REVIEW-3

PAPER EVALUATION AND HANDWRITING RECOGNITION USING AI

SUBMITTED BY

AGNIM CHAKRABORTY (18BCE2186)

DEVJYOT SINGH SIDHU (18BCE0973)

PREPARED FOR

ARTIFICIAL INTELLIGENCE (CSE3013)

UNDER THE GUIDANCE OF

DR. SUBHASHINI R.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

SLOT: E1+TE1

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Fall Semester 2020-21, November 2020

ABSTRACT :

In this bad time due to COVID-19 everyone wants to develop a fully automated online examination system. Due to increasing number of courses and appearing students many hours of examiner and a lot of efforts are required for effective evaluation. Computer and technologies can be used to solve such complex problem. The goal is to evaluate and assign scores to descriptive answer which are comparable to those of human assigned score by coupling AI technologies. Our proposed system includes an algorithm that is capable of evaluating an answer script based on your own handwriting and comparing it with the initially entered answer keywords both key word wise and also comparing the synonyms of these keywords.What it does is generates a matching percentage based on these keywords and synonyms and based on this percentage scored a student gets marks for that particular question. So the input given to the system is keywords of the answer, processing in the system is *through Handwritten Text Recognition (HTR) models and output given by the system* are marks scored out of total marks to be awarded to the system. So it benefits both the teacher and student, teachers have no need to check the whole answer they can award marks based on percentage evaluated by the system and student benefits as even if his answers do not contain the exact keywords the AI system is intelligent enough to give marks based on matching synonyms.

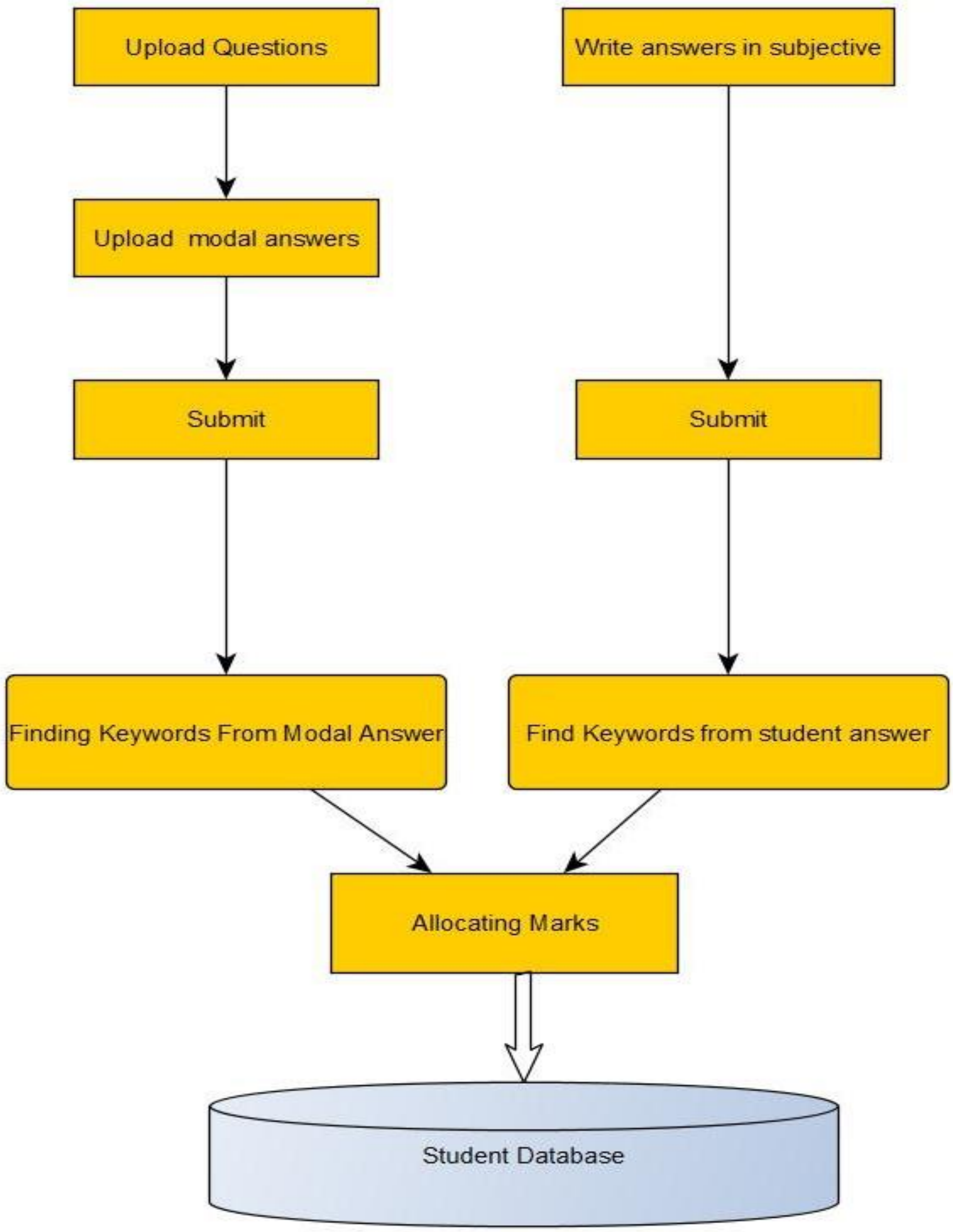
INTRODUCTION :

The manual system for evaluation of Subjective Answers for technical subjects involves a lot of time and effort of the evaluator. Subjective answers have various parameters upon which they can be evaluated such as the question specific content and writing style. Evaluating subjective answers is 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, Chunking, chinking, Lemmatizing words and Wordnetting 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, Extracting the data from the scanned images and organizing it in the proper manner.

The software will take a scanned copy of the answer as an input and then after the preprocessing step, it will extract the test of the answer. This text will again go through processing to build a model of keywords and feature sets. Model answer sets and keywords categorized as mentioned will be the input as well. The classifier will then, based on the training will give marks to the answers. Marks to the answer will be the final output.

