ITC 3/54

Information Technology and Control
Vol. 54 / No. 3/ 2025
pp. 1049-1076
DOI 10.5755/j01.itc.54.3.33580

Surveying and Evaluating Artificial Intelligence in Automated Assessment Systems for Pen-and-Paper Tests

Received 2023/03/07 | Accepted after revision 2025/07/17

HOW TO CITE: Jocovic, V., Nikolic, B., Bacanin, N., Bozovic, A. (2025). Surveying and Evaluating Artificial Intelligence in Automated Assessment Systems for Pen-and-Paper Tests. *Information Technology and Control*, 54(3), 1049-1076. https://doi.org/10.5755/j01.itc.54.3.33580

Surveying and Evaluating Artificial Intelligence in Automated Assessment Systems for Pen-and-Paper Tests

Vladimir Jocovic, Bosko Nikolic
School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11000 Belgrade, Serbia

Nebojsa Bacanin
Faculty of Informatics, Singidunum University, Danijelova 32, 11000 Belgrade, Serbia

Aleksandra Bozovic
Academy of Applied Technical Studies, Katarine Ambrozic 3, 11000 Belgrade, Serbia

Corresponding author: jocke@etf.bg.ac.rs

A huge portion of the testing process in most schools is still conducted in a traditional pen-and-paper test manner. This is not only the case in huge courses, large organizations and massive testing events that involve thousands of candidates, but also in smaller schooling communities with insufficient personal computers and human resources. These paper tests are mostly examined manually by the teaching staff, which imposes a significant burden on them. Hence, there was a need for any kind the grading process automatization that would accelerate the assessment process and disburden the teaching personnel. Therefore, software systems for automated assessment of paper tests were developed, partially or fully aiding the teachers in the examination process. These software systems already exhibit some form of artificial intelligence behavior. Artificial intelligence is already providing enormous opportunities for this type of software, and it is simply a matter of time before this software will grade all kinds of tests on its own without human supervision. Due to its recognized importance, this paper provides a detailed analysis and review of available software systems that can be used when assessing pen-and-paper tests.

KEYWORDS: Artificial intelligence, automated test assessment, machine learning, pen-and-paper test

1. Introduction

The increasing presence of personal computers, the Internet and the World Wide Web have made knowledge more accessible and at our fingertips than ever before. Consequently, this led to the development of various online learning systems, for instance Moodle [13], whose goal is to organize knowledge and present it in a more structured manner in the form of courses and lectures. Moreover, these e-Learning platforms [12], [31] started incorporating functionalities that enable the examination process of candidates signed up for a certain course.

These online learning platforms make it possible to simultaneously conduct the examination processes for many candidates. However, most of these platforms do not verify a candidate's true identity besides their login credentials. Furthermore, these systems cannot prevent any intelligent mischief a candidate may try to commit, such as solving the test with an assistance of a close friend or any form of literature, etc. except by disallowing basic unwanted actions (copy/paste online content, for instance).

This does not imply that online learning platforms are useful solely for organizing and presenting knowledge and conducting nonobligatory tests for candidates, and useless for the examination process. They can be used for testing purposes in classrooms with sufficient computing resources and teachers who control the testing process. Moreover, these systems are able to perform the grading process at a much faster rate than human graders.

Nevertheless, many examination processes must be conducted simultaneously for all applicants. The organizers of the examination process have difficulties completing it using online learning platforms, as they are already constrained by several reasons that do not concern the learning platforms themselves. These reasons include the scarcity and unreliability of computing resources, shortage of space to conduct the process, deficit of personnel responsible for monitoring the process and a large number of candidates registered to take the exam. Therefore, even today, it is common for school pupil and student examinations to be conducted using traditional pen-and-paper tests, especially in places without computer access or with many candidates [8]. These paper tests need to be manually graded by the teachers, which imposes a significant burden on the teaching staff. Moreover, manual assessment is prone to errors and takes a lot of time and there are examination processes that require examination results in the shortest period [21]. Hence, there is a need to facilitate and accelerate the examination process by automating it to the greatest extent possible, which also aids in disburdening the teaching staff. Therefore, software systems for automated assessment of paper tests were introduced [5].

To perform the grading process, these systems need to be provided with a digital version of a test that needs to be graded. These systems need to utilize some kind of computer vision to process the data presented in the digital version of the test [36]. Some systems use neural networks to identify regions of interest in the digital image. In contrast, others base their logic on the boundaries, offsets and constraints imposed by the test structure or use specific algorithms to detect certain shapes that designate questions or answers [28]. Therefore, most systems with the purpose of automated assessment of paper tests exhibit some form of artificial intelligence behavior. Nevertheless, it should be noted that although most of these systems are specialized for grading certain types of questions only, they demonstrate exceptional precision and performance. Moreover, some of these systems need separate question and answer sheets to detect regions of interest properly. In contrast, others require questions and answers to be interleaved in the same sheet [18], [34]. Joint questions and answer sheets introduce more noise in detecting regions of interest.

Recognizing the importance of software systems designed for automatic paper test assessment, the key scientific contributions of this work are as follows:

- Identifying the classes and types of questions that may appear in paper tests, along with presenting a table outlining the advantages and disadvantages of using each question type.
- Providing an overview of existing software systems in a uniform manner.
- Analyzing how each of the described systems is used to solve diverse types of questions.
- Making recommendations for a general software solution for automatic paper test grading.

The rest of this paper is organized as follows. Section 2 presents widely used types of questions in a paper test and proposes a classification method. Section 3 describes selected software systems according to the defined template. Section 4 analyzes the chosen software systems according to the classification of question types presented in Section 2. Finally, a brief conclusion and future work are stated in Section 5.

2. Advantages and Disadvantages of Various Question Types

There are diverse types of examination questions that are featured in a paper test. Some include multiple-choice, true-false, short answer, essay, computational, matching, fill-in, etc. There is no strict classification according to the types of questions, as sometimes there are even different names for the same type of question. Moreover, some types of questions partially overlap with other types or even represent a subset or superset of another type.

The proposed classification introduces four types of examination questions: multiple-choice, matching concepts, short answer and essay. True-false, fill-in and computational questions are classified as short answer question type, as they provide a small amount of text for answers. Furthermore, a table of the advantages and disadvantages of using each question type will be presented.

The advantages and disadvantages of Multiple-choice questions are depicted in Table 1.

Table 1

Multiple-choice question advantages and disadvantages. Matching Model.

Advantages	Disadvantages
Human grading can be done fairly quickly.	Human grading is still far slower than any software.
Concisely formulated questions and answers.	The presence of keywords requires only mere familiarity with the content.
Answers cover a broad range of content.	Can be time consuming to design wrong answers.
Answers can be quickly filled in by candidates.	Can facilitate cheating or introduce a dilemma of whether the answer was properly filled in.

The advantages and disadvantages of Matching questions are depicted in Table 2.

Table 2

Matching question advantages and disadvantages.

Advantages	Disadvantages
Human grading can be done fairly quickly but is slower than grading multiple-choice.	Human grading is still far slower than any software.
Reduces the amount of guessing.	Requires only recognition of the relationship between the data on both sides.
Covering maximum knowledge in a minimum amount of space.	Can be time consuming to solve the test.
Answers can be fairly quickly filled in by candidates, but somewhat slower than with the same amount of data in multiple-choice questions.	Lines connecting concepts can introduce a dilemma if the answer was properly filled in or not for many concepts present.

The advantages and disadvantages of Short answer questions are depicted in Table 3.

Table 3

Short answer advantages and disadvantages.

Advantages	Disadvantages
Human grading can be done in a fair amount of time but is slower than grading aforementioned classes of questions.	Flexibility in answering (synonyms, word order, etc.) imposes a burden even for human graders. Needs an NLP algorithm for automated grading.
Significantly reduces the amount of guessing.	Candidates can memorize small parts to answer these types of questions.
Questions hold a certain structure yet allow flexibility in answering for candidates.	Grading is more laborious compared to the aforementioned question classes.
Answers can still be fairly quickly filled in, but significantly slower than in aforementioned question classes.	Handwritten text can sometimes be difficult to recognize with absolute certainty, even in the case of a human grader.

The advantages and disadvantages of Essay questions are depicted in Table 4.

Table 4

Essay advantages and disadvantages.

Advantages	Disadvantages
Questions have almost no structure but allow a decent amount of flexibility in answering for candidates.	Human grading can include a dose of a grader's subjectivity.
Flexibility in the candidate's answering enables human graders to better understand the candidate's reasoning and progress.	Even more flexibility in answering imposes a burden even for human graders. Requires sophisticated NLP algorithms for automated grading.
Engages the candidate's ability to build on his knowledge. Provides the opportunity to thoroughly examine the candidate's understanding of the entire course material and the candidate's ability to integrate individual units of knowledge into a larger whole.	Solving these questions is time consuming for candidates.

3. Overview of Selected Software Systems

After the question classification was finished, the following step was to search for software systems that could automatically grade any of the aforementioned classes of questions. These systems were explored in the open access literature. It is worth mentioning that some authors did not name their software systems, thus the author of this paper did so for easier reference.

3.1. Methodology for Systems' Analysis

Among the numerous software systems encountered, it became necessary to establish filtering, selection, and prioritization criteria. The filtering criteria entailed excluding software systems documented in papers published over a decade ago—the criteria for selection prioritized software systems with higher citation count. Meanwhile, the prioritization criteria favored recently published software systems.

A handful of introductory details are briefly presented for each of the selected software systems. This information includes the name of the software system, the authors' affiliations, the year of publication, the domain of applicability, and the languages of test questions the system can process (or / if there are no restrictions regarding language). The software systems are sorted in the table using two-key criterion: the primary key is the year of publication, and the secondary key is the software system's title. A summary of the information is provided in Table 5.

Subsequently, we systematically presented the chosen tools and endeavored to address the "7W" questions: Who, When, Whom, Where, What, Why, and How. Additionally, we classified them according to the proposed classification scheme. The structure of the introduced template consists of:

- **Summary –** provides brief information regarding the selected software system's origin. This will be the opening of the following subsections and will not be stated in the subsequent bullets.
- **Purpose –** describes the objectives and goals of the selected software system.
- **Structure –** presents the selected software system's coarse grain structure and the main technologies that constitute it.
- **AI fields and algorithms –** lists the fields of Artificial intelligence or algorithms that the selected software system targets.
- **Separate sheets –** important for grading. If specified, it denotes if the question and the answer sheets need to be separated or if questions and answers can be interleaved. Merged sheets impose a challenge because the question text introduces additional noise.
- **Question classes –** states one or more question classes the selected software system can grade. Acronyms for the proposed question classes are: C (multiple-choice), M (matching), A (short answer) and E (essay). Additionally, a letter D will be specified in brackets after the acronym, if the software system can detect that class of questions, besides grading.
- **Evaluation –** portrays advantages and disadvantages of the selected software system, as well as the potential for further innovation.

