Hybrid ANN-Based and Text Similarity Method for Automatic Short Answer Grading in Polish

Marwah Bani Saad , Lidia Jackowska-Strumillo * , Wojciech Bieniecki

Posted Date: 9 December 2024

doi: 10.20944/preprints202412.0750.v1

Keywords: open question test; automatic test assessment; natural language processing; split algorithm; similarity measures; machine learning; artificial neural network; hybrid text-processing method

Article

Hybrid ANN-Based and Text Similarity Method for Automatic Short Answer Grading in Polish

Marwah Bani Saad, Lidia Jackowska-Strumillo * and Wojciech Bieniecki

Institute of Applied Computer Science, Lodz University of Technology, Stefanowskiego 18, 90-537 Lodz, Poland

* Correspondence: lidia.jackowska-strumillo@p.lodz.pl

Abstract: Computer-assisted grading plays an important role in an educational context, mainly by reducing the workload of teachers in manual scoring. While electronic choice surveys have long been used in many web applications, automatic scoring of open-ended responses remains an interesting research problem in natural language processing. In this article, we propose a new hybrid text-processing method for scoring students' responses based on word splitting and preprocessing, which will then combine textual algorithms with a set of artificial neural network classifiers and a set of heuristic decision rules. This concept has been implemented in the interactive e-test system operating in the local computer network of the Institute of Applied Computer Science at the Lodz University of Technology (TUL). The dataset is acquired as questions, reference answers and students' answers generated on the basis of exams conducted at our institute in the years 2015-2022 for more than a thousand students. This article extends our previous research [1–3] and includes comparative tests. The proposed method achieves excellent results and outperforms the previous approaches. The obtained precision is equal to 1, and the recall measure is 0.97 for the final results. The algorithm is designed for the Polish language but can be adapted to other inflected languages.

Keywords: open question test; automatic test assessment; natural language processing; split algorithm; similarity measures; machine learning; artificial neural network; hybrid text-processing method

1. Introduction

Evaluation of student knowledge for a large group of students is a laborious process. However, this plays a significant role in making the teaching process efficient. Invaluable support for the teacher is the use of **c**omputer-aided assessment for evaluating student work [4,5]. Today's forms of education include quite frequent electronic tests [1] or interactive surveys [6,7]. Electronic tests should provide the ability to obtain results quickly. Tests prearranged as closed-question sets (true/false, single, or multiple choice, match) are scored immediately by coding them appropriately in the test system [8].

On the other hand, open-ended questions require more knowledge and understanding of the subject from the students to write their free answers [9,10].

Evaluation for the open-ended questions takes into consideration grammar, style, and coherence [11–13]. These question types are effective in developing the level of student knowledge and skills. Due to the expected length of the answers, we can call these types of questions "essays" or "short answer questions".

Although the automatic grading of essays is the subject of scientific research, the reliability of the results obtained by the automated method is still questionable, especially in the case of inflectional languages, such as the Polish language [3].

In our e-test application, one short answer question object consists of the question phrase, one to four phrases proposed by the teacher as correct reference answers, and the answer given by the student. In our previous work, we presented a couple of text algorithms for short-answer scoring. The first algorithm first tries to match the student's answer exactly to one of the reference answers

 doi:10.20944/preprints202412.0750.v1

and then uses the approximate match method [1]. In its revised version, scoring is proportional to the degree of fit [3]. Another alternative scoring method is based on Artificial Neural Networks (ANN) [2].

The new hybrid scoring method presented in this article combines the text similarity algorithms with ANN-based evaluation and thus solves most of the shortcomings of previously developed algorithms.

The proposed algorithm combines the following previously used procedures:

- preprocessing (simplification, lemmatisation),
- using the Jaccard index to measure similarity,
- MultiLayer Perceptron (MLP) classifier.

In the newly developed algorithm, the student's answer is preprocessed and then evaluated in parallel by the text similarity algorithm and revised ANN-based algorithm. The final grading is determined in the last step based on these two scoring results and a set of heuristic decision rules.

The rest of the paper is structured as follows: Section 2 is state-of-art; Section 3 provides an overview of the most recent studies by the authors; Section 4 describes the new Hybrid ANN model joining preprocessed split method and developed ANN-based methods; Section 5 shows the results of newly developed methods compared with the Authors' previous experiments; Section 6 contains discussion and Section 7 conclusions.

2. Related Work

Automatic evaluation of short answers can be treated as a measurement of Short Text Similarity (STS). The similarity can be measured literally or meaningfully [7,14]. In [15], the following classes of STS methods were distinguished: string-based similarity, traditional models, vector space models and the most promising - neural models.

Various approaches for the automatic evaluation of short-answer tasks were proposed. One of them was the Grader Assistance System (GAS) [16]. It used DKPro Core (ready-to-use software for natural language processing) and DKPro Similarity libraries (framework for developing text Similarity algorithms), and it computed lemmas using the Stanford lemmatiser component in DKPro Core and excluded stop words using Porter's German stop word list. Furthermore, a Vector-Based System was developed [17]. This system used different sentence representations like word-vector representation and similarity measures like string similarity, knowledge-based similarity, and corpus-based similarity.

Also, the Automatic Short Answer Grading (ASAG) system was presented to explore the effectiveness of Machine Learning (ML) approaches such as Naive Bayes (NB) and Decision Trees (DT) by extending the student answer to leveraging Deep Belief Networks (DBN) with several extended features [18]. Moreover, a system was constructed to improve that it can combine a response-based method that extracts features from student answers with a reference-based method that matches the student answers with target answers, and it outperformed a non-stacked combination and is helpful for most Natural Language Processing (NLP) cases [19]. Due to the various NLP issues and tasks, several researchers put a lot of effort into investigating new approaches and applications to solve NLP problems [20,21]. Some of them considered an artificial neural network as a significant solution, such as a system that combined Siamese bi-Long Short-Term Memory (bi-LSTMs), a novel pooling layer based on the Sinkhorn distance between LSTM state sequences, and a support vector ordinal output layer and the system had scoring accuracy superior to recent baselines [22]. Another model was proposed for Automatic Text Scoring (ATS), which focused on extended word representation based on LSTM [23]. One more automatic scoring system is based on word embedding and paragraph embedding [24]. Existing attention networks use word-based or sentence-based attention, as in [25].