The need for online examination aroused mainly to overcome the drawbacks of the existing system as well as the global COVID-19 crisis. 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. Automating repetitive tasks has been the main aim of the industrial and technological revolution.

ARCHITECTURE DIGRAM :



<u>BACKGROUND STUDY</u> :

1) "The state of the art in online handwriting recognition" by C.C. Tappert,C.Y. Suen, T. Wakahara IBM Thomas J. Watson Res. Center, Yorktown Heights, NY, USA.[1]

This survey describes the state of the art of online handwriting recognition during a period of renewed activity in the field. It is based on an extensive review of the literature, including journal articles, conference proceedings, and patents. Online versus offline recognition, digitizer technology, and handwriting properties and recognition problems are discussed. Shape recognition algorithms, preprocessing and postprocessing techniques, experimental systems, and commercial products are examined.

2) "The IAM-database: an English sentence database for offline handwriting recognition" by *CH* U.-V. Marti & H. Bunke Department of Computer Science, University of Bern[2].

In this paper we describe a database that consists of handwritten English sentences. It is based on the Lancaster-Oslo/Bergen (LOB) corpus. This corpus is a collection of texts that comprise about one million word instances. The database includes 1,066 forms produced by approximately 400 different writers. A total of 82,227 word instances out of a vocabulary of 10,841 words occur in the collection. The database consists of full English sentences. It can serve as a basis for a variety of handwriting recognition tasks. However, it is expected that the database would be particularly useful for recognition tasks where linguistic knowledge beyond the lexicon level is used, because this knowledge can be automatically derived from the underlying corpus. The database also includes a few image-processing procedures for extracting the handwritten text from the forms and the segmentation of the text into lines and words.

3)"Improvement of Relative accreditation Methods Based on Data Mining and artificial Intelligence for Higher Education by Canan Tastimur, Mehmet Karakose and Erhan akin Department of Computer Engineering Firat University Elazig, Turkey"[3]:

In this paper IT-based method is proposed with help of AI which simplifies application of the accreditation system to departures. High efficiency and more accurate results can be obtained by being given results obtained from proposed method as feedback. There are determined whether success accreditation criteria by examining each of accreditation criteria with knowledge extracting and processing techniques.

4)"Teaching and learning computational thinking by solving problems in AI" by ananta Srisuphab Faculty of IT,Mahidol University,Thailand[4]:

In this paper, we resourcefully integrated elements of artificial Intelligence (AI) into introductory computing courses. In addition to a comprehension of the esence of CT,AI enabled inspiration of collaborative problem solving beyond abstraction, logical reasoning, critical and analytical thinking.

5) "Principles for Teaching, Leading, and Participatory Learning with a New Participant: AI" by Eugene G Kowch, Ph.D. Professor Chair, Leadership, Policy & Governance Unit Werklund School of Education University of Calgary Calgary, Canada[5]:

This paper discusses principles and practices that can optimize artificial intelligence (AI) in teaching and learning from the perspectives of leading organizational change and by reimaging learning activities with aI as a collaborative partner.Finally by "zooming in" on instruction and aI, we use activity theory to imagine better inclusions of social and cultural components with aI as an important, emerging, and unscripted new partner.

6) "The Determination and analysis of Factors affecting to Student Learning by artificial Intelligence in Higher Education" by Nigar Ozbey, Mehmet Karakose and aysegul Ucar Department of Computer Engineering Firat University Elazig, Turkey"[6]:

The student learning assessment tool based artificial intelligence is disclosed in this study. The main object of this tool is to develop the student's learning specific subjects in the study using concept maps by using artificial intelligence. The probability distribution of defined concepts is calculated in the concept maps developed by students. To evaluation of student understanding, the graphic curve generated by the vehicle is performed by analyzing. Researchers have developed a rating system using concept maps. The proposed artificial intelligencebased learning tool in the study. In the proposed system the basic process is to transform topics students study into concept maps. and then XML documents are obtained from these concept maps. XML document is parsed in order to perform XML analyzing. The evaluation of students' learning is supplied by reaching to analysis results teacher with user interface.

7) "Teacherbot: interventions in automated teaching" by Siam Bayne[7]:

The paper discusses of a new technology that can be deployed to make teaching automated enhancing the concepts of 'teacher-light' tuition and enhancement in teaching efficiency by automation in teaching . The teacherbot uses its robotic component along with its major component aI to compete with human intelligence and solve there problems enhancing learning for both teachers as well as the students.

8) "Education 4.0 - artificial Intelligence assisted Higher Education: Early recognition System with Machine Learning to support Students' Success by Monica Ciolacu Faculty of Business Informatics, Deggendorf Institute of Technology, DIT, Deggendorf, Germany"[8]:

The paper focuses on improving learning by contribution of aI mainly evolving two methods Fist by contribution is aI assisted Higher Education Process with smart sensors and wearable devices for self-regulated learning. Secondly by describing the first results of Education 4.0 didactic methods implemented with learning analytics and machine learning algorithms.aim of the study is to predict the final score before the Final assessment Examination.

9) "Artificial Intelligence to Enhance Learning Design in UOW Online, a Unified approach to Fully Online Learning" by Rory L. L. Sie Learning, Teaching, and Curriculum University of Wollongong Wollongong, australia[9]:

The given paper discusses about creating awareness for teachers by determining learning design of subject.To ensure the approach can be scaled up to cater for potentially hundreds of subjects, the manual labeling serves as input for an artificial Intelligence (aI) algorithm that will train a model to label intended learning activities automatically. In addition to student demographics and behavior, the learning design and subject content will be used to augment an aI model that predicts future student outcomes.

10) "Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition?" by Joan Puigcerver Pattern Recognition and Human Language Technology Research Center Universitat Politècnica de València 46022 Valencia, Spain[10]:

In this work, we provided multiple evidences that multidimensional recurrent layers may not be necessary to achieve good accuracy for HTR. We provided some intuitive explanation: although MDLSTM are in principle more powerful than ConvNets, this additional power seems not be necessary for HTR, given that similar features are learned. Moreover, a statistically-sound analysis of the experimental results also supports this claim, for two widely used datasets.