Table 5
Brief information about selected software systems.

Software system	Affiliation	Year	Question class	Language
eMatura [14]	Department of Computer Science and Information Technology, School of Electrical Engineering, University of Belgrade, Serbia	2023.	Multiple-choice	/
TARS [6]	Department of Computer Science and Information Technology, School of Electrical Engineering, University of Belgrade, Serbia	2021.	Multiple-choice	/
MCQFG [3]	School of Engineering, Edith Cowan University, Perth, Australia	2018.	Multiple-choice	/
MCG-RAS [7]	De La Salle University, Philippines	2017.	Multiple-choice	/
AMCG [24]	Department of Electronics and Telecommunication, K.C. College of Engineering & Management Studies & Research, Thane, Maharashtra, India	2016.	Multiple-choice	/
Eyegrade [4]	Department of Telematic Engineering, University Carlos III of Madrid, Madrid, Spain	2013.	Multiple-choice	/
ASSHEP [19]	School of Electronic and Information Engineering, Foshan University, Foshan, China	2021.	Matching	English, Chinese
AGHAS [29]	Prince Mohammad bin Fahd University, Al Khobar, Saudi Arabia	2019.	Matching	English
ASAGA [1]	Artificial intelligence Department, Faculty of Computers and Artificial Intelligence, Benha University, Benha, Egypt	2022.	Short answer	Arabic
SSSV-LSTM [35]	Information Technologies Division, Adana Alparslan Turkes Science and Technology University, Adana, Turkey	2021.	Short answer	Turkish
SFRN-BERT [16]	Department of Computer Science and Engineering, Pennsylvania State University, United States of America	2021.	Short answer	Chinese, French
ISSHSA [17]	School of Software South China, University of Technology Guangzhou, China; College of Medical Information Engineering, Guangzhou University of Chinese Medicine, Guangzhou, China	2020.	Short answer	Chinese
TM-ASAG [32]	University of Leicester, United Kingdom Lobachevsky University, Nizhni Novgorod, Russia	2020.	Short answer	English
ASHDA [22]	Tokyo University of Agriculture and Technology, Tokyo, Japan The National Center for University Entrance Examinations, Tokyo, Japan	2021.	Essay	Japanese
AEDHA [27]	Centre for Visual Information Technology (CVIT), International Institute of Information Technology, Hyderabad, India	2019.	Essay	English
TCS-AES [9]	TCS Innovation Labs, Kolkata, India	2018.	Essay	English
TS-AAEG [30]	Information Systems Department, Faculty of Computers and Information, Mansoura University, Egypt	2018.	Essay	Arabic
WR-CNN [11]	University of Technology and Design, Singapore	2016.	Essay	English
RNN-AES [33]	Department of Computer Science, National University of Singapore, Singapore	2016.	Essay	English
SSWE-LSTM [2]	University of Cambridge, United Kingdom	2016.	Essay	English

3.2. eMatura

The **eMatura** software system was created at the Department of Computer Science and Information Technology, School of Electrical Engineering, University of Belgrade, Serbia. The project realization lasted throughout the year 2022.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** The software system consists of several modules: detection module, which can detect questions on paper tests, verification module, which is able to verify the existence and the type of detected questions, and grading module, which can grade multiple-choice class of questions. The detection module of this software system utilizes computer vision libraries and image processing algorithms available in Python programming language. The verification module uses in-house developed algorithms combined with algorithms available in the OpenCV library to verify the existence of the previously detected questions and their type. The whole software system is written in Python programming language and its associated libraries.
- **AI fields and algorithms:** Computer Vision, In-house Algorithms.
- **Separate sheets:** Question and their answers can be interleaved so there is no need for separate question and answer sheets.
- **Question classes:** MC (D).
- **Evaluation:**
  - *Advantages:* The software system's precision of 0.999 in grading and performance is its main strength. It requires about 0.25 seconds to process and annotate one test sheet comprising 10 multiple-choice question's rows on average. It supports launching multiple instances to parallelize the grading process and accelerate it greatly. The system does not introduce any restrictions regarding the means by which the test is completed, i.e. both regular and ballpoint pens can be used. Furthermore, the system does not introduce any restrictions regarding the number of question on the test or number and layout of answers in a question. Moreover, it is capable of detecting errors in filling in answers to the multiple-choice questions.
  - *Disadvantages:* Although the system is capable of detecting multiple classes of questions, it only supports grading of multiple-choice class of questions. The system does not allow students to change their answers during testing and labels these questions as one that need manual review.
  - *Further innovation:* The system should possess the capability for students to modify their answers during testing, thereby eliminating the need for manual inspection.

3.3. TARS

The **TARS (Test Answer Recognition System)** software system was created at the Department of Computer Science and Information Technology, School of Electrical Engineering, University of Belgrade, Serbia. The first version of this project was implemented in 2010. Recently, the project has been rebuilt using modern technologies. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** The software system needs a template answer sheet, which comprises 3 parts: student identification region, represented by six 7-segments displays; answers region, represented by a regular matrix of circles, where rows designate question and columns designate answers; and code region, represented by a row of empty and filled circles representing test combination code. All of the 3 regions are filled by students. After the examination process is done, the paper forms are scanned and digital images of tests are obtained. The system utilizes computer vision algorithms in combination with in-house algorithms to perform the recognition process of each of the 3 regions of the template and grade the tests. The whole software system is written in Python programming language and its associated libraries.
- **AI fields and algorithms:** Computer Vision, In-house Algorithms.
- **Separate sheets:** Question and their answers need to be separated on disjoint sheets of paper.
- **Question classes:** MC (D).
- **Evaluation:**

- *Advantages:* The experiment was conducted on 452 paper tests filled by students. The system was able to correctly identify 94.2% student identifications (0.2% were not correctly identified and 5.6% were not correctly filled in by students). The software system's precision in detecting and grading answers was 99.3%, and 100% in test code identification. The test consists of one A4 page with 12 questions and it is processed and annotated in 310ms. The system does not introduce any restrictions regarding the means by which the test is completed, i.e. both regular and ballpoint pens can be used. Moreover, the system allows students to express the intention that they want a certain question not to be graded (i.e. after realizing they filled in the wrong answer).
- *Disadvantages:* The system is sequentially performing the detecting and grading process, although it can be achieved in parallel. The system allows only up to 12 questions and up to 5 answers. This restriction is imposed by the size of the template and the fact that only one paper per test is used. The system does not allow students to change their answers during testing.
- *Further innovation:* The system ought to be endowed with the functionality allowing students to revise their answers during testing, thereby eliminating the need for manual checking. Additionally, the system should be equipped with the capability to assess questions where the answers are inherent within the question, aiming to mitigate errors arising from the process of filling in separate answer sheets by the candidate. These features could contribute to increased efficiency and accuracy, thereby representing innovative improvements in the testing system.

3.4. MCQFG

The **MCQFG (Multiple Choice Questions with Feedback Grader)** software system was created at the School of Engineering, Edith Cowan University, Perth, Australia. The project realization lasted throughout the year 2018.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** The main motivation for developing this system was to reduce the cost and processing restrictions of up to date available systems by taking the advantage of image processing technology. The system enables the user to print the answer sheets and scan them by an ordinary scanner. Additionally, a personal computer can automatically process all the scanned sheets. After scoring, the system annotates the sheets with feedback and sends them back to students via email. Two novel features this system introduced are: segment handwritten character recognition, to recognize students' identification, and a new design of answer sheets, which allows students to change their answers during testing. The whole software system is written in MATLAB programming language and its associated libraries.
- **AI fields and algorithms:** In-house Algorithms.
- **Separate sheets:** Question and their answers need to be separated on disjoint sheets of paper.
- **Question classes:** MC (D).
- **Evaluation:**
  - *Advantages:* The system shows great accuracy in grading, which is 100%, and excellent accuracy in detecting student identification, which is 95%. It utilizes multifunctional scanner/printer rather than highly expensive Optical Marker Readers (OMR). The system has fast processing speed of 0.4 seconds without paper annotation and 2.27 seconds with paper annotation per one answer sheet comprising 72 questions.
  - *Disadvantages:* The authors have stated that user experience needs to be improved. Accuracy results were obtained on rather low number of answer sheets (88).
  - *Further innovation:* As a future innovation, the system should incorporate enhanced UI components to improve the user experience for examiners, making it easier for users to interact with the system seamlessly. The system should possess functionality that enables students to modify their answers during testing, thus eliminating the need for manual checking. Furthermore, it should be equipped with the capability to evaluate questions where the

answers are embedded within the question. This enhancement aims to reduce errors stemming from the process of candidates filling in separate answer sheets.

3.5. MCG-RAS

The **MCG-RAS (Multiple Choice Grader using Readily Available Software)** software system was created at the De La Salle University, Philippines. The project realization lasted throughout the year 2017.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** This software system is composed of the following components: 1) custom printed answer sheet, 2) scanner or camera, 3) Octave scripts, 4) Microsoft Excel or compatible spreadsheet software 5) Email software with mail merge function. The purpose of this system is not to optimize the speed of the OMR process, but to increase its accuracy. Moreover, the system's implementation approach, which takes advantage of scanners and digital cameras with image processing software using readily available software, offers a cheaper alternative. Furthermore, it offers greater flexibility and allows the user to expand functionality. The whole software system is written in Octave programming language and its associated libraries.
- **AI fields and algorithms:** Template Matching.
- **Separate sheets:** Question and their answers need to be separated on disjoint sheets of paper.
- **Question classes:** MC (D).
- **Evaluation:**
  - *Advantages:* The software system shows a high scoring accuracy (98.75%), with only 10 recorded errors out of 800 sheets. The software system is straightforward to use and utilizes regular low-priced scanner rather than highly expensive Optical Marker Readers (OMR).
  - *Disadvantages:* The system has a relatively slow processing speed of 68 seconds per one answer sheet. However, it should be noted that the answer sheet consists of 110 questions. The answer sheet has a fixed and limited number of questions, as well as the number of answers for each of these questions.
  - *Further innovation:* It would be beneficial if the system demonstrated the option for manually designing the answer form. Furthermore, an improvement to the system allowing candidates to modify their answers during the process would be advantageous. The system can be further innovated by exploring enhancements to processing speed without compromising accuracy. Introducing optimizations to the template matching algorithm or exploring parallel processing capabilities could contribute to a more efficient processing time for answer sheets with a fixed and limited number of questions. Additionally, considering advancements in image processing techniques may offer opportunities to maintain high accuracy levels while improving overall processing speed, addressing one of the current system's limitations.

