A New Natural Language Processing-Based Essay Grading Algorithm

S. Suman Rajest
Professor, Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India.

R. Regin*
Assistant Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, India. regin12006@yahoo.co.in

Shynu T
Master of Engineering, Department of Biomedical Engineering, Agni College of Technology, Chennai, Tamil Nadu, India.

Steffi. R
Assistant Professor, Department of Electronics and Communication, Vins Christian College of Engineering, Tamil Nadu, India.

Abstract: The evaluation of an English essay is one of the most significant and difficult activities that is manually carried out by knowledgeable and capable instructors and faculty members. The advancement of science and technology has made it possible to automatically evaluate an English essay by employing techniques pertaining to natural language processing. For any given English essay, the intelligent system provides a generic evaluation as well as the topic/question correlation. This evaluation is based on the NLP multiple neural network model, which was used to build the system. The evaluation of essays according to worldwide standards is the primary contribution of this innovation. Any worldwide grading system, such as the Graduate Record Examination, the International English Language Testing System, etc., is qualified to make use of the grading standard. The algorithm gives users the opportunity to test their knowledge on a range of criteria, from the most basic to the most complicated, that are included in the scoring of an English essay.

Keywords: Automatic evaluation, Natural Language Processing (NLP), General Assessment, And Topic/Question Correlation, Evaluation Method,

Introduction
Graduation Record Examination (GRE), International English Language Testing Systems (IELTS), etc., are growing in popularity since their scores are increasingly being accepted as admissions requirements at a wide range of colleges and employers [1]. The Graduate Record Examination (GRE) is a standardised test that serves as the groundwork for admission to graduate school at international institutions [2-5]. The Educational Testing Service (ETS) owns and manages the Graduate Record Examination (GRE), which attempts to assess candidates' abilities in verbal reasoning, numeric reasoning, analytical writing, and critical thinking [6]. As a result, more and more students are signing up to take the examinations, and there's a sizable cushion between now and when we need to start publishing the results of our assessments of their English essays [7-11]. English language proficiency in Europe, Canada, and Australasia is often measured by the International English Language Testing System (IELTS), an international standardisation test [12]. The enormous number of applicants causes
inconsistency in offering the evaluation service; hence, a proposed model is developed and put into action to automate the evaluation process and make it available as an API (Application Programming Interface) service [13-17].

A company that is in charge of standardised tests like the GRE, IELTS, etc., can devote considerable resources to advertising [18]. Meanwhile, a novel model called essay grader implements seven Natural Language Processing (NLP) strategies, each yielding a probabilistic outcome, and this analytical result-set is combined into five parametric models consisting of a scoring scheme ranging from 0 to 10 points, which are then used to evaluate the candidate's performance on the exam [19-22]. In order to make the grading system for these parametric models more clear and easy to grasp for the user, we have labelled each variable [23]. The User Experience (UX) in quality management is improved by this mapping pattern of NLP servers to user-understanding models [24-25].

The multiple Hidden Markov model's natural extension is the multinomial Hidden Markov model. To put it simply, an HMM is a type of statistical Markov model in which the represented system is believed to be a Markov process with hidden states [26]. Though its internal state is hidden from view, its output is entirely dependent on its outward appearance. There is a probability distribution over the set of outcomes for each state. In order to construct complicated models with sufficient unobservable states, HMM give a conceptual toolkit [27-32]. They form the backbone of numerous applications, including those used for gene mapping, profile searches, and the detection of regulatory sites. Bidirectional Neural Networks are recurrent neural networks in which the input result is reliant on the output result from the other network [33-37]. One RNN generates the two opposing neurons, one for forward states and one for reverse states. Due to the lack of interactions between the neurons in the two directions, BNN can be taught using the same algorithms as RNN [38]. During forward pass training, the forward and backward states are passed before the output neurons, while during reverse pass training, the output neurons are passed before the forward and backward states [39]. The weights are adjusted once two processes, forward and backward passes, have been completed [40].