11) "Gated Convolutional Recurrent Neural Networks for Multilingual Handwriting Recognition" by Théodore Bluche , Ronaldo Messina[11]:

In this paper, we propose another neural network architecture for cutting edge handwriting recignition, option to multi-dimensional long short term (MD-LSTM) intermittent neural networks. The model depends on a convolutional encoder of the input images, and a bidirectional LSTM decoder anticipating character sequences. In this worldview, we target creating conventional, multilingual and reusable highlights with the convolutional encoder, utilizing more information for move learning. The engineering is additionally persuaded by the requirement for a quick preparing on GPUs, and the necessity of a quick decoding on CPUs. The primary contribution of this paper lies in the convolutional gates in the encoder, empowering heirarchical context sensitive feature extraction. The trials on an enormous benchmark including seven dialects show a predictable and critical improvement of the proposed approach over our past creation frameworks. We likewise report best in class results on line and paragraph level acknowledgment on the IAM and Rimes databases.

12) ”Are 2D-LSTM really dead for offline text recognition?” by Bastien Moysset, Ronaldo Messina[12]:

In this paper, it is shown that there is a recent trend in handwritten text recognition with deep neural networks to replace 2D recurrent layers with 1D, and in some cases even completely remove the recurrent layers, relying on simple feed-forward convolutional only architectures. The most used type of recurrent layer is the Long-Short Term Memory (LSTM). The motivations to do so are many: there are few open-source implementations of 2D-LSTM, even fewer supporting GPU implementations (currently cuDNN only implements 1D-LSTM); 2D recurrences reduce the amount of computations that can be parallelized, and thus possibly increase the training/inference time; recurrences create global dependencies with respect to the input, and sometimes this may not be desirable. Many recent competitions were won by systems that employed networks that use
2D-LSTM layers. Most previous work that compared 1D or pure feed-forward architectures to 2D recurrent models have done so on simple datasets or did not fully optimize the "baseline" 2D model compared to the challenger model, which was dully optimized. In this work, we aim at a fair comparison between 2D and competing models and also extensively evaluate them on more complex datasets that are more representative of challenging "real-world" data, compared to "academic" datasets that are more restricted in their complexity. We aim at determining when and why the 1D and 2D recurrent models have different results. We also compare the results with a language model to assess if linguistic constraints do level the performance of the different networks. The results show that for challenging datasets, 2D-LSTM networks still seem to provide the highest performances and we propose a visualization strategy to explain it.

13) “Word Beam Search: A Connectionist Temporal Classification Decoding Algorithm” by Harald Scheidl, Stefan Fiel , Robert Sablatnig[13]:

In this paper, Recurrent Neural Networks (RNNs) are used for sequence recognition tasks such as Handwritten Text Recognition (HTR) or speech recognition. If trained with the Connectionist Temporal Classification (CTC) loss function, the output of such a RNN is a matrix containing character probabilities for each time-step. A CTC decoding algorithm maps these character probabilities to the final text. Token passing is such an algorithm and is able to constrain the recognized text to a sequence of dictionary words. However, the running time of token passing depends quadratically on the dictionary size and it is not able to decode arbitrary character strings like numbers. This paper proposes word beam search decoding, which is able to tackle these problems. It constrains words to those contained in a dictionary, allows arbitrary non-word character strings between words, optionally integrates a word-level language model and has a better running time than token passing. The proposed algorithm outperforms best path decoding, vanilla beam search decoding and token passing on the IAM and Bentham HTR datasets. An open-source implementation is provided.

14) “Medicine Box: Doctor’s Prescription Recognition Using Deep Machine Learning” by Dr E.Kamalanaban, M Gopinath, S Premkumar [14]

A Doctor’s prescription is a handwritten document written by doctors in the form of instructions that describes list of drugs for patients in time sickness, injuries and other disability problems. While we receiving a new prescription from doctor, it is unable to understand what drug name is prescribed on it. In most cases, however, we wouldn't be able to read it anyway because doctors use Latin abbreviations and medical terminologies on prescriptions that are not understandable by the general persons which make reading it very difficult. According to the National Academy of Sciences estimates that at least 1.5 million peoples are sickened, injured or killed each year by errors while reading prescription. This paper resolves the problems in doctor's prescriptions through Medicine Box, and Smart phone application that uses Conventional Neural Network (CNN) to recognize handwritten medicine names and return readable digital text. This mobile application uses TensorFlow as the machine learning library, and Custom Repository to match the partial string with the drug name. With Medicine Box, cases of misinterpretation of medicine names can be decreased. This makes the ordinary persons to understand what doctor is prescribed in the prescription and also help for pharmacists.

15) "An Efficient End-to-End Neural Model for Handwritten Text Recognition" by Arindam Chowdhury, Lovekesh Vig[15]

Offline handwritten text recognition from images is an important problem for enterprises attempting to digitize large volumes of handmarked scanned documents/reports. Deep recurrent models such as Multi-dimensional LSTMs have been shown to yield superior performance over traditional Hidden Markov Model based approaches that suffer from the Markov assumption and therefore lack the representational power of RNNs. In this paper, we introduce a novel approach that combines a deep convolutional network with a recurrent Encoder-Decoder network to map an image to a sequence of characters corresponding to the text present in the image. The entire model is trained end-to-end using Focal Loss, an improvement over the standard Cross-Entropy loss that addresses the class imbalance problem, inherent to text recognition. To enhance the decoding capacity of the model, Beam Search algorithm is employed which searches for the best sequence out of a set of hypotheses based on a joint distribution of individual characters. Our model takes as input a downsampled version of the original image thereby making it both computationally and memory efficient. The experimental results were benchmarked against two publicly available datasets, IAM and RIMES. We surpass the state-of-the-art word level accuracy on the evaluation set of both datasets by 3.5% & 1.1%, respectively.

