

**A NOVEL ALGORITHM FOR AUTOMATIC ESSAY GRADING USING NATURAL LANGUAGE PROCESSING TECHNIQUES**

**PROJECT REPORT**

***Submitted by***

|  |  |
| --- | --- |
| **VISHWAK SENAN G** | **211716104110** |
| **PRANAV KUMAR K** | **211716104075** |
| **SUNDARAMOORTHY T** | **211716104101** |

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**RAJALAKSHMI INSTITUTE OF TECHNOLOGY**

**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL 2020**

**ANNA UNIVERSITY: CHENNAI 600 025**

**BONAFIDE CERTIFICATE**

Certified that this project report **“A NOVEL ALGORITHM FOR AUTOMATIC ESSAY GRADING USING NATURAL LANGUAGE PROCESSING TECHNIQUES****”** is the bonafide work of **“VISHWAK SENAN G(211716104110), PRANAV KUMAR K(211716104075)** and **SUNDARAMOORTHY T(211716104101)”,** who carried out the project work under my supervision.

|  |  |
| --- | --- |
| **SIGNATURE:** | **SIGNATURE:** |
| **Dr. D. C. JOY WINNIE WISE M.E., Ph.D.,** | **Dr. D. JAYASHREE M.S., Ph.D.,** |
| **HEAD OF THE DEPARTMENT,** | **PROFESSOR,** |
| Dept. of Computer Science and Engg., | Dept. of Computer Science and Engg., |
| Rajalakshmi Institute of Technology,  Kuthambakkam Post,  Chennai - 600 124 | Rajalakshmi Institute of Technology,  Kuthambakkam Post,  Chennai - 600 124 |
|  |  |
|  |  |

**CERTIFICATE OF EVALUATION**

|  |  |
| --- | --- |
| **College Name** | **2117 - Rajalakshmi Institute of Technology** |
| **Branch & Semester** | **Computer Science and Engineering & 8th Semester** |
| **Subject** | **CS6811 - PROJECT WORK** |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO.** | **NAME OF THE STUDENTS** | **TITLE OF THE PROJECT** | **NAME OF THE SUPERVISOR WITH DESIGNATION** |
| 1 | VISHWAK SENAN G | A NOVEL ALGORITHM FOR AUTOMATIC ESSAY GRADING USING NATURAL LANGUAGE PROCESSING TECHNIQUES | Dr. D. JAYASHREE  Professor  Dept. of Computer Science and Engineering |
| 2  3 | (211716104110)  PRANAV KUMAR K  (211716104075)  SUNDARAMOORTHY T  (211716104101) |

The report of this project work submitted by the above students in partial fulfillment for the award of Bachelor of Engineering degree in COMPUTER SCIENCE AND ENGINEERING under Anna University was evaluated and confirmed to be the reports about the work done by the above students.

The University viva-voce is held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

We express our sincere gratitude to our honorable Chairperson **Dr.(Mrs.)THANGAM MEGANATHAN, M.A., M.Phil., Ph.D.,** and Chairman **Thiru. S.MEGANATHAN, B.E., F.I.E.,** for their constant encouragement to do this project and also during the entire course period.

We thank our Principal **Dr. M. VELAN, M.E., Ph.D.,** and our Head of the department **Dr. D. C. JOY WINNIE WISE, M.E., Ph.D.,** for their valuable suggestions and guidance for the development and completion of this project.

Words fail to express our gratitude to our Project coordinators **Dr. M.GEETHA, M.E.,Ph.D., Mr. MICHEAL MARIA VISWASAM, M.E.,** and Internal guide **Dr. D. JAYASHREE , M.S.,Ph.D.,** who took special interest on our project and gave their consistent support and guidance during all stages of this project.

Finally, we thank all the teaching and non-teaching faculty members of the **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING** of our college who helped us to complete this project.

Above all we thank our parents and family members for their constant support and encouragements for completing this project.

**ABSTRACT**

Evaluation of an English essay is one of the important and complex tasks which is done manually by skilled and efficient professors and faculties till date. The growth of science and technologies enables to automatic evaluation of an English essay using natural language processing (NLP) techniques. The intelligent system - built upon NLP multiple neural network model - gives out generic evaluation and the topic/question correlation for any given English essay. The core innovation lies in evaluation of essay in international standard where the grading standard can be used in any international grading systems like Gradual Record Examination(GRE), International English Language Testing System (IELTS), etc. The algorithm allows the users to test their skill from primitive to complex factors involved while grading the English essay.

**Index Terms** – Automatic evaluation, Natural Language Processing, generic evaluation, topic/question correlation.

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
| **CHAPTER NO** | **TITLE** | **PAGE NUMBER** |
|  | **ABSTRACT** | **v** |
|  | **LIST OF FIGURES** |  |
|  | **LIST OF TABLES** |  |
|  | **LIST OF ABBREVIATION** |  |
| **1** | **Introduction** | **1** |
| **1.1** | **Multinomial Hidden Markov Model** | **1** |
| **1.2** | **Bidirectional Neural Network** | **2** |
| **1.3** | **Long-Term Short-Term Memory** | **2** |
| **1.4** | **Gaussian Mixture Model** | **2** |
| **1.5** | **Continuous bag of words** | **3** |
| **1.6** | **t-Distributed Stochastic Neighbor** | **3** |
| **1.7** | **Objective** | **3** |
| **1.8** | **Organization** | **4** |
| **2** | **Literature Survey** | **5** |
| **3** | **System Design** | **14** |
| **3.1** | **User Engine** | **14** |
| **3.2** | **Back-end Framework** | **15** |
| **3.2.1** | **Text Evaluation** | **15** |
| **3.2.2** | **Maintaining User Session** | **15** |
| **3.3** | **NLP Servers** | **16** |
| **3.3.1** | **NLP Model Bucker** | **16** |
| **3.3.1.1** | **Grammar and Spell Check (Model 1)** | **16** |
| **3.3.1.2** | **Sentence Complexity (Model 2)** | **16** |
| **3.3.1.3** | **Style Continuity (Model 3)** | **17** |
| **3.3.1.4** | **Usage of Lexical Resources (Model 4)** | **17** |
| **3.3.1.5** | **Coherence and Cohesion (Model 5)** | **18** |
| **4** | **System Implementation** | **19** |
| **4.1** | **Introduction to Python 3.8** | **19** |
| **4.2** | **Introduction to Django 3.0** | **20** |
| **4.3** | **Introduction to WordNet** | **20** |
| **4.4** | **Model Algorithm for Training and Consuming** | **20** |
| **4.4.1** | **Model 1 Training and Consumption Algorithm** | **21** |
| **4.4.2** | **Model 2 Training and Consumption Algorithm** | **22** |
| **4.4.3** | **Model 3 Training and Consumption Algorithm** | **24** |
| **4.4.4** | **Model 4 Training and Consumption Algorithm** | **25** |
| **4.4.5** | **Model 5 Training and Consumption Algorithm** | **26** |
| **5** | **Experimental Results and Discussion** | **27** |
| **5.1** | **Result and Test Accuracy** | **27** |
| **5.2** | **Discussion** | **28** |
| **5.3** | **Future Works** | **28** |
| **6** | **Conclusion** | **29** |
| **APPENDIX** | | |
|  | **Journal Paper** | **30** |
|  | **Conference Certificate** | **38** |
|  | **Source Code** | **41** |
|  | **Output Screenshots** |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **TITLE** | **PAGE NO.** |
|  |  |  |
| 3.1  4.1 | Overview of SEUC  SEUC | 28  33 |
| 4.2 | SEUC Processing | 37 |
| 4.3 | Create Search History | 38 |
| 4.4 | Query Analysis | 38 |
| 4.5 | View Result | 39 |
| 4.6 | Resource needed | 40 |
| 4.7 | Steps of execution | 41 |
| 4.8 | Communication establishment | 42 |
| 4.9 | Actor specification | 43 |
| 4.10 | Component specification | 43 |
| 4.11 | Steps in flow | 44 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| SEUC | Search Engines based on User preference Click through data |
| CTR | Click Through Rate |
| OMF | Ontology-based Multi Facet |
| OPD | Open Directory Project |
| PCAT | Personalization Categorization System |
| LIST | List Interface System |
| CAT | Categorization System |
| ANN | Artificial Neural Networks |
| WWW | World Wide Web |
| UCI | User Conceptual Index |
| USB | User’s Search Behavior |
| GPS | Global Positioning System |
| LBRM | Location Based Ranking Method |
| UML | Unified Modeling Language |

**1. Introduction**

International examinations like GRE (Graduate Record Examination), IELTS (International English Language Testing Systems), etc., is gaining popularity day by day as this examination’s results are considered as criteria for various universities and companies. GRE examination is a standardized test which enacts as the fundamentals for admissions in International Universities. It is owned and administered by ETS (Educational Testing Services) and according to them, GRE aims for evaluation measures in verbal reasoning, quantitative reasoning, analytical writing and critical thinking skills. Therefore, the number of students who are taking the exams increases day by day and there is a huge time buffer to evaluate their English essays and publish the results. As for IEFTS, it is international standardised test for English language proficiencies, which is considered as criteria in European countries, Canada and in Australia. Due the large number of candidates, there exists inconsistency in providing the evaluation service, thus to overcome this latency, a proposed model is designed and implemented which to automate the evaluation process and provide it as API (Application Program Interface) service.