In [26], a novel and effective method was introduced, an attention-based deep learning method using lexicons and Word2vec model for word representation for sentiment analysis of social media text data. It combined the current state-of-the-art machine learning and NLP representation learning methods.

 doi:10.20944/preprints202412.0750.v1

As relational meaning is central to NLP, [27] presented a new type of Relation Network (RN), a Semantic Feature-wise transformation Relation Network (SFRN), that learned relational information from QRA triples: a question (Q), reference answer (R), and student answer (A) for automatic short answer grading, learning from two types of training data, using reference answers or rubrics. SFRN+ is the version with the Bidirectional Encoder Representations from Transformers (BERT) of Google company.

In [28], the relationship between natural language processing and learning English as a Foreign Language (EFL) was explored. Neural language processing in [29] showed the possibility of reducing the scoring burden and enabling more extensive studies. It showed how to develop an approach to automatically score responses using an existing language model (distilBERT) that successfully identifies the amount of internal and external content in each sentence.

In [30], the authors reviewed the automatic assessment of text-based responses in higher education. Their review focused on the period between 2017 and 2023 to capture work conducted in the years when NLP advances such as BERT and Large Language Models (LLM) became available.

Also, [31] reviewed the current understanding of online assessment, identifying major online assessment approaches and the different functions online technologies serve. While tests have been the dominant approach, in which technologies substituted existing assessments, online assignments and skills assessments involved more innovative assessment practices.

In [32], a Hierarchical Rater Model based on Signal Detection Theory (HRM-SDT) showed an optimal performance by considering it a viable solution for Automated Essay Scoring (AES) score integration regarding automated content scoring on multiple scoring items.

Traditionally, grading requires educators to design a comprehensive rubric and meticulously review each student's answers [33]. A promising application in higher education involves leveraging LLMs to enhance the grading process for students.

Instead, using Artificial Intelligence (AI) has been around for a while but was oftentimes hampered by the effort and cumbersomeness of training and validation. Even for short answers, AI systems (including LLMs like BERT) needed to be specifically trained for each class of problems [34], with very few exceptions.

Recently, the idea was focused on developing a technique to validate LLMs' output in the context of program synthesis to generate predicates for testing. In [35], an approach is to use back-translation models that generate buggy programs from explanations to validate the generated content and show the use of human-in-the-loop for validating low-confidence outputs.

Additionally, [36] used trained models on only essays of the highest quartile of students in terms of performance, proving that these models are not suitable for students from the other quartiles. Furthermore, the research emphasized highlighted, that the fairness of AES systems is compromised if such models are used on students or tasks for which they have not been trained.

A clustering model can help teachers analyse students' assignments and is also an effective method to support essay grading. For instance, automatically cluster programming assignments (C programming), *k*-means was applied in the study [37] proposed a clustering method to cluster sentences to support grading the essays written in Finnish by bachelor students.

A step toward Generative Artificial Intelligence (GAI) such as ChatGPT in [38] that showed several jobs of text summarisation, machine translation, problem-solving, sentiment analysis, question-answering, and creative writing that can be easily adjusted without training by engaging in context-aware, human-like conversations and producing clear, educational, and pertinent responses.

In [39], a new method was proposed using item response theory that scores different AES models and considers the differences in the characteristics of scoring behaviour among models.

The [40] presented a novel architecture for the problem of language-independent sentiment analysis of text classification. It was focused on English and German languages. The proposed model outperforms the baseline by a significant margin with respect to F1-score on English and mixed-language datasets.

Some researchers consider text-based sentiment analysis an essential field in NLP to be worked on [41,42]. A hybrid model has been developed by joining the lexicon-based approach with machine

 doi:10.20944/preprints202412.0750.v1

learning [43]. Moreover, another hybrid model has been proposed and focused on the issue of sharing knowledge in the Question Answer Community (QAC) based on the content and non-content models predicting the best answer to the user and reducing time-consuming during the asking process [44]. Hybrid models in various applications significantly improved results compared with traditional approaches [43–46].

This article presents a new hybrid method combining textual similarity algorithms with a set of MLP classifiers and heuristic decision rules for short answer evaluation.

3. Methods

3.1. Exact and Approximate Text Matching Scoring Algorithm [1]

The exploited case assumes that the teacher provides a question followed by one to 4 correct but different reference answers, which are not shown to the student. The student's answer is compared with reference answers, and a score is given on a continuous scale of 0 to 5. Grading was based on exact and approximate matching between student and teacher reference answers. Each answer was treated as a set of words. Numbers and acronyms appearing in the responses were treated in a special way. A dictionary of the Polish language with its declensions [47] was used.

The algorithm consisted of three phases:

1. Preprocess the students' answers: remove all punctuation, convert all to lowercase, and split them into a sequence of words.
2. Apply lemmatisation to words from step 1 to convert to the basic form.
3. Search in the dictionary for acronyms (words in all upper case) to convert them to their lemma.
   - If the word is not found, the Levenshtein distance will be applied to find its nearest word.
   - If a number exists inside the student's answer, the exact matching method will be applied.

The whole algorithm with exact matching, named the Split algorithm, is displayed in Listing 1.

```
score  ← 0
acc_ans ← list of model answers
st_ans ← Student answer
f_ans ← set(split(filter(st_ans)))
FOR EACH ans IN acc_ans
        points ← 0
        t_ans ← set(split(ans))
        common_words ← t_ans ∩ f_ans
        points ← size(common_words) / size(t_ans)
        IF points > score
                score ← points
        END IF
END FOR
```

Listing 1. Pseudocode of scoring algorithm with an exact match (Split algorithm).

This method is compared to the new automatic scoring method in Section 5.

3.2. Revised Split Algorithm [3]

This section shows some improvements to the automatic scoring method. Initially, for the comparison of word sets Jaccard index was used.

Jaccard index is a parameter that calculates the similarity of two sets [48]. In our implementation, the value is multiplied by five to convert it into a mark name.

 doi:10.20944/preprints202412.0750.v1

$$J = \frac{|X \cap Y|}{|X \cup Y|} * 5.0 \quad (1)$$

In (1), X is the set of words from the model answer, and Y is the set of words from the examined answer.

From the method in section (3.1), the Jaccard was applied to each teacher's answers. Experiments presented in [1] show that the results for scoring the student answers were reasonable only when the number of words in student answers was less or equal to the length of teacher answers.

Table 1. Example shows the differences in the length for both teacher and student answers.