3.6. AMCG

The **AMCG (Automatic Multiple-Choice Grader)** software system was created at the Department of Electronics and Telecommunication, K.C. College of Engineering & Management Studies & Research, Thane, Maharashtra, India. The project realization lasted throughout the year 2016.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** There are three main elements of this software system: the answer sheet, which contains the response grid for questions, the scanner, which is used to capture answer sheets, and the program-based application, which is used to grade specially designed multiple choice questions. The main motivation for developing such a system was to accelerate the process of grading and reduce the cost of the previously available systems that used expensive Optic Marker Readers that could process one answer sheet in roughly 10 minutes, or approximately 144 sheets per day. The system utilizes image processing algorithms available in OpenCV library for C#. The whole software system is written in C# programming language and its associated libraries.
- **AI fields and algorithms:** Computer Vision, In-house Algorithms.

- **Separate sheets:** Question and their answers need to be separated on disjoint sheets of paper.
- **Question classes:** MC (D).
- **Evaluation:**
  - *Advantages:* The software system utilizes regular low-priced scanners rather than highly expensive Optical Marker Readers (OMR). The system requires very low storage space and has a fast processing speed, since it can process around 12 000 answer sheets comprising 150 questions per day. The system does not introduce any restrictions regarding the means by which the test is completed, i.e. both regular and ballpoint pens can be used.
  - *Disadvantages:* The answer sheet has a fixed and limited number of questions, as well as the number of answers for each of these questions.
  - *Further innovation:* The system can be enhanced by exploring adaptive question structures, allowing for flexibility in the number of questions and answers. This innovation could involve implementing algorithms that dynamically adjust to varying question formats, accommodating tests with different lengths and patterns. Furthermore, the integration of machine learning algorithms for question recognition and processing could contribute to a more versatile system capable of handling a broader range of question structures beyond the fixed and limited format currently in place. Additionally, enhancing the system to enable candidates to modify their answers during the process would be beneficial.

3.7. Eyegrade

The **Eyegrade** software system was created at the Department of Telematic Engineering, University Carlos III of Madrid, Madrid, Spain. The project realization lasted throughout the year 2013.

- **Purpose:** This software system's main goal is to detect and grade multiple-choice questions on general purpose tests.
- **Structure:** This software system represents a supervised low-cost solution, which requires only a regular webcam for capturing test page images. After the image is captured, the system executes the following list of steps: 1) Applying morphological transformations on captured image. 2) Applying Optical Character Recognition (OCR) techniques to detect student's identification number from handwritten digits. 3) Detecting answer table geometry using Hough transform. 4) Making decisions and writing results. This software system was implemented using the Python programming language and its standard library, as well as three additional libraries: OpenCV for image capturing, implementation of the Hough transform, thresholding algorithm and mask drawing; Tre for approximate regular expression matching, and Pygame for the user interface.
- **AI fields and algorithms:** Computer Vision, Inhouse Algorithms.
- **Separate sheets:** Question and their answers need to be separated on disjoint sheets of paper.
- **Question classes:** MC (D).
- **Evaluation:**
  - *Advantages:* The software system demonstrates great accuracy in grading (97% correct, 2.4% needed supervision, 0.6% incorrect). The system is portable, since it is available online and requires only a webcam. It is a low-cost system, since it requires only a regular web camera, compared to many systems which rely on scanners to obtain digital images of a test. Also, the system does not introduce any restrictions regarding the number of questions on the test or the number and layout of answers in a question. The system allows students to change their answers during testing.
  - *Disadvantages:* Physical system setup, graphical user interface for configuration. The authors have stated 7 seconds of processing time per one answer sheet with 20 questions, yet it should be noted that this time includes capturing the image and a supervisor manually checking if the system managed to correctly grade the answer sheet.
  - *Further innovation:* In the future, it would be beneficial for the system to minimize reviewer involvement in the evaluation, resulting in less monitoring by supervisors. The system can be improved by implementing real-time processing enhancements to reduce the overall processing time per answer sheet. This

innovation involves optimizing the image capturing and grading steps, possibly leveraging parallel processing capabilities to speed up the system's performance. Additionally, refining the graphical user interface for configuration and simplifying the physical system setup could contribute to a more user-friendly and efficient experience for both administrators and users. Furthermore, exploring machine learning algorithms for adaptive decision-making during grading could enhance the system's accuracy and reduce the need for manual supervision.

3.8. ASSHEP

The **ASSHEP (Automatic Scoring System for Handwritten Examination Papers)** software system was created at the School of Electronic and Information Engineering, Foshan University, Foshan, China. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to detect and grade matching questions on general purpose tests.
- **Structure:** This software system utilizes YOLOv3 algorithms to detect and recognize handwritten numbers and characters on examination papers. These answers can be written anywhere in the question region. Firstly, it was necessary to construct the Examination Paper datasets, to apply deep learning techniques and train the network better. There is a need for an annotation tool to annotate each image of the paper test with the specific location of each of the questions present on that image. After the annotation process is done, a corresponding .xml file is generated for each image, which is then transformed into a txt file. Training is performed in several steps, which include prediction and classification of bounding boxes, cross-scale prediction and feature extraction. The whole software system is implemented in Python programming language and its associated libraries.
- **AI fields and algorithms:** YOLO.
- **Separate sheets:** Question and their answers can be interleaved so there is no need for separate question and answer sheets.
- **Question classes:** M (D).
- **Evaluation:**
  - *Advantages:* The software system is capable of handling problems of incorrect recognition due to scribbles. Moreover, the system can recognize handwritten answers without the need for extra answer cards. Furthermore, the system does not restrict the student in terms of the place where the answer can be written.
  - *Disadvantages:* There are some areas of improvement with a significant potential including more convenient input methods, more accurate character and number segmentation areas and a more efficient and accurate identification method.
  - *Further innovation:* The implementation of a more intelligent and user-friendly system can streamline the user onboarding process, allowing users to initiate their tasks with greater ease. Moreover, the system can be improved by incorporating features that facilitate user engagement and participation in the training process. Introducing interactive elements, such as real-time feedback mechanisms for bounding box predictions, empowers users to actively correct and refine the model's understanding, fostering a collaborative and dynamic training experience. This innovation not only enhances the model's accuracy but also promotes user involvement, contributing to a more effective and user-friendly training interface.

3.9. AGHAS

The **AGHAS (Automatic Grading for Handwritten Answer Sheets)** software system was created in cooperation of the departments of Computer Engineering and Computer Science, Prince Mohammad bin Fahd University, Al Khobar, Saudi Arabia. The project realization lasted throughout the year 2019.

- **Purpose:** This software system's main goal is to detect and grade matching questions on general purpose tests.
- **Structure:** This software system consists of a personal computer, a portable scanner and an application program that can automatically grade the scanned handwritten answer sheets. Firstly, the paper test is scanned using a regular scanner. Secondly, the scanned images are segmented

using MATLAB segmentation code to extract only the handwritten alphabets and numbers, which are fed to the machine learning algorithm. The system uses Convolutional Neural Network (CNN) for detection of handwritten characters. After segmentation, the training data is fed to the CNN models, which are trained to recognize the handwritten answers. The Python scoring script loads the trained CNN model and outputs the score for each student. The whole software system is implemented in Python programming language and its associated libraries and MATLAB programming language.

- **AI fields and algorithms:** Convolutional Neural Network.
- **Separate sheets:** Question and their answers can be interleaved so there is no need for separate question and answer sheets.
- **Question classes:** M (D).
- **Evaluation:**
  - *Advantages:* The conducted experiments shown that the software system exhibits a very high accuracy of 92.86%. This accuracy could be better, but it should be noted that the system uses its own handwritten dataset, which is rather small (250 answer sheets).
  - *Disadvantages:* If some of the scanned images are slightly tilted in orientation, this poses a problem, because the segmentation algorithm considers each pixel value of the template. That means that a slight difference in the orientation of the template causes different pixel values which results in discarding of those scanned images.
  - *Further innovation:* The system can be improved by implementing advanced image preprocessing techniques to handle variations in the orientation of scanned images. This enhancement could involve incorporating algorithms that automatically correct slight tilts and rotations in the scanned images, ensuring more accurate segmentation and recognition. Additionally, expanding the training dataset with a more diverse range of handwritten samples could improve the model's generalization capabilities and overall accuracy. Exploring data augmentation methods, such as introducing variations in orientation during training, might also contribute to the system's robustness in handling different input scenarios. Furthermore, considering the integration of real-time feedback mechanisms for administrators during the grading process could enhance user interaction and provide insights into the system's decision-making.

3.10. ASAGA

The **ASAGA (Automatic Short Answer Grading in Arabic)** software system was created at the Artificial intelligence Department, Faculty of Computers and Artificial Intelligence, Benha University, Benha, Egypt. The project realization lasted throughout the year 2022.

- **Purpose:** This software system’s main goal is to predict the grade of responses of a short answer class of questions on general purpose tests.
- **Structure:** This software system presents a hybrid approach in grading short answer questions that optimizes a deep learning technique called LSTM (Long Short-Term Memory) with a recent optimization algorithm called a Grey Wolf Optimizer (GWO). The purpose of GWO is to optimize the LSTM by selecting the best dropout and recurrent dropout rates of LSTM hyper-parameters rather than manual choice. GWO makes the LSTM model more generalized and can also avoid the problem of overfitting in forecasting the students’ scores. This system utilizes machine learning algorithms and techniques of deep learning to grade short answer responses for questions in Science written in Arabic. The whole software system is written in Python programming language and its associated libraries.
- **AI fields and algorithms:** Machine Learning, Deep Learning.
- **Separate sheets:** /
- **Question classes:** S.
- **Evaluation:**
  - *Advantages:* The software system’s main quality is its precision, as the authors have conducted several experiments with various datasets in which the lowest Pearson’s coefficient was 0.772 and the highest was 0.941. In all experiments, the proposed system showed

better results than the comparison models.