Organisation which is responsible for GRE and IELTS, etc., can have major focus on marketing strategies meanwhile, the candidate’s exam evaluation process is entertained by novel model called as essay grader which implements seven Natural Language Processing (NLP) strategies, each yield a probabilistic outcome and these analytical result-set is combined into five parametric models consists of marking scheme scaled from 0 to 10 points. These parametric models are labelled for user’s understanding and clarity over the grading system and this pattern of mapping the NLP servers to user understanding models increases the User Experience (UE) in quality management.

**1.1 Multinomial Hidden Markov Model**

Multinomial HMM is the generalization of the multiple Hidden Markov model, HMM is a statistical Markov model in which the system being modelled is assumed to be Markov process with hidden states. Its state is not directly visible but the output dependent on the state is visible. Each state has a probability distribution over possible result-set. HMM provide conceptual toolkit for building complex models with adequate amount of unobservable states. They are at the heart of a diverge range of programs, including gene-mapping, profile searches and regulatory site identification.

**1.2 Bidirectional Neural Network**

Major two opposite neural network are connected together, where one is dependent with other neural network to yield the output result, such recurrent neural network is are termed as Bidirectional Neural Network. The two opposite neurons are formed from one RNN, one for the forward states and the other for the backward states. BNN is trained using similar algorithms to RNN because the two directional neurons do not have any interactions. For training in forward pass, first the forward states and backward states are passed first, then the output neurons are passed, and for the backward pass, first the output neurons are passed first then the forward and backward states are passed. After the completion of two process, forward and backward pass, the weights are updated.

**1.3 Long-Term Short-Term Memory (LSTMs)**

Since the bidirectional neural network has higher chances of hit ratio in obstacles like vanishing error problem and time delays, thus LSTMs are implemented since they can learn to bridge minimal time lags in excess of 1000 discrete time steps by enforcing constant error flow through “constant error carrousels” (CECs) within special units, called cells.

**1.4 Gaussian Mixture Model**

In the aspect of machine learning, there are two field regions and they are , supervised and unsupervised learning. Clustering is the example of unsupervised learning which consists of two different approaches, one is hard clustering which can implemented using k-means algorithm and on the other hand, its soft clustering, which is achieved with gaussian mixture model. This model is based on normal distribution, from which, it categories the datapoints and clusters them based on the mean and covariance, where mean specifies the centroid of the cluster and covariance indicates the width of datapoints.

**1.5 Continuous bag of words**

In word2vector model, each word is embedded with a vector without any relation with multiple word-sets whereas the continuous bag of words achieves multiple input corpus to target word. For example, we could use “virus” and “pandemic” as context words for “corona” as the target word.  We can model this CBOW architecture now as a deep learning classification model such that we take in the context words as our input and try to predict the target word.

**1.6 t-Distributed Stochastic Neighbor**

t-Distributed Stochastic Neighbour Embedding (t-SNE) is an unsupervised, non-linear technique primarily used for data exploration and visualizing high-dimensional data. This provides with a visual stimulation of data arranged in high dimensional space. The t-SNE algorithm calculates a similarity measure between pairs of instances in the high dimensional space and in the low dimensional space. It then tries to optimize these two similarity measures using a cost function.

**1.7 Objective**

Evaluation of an English essay is one of the important and complex tasks which is done manually by skilled and efficient professors and faculties till date. With the growth of science and technology, we can simply automate the evaluation task by using state of the art natural language processing techniques. To reduce the stress on the organization who are hosting these examinations and students to practice their writing skills, our project aims to evaluate the English essay so that the organization can focus their work in other aspects of examinations and students can practice at free will. The project acts as an intelligent system - built upon multiple neural network model - gives out generic evaluation for any given English essay.

**1.8 Organization**

The rest of the report is organized as follows. Chapter 2 includes a detailed literature survey related to the work. In Chapter 3, system design is explained. Chapter 4 comprises of implementation and the novel algorithm to evaluate the essay. Chapter 5 provides the results of the work and Chapter 6 consists of conclusion of the project.

**2. LITERATURE SURVEY**

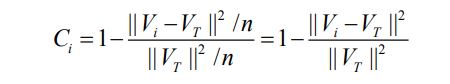
Many previous works related to this idea has already been proposed. Many authors have published many research papers and articles for the betterment in advancing this idea. Some of the works are discussed below.

J. A. O'Sullivan, K. Mark and M. I. Miller published a research paper titled “Markov random fields on graphs for natural languages”. In this paper, they have proposed a way to find the probability of the occurrence of the next word given the n-1 words. This model will be very much useful when it comes to finding the next suitable word when the user is typing his essay or otherwise for auto spell check. The model works with the concept of probability. Suppose if a sentence is being typed by a user, this model will help suggesting words which match that sentence. A huge set of sentences are given as input to the model so it can train in a correct way and make predictions. The model “n-gram” means that it considers the first n words in a sentence which is given as training data and makes predictions to obtain meaningful sentences as output. This model can be applied in our idea which helps to logically connect sentences based on the dataset used for training. Apart from these, it has a lot of applications such as the auto correct feature in various messaging applications, recommending words based on the sentence which was typed in message applications, etc. This saves time and we won’t be in situation where we are required to type each and every word again and again. We can simply click on the word from the suggestions which best suits our sentence. Using this way, the sentences can be evaluated and the essay too. Therefore, this is a probabilistic model with a good accuracy if trained from a sophisticated dataset.

S. Li, Q. Wang, X. Liu and J. Chen had done a research in how to optimize the training time of a neural network. Basically, they have developed a low-cost high memory device. Inside of running the neural networks by renting GPUs and local servers, we can make use of these low-cost hardware devices which will help in efficiently developing neural networks. S. Li, Q. Wang, X. Liu and J. Chen research shows that the neural network when trained in this low-cost hardware is more optimized. The low-cost LSTM is achieved by applying stochastic function rather than hyperbolic activation function. The complex arithmetic operations involved in the neural networks will get converted to gates. Gates are the building blocks in a LSTM and they will know the most important data from the sequence which needs to be kept and the ones which needs to be thrown away. As it collects important data, it forms a chain of sequences and passes it. LSTM is mainly used for word generation. The main aspect in the working of LSTM is that it looks for the most important keywords from the sentences. Using these keywords, it makes predictions to find the appropriate word. For example, if want to buy a mobile phone from an ecommerce website, I will be looking the reviews which are posted. I won’t remember the whole content, instead the words which describe the product. In this same way, the LSTM also works. The Gates in the LSTM uses sigmoid function. Since it uses sigmoid function, its values range from 0 to 1 and not from -1 to 1 which is in the case of tanh function. As a result, the sigmoid function can help the neural network to learn which data is important and which is not. There are three different gates that regulate information flow in an LSTM cell. They are forget gate, input gate, and output gate. So, this is how the LSTM works and helps improving the neural networks in terms of efficiency.

Ma, Q., Uchimoto, K., Murata, M., & Isahara, H. had proposed a solution to find the parts of speech in the sentences. This was used exclusively in linguistics in which we can find the words in a particular text and classify it under the different types of parts of speech based on its usage and definition. In order to find the parts of speech, we will be using a 3-layer perceptron layer with n-inputs, n representing the total number of words in the dataset. While the computational cost of the training the network is drastically higher than the n-gram model but according the previously conducted experiment, the accuracy was 99.4% without over- fitting into the data. The accuracy was obtained by using the elastic hidden perceptron layer. The tagging process takes place at given word level using a variable length of a context. The training process takes place with the help of an error back propagation algorithm. To make the same set of connection weights of the elastic neuro tagger available as much as possible even when the length of input is reduced in tagging, a new training method is adopted. The new training method aims to obtain a set of connection weights whose subsets corresponding to the neuro taggers with short inputs are as same as those that are obtained directly by training these taggers. For this purpose, in the training phase the elastic neuro tagger is regarded as a perceptron that has gradually grown from a small one and the training is therefore performed step by step from small ones to large ones. In detail, the smallest perceptron is trained at first. After training, a new perceptron is then formed through incremental growth and is trained again. Such a gradual growth and training process is repeated until the perceptron with the largest (I, r) is formed and trained. So, after the training process gets complete, we can see that the words from the dataset will be tested and the words will be classified based on the type of parts of speech.

B. Roy and H. Cheung proposed a method to have multiple activation functions in one single RNN. RNN stands for Recurrent Neural Network and is a form of neural network which works in bidirectional i.e., both the forward as well as backward. RNN is exclusively used in NLP for language modelling and text generation. Clustering algorithm is implemented to categorize the word component to particular to cluster, there exists two major types of cluster, one is soft cluster and other is hard cluster. Hard clusters are the concept of binary categorization, a component can exist in only one cluster. K-means is the algorithm utilized for clustering, where components are fed into nods as training dataset in an iterative methodology. K-means algorithm can be exhaustive and some components may not be segregated into one cluster and its left out. The process of k-means is to determine the distance between the centroid and the datapoints. By the use of keywords combination and Word2vec to improve topic relevance and semantic expression. Every document will have a set of keywords and these keywords will have an unique vector. Correlation between the words and the topics needs to be found out and hence we must find out a reference word after which the Word2Vec method is applied. Euclidean distance between keyword and the topic is found out and a variable called “Relevant weight” is initialized and found out using the formula given below,



The best centroid is where it has a mean of zero and standard deviation of one. For determining the centroid points, k-means follows Expectation minimization (EM), this value assign a centroid point where the density of data points is high. In initial iterations, random set of data points are subjected to undergo Euclidean distance calculation and the smallest distance obtained between any two datapoints, from them one of the datapoint is assigned as centroid and once the centroid is formed, in every iterations, its coordination is altered as to include the higher number of datapoints based on the RSS (residual sum of squares) value. Experimental results show that the number of each cluster in weighted K-means is striking whereas the normal K-means is insignificant. Also weighted K-means produces more advanced semantic information and relevance of topic to assist cluster division.