16) "A Scalable Handwritten Text Recognition System" by R. Reeve Ingle, Yasuhisa Fujii, Thomas Deselaers, Jonathan Baccash, Ashok C. Popat[16]

Most studies on hand-written text recognition systems are focused on building state-of-the-art models for line recognition on small datasets. However, adding hand-written text recognition ability to a large multilingual optical character recognition system comes with new hindrances. In this paper, three problems in building such systems are addressed: integration, data, and efficiency. One of the challenges is to obtain adequate amounts of good-quality training data. This problem was addressed by the use of online handwriting data collected for a large-scale production online handwriting recognition system. They describe the image data generation pipeline and study how online data can be used to build hand-written text recognition models. They also show that the data improve the models significantly under the condition where only a small number of real images is available, which is usually the case for hand-written text recognition models. They are able to support a new script at substantially lower cost. They also propose a line recognition model based on neural networks without recurrent connections. The model achieves a comparable accuracy with LSTM-based models while allowing for better parallelism in training and inference. They present a simple way to integrate hand-written text recognition models into an optical character recognition system. These constitute a solution to bring HTR capability into a large-scale OCR system.

17) "Automated Grading for Handwritten Answer Sheets using Convolutional Neural Networks" by Eman Shaikh, Iman Mohiuddin, Ayisha Manzoor, Ghazanfar Latif, Nazeeruddin Mohammad[17]

Optical Character Recognition is an integral part of a lot of existing systems today. Traditional character recognition methods cannot differentiate characters or words in a scanned image. In this paper, a simple system is proposed, that uses a personal computer, a portable scanner and an app whereby, handwritten answer scripts are automatically corrected. In order to achieve handwritten character recognition, the scanned images are fed through a machine learning classifier known as the Convolutional Neural Network (CNN). They trained two CNN models on 250 images that were collected from students at Prince Mohammad Bin Fahd University. Their system outputs the final score of the student by comparing each classified answer with the correct answer. The experimental results showed that the proposed system had a high testing accuracy of 92.86%. They believe that system can be used by the instructors in educational institutions to automatically grade the handwritten answer sheets of students effectively, saving the time of the faculties.

18) "Automated content grading using machine learning" by Rahul Kr Chauhan, Ravinder Saharan, Siddhartha Singh, Priti Sharma[18]

Answer sheet grading is a hectic, time and labour-intensive work and is exposed to inefficiency and bias in checking. This research project is an experiment that aims to automate the grading of theoretical answers written in examinations by pupils in technical courses which had been graded by humans in the past. The authors of this paper show how an algorithmic approach in machine learning can be used to automatically examine and grade theoretical content in exam answer papers. Bag of words, their vectors & centroids and a few semantic and lexical text features have been used overall. Machine learning models have been implemented on datasets manually built from exams given by graduating students enrolled in technical courses. These models have been compared to show the effectiveness of each model.

19) "Automated Grading System Using Natural Language Processing" by Amit Rokade, Bhushan Patil, Sana Rajani, Surabhi Revandkar, Rajashree Shedge[19]

The authors noticed that most of the articles which cover automated grading consider keyword matching to be a crucial aspect while grading answers. They realised that even though these are important, it is human to forget several uncommon terms and instead replace them with words that have a similar meaning. In this paper, the authors have proposed a solution to grading of papers of theory-based subjects using Natural Language Processing. Machine learning techniques like Semantic Analysis were adopted. As a single answer can be presented in a number of ways by different students, matching keywords is inefficient. That is why, using ontology, extraction of words and their synonyms related to the domain is done which makes the evaluation process holistic as presence of keywords, synonyms, the right word combination and coverage of concepts can now be checked. The above-mentioned techniques were implemented along with Ontology and were tested on common input data consisting of technical answers. The results were analysed and an unbiased, high accuracy automated grading system for a theory-based subject was obtained with very little error rate which is comparable to a differential human-to-human error rate. The algorithm is designed based on the responses collected during the survey conducted amongst teachers regarding their parameters when correcting papers manually.

20) "Design of an Automated Essay Grading (AEG) system in Indian context" by Siddhartha Ghosh, Sameen S. Fatima[20]

The TOEFL exam is one of the best examples of this application. The students essays are evaluated both by human and automated essay grading system. Then the average is taken. AEG might provide precisely the platform we need to explicate many of the features those characterize good and bad writing and many of the linguistic, cognitive and other skills those underline the human capability for both reading and writing. They can also provide time-to-time feedback to the writers/students by using that the people can improve their writing skill. A meticulous research of last couple of years has helped us to understand the existing systems which are based on AI & Machine Learning techniques and finding the loopholes and at the end to propose a system, which will work under Indian context, presently for English language influenced by local languages. This paper focuses on the existing automated essay grading systems, basic technologies behind them and proposes a new framework to overcome the problems of influence of local Indian languages in English essays while correcting and by providing proper feedback to the writers.

<u>Innovation Introduced in Design:</u>

Automated script-based cropping of images, Handwriting recognition combined with keyword/tag search, Training on multiple datasets to improve accuracy.

<u>Ultimate Utility value of the project to the society and industry:</u>

- It will reduce the workload of teachers
- The evaluation process would be fast
- Physical presence in exam halls is not required
- Answer scripts can be stored in a virtual database which is safer.
- Getting test results immediately give students peace of mind
- Students can take the exam in a more comfortable environment
- Avoid commute that adds stress and saves money
- It accommodates students with disabilities
- Best method to conduct descriptive handwritten exams in these COVID times.

<u>New Learning experience gained in the process of the project Like using new software package, new fabrication technique etc.:</u>

TENSORFLOW, PIL

METHODOLOGY:

Proposed system of different types of neural networks to predict sequences of characters or words. For many years, handwritten text recognition systems have used the Hidden Markov Models (HMM) for the transcription task, but recently, through Deep Learning, the Convolutional Recurrent Neural N etworks (CRNN) approach has been used to overcome some limitations of HMM. To exemplify a CR NN model, given figure.

