JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Automatic Descriptive Answer Evaluation

Using Natural Language Processing

[1]Niyati Malik
[1]Student
[1]Computer Science and Engineering,
[1]Indira Gandhi Delhi Technical University for Women, Delhi, India

Abstract : Various methods have been implemented for Automatic Answer Evaluation of Subjective or Objective answers but the methodologies used by us have helped us in the evaluation of descriptive answers. Since there is a plethora of data available, developments in systems are extremely important to keep the knowledge base and the user base more efficient. Here we present a report on how people, especially teachers can use Automatic Descriptive Answer Evaluation for the checking of examination sheets. The bare minimum human intervention makes this a very effective and useful model in personal as well as professional life.

IndexTerms **- Component,formatting,style,styling,insert.**

I. INTRODUCTION

In today's world, with unlimited storage and everything being internet based, it is important to make models with features that reduce human intervention and give maximum efficiency, making lives easier for everyone who requires the use of technology. Agile steps must be taken to make advanced and intricately designed techniques which help the people with their personal and professional lives. This project aims at reducing human intervention while evaluating answer sheets of students and making the whole process automatic so as to get rid of any bias that may arise due to manual evaluation and also to make the task faster and more efficient.

There are various assessment strategies used to assess a student's performance. The most used technique is the response to a descriptive question. In this technique, a student expresses their opinion in response to the question verbatim. The descriptive response automatic rating system will be very cooperative for various universities and educational institutions to very effectively assess a student's performance.

A student can answer a question using different grammatical styles and choosing different words that are similar to the actual answer. The motivation behind the automated response script assessment comes from less time, less workforce involvement, prohibition of psychological changes in the human assessor, and is very easy to keep and retrieve. It also ensures that mood swings or changes in perspective of the human reviewer will not affect the review process. The automatic evaluation of the response script will help us overcome the difficulties encountered in the manual evaluation. Here, a student's written response is provided as input and the system will automatically score grades after the assessment. The system takes into account all possible factors such as spelling error, grammatical error, and cosine similarity measures for scoring marks.

The scope of this project extends to both personal and professional uses. It would come in handy for a variety of applicants like Answer Sheet Evaluation: (Digital Correction of Exam Papers), Result Generation, Student request for scan copy, re-evaluation etc.

Abbreviations and Acronyms

NLP – Natural Language Processing
ML – Machine Learning
NLTK - Natural Language Toolkit

I. RESEARCH METHODOLOGY

The methodology section outline the plan and method that how the study is conducted. This includes Universe of the study, sample of the study,Data and Sources of Data, study's variables and analytical framework. The detailsare as follows;

3.1Population and Sample

For our model, our dataset was basically about 150-200 answer sheets that were sampled. We made an application on a cloud platform called Heroku for the students to fill in their answers. These answer sheets were then converted to a csv file. The csv file was then preprocessed for tokenizing the words present in it and for the implementation of stemming and lemmatization.

This tokenized file was then sampled against the 'ideal' answer provided by us. Our provided answers were also tokenized for easier computation.

3.2 Theoretical framework

The basic idea behind our project's approach is to evaluate a descriptive/subjective answer based on a model or a sample answer provided by the teacher/examiner. The students would be required to answer questions pre-prepared on an application, and those answers will be evaluated. Each answer goes through preprocessing using natural language processing methods like stemming, lemmatization and tokenization. After the preprocessing, keywords in the model answer and the student's answer are compared using cosine similarity. The grammar of the answer is also checked. For every answer, some question related concepts are then checked using fuzzy logic. The combined results of cosine similarity and fuzzy logic are fed to the Naive Bayes algorithm which provides us with the final marks.

We usually use Cosine similarity for automatic evaluation of answers. The biggest advantage of this method is that it can be done regardless of the difference of the size of the two documents that are to be analysed. Evaluation of keywords is done based on the results of pre-processing. It is based on Cosine Similarity between the student's answer and the model answer provided by the teacher. The implementation involved here is the conversion of documents into vectors that are present in a multidimensional system. After this conversion the 'cos' of the angle between these two vectors is calculated. Lesser the angle, the more the similarity between the documents. This methodology helped us in faster answer evaluation and allotting of marks. We also used Fuzzy Logic and Naive Bayes algorithms to get better and more efficient results. So, the design of our project basically revolves around 4 main concepts - Natural Language Processing, Cosine Similarity, Fuzzy Logic and Naive Bayes algorithm.