- *Disadvantages:* The limitation of the system is that it does not consider recent language models, such as BERT and its variants. Moreover, training time was higher than in traditional deep learning models.
- *Further innovation:* The system should be capable of extending language support beyond Arabic to accommodate diverse educational contexts. It should improve the user interface to enhance accessibility and user-friendliness, benefiting educators and administrators. Additionally, the system should be capable of exploring adaptive techniques to dynamically determine dropout and recurrent dropout rates in the LSTM model during runtime, enhancing overall model adaptability.

3.11. SSSV-LSTM

The **SSSV-LSTM (SemSpace Sense Vectors and Long Short-Term Memory)** software system was created at the Information Technologies Division, Adana Alparslan Turkes Science and Technology University, Adana, Turkey. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to predict the grade of responses of a short answer class of questions on general purpose tests.
- **Structure:** The system presents a hybrid approach in grading short answer questions that utilizes Manhattan Long Short-Term Memory (MaLSTM) network and the sense representations obtained from concepts on the WordNet lexical-semantic network using the SemSpace method The SemSpace algorithm generates sense vectors for each word sense defined on WordNet using synsets and their relations. Firstly, synset representations of the student's answers and reference answers are given as input into parallel LSTM architecture. Secondly, they are transformed into sentence representations in the hidden layer. Thirdly, the vectorial similarity of these two representation vectors is computed with Manhattan Similarity in the output layer. The whole software system is implemented in Python programming language and its associated libraries and is using WordNet lexical-semantic network.
- **AI fields and algorithms:** Long Short-Term Memory, Natural Language Processing, Machine Learning.
- **Separate sheets:** /
- **Question classes:** S.
- **Evaluation:**
  - *Advantages:* The software system's main quality is its precision, as the authors have conducted several experiments with various datasets in which, mostly, Pearson's coefficient was over 0.95. In all experiments, the proposed system showed better results than the comparison models.
  - *Disadvantages:* During the word sense disambiguation (WSD) process, the increase in both the number of words represented in the context set and the number of ambitious words causes highly increase in processing time.
  - *Further innovation:* The system should be capable of advancing its scalability through the exploration of parallel processing capabilities, ensuring the efficient management of extensive datasets and concurrent processing for heightened overall performance. Additionally, the system should be capable of broadening its language support by extending capabilities beyond WordNet for English, incorporating multilingual resources and semantic networks to enhance applicability across diverse linguistic contexts. Furthermore, the system should implement real-time performance metrics, offering continuous feedback on precision and efficiency to assist users in monitoring and optimizing the system's performance over time.

3.12. SFRN-BERT

The **SFRN-BERT (Semantic Feature-wise transformation Relation Network - Bidirectional Encoder Representations from Transformers)** software system was created at the Department of Computer Science and Engineering, Pennsylvania State University, United States of America. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to predict the grade of responses of a short answer class of questions on general purpose tests.

- **Structure:** The system presents a hybrid approach in grading short answer questions that includes a novel type of relation network called Semantic Feature-wise transformation Relation Network. Firstly, this network learns relational information from QRA (Questions, Reference answers and labeled student Answers) triples. Secondly, it combines the learned representations using learned semantic feature-wise transformations. Thirdly, translation-based data augmentation is applied to address the two problems of limited training data, and high data skew for multi-class automatic short answer grading tasks. This model is combined with a BERT encoder. The authors did not state the technologies and libraries used to implement this software system.
- **AI fields and algorithms:** Relation Networks, Natural Language Processing, Bidirectional Encoder Representations from Transformers.
- **Separate sheets:** /
- **Question classes:** S.
- **Evaluation:**
  - *Advantages:* The software system's main quality is its precision, as the authors have conducted several experiments with various datasets and obtained results that are 8-11% better than the models of the proposed system was compared to.
  - *Disadvantages:* Data augmentation needs to be improved.
  - *Further innovation:* The system should be capable of further innovation by researching and incorporating state-of-the-art data augmentation techniques to enhance the diversity and quality of training data, thereby addressing identified limitations and improving overall performance. Additionally, the system should consider implementing dynamic learning rate adjustment mechanisms to optimize the training process, enabling adaptive modification of learning rates during training and potentially enhancing convergence speed and overall model performance.

3.13. ISSHSA

The **ISSHSA (Intelligent Scoring System for Handwritten Short Answer)** software system was created in cooperation of School of Software South China, University of Technology Guangzhou, China and College of Medical Information Engineering, Guangzhou University of Chinese Medicine, Guangzhou, China. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to predict the grade of responses of a short answer class of questions on general purpose tests.
- **Structure:** This software system consists of specially designed answer cards, scanner capable of capturing photos and a computer or other hardware needed to run the modules of the system. These modules include image preprocessing module, handwriting text recognition module, semantic recognition and comparison module. The image preprocessing module is used for locating the handwritten answer and performs various image manipulation operations to improve text recognition accuracy. The handwriting text recognition module is based on Convolutional Neural Network, which is enhanced by traditional feature extraction methods, such as Gabor or gradient feature maps. The semantic recognition and comparison module is represented by Max-pooling Convolutional Neural Network model. The whole software system is implemented in Python programming language and its associated libraries.
- **AI fields and algorithms:** Deep Learning Network, Semantic Matching.
- **Separate sheets:** /
- **Question classes:** S (D).
- **Evaluation:**
  - *Advantages:* The handwriting text recognition module shows great accuracy which is over 95%.
  - *Disadvantages:* The performance of the semantic recognition and comparison module can be improved, as it achieves an accuracy of about 74%.
  - *Further innovation:* The system should consider incorporating a continuous learning framework, enabling adaptation and improvement over time through interactions with new data. This innovation ensures that the system stays current with evolving handwriting styles and semantic patterns in student responses. Additionally, the system should consider integrating Explainable AI (XAI) techniques, enhancing transparency

in the decision-making process of the semantic recognition and comparison module. This ensures that users gain insights into how the system arrives at its conclusions, fostering a deeper understanding of its operations.

3.14. TM-ASAG

The **TM-ASAG (Text Mining – Automatic Short Answer Grading)** software system was created in cooperation of the University of Leicester, United Kingdom and the Lobachevsky University, Nizhni Novgorod, Russia. The project realization lasted throughout the year 2020.

- **Purpose:** This software system's main goal is to predict the grade of responses of a short answer class of questions on general purpose tests.
- **Structure:** This software system represents a text mining approach to automatically grading short answer questions. Firstly, standard data mining techniques are applied to the corpus of student answers. This is done for the purpose of measuring the similarity between the student answers and the model answer. This similarity is based on the number of common words. Secondly, the evaluation of the relation between these similarities and the marks awarded by the scorers is performed. Thirdly, student answers are grouped into clusters using the K-means clustering algorithm. Each cluster is awarded the same mark, and the same feedback is given to each answer in a cluster, so that the clusters represent the groups of students who are awarded the same or similar scores. Lastly, words in each cluster are compared to show that clusters are constructed based on how many and which words of the model answer have been used. The authors did not state the technologies and libraries used to implement this software system.
- **AI fields and algorithms:** Text Mining, Machine Learning, Natural Language Processing.
- **Separate sheets:** /
- **Question classes:** S.
- **Evaluation:**
  - *Advantages:* The analysis shows that there is a strong relationship between the clusters and the model vocabulary in student answers, as well as the grades.
  - *Disadvantages:* Small datasets were used. Spelling mistakes or synonyms lower the precision.
  - *Further innovation:* The system should be consider enhancing the similarity measurement process by incorporating semantic similarity metrics beyond common word counting. This involves leveraging advanced natural language processing techniques to capture the semantic context of words, ensuring a more nuanced evaluation of student answers. Additionally, the system should be capable of implementing mechanisms to handle synonyms and spelling errors, aiming to enhance precision in grading short answer questions. This innovation entails the integration of natural language processing techniques or external resources to identify and accommodate variations in language usage.

3.15. ASHDA

The **ASHDA (Automatic Scoring of Handwritten Descriptive Answers)** software system was created in cooperation of Tokyo University of Agriculture and Technology and The National Center for University Entrance Examinations, Tokyo, Japan. The project realization lasted throughout the year 2021.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** This software system represents a combined architecture consisting of different deep neural network models for recognizing Japanese handwritten answers and answer sheet grading. Handwritten answer recognition is performed using Convolutional Neural Network, which is a state-of-the-art approach for image classification, modified to adapt to small input images. Automatic scoring is considered as a text classification problem, given that the predicted text is classified into 4 ranks, which are the scores assigned by the examiners. Therefore, Bidirectional Encoder Representations from Transformers (BERT) is used, which is previously trained on Japanese Wikipedia. The authors did not state the technologies and libraries used to implement this software system.
- **AI fields and algorithms:** Convolutional Neural Network, Bidirectional Encoder Representations from Transformers.

- **Separate sheets:** /
- **Question classes:** E (D).
- **Evaluation:**
  - *Advantages:* The experiments are conducted on about 65 thousand answer sheets for the Japanese language tests collected between 2017 and 2018. The handwritten answers model achieved a high character accuracy of over 97%. The automatic scoring model performed high, yielding almost the same results as the human graders.
  - *Disadvantages:* The difference between the automatic scoring model and the human graders tends to rise with the rise of the difficulty of the questions.
  - *Further innovation:* The system's authors should explore the integration of a hybrid model that combines Convolutional Neural Network (CNN) and Bidirectional Encoder Representations from Transformers (BERT) to achieve enhanced performance. This innovation aims to leverage the strengths of both models, providing a more comprehensive solution for recognizing handwritten answers and scoring. Additionally, the system should be capable of extending its capabilities to support recognition and grading for languages beyond Japanese. This involves incorporating multilingual pre-trained models for both image recognition and text classification, thereby expanding the system's applicability to a broader range of language assessments.