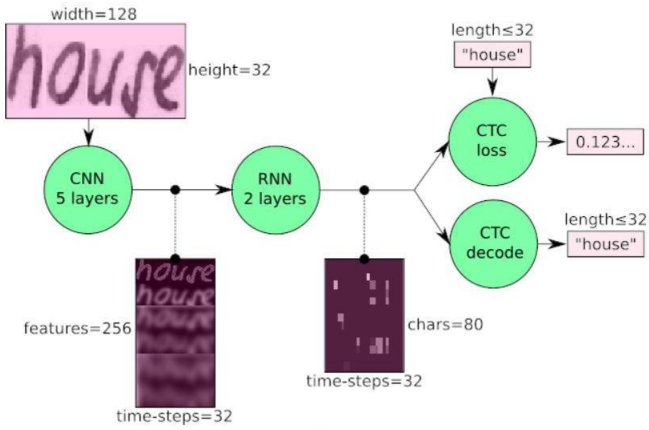


Fig. 2: Overview of a CRNN (source: Build a Handwritten Text Recognition System using TensorFlow)

The workflow can be divided into 3 steps.

Step 1: the input image is fed into the CNN layers to extract features. The output is a feature map.

Step 2: through the implementation of Long Short-Term Memory (LSTM), the RNN is able to propag ate information over longer distances and provide more robust features to training.

Step 3: with RNN output matrix, the Connectionist Temporal Classification (CTC) [9] calculates loss value and also decodes into the final text.

Explaining step 3 (CTC) is the same for all architectures presented, then the Vanilla Beam Se arch method is used, since it doesn't require a dictionary for its application, unlike other know n methods such as Token Passing and Word Beam Search. Therefore, the architectures presen ted in the following sections only act in steps 1 and2.

In addition, the charset for encoding text is also the same for all datasets. So, the list used cons ists of 95 printable characters from ASCII table (Figure) by default and doesn't contain accent ed letters.

```
 !"#$%&\'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}
```

<u>Proposed model-</u>

Architectures that are selected for handwritten text recognition in this project correspond to approache s that use BLSTM in the recurrent layer (RNN) and that were eventually compared with MDLSTM as an alternative to computational cost and high complexity.

In this way, the first architecture (CNN-BLSTM), introduced by Puigcerver, has a high level of recog nition rate and a large number of parameters (around 9.6 million). The Figure 8 shows in detail the dis tribution of parameters and hyperparameters through 5 convolutional layers and 5 BLSTM.
The second architecture (Gated-CNN-BLSTM), now introduced by Bluche and Messina [18], has the highlight of the Gated-CNN approach. In summary, this technique aims to extract more relevant featu res of the image. Instead Puigcerver approach, this model has very few parameters (around 730 thousa nd), making it more compact and faster. The Figure shows in detail the distribution of parameters and hyperparameter sthrough 8 convolutional layers (3 gated included) and 2 BLSTM.

Lastly, the proposed architecture (Gated-CNN-BLSTM) that was inspired by [17] and [18], aiming at: (*i*) to achieve results compatible with Puigcerver model; and (*ii*) to keep the small number of paramete rs (not exceeding one million), such as Bluche et al. model.

It was a great challenge to combine the two advantages of each architecture into a single one, however, in the developmental, I applied another Gated approach, once presented by in the context of Languag e Model. Other methods that also brought differences in results (improvements) were the Batch Renor malization and the Parametric Rectified Linear Unit (PReLU) activator.

Thus, the proposed Gated-CNN-BLSTM architecture preserves the low number of parameters (around 820 thousand) and high recognition rate. The Figure 10 shows in detail the distribution of parameters and hyperparameter through 11 convolutional layers (5 gated included) and 2 BLSTM.

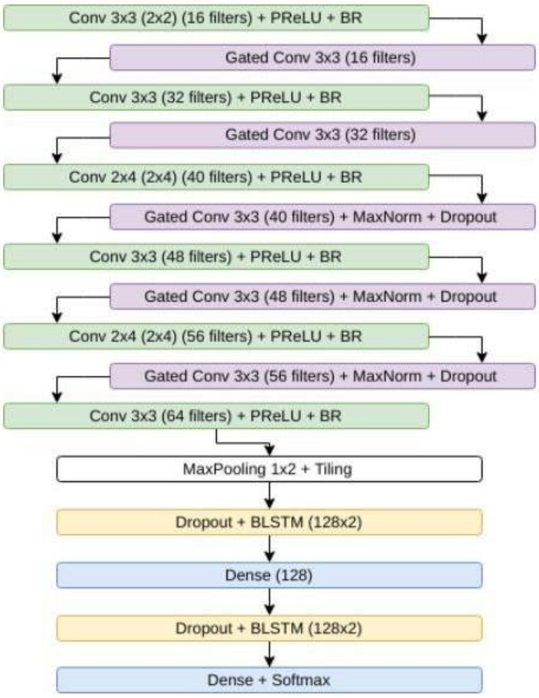


Fig. 10: Workflow of the Flor architecture

<u>Datasets-</u>

For the experiment, it was used the free segmentation approach of text lines of Bentham, IAM, Rimes and Saint Gall datasets.

The Institut für Informatik und Angewandte Mathematik (IAM) database contains forms with English manuscripts, which can be considered as a simple base, since it has a good quality for text recognition (Figure 5). However, it brings the challenge of having several writers, that is, the cursive style is unres tricted.

Input: it is a gray-value image of size 128×32. Usually, the images from the dataset do not have exact ly this size, therefore we resize it (without distortion) until it either has a width of 128 or a height of 3 2. Then, we copy the image into a (white) target image of size 128×32. This process is shown in Fig. 3. Finally, we normalize the gray-values of the image which simplifies the task for the NN. Data aug mentation can easily be integrated by copying the image to random positions instead of aligning it to t he left or by randomly resizing the image.

