International Conference on Recent Trends in Data Science and its Applications
DOI: rp-9788770040723.044

Gramcorrector: Improve English writing skills using Deep Learning

A. Shobanadevi
Department of Data science and business systems
SRM Institute of Science and technology
Chennai, India
shobanaa3@srmist.edu.in

L. AnanthaPadmanabhan
Department of Data science and business systems
SRM Institute of Science and technology
Chennai, India
ananthu017@gmail.com

G. Vadivu,
Department of Data science and business systems
SRM Institute of Science and technology
Chennai, India
vadivug@srmist.edu.in

Pao-Ann Hsiung
Department of Information Technology
National Chung Cheng University Taiwan
pahsiung@ccu.edu.tw

***Abstract*— English has now become a global language and is commonly spoken even amongst non-indigenous English societies. Grammar can be understood as an arrangement of words to form proper and meaningful sentences. English has a vast vocabulary and hence grammatical errors and typos are likely to happen for an average person. It is therefore important to have a medium that can help correct these errors and in doing so can help users improve their writing skills. The proposed work called Gramcorrector is an application that does both Grammar & spell correction. The availability of large volumes of data and good compute power paved the way for the development of such deep learning models. Gramcorrector is based on the Gramformer model, which uses the state of art transformer architecture. The core of this application is a Grammar & spell corrector model that has the capability to understand the intent of the sentence, identify all the possible errors and also apply accurate corrections. The corrected sentence or paragraph is displayed in the application UI along with the edits made on each sentence. This helps users to learn from their mistakes and also to improve his/her proficiency in English.**

***Keywords*— Grammar Correction, Natural Language Processing, Deep Learning, Transformer, SymSpell, Gramformer**

I. INTRODUCTION

English today has become a global language for communication. As a result, the number of English speaking people is increasing drastically even among the non-native communities. Learning the syntax and semantics of the English language is asignificant barrier for non-native speakers since it is a language with a big vocabulary as well as pre- defined grammar rules. This is made much more difficult when there aren't enough resources available to non-native English-speaking communities' education institutions to successfully teach the languages. Another significant barrier for amateur English language learners is the absence of proper contact with knowledgeable and deep English speakers. [1]

English is mostly learned by international students in academic settings including schools, universities, and other related institutions. Additionally, these students are more likely to engage with native speakers, which facilitates effective learning. On the other hand, educational institutions that help non-native groups learn English typically have few resources and a shortage of staff members who have the necessary training to impart English language instruction. The effectiveness with which the pupils learn the English language is directly impacted by these issues.

It is highly probable that non-native pupils who are enrolled in a school in a nation where the language is widely spoken will not have the chance to interact freely among native speakers. When given the chance to interact with native speakers, fresh language learners are often cognizant of and reluctant to use bad sentence structure and grammar. This not just impedes the process of learning but also has an impact on the pupils' self-confidence. Unrestricted and unfettered access to conversationalpractice significantly enhances and accelerates language learning. Moreover, this also prepares the users/students for effectivecommunications. Therefore, it is important to develop a system that addresses this scaled educational problem and also offers an unrestricted chance to learn English quickly and effectively.

Artificial Intelligence (AI) techniques are widely used today to solve grammar and spell correction [2]. With the development of Deep Learning (DL) architectures in NLP, rapid advancements were made in this area. The complex architectures such as Transformers have enabled models like Gramformer that work well on such problems. Gramformer can take as input any sentence or a group of sentences (2 or 3 sentences at max), apply grammar and spell correction and output the result. Gramformer works well when the number of input sentences or the sentence length are small. As sentences get bigger, the accuracy of the predictions also goes down. The proposed work identifies and addresses the shortcomings in such existing models and develops a new model that is more effective for grammar and spell correction.