LSTMs are used because they can learn to bridge minimal time lags over 1000 discrete time steps by enforcing constant error flow through "constant error carrousels" (CECs) within special units, called cells; this is because the bidirectional neural network has a higher hit ratio in obstacles like vanishing error problems and time delays. Two major subfields of machine learning are supervised and unsupervised learning [41-47]. For instance, the k-means algorithm can be used to create hard clustering, which is one of two techniques to clustering that make up unsupervised learning. Conversely, the Gaussian Mixture Model can be used to create soft clusters [48-52]. The model uses a normal distribution to classify and cluster the data, with the mean determining the cluster's centre and the covariance providing an indication of the spread of the data. Each word in the word2vector model is independently embedded with a vector. However, the continuous bag of words accomplishes the same goal while using a larger number of words as input. Target word "corona" might be placed in the context of "virus" and "pandemic," for instance. With the ability to represent the CBOW architecture as a deep learning classification model, we can now use context words as input and attempt to predict the target word [53-61].

The primary use of the unsupervised, non-linear method known as t-SNE (T-Distributed Stochastic Neighbor Embedding) is in the realm of exploring and visualising high-dimensional data. This gives us a visual representation of data in high dimensional space [62-77]. Similarity measures between pairs of instances in high and low dimensions spaces are computed by the t-SNE algorithm. It then applies a cost function to try to find the sweet spot between these two metrics of similarity [78]. As one of the most crucial and time-consuming academic jobs, evaluating an English essay is typically performed manually by trained and competent faculty members [79]. Now that we have advanced enough in our understanding of how to analyse natural language, we can easily automate this evaluative process. Our project's goal is to analyse the English essay, allowing the organisation holding these examinations to concentrate on other examinations and students to practise free will while improving their writing skills [80]. The project serves as a smart system, constructed on many neural network models, that provides a standard score for each essay written in English [81-93].

**Literature Survey**

There have been a lot of earlier proposals similar to this one. Numerous authors have contributed to the development of this concept by publishing numerous academic works [94]. We'll talk about a few of the books and movies below. The model will serve the user well when auto-spell checking his essay or other text for the next appropriate term. Probability theory is at the heart of the model [95-99]. Let's say a user is typing a sentence; this model can offer suggestions for terms that work well with what they've already typed [100-101]. For the
model's optimal training and prediction abilities, a massive corpus of texts is provided as input. Taking as input only the first n words of each phrase, a "n-gram" model uses those words to generate predictions about the rest of the sentence in order to produce meaningful sentences [102-111]. Our proposal can make use of this model, which improves sentence connectivity based on the training dataset [112-119]. However, it has many more potential uses, including auto-correct in various messaging apps, word suggestions depending on the sentence written, etc. In this way, we can avoid situations when we have to retype the same words over and over again, saving both time and effort. To use, we just need to select the suggested word and use it in our phrase [120]. The essay's sentences and overall quality can be judged in this way. So, it's a probabilistic model with high precision, provided it's trained on a high-quality dataset [121-125].

A stochastic function, rather than a hyperbolic activation function, is used to create the cheap LSTM. Gates will replace the complicated mathematics used by neural networks [126]. The LSTM's "gates" are the basic computational units, and they may choose which sequence elements are most crucial to keep and which can be discarded [127]. As it gathers information, it organises it into sequences and passes them along. The primary application of LSTM is in the creation of new words [128-131]. LSTM's primary function is to identify the most significant words in a given phrase. It then uses these keywords to create predictions about what the correct word will be. The sigmoid function is used in the LSTM's Gates. Since it makes use of the sigmoid function, rather than the tanh function, its values are strictly between 0 and 1, rather than between -1 and 1. Consequently, the sigmoid function might instruct the neural network to focus on more relevant information while disregarding less significant input [132-139]. An LSTM cell utilises three gates to control the flow of data. They ignore the input and output gates and the gate altogether. This is how the long short-term memory (LSTM) works to boost the performance of neural networks [140].

Its sole application was in the field of linguistics, where it was used to identify words in a text and place them in the appropriate grammatical category [141-145]. To classify the words into their component components, we'll use a three-layer perceptron layer with n inputs, where n is the number of words in the dataset. According to the previous experiment, while the computational cost of training the network is significantly higher than the n-gram model, the accuracy was 99.4% without over-fitting the data [146]. To achieve this precision, an elastic hidden perceptron layer was used. Context length is dynamic at every given word level in the tagging process. A backpropagation of error algorithm is used during the training procedure [147]. To ensure that the elastic neural tagger's connection weights remain consistent regardless of the input length, a new training approach has been adopted. The goal of the novel training approach is to arrive at the same subsets of connection weights for the neural taggers with brief inputs as are achieved from training these taggers directly [148-151]. Training the elastic neural tagger involves treating it as a perceptron that has evolved from a simpler version; steps are taken to increase the complexity of the model from within. In particular, training begins with the smallest perceptron possible. As a result of training, a new perspective is established through gradual improvement, and then trained once more [152-163]. This procedure of incremental expansion and training is repeated until a perceptron with the greatest (l, r) is constructed and taught. Consequently, we can see that the words from the dataset will be tested, and the words will be classified based on the kind of parts of speech, once the training process is complete [164].