Question 1: Do zapisu znaków alfanumerycznych w komputerach PC stosuje się				
Teacher answers	**Student #1 answer**	**J**	**Student #2 answer**	**J**
kod ASCII	tablice kodow ASCII	3.33	ASCII	2.5
ASCII		1.66		5
znaki ASCII		1.66		2.5
ASCI		0		0
Question 2: Dyski twarde zaliczamy do pamięci				
Teacher answers	**Student #1 answer**	**J**	**Student #2 answer**	**J**
magnetycznych	elektryczny magnetyczny	2.5	trwałych	0
trwałych		0		5
zewnętrznych		0		0
masowych		0		0
Question 3: Pojęcie "hardware" określa:				
Teacher answers	**Student #1 answer**	**J**	**Student #2 answer**	**J**
wszystkie elementy materialne komputera	Sprzęt który składa się na komputer podzespoły	0.5	sprzęt	0
elementy materialne komputera	komputera	0.55		0
sprzęt		0.71		5
sprzęt komputera		1.42		2.5
Question 4: Budowa i zasada działania płyty DVD jest najbardziej zbliżona do budowy				
Teacher answers	**Student #1 answer**	**J**	**Student #2 answer**	**J**
Płyty CD	płyty CD ROM	3.33	CD	2.5
CD		1.66		5
Dysku CD		1.25		2.5
Krążka CD		1.25		2.5

In Table 1, four questions were given to two students. Both students answered each question correctly. Student #1 gave longer answers than each teacher's answer. Student #2 gave answers shorter or equal to each teacher's answers. We can see that Student #1 received scores of 3.33, 2.5, 1.42 and 3.33, while Student #2 got 5, 5, 5, 5.

For that reason, the new concept for the Automatic Scoring Algorithm has been considered. It focused on correcting the deficient results from the previous algorithm and applying the following steps:

- Length of Data Sets

Due to the fact that the length for both teacher's and student's answer sets is important, the new factors were applied in this method:

- If the teacher's answer is longer than the student's answer, the Jaccard measure will be calculated as in section 3.1
- Otherwise, we expect the teacher's answer to be included in the student's answer. We use the split algorithm and then count the matching words to calculate the score.
  - If all the words from the teacher's answer are found inside the student's answer, the score will be five.
  - If some of the teacher's answer words are found inside the student's answer, the percentage of found words will be calculated for the whole teacher's answer words. The score is five for 100% of the found words, and the score is 0 for less than 70% of the found words.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 9 December 2024 doi:10.20944/preprints202412.0750.v1

- Levenshtein Distance

After obtaining the result, we analysed the zero scores and checked for student's spelling mistakes. To find them, we calculated the Levenshtein Distance to the student's answer word by word.

- Calculate the Score

Finally, we chose the maximal value from both above sections (Listing 2).

```
Score ← 0
threshold ← 80%
st_ans ← GetEditArea( )
s ← Filter (st_ans)
similarities ← Array( )
FOR    i← 0 TO Length(acc_ans)
a ← Filter(acc_ans[i])
similarities[i] = SimilarityMeasure(a, s)
END FOR
k ← IndexOfMax(similarities)
IF similarity[k] > threshold THEN score ← 5
```

Listing 2. Pseudocode of new automatic scoring algorithm with Levenshtein distance (New split algorithm).

3.3. ANN-Based Method [2]

The next step toward improving our algorithm was to use a small neural network to generate a score for a given short answer. For each question, an individual MLP structure was created. It required data preparation for the training process as in the following steps:

1. The input set has been filtered from all the punctuation and special characters, separated the numbers, split the sentence into words and removed all extra spaces.
2. Using the dictionary [47], assign an integer value to each word. If the word is in a base form or the same integer number with a float extension from 0.1 to 0.5, the word is in a different form. If there is no exact match for a word in the dictionary, the search is repeated using the Levenshtein distance, assuming that this distance cannot exceed 1, as shown in Listing 3.
3. After step 2, we obtain a vector of numbers, but its length varies. The structure of the input layer of the network requires that the dimension of the vector be N. If the sequence of numbers is longer, we choose the first N numbers; if it is shorter - we fill it with zeros.
4. In this step, the sequence of symbols from step 3 can be very long, and it would slow down the ANN training. Therefore, rescaling the input is necessary, and one of the best methods for rescaling is Min-Max Normalization. So, the input set will be scaled in the range of 0 and 1. The following equation is the Formula of Min-Max:

$$V' = \frac{V - Min_A}{Max_A - Min_A} \left(New_{Max_A} - New_{Min_A}\right) + New_{Min_A} \quad (2)$$

5. The training set was divided into two parts: 80% for learning and 20% for testing. The evaluation of the ANN performance was resolute by applying two accuracy measures: Sum Square Error (SSE) and Mean Square error (MSE). SSE was applied during the learning process, and MSE was used to choose the best ANN structure for the testing data set.

 doi:10.20944/preprints202412.0750.v1

MSE Equation:

$$MSE = \frac{1}{n}\sum_{k=0}^{n}(Y - X)^2 \quad (3)$$

where n is the number of symbols in the input data set, (Y-X) is the difference between the actual output and the desired output from the ANN training process. This ANN-based method will be compared with the new ANN method in Section 5.

```
function Read_Dict;
Input:
f - text file with words
Output:
DI - associative array (key: string, value: float)
N - number of lines (and integer index of last word in a basic form)
- - - - - - - - - - - - - - - - -
lines ← split_to_lines(f);
N ← length(lines)
for j from 0 to N-1 do
        words ← split_to_words(lines[j]);
        M ← length(words);
        nums ← [j, j + 0.5/M,..., j + 0.5]
        for k from 0 to M-1 do
                DI[words[k]] ← nums[k];
        end for;
end for;
end.
```

Listing 3. Reading dictionary for phrases.

3.4. ANN-Based Method with Lemmatisation

In this section, we focus on presenting the major differences and developing and adding new concepts for the previous ANN-based method.

New modifications for the primary ANN-based method are proposed to improve the result.

Initially, we apply the same step 1 from Section 3.3 as filtering and preprocessing the input data set before training.

The first modification is applying lemmatisation to convert all words to their basic forms. As a result of lemmatisation, an integer index is assigned to all words using the dictionary. It increases the speed of the training phase. After the new concepts for lemmatisation and converting the input datasets to integer symbols, same as in Section 3.3 steps 4 and 5, the normalisation method will be applied to rescale the numbers to avoid the slowness of the training process, using the same Formula of Min-Max equation (2) and shrink the input data set to N symbols.

