as much as possible.
* Processing Time: How long the Application take to make the function call should be less.
* Easy to use: The images and visuals used in the applications should make it easy to use the application.

## 

## 3.1 Workflow

Our System Design is divided into two modules:

* First one includes Extracting the data from the scanned images or using a Graphical User Interface to take student credentials and internal answers and organizing it in the proper manner.
* Second one includes inputting the model answers against which the students answers will be evaluated by the Faculty.
* Next step involves applying Machine Learning and Natural Language Processing Techniques to the text retrieved from the above step and giving marks to them.
* Procedure: The software will take a scanned copy of the answer as an input and then after the preprocessing step, it will extract the text of the answer. This text will again go through processing to build a model of keywords, grammar and feature sets. Model answer sets and categorized keywords are mentioned as well in the input. The classiﬁer will then, based on the training, will give marks to the answers. Marks to the answer will be the ﬁnal output along with class category it belongs to, based on the numerical result obtained and no. of errors in the answers.
* The data comprising of text of answers i.e. Students answers and Model answers both will be stored in Google Firebase after collecting each respective author through a GUI component or scanned images of written submission. That require setting up the Firebase and configuring the libraries in the specific modules to import and push the answers and keywords and final marks onto the real time database.

## 3.2 Class Diagram

This system can be widely used in academic institutions such as schools, colleges, coaching and institutes for checking answer sheets. It can also be implemented in diﬀerent organizations which conduct competitive examinations. Our Algorithm performs a task like Tokenizing words and sentences, Part of Speech tagging, Chunking, chinking, Lemmatizing words and Wordnetting to evaluate the subjective answer. Along with it, our proposed algorithm provides the semantic meaning of the context.

## 

Fig 1: Workflow diagram

## 3.3 Other Techniques Used

* Prior to Cosine Similarity we tried using Euclidean Distance which give good results but was not working well for all keywords and answers
* We tried bit-match string before fuzzy logic but that gave wrong answers for small changes

## 3.4 System Requirements

* Operating System: Windows 10 and Linux Ubuntu 18.04.2 LTS
* Processor: Intel Core i3 and above
* OS type: 64-bit
* RAM: 4 GB and above

**Chapter 4**

# 

# Modelling and Implementation

**Mathematical model:**

P(Class — Keyword, Grammar, QST) = P(Keywords — Class) \*P(grammar — class)\*P(QST — class)\*P(class).

For example, if we have the values of Keywords, grammar, QST as 2,0,2 respectively. Then we can evaluate the class. For the input values given, it is evaluated against all the classes and then the class having the maximum probability is the class to which the given input will belong. Each Answer will be given biased value in between 1-10. Actual marks will be evaluated on that basis. For example, If after evaluation 6 is the value we are getting and question is of 5 Marks then the actual marks calculated from this using Marks obtained for the answer out of 10 = 6 \* 5 /10 = 3 i.e. generically we can deﬁne the formula as Actual Marks=(Biased Value After Evaluation) \* (Max marks that can be obtained for the answer/ 10) As the dataset in ML approaches works well with numeric dataset we have mapped our six values as follows:

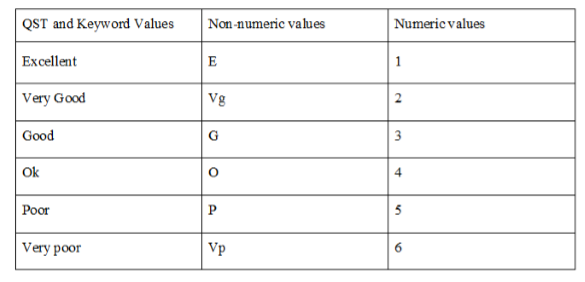


Table 1- Adding non numeric values of Keywords and Key Sentences

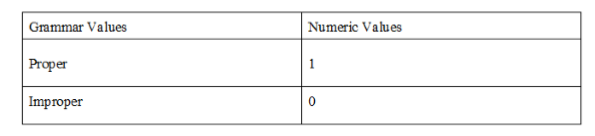


Table 2- Adding non numeric values of grammar

These 3 values i.e. Keywords, Grammar and Question Speciﬁc things is passed to Naive Bayes classiﬁer as an input. Naive Bayes classiﬁer is probabilistic classiﬁer which is based on the maximum probability to which the given input belongs. Now in order to evaluate these 3 parameters we are using following strategy:

1. **Keywords and key sentences: (e, vg, g, o, p, vp)**

In Cosine Similarity ﬁrst we will make the vector of both model answer and users answer. We transfer the answer into vector form using cosine method. Lesser the Angle greater the similarity and greater the value of cosθ.

It is calculated using two components numerator(num) and denominator(den)

Num =P(vec1[x] \* vec2[x]) (Where vec1[x] andvec2[x] are answer vectors and model answers vector respectively).

Den= sum1 den ∗√sum2 (Where sum1,sum2 are keys obtained from model answer and user answer.)