The approach towards the development of the algorithm involves amalgamating concepts of Natural Language Processing like context identification and text similarity. The ideal answer provided by the examiner and the student's answer are both taken into account. The process of Tokenization, Stemming and Lemmatization is performed. These methods are important for easier implementation of algorithms and for finding. Three parameters are specified here :

1. Keywords
2. Grammar
3. Question Related Concepts

3.4 Statistical tools and econometric models

The automated descriptive answer sheet evaluation process can be divided broadly into 4 stages. These are:-

- Text Summarization
- Text Preprocessing
- Similarity Measures
- Allotment of marks

3.4.1. Text Summarization

Text summarization refers to the technique of shortening long pieces of text. The intention is to create a coherent and fluent summary having only the main points outlined in the document

Algorithm of Text summarization

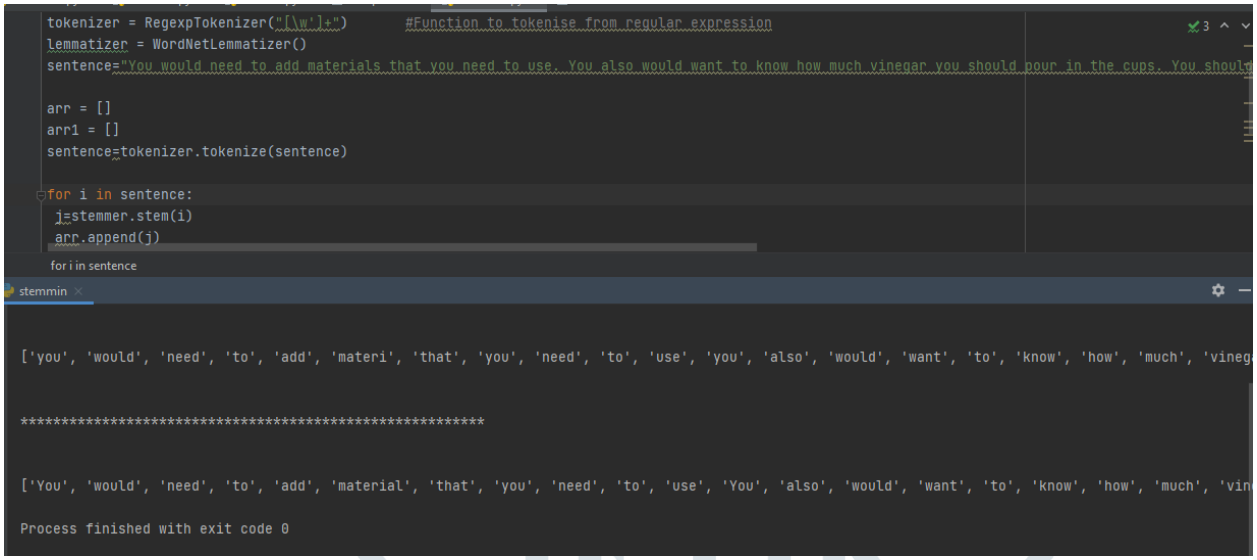
1. Accept the input text
2. Now tokenize the accepted text into word
3. Removal duplicates from word list
4. Put a counter for frequency of each word
5. Compute the word percentage by dividing word frequency by the length of word list
6. Compare the word percentage with a maximum and minimum threshold value and select average frequent word as keywords and remove most frequent word and less frequent word.
7. Counter for the size of window for each sentence using keywords
8. Compute weight of each sentence by dividing square of no of keyword in sentence by window size
9. Sort the sentence in a descending order based on weight value and select first n sentence as summary

3.4.2. Text Preprocessing

In the summarized texts there are certains words which carry less information and can be ignored so that further text processing tasks can be facilitated. Pre processing refers to the way in which data can be converted to computer understandable form. A very efficient way to handle text preprocessing is natural language processing. It contains tokenize text into word, removes StopWord, lemmatize word, remove duplicate word etc. Natural Language Toolkit (NLTK) is considered a leading platform for building python programs to work with human language data. It has huge amount of built in libraries which can be used in the text preprocessing by providing us with the benefit of typing less commands. The NLTK function word_tokenize, which is a built in function, is used to split the text into word and store in a list. The most crucial step in text preprocessing is to filter out the unnecessary words and remove the redundancy of the documents.NLTK contains StopWord corpus which consists of the unnecessary words which are irrelevant to define the meaning of the sentence.