When we talk about neural networks, we often refer to RNNs, which stands for "Recurrent Neural Network" and describes a type of neural network that can process data in both forward and reverse directions [165]. For both language modelling and text creation, NLP relies only on RNN. There are two main types of clusters, soft and hard, and they are both used in the clustering technique to classify the word component into specific grouping. A component can only ever belong to one of several "hard clusters," or binary categories. The K-means algorithm is used for clustering, where training datasets are fed into nodes iteratively [166]. There are cases where the K-means method is too thorough, and necessary parts are left out because they cannot be assigned to a single cluster. K-means is a method for computing the average distance between a central point and a set of data points. Integrating Word2vec with additional keywords to enhance semantic expression and subject relevance [167]. The keywords associated with each paper will each have their own distinct vector. In order to discover the relationship between the words and the themes, we need to identify a reference word and then employ the Word2Vec technique [168].

The optimal centroid is found at the point where the mean and standard deviation are both zero. As a result, k-means is used to find the centroid positions [169]. The EM value represents the centroid of a region with a high concentration of data points. Initial iterations involve calculating the Euclidean distance between a random collection of data points. One data point is chosen as the centroid based on the minimum distance between any
two data points [170]. Using the RSS (residual sum of squares) value from each iteration, the centroid's coordinate is adjusted to incorporate new data points. The weighted K-means algorithm produces stunning results in terms of the number of clusters, but the standard K-means algorithm produces negligible results, as shown by experimental data [171]. In addition, the weighted K-means algorithm generates more refined semantic information and subject relevance to aid cluster splitting [172-174].

The primary benefit of this approach is that it simplifies the utilisation of massive datasets by letting them execute in lower-dimensional areas [175]. To lower the dimensionality of the data and profit from the topological preservation of information, they have concentrated on models based on spectral clustering and topological unsupervised learning, i.e. the t-SNE (Stochastic Neighbor Embedding). Data will be classified as "similar" or "dissimilar" once the t-SNE algorithm is learned. It's no surprise that this data is more malleable than the raw variety. Taking into account two parameters in a 2d graphical representation of multiple data points, these points are mapped and clustered with soft or hard clustering using the algorithms k-mean or gaussian mixture model; consequently, it is necessary to evaluate these clusters for relevance or degree of similarity using the normal distribution to scale the datapoint's value against the distribution curve [176]. Vector values denoting similarities are allocated from a conditional probability distribution to the data points with the greatest distribution and the smallest high dimensional Euclidean distance [177]. With the T-SNE algorithm, we estimate the similarity between word components by plotting them in a matrix based on the distance between dissimilar and similar data points, with the maximum similarity value located near the diagonal of the matrix [178]. This matrix is called a diagonal matrix because the highest similarity is one. This algorithm uses just two parameters to preserve cluster similarity, word covariance, and relevance when clustering data points in a low-dimensional graph. Multiple iterations are required to cluster the data points based on their similarity, which is implemented in a low-dimensional network. Each iteration of the procedure yields a matrix; this continues until the matrix value obtained from a high-dimensional graph is reached [179]. Similarity between data points is determined by mapping them onto a t-distribution curve graph in a low-dimensional space.

The framework will be broken down into four types of writing: narrative, persuasive, descriptive, and expository. The use of first-person pronouns like "I," “she,” "he," etc. is a defining characteristic of narrative. The goal of argumentative or persuasive writing is to persuade the reader to accept the author's point of view. In most cases, the differences between competing points of view are striking. An attempt is made to back up the author's claim with evidence, arguments, or quotes from authorities in the field. The purpose of descriptive writing is to create an image in the reader's mind of the scene or experience being described. It could be a person, a place, or even just the vibe of a particular area. To elaborate; to provide further explanation. Therefore, the primary goal of this type of writing is to educate the reader about a specific subject. Writing like this is typical of scholarly publications, guidebooks, and other specialised types of writing. Named Entity Recognition, Part of Speech tagging, and Sentence Parsing are used for this purpose. This strategy employs a rule-based evaluation method known as rubrics. Narrative essays are the most common type of assignment graded using the NAPLAN (National Assessment Program - Literacy and Numeracy) marking rubric. In the first stage, we use a Stanford NER Tool to count the amount of words in an essay. A narrative essay often follows a single point of view character and can be written in either the first or third person. The 10 criteria on the rubric are used to evaluate various aspects of a student's essay (figure 1).