1. **Grammar: (y, e)**

API which gives number of errors in the answer. This is evaluated only if above phase has some value. For Improper Grammar: 0, For Correct Grammar: 1

1. **Question Speciﬁc things: (e, vg, g, o, p, vp)**

Here we are using Fuzzy wuzzy - using multiple ratio functions available in Fuzzy Logic. The Fuzzywuzzy library analyses the text using degrees/features of text instead of the rigid Boolean values of 0 or 1.

After having some observations, we have 21 of them as our training dataset. We will get any one of 1-9 value as the output from this Classiﬁer. NB classiﬁer gives the output with evidence/ probability value. Further Depending on that value, we can increase the accuracy.

Example: if output is 6 and the probability that given query belongs to 6 is 70% then we can increase this 6as 6 + 0.7 = 6.7. So, the total marks evaluated on the basis of 6.7.

## 4.1 Case Diagram

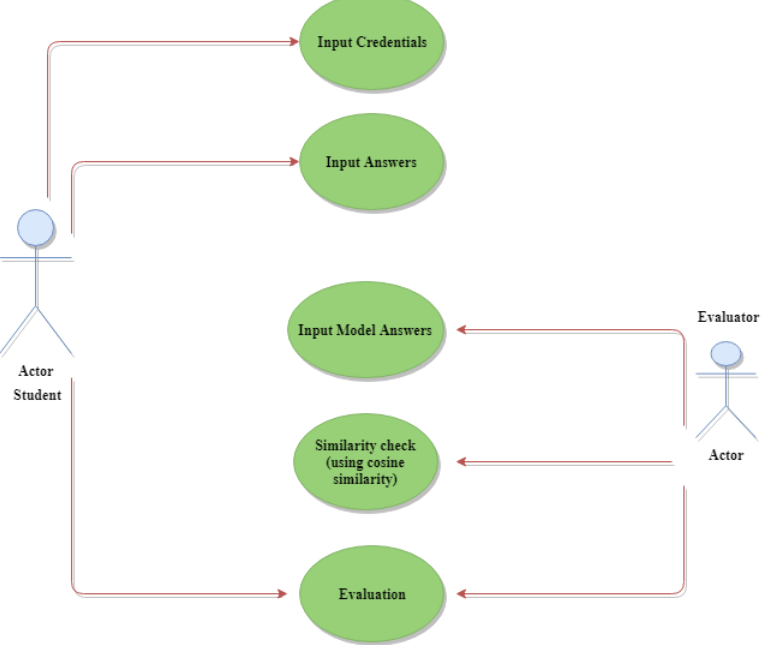
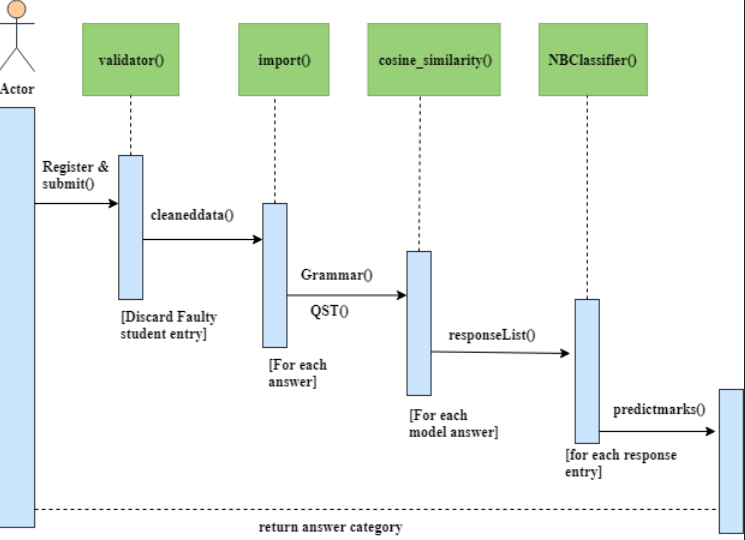


Fig 2- Case diagram

## 4.2 Sequence Diagram



**Fig 3 -sequence diagram**

**Chapter 5**

# Testing, Results and Discussion

## 5.1 Testing

We have trained our model using the dataset of Student written answers and Faculty’s model answers. The values that we have deﬁned in the table are set according to the requirement of the answer. The evaluator/moderator/teacher of the answer sheet can deﬁne these values for themselves to suit their needs.