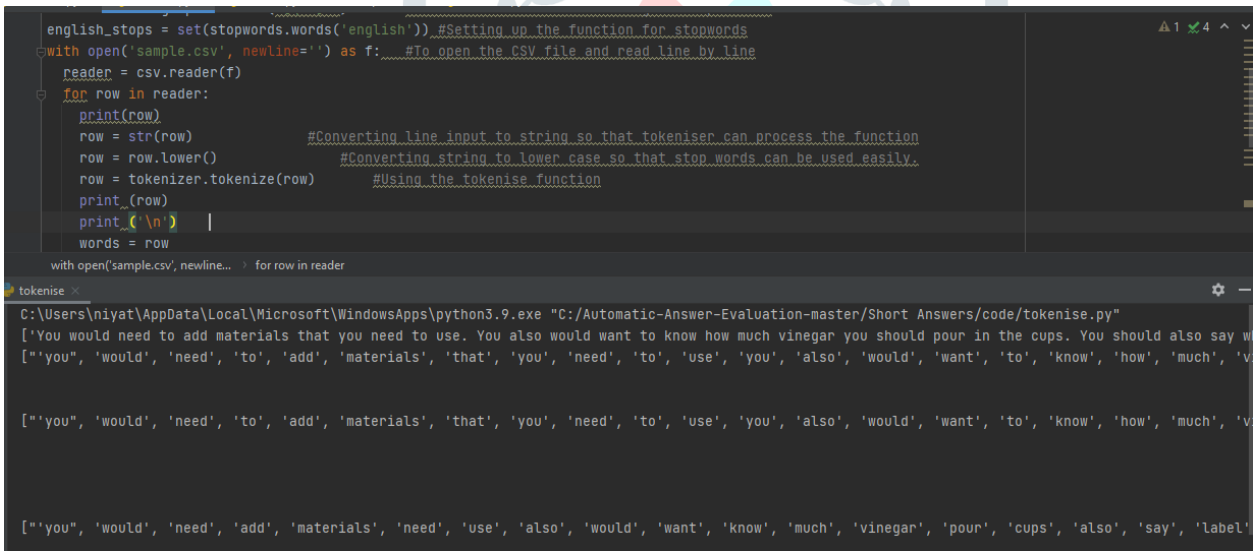
Lemmatization can be defined as the way of changing words to its basic form called lemma. The purpose is to save time used in processing as the number of words in the word list are reduced. WordNet lemmatizer which is a built-in function in NLTK

converts the words from word list to their basic forms. The purpose is to keep the format of all the data the same, this is required to perform some application on the data. The format can be bigram or digram i.e. sequence of the neighbouring elements in a token string. Similarity of structure of text is analyzed using the frequency distribution of bigram. Bigrams are generated using an inbuilt function in NLTK and it generates a bigram list of all words. A dictionary is used to store the word as a key and frequency count of each word. After making this dictionary of the bigrams and frequency count of each word, it is used as a way to measure similarity.



(1)

Code snippet for tokenisation



(2)

Code snippet for Stemming

3.4.3. Similarity Measures

In many cases, it is needed to define whether two sentences are similar or not. Similarity measures is a term which tells if two sentences are similar or not by considering the different angle of similarity. Several similarity measure techniques are available that can be performed. In this experiment, cosine similarity is performed.

Cosine similarity

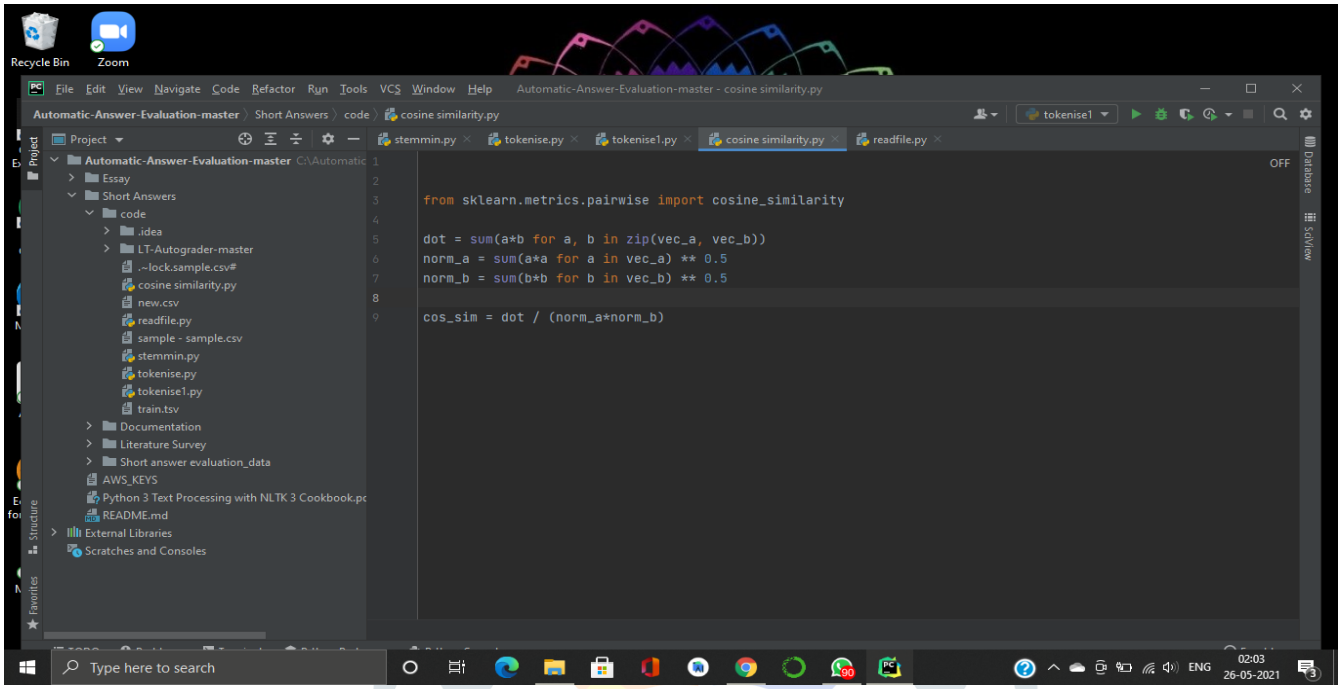
Cosine similarity is an efficient similarity measure technique. It looks at the angle by two documents and tells how similar they are.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}, \quad (3)$$