3.16. AEDHA

The **AEDHA (Automatic Evaluation of Descriptive Handwritten Answers)** software system was created at the Centre for Visual Information Technology (CVIT), International Institute of Information Technology, Hyderabad, India. The project realization lasted throughout the year 2019.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** This software system presents a self-supervised, feature-based classification approach in automatically detecting and grading descriptive handwritten answers from the digitalized images, unlike traditional non-semantic approaches. Semantic analysis for auto-evaluation in handwritten text answers is performed using the combination of Information Retrieval and Extraction (IRE) and Natural Language Processing (NLP) methods to derive a set of useful features. To automatically determine the evaluation score, the system detects the keywords present in the student's handwritten answer. These keywords include keywords from the textual reference answer and semantically relevant keywords, which are obtained using IRE and NLP methods. The whole software system is written in Python programming language and its associated libraries.
- **AI fields and algorithms:** Neural Networks, Natural Language Processing, Information Retrieval and Extraction.
- **Separate sheets:** Question and their answers can be interleaved so there is no need for separate question and answer sheets.
- **Question classes:** E (D).
- **Evaluation:**
  - *Advantages:* The software system is capable of detecting and evaluating handwritten answers with a precision that is highly correlated to human examinations.
  - *Disadvantages:* The system struggles to segment text with improper wording, text highlighting using boxes, less spacing between words, high skew and excessive word scribbling. These are add up in word count, which effects the final scores. Answers paraphrased with simple non-technical terms are also found relatively hard to evaluate.
  - *Further innovation:* The system should be capable of enhancing its evaluation of paraphrased answers with simple non-technical terms by incorporating contextual analysis. This involves leveraging natural language processing methods to deepen its understanding of the context and meaning behind paraphrased content, ensuring a more nuanced and accurate evaluation. Additionally, the system should be capable of implementing a dynamic keyword expansion mechanism, allowing it to adaptively identify

semantically relevant keywords beyond the initial set obtained from Information Retrieval and Extraction (IRE) and Natural Language Processing (NLP) methods. This innovation ensures flexibility in capturing diverse and contextually relevant keywords, further enhancing the system's accuracy in evaluation.

3.17. TCS-AES

The **TCS-AES (Tata Consultancy Services – Automatic Essay Scoring)** software system was created at the TCS Innovation Labs, Kolkata, India. The project realization lasted throughout the year 2018.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** The system uses an enhanced deep convolutional recurrent neural network (CNN) connected with a bidirectional long short-term memory (LSTM) model for automatic grading of essay question class. The convolutional neural network comprises five layers: Embedding, Convolution, Long short-term memory, Activation and Sigmoid activation. Besides considering the words and sentence representation in a text, the system augments the different complex linguistic, cognitive and psychological features associated with a text. The whole software system is written in Python programming language and its associated libraries.
- **AI fields and algorithms:** Convolutional Neural Network, Long Short-Term Memory.
- **Separate sheets:** /
- **Question classes:** E.
- **Evaluation:**
  - *Advantages:* The software system's main quality is its precision, which is stated in 0.94 Pearson's and 0.97 Spearman's coefficients.
  - *Disadvantages:* The limitation of the system is that all the linguistic and qualitative features used are computed offline and then fed into the deep neural network learning architecture, and not deduced by the network.
  - *Further innovation:* The system should be capable of innovating through the implementation of real-time feature extraction mechanisms, allowing the deep neural network to deduce linguistic, cognitive, and psychological features directly during the learning process. This real-time approach enhances the system's adaptability and responsiveness to variations in essay content. Additionally, the system should be capable of developing an interactive learning architecture that dynamically adjusts its linguistic and qualitative feature extraction based on ongoing interactions. This innovation fosters a more adaptive and context-aware grading process, capturing nuances in essay responses that may evolve over time, thereby improving the overall grading accuracy and relevance.

3.18. TS-AAEG

The **TS-AAEG (Text Similarity – Automatic Arabic Essay Grading)** software system was created at the Information Systems Department, Faculty of Computers and Information, Mansoura University, Egypt. The project realization lasted throughout the year 2018.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** This software system measures the similarity of student answer by comparing each word in the model answer with each word in the student answer. This comparison is performed using a bag of words model and the similarity values are obtained in the following steps: 1) Tokenization, where the text is divided into sentences and sentences into tokens. 2) Stopwords, where the words that do not convey significant meaning in measuring similarity are removed. 3) Stemming, where the lexical root (stem) for words is found, by removing affixes (prefixes, suffixes and postfixes) attached to the root of the word. The system uses N-gram approach, which slides a window of length n over a string to generate grams of length n that are utilized in the matching process. The authors did not state the technologies and libraries used in the implementation of this software system.
- **AI fields and algorithms:** Natural Language Processing.

- **Separate sheets:** /
- **Question classes:** E.
- **Evaluation:**
  - *Advantages:* The N-gram approach is simple, more reliable for noisy data (grammatical errors, misspellings, etc.) and outputs more N-grams, which lead to collecting more N-grams that are significant in similarity measurement.
  - *Disadvantages:* The combination of string and corpus algorithms would achieve higher results and decrease automatic grading errors.
  - *Further innovation:* The system can be improved by exploring advanced techniques in hybrid algorithm integration, combining string-level (such as the N-gram approach) and corpus-level approaches. This enhancement would further optimize accuracy in similarity measurement, addressing noise in data like grammatical errors and misspellings. Additionally, the system can be improved by implementing a dynamic stopwords identification mechanism that adapts to each essay's specific context. This improvement ensures more contextually relevant removal of stopwords, resulting in a more precise similarity measurement that emphasizes words carrying significant meaning in the given context.

3.19. WR-CNN

The **WR-CNN (Word Representations – Convolutional Neural Network)** software system was created at the University of Technology and Design, Singapore. The project realization lasted throughout the year 2016.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** The model contains two parts: Words Representations and a two-layer convolutional neural network (CNN) structure. The convolutional layer is used to extract representations of sentences, while the other layer is stacked on sentence vectors and used to learn essay representations. The used model does not rely on POS-tagging or other external pre-processing methods. The authors did not state the technologies and libraries used in the implementation of this software system.
- **AI fields and algorithms:** Convolutional Neural Network, Word Embedding Matrix.
- **Separate sheets:** /
- **Question classes:** E.
- **Evaluation:**
  - *Advantages:* The software system exhibits a confidence level of 95% for in-domain experiments.
  - *Disadvantages:* Confidence level is lower for cross-domain experiments compared to the one obtained by in-domain experiments.
  - *Further innovation:* The system can be improved by further investigating transfer learning strategies, emphasizing the exploration of diverse related tasks or domains for pre-training. This comprehensive approach enhances the model's ability to leverage knowledge gained from various contexts, thereby potentially boosting performance and confidence not only in in-domain scenarios but also in cross-domain evaluations. Additionally, the system can benefit from the development of a dynamic convolutional layer configuration, adapting to varying sentence structures and lengths. This innovation ensures that the convolutional layer remains effective in extracting meaningful sentence representations across different essay characteristics, ultimately contributing to improved overall performance and versatility.

3.20. RNN-AES

The **RNN-AES (Recurrent Neural Network – Automatic Essay Scoring)** software system was created at the Department of Computer Science, National University of Singapore, Singapore. The project realization lasted throughout the year 2016.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** The system uses a recurrent neural network (RNN) approach, which does not rely on any feature engineering and automatically discovers relations between input essays and

output grades. This network model architecture comprises five layers: Lookup table, Convolution, Recurrent, Mean over Time and Linear Layer with Sigmoid Activation. The authors did not state the technologies and libraries used to implement this software system.

- **AI fields and algorithms:** Recurrent Neural Network.
- **Separate sheets:** /
- **Question classes:** E.
- **Evaluation:**
  - *Advantages:* The implemented model is able to properly learn the task and is competent with other baseline models.
  - *Disadvantages:* Analysis shows that the system, which consists of an ensemble of RNN and LSTM models, performs significantly better than the RNN model.
  - *Further innovation:* The system can be improved by delving deeper into the ensemble learning approach, seeking ways to refine the collaboration between the recurrent neural network (RNN) and long short-term memory (LSTM) models. Optimizing the integration of these models within the ensemble should be undertaken, with a focus on achieving a more synergistic performance that surpasses the current results. Additionally, further exploration of attention mechanisms within the recurrent neural network architecture is essential. This innovation has the potential to enhance the model's capacity to concentrate on specific parts of input essays during the learning process, thereby improving its efficacy in capturing intricate relationships between essay content and grading outcomes.

3.21. SSWE-LSTM

The **SSWE-LSTM (Score-Specific Word Embedding – Long Short-Term Memory)** software system was created at the University of Cambridge, United Kingdom. The project realization lasted throughout the year 2016.

- **Purpose:** This software system's main goal is to compute the quality of answers to essay class of questions on general purpose tests.
- **Structure:** The system uses deep learning neural network model which automatically discovers its features. The model contains two parts: Score-Specific Word Embedding connected with a two-layer Bidirectional Long Short-term Memory (LSTM) model for automatic grading of essay question class. The creators of the model augmented their model not only to apprehend the local linguistic environment of each word, but also to capture how each word contributes to the overall score of the essay. To apprehend SSWEs, the model was enhanced by adding a linear unit in the output layer that performs linear regression, thus predicting the essay score. The authors did not state the technologies and libraries used in the implementation of this software system.
- **AI fields and algorithms:** Long Short-Term Memory, Score Specific Word Embedding.
- **Separate sheets:** /
- **Question classes:** E.
- **Evaluation:**
  - *Advantages:* The software system's main quality is its precision, which is stated in 0.96 Pearson's and 0.91 Spearman's coefficients.
  - *Disadvantages:* The limitation of the system is that if a word appears multiple times within an essay, sometimes correctly and sometimes incorrectly, the model would not be able to distinguish between them.
  - *Further innovation:* The system can be improved by further innovating dynamic weighting mechanisms for Score-Specific Word Embeddings (SSWEs). This enhancement could involve developing an adaptive learning mechanism that dynamically adjusts the weights assigned to words based on their evolving contribution to the overall essay score during the learning process. Additionally, the model's capability to distinguish between multiple occurrences of a word within an essay can be refined by exploring advanced contextual word disambiguation techniques. Leveraging context-aware disambiguation methods ensures a more accurate differentiation between correct and incorrect instances of the same word, thereby enhancing the model's precision in understanding nuanced linguistic contexts.

