**Classification of Readability Level of Vietnam Text Documents**

Contents

[Table of Figures 3](#_Toc109310488)

[Table of Tables 4](#_Toc109310489)

[2.1. Factors affecting readability 5](#_Toc109310490)

[2.1.1. vocabulary 5](#_Toc109310491)

[2.1.2. Understanding 5](#_Toc109310492)

[2.1.3. Alphabetical principle 5](#_Toc109310493)

[2.1.4. Proficiency 6](#_Toc109310494)

[2.1.5. Our study area 6](#_Toc109310495)

[2.2. The readability assessment of text 6](#_Toc109310496)

[2.2.1. Related work 6](#_Toc109310497)

[2.2.2. Measurement of text readability 10](#_Toc109310498)

[2.2.3. Applications for computation-based readability assessment 12](#_Toc109310499)

[2.3. Pre-Trained Models 13](#_Toc109310500)

[2.5.1. Definitions 13](#_Toc109310501)

[2.5.2. Existing pre-train models 13](#_Toc109310502)

[2.4. Latent Semantic Analysis 14](#_Toc109310503)

[2.5. The main applications for LSA-based textual research 14](#_Toc109310504)

[2.5.1. Textual readability summary 14](#_Toc109310505)

[2.5.2. Quality of the text 15](#_Toc109310506)

[2.5.3. Tests for Text Quality 16](#_Toc109310507)

[3.1. Related Knowledge 16](#_Toc109310508)

[3.1.1. TF-IDF 16](#_Toc109310509)

[3.1.2. Random Forest (RF) 17](#_Toc109310510)

[3.1.3. Support Vector Machine (SVM) 18](#_Toc109310511)

[3.1.4. K-Nearest Neighbor Model 19](#_Toc109310512)

[3.1.5. Neural Network Model 20](#_Toc109310513)

[3.1.6. PhoBERT 20](#_Toc109310514)

[3.1.7. Roberta 21](#_Toc109310515)

[3.2. Methodology 22](#_Toc109310516)

[3.2.1. Manual Feature Extraction 22](#_Toc109310517)

[3.2.2. TFIDF Feature Extraction 22](#_Toc109310518)

[3.2.3. Train Test Split 22](#_Toc109310519)

[3.2.4. Models Development and Training 23](#_Toc109310520)

[3.2.5. Models Evaluation 25](#_Toc109310521)

[4.1. Dataset 26](#_Toc109310522)

[4.2. Data Preprocessing 26](#_Toc109310523)

[4.3. Data Exploratory Analytic 27](#_Toc109310524)

[4.3.1. Target Variable Description 27](#_Toc109310525)

[4.3.2. Most Frequent Words 27](#_Toc109310526)

[4.3.3. Class Distribution Analysis 28](#_Toc109310527)

[4.3.4. Length of text in Class Distribution 29](#_Toc109310528)

[4.4. Experiments 30](#_Toc109310529)

[4.5.1. Experimental Setup 30](#_Toc109310530)

[4.5.2. Experimental Details 31](#_Toc109310531)

[4.5. Experimental Results 32](#_Toc109310532)

[4.5.1. Train Test Split 32](#_Toc109310533)

[4.5.2. Support Vector Machine 33](#_Toc109310534)

[4.5.3. Random Forest 34](#_Toc109310535)

[4.5.4. KNN Model 35](#_Toc109310536)

[4.5.5. Customize Model (NN) 37](#_Toc109310537)

[4.5.6. Customized Model 2 (Roberta) 38](#_Toc109310538)

[4.5.7. Comparative Study 41](#_Toc109310539)

[4.6. References 42](#_Toc109310540)

# Table of Figures

[**Figure 1:** Classification method of Support Vector Machine. 19](#_Toc109310541)

[**Figure 2:** Classification Method of K Nearest Neighbor. 20](#_Toc109310542)

[**Figure 3:** Architecture of Customized Neural Network Model. 24](#_Toc109310543)

[**Figure 4:** Architecture of Customized Neural Roberta Model. 25](#_Toc109310544)

[**Figure 5:** Top 10 Most Frequent Words 28](#_Toc109310545)

[**Figure 6:** Distribution of Sentiment Classes in Tweets Dataset 29](#_Toc109310546)

[**Figure 7:** Average length bar chart for each category. 30](#_Toc109310547)

[**Figure 8:** Class Distribution in split datasets 32](#_Toc109310548)

[**Figure 9:** Readability Classification - SVM Confusion Matrix 34](#_Toc109310549)

[**Figure 10:** Readability Classification - RF Confusion Matrix 35](#_Toc109310550)

[**Figure 11:** Readability Classification - KNN Confusion Matrix 36](#_Toc109310551)

[**Figure 12:** Readability Classification - NN Confusion Matrix 38](#_Toc109310552)

[**Figure 13:** Training and Testing accuracy plot. 39](#_Toc109310553)

[**Figure 14:** Readability Classification - Roberta Confusion Matrix 40](#_Toc109310554)

[**Figure 15:** Evaluation Scores of trained models. 41](#_Toc109310555)

# Table of Tables

[**Table 1:** Description of manually generated features. 22](#_Toc109310556)

[**Table 2:** Summary of samples and features in dataset. 23](#_Toc109310557)

[**Table 3:** Sample of Dataset 26](#_Toc109310558)

[**Table 4:** Sample of Text before and after preprocessing 26](#_Toc109310559)

[**Table 5:** General Statistics about dataset 27](#_Toc109310560)

[**Table 6:** List of libraries used in environment 30](#_Toc109310561)

[**Table 7:** List of Proposed Experiments in proposed study 31](#_Toc109310562)

[**Table 8:** Readability Classification - SVM Report 33](#_Toc109310563)

[**Table 9:** Readability Classification - RF Report 34](#_Toc109310564)

[**Table 10:** Readability Classification – KNN Report 35](#_Toc109310565)

[**Table 11:** Readability Classification – Customize NN Report 37](#_Toc109310566)

[**Table 12:** Readability Classification – Customize Roberta Report 39](#_Toc109310567)

[**Table 13:** Readability Classification – Comparative Classification Report 41](#_Toc109310568)

# Factors affecting readability

Proficiency in the five essential reading skills has a significant impact on a person's reading ability. The underlying understanding, aptitude, family circumstances, educational environment, and degree of the desire of a person can all have an impact on their reading abilities. A language is difficult for us if there is a high level of vocabulary used and sentences are not short and clear. Multiple sentences are combined to express something. So, it is essential to make readability simple and easy. Sentences must be short and clear. Secondly, vocabulary must be simple. Nevertheless, the five early literacy abilities of phonic awareness, the alphabetical principle, proficiency, vocabulary, and comprehension are very closely related to people's success regarding improved reading performance. Home and educational institutes can provide better support for effective reading performance by comprehending and fostering all five essential skills.

## vocabulary

Throughout the reading, vocabulary requires deriving meaning from words. In essence, students are unable to understand or make any sense of language unless they are familiar with that passage's language. Excellent vocabulary skills enable people to understand very quickly as well as purposefully. Through discussion, reading, direct education, and personal experiences, a person's vocabulary increase every day. A good vocabulary is fostered by frequently and specifically introducing their particular terms.

## Understanding

The development of meaning from reading material is a key component of comprehension. In the absence of understanding, the text genuinely provides no value. Textual comprehension requires the readers to constantly consider and analyze meanings as they are reading. Substantial proficiency in every other essential reading skill is necessary for effective understanding. To read and understand a text, learners need to take into account a variety of factors during reading, including subject, textual structure, the writer's intention, known and new vocabulary, and style. The focus on knowledge and continuous reading is encouraged by excellent understanding.

## Alphabetical principle

The capability to recognize the letters, understand which words are formed up of separate letters, and associated sounds with these letters in writing all fall under the umbrella of alphabetical structure. Understanding the alphabetical order of letters is necessary when using the alphabetical principle. Sequencing, or the ability to sound out language words, is a crucial ability related to the alphabetical principle. The English alphabet, moreover, is complicated and challenging to learn. Numerous letters can have several different sounds, as well as multiple letters can have many different sounds. A concrete understanding of alphabetical principles is encouraged through diligent practice of letter identification, matching letters' sounds to characters, and using the skills inside the words.