N. Rogovschi, J. Kitazono, N. Grozavu, T. Omori and S. Ozawa introduced a new clustering algorithm which is based on the topology and its used to cluster high dimensional datasets on the basis of t-SNE (Stochastic Neighbor Embedding). The Main advantage of this method is to allow the dataset to run in low dimensionality spaces and hence it makes it easy to use large datasets. They have focused on models that are based on spectral clustering and topological unsupervised learning i.e. the t-SNE (Stochastic Neighbor Embedding) in order to reduce the data dimensionality of the data and to take advantage from the topological preservation of information. At the end of learning the t-SNE, the “similar” data will be separated from the dissimilar data. As expected, this information is easier to manipulate than the original data points. Considering two parameters involving 2d graphical representation of multiple datapoints, these datapoints are the mapped and clustered with soft or hard clustering using the algorithms k- mean or gaussian mixture model, thus these clusters are needed to be evaluated for relevance or degree of similarity by using normal distribution to scale the datapoint’s value against the distribution curve. The datapoints with highest distribution and with the lowest high dimensional Euclidean distance, gets assigned with vector value obtained from conditional probability distribution that represent similarities. T-Sne algorithm forms a matrix based on the distance between dissimilar and similar datapoints representing the maximum similarity value is near the diagonal of the matrix, hence this matrix called diagonal matrix , since highest similarity is one, hence the similarity between a word to itself is neglected and similarity between word components are estimated n plotted in the matrix. Now, this algorithm has two parameters by which it can cluster the datapoints in low dimensional graph retaining the cluster similarity, word covariance and relevancy level. Similarity factor is implemented in low dimensional graph to cluster the datapoints at a cost of multiple iterations. Simply the algorithm gets one matrix in each iteration, thus the iteration is kept on progressed until it reaches the matrix value formed with high dimensional graph. In low dimensional graph, datapoints are mapped against t-distribution curve graph to obtain similarity between the datapoints.

H. W. Lam, T. Dillon and E. Chang had proposed a model which will help in identifying the genre of a particular essay. Particularly this model will be categorizing in four genres namely Narrative, Persuasive, Descriptive and Expository. In the case of narrative, the prominent feature is the frequent use of personal pronouns such as “I”, “She”, “He”, etc. In persuasive or argumentative writing, the purpose is to convince the reader of the author’s opinion or belief. There is usually a clear distinction of between two opposing views, wherein the author tries to support his or her point of view using facts, arguments or supporting statements from others with highly valued opinions. In Descriptive writing, the main objective is to be able to allow the reader to visualize the setting which the author is trying to portray. This can be in relation to a location, person or even the ambience of a place. To expound, in other words, to explain. Thus, the main objective of this genre of writing is to inform or instruct the reader of a certain topic or theme. This form of writing is most common in research or journal articles and guidebooks. This process is being done with the help of Named Entity Recognition, Part of Speech tagging and Sentence Parsing. This approach uses Rubric based evaluation technique which is a hard-coded rule-based evaluation. The NAPLAN (National Assessment Program - Literacy and Numeracy) marking rubric is primarily used to grade essays of a narrative genre. The First step in identifying the number of characters in an essay, a Stanford NER Tool is used. Primarily, in a narrative essay, the story is told from the perspective of a single main character, either in first or third person. The rubric is made up of ten criteria, each assessing a different characteristic of a student’s essay:

• Audience – writer’s capacity to orient, engage and affect the reader.

• Text Structure – organization of narrative features in an appropriate and effective structure.

• Ideas – creation, selection and crafting of ideas.

• Character and Setting – portrayal of character and/or development of a sense of place, time and atmosphere.

• Vocabulary – the range and precision of language choices.

• Cohesion – the control of multiple threads and relationships.

• Paragraphing – segmenting of text into paragraphs that assist in reading.

• Sentence Structure – production of grammatically correct, structurally sounds and meaningful sentences.

• Punctuation – used of correct and appropriate punctuation.

• Spelling – accuracy of spelling and difficulty of words used.

Once an essay is classified as a narrative, the NAPLAN marking rubric will be applied to attain the final score. The main limitation is that it is very topic focused and hard-coded. Although the methods allow for specific classification of narrative essays, the processes of genre classification are not yet compiled into a single component, which affects the overall efficiency of the proposed system.

The E-rater is one system that considers the linguistic features of the text. The system incorporates the use of several Natural Language Processing (NLP) techniques, picking out features from a set of sample essays thus providing the base of the scoring model. E-rater automatically assumes that the traits of a good essay would not stand too far apart from a similarly well written essay and likewise for poor essays. In the current practice, e‐rater scores are generated by a linear combination of a set of high‐level features computed for each essay with weights determined by regressing human ratings on the features. These features are also called macro features. Most of these macro features are composed of sets of lower‐level features called microfeatures that are combined to produce the microfeature values. All of these macro features and microfeatures are extracted by using NLP. 10 macro features are most commonly used to predict human scores. These 10 macro features are organization, development, grammar, usage, mechanics, style, word length, word choice, collocation and preposition, and sentence variety. In addition to these 10 macro features, two prompt‐specific vocabulary usage features are used to predict human scores when the scoring model is custom built for each prompt. The E-rater V.2 scoring system remains relatively unchanged but requires significantly less features by condensing them into a smaller, more meaningful set which include Style Measures and Lexical Complexity.

**Evaluation criteria**:

Once the automated (e-rater) scores for all essays have been calculated, ETS uses certain evaluation criteria to assess the quality of the models. There are guidelines for performance that are applied to the independent evaluation sample used to validate the scoring models. The results on the evaluation sample independent from the model-building sample represent a more generalizable measure of performance that would be more consistent with what would be observed on future data. The criteria are as follows:

**Construct evaluation**:

Automated scoring capabilities, in general, are designed with certain assumptions and limitations regarding the tasks they will score. Therefore, the initial step in any prospective use of automated scoring is the evaluation of fit between the goals and design of the assessment (or other use of automated scoring) and the design of the capability itself. The process includes a comparison of the construct of interest with that represented by the capability, review of task design, review of scoring rubric, review of human scoring rules, review of score reporting goals, and review of claims and disclosures. Other improvements made to the system allows for the creation of a single standardized scoring model. Despite the advancements from its predecessor, the same issues remain; the feature module ‘Lexical Complexity’ considers word-based characteristics, key word frequency and word length but does not consider fully the context of which the words are used. The feature can thus be fooled by ‘nonsense text which contain a high number of complex words but fail to add meaning to the passage.

Robert Williams and Heinz Dreher had proposed a system called Markit which is an automatic Essay grading system. The working of Markit starts when a student essay is processed using a combination of NLP (Natural Language Processing) techniques to build the corresponding propriety knowledge representation. Pattern matching techniques are then employed to ascertain the proportion of the model answer knowledge that is present in the student answer, and a grade assigned accordingly. An electronic version of Roget’s Thesaurus is used to extract lexical information for the building of the document knowledge representation. The approach used has a need for a semantic representation that does not need substantial hand coding of knowledge structures prior to use, and that can deal with unlimited unseen text. Many Natural Language Processing (NLP) systems use some kind of a parser to initially extract the syntax of sentences in a document as an initial step prior to further processing. Semantic analysis then follows. The use of Context Free Phrase Structure Grammar (CFPSG) parsers is commonly suggested in the literature. However, CFPSG parsing cannot be used in all but simple toy domains. The reason for this is that free unseen text is very hard to parse, because the set of grammar rules required is very large, and the time taken to evaluate every possible parse tree generated is too great for a practical system. So, while CFPSG parsing has been tried with the prototype system described in this article, it has been abandoned in favor of using “Chunking” to determine the phrases and clauses used for further processing. “Chunking” enables one to use grammar heuristics to derive noun phrases and verb clauses very quickly from unseen text. The problem of unrealistic parsing time is thus eliminated. The extraction of information from Roget’s Thesaurus (Roget, 1991) is slow, due to the fact that approximately 500 pages of a Microsoft Word document have to be scanned for each word in a sentence, using Visual Basic for Applications code. This process can take up to about 10 minutes for a 40-word sentence and clearly needs to be modified to access a database version of Roget’s Thesaurus as this is one of the greatest disadvantages of the system.

