A Platform Independent Methodology For Examination Paper With Smart Answer Checking System

Mr. Gajanan N. Patil[1], Prof. Mr. Sachin P. Godse[2]
[1]Dept of Computer Engineering
[2]Assistant Professor, Dept of Computer Engineering
[1, 2] Sinhagad Academy of Engineering, Pune

Abstract- *Writing is a skill that is used in all academic coursework as well as through a person's professional and personal life. Technology has become an influential factor when it comes to education. Smartphone's are used in developed countries to complement established exam system and develop the new ways of Handwriting Software. For this purpose, we present Smart Handwriting Software. It is a collection of interface tools that allows the user to take input text by using a pen or finger touch. Smart Handwriting Software explores handwriting as a new modality. In this paper, we have reviewed the different techniques of this system, the description of currently available systems makes short free text response and finally propose a system that would evaluate the descriptive type answers using Natural Language Processing. Then it will check the paper using module answer sheet and final result will be generated.*

Keywords- Short free text response, Descriptive type answer, Natural Language Processing, Optical Character Recognition.

I. INTRODUCTION

Writing is a ubiquitous everyday activity. Using computers, we type documents, complete forms and enter database queries. However, writing, taking notes or entering text in immersive smart software is almost impossible. Cut off from conventional text input devices, such as keyboards, immersed participants are unable to effectively communicate textual information in smart software.

We present Smart handwriting Software, a collection of interface tools that allows the user to take notes, annotate documents and input text simply by writing with a pen or finger touch. Using a spatially-tracked, pressure- sensitive graphics tablet, pen Smart Notepad Software, Smart Notepad Software explores handwriting input as a modality for interaction in immersive smart software. This paper reports details of the Virtual Notepad interface and interaction techniques, discusses implementation and design issues, reports the results of initial evaluation and overviews possible applications of virtual handwriting.

The different methods used for automatic marking of free-text responses are being classified into three main types: Statistical, Information Extraction and Full Natural Language Processing.

A. Statistical Technique

It is only based on keyword matching, hence considered as poor method. It cannot tackle the problems such as synonyms in student answers, nor does it takes into account the order of Words, nor can it deals with lexical variability.

B. Information Extraction (IE) Technique:

Information Extraction consists in getting structured information from free text. IE may be used to extract dependencies between concepts. Firstly, the text is broken into concepts and their relationships. Then, the dependencies found are compared against the human experts to give the student's score.

C. Full Natural language processing (NLP):

It involves parsing of text and find the semantic meaning of student answer and finally compare it with instructors answer and assign the final scores.

II. RELATED WORK

To investigate the potential of handwriting, we have developed Smart handwriting Software, a collection of simple interface tools for note taking and text input in smart software. Smart handwriting Software allows the immersed participant to write notes, images, and modify previously entered notes, as well as distribute them in smart software.

It is underpinned by NLP and Knowledge Representation (KR) techniques. In this system model answers are generated with the help of concepts already given and later student's answers are processed by NLP technique. Later on only concepts detection is done and finally scores are assigned.

In this paper [1], they have presented a paper reports details of the Virtual Notepad interface and interaction techniques, discusses implementation and design issues, reports the results of initial evaluation and overviews possible applications of virtual handwriting.

In this paper [2], a reviewed the techniques underpinned this system, the description of currently available systems for marking short free text response and finally proposed a system that would evaluate the descriptive type answers using Natural Language Processing.

In this paper [3], they have presented a tool which does not segment the characters on the basis of their shapes. Instead of segmenting the character on the basis of their shapes, it divides the character into the points. It finds the co-ordinate of the every point where pen move from the pen down to pen up. This is a platform independent tool that runs online on any browser which is HTML5 enabled. Since it does not depend upon the shape of the character, thus it segments more efficiently than other tools.

In this paper [4], the Textual Entailment Module in the present system are: Word Net based unigram match, bigram match and skip-gram. The results obtained from the four answer validation modules are integrated using a voting technique. Evaluation scores obtained on the AVE 2008 test set show 67% precision for "VALIDATED" decisions.