Figure 1: National Assessment Program - Literacy and Numeracy (few)
The NAPLAN grading rubric is used to determine an essay's final grade if it has been determined to be a narrative. The fact that it is so narrowly and rigidly focused on one subject is its biggest drawback. Although the techniques do for a distinct categorization of narrative essays, the genre categorization algorithms are not yet compiled into a single component, diminishing the proposed system's overall efficacy. One such system that takes into account linguistic aspects of the text is the E-rater. Several Natural Language Processing (NLP) methods are incorporated into the system to extract features from a database of example essays that will serve as the foundation for the grading algorithm. To simplify its analysis, E-rater considers that the characteristics of a good essay would not be significantly different from those of a similarly well-written essay, and vice versa for poor essays. To date, e-rater scores have been derived by a linear combination of high-level features computed for each essay, with weights set via regression of human evaluations on the features. The term "macro feature" is also used to describe these aspects. The values of these microfeatures are the result of a combination of a number of smaller, more specific features called macrofeatures. NLP is used to extract all of these macro and microfeatures. Common methods for predicting human performance typically involve 10 macro characteristics. Organization, progression, grammar, usage, mechanics, style, word length, word choice, collocation, preposition, and sentence variation are the 10 macro features we'll be looking at. When the scoring model is tailored to each query, these 10 macro factors are employed in conjunction with two word usage features unique to the prompt to forecast human scores. The E-rater V.2 scoring system is mostly unaltered from its predecessors, but it now makes use of a much smaller and more meaningful set of criteria, such as Style Measures and Lexical Complexity.

When all of the essays' automated (e-rater) ratings have been tallied, ETS employs a set of evaluation criteria to determine the models' efficacy. There are performance standards that are applied to the independent assessment sample that is used to check the accuracy of the scoring models. A more generalizable measure of performance that is in line with what would be observed in future data would be the outcomes of the evaluation sample separate from the sample used to develop the model. Please consider the following as criteria: In general, automated scoring capabilities are built with preconceived notions and restrictions on the kind of assignments they will evaluate. Therefore, before implementing automated scoring, it is important to determine whether or not the capability's design is a good fit for the intended purposes of the assessment or other application. The procedure consists of a comparison between the construct of interest and the capability, a review of the task design, a review of the scoring rubric, a review of the human scoring criteria, a review of the goals for reporting scores, and a review of any claims or disclosures. Other system enhancements permit the development of a uniform scoring system. The 'Lexical Complexity' feature module, for example, takes into account word-based characteristics, word frequency, and word length, but it does not take into account the whole context in which the words are employed. Therefore, "nonsense text," which uses sophisticated language but adds nothing to the passage's meaning, can trick the feature.

Markit is an automatic Essay grading system proposed by Robert Williams and Heinz Dreher. Markit gets to work once an academic paper is fed through a variety of Natural Language Processing (NLP) methods in order to construct a corresponding proprietary knowledge representation. The student's grade is then calculated based on the percentage of correct information contained in the model response that was also present in their answer, using pattern matching techniques. The document's knowledge representation is constructed in part by mining an electronic version of Roget's Thesaurus for relevant linguistic data. The method relies on a semantic representation that can handle unbounded unseen text without requiring extensive hand coding of knowledge structures in advance. In the first step of processing a text, many Natural Language Processing (NLP) systems employ a parser to extract the sentence grammar. And then comes the semantic dissection. The literature is replete with recommendations for parsers based on Context-Free Phrase Structure Grammar (CFPSG). While useful in some situations, CFPSG parsing is limited to only the simplest of play settings. This is because it's really challenging to make sense of the free, invisible text. After all, a system that would need to analyse every conceivable parse tree generated would take too much time, and the required collection of grammar rules is already quite enormous. This article describes a prototype system that attempted CFPSG parsing, but ultimately gave up and relied on "Chunking" to identify the clauses and phrases that would be processed further. By employing grammatical heuristics, "chunking" makes it possible to rapidly infer noun phrases and verb sentences from uninitiated text. So, we can stop worrying about parsing times that are impossible to achieve. Roget's Thesaurus (Roget, 1991) data extraction is time-consuming since Visual Basic for Applications code must scan around 500 pages of a Microsoft Word document for each word in a sentence. One of the system's major drawbacks is that it can take up to 10 minutes to look up synonyms for a 40-word statement, and it needs to be tweaked in order to use a database version of Roget's Thesaurus.
3. System Design:
There are several moving parts in the essay-grading system architecture. All the modules are organised into 3 distinct layers in the architecture (figure 2):