**3. System Design:**

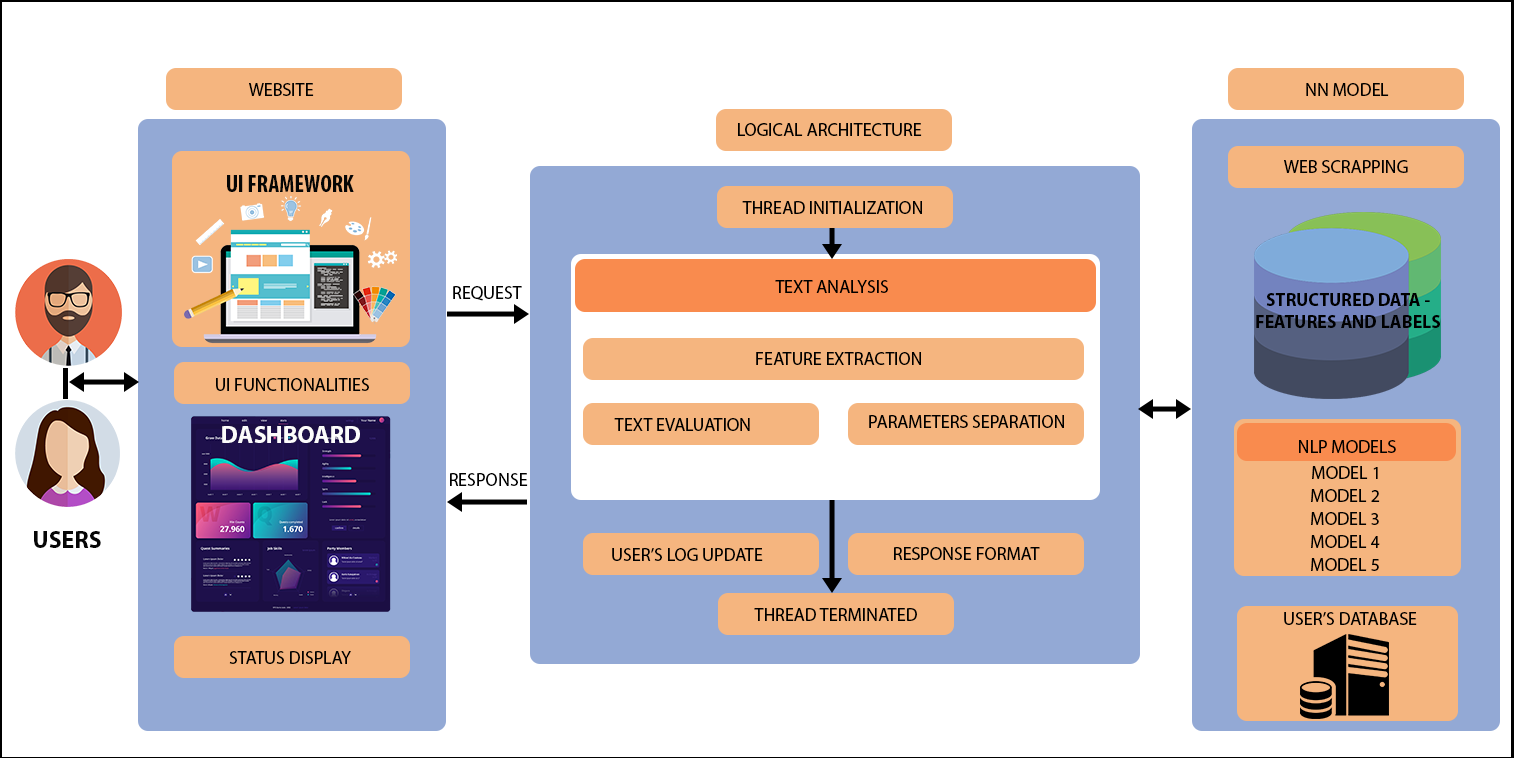
The system architecture of the essay grading revolves around various modules. The architecture categories all the modules into 3 tiers:

1. User Engine.

2. Back end framework.

3. NLP Servers.

The system architecture is depicted in *Figure I (System Architecture)*.



*Figure 1: System Architecture*

**3.1 User Engine:**

User engine mainly consists of devices like Personal Computers (PC), laptops, mobile phones, etc. The role of these devices is to render the user interactive and user-friendly interfaces for them to interact with the platform and have a better place to evaluate their essay consistently and easily.

The devices hit the API endpoint and asks the server for the webpage that should be rendered for that URL. These servers will then return the corresponding web page to load. Since the whole platform is designed to be a Progressive Web Application (PWA), it gives the mobile phone users their native feel in both android and iOS devices.

**3.2 Back-end Framework**

The Back-end Framework consists of modules which categorizes the text and keeps the track of the users who are using the platform. The modules are

1. Text Evaluation.

2. Maintaining User Session.

3. Parameter Separation.

**3.2.1 Text Evaluation**

Text Evaluation is the main module which evaluates the entire English essay. This module sends the entire essay into the NLP Servers which then various NLP algorithm applied onto the essay and returns the values for its corresponding parameters.

**3.2.2 Maintaining User Session**

When the user login into our platform, it keeps maintains a user session till the user logs out. This also has single in on option so that it keeps the session maintained across browser’s tabs and windows.

**3.2.3 Parameter Separations**

The Text evaluation module captures all the output from the NLP server and categorizes the parameters into its corresponding categories and update the user session and in the database.

**3.3 NLP Servers**

NLP Servers contains the main evaluation engine which evaluates the essay and return the parameters score. The NLP servers contains 2 main modules

1. NLP Model Bucket.

2. No-SQL Database.

**3.3.1 NLP Model Bucket**

NLP Model Bucket is a class which contains all the function and methodology to evaluate the essay according to these parameters.

1. Grammar and Spell Check. (Model 1)

2. Sentence Complexity. (Model 2)

3. Style Continuity. (Model 3)

4. Usage Lexical Resources. (Model 4)

5. Coherence and Cohesion. (Model 5)

**3.3.1.1 Grammar and Spell Check (Model 1)**

Grammar and Spell Check module checks for inconsistent grammatical errors and spelling errors in the essay. From the essay, the sentences are obtained by sentence tokenization and the module evaluates the score for this parameter in the range of 0 – 10. When the sentence is classified as a perfect sentence or not, we calculate the percentage of correctness by

**3.3.1.2 Sentence Complexity (Model 2)**

Sentence complexity checks for the construction of the sentence in a sophisticated way. For instance, take the example of 2 sentences – “I know everything.” and “No secret lies beyond my grasps.”. The sentences almost mean the same, but the 2nd sentence’s sophistication level is on much higher level than the 1st sentence. This module also evaluates the sophistication level of the sentence and returns the score in the range of 0 – 10.

**3.3.1.3 Style Continuity (Model 3)**

Generally, an essay would consist of various paragraphs and the style in which each paragraph is constructed should follow the same English style. Model 3 refers to as whether the user continuous the same English accent throughout the essay. A well-established English literature person will formulate sentences in particular accent so that it is uniform for the people reading the sentence can understand easily. Combining various English accents like American English, British English, Australian English in a single essay will create confusions in readers mind. Using I-vector approach and Gaussian Mixture Model (GMM), we can get the consistency by dictionary vector which will even work for a small dictionary dataset. The marks are calculated in such a way that the same consistency is maintained throughout the essay. The score is calculated as

The score is evaluated in the range from 0 – 10 in this parameter also.

**3.3.1.4 Usage of Lexical Resources (Model 4)**

This model categorizes the sophistication level of the user.  Based on the input word provided, the word is mapped to its relevant word-set available in bucket and these word-set are word embedding obtained by Continuous bag-of-words (CBOW) [5]. Total of five buckets are utilized and each of them have a certain level of word standard. From bucket one to five, the word complexity increases and each bucket consists of fuzzy vector value based on the complexity of the word-set. Using word2vec word embedding systems, cosine relevance between input corpus and bag-of-words is determined and on the basis of IBW (Intimacy Between Words), the highest intimacy fraction obtained from input word and word-set determines to the bucket input word belongs, and mapped word is assigned with the vector value using term frequency(TF). When the frequency vector of the word is determined, then t-distributed Stochastic Neighboring, the vector distribution gives optimizes the vector value for each back while we train our model. When the model is trained, the sentence is tokenized then the noun POS (parts of speech) tagged word by perceptron-based POS tagging due to its high efficiency compared to another POS tagging algorithm. Then the model classifies the word to its nearest neighbor bucket and the evaluation is assigned for the sentence. Likewise, multiple sentences have their corresponding score and the scores are averaged out to get the final lexical resource credits.

**3.3.1.5 Coherence and Cohesion (Model 5)**

In model 5, sentences are extracted as the candidate submits the essay. Then sentences are tokenized by letter string recognition and coding algorithm. The coded words will be given as input to the neural network for recognizing the words. After passing as inputs, the coded words are transferred to the sentence syntax analysis module where it indexes the words before they are being processed. A 3-layer hamming neural network is being used to recognize the meaning of the sentence. The model is trained in such a way that it can recognize the most related subject word for a sentence. Therefore, we can get the actual subject/topic of each sentence. Using Task based knowledge and collaborative filtering techniques used to find relevance between the subjects of the consecutive sentence. In this way we can identify the cohesion and cohesion of the entire essay.

**4. System Implementation**

In this chapter, we discuss the implementation of the system design. The programming language that was used is Python 3.8. Since the model has a complex structure, the algorithm to train the models and to consume the model are executed in different environment. We use react library to make our application into a PWA. The web framework used is Django 3.0. Django provides a RESTful API to integrate the front-end components with the python’s classes and functions.

**4.1 Introduction to Python 3.8**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed. Some of the packages we used in the projects are

1. tensorflow.

2. pandas.

3. language\_check.

4. nltk.

5. spaCy.