III. PROBLEM DEFINITION

So if these types of characters are recognized on the basis of their shapes then it will not be an easy task because these similar shapes will create a great ambiguity for the system for the recognition of the handwritten character. There are also several other problems associated with these tools which are follows:

- Various tools which are available for the tacking input from the user and segment these are machine dependent and there is not any tool available which works on the all type of systems and platforms.
- Every tool need to install some type of framework or application the system for the working.
- Tools which are available for the segmentation, segments the handwritten character according to their shapes and then provide these shapes to the recognition machine to recognize these shapes but most of the times these segments create the ambiguity for the system to recognize these shapes, because most of the characters shares the same segments.
- Most of the tools are language dependent so they need to install these language frameworks on the system.
- There are some technologies that work upon the concept of n-gram model which work on the concept of probability, i.e. the probability of occurrence of a character when another character has already occurred, most of the time does not work correctly.
- There are several difficulties involved in segmenting a character into the sub-character. Another problem is that it is not clear how to break a letter into sub- character units.

The elements of the handwriting segmentation interface typically include:

- A finger touch or pen from which user write the input.
- Any touch sensitive surface, which may be adjacent to or integrated with, an output display for tacking input.
- A software application which captures the movements of the pen or finger touch from the writing
- surface and then translate the resulting stroke into digital text.

Developing the Handwritten character Segmentation system has some greater challenge because of the following reasons:

- Presence of very large number of character sets.
- Different writing styles.
- Different writing speed.
- Complexity of the characters.
- High degree of similarity of shapes between the characters.
- Due to variance in handwriting, poor reliability of extracted stroke features.

These systems fail to check spelling & grammatical mistakes made by students. As well as they were unable to check the correct word order. Even the answers with wrong word order were awarded assigned scores by mere presence of words in student response. So to overcome the encountered problems the system is going to be developed that evaluated students descriptive answers by considering the collective meaning of multiple sentences. Also system will mark spelling mistakes made and finally scores will be assigned to student answer.

IV. PROPOSED SYSTEM

There are several applications and techniques available for the segmentation of a handwritten character but there is a major problem that how to break a handwritten character into multiple parts so that these parts can be passed to the recognition system for the recognition of that handwritten character and paper checking system. For solving this type of problems.

The existing system checks single line text response without considering word order. So the proposed system will try to avoid this problem by considering collective meaning of multiple sentences. The primary focus of this newly proposed system is to determine the semantic meaning of student answer with a consideration that student responses to question in number of ways.

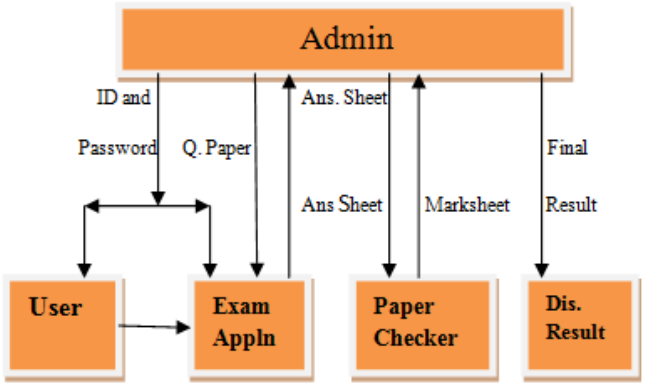

Fig. Propose Diagram

The system basically focuses on multiple sentences response. It is basically composed of following components:

A. User Module:

It consists of accept Id and password to admin then enter exam application

B. Exam Application Module:

In this module question as well as correct response to respective question is entered by tutor. Tutor will also identify and enter the keywords from correct answer with their respective weights. In a question editor where question will be displayed and response editor to enter student response.

C. Paper Checker Module:

Following text processing module, that actual evaluation of student response with correct answer takes place. Each and every word of student response is compared with correct answer with presented answer sheet then scores are assigned.

After score assignment Final scores are calculated by making summation of assigned scores of all words.

D. Display Result:

Final calculated scores assigned to student response are given in report.