The system developed in the proposed work consists of a UI where the users can type in or paste any text which needs to be evaluated. The text size can vary from a single sentence to a few paragraphs. The system uses deep learning to identify and correct both spelling and grammatical errors in the input text. The corrected text will be displayed in the UI. The UI contains an additional section that lists the edits or corrections made on each sentence in the text. This is useful for the users to understand their mistakes and learn from them. Thus, the proposed work can improve the language skills among users and in doing so effectively

DOI: rp-9788770040723.044
eliminate the aforementioned barriers to language acquisition in English.

II. Literature Survey

Every language has a predefined set of rules in which words can be arranged, called grammar. Following proper grammar is important in all languages to convey the required information. English is a language having a large vocabulary and as a result there can be frequent errors related to grammar and spelling when writing in English. Having a system that can identify and correct these errors is hence important. Such systems help users to see their mistakes, learn from them and in turn improve their language skill. At present, there are tools like grammarly that serve the need, but they still miss out on several errors and might sometimesconvert a grammatically correct sentence to an incorrect one. So, grammar correction is indeed a domain that still needs a lot of improvement.

In the English language, the same words can have different meanings depending on the context it is used. It is hence important for the system to have intelligence to understand the context and to make appropriate corrections. Intelligence stands the capacity to pick up information and abilities, and also to use them to solve a well-defined problem [6]. As described, it is difficult to find human intellect to teach English to non-native speakers. So it is key to develop an AI to overcome the lack of human resources. Even Though a few AI based solutions have emerged for the same cause, AI has never been successfully used to create cutting-edge instructional technologies. This is proven by looking at research literature that clearly describes the practices for promoting high-quality education as well as conversational language learning systems.

The authors of [4] argue that as new applications of AI are discovered and difficulties in educating and learning arise, a direct impact of AI on the subject of education is quickly approaching. The authors also mention how the area of education will be affected by the present wave of AI developments and how it will be changed in the future by new technologies and even the processing power of intelligent devices, one of which this research effort is modelled on. The success stories that have been achieved via the use of AI applications in education are highlighted by the authors in [8]. The outcomes are clear in the promotion of self-learning or self- development, virtual classrooms or laboratories, innovative schools that offer alternatives to traditional curriculum to all students, as well as virtual teaching [9]. In a study delivered to the Joint Research Centre of the European Union, the authors of [10] discuss the effects of AI on learning, teaching, and education. The adoption of customized and proactive students learning spurs the creation of ITSs based on AI algorithms [11].

English dialogue accompanies the Spanish dialogue in [12]. (EDC). The suggested EDC system supports elementary pupils in learning English as their second language of choice by taking the English language into account. The EDC approach motivates students by dividing English language instruction into two segments, namely the teaching phase and the discussion phase. The authors perform a pilot study to gauge the effectiveness of the EDC system. The findings indicate that the study's participants benefited from it. Additionally, theauthors note that most participants favoured the teaching phase over the conversing phase.

In [15], a more consistent body of research is given. The authors evaluate the 250 participants' English language acquisition using the Rasch model. Similar to [16], authors conduct research to look at how reading comprehension among non- native English-speaking students who are studying the language relates to lexical and grammatical understanding. 825 students who undertake the English as a Foreign Language (EFL) exam are included in the research. On the basis of grammar and vocabulary, the authors divide participants into high skills and low intelligence readers using a multi-layered neural network (NN). The study described in [17] is characterised by artificial language learning (ALL). As it focuses with the linguistic systems and researches the ecological validity of assessments of natural language ability, ALL examines language concepts as well as language development. Thus, it is clear from the analysis of current approaches that a fresh approach to enhancing language acquisition is feasible.

A neural network may be created in a manner similar to other Machine Learning applications to process unstructured text input. However, text data's large dimensionality will be a significant factor to consider while developing a solution. Neural network manipulation of high dimensional objects is challenging. Language modelling is the next significant NLP issue to be addressed. Any word sequence is given a probability by a language model, which is a probabilistic model. Learning a language model that gives strong probability to very well sentences is regarded as the task at hand.

The input of the neural network, that is the text data has to be converted to numerical form like vectors so that the network can process the data. The text data may be described mathematically using word embeddings, in which each word is represented by a higher-dimensional vector. Thus, the relationship between the words may be determined with the use of these word embedding approaches. Word embeddings are, in essence, learnt representations in which words with similar meanings have similar representations.