## Proficiency

Accuracy, as well as the speed of the learners' reading, are important factors in proficiency. A proficient reader can read the text accurately, swiftly, and in the right tone of voice. For learners, reading becomes much more enjoyable as well as less difficult as they develop their proficiency. Normally, proficient readers can read with absolutely no difficulty. This enables people to focus their energy on understanding the knowledge of vocabulary rather than word recognition and interpretation. The learner must be fluent in reading before he can become proficient in understanding the content.

## Our study area

Machine Learning and deep learning play a crucial role in the text assessment. Numerous studies have been published based on the text assessment. In the field of text assessment, here we are going to classify the document readability level as either difficult or easy.

# The readability assessment of text

People and businesses who wish to create their content material simple to read as well as understand. Those people appreciate simplicity and visibility in the business practices employed throughout all industries as well as professions. Textual readability is the gauge of how generically it can be understood. Complexity, irreconcilability, readability, and design can all be present. In their computations, reading formulas frequently take into account elements like sentencing guidelines, consonant frequency, and vocabulary familiarity. The readability score can be used to determine the amount of knowledge or training someone needs to be capable of reading a piece of text comfortably. The score designates the grade level that roughly corresponds to the total years of education.

## Related work

The past work on textual readability assessments can be divided into two primary groups: machine learning (ML)-based models and conventional formula-based assessments. In conclusion, standard or conventional formulas are a naive combination of simplistic and definable properties. These calculations are done by hand and are adjusted to reflect textual readability. But on the other hand, machine learning-based models incorporate a lot of easy-to-complicated machine-extractable characteristics to make up for conventional models' poor accuracy. Machine learning algorithms are used with a large dataset of pre-labeled texts to discover the relationship between the value of the attributes and the readability of the accompanying text. The proposed models can be used by an employee to reliably score the readability of recently discovered literature.

The most popular as well as the oldest readability algorithm for the English language is the Flesch-Kincaid grading level [1]. Only the average number of words per phrase and the average number of syllables per word are used in the Flesch-Kincaid readability calculation formula to assess textual readability. The Gunning-fog  [2] and Chell-Dale [3] are two additional formulas that use comparable equations and straightforward properties to gauge the overall readability of the text. A predetermined collection of these keywords is used to compute the number of "IJ complicated words" and "J tough words." Equations 1, 2, and 3 show these calculation formulas, accordingly.

Eq. 1

Eq. 2

Eq. 3

The characteristics that are used in the computation above can be computed manually. Programs for assessing textual readability utilize electronically calculated as well as extracted characteristics owing to developments in automatic computation. Word-frequencies and language-based models were utilized in works of by the Collins-Thompson and Callan [4], Lexile [5], and other authors. The overall accuracy of these used models has improved as an outcome of the application of the statistical algorithms in textual readability evaluation. The Flesch-Dayani equation, which is a modification of the Flesch-Kincaid equation, is the sole textual readability evaluation formula that is currently accessible for the Persian language [6]. These constants used in the formula have been particularly adjusted to fit the Persian textual readability rating. In Equation 4, the formula is displayed.

Eq. 4

The conventional equations are simple to use, consume little processing power, and provide easy-to-understand outcomes. Despite the advantages, the used approaches' biggest flaws are their poor accuracy and a stark discrepancy between their findings and subjective assessments [8–11]. Additionally, because the proposed formulas were created specifically for a single language, it is difficult to use them to evaluate the comprehension of writing in some other different languages. Additionally, these algorithms are not appropriate for the short textual applications, that are increasingly common in digital media as well as web now a days [12]. Since the development of machine-learning-based methods, investigators have used these algorithms to produce a more accurate and thorough textual readability evaluation system in order to improve the shortcomings of conventional textual readability evaluation equations.

The evaluation of textual readability can be seen as either a classification or regression challenge. However, literature has demonstrated that assessing textual readability as the classification job is more accurate and useful [13]. The main benefits of machine-learning algorithms over conventional formulas are the inclusion of a larger list of characteristics (naive and sophisticated), as well as the autonomous learning of the relationship between characteristics and readability grade [14]. The machine learning algorithm's selection of characteristics is an essential stage because the model is only as effective as its characteristics. Therefore, the list of characteristics used by several suggested machine learning algorithms for automatic textual readability evaluation distinguishes them from one another. Previous As [14], employed easy variables just like the average range of the characters and syllables in the words; the average range of wording in the sentences; the range of paragraphs in the text; and normal statistical language-based models. Unified characteristics [15] and syntactic features [16] were also used to enhance the realization of the algorithm with improved accuracy.

Vajjala and Meurers [17] offer a cutting-edge methodology for automatic English textual readability evaluation for native users of English reading. The Support Vector Machine (SVM) classification is used in the newly proposed model, which makes use of a wide range of characteristics. There are 46 different aspects in all, including various conventional, lexical, and grammatical characteristics. One subdomain of textual readability evaluation is the assessment of textual readability in second languages. Given the wide utilization of English as a second language, very little comprehensive research has been conducted in this area. Since numerous textual features have varying degrees of influence on the textual level of readability for readers of second languages, several approaches are needed to evaluate the overall readability of English-written texts for a second-language user. A comprehensive study on the evaluation of a second language's textual readability was conducted by Xia et al. [18]. The research by Vajjala and Meurers [19], which also utilized the SVM classification algorithm, used a collection of natural language processing characteristics that included discourse-based features in addition to conventional, lexicon-based, and tree-structured language modeling elements.

In various languages, like French, numerous analogous studies have been conducted to develop automatic textual readability evaluation models [20]. German [21] and Russian language Arabian language [22] and Chinese language Portuguese language [23]. Their paper focuses on English as well as Persian as a testing case for evaluating the readability of bilingual texts. The approach suggested by Mohammadi and Khasteh [24], which also employs the SVM algorithm, is the only machine learning-based textual readability evaluation process for the Persian language that is currently available. In summary, machine learning techniques, assuming the existence of a good database, can achieve improved accuracy in comparison to classical equations as well as be simpler to create. Instead, since they employ so many complex elements, they are difficult to deploy, expensive, language-based, and difficult to understand.

The goal of the project is to reduce the dependence of textual readability evaluation methods on language and complex feature engineering. The most recent developments in deep learning and reinforcement learning, particularly their combination, deep reinforcement learning, have proved helpful in overcoming these issues.

In the past years, scientists have been able to construct larger, stronger, and much more complex neural networks owing to the enormous increase in data that is now obtainable and the advancement of hardware facilities, particularly Graphical Processing Units (GPUs). Additionally, the growing use of the recurrent [25,26], as well as convolutional, frameworks in NLP-based applications has led to an unheard-of rise in the precision of computation-based linguistic frameworks. The development of the latest list of deep NLP algorithms that can achieve cutting-edge accuracy on many common NLP problems without the use of antiquated feature engineering as well as the extraction of features has been sparked by the introduction of vector-based representation of the text (i.e., words-to-vec [27]), which can lessen the requirement for sophisticated feature engineering, The fundamental benefit of the deep NLP classifiers is their capacity to attain improved accuracy with basic characteristics. However, to reach greater levels of accuracy, these artificial neural networks necessitate large amounts of data, specialized hardware, as well as more computational processing capabilities.

One example of a semi-supervised machine learning-based technique is reinforcement learning. Reinforcement learning is particularly useful in NLP problems because it can be trained from partially tagged input. As a result, utilizing a reinforcement learning approaches in the NLP tasks including machine-based translation [28–31], phrase simplifying [32], textual summarization [33, 34], conversation creation [35], questions answering [36], questions generation [37], and textual generation [38] is becoming more popular. Additionally, the deep learning algorithm models that combine reinforcement learning with deep learning serve to combine the benefits of all fields to develop better efficient and precise algorithms for NLP tasks. These features are capable of actively altering their input and deliberately concentrating on the textual content that will be most helpful to them in completing their tasks. These algorithms can attain higher accuracy and efficiency in the NLP problem than the prior algorithms, despite some limitations, including training instabilities.