V. ALGORITHMS

5.1 Text detection API

- **Contents**

1. Detecting text in a local image
2. Detecting text in a remote image

Text Detection performs Optical Character Recognition. It detects and extracts text within an image with support for a broad range of languages. It also features automatic language identification.

Detecting text in a local image and remote image

- To perform Text Detection, make a POST request and provide the appropriate request body.
- Refer to the images:annotate API endpoint for complete details.
- See the AnnotateImageRequest reference documentation for more information on configuring the request body.
- If the request is successful, the server returns a 200 OK HTTP status code and the response in JSON format.
- For your convenience, the Cloud Vision API can perform Text detection directly on an image file located in Google Cloud Storage or on the Web without the need to send the contents of the image file in the body of your request.
- Text recognition is the process of detecting text in images and video streams and recognizing the text contained therein. Once detected, the recognizer then determines the actual text in each block and segments it into lines and words. The Text API detects text in

Latin based languages (French, German, English, etc.), in real-time, on device.

5.2 Optical Character Recognition (OCR)

Optical Character Recognition (OCR) gives a computer the ability to read text that appears in an image, letting applications make sense of signs, articles, flyers, pages of text, menus, or any other place that text appears as part of an image. The Mobile Vision Text API gives Android developers a powerful and reliable OCR capability that works with most Android devices and won't increase the size of your app.

In this codelab, you will build an app that shows a live camera preview and speaks any text it sees there. Along the way, you'll learn how to use the Mobile Vision API to delight and empower your users.

The technology extracts text from images, handwriting, and even scan of printed text.

5.3 Compute Levenshtein Distance

In approximate string matching, the objective is to find matches for short strings in many longer texts, in situations where a small number of differences is to be expected. The short strings could come from a dictionary, for instance. Here, one of the strings is typically short, while the other is arbitrarily long. This has a wide range of applications, for instance, spell checkers, correction systems for optical character recognition, and software to assist natural language translation based on translation memory.

The Levenshtein distance can also be computed between two longer strings, but the cost to compute it, which is roughly proportional to the product of the two string lengths, makes this impractical. Thus, when used to aid in fuzzy string searching in applications such as record linkage, the compared strings are usually short to help improve speed of comparisons.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we present Smart handwriting Software, a collection of interface tools which introduce and explore immersive handwriting interfaces. Using Smart handwriting Software, the user can take notes while still inside a Smart Software, experiencing and evaluating phenomena of interest. Tools which are available for the segmentation segment the handwritten character according to their shapes and then provide these shapes to the recognition machine to recognize these shapes. The tools which recognize the characters on the basis of their shapes cannot work correctly. They create the ambiguity for the system. then work on paper checking and result system.

Our future research will both improve and add to the current functionality of Smart handwriting Software. The problem of latency should be addressed by optimizing algorithms and the software architecture of the system. The current use of handwriting recognition is limited and more work should be done to investigate its potential. Our informal evaluation suggests that additional gestures for stroke manipulation might be useful, as well as tools that allow the user to magnify parts of images or diagrams on the Smart handwriting Software for detailed examination. And perfect accuracy on paper checking system.

REFERENCES

[1] Ivan Poupyrev 1, 2, Numada Tomokazu 1, Suzanne Weghorst 2 1 IS Lab, Hiroshima University 2 HIT Lab, University of Washington “Virtual Notepad: Handwriting in Immersive VR” 1 IS Lab, Hiroshima University 2 HIT Lab, University of Washington 1-4-1 Kagamiyama Box 352142 Higashi-Hiroshima, 739 Japan Seattle, WA, 98195 USA.

[2] Ms. Shweta M. Patil, Prof. Ms. Sonal Patil “Evaluating Student Descriptive Answers Using Natural Language Processing” Vol. 3 Issue 3, March – 2014 International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181

[3] Mayur Vyas1, Karun Verma2, “A Platform Independent Methodology for On-line Handwritten Character Segmentation” 2014 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)

[4] Partha Pakray Santanu Pal, Alexander Gelbukh,” Automatic Answer Validation System on English Language ” 2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)