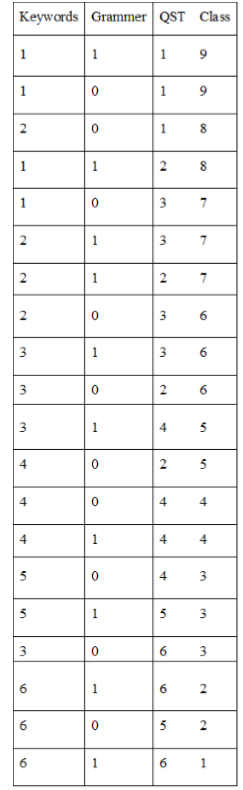


Table 3 - Dataset marking scheme

## 5.2 Results

We have given 3 questions to be filled by each student. Each question carries 5 Marks. All answers are evaluated ﬁrstly by our Professors then our algorithm will evaluate them. Then the similarity between Professor Evaluation and our algorithm evaluation is taken into consideration. we have found:

## 

Table 4 - Comparative Evaluation Result table

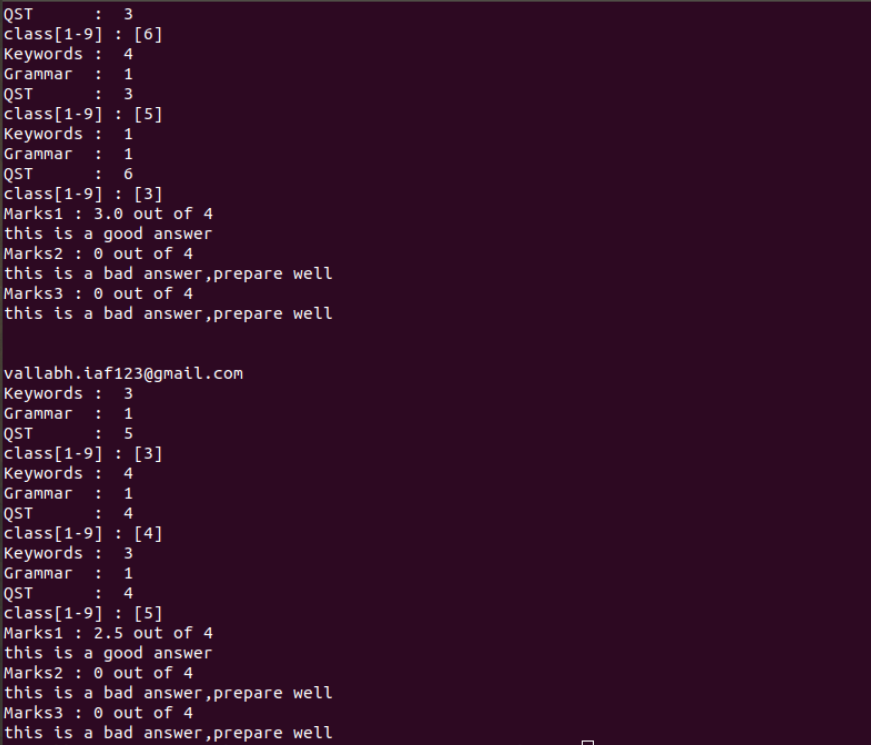


Fig: 4- Terminal Output Screenshot

## 5.3 Discussion

* The accuracy of the evaluation can be increased by feeding it a huge and accurate training dataset
* As the technicality of the subject matter changes diﬀerent classiﬁers can be employed.
* Further improvement by taking feedback from all the stakeholders such as students and teachers can improve the system meticulously.
* We use parts of speech as of the features,length,and previous answer classification.

**Chapter 6**

# Conclusion and Future Work

The techniques discussed and implemented in this project should have a high agreement (up to 90 percent) with Human Performance. The project works with the same factors which an actual human being considers while evaluation such as length of the answer, presence of keywords, and context of key-words. Use of Natural Language Processing coupled with robust classiﬁcation techniques, checks for not only keywords but also the question speciﬁc things. Students will have certain degree of freedom while writing the answer as the system checks for the presence of keywords, synonyms, right word context and coverage of all concepts. It is concluded that using ML techniques will give satisfactory results due to holistic evaluation. The accuracy of the evaluation can be increased by feeding it a huge and accurate training dataset. As the technicality of the subject matter changes diﬀerent classiﬁers can be employed. Further improvement by taking feedback from all the stakeholders such as students and teachers can improve the system meticulously.

**Bibliography**

1. P. D. Turney and P. Pantel, From frequency to meaning: Vector space models of semantics, J.Artif. Intell. Res., vol. 37, pp. 141188, 2010.
2. T. K. Landauer, Automatic Essay Assessment, Assess. Educ. Princ. Policy Pract., vol. 10, no. 3, pp.295308, 2003.
3. T. K. Landauer and P. W. Foltz, An introduction to latent semantic analysis, Discourse Process.,no. April 2012, pp. 3741, 2012.
4. P. W. Foltz, Latent Semantic Analysis for Text-Based, Behav. Res. Methods, Instruments Comput.,vol. 28, no. 2, pp. 197202, 1996.
5. P. W. Foltz, Latent Semantic Analysis for Text-Based, Behav. Res. Methods, Instruments Comput.,vol. 28, no. 2, pp. 197202, 1996.
6. L. Bin, L. Jun, Y. Jian-Min, and Z. Qiao-Ming, Automated essay scoring using the KNN algorithm,Proc. - Int. Conf. Comput. Sci. Softw. Eng. CSSE 2008, vol. 1, pp. 735738, 2008.
7. F. Noorbehbahani and a. a. Kardan, The automatic assessment of free text answers using a modiﬁed BLEU algorithm, Comput. Educ., vol. 56, no. 2, pp. 337345, 2011.