## Measurement of text readability

The evaluation of textual readability typically entails figuring out how hard it is for readers to comprehend a given text [39, 39]. Textual readability can often be assessed using a predetermined comprehension class or text readability score. A text readability index is employed in the essay to gauge how readable the content is. It is possible to think of the measurement of textual readability as a classification task, i.e., how to develop an accurate prediction algorithm based on the corpus of text with established textual readability levels and apply which model to forecast text with an unknown readability level.

Textual readability assessment studies have been studied for at least a century. The automated assessment of textual readability is still a difficult research domain, and so this subject is far from being "resolved." Investigation into the measuring of readability dates back to the 1920s. The linguistic aspects of text were the main objective of initial comprehension research, as well as relevant lexicon features, just like complexity, variation, and breadth of being utilized were represented by the proxy variables. In order to determine a vocabulary, the complexity criterion is superior to the other. Correlational analysis and the experience of professional judges are primarily used. These studies demonstrate that the study of textual readability has started to consider all facets of characteristic selection. A readability investigation method was originally developed between the 1940s and 1990s. In the effort to accurately assess the textual readability and create the best possible reading problems measurement benchmark, investigators continued to experiment with different comprehension methodologies throughout this time [40–42]. They also added the proxy factors of morphological and grammatical knowledge into an algorithm and created linear parameterization.

In order to better understand how individuals, record and maintain all information in durable memory, investigators started to focus on basic structured information in documents from the 1980s to the 1990s. They also presented cognitive ideas further into the domain of textual readability, just as correlation theory, theoretical schema principle, technology demonstrator philosophy, and propagation amplification principle.) Textual organization and its style are related to textual readability as concepts of language coherence, as well as cohesion, was introduced [43]. Basic mental cognitive design is sometimes referred to as the idea of textual readability. When measuring vocabulary aspects, we also take into account the occurrence of distinct words or word combinations in the textual set and gauge the overall difficulty of the vocabulary comprehension using this likelihood.

Website pages about research and engineering are evaluated for comprehension using the statistical textual language algorithm [44]. Following that, scientists could more thoroughly explore the text as well as the structure of the text thanks to advancements in human language processing technologies, including part-of-speech labeling, grammatical evaluation, as well as language models, that have led to the latest advancements in the study of readability. In addition to the ongoing investigation of verified textual elements, the research of readability has also incorporated the latest ideas in cognitive science. At the moment, verified machine learning techniques like segmentation, extrapolation, and grouping are also applied to create the latest readability assessment techniques, giving rise to the latest textual readability assessment technique, namely the textual readability assessment technique, which depends on artificial intelligence and machine learning for more complex characteristics [45].

Since the start of the 21st century, some machine learning-based methods for measuring textual readability have consistently embedded a wide range of extensive features as well as presented a wide range of potent deep learning methodologies to repeatedly reload the effectiveness of a textual readability assessment model that is still being developed [46]. The latest measurement technique for textual readability is being introduced as a result of the rapid development of the big-data and the advent of deep learning. This technique, which depends on the deep-learning, exhibits significant benefits in terms of precision and mechanization of textual readability assessment [47]. As a result, it represents the most recent development in the field of textual readability assessment methods.

Nevertheless, there are a number of difficulties with the studies on textual readability assessment. First of all, conventional readability assessment methods, including such readability assessment formula methods as well as artificial intelligence-based assessment techniques, significantly rely on the extraction of special artificial characteristics, leaving them farther behind the automated reading assessment. The research challenge is how to free up numerous laborers and autonomously extract features in the age of the big-data is the research challenge. Second, as NLP and ML-based technologies advance, there are ever more humanly recoverable elements (such as semantics and syntactical structure) that influence how challenging it is to read texts. It becomes increasingly challenging to carefully extract the latest characteristics. The challenge is how to more fully capture the characteristics of a combination of letters without adding unnecessary characteristics.

## Applications for computation-based readability assessment

Many applications made possible by these language readability assessment prediction techniques are maybe even more appealing than the latest computerized approaches to text readability prediction [48]. For instance, annotating the Web pages using the readability estimations provides not just certain interesting instructional situations, such as recommending a grade-appropriate material, as well as a little unexpected current feature, just like gauging user motivation throughout the Web-based search, as we detail further. We will now mention the number of significant automatic readability predicting expansions and some applications that have been created for various workloads and demographics. The main applications are given below for the readability assessment.

* Improved readability for second-language students
* Multinational language assistance.
* Assisting readers who have some disabilities
* Computer-supported educational training systems
* Readable content prediction for the web

# Pre-Trained Models

## Definitions

Convolution neural network (CNN) [49], recursive neural network (RNN) [50], graphical neural network (GNN) [51], and attention-based neural network [52] are little examples of the deep network which are widely used for the variety of the artificial intelligence (AI) works in near past years. Machine learning models can autonomously learn low-dimensional consistent vectors (also known as distribution-based representations) from given data as a task-specific characteristic, eliminating the need for complicated feature-based engineering. This is in contrast to earlier non-intelligent or non-machine-learning-based models, which are primarily dependent on manually created characteristics and statistical techniques. Despite these deep learning models' effectiveness, numerous research papers have discovered that one of the major problems is that they are data-hungry. Deep networks typically include a lot of the parameters, which makes them simple to avoid overfit and have poor generalization capability without enough training data [53].

## Existing pre-train models

In machine learning, we have the support vector machine algorithm, a supervised learning technique used for the learning data to perform classification and regression. In the research studies [54,55,56], the author uses SVM for the classification of text to find whether it is easily readable or not. The language they used was Italian, and a dataset was collected from newspapers. They have achieved 80% accuracy. They have got lexical and syntactic features taken into account.

The easiest language model just ignores any conditioned context and autonomously evaluates every term. A linguistic structure like this is referred to as a unigram. A Bigram language model, for example, is that which conditions on the prior sentence, among many more sophisticated varieties of the language model. In [57,58,59] papers, researchers used a unigram model to classify the English language as readable or non-readable. They have used the educational web pages as their dataset. Their claim is to achieve 75% accuracy by using the unigram model. The elements which they have taken into account are the surface linguistic features as well as content features.

Deep learning is a part of data science and machine learning. In deep learning, we use artificial neural networks to train our models on some datasets with representation learning. Here we can have supervised, unsupervised, or semi-supervised learning. A portion of the LSTM-based neural network (NN) algorithm has been used to quantify the intricacy of the Italian language [60,61]. They made use of the textual vocabulary as well as syntactical features. In addition, a system dependent on a recursive neural network (RNN) is presented to operate on a neural network and analyze data patrons [62]. Word placement and grammar are regarded as patterns. Their suggested solution uses grammatical structure as well as assumptions about phrases to determine how difficult the statement is.

In [63,64,65], the different algorithms that can evaluate the multiword complexity are presented. It also uses recurrent neural networks as its foundation. The algorithm assesses the overall syntactical complexity of an Italian-language word sequence. These sentences' grammar is represented by a series of components of the speech labels. The Italian phrases are categorized according to their reading difficulties using the syntactical complexity determined by the RNN once it learns the sequence.

# Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a theory and method for extracting and representing the contextual‐usage meaning of words by statistical computations applied to a large corpus of text (Landauer & Dumais, 1997). The underlying idea is that the aggregate of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other. The adequacy of LSA's reflection of human knowledge has been established in a variety of ways. For example, its scores overlap those of humans on standard vocabulary and subject matter tests; it mimics human word sorting and category judgments;

it simulates word‐word and passage‐word lexical priming data; and, as reported in 3 following articles in this issue, it accurately estimates passage coherence, learnability of passages by individual students, and the quality and quantity of knowledge contained in an essay.