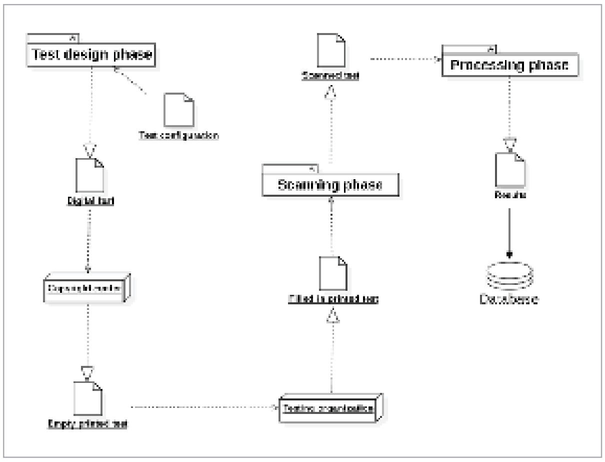
4. Analysis and Discussion

Although the selected software systems are intended for a certain class of questions, regularity was observed in the order of the phases that represent the implementation of the entire examination process - from generating tests to obtaining results. The implementation of certain process phases varies from system to system, yet their order of execution remains the same. This enabled the creation of a generalized flow diagram of the phases required to process the introduced classes of questions.

Firstly, given the requirements recorded in the configuration file, the digital form of the test is created in the test design phase. Afterward, this test is transferred in digital form to the test copying centers and based on it, as a result, printed blank tests are created in the required number. These empty printed tests are then delivered to the institution in charge of the candidate testing process. After the testing in an organization is completed, paper tests completed by candidates are produced as a result. Then these completed paper tests are delivered to the scanning phase, where the completed tests are obtained in digital form with the help of a scanner machine or camera. Such tests represent the entrance to the test processing phase, which consists of two parts: the phase of detecting candidate questions and answers and the phase of processing detected questions and candidate answers. This processing phase differs the most in the selected analyzed software systems, and in the following subsections attention will be devoted to similarities and differences precisely in this phase. At the end, the results obtained for each candidate in this phase are entered into the database. The described process is illustrated in Figure 1.

The analysis of the selected software systems has shown that each system is specialized to grade only one of the presented question classes. This allowed for observing certain similarities in the given software systems in the same class. These similarities are not only observed in the technologies used, but also in the very process of detection and grading of certain classes of questions. However, among software systems that can automatically score the same class of questions, there are different approaches to detecting and scoring, even for those systems that use the same technology. These similarities and differences are grouped, for each of the previously introduced question classes. Moreover, their accuracy, performance and constraints are discussed, as well. Qualitative comparison of the software systems is shown in Table 6.

Figure 1
Generalized flow diagram.



4.1. Multiple-choice

The selected systems described in Section 3 have stated their artificial intelligence field. They utilize various computer vision algorithms in the processing phase. All of the systems stated the technologies they used and are implemented in one of the following programming languages: Python, C#, MATLAB, or Octave.

Each selected system, except the MCG-RAS system, uses the OpenCV library alongside in-house developed algorithms to detect various shapes (i.e., circles, rectangles) that represent questions and answers regions. Some of them perform the recognition process on a template test and translate the obtained questions and answers schema to completed tests, aligning the template according to the borders of the test. On the other hand, others solely perform the detection process on completed tests. The MCG-RAS system uses template matching techniques to detect regions of interest.

All selected systems, except the AMCG system, exhibit a high accuracy of at least 95% (MCQFG system) in grading. In comparison, others achieve accuracy close to 100% (eMatura 99.9%, Eyegrade 99.4%, TARS 99.3%, MCG-RAS 98.75%). The quickest of them, eMatura, can process one test sheet in about 250ms,

Table 6
Qualitative comparison of the software systems.

Software system	Structure	Algorithms	Separate sheets	Advantages	Disadvantages
eMatura [14]	Test designer, detection, verification, and grading modules (Python)	Computer Vision, In-house Algorithms	No	Precision, no test format restrictions, parallel processing.	No changing answers.
TARS [6]	Predefined answer sheets, scanner, modules (Python)	Computer Vision, In-house Algorithms	Yes	Allows response undo.	Sequential processing, single answer undo restriction.
MCQFG [3]	Predefined answer sheets, scanner, modules (MATLAB)	In-house Algorithms	Yes	Allow response change, accuracy.	User experience, limited number of questions and answers.
MCG-RAS [7]	Predefined answer sheets, scanner or camera, modules (Octave)	Template matching	Yes	User-friendly and cost-effective.	Slow processing rate, constrained question and answer count.
AMCG [24]	Test designer, scanner, and grading module (C#)	Computer Vision, In-house Algorithms	Yes	Efficient processing with minimal storage.	Restricted number of questions and answers.
Eyegrade [4]	Custom answer sheet, webcam, grading and annotation module (Python)	Computer Vision, In-house Algorithms	Yes	Allows response change, portable, budget-friendly.	Physical system setup, graphical user interface.
ASSHEP [19]	Detection, annotation and grading modules (Python)	Deep Learning, YOLO	No	Scribble recognition robust, flexible answer placement allowed.	More convenient input methods, more accurate character and number segmentation areas.
AGHAS [29]	Personal computer, scanner and modules (Python, MATLAB)	Convolutional Neural Network	No	High accuracy in grading.	Scanned image tilt impacts accuracy.
ASAGA [1]	Long short-term memory with Grey Wolf optimization module (Python)	Machine Learning, Deep Learning	/	High precision in grading.	Lack of consideration of recent language models.
SSSV-LSTM [35]	Hybrid Manhattan Long Short-Term Memory with SemSpace algorithm (Python)	Natural Language Processing, Machine Learning	/	Precision in grading is excellent.	Processing time escalates with contextual complexity.
SFRN-BERT [16]	Hybrid Semantic Feature-wise transformation Relation Network with BERT	Natural Language Processing, Bidirectional Encoder Representations from Transformers	/	Achieves superior precision in comprehensive experiments.	Data augmentation needs improvement.

Software system	Structure	Algorithms	Separate sheets	Advantages	Disadvantages
ISSHSA [17]	Custom designed answer cards, scanner, personal computer, modules (Python)	Deep Learning Network, Semantic Matching	/	Boasts exceptional accuracy.	Enhancement needed for semantic recognition.
TM-ASAG [32]	Personal computer, detection modules	Text Mining, Machine Learning, Natural Language Processing	/	Strong correlation between clusters and vocabulary in students' answers.	Sensitive to spelling mistakes or synonyms usage, which lower the precision.
ASHDA [22]	Detection and grading modules	Convolutional Neural Network, Bidirectional Encoder Representations from Transformers	/	High character accuracy in handwritten answers. Automated scoring parallels human graders closely.	Increasing divergence between the model and human graders in challenging question scoring.
AEDHA [27]	Combination of Information Retrieval and Extraction and Natural Language Processing for grading module (Python)	Neural Networks, Natural Language Processing	No	The system accurately evaluates handwritten answers, correlating closely with human assessments.	Challenges include text segmentation issues, skewed formatting, and difficulty assessing paraphrased answers.
TCS-AES [9]	Deep convolutional recurrent neural network connected with a bidirectional long short-term memory model (Python)	Convolutional Neural Network	/	High precision in grading.	Offline computation hinders dynamic feature deduction within neural networks.
TS-AAEG [30]	Bag of words model, N-gram approach	Natural Language Processing	/	N-gram approach excels in noisy data, ensuring robust similarity measurement.	Hybrid string and corpus algorithms would enhance grading precision.
WR-CNN [11]	Words Representations and a two-layer convolutional neural network structure	Convolutional Neural Network	/	Confidence in grading is excellent for in domain experiments.	Confidence level is lower for cross-domain experiments
RNN-AES [33]	Regular recurrent neural network approach	Recurrent Neural Network	/	Competent with other baseline models.	Ensemble of RNN and LSTM models, performs significantly better than the RNN model
SSWE-LSTM [2]	Hybrid Score-Specific Word Embedding - two-layer Bidirectional Long Short-term Memory model	Deep Learning	/	High accuracy in grading.	Word ambiguity hinders accurate multiple occurrences differentiation in essays.

while the slowest requires several tens of seconds. However, it should be noted that answer sheets vary in the number of questions and answers. Moreover, some systems restrict the number of questions and answers per test sheet. All systems, except the eMatura system, require that questions and their answers need to be separated on disjoint sheets of paper.

In our opinion, the eMatura system has shown the best results. The system records the lowest processing time of all the selected systems capable of grading multiple-choice questions. The system exhibits impressive accuracy in grading. Furthermore, it can detect errors in filling in answers to multiple-choice questions. Lastly, it allows questions and answers to be interleaved on the same sheet.

Comparing execution speed performance equitably presents several challenges. Firstly, the test configurations utilized by the authors differ in capabilities. Secondly, most solutions lack available source code, making it challenging to gauge performance accurately on diverse arbitrary setups. Additionally, the variations in the number of questions per test page, particularly due to the time-intensive morphological image processing operations, and the quantity of answer choices further compound this complexity.

However, to provide a rough performance comparison of the selected systems, the authors took the measures available and conducted an evaluation. These metrics are provided exclusively for software systems capable of assessing multiple-choice questions, while they are absent for systems evaluating other question types. This evaluation employed the time required to process a single multiple-choice question as a metric, as displayed in Table 7.

Table 7

Processing time per annotated question.

System	Time per sheet (ms)	Questions per sheet	Time per question (ms)
eMatura	250	10	25
TARS	310	12	25.83
MCQFG	2 270	72	31.5
MCG-RAS	68 000	110	618.2
AMCG	7200	150	48
Eyegrade	7 000	20	350

4.2. Matching

There are not so many systems capable of grading matching class of questions. Both selected systems described in Section 3 have stated their artificial intelligence field. They utilize convolutional neural networks in the processing phase. The ASSHEP system is implemented using Python programming language, while the AGHAS system uses MATLAB programming language.

Each selected system uses Python libraries for convolutional neural networks alongside in-house developed algorithms to detect questions and answers regions. Both of them perform the recognition process on the completed test solely. The ASSHEP system uses the YOLO convolutional network, while the AGHAS system constructs its convolutional neural network.

The AGHAS system exhibits a high accuracy of at least 92% in grading. The ASSHEP system did not state the achieved accuracy in grading. None of the selected systems have stated the time required to process one test sheet. None of the chosen systems restrict the number of questions and answers per test sheet. Moreover, none of them requires that question and their answers need to be separated on disjoint sheets of paper.

In our opinion, the ASSHEP system has shown better results, as it can manage problems of incorrect recognition due to scribbles. Moreover, the system does not restrict the student in terms of where the answer can be written. Also, the system supports additional languages besides English. On the other hand, the AGHAS system is facing problems if some of the scanned images of tests are slightly tilted in orientation, resulting in discarding those tests.