**4.2 Introduction to Django 3.0**

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so we can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support. Django allows our application to be complete, versatile, secure, scalable, maintainable and portable.

**4.3 Introduction to WordNet**

WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. WordNet superficially resembles a thesaurus, in that it groups words together based on their meanings. However, there are some important distinctions. First, WordNet interlinks not just word forms—strings of letters—but specific senses of words. As a result, words that are found in close proximity to one another in the network are semantically disambiguated. Second, WordNet labels the semantic relations among words, whereas the groupings of words in a thesaurus does not follow any explicit pattern other than meaning similarity.

**4.4. Model Algorithm for training and consuming**

In this part of the chapter, we discuss the databases used, algorithm used to train the 5 models and its consumption of the model in the project.

**4.4.1 Model 1 Training and Consumption algorithm**

**Training Algorithm:**

*Step 1:* Download the word collection database from WordNet.

*Step 2:* Separate the nouns, verbs, pronouns, proverb, adjective, adverb into separate text file from the downloaded database.

*Step 3:* Load it into the python class *train\_model* using pandas and allocate corresponding label coming from corresponding text file.

*Step 4:* Separate the labelled database into training and testing database in 7:3 ratio.

*Step 5:* Apply the feature extraction Bag of Words model on the dataset.

*Step 6:* Save the trained model

**Consuming Algorithm:**

*Step 1:* Get the sentences from the essay by separating sentence end by full stop marks.

*Step 2:* Apply the trained model 1 on each sentence.

*Step 3:* If the sentence has any grammatical or spelling error, assign 0. Else 1

*Step 4:* Repeat *Step 3* for all the sentences and count the total number of sentences.

*Step 5:* Count the total number of 1’s and divide it by the total number of sentences.

*Step 6:* Multiply the value done on *Step 5* by 10 and this score is projected to the user.

**4.4.2 Model 2 Training and Consumption Algorithm**

**Training Algorithm**

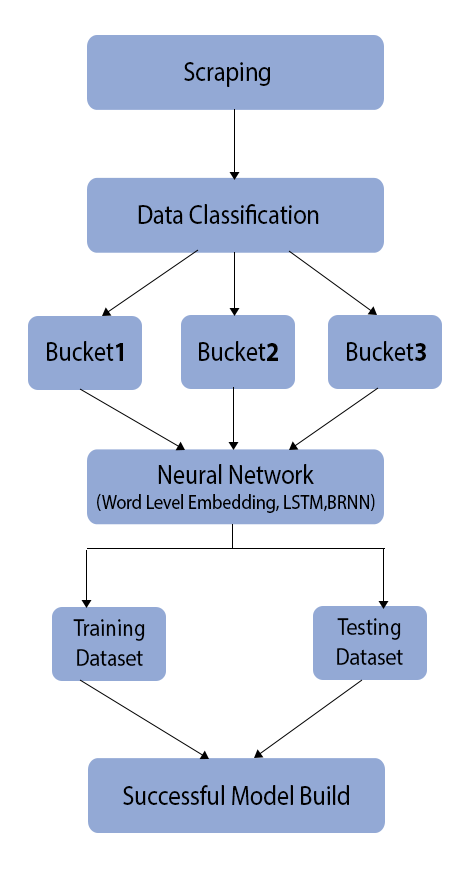
*Step 1:* Download the lexical database from WordNet.

*Step 2:* Separate the lexicons into 3 buckets according to their difficulty level.

*Step 3:* Assign the vector value to each bucket randomly initially using embedding algorithm.

*Step 4:* Apply Bidirectional LSTM on the word and train the words according to the bucket value.

*Step 5:* Save the Model



*Figure 2: Model 2 training flow chart*

**Consumption Algorithm**

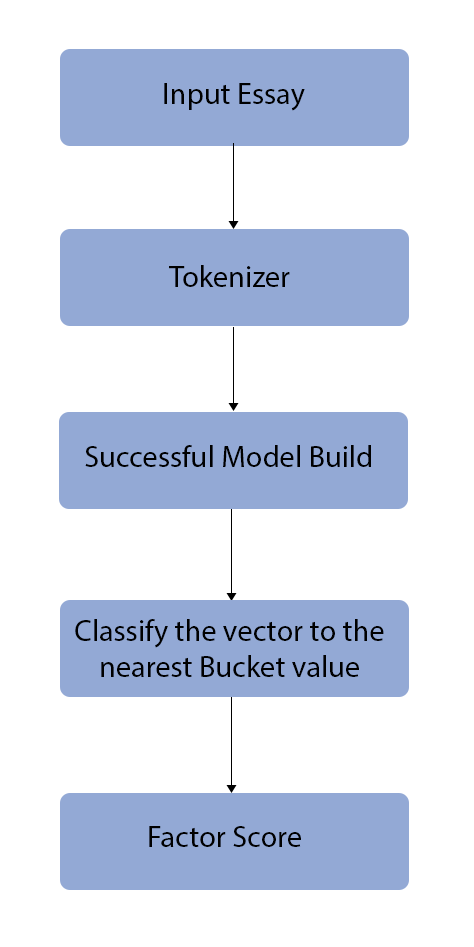
*Step 1:* Get the whole essay as the input

*Step 2:* Tokenize the sentences from the essay

*Step 3:* Apply the built model on the sentences and get the bucket value for each sentence.

*Step 4:* Average the bucket vector value from each sentence.

*Step 5:* Round off and provide the mark for the parameter – Sentence Complexity.



*Figure 3: Model 2 Consumption Algorithm*

**4.4.3** **Model 3 Training and Consumption Algorithm**

**Training Algorithm**

*Step 1:* Download the lexicon database from WordNet.

*Step 2:* Separate the words into various buckets according the style (American, British, Australian English variations).

*Step 3:* Apply Multinomial Gaussian Mixture Model (GMM) on the bucket to get probability distribution model ready.

*Step 4:* Save the Model.

**Consumption Algorithm**

*Step 1:* Get the essay as the input.

*Step 2:* Tokenize the essay into sentences.

*Step 3:* Give the numbers

1 – American English

2 – British English

3 – Australian English and so on for each sentence

*Step 4:* Count the number of 1s, 2s, 3s on the number.

*Step 5:* Assign the mark according to the table.

|  |  |
| --- | --- |
| Continuity of a number | Assign the mark |
| 0% - 20 % | 2 |
| 21% - 40 % | 4 |
| 41% - 60% | 6 |
| 61% - 80% | 8 |
| 81% - 100% | 10 |

*Step 6:* Repeat *step 5* for each number.

*Step 7:* Average out the mark for each category.

*Step 8:* Assign the value from *step 7* and project it as a mark in Style Continuity.

**4.4.4 Model 4 Training and Consumption Algorithm**

**Training Algorithm**

*Step 1:* Download the lexical resource from WordNet

*Step 2:* Separate into 5 buckets naming from primitive to advanced.

*Step 3:* Apply Bag of Words Feature extraction algorithm into these separated database

*Step 4:* Save the Model

**Consumption Algorithm**

*Step 1:* Get the essay as the input

*Step 2:* Tokenize the essay into list of sentences.

*Step 3:* For each sentence, find the most advanced word in that sentence

*Step 4:* Assign the vector value of the most advanced word for that sentence.

*Step 5:* Repeat *step 3* and *step 4* for all the sentence.

*Step 6:* Average out the all the vector value of sentences.

*Step 7:* Project the *step 6* score for the parameter – Lexical Resource

**4.4.5 Model 5 Training and Consumption Algorithm**

**Training Algorithm**

*Step 1:* Get the questions database by scrapping ETS GRE Sample Analytical Writing Assessment Questions.

*Step 2:* Find the subject word in the question and initiate a vector value for all the subject word from the question.

*Step 3:* Find all the relative words according to the initial vector and build a tree/map for the root vector which the subjective word from question.

*Step 4:* Build the tree by repeating *step 3* considering the current vector value to be root node. Thus, a tree is build depth wise.

*Step 5:* Save the built model along with its corresponding root node which the question.

**Consumption Algorithm**

*Step 1:* Get the essay as the input from the user

*Step 2:* Tokenize the essay into list of sentences.

*Step 3:* Get the initial vector value of the question assigned to the user.

*Step 4:* For each sentence, use the model to get the vector value for each sentence.

*Step 5:* From the node, traverse into the built model (tree) according the vector value of each sentence.

*Step 6:* If the path has minimum vector difference, then the sentences are most related to each other.

*Step 7:* Normalize the difference in the range of 0 – 10 and assign the end normalized value for the parameter coherence and cohesion.

**5. Experimental Results and Discussion**

**5.1 Result and Test Accuracy**

From the developed models, we can see 89% of accuracy. The neural network models are developed for each and every constraint which are metrics which the English language testing centers use to evaluate the essay. The constraints based on which the models are developed are

I. Grammar and spell check

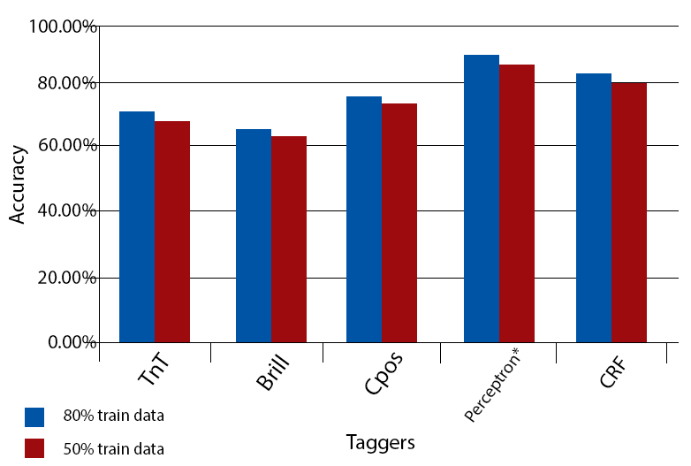
II. Sentence complexity

III. Style continuity

IV. Usage of advanced lexical resources

V. Coherence and cohesion

Based on the above five parameters, our models will be working. For each and every model separate dataset is being given as input. Efficiency of approximately 85% is being obtained from each and every model. When the accuracy is being compared with the previous works, there is a marginal difference.