Cosine Similarity formula

Algorithm of cosine similarity

1. Input dictionary of words and frequency.
2. Two vectors are to be taken, one for the model answer and another for the student answer. Length of Total word list should be the length of each vector.
3. Compute dot product of two vector
4. Find norm of first vector
5. Find norm of second vector
6. Multiply the first and second norm
7. To find cosine similarity divide the dot product result by multiplication result.



(4)

Code snippet for Cosine Similarity

3.4.4. Allotment of Marks

The aim of this project is to assess the descriptive answers using automation techniques and assign marks. This will lessen the time for assessing answer scripts and bring equality for evaluation. For satisfying those requirements, we have used a weighted parameter-based-technique for automatic assessment.The summary generated by the extracted text is vital to the experiment. For keeping the summary generation efficient we have generated a summary of two techniques, that is keyword based summarization and bag of word based summarization, keeping in consideration the reference summary. The estimated score of the answers is displayed in the Table.
Table indicates that the F-score of our used summarization technique is greater than the bag-of-words based summarization technique. We have considered five parameters in this experiment for scoring marks. These are synonym similarity, bigram similarity, grammatical-spelling error, cosine similarity and Jaccard similarity.

Feature	Keyword based summarization	Bag of word based summarization
Precision	0.9	0.83
Recall	0.83	0.41
F--score	0.86	0.53

Table 3.1: Score Calculation

The above parameters are used to automatically assess two types of questions (M50and M100) based on marks. A unique value of weight is allotted to each of the parameters depending on the type of question. The value of the weight is assigned after taking an average of the model answers for each parameter. Through the model answers we have observed that the significance of grammatical and spelling error type parameters is less compared to the synonym for evaluation of answer script.

The higher the value of the weight, the more significance it holds for marks allocation. The values of the parameters can range from 0 to 1 depending on the similarity and presence of the error.Higher parameter value symbolises that similarity between the model answer and student answer is more the similarity between the two and vice versa. In this project we have taken 100 sample answers for testing the accuracy of our model. The answer script consists of two types of questions.

Further the five above mentioned parameters are computed from the 100 sample answer sheets and are used for automatic scoring. Most of the cases the automatically allocated score and manually assigned marks are very close. When the student answer and the true answer contain more structural similarity as well as synonym similarity, the automated scored marks are very close to the manually scored marks. On the other hand, a notable difference between the automated scored marks and manually scored marks exist when the student answer and the true answer have less structural similarity while more Cosine similarity.

IV. RESULTS AND DISCUSSION

The basic motive of this project was to reduce the manual work required for the evaluation of descriptive answer sheets. Examiners aren't always able to give their complete focus while checking answer sheets and sometimes may become biased while allotting marks to the students. Hence, we tried and were somewhat successful in achieving the goal of decreasing the manual load of teachers and also the unbiased allotment of marks by creating this project.

To fulfill these needs, we used weighted parameter based technique, the generation of summary played an important role and it was done using F-score. High parameter value would indicate more similarity in the two answers.

The methodology used was able to give accurate results to an extent. The answers provided by us for comparison were first checked to be correct before further evaluation of the sample set. The marks allotted to the sample set were also unbiased as they were based on the text similarities and not on the teacher's judgement.

The implementation of NLP allowed us to get precise results with precision value of 0.8 and a recall of 0.93. All possible words and their combinations were created with accuracy and helped us in the completion of the project.

V.ACKNOWLEDGMENT

I would like to thank my mentor Dr. DK Tayal for his constant guidance and support throughout the project and for helping me complete the paper.

REFERENCES

[1] Semantic Analysis and Evaluation of Subjective Passage**,** Jaco van de Pol and Michael Weber
[2] WME and Automatic Mathematical Answer Checking, G Hanna, DA Reid & M de Villiers
[3] Semantic Analysis and Evaluation of Subjective Passage, Douglas Grimes & Mark Warchauer