# The main applications for LSA-based textual research

## Textual readability summary

When analyzing the description, it is frequently crucial to know where the subject obtained the knowledge represented in the summary. Corresponding to this, it is crucial to understand that documents have the greatest impact on a subject's memory while conducting research on the subject's thinking after reviewing several texts. Recently, historical memory experiments have discovered that various sorts of texts have varying effects on the subject's understanding and memory [66]. As a part of a few experiments outlined by the author, university students studied 21 documents about the occasions immediately prior to the construction of the Panama Canal as a part of a few experiments. The text-based contained historical accounts, participation, and researcher accounts, including primary sources like agreements and textual messages. The message's 6,097 total words made up the entire document. The participants produced the essay "To what extent is U.S. intervention in Panama justified?" following studying the documents. The paper was propositionalized in an original way outlined by Britt, and then statements from these essays were subsequently compared to others in an original textual document to ascertain whether words had the most effect on the subjects' writings. [67] conducted a reanalysis of the studies and used LSA to forecast whether the materials affected the subjects' thoughts. The objective was to compare specific phrases from these subjects' studies to those in an original document they had studied. The foundation of a subject's information might likely be shown by phrases in essays that were extremely conceptually related to phrases in the original texts.

## Quality of the text

The aforementioned findings show how LSA is capable of semantically content-based comparison. One way to gauge how much knowledge was learned from these texts is to consider the overall degree of similarity measured over what was read in the materials or what was expressed in this essay. As a result, students that produce writings of a better caliber ought to have retained much more of the materials semantically. This investigation utilized data on how linguistically close the subject written in the essay was to the subject studied, as opposed to the first investigation, which only utilized data on that text that was most comparable. Depending upon the cosine in between these vectors of two documented texts, an LSA assessment of this essay produces the rank-ordered collection of matching phrases in the original texts. Foltz utilized the cosine metric to describe the standard of the essay when evaluating it. The greater the grade, the much more comparable the essay's phrases are to the source texts. This strategy acts as a gauge for information retention. It displays how well the participants can remember and apply the material from the books in the writing.

Now, the very same 24 pieces of writing that were utilized in the first investigation were again utilized. We sought out four historical graduate pupils who had previously taught. The participants rated the writings, utilizing a 1-point score and a letter grade ranging from A to F upon becoming acquainted with the 21 materials they had studied. Students did this to determine whether whatever the knowledge was mentioned and how well it was referenced. Students were told to grade the writing in a manner similar to how they graded the graduate courses they had delivered. The assessors were also required to read through the initial 21 articles and select the 10 most crucial passages that would be useful for the essay.

## Tests for Text Quality

The assessment of consistency is a very distinct technique of study utilized in textual comprehension that may also be done using the LSA technique. It has been discovered that using predicate overlapping metrics of text coherence is a useful way to forecast how comprehensible the text will be [68]. This metric determines the consistency of the text by assessing the repetition of the reference group utilized in assertions throughout the text. Also, a localized layer and a global layer of text can be used to calculate the predicate overlapping. The frequency of these arguments and the writer's recollection are closely correlated [69]. For instance, a book that is extremely cohesive will be most successful with users who lack field expertise [70].

As a result, the propositional examination of the text might point out passages in which the consistency fails and will have an impact on the writer's memory. Then, by having these areas repaired, understanding can be further enhanced [71].

Similar to probabilistic frameworks, LSA can assess the degree of conceptual similarity among adjacent textual segments in order to determine consistency. Using LSA, [67] predicted consistency for a collection of sets of texts created by [72]. By two orthogonal altering both levels of local-based coherence and the text on heart disease into the four distinct messages, they altered the actual document. Moreover, by writing subject headers as well as paragraph connection sentences, after the topic had studied one of these four passages, their understanding was evaluated.

# Related Knowledge

## TF-IDF

TF-IDF stand for Term Frequency Inverse Document Frequency. It used for the transformation of text data into numeric features. For the extraction of numeric features, TF-IDF find the importance of the word unlike Count Vectorizer that simply count the frequency of the word in document. The significance of a term in a document is not accurately reflected by utilizing a word's word count as its only feature value. For instance, if a term occurs often throughout all of the articles in a dataset, its count value in various documents is useless for differentiating between them. However, if a word is only found in a small number of papers, then that word's count value in those documents can assist in differentiating them from the other documents. As a result, a word's feature value, or relevance, for a text depends on both the frequency of its use in that document as well as the corpus as a whole. This method of finding word importance in a document is known as term frequency-inverse document frequency (TF-IDF) weighting scheme.

The word "frequency" refers to the ratio between the number of times a word appears in a text and the total number of words in that text. As a result, it is a normalized measure that accounts for document length. Let's use tfij to display the frequency of words in document j. The number of documents in the corpus that contain word I is indicated by the document frequency of word i. Let's use dfi to denote the document frequency for word i. The following formula is used to calculate the TF-IDF weight for word I in document j with N being the number of documents in the corpus:

## Random Forest (RF)

Random Forest is a tree-based ensemble learning model for the classification and regression problems. It develops the multiple trees during the training of the model. The final output is the class that selected by the majority trees. It can also be used to rank the features based on their importance. In a random forest model, there are many different decision trees. The random forest algorithm creates a "forest" that is trained via bagging or bootstrap aggregation. The efficiency of machine learning algorithms is increased by bagging, an ensemble meta-algorithm. Depending on the predictions of the decision trees, the (random forest) algorithm determines the result. It makes predictions by averaging or averaging out the results from different trees. The accuracy of the result grows as the number of trees increases. The decision tree algorithm's shortcomings are eliminated with a random forest. It improves precision and lowers dataset overfitting.

Features of Random Forest model are following:

* Compared to the decision tree algorithm, it is more accurate.
* It offers a practical method for dealing with missing data.
* Without hyper-parameter adjustment, it can generate a reasonable prediction.
* It fixes the overfitting problem with decision trees.
* At the node's splitting point in each random forest tree, a set of features is chosen at random.

## Support Vector Machine (SVM)

A supervised machine learning approach called "Support Vector Machine" (SVM) can be applied to classification or regression problems. However, classification issues are where it is most frequently utilized. SVM uses different types of kernels for the classification of data like Linear, Non-Linear, Polynomial and Sigmoid. Kernel is a function in SVM that helping in solving the problem. SVM is a very useful model high dimensional dataset.

When using the SVM classifier, each data point is represented as a point in n-dimensional space (where n is the number of features you have), with each feature's value being the value of a certain coordinate. Next, we perform classification by identifying the hyper-plane that effectively distinguishes the two classes (look at the below snapshot).



**Figure 1:** Classification method of Support Vector Machine.

## K-Nearest Neighbor Model

The KNN model believes that related things are located nearby. In other words, related things are located close to one another. The majority of the time, related data items in the below image are near to one another. This presumption must be true enough for the KNN algorithm to be effective. KNN uses the arithmetic we may have learned as children—calculating the distance between points on a graph—to encapsulate the idea of similarity (also called distance, proximity, or closeness).

**KNN Algorithm**

* One of the simplest machine learning techniques, based on the supervised learning method, is K-Nearest Neighbor.
* The K-NN algorithm makes the assumption that the new case and the existing cases are comparable, and it places the new instance in the category that is most like the existing categories.
* A new data point is classified using the K-NN algorithm based on similarity after all the existing data has been stored. This means that utilizing the K-NN method, fresh data can be quickly and accurately sorted into a suitable category.
* Although the K-NN approach is most frequently employed for classification problems, it can also be utilized for regression.
* Since K-NN is a non-parametric technique, it makes no assumptions about the underlying data.



**Figure 2:** Classification Method of K Nearest Neighbor.

## Neural Network Model

Artificial neural networks are made to mimic the functioning of neural networks in the brains of humans and other animals. Machine learning acquires the model architecture needed to handle increasingly complicated data by mimicking and modelling the function of neurons. Artificial neural networks come in a wide variety of forms, with many early incarnations appearing straightforward in comparison to contemporary methods. For advanced deep learning models, artificial neural networks are utilized as the architecture. The brain's neurons are modelled as simplified versions of artificial neurons or nodes. The number and strength of connections between each artificial neuron and other nodes varies depending on the type of artificial neural network. Between the input and output layers of the network, there are often layers of nodes. Because of the density of these layers, this multi-layered network architecture is also referred to as a deep neural network. These many layers in artificial neural network models can pick up on various data aspects. Hidden hierarchical layers enable complicated concepts or patterns to be understood from processed data.