Word embedding can be done using standard techniques such as one hot encoding, count vectorizer etc. Each unique word in the corpus constitutes a dimension of the final word vector in a single hot encoding. Every dimension could have a value of zero or one when creating word vectors, depending on whether the word is present or not. Asentence is vectorized in Count vectors depending on the number of words in the sentence. Regarding word embeddings, the full vocabulary may be taken into account, or it may be limited to a select few top terms that are taken into account depending on its frequency. This gives control over the created word vector's dimensionality.

Apart from the standard techniques, pre-trained word embedding models can also be used to generate word vectors. These models are trained on large volumes of data and hence such models RNN+ LSTM + Transformer comparison

DOI: rp-9788770040723.044

There also exists python libraries that assist in correcting grammar. These libraries take in the input sentence, apply respective algorithms and return as output either the corrected sentence or the list of possible edits. Gingerit is an open-source python library that can be used for correcting grammatical errors. It is a wrapper around the gingersoftware.com API. Ginger software is an AI- powered writing assistant that corrects texts, improves and boosts the style and creativity of writing. It corrects spelling, punctuational and grammatical errors. The major drawback of Gingerit is that it is dependent on gingersoftware.com API. Updates in this API can cause the library to stop functioning.

Gramformer is a different open-source python library that can detect, highlight and correct grammatical errors. Gramformer exposes three separate interfaces to a family of algorithms used for detecting, highlighting and correcting grammar errors. Gramformer is a generative model even though its goal is to post-process generative model results. In general, all generative models have an uncontrollable tendency to occasionally produce erroneous text. A quality estimator (QE) is therefore introduced to ensure that the gramformer grammar fixes (and highlights) are as accurate as feasible. To return just the top-N candidates, it calculates an error correction quality score and applies it as a filter on the candidates.

According to the official documentation some of the use cases of Gramformer includes post- processing machine generated text, Assisted writing for humans, Custom Platform integration etc.

The major shortcomings of the Gramformer library is in dealing with multiple sentences. When more and one sentence is fed to the Gramformer model, the performance drops rapidly. Also, Gramformer is not efficient at correcting spell errors and in some cases, the incorrect spelling errors can lead to the model not being able to correct grammar.This objective of the research is to address these issues observed in the existing work.

III. Methodology

A. Grammar Correction Application

Grammar correction application is one that helps the user check for grammatical errors in the text they type. The text can either be a sentence or a few sentences or even paragraphs. The users can type in or paste or text for which they perform grammar checks. The application pre-process the input text and uses a deep learning model to check for grammatical mistakes and correct the input text. The corrected text along with the list of corrections for each sentence in the text are displayed in the application UI. The list of corrections for the input text helps people to understand their mistakes and in turn help them improve their language. Fig 1 shows the architecture of the proposed grammar correction application.

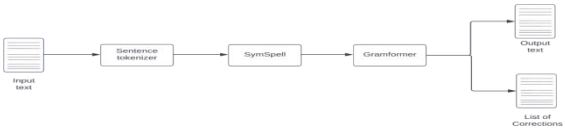
The application makes use of the Gramformer model at its core for performing grammatical corrections. One limitation of using the Gramformer model is that it fails to do well when dealing with text having multiple sentences. Gramformer, on the other hand, works well when the text contains only a single sentence. For real world scenarios, the user inputted text can have multiple sentences and or even a few paragraphs. This application assumes that the grammar of a particular sentence is independent of the other sentences in the text. Taking this assumption into consideration, the input text is tokenized to separate out each sentence in the text. This process is taken care of by the “Sentence tokenizer” component shown in Fig. 1. This component uses the sentence tokenizer function in NLTK library to split the input text to different sentences. This function loads a saved tokenizer model for the English language and uses the model to generate a list of sentences. The model uses whitespaces and punctuation marks to tokenize sentences. It also realigns punctuation that falls after the period but should otherwise be included in the same sentence.