The evaluation process in Section 3.3 Step 5 shows that the training process does not depend on a specific number of the hidden layer neurons and is changeable until the optimal error rate is reached.

 doi:10.20944/preprints202412.0750.v1

Moreover, in this section, we use a fixed number of neurons and train the network for 15000 epochs, and the process ends when the MSE reaches the optimal error rate of 0.0001. Similarly, the learning data set is divided into two parts: 80% for the training set and 20% for the testing set.

3.5. Additional Reference Answers

After analysing the errors from previous methods, we noticed that sometimes a correct student answer is not included in the four teacher's acceptable answers. In this work, we decided to increase the number of teacher answers to five or more for some of the 25 questions—an example of adding the fifth teacher-acceptable answer is shown in Table 2.

Table 2. Example of the addition for the fifth Teacher Acceptable Answer.

Question	Teacher 1	Teacher 2	Teacher 3	Teacher 4	New teacher
Dyski twarde zaliczamy do pamięci	magnetycznych	trwałych	zewnętrznych	masowych	nieulotnych
Budowa i zasada działania płyty DVD jest najbardziej zbliżona do budowy:	płyty CD	CD	dysku CD	krążka CD	płyty która ma pity i landy

4. Hybrid ANN-Based Method

This section presents the proposed hybrid approach consisting of two parallel modules: the developed revised similarity split algorithm described in Section 3.2, called the *New Split Algorithm*, and the *New ANN Algorithm* with lemmatisation shown in Section 3.4.

Figure 1 depicts the structure of the Hybrid ANN-based method. STUD stands for the student's answer, TECH is the teacher's answer, DICT is the dictionary, Lev is Levenshtein distance, and J is the Jaccard measure.

In this method, the input dataset is prepared by preprocessing and lemmatisation, and then the new ANN-based method and the new Split algorithm are applied in parallel.

For the new Split algorithm, all the steps from Section 3.2 are used to calculate the evaluation score S1 at the end. Moreover, the new ANN-based method is applied with all its modifications from Section 3.4 to calculate the evaluation score S2 at the end.

The final grading is determined in the last step based on S1 and S2 scoring results and a set of decision rules.

The heuristic decision rules were designed based on the analysis of the results obtained for the evaluation algorithms presented in Tables 3 and 4. It was observed that for the New Split Algorithm, the True Negative cases were correctly recognised in 100%, and the True Positive cases were recognised the best by the New ANN Algorithm, i.e. in 86.26%. Therefore, if the S1 score given by the New Split Algorithm equals zero, the Hybrid ANN-based method result S equals zero also. If the S2 score given by the New ANN Algorithm equals five, the Hybrid ANN-based method result S equals five also. In the other cases, the hybrid method score S is calculated as the maximum for both scores: S = Max (S1, S2).

 doi:10.20944/preprints202412.0750.v1

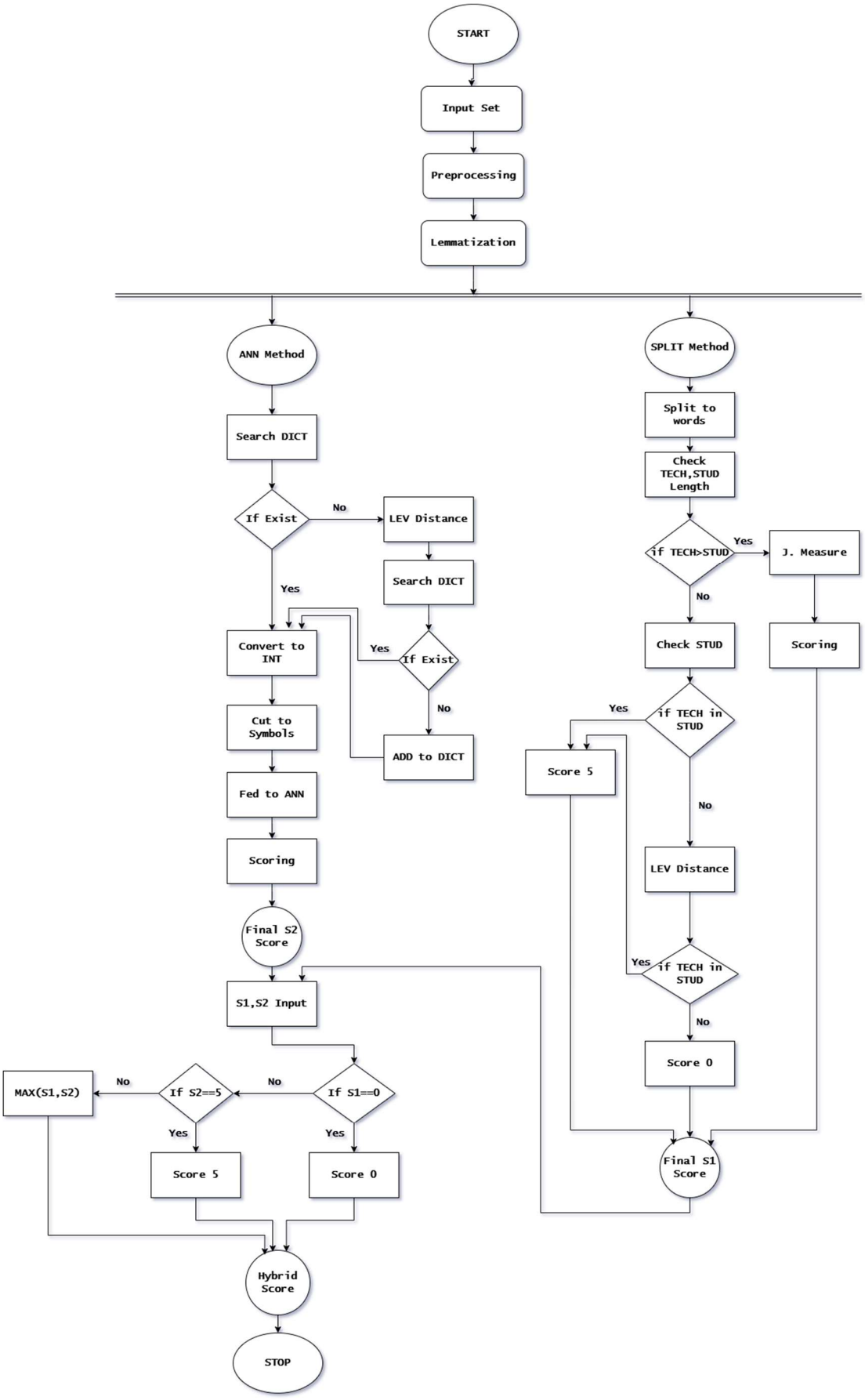


Figure 1. The Hybrid Method Structure.

 doi:10.20944/preprints202412.0750.v1

5. Results

The proposed Hybrid method has been tested on a large number of input data sets from real student exams in the Institute of Applied Computer Science in Lodz University of Technology for a period of 5 years selected from 2015-2022, excluding the COVID-19 period.