<u>RESULT AND DISCUSSION:</u>

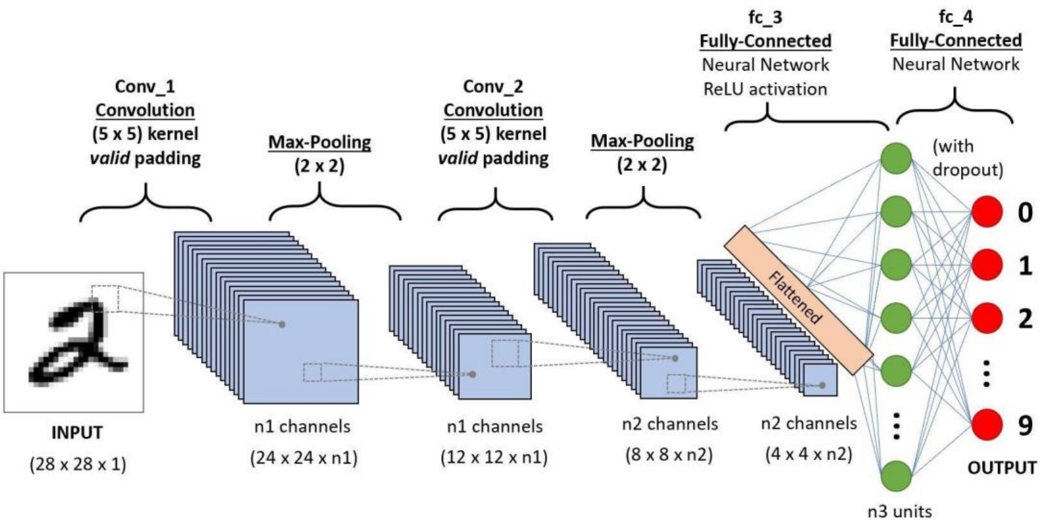
Offline Handwritten Text Recognition (handwritten text recognition) systems transcribe text containe d in scanned images into digital text, an example is shown in Fig. 1. We will build a Neural Network (

NN) which is trained on word-images from the IAM dataset. As the input layer (and therefore also all the other layers) can be kept small for word-images, NN-training is feasible on the CPU (of course, a GPU would be better). This implementation is the bare minimum that is needed for handwritten text r ecognition using TF.

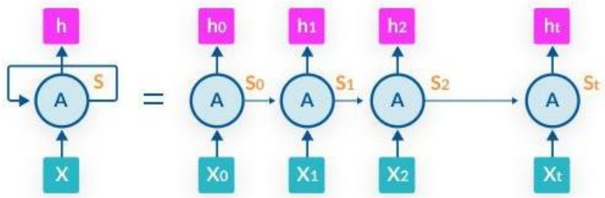
We use a NN for our task. It consists of convolutional NN (CNN) layers, recurrent NN (RNN) layers and a final Connectionist Temporal Classification (CTC) layer. Fig. 2 shows an overview of our hand written text recognition system.

Operations-

CNN: the input image is fed into the CNN layers. These layers are trained to extract relevant features from the image. Each layer consists of three operation. First, the convolution operation, which applies a filter kernel of size 5×5 in the first two layers and 3×3 in the last three layers to the input. Then, the non-linear RELU function is applied. Finally, a pooling layer summarizes image regions and outputs a downsized version of the input. While the image height is downsized by 2 in each layer, feature maps (channels) are added, so that the output feature map (or sequence) has a size of 32×256.



RNN: the feature sequence contains 256 features per time-step, the RNN propagates relevant informat ion through this sequence. The popular Long Short-Term Memory (LSTM) implementation of RNNs is used, as it is able to propagate information through longer distances and provides more robust traini ng-characteristics than vanilla RNN. The RNN output sequence is mapped to a matrix of size 32×80. The IAM dataset consists of 79 different characters, further one additional character is needed for the CTC operation (CTC blank label), therefore there are 80 entries for each of the 32 time-steps.

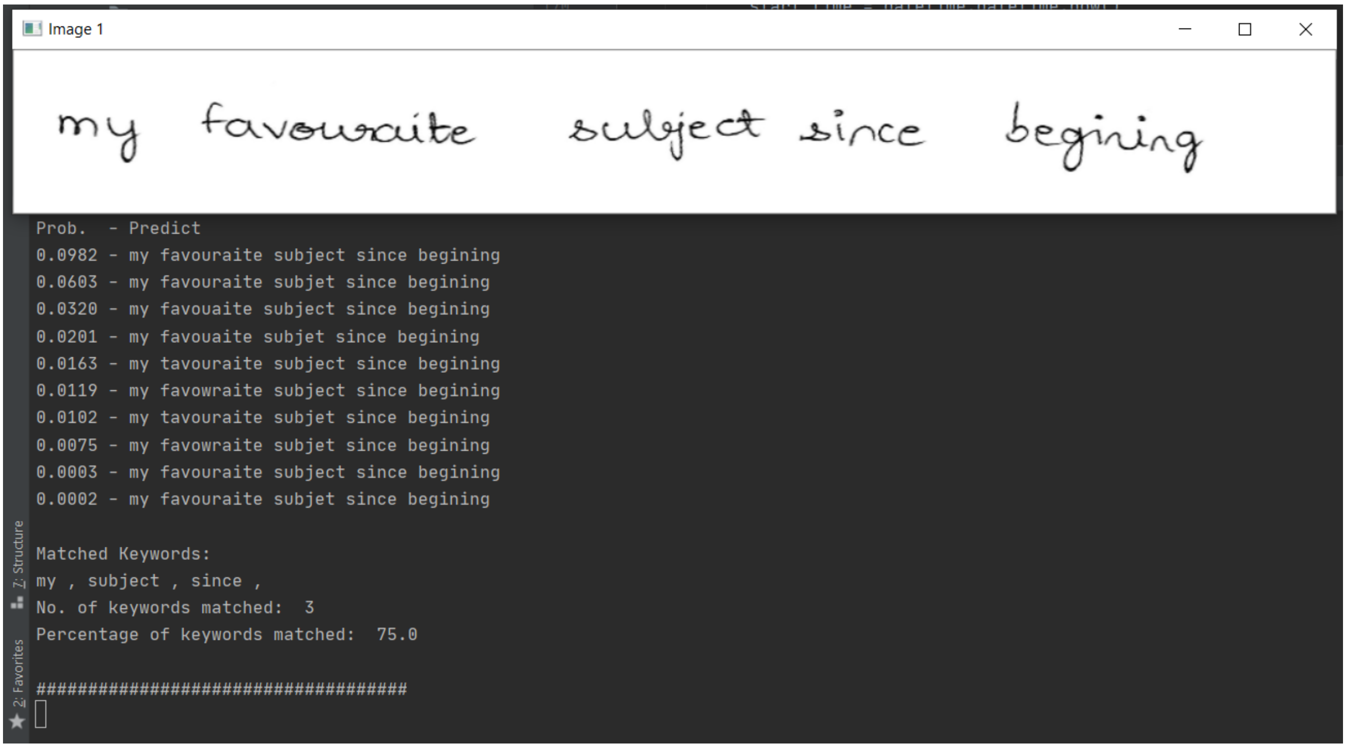


CTC: while training the NN, the CTC is given the RNN output matrix and the ground truth text and it computes the **loss value**. While inferring, the CTC is only given the matrix and it decodes it into the **final text**. Both the ground truth text and the recognized text can be at most 32 characters long.