There are five types of neural networks models:

* Feedforward artificial neural networks
* Perceptron and Multilayer Perceptron neural networks
* Radial basis function artificial neural networks
* Recurrent neural networks
* Modular neural networks

## PhoBERT

BERT (Bidirectional Encoder) is a transformer-based machine learning model in NLP. It is pretrained model that jointly work from left to right and right to left.  It is pre-trained on theEnglish Wikipedia with 2,500M and words-Books-Corpus with 800M words. Due to its bidirectional work, we can achieve significant results. As the BERT uses transformer to get the meaning of the word, BERT is more attention-based algorithm compare to the LSTM model.

During the training of the model, BERT assign the small probability value to each token. After assigning the probability value, an activation function calculates the final score or polarity value. If the final score is greater than the specified threshold value then BERT predict the difficult class. If the value of final score is below the threshold value, then class may be normal or easy.

## Roberta

The Robust Optimized BERT Pretraining Approach (Roberta) is a Natural Language Processing based algorithm and tweaked version of the well-known NLP framework BERT, created by Facebook. It much more closely resembles a method to effectively train and optimize the Bidirectional Encoders' Representation from Transformer (BERT). Modern consequences in the variety of different NLP tasks have been achieved since the emergence of the BERT. Instead of immediately training a model on the dedicated annotated dataset for that job, BERT employs a robust strategy that pre-trains a model on the very huge dataset beforehand. Facebook developed Roberta, which expands on the BERT. It alters the following elements of the BERT:

**Size of the Training:**  Roberta mostly uses a large dataset as compared to BERT. It utilized CC-NEWS which is 76G, Open Web Texts, which is 38G, Stories, which is 31G, and Books Corpus, which is 16G, which amounted to a total of like 160 GBs of data for the pre-training of the model. This huge dataset was quite possible to be trained on a model due to the 1024-V100 Tesla Graphics processing unit that ran for a couple of days. Although BERT merely uses the Books Corpus dataset for the pre-training (16GB).

**Dynamic Mask Patterns:** A token used for the masking Language Model (MLM) aim is hidden by the BERT throughout the pre-training. Such randomized masks are carried out during the pre-processing phase only once, producing a single stationary mask. Replicating the training data ten times ultimately leads to the masking of every sequence in ten different ways. Roberta prevents using repeated training masks for every training example. The outcome is dynamically depicted in the following picture, which demonstrates that it outperforms stationery masking.

**Learning Sequences:** Compared to the BERT, the Roberta was trained on larger patterns. The model of BERT is learned using 1 million iterations and 256 pattern batches. Neural Machines Translation (NMT) research from the past has demonstrated how learning, including with really large small batches, can enhance either optimizing speed or overall performance management. Roberta is trained using 125k movements and 2k sequencing as well as 31k phases and 8k patterns for this reason. The experiment's image demonstrates that both 125k increments and 2k sequencing produce superior results.

# Methodology

In this section, we discuss the detail methodology of predicting the readability level of Vietnam documents. The rest of the section will conclude the methodology of feature extraction, models development and training followed by the models evaluation.

## Manual Feature Extraction

In the manual feature extraction module, we extract the six rule-based features for each document and prepared a featured dataset based on six features. We extracted the six features label as NSEN, NWO, NCHAR, NSYL, NDWO, and AWLS. The meaning of each feature is described in below table.

**Table 1:** Description of manually generated features.

|  |  |  |
| --- | --- | --- |
| No. | Name | Description |
| 1 | NSEN | It describes the number of sentences in the document. |
| 2 | NWO | NOW represent the count of total words in document. |
| 3 | NCHAR | It calculated by the total number of characters in the document |
| 4 | NSYL | Count of Syllables in the document. |
| 5 | NDWO | NDWO is the count o distinct words in the document. |
| 6 | AWLS | AWLS is the average word length for each category |

## TFIDF Feature Extraction

Feature extraction process refer to the process of extracting the features from raw data and unformatted data. As the selected dataset for proposed study base on the non-structural text data, the data need to pass from feature extraction process. We convert the text data into structural data by extracting the features from text data. From the well-known methods (Count Vectorizer, TF-IDF, and ##) in NLP domain, we used the TF-IDF vectorizer for the extraction of features from text data.

TF-IDF extract the features based on the frequency of the word in the documents (section ##). By using the TF-IDF, we convert each document into the 100 features base on the frequency of the words. After the complete training of the TF-IDF vectorizer, it extracts the 100 features for each document.

## Train Test Split

After the complete extraction of the features from dataset, the dataset was split into three different subsets label as training set, testing set and validation set. The converted feature dataset was slit into training, testing and validation set with the ration of 70%, 20% and 10%. After the split of dataset, training, testing and validation set contain the 1314, 365 and 146 samples. The shape of the all subsets is available in the Table 2.

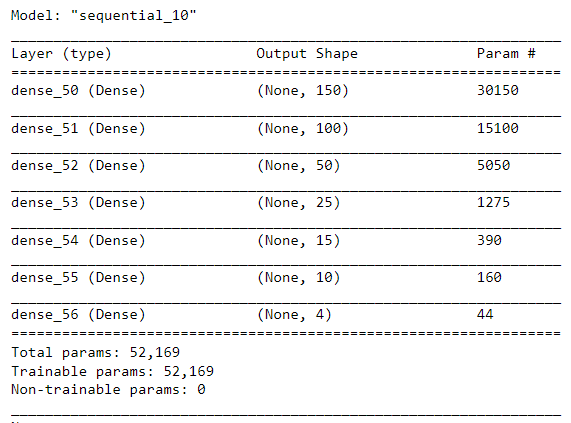
**Table 2:** Summary of samples and features in dataset.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Training Set | Testing Set | Validation Set |
| Number of Samples | 1314 | 365 | 146 |
| Number of Features | 100 | 100 | 100 |

## Models Development and Training

For the prediction of the readability labels of the text document (text features), we initialize the different machine learning and deep learning models. From the machine leaning models, we used the Random Forest, Support Vector Machine, and K Nearest Neighbor algorithm for the classification of documents. From the deep learning models, we used the pretrained Roberta model and develop a neural network model. For the training of the models, we used the training and validation set. For the deep learning, categorical cross entropy was used as loss function with learning rate of 0.001. we also set different hyper parameters including the Adam Optimizer and 50 epochs.

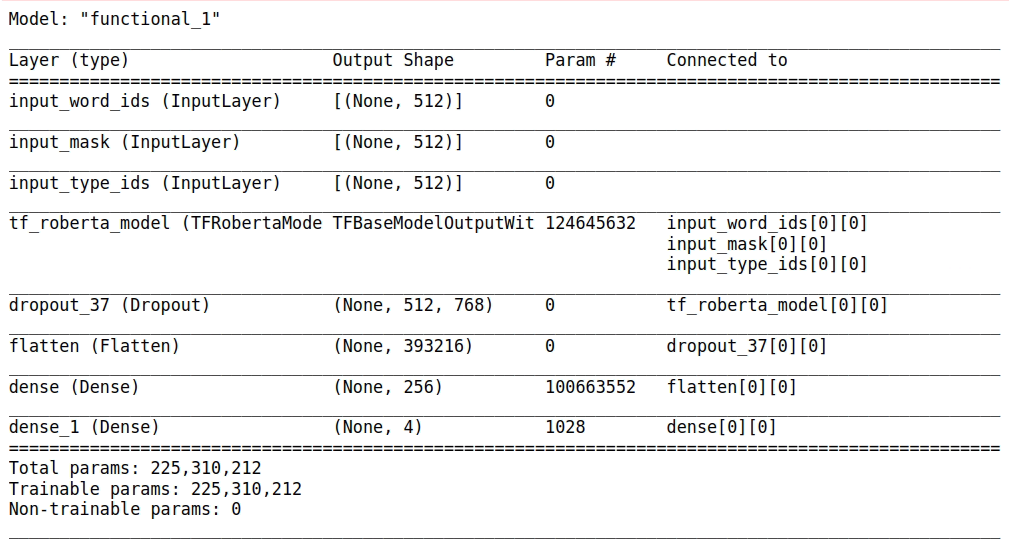
For the development of the neural network model, we used the input layer, output layer and five hidden layers. We used the 150 and 4 neurons on input layer and output layer respectively. The hidden layer also used the 100, 50, 25, 15, 10 neurons respectively with ReLU activation function. The summary of customize develop model is presented in Figure 3.