![Figure 2: Architecture Categories](image)

Internet-connected computers, laptops, smartphones, and other mobile devices make up the bulk of the user engine. The purpose of these gadgets is to provide users with intuitive interfaces that allow them to more simply and effectively interact with the platform and receive feedback on their essays. The gadgets connect to the API endpoint, where they query the server for the appropriate web page to display at the given URL. The requested web page will subsequently be sent back to your browser from these servers. The entire system is built as a Progressive Web Application (PWA), so it looks and feels much like a native app for iOS and Android. There are modules in the framework's back end that do things like classify the text and keep tabs on platform users. These are the modules (figure 3):

![Figure 3: Back-end Framework](image)

The central section of the English essay grader is called Text Evaluation. When this section is activated, the full essay is forwarded to the NLP Servers, where it is processed by a number of NLP algorithms and its resulting parameter values are returned. When a user enters in to our system, a session is created and kept alive until the user manually ends it. Similarly, this offers a single setting that allows it to save the session across all open browser tabs and windows. All of the NLP server's output is captured by the Text evaluation module, where it is then sorted into appropriate groups and the user’s session and database are refreshed. The main assessment engine that grades the essay and returns the parameter score is housed in NLP servers. There are two primary modules on the NLP servers.

- NLP Model Bucket
- No-SQL Database

The NLP Model Bucket class provides all the necessary capabilities and methodology to assess the essay in light of these criteria (figure 4).
An essay’s consistency in grammatical and spelling errors is checked by the grammar and spell check module. Sentence tokenization extracts sentences from the essay, and a module’s evaluation of them yields a score between 0 and 10 on this metric. To determine the accuracy rate, we first determine whether or not the sentence is a perfect sentence by the sophistication of the sentence’s composition is evaluated using the sentence complexity measure. Consider the two phrases "I know everything" and "No secret lies beyond my grasp" as an illustration. Sentence complexity is calculated and scored from 0 to 10 by this component. All the paragraphs in your essay should be written in the same English style. The third model considers whether the author maintains a consistent English dialect throughout the piece. A renowned writer in English will use a consistent accent when composing their phrases, making it easier for their readers to grasp what they're reading no matter where they are located. The reader will be thrown off if you use a mix of American English, British English, and Australian English in the same essay. We can achieve consistency via dictionary vector using the I-vector approach and Gaussian Mixture Model (GMM), and this method will work even with a small dictionary dataset. All parts of the essay will have the same weight in the final grade.

The skill level of the user is a criterion for this framework. A word's embedding, derived using Continuous bag-of-words (CBOW), is mapped to the bucket's appropriate word set based on the input word. There are five distinct containers, each corresponding to a distinct vocabulary requirement. Each of the five categories, labelled "buckets," contains a fuzzy vector value proportional to the level of difficulty of the words in that bucket. Word2vec, a word embedding system, calculates the cosine significance between the input corpus and a bag-of-words. The IBW (Intimacy Between Words) algorithm takes the input word and the entire set of words and uses the intimacy percentage that is the highest to determine which bucket the input word belongs in. The frequency of occurrences of a mapped word is used to determine the vector value (TF). After a word's frequency vector is calculated, we use t-distributed Stochastic Neighboring to optimise the vector value for each back iteration of the training process. Because of its superior efficiency compared to other POS tagging algorithms, perceptron-based POS tagging is used to tag each noun POS (parts of speech) word in the sentence during model training. The model then assigns a score to the sentence based on where each word falls inside its network of semantic neighbours. Credits for lexical resources are calculated similarly by averaging the scores for numerous sentences.

Model 5 automatically pulls out sentences from an application essay as it is submitted. After that, an algorithm that recognises and codes letter strings is used to tokenize the sentences. The neural network will be trained to decipher the coded phrases. Coded words are passed as inputs to the sentence syntax analysis module, which indexes the words before applying any processing. The meaning of the statement is determined via a three-layer hamming neural network. The model has been trained to identify the most relevant subject word in a given sentence. As a result, we may learn what each line is actually about. To determine if there is any connection between the two topics of the adjacent sentences, we employ collaborative filtering and task-based knowledge approaches. This allows us to determine the overall cohesiveness of the essay.