4.3. Short Answer

The selected systems described in Section 3 have stated their artificial intelligence field. They utilize various artificial intelligence techniques, i.e., machine learning, natural language processing, deep learning, relation networks, long short-term memory, bidirectional encoder representations from transformers, semantic matching and text mining. Not all systems have stated the technology they used for system implementation, yet the ones that did (ASAGA, SSSV-LSTM, ISSHSA) are implemented in Python.

Two systems, ASAGA and ISSHSA, use convolutional neural networks to detect handwritten short answers on paper tests, before performing the automatic grading process. On the other hand, other systems automatically grade the already obtained text from digital images or input forms, thus not including the step of recognizing handwritten text from images of paper tests.

Not all of the systems have stated the achieved accuracy in grading. ASAGA shows a Pearson score in the range of 0.77-0.95, SSSV-LSTM shows 0.95, while ISSHSA exhibits 95% accuracy in grading. None of the systems have stated the time required to process the test.

In our opinion, the ISSHSA system has shown the best results. The system exhibits impressive accuracy in grading. Furthermore, it incorporates the handwriting text recognition module, which is assisted by the preprocessing module, used for locating the handwritten answer and performing various image manipulation operations to improve text recognition accuracy.

4.4. Essay

The selected systems described in Section 3 have stated their artificial intelligence field. They utilize various artificial intelligence techniques, i.e., neural networks, recurrent neural networks, convolutional neural networks, long short-term memory, natural language processing, bidirectional encoder representations from transformers, information retrieval and extraction, word embedding matrix, and score specific word embedding. Not all systems have stated the technology they used for system implementation, yet the ones that did (AEDHA, TCS-AES) are implemented in Python.

Two systems, ASHDA and AEDHA, use convolutional neural networks to detect handwritten essay answers on paper tests, before performing the automatic grading process. On the other hand, other systems automatically grade the already obtained text from digital images or input forms, thus not including the step of recognizing handwritten text from images of paper tests.

Not all of the systems have stated the achieved accuracy in grading. TCS-AES shows 0.94 Pearson score, SSWE-LSTM shows 0.96, while WR-CNN and ASHDA exhibit 95% and 97% accuracy in grading, respectively. None of the systems have stated the time required to process the test.

In our opinion, the ASHDA system has shown the best results. The system exhibits impressive accuracy in grading, outperforming the other systems. Furthermore, it incorporates the handwriting text recognition module. Although the AEDHA system also performs handwritten text recognition, it struggles to segment text with improper wording, text highlighting, less spacing, high skew and excessive word scribbling.

4.5. Proposal of the New System

The architecture of the proposed system closely resembles the flow control depicted in Figure 1. It suffices to explain the question processing phase, which varies among different systems. The proposed system is proficient in evaluating multiple-choice, matching, and short-answer questions [15]. The configuration file for each question type conveys information about its type.

The system detects the questions' and answers' regions for multiple-choice questions and cross-references this information with the test configuration file. Inside the response area, it identifies a regular grid of circles representing the provided responses. Subsequently, it determines the level of shading in these circles, representing the chosen answers.

Regarding matching questions, the system identifies the questions' and answers' regions represented by lines used to connect concepts. It validates these regions against the data in the test configuration file. It then removes the lines for writing the answers and reconstructs the symbols, particularly if they intersect with the answer lines. A pre-trained neural network is subsequently employed to recognize the typed symbols, serving as the processing result.

Short-answer questions involve detecting the questions' and answers' regions, which are demarcated by short-answer entry lines. The system cross-references this information with the test configuration file. It eliminates the short-answer writing lines and reconstructs the written text, should it overlap with these lines. A pre-trained deep neural network is employed to identify the written short answers, which are returned as a result.

For essay questions, which entail longer textual responses, additional artificial intelligence techniques, such as natural language processing, are requisite to establish relationships between written words and derive the meaning of the text. The system is designed to accommodate future extensions, allowing new text evaluation modules to be incorporated. Already, the system is proficient in text detection.

5. Evaluation and Discussion

Nowadays, paper-based tests are still widely used [26], [20]. This is why it is of significant importance to implement the capability of automated assessment [10]. Although there are diverse types of questions, they were grouped into four classes: multiple-choice (MC), matching (M), short answer (S), and essay (E). The previous sections thoroughly analyzed these classes of questions and software systems capable of automatically assessing them.

Contemporary advancements in artificial intelligence have opened the door to the possibility of completely automating the assessment of paper-based tests. The systems under examination are underpinned by algorithms drawn from various domains within artificial intelligence [23]. Notably, these encompass Computer Vision, Machine Learning, Deep Learning, Long Short-Term Memory, Natural Language Processing, Convolutional Neural Networks, Recurrent Neural Networks, Template Matching, Semantic Matching, Text Mining, Information Retrieval and Extraction, and Word Embedding Matrix. Additionally, the authors employed established implementations, including YOLO and BERT, alongside proprietary algorithms they developed in-house.

A software system that provides the best results has been identified for each of the identified classes of questions. Among the systems that can grade multiple-choice class questions, eMatura exhibited the highest accuracy and the shortest processing time. The system ASSHEP is well-suited for matching question class, as it is resistant to scribbled answers, allows answers to be placed in arbitrary locations within the question region, and is multilingual. The ISSHSA system is the only system capable of recognizing handwritten answers for the short answer question class and digital forms of short answers. The ASHDA system has shown the best results with impressive accuracy for essay class grading, outperforming other systems.

The implementation of automated assessment systems, as discussed in the analysis, presents several practical applications and challenges that are crucial in shaping the future of educational practices and assessment policies [25]. Here, we delve into key aspects such as scalability, cost implications, user-friendliness, and the potential impact on educational practices, as displayed in Table 8.

The primary limitation of this analysis is the inability to test the described systems under uniform conditions, encompassing a wide range of possibilities and intricacies, for direct result comparisons. Consequently, it is not feasible, given the data at hand, to assess and contrast the execution performances of these systems. Moreover, for questions necessitating the processing of responses in natural languages, most systems are geared toward English, with fewer representations for languages with limited resources. Too, many of these systems rely on proprietary algorithms, the specifics of which are not consistently disclosed.

In exploring the current landscape of automated assessment, the analysis has uncovered notable gaps that beckon further investigation and innovation. One conspicuous void lies in the integration of diverse question classes, as existing systems predominantly focus on specific types. There's a compelling need for research that ventures into the development of a unified platform capable of seamlessly handling multiple question classes, fostering a more comprehensive automated assessment tool.

Multilingual inclusivity remains another prominent gap, with many systems primarily tailored for the English language. Future research endeavors should concentrate on adapting and expanding these systems to support various languages, ensuring a global relevance that transcends linguistic barriers. Additionally, while objective assessments have seen remarkable advancements, subjective evaluations often lack explanatory depth. Addressing this gap calls for the infusion of Explainable AI (XAI) techniques, enhancing transparency and providing detailed justifications for subjective grading decisions.

A critical area demanding attention is the development of adaptive learning systems that integrate

Table 8

Practical applications and challenges.

Aspect	Practical Applications	Challenges
Scalability	**Efficient Grading Workflow:** Automated assessment systems offer the potential to streamline grading processes, allowing educators to handle a large volume of assessments swiftly and accurately. **Consistency and Standardization:** Scalable automated systems contribute to the maintenance of consistency and standardization in grading, reducing the likelihood of subjective variations across different assessors.	**Adaptation to Diverse Formats:** The challenge lies in developing systems that can seamlessly adapt to diverse question formats, especially as educational assessments evolve beyond traditional structures. **Processing Speed:** Ensuring that the scalability of these systems does not compromise processing speed is critical for maintaining their efficiency, particularly in high-stakes testing scenarios.
Cost implications	**Resource Optimization:** Automated assessment systems have the potential to optimize resources by reducing the time and manpower required for manual grading, thereby potentially lowering overall costs. **Accessible Technology:** As technology becomes more affordable, these systems can become a cost-effective alternative to traditional grading methods.	**Initial Implementation Costs:** Developing and implementing robust automated assessment systems may involve significant upfront costs, posing a challenge for institutions with limited budgets. **Maintenance and Updates:** Ongoing maintenance and updates to keep the systems relevant and secure could contribute to long-term costs.
User-friendliness	**Enhanced Efficiency for Educators:** User-friendly interfaces can empower educators by providing intuitive tools that require minimal training, making the integration of these systems into existing workflows smoother. **Quick Adoption:** Intuitive systems encourage quick adoption by educators, allowing them to leverage the benefits without extensive training.	**Integration into Educational Practices:** Ensuring that these systems align seamlessly with existing educational practices and curriculum requirements is essential for user acceptance. **Accessibility and Inclusivity:** User-friendliness should extend to ensuring that the systems are accessible to a diverse range of educators, including those with varying levels of technological proficiency.
Potential impact on educational practices	**Personalized Learning:** Automated assessments can provide real-time feedback, enabling personalized learning experiences tailored to individual student needs. **Data-Driven Decision Making:** The data generated by these systems can inform educators, administrators, and policymakers in making data-driven decisions to enhance teaching strategies and educational policies.	**Fairness and Bias:** Ensuring that automated systems do not perpetuate biases and are fair to students from diverse backgrounds is a critical challenge that needs to be addressed to maintain the integrity of assessments. **Balancing Automation and Human Touch:** Striking the right balance between automated assessments and the need for human involvement in subjective evaluations, especially in essay-type questions, is crucial.

with automated assessments. Current systems excel in grading but lack synergy with platforms that dynamically tailor educational experiences based on assessment outcomes. Furthermore, the complexity of questions, especially those requiring critical thinking, poses a challenge for existing automated systems. Innovative AI approaches, possibly leveraging advanced Natural Language Processing and Deep Learning, should be explored to enhance the systems' ability to evaluate nuanced and complex answers effectively. These research endeavors collectively aim to propel automated assessment systems beyond their current capacities, fostering fairness, transparency, and adaptability in the realm of education.

6. Conclusion

This manuscript offers an extensive survey and evaluation concerning the automated assessment of pen-and-paper tests. The primary focus of the analysis pertains to the utilization of diverse artificial intelligence methodologies within this domain. In the future, the adoption of more advanced and robust algorithms is poised to significantly augment their utilization in the automated assessment process. This not only expedites the assessment process but also ensures consistency and objectivity in grading across a large volume of responses.

For instance, ChatGPT, powered by cutting-edge natural language processing (NLP) algorithms, possesses the capability to comprehend and generate human-like text responses. The system can analyze and evaluate student responses in a manner that closely mirrors human grading standards. By generating tailored feedback messages, the system can provide insightful guidance to students, helping them understand their strengths and areas for improvement.