The number of student answers from these exams was 3054, gathered for 25 different questions. The data set for each question consisted of more than 125 student answers with four or more teacher answers. All the students' answers were evaluated by the teacher (Expert). Of the 3054 student answers, there are 1288 correct answers and 1766 incorrect answers.

Firstly, the new concept of the Split algorithm with the length check of both teacher and student answers and the calculation of the similarity measures described in Section 3.2 has been examined. A comparison of the previous and newly developed split algorithms is shown in Table 3. The new modifications of the Split algorithm improved the recognition of the correct answer from 58.16% to 74.3% and perfectly recognised the whole 1766 incorrect answers in 100%.

Secondly, the new ANN-based method was examined. The assessment task was divided into 25 subtasks, and 25 small networks were built, with one ANN for each question type to facilitate a fast and flexible training process. This solution was considered earlier in other applications [46]. The ANN structure was in the form of (5-3-1): 5 introduced the number of inputs, 3 introduced the number of hidden layer neurons, and 1 introduced the output if it is true or false.

As mentioned before, the whole training set was divided into two separate sets, 80% for learning and 20% for testing. Therefore, 90-100 ANN networks represented each of the 25 questions. We trained the networks for a maximum of 15000 epochs. The training process was stopped for some number of epochs according to the minimum MSE error rate. We checked the MSE in each epoch step with an average of 0.0001 or less to stop the training process. The best ANN for each case was chosen from the ANN set considering the smallest MSE for the testing data.

In the training set, the new ANN method recognised 910 from 1078 correct answers (84.5%), and it recognised 1293 from 1488 incorrect answers (86.8%). In the testing set, the ANN recognised 201 from 210 correct answers (95.7%) and 144 from 278 incorrect answers (51.7%). Results for the whole data set are shown in Table 4.

Results for the New Split Algorithm, New ANN Algorithm and Hybrid Method are gathered in Table 5. All algorithms were compared and evaluated for the same input data set.

Table 3. Comparison of the split algorithms.

Type	Expert	Split Algorithm		New Split Algorithm	
		As Correct	***As Incorrect***	***As Correct***	***As Incorrect***
Correct	**1288**	*True Positive* 749 (58.16%)	*False Negative* 539 (41.84%)	*True Positive* 958 (74.3%)	*False Negative* 330 (25.6%)
Incorrect	**1766**	*False Positive* 1(0.1%)	*True Negative* 1765 (99.9%)	*False Positive* 0 (0%)	*True Negative* 1766 (100%)

Table 4. Comparison of the ANN-based methods.

Type	Expert	ANN Algorithm		New ANN Algorithm	
		As Correct	***As Incorrect***	***As Correct***	***As Incorrect***
Correct	**1288**	*True Positive* 550 (42.71%)	*False Negative* 738 (57.29%)	*True Positive* 1111 (86.26%)	*False Negative* 177 (13.74%)
Incorrect	**1766**	*False Positive* 293 (16.59%)	*True Negative* 1473 (83.41%)	*False Positive* 329 (18.6%)	*True Negative* 1437 (81.3%)

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 9 December 2024 doi:10.20944/preprints202412.0750.v1

Table 5. Comparison of New Split, New ANN-based and Hybrid Methods .

Type	Expert	New Split Algorithm		New ANN Algorithm		Hybrid Method	
		As Correct	*As Incorrect*	*As Correct*	*As Incorrect*	*As Correct*	*As Incorrect*
Correct	**1288**	*True Positive* 958 (74.3%)	*False Negative* 330 (25.6%)	*True Positive* 1111 (86.26%)	*False Negative* 177 (13.74%)	*True Positive* 1250 (97%)	*False Negative* 38 (3%)
Incorrect	**1766**	*False Positive* 0 (0%)	*True Negative* 1766 (100%)	*False Positive* 329 (18.6%)	*True Negative* 1437 (81.3%)	*False Positive* 0 (0%)	*True Negative* 1766 (100%)

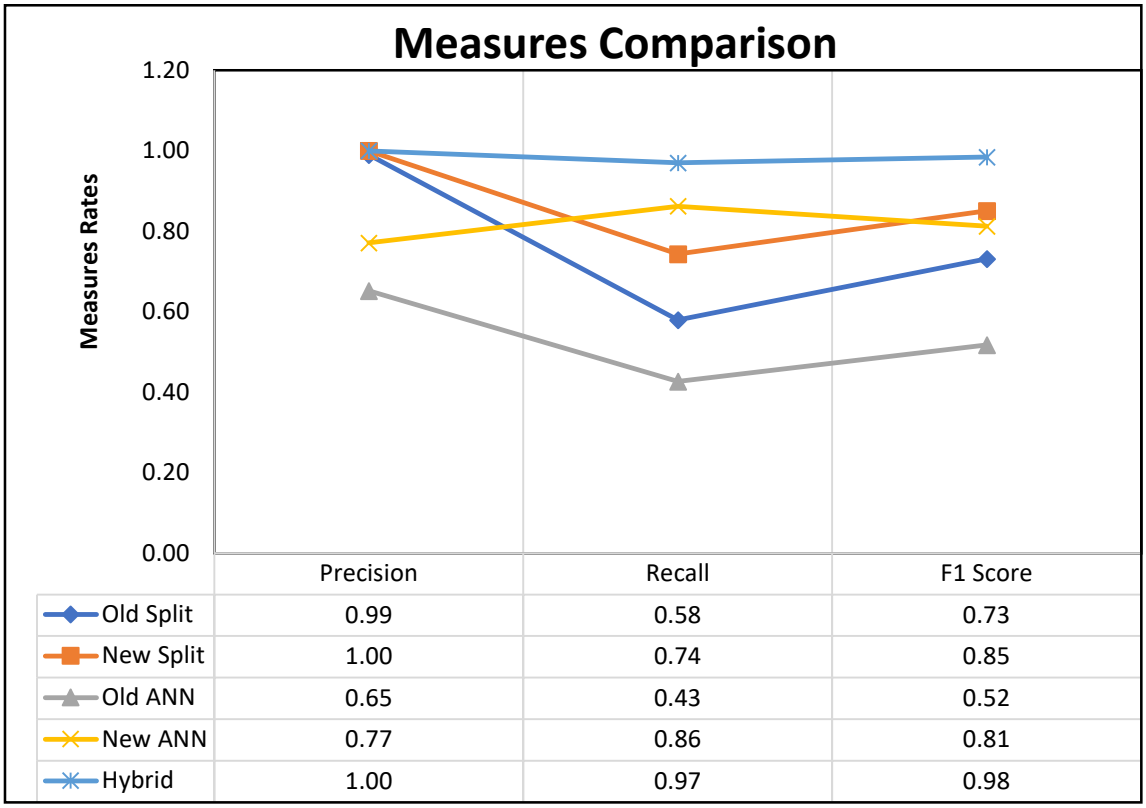
For better results evaluation, the following measures were applied for all considered algorithms:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positive} \quad (4)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negative} \quad (5)$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (6)$$