**System Implementation**

This paper focuses on the logistics of putting the system architecture into action. To accomplish this, Python 3.8 was employed. Due to its intricate make-up, the model is run in a separate setting from the one in which the training and consumption algorithms are implemented. Using the react library, we were able to convert our app...
Django is an advanced Python web framework that facilitates the creation of reliable and easily maintained websites rapidly. Django was created by seasoned programmers to streamline and simplify web development, allowing us to focus on creating your project without having to invent the wheel. It's open source, free, and well-supported by both a large and helpful user community and extensive online resources. Because of Django, our application is comprehensive, flexible, safe, scalable, maintainable, and portable. The English dictionary WordNet contains a huge amount of information. Synsets are collections of cognate words that have a meaning but are otherwise dissimilar, including nouns, verbs, adjectives, and adverbs. Relationships between concepts and between words are used to connect synsets. At first glance, WordNet looks like a thesaurus since it organises words according to their meanings. There are, however, notable differences. The first thing to know is that WordNet connects not just letter strings but also meanings of words. Therefore, the semantic ambiguity between words that are physically close to one another in the network is resolved. In addition, WordNet provides labels for the semantic relations between words, whereas a thesaurus just groups words together based on their semantic similarity.

**Experimental Results and Discussion**

The results of our evaluations of our NLP models are discussed, along with the effects of the implementation and assessment metrics we utilised. The created models show an accuracy of 89.1%. Models of neural networks are built for each of the constraints used to judge essays in standardised tests of the English language. Each individual model dataset serves as input, therefore our models will function based on the aforementioned five parameters. Every single model achieves an efficiency of around 85%. When comparing the accuracy to prior works, there is a small variation. The above chart displays the varying degrees of accuracy achieved by various algorithms. The grammatical errors are generated by the Multinomial HMM (Hidden Markov Model), a probability-based classification model. Dictionary The quality of our n-gram model is entirely reliant on the data we used to train it. Data used to determine proper spelling is gathered from the aforementioned WordNet database. The Gaussian Mixture Model is a type of probabilistic clustering technique that assigns each datapoint a probability in relation to a random centroid. It's a way to categorise how advanced the vocabulary is that was used to construct the statement. Because of its generative nature, Bidirectional Neural Network can be applied to evolving data. The intricacy of the sentence varies from writing style to writing style, which is why we require dynamic data points. The scope and effectiveness of the project can be increased by making it available in multiple languages. As more information becomes available, we will be able to fully automate the grading of essays. Providing support for voice commands and evaluating speech also has significant implications for NLP.

**Conclusion**

It takes a lot of time and effort for evaluators to go over and correct the answers. The testing facilities should also set aside some space for grading the submitted responses. All of this takes effort and additional resources to do. Our essay grader makes it easy for testing centres to administer these exams. Using this programme, grading and correcting responses is no longer a time-consuming manual process. We intend to create this programme so that institutions specialising in evaluating candidates' command of the English language, such as the IDP British Council, can use it to assess the candidate's command of the language as manifested in his written work (grammar, spelling, sentence structure, vocabulary, etc.). We took into account the limitations that assessors have when developing the programme. Neural networks play a pivotal role in the system's operation, and we've built custom networks to handle each type of operation. They're engaging in all four major modes of communication. In the future, we will be able to grade essays written in a variety of languages, including Tamil, Telugu, Malayalam, etc., all of which will be useful in the classroom. The candidate's performance in the speaking segment is currently assessed in real-time, as they speak. The candidate will be asked several questions
by the Evaluator, and they must respond. In the near future, this can be done mechanically with the use of speech processing. The candidate’s eloquence can be uncovered by recording the audio and applying ML algorithms to it. As a result, the product can benefit from a variety of ongoing and future works.

References

obstructive Lung Disease Using Outlier Detection with Decision Tree Algorithm”, Pre-print, 2022 (https://doi.org/10.21203/rs.3.rs-2072803/v2)


111. Von Louie, A. S., Lumapenet, H. T., & Mamburao Jr, R. S. Leadership and Fiscal Management Skills Among Public Elementary School Principals In Southern Philippines