*Figure III: POS Tagger Comparison*

**5.5 Discussion**

The above graph shows the different accuracies obtained out of the algorithms applied. Multinomial HMM (Hidden Markov Model) is a probability-based classification Model which yields out grammatical mistakes in the sentence. Dictionary Based n-gram model accuracy depends solely on the dataset we used. The dataset to find the correct spelling is scrapped from the WordNet Database.

Gaussian Mixture Model is a probabilistic clustering algorithm which takes random centroid and clusters the datapoint with the probability regarding the centroid. This classifies the level the lexicon used in the sentence.

Since Bidirectional Neural Network is a generative neural network, it works for dynamic datapoints. The need for the dynamic datapoints is due to the complexity of the sentence varies from style to style.

**5.6 Future Works**

**6. Conclusion**

Evaluators take time in correcting the answers and a lot of manpower is used for it. Also, the testing centers must allot places for evaluating the answers. All these steps take time and more money needs to be spent. Thus, by the use of our essay grader, the test centers can organize these tests in a convenient way. This software solves the problem of manually correcting the answers and grading it. We aim to develop this software so that the English language testing centers such as IDP, British Council can make use of this and evaluate the candidates writing skills, the way he has logically connected the sentences, vocabulary, etc. We have developed the software by keeping in mind the different constraints which the evaluators keep in mind while evaluating. The Main working part is with the help of Neural networks and we have created individual network for each and every task being performed. When a candidate appears for the IELTS/TOEFL test, he will be asked to appear for four sections. They are listening, speaking, reading and writing. In future, we can do the essay grading for multiple languages such as Tamil, Telugu, Malayalam, etc. which will be helpful for school education. As of now, speaking section is being evaluated when the candidate speaks. The Evaluator will be asking few questions and the candidate will be answering. This can be automated in future using speech processing. We can record the audio and apply ML algorithms to it to find the eloquence of the candidate. So, these are several future works which can be applied to enhance the product.

**A Novel Algorithm for Automatic Essay Grading using Natural Language Processing Techniques**

# **1. Vishwak Senan G, 2. Pranav Kumar K, 3. Sundaramoorthy T**

**4. Dr D Jayashree, 5. Dr O Pandithurai**

4. Professor at Rajalakshmi Institute of Technology

1. vishwaksenang.2016.cse@ritchennai.edu.in,

2. pranavkumark.2016.cse@ritchennai.edu.in,

3. sundaramoorthyt.2016.cse@ritchennai.edu.in,

4. jayashree.d@ritchennai.edu.in,

5. pandics@ritchennai.edu.in,

## Department of Computer Science and Engineering

## Rajalakshmi Institute of Technology

**Abstract** - Evaluation of an English essay is one of the important and complex tasks which is done manually by skilled and efficient professors and faculties till date. The growth of science and technologies enables to automatic evaluation of an English essay using natural language processing (NLP) techniques. The intelligent system - built upon NLP multiple neural network model - gives out generic evaluation and the topic/question correlation for any given English essay.

**Index Terms** – Automatic evaluation, Natural Language Processing, generic evaluation, topic/question correlation.

1. **Introduction**

International examinations like GRE (Graduate Record Examination), IELTS (International English Language Testing Systems), etc., is gaining popularity day by day as this examination’s results are considered as criteria for various universities and companies. Therefore, the number of international students across various counties are taking the exams, increasing the count day by day and there is a huge time buffer to evaluate their English essays and publish the results. To reduce the stress on the organization who are hosting these examinations and students to practice their writing skills, our project aims to evaluate the English essay so that the organization can focus their work in other aspects of examinations and students can practice at free will.

**II. Literature Survey**

There are various existing NLP techniques which are considered to be best for text classification which is essential for evaluation of the essay. The initial text classification techniques were to classify parts-of-speech of a sentence.

***N-gram model***

The n-gram model otherwise known as Shannon’s Markov Chain [15], estimates the probability of the occurrence of the next word given the n-1 words. Therefore, this is a probabilistic model with a good accuracy if trained from a sophisticated dataset.

***Perceptron Layer***

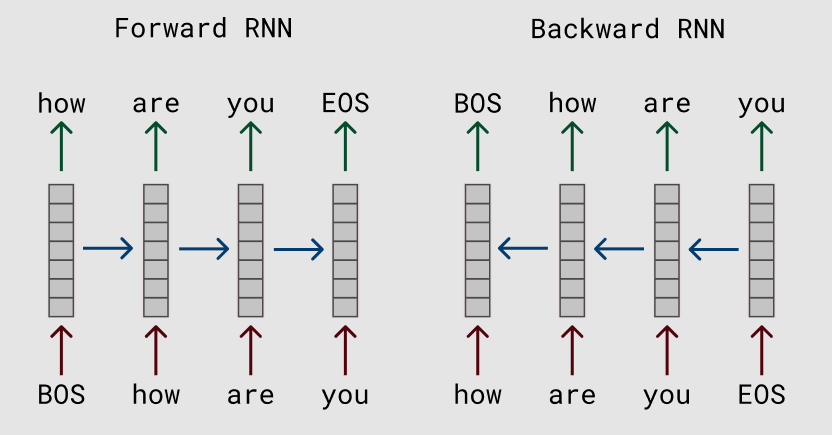
This Model of finding the parts-of-speech tagger of a sentence uses 3-layer perceptron layer with n-inputs, n representing the total number of words in the dataset. While the computational cost of the training the network is drastically higher than the n-gram model but according the previously conducted experiment [14], the accuracy was 99.4% without over-fitting into the data. The accuracy was obtained by using the elastic hidden perceptron layer.

***Long Short Term Memory (LSTM)***

Long Short Term Memory is a type of Recurrent Neural Network (RNN) which is a very famous model to predict a series of a vector. This model has high computational requirement but yields a high accuracy for a large dataset. In a research, the low-cost hardware implementation of LSTM [16] made the neural network to be more optimized and efficient for any given low-cost machine. This invokes the possibility of training the neural network locally rather than renting GPUs online. The low-cost LSTM is achieved by applying stochastic function rather than hyperbolic activation function.

***Bi-directional RNN***

Bi-directional Recurrent Neural Network enables the RNN to work for dynamic dataflow and have multiple activation functions [17].

*Figure: I Bidirectional RNN Flow*

***Clustering***

Clustering algorithm is implemented to categorise the word component to particular to cluster, there exists two major types of cluster, one is soft cluster and other is hard cluster. Hard clusters are the concept of binary categorization, a component can exist in only one cluster. K-means is the algorithm utilised for clustering, where components are fed into nods as training dataset in an iterative methodology. K-means algorithm can be exhaustive and some components may not be segregated into one cluster and its left out. The process of k-means is to determine the distance between the centroid and the datapoints. The best centroid is where it has a mean of zero and standard deviation of one. For determining the centroid points, k-means follows Expectation minimization(EM) [21], this value assign a centroid point where the density of data points is high. In initial iterations, random set of data points are subjected to undergo Euclidean distance calculation [20] and the smallest distance obtained between any two datapoints, from them one of the datapoint is assigned as centroid and once the centroid is formed, in every iterations, its coordination is altered as to include the higher number of datapoints based on the RSS (residual sum of squares) [21] value.

***t- Stochastic Neighbour Embedding***

t-Distributed Stochastic neighbour embedding (t-SNE) is the algorithm implemented to determine the highest similarity between the subjected word component and the target component (subject component is obtained from variance of Gaussian [22]) by converting the high dimensional datapoints into low dimensional datapoints. Considering two parameters involving 2d graphical representation of multiple datapoints, these datapoints are the mapped and clustered with soft or hard clustering using the algorithms k-mean or gaussian mixture model, thus these clusters are needed to be evaluated for relevance or degree of similarity [22] by using normal distribution to scale the datapoint’s value against the distribution curve. The datapoints with highest distribution and with the lowest high dimensional Euclidean distance [22], gets assigned with vector value obtained from conditional probability distribution that represent similarities [22]. T-Sne algorithm forms a matrix based on the distance between dissimilar and similar datapoints representing the maximum similarity value is near the diagonal of the matrix, hence this matrix called diagonal matrix [22], since highest similarity is one, hence the similarity between a word to itself is neglected and similarity between word components are estimated n plotted in the matrix. Now, this algorithm has two parameters by which it can cluster the datapoints in low dimensional graph retaining the cluster similarity, word covariance and relevancy level. Similarity factor is implemented in low dimensional graph to cluster the datapoints at a cost of multiple iterations. Simply the algorithm gets one matrix in each iteration, thus the iteration is kept on progressed until it reaches the matrix value formed with high dimensional graph. In low dimensional graph, datapoints are mapped against t-distribution curve graph to obtain similarity between the datapoints.