CNN output: Fig. 4 shows the output of the CNN layers which is a sequence of length 32. Each entry contains 256 features. Of course, these features are further processed by the RNN layers, however, some features already show a high correlation with certain high-level properties of the input image: there are features which have a high correlation with characters (e.g. “e”), or with duplicate characters (e.g. “tt”), or with character-properties such as loops (as contained in handwritten “l”s or “e”s).

RNN output: Fig. 5 shows a visualization of the RNN output matrix for an image containing the text “little”. The matrix shown in the top-most graph contains the scores for the characters including the CTC blank label as its last (80th) entry. The other matrix-entries, from top to bottom, correspond to the following characters: “ !”#&’()*+,-./0123456789:;?ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz”. It can be seen that most of the time, the characters are predicted exactly at the position they appear in the image (e.g. compare the position of the “i” in the image and in the graph). Only the last character “e” is not aligned. But this is OK, as the CTC operation is segmentation-free and does not care about absolute positions. From the bottom-most graph showing the scores for the characters “l”, “i”, “t”, “e” and the CTC blank label, the text can easily be decoded: we just take the most probable character from each time-step, this forms the so called best path, then we throw away repeated characters and finally all blanks: “l---ii--t-t--l-…-e” → “l---i--t-t--l-…-e” → “little”.

Final Result after training-



Prediction error rate-

```
Terminal:  Local  +
Epoch 00005: val_loss improved from 14.28168 to 14.11918, saving model to ..\output\iam\flor\checkpoint_weights.hdf5
336/336 [==============================] - 1112s 3s/step - loss: 13.6473 - val_loss: 14.1192
2020-11-02 10:04:17.809132: W tensorflow/core/kernels/data/generator_dataset_op.cc:103] Error occurred when finalizing GeneratorDataset iterator: Cancelled: Operation was cancelled
2020-11-02 10:04:17.872428: W tensorflow/core/kernels/data/generator_dataset_op.cc:103] Error occurred when finalizing GeneratorDataset iterator: Cancelled: Operation was cancelled
Total train images:      5369
Total validation images: 744
Batch:                   16

Total time:              1:33:36.437795
Time per epoch:          0:18:43.287559
Time per item:           0:00:00.183754

Total epochs:            5
Best epoch               5

Training loss:           13.65284288
Validation loss:         14.11917709

(myenv) (base) D:\Projects\AI\Code\source_main>
TODO  6: Problems  Terminal  Python Console                Event Log
```

Training status-

```
Terminal:  Local  +
2020-11-02 08:30:39.099003: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1096] Device interconnect StreamExecutor with strength 1 edge matrix:
2020-11-02 08:30:39.099152: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1102]
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input (InputLayer)           [(None, 1024, 128, 1)]    0
_________________________________________________________________
conv2d (Conv2D)              (None, 512, 64, 16)       160
_________________________________________________________________
p_re_lu (PReLU)              (None, 512, 64, 16)       16
_________________________________________________________________
batch_normalization (BatchNo (None, 512, 64, 16)       112
_________________________________________________________________
full_gated_conv2d (FullGated (None, 512, 64, 16)       4640
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 512, 64, 32)       4640
_________________________________________________________________
p_re_lu_1 (PReLU)            (None, 512, 64, 32)       32
```

```
Total params: 824,050
Trainable params: 822,770
Non-trainable params: 1,280
_________________________________________________________________
Train for 336 steps, validate for 47 steps
Epoch 1/5
335/336 [============================>.] - ETA: 3s - loss: 14.5654
Epoch 00001: val_loss improved from inf to 14.78593, saving model to ..\output\iam\flor\checkpoint_weights.hdf5
336/336 [==============================] - 1202s 4s/step - loss: 14.5604 - val_loss: 14.7859
```

Validation loss was 14.78% at 1st epoch.

```
Epoch 00005: val_loss improved from 14.28168 to 14.11918, saving model to ..\output\iam\flor\checkpoint_weights.hdf5
336/336 [==============================] - 1112s 3s/step - loss: 13.6473 - val_loss: 14.1192
```

Validation loss decreased to 14.11% after 15 epochs.

```
Epoch 00005: val_loss improved from 14.28168 to 14.11918, saving model to ..\output\iam\flor\checkpoint_weights.hdf5
336/336 [==============================] - 1112s 3s/step - loss: 13.6473 - val_loss: 14.1192
2020-11-02 10:04:17.809132: W tensorflow/core/kernels/data/generator_dataset_op.cc:103] Error occurred when finalizing GeneratorDataset iterator: Cancelled: Operation was cancelled
2020-11-02 10:04:17.872428: W tensorflow/core/kernels/data/generator_dataset_op.cc:103] Error occurred when finalizing GeneratorDataset iterator: Cancelled: Operation was cancelled
Total train images:      5369
Total validation images: 744
Batch:                   16

Total time:              1:33:36.437795
Time per epoch:          0:18:43.287559
Time per item:           0:00:00.183754

Total epochs:            5
Best epoch               5

Training loss:           13.65284288
Validation loss:         14.11917709

(myenv) (base) D:\Projects\AI\Code\source_main>
```

Total time of training: 1.5 hours

Testing Status-

```
  return array(a, dtype, copy=False, order=order)
90/90 [==============================] - 20s 227ms/step
Total test images:     1425
Total time:            0:01:10.097645
Time per item:         0:00:00.049191

Metrics:
Character Error Rate:  0.10257559
Word Error Rate:       0.32261738
Sequence Error Rate:   0.94736842
```

Total testing time- 1 minute and 10 seconds

FURTHER IMPROVEMENT AND SCOPE-

Accuracy can be further increased or loss can be decreased further by training on more number of epo chs. But due to limitation with resources in hardware we were not able to train on more than 5 epochs. So, for further improvement epochs can increase.