Moreover, the analysis of existing software systems has shown that no system currently exists that can detect and grade multiple classes of questions. Conversely, analysis of existing paper tests has indicated the need for a tool that unites different classes of questions. Therefore, the challenge for the future is to develop such a system. The authors will attempt to implement such a system in their research.

Acknowledgement

This research was supported by the Science Fund of the Republic of Serbia, grant no. 6526093, AI-AVANTES (www.fondzanauku.gov.rs). The authors gratefully acknowledge the support. The research was partially conducted in the premises of the Palace of Science, Miodrag Kostić Endowment - Centre for Applied Artificial Intelligence.

References

1. Abdul Salam, M., El-Fatah, M. A., Hassan, N. F. Automatic Grading for Arabic Short Answer Questions Using Optimized Deep Learning Model. Plos one, 2022, 17(8), 269-272. https://doi.org/10.1371/journal.pone.0272269
2. Alikaniotis, D., Yannakoudakis, H., Rei, M. Automatic Text Scoring Using Neural Networks. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016, 715-725. https://doi.org/10.18653/v1/P16-1068
3. Alomran, M., Chai, D. Automated Scoring System for Multiple Choice Test with Quick Feedback. International Journal of Information and Education Technology, 2018, 8(8), 538-545. https://doi.org/10.18178/ijiet.2018.8.8.1096
4. Arias Fisteus, J., Pardo, A., Fernández García, N. Grading Multiple Choice Exams with Low-Cost and Portable Computer-Vision Techniques. Journal of Science Education and Technology, 2013, 22(4), 560-571. https://doi.org/10.1007/s10956-012-9414-8
5. Bošnjaković, A., Protic, J., Bojić, D., Tartalja, I. Automating the Knowledge Assessment Workflow for Large Student Groups: A Development Experience. The International journal of engineering education, 2015, 31(4), 1058-1070. ISSN-e 0949-149X
6. Bosnjakovic, A., Protic, J., Tartalja, I. Development of a Software System for Automated Test Assembly and Scoring, ICERI2010 Proceedings, 2010, 6012-6016. ISBN: 978-84-614-2439-9
7. Catalan, J. A. A Framework for Automated Multiple-Choice Exam Scoring with Digital Image and Assorted Processing Using Readily Available Software. DLSU Research Congress 2017, De La Salle University, Manila, Philippines, 2017, 1-5.
8. Chamorro, M. E. G. Cognitive Validity Evidence of Computer- and Paper-Based Writing Tests and Differences in the Impact on EFL Test-Takers in Classroom Assessment. Assessing Writing, 2022, 51(1), 1-21. https://doi.org/10.1016/j.asw.2021.100594
9. Dasgupta, T., Naskar, A., Dey, L., Saha, R. Augmenting Textual Qualitative Features in Deep Convolution Recurrent Neural Network for Automatic Essay Scoring. In Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, 2018, 93-102. https://doi.org/10.18653/v1/W18-3713

10. De Lorenzo, A., Nasso, A., Bono, V., Rabaglietti, E. Introducing TCD-D for Creativity Assessment: A Mobile App for Educational Contexts. International Journal of Modern Education and Computer Science, 2023, 1, 13-27. https://doi.org/10.5815/ijmecs.2023.01.01

11. Dong, F., Zhang, Y. Automatic Features for Essay Scoring - An Empirical Study. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, 1072-1077. https://doi.org/10.18653/v1/D16-1115

12. Đukić, J., Jocović, V., Mišić, M., Tomašević, M. Automated Grading System for PicoComputer Assembly Codes Integrated Within E-Learning Platform. Proceedings, IX International Conference IcEtran, Novi Pazar, Serbia, 2022, 6-9. ISBN 978-86-7466-930-3

13. Jocović, V., Đukić, J., Mišić, M. First Experiences with Moodle and Coderunner Platforms in Programming Course. In Proceedings of the Tenth International Conference on e-Learning, Belgrade Metropolitan University, Belgrade, 2019, 81-86.

14. Jocovic, V., Marinkovic, M., Stojanovic, S., Nikolic, B. Automated Assessment of Pen and Paper Tests Using Computer Vision. Multimedia Tools and Applications, 2023, 1-22. https://doi.org/10.1007/s11042-023-15767-2

15. Jocovic, V., Nikolic, B., Bacanin, N. Software System for Automatic Grading of Paper Tests. Electronics, 2023, 12(19), 1-24. https://doi.org/10.3390/electronics12194080

16. Li, Z., Tomar, Y., Passonneau, R. J. A Semantic Feature-Wise Transformation Relation Network for Automatic Short Answer Grading. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, 6030-6040. https://doi.org/10.18653/v1/2021.emnlp-main.487

17. Lin, Y., Zheng, L., Chen, F., Sun, S., Lin, Z., Chen, P. Design and Implementation of Intelligent Scoring System for Handwritten Short Answer Based on Deep Learning. In 2020 IEEE International Conference on Artificial Intelligence and Information Systems (ICAIIS), 2020, 184-189. https://doi.org/10.1109/ICAIIS49377.2020.9194943

18. Loudon, C., Macias-Muñoz, A. Item Statistics Derived from Three-Option Versions of Multiple-Choice Questions Are Usually as Robust as Four- or Five-Option Versions: Implications for Exam Design, Advances in Physiology Education, 2018, 42(4), 565-575. https://doi.org/10.1152/advan.00186.2016

19. Lu, M., Zhou, W., Ji, R. Automatic Scoring System for Handwritten Examination Papers Based on YOLO Algorithm. In Journal of Physics: Conference Series 2021, 2026(1), 12-30. https://doi.org/10.1088/1742-6596/2026/1/012030

20. Martin, D. M., Kumar, D., Wong, A., Loo, C. K. A Comparison of Computerized Versus Pen-and-Paper Cognitive Tests for Monitoring Electroconvulsive Therapy-Related Cognitive Side Effects. The Journal of ECT, 2020, 36(4), 260-264. https://doi.org/10.1097/YCT.0000000000000687

21. Nardi, A., Ranieri, M. Comparing Paper-Based and Electronic Multiple-Choice Examinations with Personal Devices: Impact on Students' Performance, Self-Efficacy and Satisfaction. Br J Educ Technol, 2019, 50(3), 1495-1506. https://doi.org/10.1111/bjet.12644

22. Nguyen, H. T., Nguyen, C. T., Oka, H., Ishioka, T., Nakagawa, M. Fully Automatic Scoring of Handwritten Descriptive Answers in Japanese Language Tests. In IEICE technical report PRMU2021-32, 2022, 45-50. https://doi.org/10.48550/arXiv.2201.03215

23. Okewu, E., Adewole, P., Misra, S., Maskeliunas, R., Damasevicius, R. Artificial Neural Networks for Educational Data Mining in Higher Education: A Systematic Literature Review. Applied Artificial Intelligence, 2021, 35(13), 983-1021. https://doi.org/10.1080/08839514.2021.1922847

24. Patole, S., Pawar, A., Patel, A., Panchal, A., Joshi, R. Automatic System for Grading Multiple Choice Questions and Feedback Analysis. IEEE International Journal of Technical Research and Applications, 2016, 12(39), 16-19. ISSN: 2320-8163

25. Perry, K., Meissel, K., Hill, M. F. Rebooting Assessment: Exploring the Challenges and Benefits of Shifting from Pen-and-Paper to Computer in Summative Assessment. Educational Research Review, 2022, 36, 100451. https://doi.org/10.1016/j.edurev.2022.100451

26. Petrova-Antonova, D., Spasov, I., Petkova, Y., Manova, I., Ilieva, S. Cognisoft: A Platform for the Automation of Cognitive Assessment and Rehabilitation of Multiple Sclerosis. Computers, 2020, 9(4), 1-13. https://doi.org/10.1007/s11042-019-7423-9

27. Rowtula, V., Oota, S. R., Jawahar, C. V. Towards Automated Evaluation of Handwritten Assessments. In 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, 426-433. https://doi.org/10.1109/ICDAR.2019.00075

28. Santosh, K. C., Antani, S. K. Recent Trends in Image Processing and Pattern Recognition. Multimedia and Tools Application, 2020, 79(47), 34697-34699. https://doi.org/10.1007/978-981-16-0507-9

29. Shaikh, E., Mohiuddin, I., Manzoor, A., Latif, G., Mohammad, N. Automated Grading for Handwritten Answer Sheets Using Convolutional Neural Networks. In 2019 2nd International Conference on New Trends in Computing Sciences (ICTCS), 2019, 1-6. https://doi.org/10.1109/ICTCS.2019.8923092

30. Shehab, A., Faroun, M., Rashad, M. An Automatic Arabic Essay Grading System Based on Text Similarity Algorithms. International Journal of Advanced Computer Science and Applications 2018, 9(3), 263-268. https://doi.org/10.14569/IJACSA.2018.090337

31. Stanisavljevic, Z., Nikolic, B., Tartalja, I., Milutinovic, V. A Classification of eLearning Tools Based on the Applied Multimedia. Multimedia Tools and Applications, 2015, 74(1), 3843-3880. https://doi.org/10.1007/s11042-013-1802-4

32. Süzen, N., Gorban, A. N., Levesley, J., Mirkes, E. M. Automatic Short Answer Grading and Feedback Using Text Mining Methods. Procedia Computer Science, 2020, 169(1), 726-743. https://doi.org/10.1016/j.procs.2020.02.171

33. Taghipour, K., Ng, H. T. A Neural Approach to Automated Essay Scoring. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, 1882-1891. https://doi.org/10.18653/v1/D16-1193

34. Tractenberg, R., Gushta, M., Mulroney, S., Weissinger, P. Multiple Choice Questions Can Be Designed or Revised to Challenge Learners' Critical Thinking. Advances in Health Sciences Education, 2013, 18(1), 945-961. https://doi.org/10.1007/s10459-012-9434-4

35. Tulu, C. N., Ozkaya, O., Orhan, U. Automatic Short Answer Grading with SemSpace Sense Vectors and MaLSTM. IEEE Access, 2021, 9(1), 19270-19280. https://doi.org/10.1109/ACCESS.2021.3054346

36. Xiao, S., Li, T., Wang, J. Optimization Methods of Video Images Processing for Mobile Object Recognition. Multimedia Tools and Applications, 2020, 79(25), 17245-17255. https://doi.org/10.1007/s11042-019-7423-9

This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).