The second limitation of the Gramformer model is in correcting spell errors. Gramformer is able to correct some words that are spelled incorrectly but it fails to do so for the majority of the words, especially those words that are not frequently used. The typo related errors can have a significant impact on the grammar. For example, a word having spell error “ther” can be corrected to “there” or “their” or be used as it is. Using any of the mentioned words will have a different impact on how the Gramformer model treats the sentence. Hence, it is important to have a check on the typo errors in the words belonging to the input text.

To handle spell errors, the Symspell module was added to the pipeline. Symspell is yet another open-source python library for spelling correction and fuzzy search. It makes use of the Symmetric Delete spelling correction method, which lessens the difficulty of creating edit candidates and looking them up in dictionaries for a certain Damerau-Levenshtein distance. Compared to the conventional method of deletes, transpositions, replacements, and inserts, it is six orders of magnitude quicker and language agnostic. In contrast to other algorithms, only deletions are necessary; there are no transpositions, replacements, or additions. Deletes of the dictionary word are created by transposing + replacing + inserting the input phrase.

As mentioned, the input text will be divided into different sentences. Each sentence is made up of several words and each word has to be checked for any errors. NLTK library is again used here to tokenize the sentence to a list of words. Each word is then fed to the Symspell model which uses a look up dictionary to suggest alternative words if any error is identified. Among these suggested words, the word that is closest to the word having error is chosen. The following configurations are possible for Sympell module:

- Return the closet word to the error word
- Return back the original word if no correct words are found within an edit distance
- Avoid correcting words that follow a predefined pattern
- Maintain origins casing (for example the word pATTern should be maintained as it is without correcting it to pattern)



DOI: rp-9788770040723.044

Fig. 1: Grammar Correction application architecture

The current implementation uses the second configuration that returns the original word if no matches are found within an edit distance of 2. Once the spell errors are corrected for each word, then the words are combined back to generate the original sentence. The new sentence will hence be free of spell errors. These corrected sentences are then fed to the Gramformer stage.The individual words present in the sentence; it is important to perform spell correction.

Once the spell correction is done for each sentence, it is then fed to the Gramformer model. Gramformer model is a pre-trained transformer model which is available as an open-source python library. Gramformer is a library that exposes 3 separate interfaces to a family of algorithms to detect, highlight and correct grammar errors. Also, to ensure the corrections and highlights recommended by the model are of high quality, it comes with a quality estimator. Gramformer can detect, highlight and correct grammatical errors. Gramformer is a generative model, and it has been noted that generally speaking, all generative models occasionally have a tendency to produce erroneous text that is impossible to control. Therefore, a quality estimator (QE) is introduced to ensure that the quality of the predictions (grammar correctionsand highlights) from the gramformer model remain as accurate as feasible. The error correction quality score may be estimated using the quality estimator, and it is used as a filter on the Top-N candidates to just return the top options.

Gramformer model takes in each input sentence and it has in-built functions to correct the grammatical errors as well as to return the list of edits made in each sentence. Having a sentence tokenizer overcomes one of the major limitations of the Gramformer model, where it was not able to do grammar correction properly if there are multiple sentences. With a sentence tokenizer component in place, the input text can contain any number of sentences since this component ensures that only one sentence is processed at a time. The application process and corrects one sentence at a time. The output from Gramformer stage is finally rendered in UI. The UI contains the sentence that is grammar corrected and also the list of edits done for each sentence. The list of edits helps the users easily understand the mistakes made in the input text, rather than having to compare both input text and corrected output to retrieve the list of corrections. The screenshot of the application is displayed in Fig. 2.