Moreover, handwriting recognition is a research area with very less amounts of accuracy achieved so far, our model too has a drawback that it can recognize only one single line of text and will fail if more than one line texts are inputted as images. This is something we aim to work on for the future and incorporate multiple line handwriting recognition in our model.

CONCLUSION :

The paper evaluation model suggested here can do great help for our education system especially in future times as our education systems would change these systems will help reduce teachers time in checking descriptive answers and give them a opportunity to work on more productive things.

The system here has proved to be capable of matching the keywords of the modal answer and awarding marks based on the percentage of matching of these keywords. Hence the said system could be of great utility to the educators whenever they need to take a quick test for revision purpose,as it saves them the trouble of evaluating the bundle of papers.

In future online teaching learning method s will be widely used in many institutions.
Descriptive checking methods will help to evaluate students answer. Our proposed method evaluate it more efficiently and accurately.

In our system, computers have been programmed to scan the papers, recognize the possible right responses and compile the marks. What is different about ‘artificial intelligence’ for exam marking is that platforms can mark complex, open-ended questions designed to test students’ understanding.
Intelligent software can learn to focus on key words in exam answers and to run these against a model answer.

REFRENCES :

[1] C.C. Tappert,C.Y. Suen, T. Wakahara IBM Thomas J. Watson Res. Center, Yorktown Heights, NY, USA "The state of the art in online handwriting recognition" https://ieeexplore.ieee.org/abstract/document/57669

[2] *CH* U.-V. Marti & H. Bunke Department of Computer Science, University of Bern "The IAM-database: an English sentence database for offline handwriting recognition" https://link.springer.com/article/10.1007/s100320200071

[3] Canan Tastimur, Mehmet Karakose and Erhan akin "Improvement of Relative accreditation Methods Based on Data Mining and artificial Intelligence for Higher Education"(2003) https://sci-hub.tw/https://ieeexplore.ieee.org/document/7760736

[4] Piyanush Silapachote and ananta Srisuphab "Teaching and learning computational thinking by solving problems in In AI " (2016**)**

https://sci-hub.tw/https://ieeexplore.ieee.org/document/7851769

[5] Eugene G Kowch and Juhong Christie Liu**"**Principles for Teaching, Leading, and Participatory Learning with a New Participant: AII" (2018)

https://sci-hub.tw/https://ieeexplore.ieee.org/document/8609553

[6] Nigar Ozbey, Mehmet Karakose and aysegul Ucar "The Determination and analysis of Factors affecting to Student Learning by artificial Intelligence in Higher Education"(2010)

https://ieeexplore.ieee.org/document/7760740

[7] Siam Bayne "Teacherbot: interventions in automated teaching"(2015)

https://www.tandfonline.com/doi/full/10.1080/13562517.2015.1020783

[8] Monica Ciolacu ali Fallah Tehrani, Leon Binder and Paul Mugur Svasta"Education 4.0 - artificial Intelligence assisted Higher Education: Early recognition System with Machine Learning to support Students' Success"(2018)

https://ieeexplore.ieee.org/document/8599203

[9[Rory L. L. Sie and Janine Delahunty "artificial Intelligence to Enhance Learning Design in UOW Online, a Unified approach to Fully Online Learning" (2018)

https://sci-hub.tw/https://ieeexplore.ieee.org/document/8615283

[10] Joan Puigcerver Pattern Recognition and Human Language Technology Research Center Universitat Politècnica de València "Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition?" (2017)

http://www.jpuigcerver.net/pubs/jpuigcerver_icdar2017.pdf

[11] Théodore Bluche , Ronaldo Messina "Gated Convolutional Recurrent Neural Networks for Multilingual Handwriting Recognition" (2017)

https://ieeexplore.ieee.org/document/8270042

[12] Bastien Moysset, Ronaldo Messina ”Are 2D-LSTM really dead for offline text recognition?” (2018)

https://arxiv.org/abs/1811.10899

[13] Harald Scheidl, Stefan Fiel , Robert Sablatnig “Word Beam Search: A Connectionist Temporal Classification Decoding Algorithm” (2018)

https://ieeexplore.ieee.org/document/8583770

[14] Dr E.Kamalanaban, M Gopinath, S Premkumar “Medicine Box: Doctor’s Prescription Recognition Using Deep Machine Learning” (2018)

https://www.sciencepubco.com/index.php/ijet/article/view/18785

[15] Arindam Chowdhury, Lovekesh Vig “An Efficient End-to-End Neural Model for Handwritten Text Recognition” (2018)

https://arxiv.org/abs/1807.07965

[16] R. Reeve Ingle, Yasuhisa Fujii, Thomas Deselaers, Jonathan Baccash, Ashok C. Popat “A Scalable Handwritten Text Recognition System” (2019)

https://arxiv.org/abs/1904.09150

[17] Eman Shaikh, Iman Mohiuddin, Ayisha Manzoor, Ghazanfar Latif, Nazeeruddin Mohammad “Automated Grading for Handwritten Answer Sheets using Convolutional Neural Networks” (2019)

https://ieeexplore.ieee.org/document/8923092

[18] Rahul Kr Chauhan, Ravinder Saharan, Siddhartha Singh, Priti Sharma “Automated content grading using machine learning” (2020)

https://arxiv.org/abs/2004.04300

[19] Amit Rokade, Bhushan Patil, Sana Rajani, Surabhi Revandkar, Rajashree Shedge “Automated Grading System Using Natural Language Processing” (2018)

https://ieeexplore.ieee.org/document/8473170

[20] Siddhartha Ghosh, Sameen S. Fatima “Design of an Automated Essay Grading (AEG) system in Indian context” (2008)

https://ieeexplore.ieee.org/document/4766677