***Existing Works***

There are currently many researches in progress to automate the evaluation of English essay. Each research accounts different factors to evaluate an essay.

The rubric based evaluation [18] is a hard-coded rule-based evaluation. The mentioned research explicitly aims to find and describe the narrative genre of the essay. The main limitation is that it is very topic focused and hard-coded.

Another research [19] aims to prepare answer documents for history subject. The goal is achieved by using an information retrieval technique, document pre-processing techniques, yielding good results related to history subject. The text correlation is well implemented, yet it particularly focuses on single subject which is the limitation of this research.

**III. System Architecture**

The main goal of the project is achieving the five crucial factors for evaluating an English essay. The factors are

1. Grammar and spell check.
2. Sentence complexity.
3. Style continuity.
4. Usage of advanced lexical resources.
5. Coherence and cohesion.

The above-mentioned factors are checked by developing a dedicated model for each factor having the input of the whole essay. The primary services of the project (evaluation engine which evaluates the essay) is hosted in any public clouds so that organizations can utilize our services too. Since there are 5 generic factors involved in evaluation of an essay, we will develop 5 neural network model which evaluates on its each of the factor.

Model 1 – Grammar and spell check.

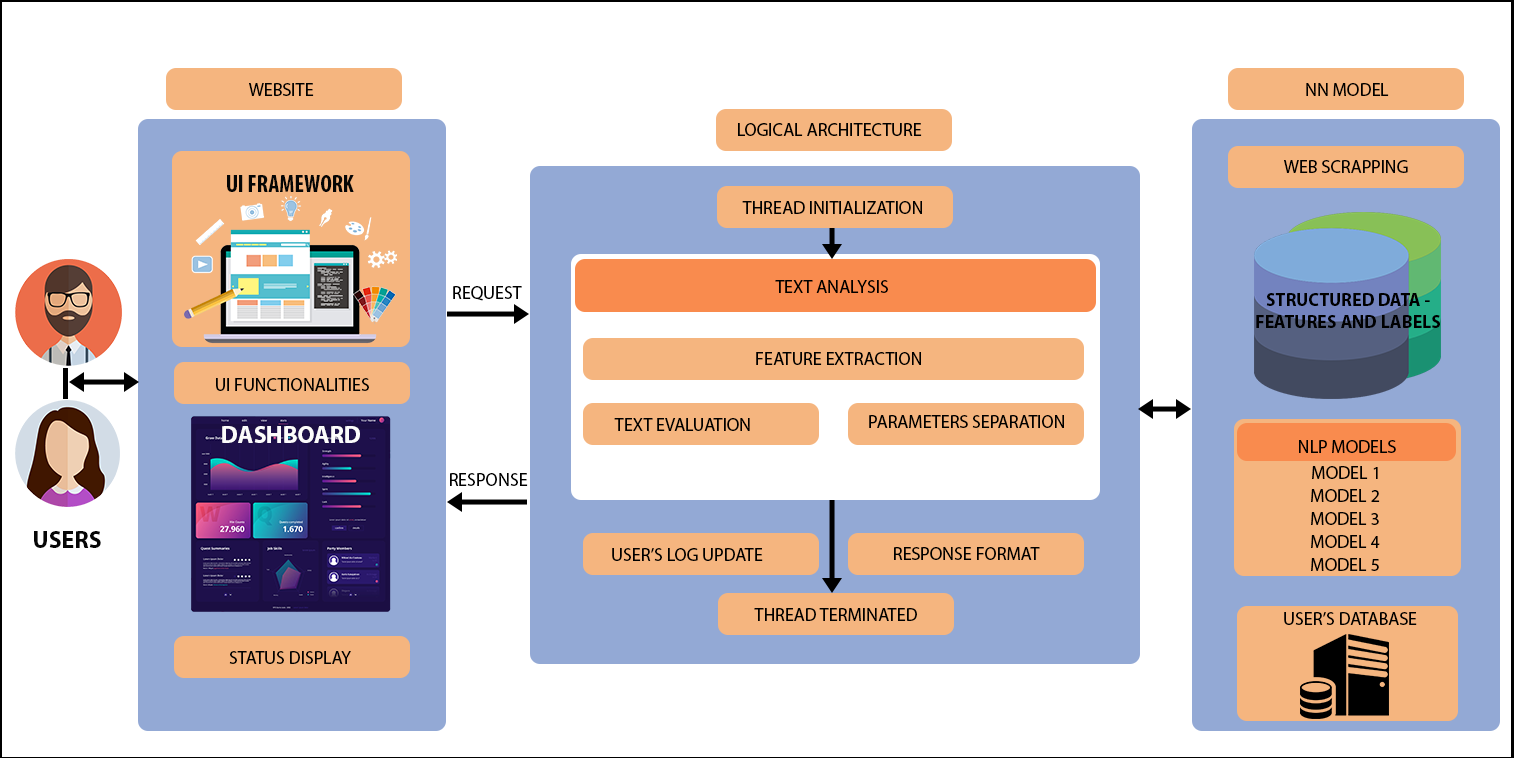
Model 2 – Sentence complexity.

Model 3 – Style continuity.

Model 4 – Lexical Resources.

Model 5 – Coherence and cohesion.

Full architecture is implemented using python as the programming language. The architecture of the project is as given in *Figure 1*.



*Figure I1: System Architecture*

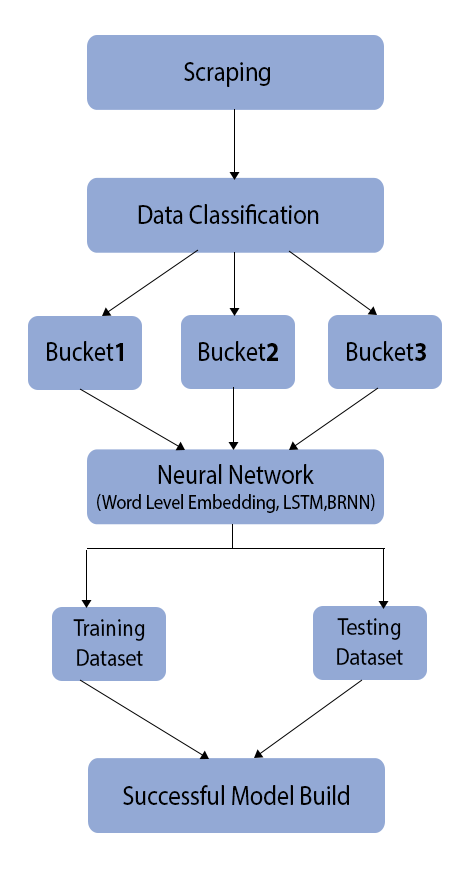
**IV. Novel Algorithm**

The Models which we are developing using different strategies to calculate the scores of each factor required by the project

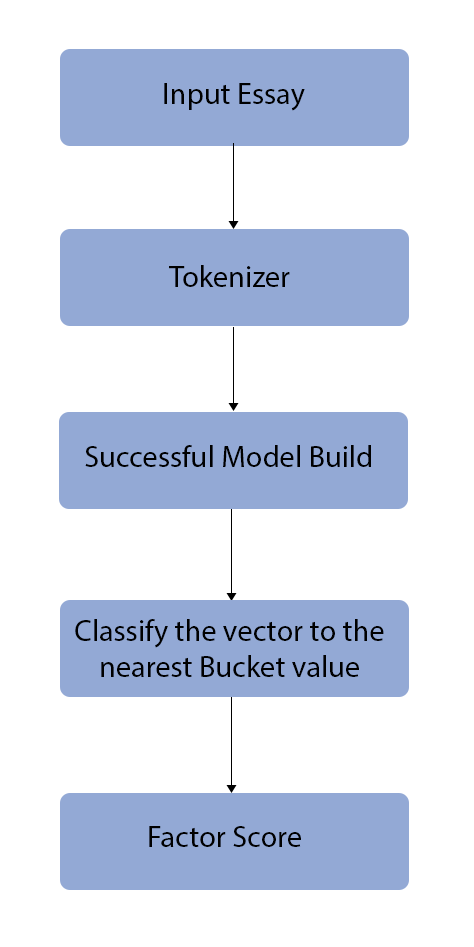
**Model 1** mainly classifies multiple sentences in the paragraph as correct or wrong. The classification mainly revolves around 2 factors. First factor is spelling mistake and second is grammatical errors in that sentence. Using rule based Gaussian Hidden Markov Model (HMM) [1] to find Parts-Of-Speech (POS) tags of a single English sentence from the essay. Using dictionary method and N-gram technique [2] to find the misspelled word and totally irrelated word which is not present in the English dictionary. When the sentence is classified as a perfect sentence or not, we calculate the percentage of correctness by

Correctness score will range from [0-10].

**Model 2** predicts how well the sentence is structured and expressed in higher standards. For example, take the phases “I know everything” and “No secret lies beyond my grasp” conveys the same meaning but the 2nd phase specifically conveys the message that he is a high level English user and thus 2nd phase should receive more score than the first phase. Data is scrapped from the internet and it segregated into 3 buckets namely bucket 1,2,3. The project utilizes Word level embedding, LSTM and Bidirectional Recurrent Neural Network (BRNN) [3]. Hence, we have 3 hidden layers each layer doing different mathematical operation to produce a necessary output. When the neural network is trained, a vector will be produced for the given input word. This vector

will be compared to the three bags and the nearest bag vector value is considered and the output is given. *Figure II* shows the consuming

of the neural network model and *Figure III* shows the training of the mode.