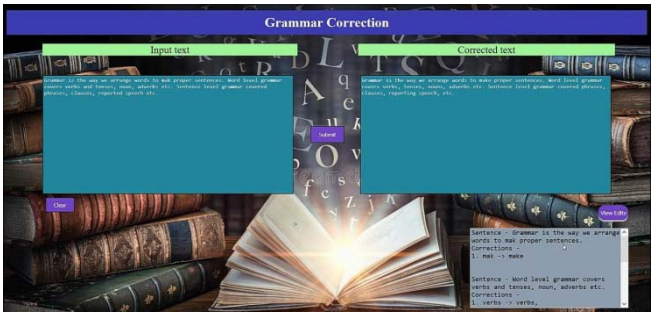


Fig. 2. Grammar Correction Application

The application was developed using the Flask framework in python, along with HTML and JavaScript. Flask is a web application framework written in python. The landing page of the application is as shown in Fig.2. The user has a text box under the ‘Input text’ section where they can either type in the required contents or copy paste from any document. Once done, they can click on the submit button to perform grammar correction. The corrected text will be displayed in the text box under the ‘Corrected text’ section. This text box is disabled and cannot be edited by the user. They can only view or copy the contents in the corrected text box. There is a clear button that can be used to clear the data from input text box, corrected text box and view edits window. The view edits button on thebottom right hand side of the screen can be used to view or hide the list of edits. The data in the ‘View Edits’ section will have the sentence and the list of edits made for each sentence is displayed right below it. This will be followed by the second sentence and so on. Thus, the application can be utilized to perform grammar correction and also view the errors in the input text.

IV. Results and Discussion

The grammar correction application can be used to both correct the grammatical errors and also view the corrections. It can correct errors in a single sentence, multiple sentences and even multiple paragraphs. Fig. 3 shows a few examples of input text containing just one sentence and the corrections made by the grammar correction application.

	Input text	Corrected text	List of edits
1	Matt like fish	Matt likes fish.	like -> likes fish -> fish.
2	the collection of letters was original used by the ancient Romans	the collection of letters was originally used by the ancient Romans.	original -> originally Romans -> Romans.
3	We enjoys horror movies	We enjoy horror movies.	enjoys -> enjoy movies -> movies.
4	Anna and Mike is going skiing	Anna and Mike are going skiing.	is -> are skiing -> skiing.
5	I walk to the store and I bought milk	I went to the store and bought milk.	walk -> went I -> milk -> milk.
6	I will eat fish for dinner and drank milk	I ate fish for dinner and drank milk.	will eat -> ate milk -> milk.
7	what be the reason for everyone leave the company	what is the reason for everyone leaving the company?	be -> is leave -> leaving company -> company?
8	Everyon is happy	Everyone is happy.	Everyon -> Everyone happy -> happy.

Fig. 3. Input text, Corrected text and list of correction for single sentence text by grammar correction application

From Fig. 3, the single sentence input text, the corrected text and the list of corrections can be observed. For sentences 1-4, the application was able to correct the errors such as using proper singular or plural form of the word and also adding fullstops at the end of each sentence. In sentences 5 and 6, the application was able to correct the sentence to maintain the past tense. Multiple corrections were also made for each sentencedepending on the necessity. For sentence 7, the application was able to rephrase the question such that it becomes more meaningful and correct the tense. It was also able to identify that the text is a question and added a question mark symbol at the end.

For sentence 8, it was able to identify a spell error for the word “Everyone” and correct it accordingly.

	Input text	Corrected text	List of edits
1	How is you? Where is you going?	How is your work? Where are you going?	your work? is -> are
2	Because of new computing technologies, machine learning today are not likes machine learning of the past. It is born from pattern recognition and the theory that computers cans learning without being programmed to perform specific tasks; researchers interested in artificial intelligence wanting to see if computers could learn from data. The iterative aspect of machine learning are important because as models are exposed to new data, they are able to independently adapt. They learn from previo computations to produce reliable, repeatable decision and results. It’s a science that’s not new – but one that has gained fresh momentum. While many machine learning algorithms have been around for a long time, the abilities to automatically apply complex mathematical calculation to big data – over and over, faster and faster – is a recent development. "	Because of new computing technologies, machine learning today was not like the machine learning of the past. It is born from pattern recognition and the theory that computers can learn without being programmed to perform specific tasks; researchers interested in artificial intelligence wanted to see if computers could learn from data. The iterative aspect of machine	are -> was likes -> like -> the cans -> can learning -> learn wanting -> wanted machine learning are important because as models are exposed to new data, they are able to independently adapt. They learn from previo computations to produce reliable, repeatable decision and results. It’s a science that’s not new – but one that has gained fresh momentum. While many -> learning algorithms have been around for a long time, the abilities to automatically apply complex mathematical calculation to big data – over and over, faster and faster – is a recent development. " ->