**Figure 3:** Architecture of Customized Neural Network Model.

For the development of Roberta model, we used the pretrained Roberta model. We extend the Roberta pretrained model with a drop layer, flatten layer and two dense layers. Last dense layer was used as the out layer that have the dimension equal to the number of classes in dataset. The dropout layer used the 0.1 dropout rate that’s means it will drop the 10% data coming from the Roberta pretrained model. The flatten layer convert the data metrics into the data vector and dense layer normalize the data.

For the training of the model, firstly data was prepared. Roberta Tokenizer was used to tokenize the text in the document. Further, all the tokens were transformed with their corresponding numerical id value. After getting the tokens numerical value, manually generated features for each sample were appended at the end of the tokenizer features. Maximum 256 features were used for the classification of readability level. After preparing the dataset, the model was trained on the training set and evaluated on the test set. The architecture of complete model is presented in Figure 4.



**Figure 4:** Architecture of Customized Neural Roberta Model.

## Models Evaluation

For the evaluation of the trained models, we used the selected evaluation measure (Section ##). The test set with 146 samples was used for the testing of the model. We used the accuracy, precision, recall and f1-score for the evolution of the trained model. The evaluation measures were calculated by using the formula of that measure. The equation of calculating the measures is presented in eq 5-8.

Eq. 5

Eq. 6

Eq. 7

Eq. 8

# Dataset

The dataset of text documents was downloaded from ## that was consist of 1825 text documents. The documents of the downloaded dataset were labeled with the readability level. There were four readability levels of the documents in the dataset labeled as very-easy, easy, medium, and difficult. The label of each document represents the readability difficulty level of content in that document. The documents in the dataset are based on average ## words. The content of the document in dataset was available Vietnam language. Collectively, our selected dataset was based on 1825 document in Vietnam language with different difficulty levels of readability. The sample content from each class is represented in Table 3.

**Table 3:** Sample of Dataset

|  |  |
| --- | --- |
| **Text** | **Class** |
| Có một cô gái đi ngang một con đường nhỏ và thấy một chàng trai đeo tấm bảng với dòng chữ : " Free hugs " . | Very-Easy |
| Hồi đó đồ ăn sáng còn chưa có chứ đừng nói đến việc mua một cuốn báo để đọc . | Easy |
| Thì ra không phải tôi đang cô đơn , vẫn còn có người đang nhớ tôi , đang nghĩ về tôi , chỉ có điều họ hơi xa tôi một chút . | Medium |
| Trên cơ sở khảo sát và thống kê , ngôi sao là biểu tượng có tần suất xuất hiện khá lớn trong truyền thuyết dân gian Việt Nam . | Difficult |

# Data Preprocessing

Preprocessing of dataset in the essential step to clean and transform data for optimal performance of machine learning models. In this regard, we use different techniques of data preprocessing like cleaning and tokenization.

For the cleaning of the data, we remove all the stop words from tweets. In addition, we also remove the punctuations, special characters, numbers, URLs from tweets dataset. Furthermore, the emojis and hash tags between the tweets text were removed and remaining text was converted into lower case. In the preprocessing phase last step was stemming that convert the different form of words into its standard form. For the stemming of tweets, we used Porter Stemmer Algorithm from NLTK tool. It also helps in the extraction of the important features. The sample of tweet after preprocessing is shown in Table 4.

**Table 4:** Sample of Text before and after preprocessing

|  |  |
| --- | --- |
| **Text before Preprocessing** | **Text After Preprocessing** |
| Khái niệm diễn ngôn trong nghiên cứu văn học hôm nayThời gian gần đây khái niệm diễn ngôn đã xuất hiện rất nhiều trong các bài nghiên cứu đủ loại , nhiều đến mức không sao có thể định nghĩa thông suốt hết | khái niệm diễn ngôn trong nghiên cứu văn học hôm naythời gian gần đây khái niệm diễn ngôn đã xuất hiện rất nhiều trong các bài nghiên cứu đủ loại nhiều đến mức không sao có thể định nghĩa thông suốt hết |

# Data Exploratory Analytic

## Target Variable Description

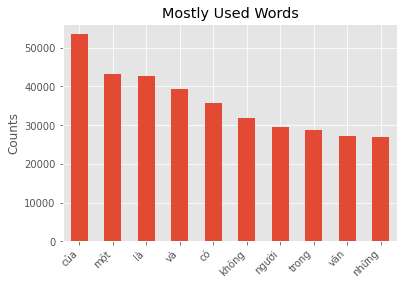
The Vietnam text document dataset contain the difficulty of readability feature that have the most suitable target variables for our proposed solution. After the initial understanding of the dataset and by considering our problem statement, we finalize the readability label feature as our target variable. The target variable contains the four unique values that consider as classes for the prediction of readability.

**Table 5:** General Statistics about dataset

|  |  |
| --- | --- |
| Features | ID, Text, Label |
| Target | Readability |
| Classes | Very-Easy, Easy, Medium, Difficult |

## Most Frequent Words

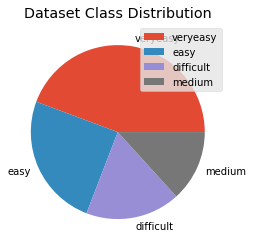
We perform the preprocessing steps on the text content of dataset to found the most frequent words in dataset. For the extraction of most frequent words, we remove stop words, punctuation, hashtag sign, special characters and change them to lover case. The top 10 most frequent words is shown in Figure 5.



**Figure 5:** Top 10 Most Frequent Words

## Class Distribution Analysis

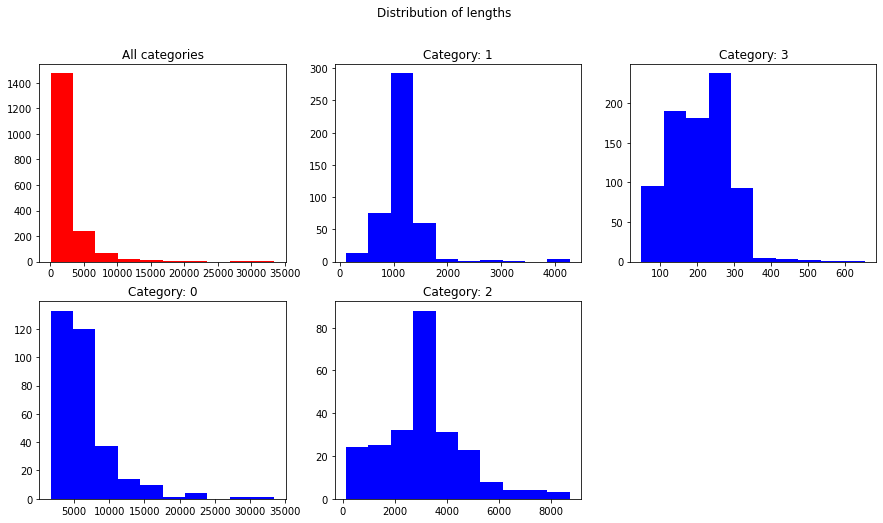
For the understanding of dataset, we tried to find out the number of documents against each category. We found the four types of readability level against the documents. The number of documents for each class is shown in Figure 6.



**Figure 6:** Distribution of Sentiment Classes in Tweets Dataset

## Length of text in Class Distribution

We also analyze the length of the documents in each category. By analyzing the document length, we see the large length documents in difficult category and small length document in very easy category. For the analysis of document length, we plot the histogram for each category in Figure 7. In the Figure 7, category 0, 1, 2, and 3 represent the difficult, easy, medium and very easy category. The plotted document showed that the difficult category contains the majority documents with 4000 words length while the very easy category contains the document with average 250 words. The histogram plot also reveal that the average words proportionally increase with the increase of readability level. The histogram plot is represented in Figure 7.