As the F1 score is a measure used to calculate the balance between precision and recall, it is essential to use this measure when we have many true negatives. We didn't use the accuracy measure because it is only useful when we have a small number of false negatives and false positives. In our case, the F1 score is more appropriate.



	Precision	Recall	F1 Score
Old Split	0.99	0.58	0.73
New Split	1.00	0.74	0.85
Old ANN	0.65	0.43	0.52
New ANN	0.77	0.86	0.81
Hybrid	1.00	0.97	0.98

Figure 2. Measures (Precision, Recall and F1) Comparison.

6. Discussion

The precision for the old split method was very good (0,99), but the new split method reached the optimal perfect rate (1). Also, the new ANN method has a precision measure rate (0,77), and it is much better than the old ANN method, which has a precision rate (0,65). Furthermore, both the new Split and new ANN-based methods have a Recall measure rate (0,74 – 0,86), which is also better than the old Split and ANN recall measure rate (0,58 – 0,42).

 doi:10.20944/preprints202412.0750.v1

F1 measure is optimal when there are differences and a gap between false negatives and false positives. In our case, the F1 measure was very good for both new ANN and Split (0.81 – 0.85) compared with old ANN and Split, which have F1 measure rates (0.52 – 0.73).

All measures for the Hybrid method are the best compared with New ANN and New Split algorithms. The hybrid method reached the optimal results for the Recall measure (0.97) and perfect results for the precision measure (1). For that result, the F1 measure becomes the highest for the Hybrid method (0.98)

Moreover, the addition of an extra teacher answer for some of the questions plays a significant role in this achievement of good results.

7. Conclusions

In this study, we have introduced the new hybrid text processing method for scoring students' answers, which is based on splitting text into words, preprocessing and then combining two scoring algorithms applied in parallel: textual similarity algorithm and ANN-based algorithm. The final grading is based on information fusion of both scoring results and a set of heuristic decision rules.

The developed method is the result of continuous improvement and extension of our work on automatic scoring of open-ended questions for more than ten years. In this article, we discussed and analysed the results of the authors' newly developed and previously developed algorithms. We compared the results of all these methods using the same dataset from real student exams conducted in the years 2015-2022 at the Institute of Applied Computer Science at the Lodz University of Technology for more than a thousand students.

The analysis of the results shown in Figure 2 clearly shows that the proposed hybrid method achieves the best results among all the considered methods and their main components: the new Split and ANN-based algorithms achieve better results than their predecessors. The presented hybrid ANN method outperforms the previous approaches in the quality measures, such as: precision equal to 1, recall 0.97 and F1 score 0.98 for the final results.

Although the results obtained are excellent, we plan to apply the deep learning method in the future to see if it can help to achieve better results. The main difficulty in realising this further development will be the limited amount of data for training deep networks.

The algorithm is intended for the Polish, but can be adapted to other inflected languages.

Author Contributions: Conceptualisation, L.J.S., M.B.S. and W.B.; methodology, M.B.S., L.J.S. and W.B.; software, M.B.S. and W.B.; validation, M.B.S., L.J.S. and W.B.; formal analysis, M.B.S., L.J.S. and W.B.; investigation, M.B.S., L.J.S. and W.B.; resources, L.J.S.; data curation, L.J.S. and M.B.S.; writing—original draft preparation, M.B.S.; writing—review and editing, L.J.S. and W.B.; visualisation, M.B.S.; supervision, L.J.S.; funding acquisition, L.J.S.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was financed from the statutory funds of the Institute of Applied Computer Science, Lodz University of Technology, Poland, No. 501/2-24-1-2.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Abbreviations	Definition
TUL	Lodz University of Technology
ANN	Artificial Neural Network
MLP	MultiLayer Perceptron
STS	Short Text Similarity
GAS	Grader Assistance System
ASAG	Automatic Short Answer Grading
ML	Machine Learning
NB	Naive Bayes
DT	Decision Trees
DBN	Deep Belief Networks

 doi:10.20944/preprints202412.0750.v1

NLP	Natural Language Processing
LSTM	Long Short-Term Memory
ATS	Automatic Text Scoring (ATS)
RN	Relation Network
SFRN	Semantic Feature-wise transformation Relation Network
QRA	Question (Q), Reference answer (R) and student Answer (A)
BERT	Bidirectional Encoder Representations from Transformers
EFL	English as a Foreign Language
LLM	Large Language Models
HRM-SDT	Hierarchical Rater Model based on Signal Detection Theory
AES	Automated Essay Scoring
AI	Artificial Intelligence
GAI	Generative Artificial Intelligence
QAC	Question Answer Community
J	Jaccard index
SSE	Sum Square Error
MSE	Mean Square error