Fig. 4. Input text, Corrected text and list of correction for multiple sentence text by Gramformer model

Fig. 4 highlights one of the shortcomings of the gramformer model. Due to technical constraints, it was not made very complex or trained on a very large dataset and hence it fails to interpret certain sentences properly. Thus, it is not suitable for input texts having multiple sentences. In the first example, the first sentence “How is you?” is incorrectly updated as “How is your work” rather than correcting it to “How are you?” The second sentence is corrected properly. The second example hasseveral sentences and it makes the gramformer limitations clearer. Few of the words in the first two sentences are corrected and the remaining data is just ignored. The sentences towards the end of the input text are replaced with an empty string as shown in Fig. 4. This shows the importance of having the sentence tokenizer component.

	Input text	Corrected text	List of edits
1	How is you? Where is you going?	How are you? Where are you going?	is -> are is -> are
2	Because of new computing technologies, machine learning today are not likes machine learning of the past. It is born from pattern recognition and the theory that computers cans learning without being programmed to perform specific tasks; researchers interested in artificial intelligence wanting to see if computers could learn from data. The iterative aspect of machine learning are important because as models are exposed to new data, they are able to independently adapt. They learn from previo computations to produce reliable, repeatable decision and results. It’s a science that’s not new – but one that has gained fresh momentum. While many machine learning algorithms have been around for a long time, the abilities to automatically apply complex mathematical calculation to big data – over and over, faster and faster – is a recent development. "	Because of new computing technologies, machine learning today is not like the machine learning of the past. It is born from pattern recognition and the theory that computers can learn without being programmed to perform specific tasks; researchers interested in artificial intelligence want to see if computers could learn from data. The iterative aspect of machine learning is important because, as models are exposed to new data, they are able to independently adapt. They learn from previous computations to produce reliable, repeatable decisions and results. It’s a science that’s not new – but one that has gained fresh momentum. While many machine learning algorithms have been around for a long time, the ability to automatically apply complex mathematical calculations to big data – over and over, faster and faster – is a recent development.	are -> is likes -> like -> the cans -> can learning -> learn wanting -> want are -> is because -> because, previo -> previous decision -> decisions abilities -> ability calculation -> calculations " ->

Fig. 5. Input text, Corrected text and list of correction for multiple sentence text by grammar correction application

Fig. 5 shows the benefits of having the sentence tokenizer component. In a text having multiple sentences, each sentence is separated and treated independently. As a result, it can work around the limitations of the Gramformer model to deliver better results. Comparing Fig. 4 data with Fig. 5 data proves that the grammar correction application is able to generate more accurate predictions than the standalone Gramformer model. It can correct spell errors, punctuation errors and also take care of the grammatical mistakes. Hence this can be used by anyone who is willing to improve their language skills by understanding and learning from the mistakes.

V. Conclusion and Future Work

English has a vast vocabulary and hence grammatical errors and typos are likely to happen for an average person. It is therefore important to have a medium that can help correct these errors. We proposed a new application called Gramcorrector that can help users improve their writing skills. The users can type in the required text and get it validated and also corrected in case of any grammatical errors. Transformer architecture has brought a lot of advancements in the field of NLP. As a result the proposed application uses a transformer based model called Gramformer to correct grammatical errors.. Corrected sentences and the edits made are displayed in the application UI. This helps users to understand the mistakes they made and in turn improve his/her proficiency in English. Even Though the application can handle the majority of language errors, there is a scope for further improvement, especially when dealing with spell errors. As a future scope, the spell correction module SymSpell needs to be improved so that it can suggest propercorrections also taking into account the context where the particular word is used in. Also, the training data can be enhanced to improve the vocabulary of this module. Similarly, the Gramformer model can be improved using larger datasets and also using more complex architecture such as GPT. Availability of good quality datasets in other languages other than English can also drive the development of this application towards such languages. This is also in the future scope of the Gramcorrector application.