**Figure 7:** Average length bar chart for each category.

# Experiments

## Experimental Setup

We will discuss the development environment, used libraries, programming language and all other related setting for the training of the proposed models. All the experiment s were performed by using the python programming language. The python 3.7 version was use to perform the experiments. A Conda environment was established with python version 3.7 for the training of the models. All the essentials library were installed in the developed environment.

From the well-known framework for deep learning models, we used the TensorFlow framework for the development and the training of the proposed models. We installed different libraries in our established environment. The list of all core libraries with their version number is listed in below table (Table 6).

**Table 6:** List of libraries used in environment

|  |  |
| --- | --- |
| **Library Name** | **Version** |
| PyTorch | 1.7.1 |
| TensorFlow | 2.3.0 |
| Matplotlib | 3.5.2 |
| scikit-learn | 1.0.1 |
| Pandas | 1.3.5 |
| Numpy | 1.19.0 |

## Experimental Details

We perform number of experiments for the classification of Vietnam text documents. The detail of all experiments in presented in Table 7.

**Table 7:** List of Proposed Experiments in proposed study

|  |  |  |
| --- | --- | --- |
| Experiment no. | Experiment Name | Details |
| Experiment 1 | Feature Extraction | Extract features from text by using the frequency of words through TF-IDF vectorizer. |
| Experiment 2 | Train Test Split | Split the dataset with the ration of 70%, 20, and 10% in raining, testing and validation set. Use the Train test Split function of scikit learn library. |
| Experiment 3 | Random Forest | Train the Random Forest model with the 1314 samples. Use the default value for all hyper parameters of Random Forest. After the Complete training of the model, Evaluate the model on test set. |
| Experiment 4 | SVM | Train the Support Vector Machine model with the 1314 samples. Use the default value for all hyper parameters of SVM. After the Complete training of the model, Evaluate the model on test set. |
| Experiment 5 | KNN | Train the KNN model with the 1314 samples. Use the default value for all hyper parameters of KNN. After the Complete training of the model, Evaluate the model on test set. |
| Experiment 6 | DNN | We train the customize Dense Net model with the training and validation set. Model was trained with 50 Epochs and 0.001 learning rate. After the complete training of the model, use the test set to calculate the performance of the model. |
| Experiment 7 | PhoBERT |  |
| Experiment 8 | Roberta |  |
| Experiment 9 | Comparative Study | After the training of the all models, we perform the comparative study. In this study, we plot the comparison graph to evaluate the best model. |

# Experimental Results

The experimental result section will present the result of all proposed experiments via different tables and graphs.

## Train Test Split

By splitting the dataset into three different set, we get the train, test and validation set. Train Test Split function of scikit learn generate these subsets from the text document dataset. The class distribution in the generated train test and validation set is presented in Figure 8.

|  |  |
| --- | --- |
|  |  |
|  | |

**Figure 8:** Class Distribution in split datasets

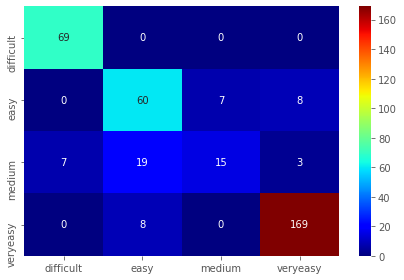
## Support Vector Machine

After the complete training of the SVM model, random forest showed the 0.92% test accuracy on test set. The complete classification report of SVM model on test set in presented in Table 8.

**Table 8:** Readability Classification - SVM Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report for SVM Model** | | | | |
|  | precision | recall | f1-score | support |
| difficult | 0.91 | 1.00 | 0.95 | 69 |
| easy | 0.69 | 0.80 | 0.74 | 75 |
| medium | 0.68 | 0.34 | 0.45 | 44 |
| veryeasy | 0.94 | 0.95 | 0.95 | 177 |
|  |  |  |  |  |
| accuracy |  |  | 0.86 | 365 |
| macro avg | 0.80 | 0.77 | 0.77 | 365 |
| weighted avg | 0.85 | 0.86 | 0.85 | 365 |
| Training accuracy Score: 0.8492 Test accuracy Score: 0. 8575 | | | | |

Here the confusion matrix of SVM model



**Figure 9:** Readability Classification - SVM Confusion Matrix

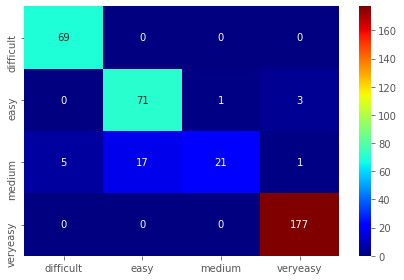
## Random Forest

After the complete training of the random forest model, random forest showed the 0.92% test accuracy on test set. The complete classification report of random forest model on test set in presented in Table 9.

**Table 9:** Readability Classification - RF Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report for RF Model** | | | | |
|  | precision | recall | f1-score | support |
| difficult | 0.93 | 1.00 | 0.97 | 69 |
| easy | 0.81 | 0.95 | 0.87 | 75 |
| medium | 0.95 | 0.48 | 0.64 | 44 |
| Very easy | 0.98 | 1.00 | 0.99 | 177 |
|  |  |  |  |  |
| accuracy |  |  | 0.93 | 365 |
| macro avg | 0.92 | 0.86 | 0.87 | 365 |
| weighted avg | 0.93 | 0.93 | 0.92 | 365 |
| Training accuracy Score: 0.9266 Test accuracy Score: 0. 9260 | | | | |

Here the confusion matrix of random forest model



**Figure 10:** Readability Classification - RF Confusion Matrix

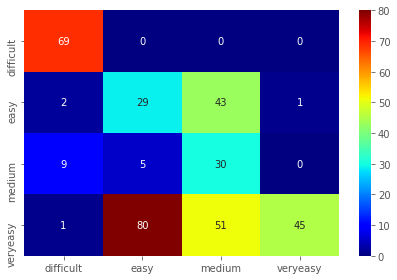
## KNN Model

KNN model showed the 47% test accuracy on test set after the complete training on train set. The complete classification report of KNN model is shown in Table 10.

**Table 10:** Readability Classification – KNN Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report for KNN Model** | | | | |
|  | precision | recall | f1-score | support |
| difficult | 0.85 | 1.00 | 0.92 | 69 |
| easy | 0.25 | 0.39 | 0.31 | 75 |
| medium | 0.24 | 0.68 | 0.36 | 44 |
| Very easy | 0.98 | 0.25 | 0.40 | 177 |
|  |  |  |  |  |
| accuracy |  |  | 0.47 | 365 |
| macro avg | 0.58 | 0.58 | 0.50 | 365 |
| weighted avg | 0.72 | 0.47 | 0.48 | 365 |
| Training accuracy Score: 0.4795 Test accuracy Score: 0. 4740 | | | | |

Here the confusion matrix of random forest model



**Figure 11:** Readability Classification - KNN Confusion Matrix

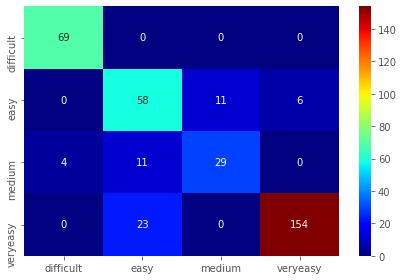
## Customize Model (NN)

We train the customize model with samples training and validation set. Training set was used for the training of the model, while the validation set was used to validate the model during training. After the complete training of the model, the trained customized model was evaluated by using the evaluation measures on test set. Customized NN model showed the 0.84% accuracy on test set. The complete classification report of the trained model on test set in shown in Table 11.