References

1. Jackowska-Strumiłło, L.; Bieniecki W.; and Saad M.B. A web system for assessment of students' knowledge. In Proceedings of the 8th International Conference on Human System Interaction (HSI 2015), Warszawa, Poland, 25-27 June 2015, pp. 20-26, https://doi.org/10.1109/HSI.2015.7170638.
2. Saad, M.B.; Jackowska-Strumiłło L.; and Bieniecki W. ANN Based Evaluation of Student's Answers in E-tests. In Proceedings of the 11th International Conference on Human System Interaction (HSI 2018), Gdansk, Poland, 4-6 July 2018, pp. 155-161, https://doi.org/10.1109/HSI.2018.8431340.
3. Saad M.B.; Jackowska-Strumiłło L.; Bieniecki W. Algorithms for Automatic Open Questions Scoring in E-Learning Systems. In Proceedings of the International Interdisciplinary PhD Workshop 2017 (IIPhDW 2017), Lodz, Poland, 9- 11 Sep. 2017, pp. 176-182.
4. Duch, P.; Jaworski, T. Dante - Automated Assessments Tool for Students' Programming Assignments. In Proceedings of the 11th International Conference on Human System Interaction (HIS 2018), Gdansk, Poland, 4-6 July 2018, pp. 162-168.
5. Stoliński, S.; Bieniecki W.; and Stasiak-Bieniecka M. Computer aided assessment of linear and quadratic function graphs using least-squares fitting. In Proceedings of the 2014 Federated Conference on Computer Science and Information Systems (FedCSIS 2014), pp. 651-658, https://doi.org/10.15439/2014F365.
6. Jackowska-Strumiłło, L.; Nowakowski J.; Strumiłło P.; Tomczak P. Interactive question based learning methodology and clickers: Fundamentals of computer science course case study. In Proceedings of the 6th International Conference on Human System Interactions (HIS 2013), Sopot, Poland, 6-8 June 2013, pp. 439-442.
7. Dzikovska, M.O.; Nielsen R.D.; Leacock C. The Joint Student Response Analysis and Recognising Textual Entailment Challenge: Making Sense of Student Responses in Educational Applications. *Language Resources and Evaluation* **2016**, *50*, no. 1, 67–93, https://doi.org/10.1007/s10579-015-9313-8.
8. McDonald, J.; Bird, R.J.; Zouaq, A.; Moskal, A.C.M. Short answers to deep questions: supporting teachers in large-class settings. *Journal of Computer Assisted Learning*, **2017,** *33*, no. 4, 306-319.
9. Burrows, S.; Gurevych I.; Stein B. The Eras and Trends of Automatic Short Answer Grading. *International Journal of Artificial Intelligence in Education* **2015**, *25*, no. 1, 60–117, https://doi.org/10.1007/s40593-014-0026-8.
10. Pulman, S.; Sukkarieh J. Automatic short answer marking. In Proceedings of the second workshop on Building Educational Applications Using NLP, 2005, Ann Arbor, Michigan. Association for Computational Linguistics, pp. 9-16.

 doi:10.20944/preprints202412.0750.v1

11. McClelland, J. L.; John M. St.; Taraban R. Sentence comprehension: A parallel distributed processing approach. *Language and cognitive processes* **1989**, *4*, no. 3-4, pp. SI287-SI335.
12. Elsayed, E.; Eldahshan K.; Tawfeek S. Automatic evaluation technique for certain types of open questions in semantic learning systems. *Hum. Cent. Comput. Inf. Sci.* **2013**, *3*, no. 1, 19, https://doi.org/10.1186/2192-1962-3-19.
13. Leacock, C.; Chodorow M. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities* **2003**, *37*, no. 4, 389-405.
14. Singthongchai, J.; Niwattanakul S. A method for measuring keywords similarity by applying Jaccard's, n-gram and vector space. *Lecture Notes on Information Theory* **2013**, *1*, no. 4, 159-164, DOI: 10.12720/lnit.1.4.159-164.
15. Aghahoseini, P. (2019). Short Text Similarity: A Survey. Available online: https://www.researchgate.net/publication/337632914_Short_Text_Similarity_A_Survey (accessed on 22/11/2024).
16. Pado, U.; and Kiefer C. Short answer grading: When sorting helps and when it doesn't. In Proceedings of the 4th workshop on NLP for Computer Assisted Language Learning (NODALIDA 2015), Vilnius, Lithuania, 11th May, 2015, pp. 42-50.
17. Magooda, A.; Zahran M.A.; RashwanM.; Raafat H.; Fayek M.B. Vector based techniques for short answer grading. In Proceedings of the Twenty-Ninth International Florida Artificial Intelligence Research Society Conference (FLAIRS), Florida, USA, May 2016, pp. 238-243.
18. Zhang, Y.; Shah R.; Chi M. Deep Learning+ Student Modeling+ Clustering: A Recipe for Effective Automatic Short Answer Grading. In Proceedings of the 9th International Conference on Educational Data Mining (EDM), Raleigh, NC, Jun 29-Jul 2, 2016, International Educational Data Mining Society, pp. 562–567.
19. Sakaguchi, K.; Heilman M.; Madnani N. Effective feature integration for automated short answer scoring. In Proceedings of the 2015 conference of the North American Chapter of the Association for Computational Linguistics: Human language Technologies, Denver, Colorado, May 31 – June 5, 2015, pp. 1049-1054, https://doi.org/10.3115/v1/N15-1111.
20. Jivani, A.G. A Comparative Study of Stemming Algorithms. *International Journal of Computer Technology and Applications*, **2016**, *2*, no. 3, 1930-1938.
21. Heilman, M.; Madnani N. The impact of training data on automated short answer scoring performance. In Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications, Denver, Colorado, June 4, 2015, pp. 81-85.
22. Kumar, S.; Chakrabarti S.; Roy S. Earth Mover's Distance Pooling over Siamese LSTMs for Automatic Short Answer Grading. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17), Melbourne, Australia 19-25 August 2017, pp. 2046-2052.
23. 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 (Volume 1: Long Papers), Berlin, Germany, August 2016, pp. 715–725, https://doi.org/10.18653/v1/P16-1068.
24. Nowak, J.; Taspinar A.; Scherer R. LSTM recurrent neural networks for short text and sentiment classification. In Proceedings of the International Conference on Artificial Intelligence and Soft Computing ICAISC 2017. *Lecture Notes in Computer Science*, **2017**, vol *10246*. Springer, Cham. pp. 553-562, https://doi.org/10.1007/978-3-319-59060-8_50.
25. Jing, R. A self-attention based LSTM network for text classification. *Journal of Physics: Conference Series*, **2019**, vol. *1207*, p. 012008. IOP Publishing, DOI: 10.1088/1742-6596/1207/1/012008.
26. Le, T. An attention-based deep learning method for text sentiment analysis. In Proceedings of the 2020 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 16-18 December 2020, pp. 282-286, IEEE, 2020, DOI: 10.1109/CSCI51800.2020.00054.
27. 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, 7–11 November, 2021. Association for Computational Linguistics, pp. 6030-6040.
28. Oraif, I. Natural Language Processing (NLP) and EFL Learning: A Case Study Based on Deep Learning. *Journal of Language Teaching and Research* **2024**, *15*, no. 1, 201-208.