References

[1] D. Crystal, English as a Global Language - Second Edition. 2003.

[2] B. Turnbull, “The use of English as a lingua franca in the Japanese second language classroom,” J. English as a Ling. Fr., vol. 7, no.1, pp. 131–151, 2018.

[3] E. Protassova, “Teaching Second Languages at Pre-School Age,” in The History of Language Learning and Teaching Vol. II. 19th-20th Century Europe, 2018.

[4] S. A. D. Popenici and S. Kerr, “Exploring the impact of artificial intelligence on teaching and learning in higher education,” Res. Pract. Technol. Enhanc. Learn., vol. 12, no. 22, pp. 1–13, 2017.

[5] J. Schmidhuber, “Deep Learning in neural networks: An overview,” Neural Networks, vol.61, pp. 85–117, 2015.

[6] J. Shabbir and T. Anwer, “Artificial Intelligence and its Role in Near Future,” Jounral Latex Cl. Files, vol. 14, no. 8, pp. 1–11, 2015.

[7] S. Sumathi and P. Surekha, Computational Intelligence Paradigms Theory and Applications. 2010.

[8] Sitharthan, R., Vimal, S., Verma, A., Karthikeyan, M., Dhanabalan, S. S., Prabaharan, N., ...&Eswaran, T. (2023). Smart microgrid with

the internet of things for adequate energy management and analysis. Computers and Electrical Engineering, 106, 108556.

[9] E. Gomede, F. H. Gaffo, G. U. Briganó, R.M. De Barros, and L. D. S. Mendes, “Application of Computational Intelligence to Improve Education in Smart Cities,” Sensors, vol. 18, no. 267, pp. 1–26, 2018.

[10] I. Tuomi, “The Impact of Artificial Intelligence on Learning, Teaching, and Education,” European Union JRC Science for Policy Report.
pp. 1–47, 2018.

[11] F. S. Gharehchopogh and Z. A. Khalifelu, “Using Intelligent Tutoring Systems in Instruction and Education,” in International Conference on Education and Management Technology, vol. 13, pp. 250–254, 2011.

[12] Y. T. Huang, J. C. Yang, and Y. C. Wu, “The development and evaluation of English Dialogue Companion system,” Proc. - 8th IEEE Int. Conf. Adv. Learn. Technol. ICALT 2008, no. July 2008, pp. 864–868, 2008.

[13] Rajesh, M., &Sitharthan, R. (2022). Introduction to the special section on cyber-physical system for autonomous process control in industry 5.0. Computers and Electrical Engineering, 104, 108481.

[14] R. K. Al-Tuwayrish, “An evaluative study of machine translation in the EFL scenario of Saudi Arabia. Advances in Language and Literary Studies, 7(1), 5-10.,” Adv. Lang. Lit. Stud., vol. 7, no. 1, pp. 5–10, 2016.

[15] V. Aryadoust, “Fitting a Mixture Rasch Model to English as a Foreign Language Listening Tests: The Role of Cognitive and Background Variables in Explaining Latent Differential Item Functioning,” Int. J. Test., vol. 15, no. 3, pp. 216–238, 2015.

[16] V. Aryadoust and P. Baghaei, “Does EFL Readers’ Lexical and Grammatical Knowledge Predict Their Reading Ability? Insights From a Perceptron Artificial Neural Network Study,” Educ. Assess., vol. 21, no. 2, pp. 135–156, 2016.

[17] M. Ettlinger, K. Morgan-Short, M. FarettaStutenberg, and P. C. M. Wong, “The Relationship Between Artificial and Second Language Learning,” Cogn. Sci., vol. 40, no. 4, pp. 822–847, 2016.