  
 *Figure III: Model 2 Training Procedure*

*Figure IV: Consuming Model 2*

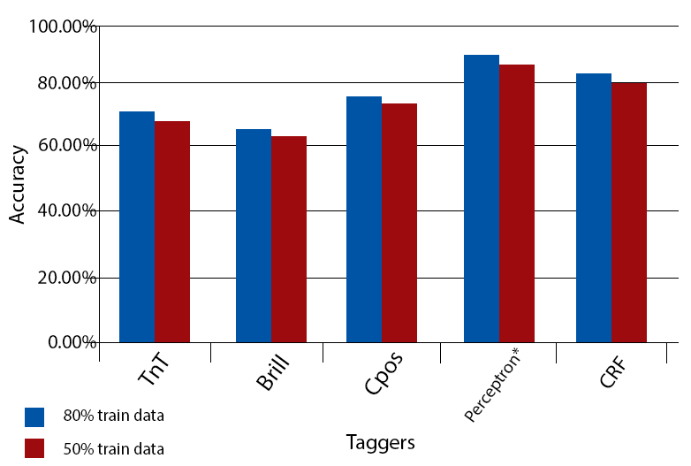
**Model 3**, refers to as whether the user continuous the same English accent throughout the essay. A well-established English literature person will formulate sentences in particular accent so that it is uniform for the people reading the sentence can understand easily. Combining various English accents like American English, British English, Australian English in a single essay will create confusions in readers mind. Using I-vector approach and Gaussian Mixture Model (GMM) [4], we can get the consistency by dictionary vector [2] which will even work for a small dictionary dataset. The marks are calculated in such a way that the same consistency is maintained throughout the essay. The score is calculated as

**Model 4**, categorizes the sophistication level of the user.  Based on the input word provided, the word is mapped to its relevant word-set available in bucket and these word-set are word embedding obtained by Continuous bag-of-words (CBOW) [5]. Total of five buckets are utilised and each of them have a certain level of word standard. From bucket one to five, the word complexity increases and each bucket consists of fuzzy vector value based on the complexity of the word-set. Using word2vec [6,7] word embedding systems, cosine relevance between input corpus and bag-of-words is determined and on the basis of IBW (Intimacy Between Words) [8], the highest intimacy fraction [8] obtained from input word and word-set determines to the bucket input word belongs, and mapped word is assigned with the vector value using term frequency(TF). When the frequency vector of the word is determined, then t-distributed Stochastic Neighbouring [11], the vector distribution gives optimizes the vector value for each back while we train our model. When the model is trained, the sentence is tokenized then the noun POS (parts of speech) tagged word by perceptron-based POS tagging [12] due to its high efficiency compared to another POS tagging algorithm. Then the model classifies the word to its nearest neighbour bucket and the evaluation is assigned for the sentence. Likewise, multiple sentences have their corresponding score and the scores are averaged out to get the final lexical resource credits.

In **model 5**, sentences are extracted as the candidate submits the essay. Then sentences are tokenized by letter string recognition and coding algorithm. The coded words will be given as input to the neural network for recognising the words. After passing as inputs, the coded words are transferred to the sentence syntax analysis module where it indexes the words before they are being processed. A 3-layer hamming neural network is being used to recognize the meaning of the sentence[9]. The model is trained in such a way that it can recognize the most related subject word for a sentence. Therefore, we can get the actual subject/topic of each sentence. Using Task based knowledge and collaborative filtering [13] techniques used to find relevance between the subjects of the consecutive sentence. In this way we can identify the cohesion and cohesion of the entire essay.

**IV. Models Evaluation**

There were variety of POS tagger produced various results, yet perceptron POS tagger yields out higher accuracy (*Figure IV*).



*Figure IV: POS Tagger Comparison*

Multinomial HMM (Hidden Markov Model) is a probability-based classification Model which yields out grammatical mistakes in the sentence. Dictionary Based n-gram model accuracy depends solely on the dataset we used. The dataset to find the correct spelling is scrapped from the WordNet Database.

Guassian Mixture Model is a probabilistic clustering algorithm which takes random centroid and clusters the datapoint with the probability regarding the centroid. This classifies the level the lexicon used in the sentence.

Since Bidirectional Neural Network is a generative neural network, it works for dynamic datapoints. The need for the dynamic datapoints is due to the complexity of the sentence varies from style to style.

**V. Conclusion**

Evaluation of the given English essay is up to date within the IELTS (International English Language Testing Systems) and GRE (Graduate Record Examination) standards and we aim to include various other models which can evaluate other factors of the English essay. The current NLP models used are

1. Multinomial Hidden Markov Model.

2. Bidirectional Neural Network.

3. Long Short Term Memory.

4. Guassian Mixture Model.

5. Continous Bag of Words Method.

6. t-Distributed Stochastic Neighbour.

7. Hamming Neural Network.

8. Task Based Knowledge Filtering.

9. Collaborative Filtering.

**IV. References**

[1] F. Mandita, H. M. Abdullah, T. Anwar and P. Assawinjaiptech, "A novel algorithm for a grammar model checking using statistical Markov model," 2018 Seventh ICT International Student Project Conference (ICT-ISPC), Nakhonpathom, 2018, pp. 1-6.

[2] S. P. Singh, A. Kumar, L. Singh, M. Bhargava, K. Goyal and B. Sharma, "Frequency based spell checking and rule based grammar checking," 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, 2016, pp. 4435-4439.

[3] A. Hassan and A. Mahmood, "Efficient Deep Learning Model for Text Classification Based on Recurrent and Convolutional Layers," 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, 2017, pp. 1108-1113.

[4] O. P. Singh and R. Sinha, "Sparse representation classification over discriminatively learned dictionary for language recognition," TENCON 2017 - 2017 IEEE Region 10 Conference, Penang, 2017, pp. 2632-2636.

[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems. 26:3111–3119. 2013.

[6] Efficient estimation of word representations in vector space, 2013.

[7] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality[J]. In Advances in neural information processing systems. 26:3111–3119. 2013.

[8] R. Wang, W. Pan, J. Ma and H. Wang, "A Synonym Extraction Method Based on Intimacy\*," 2019 IEEE International Conference on Agents (ICA), Jinan, China, 2019, pp. 57-60.

[9] Computational Intelligence: ImitatingLife,

J.M. Zurada, R.J. Marks, C.J. Robinson, IEEE Press, New York, 1994.

[10] Sentence recognition using artiﬁcial neural networks, Maciej Majewski,Jacek M. Zurada,Elsevier B.V. 2008.

[11] N. Rogovschi, J. Kitazono, N. Grozavu, T. Omori and S. Ozawa, "t-Distributed stochastic neighbor embedding spectral clustering," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 1628-1632.

[12] R. Banga and P. Mehndiratta, "Tagging Efficiency Analysis on Part of Speech Taggers," 2017 International Conference on Information Technology (ICIT), Bhubaneswar, 2017, pp. 264-267.

[13] A. Enaanai, A. S. Doukkali, I. Saif, H. Moutachaouik and M. Hain, "The collaborative relevance in the distributed information retrieval," 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, 2016, pp. 1-6.

# [14] Q. Ma, K. Uchimoto, M. Murata and H. Isahara, "Elastic neural networks for part of speech tagging," IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339), Washington, DC, USA, 1999, pp. 2991-2996 vol.5.

# [15] J. A. O'Sullivan, K. Mark and M. I. Miller, "Markov random fields on graphs for natural languages," Proceedings of 1994 Workshop on Information Theory and Statistics, Alexandria, VA, USA, 1994, pp. 47-.

# [16] S. Li, Q. Wang, X. Liu and J. Chen, "Low Cost LSTM Implementation based on Stochastic Computing for Channel State Information Prediction," 2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), Chengdu, 2018, pp. 231-234.

# [17] B. Roy and H. Cheung, "A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network," 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, 2018, pp. 1-6.

# [18] H. W. Lam, T. Dillon and E. Chang, "Determining Writing Genre: Towards a Rubric-based Approach to Automated Essay Grading," 2011 IEEE International Conference on Advanced Information Networking and Applications, Singapore, 2011, pp. 270-274

[19] S. M. F. D. Mustapha, N. Idris and R. Abdullah, "Embedding Information Retrieval and Nearest-Neighbour Algorithm into Automated Essay Grading System," Third International Conference on Information Technology and Applications (ICITA'05), Sydney, NSW, 2005, pp. 169-172.

[20] V. K. Singh, N. Tiwari and S. Garg, "Document Clustering Using K-Means, Heuristic K-Means and Fuzzy C-Means," 2011 International Conference on Computational Intelligence and Communication Networks, Gwalior, 2011, pp. 297-301.

[21] H. Tao, J. Li, T. Luo and C. Wang, "Research on topics trends based on weighted K-means," 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), Macau, 2017, pp. 457-460.

[22] N. Rogovschi, J. Kitazono, N. Grozavu, T. Omori and S. Ozawa, "t-Distributed stochastic neighbor embedding spectral clustering," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 1628-1632.