 doi:10.20944/preprints202412.0750.v1

29. van Genugten, R.D.; Schacter D.L. Automated scoring of the autobiographical interview with natural language processing. *Behavior Research Methods* **2024**, *56*, no. 3, 2243-2259, https://doi.org/10.3758/s13428-023-02145-x.
30. Rujun, G.; Merzdorf H.E.; Anwar S.; Hipwell M.C.; Srinivasa A. Automatic assessment of text-based responses in post-secondary education: A systematic review. *Computers and Education: Artificial Intelligence* **2024**, *6*, 100206, https://doi.org/10.1016/j.caeai.2024.100206.
31. Liu, Q.; Hu, A.; Daniel B. Online assessment in higher education: a mapping review and narrative synthesis. Research and Practice in Technology Enhanced Learning **2025**, *20*, 7, https://doi.org/10.58459/rptel.2025.20007.
32. Fink, A.; Gombert, S.; Liu, T.; Drachsler, H.; Frey, A. A hierarchical rater model approach for integrating automated essay scoring models. *Zeitschrift für Psychologie*, **2024**, *232*(3), 209–218, https://doi.org/10.1027/2151-2604/a000567.
33. Chang, L.-H.; Ginter F. Automatic Short Answer Grading for Finnish with ChatGPT. In Proceedings of the 38th AAAI Conference on Artificial Intelligence, Vancouver, Canada (AAAI-24), 20-27 February 2024, vol. 38, no. 21, pp. 23173-23181, https://doi.org/10.1609/aaai.v38i21.30363.
34. Haller, S.; Aldea A.; Seifert C.; Strisciuglio N. Survey on automated short answer grading with deep learning: from word embeddings to transformers. arXi v preprint arXiv:2204.03503 (2022).
35. Funayama, H.; Sato T.; Matsubayashi Y.; Mizumoto T.; Suzuki J.; Inui K. Balancing cost and quality: an exploration of human-in-the-loop frameworks for automated short answer scoring. In Proceedings of the International Conference on Artificial Intelligence in Education (AIED 2022), *Lecture Notes in Computer Science* (LNCS,vol. 13355), Cham: Springer International Publishing, 2022, pp. 465-476.
36. Yang, K., Raković, M., Li, Y., Guan, Q., Gašević, D., & Chen, G. Unveiling the Tapestry of Automated Essay Scoring: A Comprehensive Investigation of Accuracy, Fairness, and Generalizability. In Proceedings of the 38th AAAI Conference on Artificial Intelligence, Vancouver, Canada (AAAI-24), 20-27 February 2024, vol. 38, no. 21, 22466-22474, https://doi.org/10.1609/aaai.v38i20.30254.
37. Chang, L.-H.; Rastas I.; Pyysalo S.; Ginter F. Deep learning for sentence clustering in essay grading support. In Proceedings of the 14th International Conference on Educational Data Mining (EDM 2021), pp. 614-618, arXiv preprint arXiv:2104.11556.
38. Lee, G.-G.; Latif E.; Wu X.; Liu N.; Zhai X. Applying large language models and chain-of-thought for automatic scoring. *Computers and Education: Artificial Intelligence,* **2024**, *6*, 100213, https://doi.org/10.1016/j.caeai.2024.100213.
39. Uto, M.; Itsuki A.; Tsutsumi E.; Ueno M. Integration of prediction scores from various automated essay scoring models using item response theory. *IEEE Transactions on Learning Technologies* **2023**, *16*, no. 6, 983-1000.
40. Shakeel, M.H.; Faizullah S.; Alghamidi T.; Khan I. Language independent sentiment analysis. In Proceedings of the IEEE 2019 International Conference on Advances in the Emerging Computing Technologies (AECT), Al Madinah Al Munawwarah, Saudi Arabia, 2020, pp. 1-5, doi: 10.1109/AECT47998.2020.9194186.
41. Lintean, M.; Rus V. Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics. In Proceedings of the Twenty-Fifth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2012), Marco Island, Florida, USA, 2012, pp. 244-249.
42. Nakamura, T.; Shirakawa M.; Hara T.; Nishio S. Semantic similarity measurements for multi-lingual short texts using Wikipedia. In Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Warsaw, Poland, 11-14 August 2014, vol. 2, pp. 22-29, DOI: 10.1109/WI-IAT.2014.76.
43. Le, T. A hybrid method for text-based sentiment analysis. In Proceedings of the 2019 International Conference on Computational Science and Computational Intelligence (CSCI) Las Vegas, NV, USA, 2019, pp. 1392-1397, https://doi.org/10.1109/CSCI49370.2019.00260.
44. Elalfy D.; Gad W.; Ismail R. A hybrid model to predict best answers in question answering communities. *Egypt. Inform. J.*, **2018**, *19* (1), pp. 21-31, https://doi.org/10.1016/j.eij.2017.06.002.

 doi:10.20944/preprints202412.0750.v1

45. Paluch M.; Jackowska-Strumiłło L. Hybrid Models Combining Technical and Fractal Analysis with ANN for Short-Term Prediction of Close Values on the Warsaw Stock Exchange. *Applied Sciences*, (MDPI), **2018**, *8*(12), 2473; doi:10.3390/app8122473.
46. Jackowska-Strumillo, L.; Cyniak, D.; Czekalski, J.; Jackowski, T. Neural model of the spinning process dedicated to predicting properties of cotton-polyester blended yarns on the basis of the characteristics of feeding streams. *Fibres Text. East. Eur*. **2008**, *16*, 28–36.
47. Słownik jezyka polskiego (Polish language dictionary), https://sjp.pl/slownik/odmiany/
48. Fletcher, S.; Islam M.Z. Comparing sets of patterns with the Jaccard index. *Australasian Journal of Information Systems* **2018**, *22*, https://doi.org/10.3127/ajis.v22i0.1538.
49. Graupe, D. *Principles of artificial neural networks*. 3rd ed.; Advanced Series in Circuits and Systems, *7*, World Scientific: Singapore, 2013, 363 pages.