**Table 11:** Readability Classification – Customize NN Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report for Customize NN Model** | | | | |
|  | precision | recall | f1-score | support |
| difficult | 0.95 | 1.00 | 0.97 | 69 |
| easy | 0.63 | 0.77 | 0.69 | 75 |
| medium | 0.72 | 0.66 | 0.69 | 44 |
| Very easy | 0.97 | 0.86 | 0.91 | 177 |
|  |  |  |  |  |
| accuracy |  |  | 0.85 | 365 |
| macro avg | 0.82 | 0.83 | 0.82 | 365 |
| weighted avg | 0.86 | 0.85 | 0.85 | 365 |
| Training accuracy Score: 0.8502 Test accuracy Score: 0. 8493 | | | | |

We also plot the confusion metrics for the clear understanding of the model predictions. The confusion metrics of the customize NN on test set in shown in Figure 10.

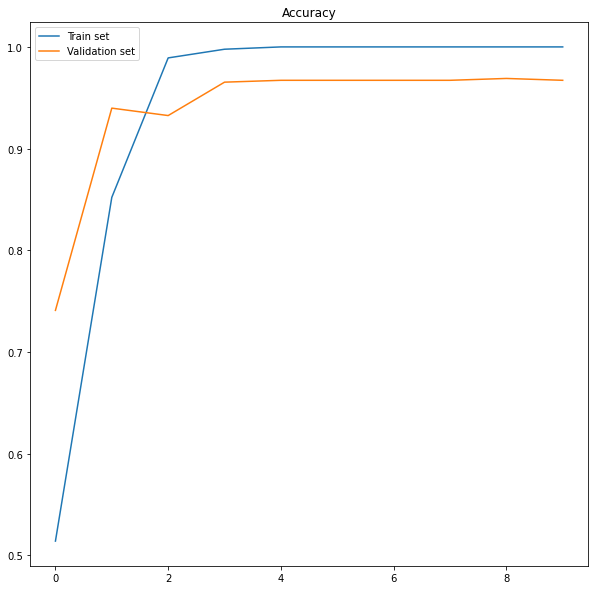


**Figure 12:** Readability Classification - NN Confusion Matrix

## Customized Model 2 (Roberta)

We trained the Roberta model for the classification of document readability level. For the training of the model, six rule based features were extracted. Roberta tokenizer was used to generate more features based on the text of the document. The rule-based features and tokenize features were joined to generate features set. Further, dataset was split into training and testing set with the 70% and 30% ratio.

The model was trained using the training set while the model was evaluated on test set. During the training of the model, model showed the # training accuracy. After the complete training of the model, 548 samples of test set were used to evaluate the performance of the model. Trained Roberta model showed the 0.97% testing accuracy on test set. the accuracy plot on train and test set is shown in Figure 13. The complete classification report of the trained model on test set is also in presented in Table 12.

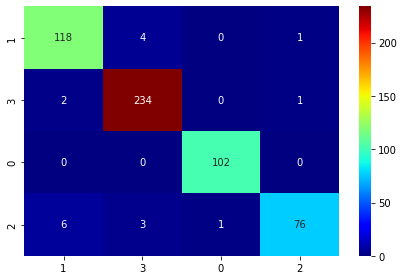


**Figure 13:** Training and Testing accuracy plot.

**Table 12:** Readability Classification – Customize Roberta Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Report for Customize Roberta Model** | | | | |
|  | precision | recall | f1-score | support |
| difficult | 0.97 | 0.99 | 0.98 | 128 |
| easy | 0.97 | 0.88 | 0.93 | 86 |
| medium | 0.94 | 0.96 | 0.95 | 123 |
| Very easy | 0.99 | 1.00 | 1.00 | 102 |
|  |  |  |  |  |
| accuracy |  |  | 0.97 | 548 |
| macro avg | 0.97 | 0.96 | 0.96 | 548 |
| weighted avg | 0.97 | 0.97 | 0.97 | 548 |
| Training accuracy Score: 0.8502 Test accuracy Score: 0. 8493 | | | | |

We also plot the confusion metrics for the clear understanding of the model predictions. The confusion metrics of the customize NN on test set in shown in Figure 14.



**Figure 14:** Readability Classification - Roberta Confusion Matrix

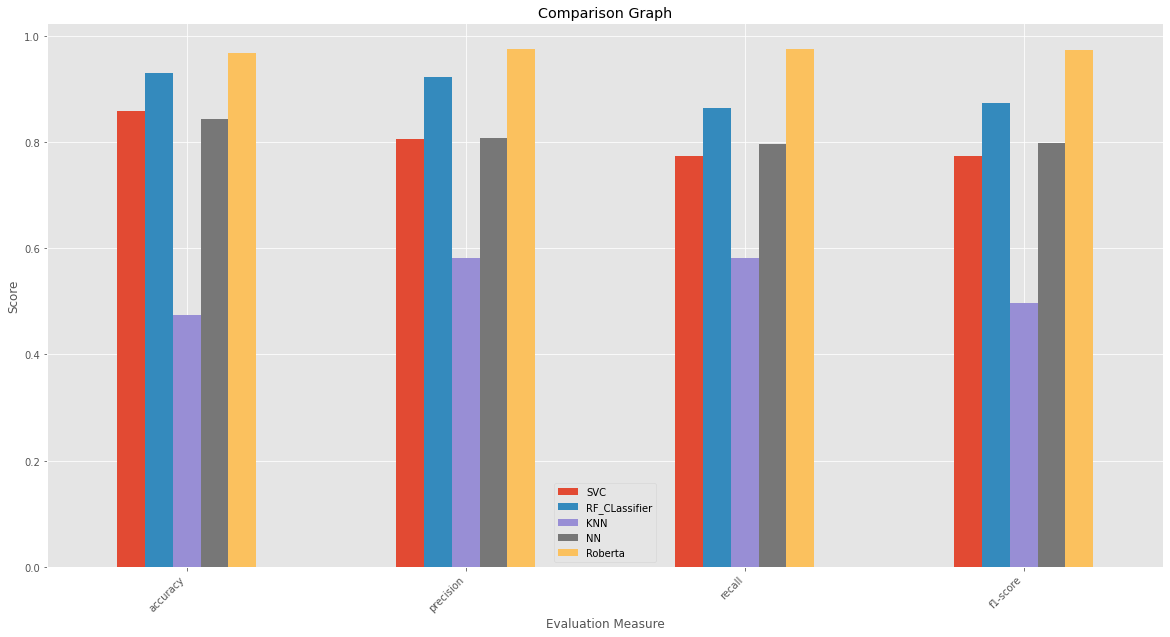
## Comparative Study

Lastly, we perform the comparative study on the results of the trained model. After the training of the proposed model, the results were compiled on test set. The comparative view of all trained models is presented in Table 13.

**Table 13:** Readability Classification – Comparative Classification Report

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Classification Report for all trained model** | | | | | |
| Metrics | SVC | RF Classifier | KNN | NN | Roberta |
| accuracy | 0.857534 | 0.928767 | 0.473973 | 0.843836 | 0.967234 |
| precision | 0.804564 | 0.921242 | 0.581609 | 0.806893 | 0.974564 |
| recall | 0.773928 | 0.863588 | 0.580681 | 0.796157 | 0.973928 |
| f1-score | 0.773447 | 0.87251 | 0.496902 | 0.797691 | 0.973447 |

Further, the comparison bar chart for all evaluation measure was plotted for the comparison of the results. The comparative bar chart of all trained models is presented in Figure 15.



**Figure 15:** Evaluation Scores of trained models.

# References

[1] R. Flesch, “Marks of readable style; a study in adult education." Teachers' College, Contributions to Education, 1943.

[2] R. Gunning, “Technique of clear writing,” 1968.

[3] J. S. Chall and E. Dale, Readability Revisited: The new Dale-Chall readability formula. Brookline Books, 1995.

[4] A. J. Stenner, “Measuring Reading Comprehension with the Lexile Framework," 1996.

[5] K. Collins-Thompson and J. Callan, “Predicting reading difficulty with statistical language models,” Journal of the American Society for Information Science and Technology, vol. 56, no. 13, pp. 1448–1462, 2005.

[6] M. Dayani, “A criteria for assessing the Persian textsâĂŹ readability,” Journal of Social Science and Humanities, vol. 10, pp. 35–48, 2000.

[7] R. G. Benjamin, “Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty,” Educational Psychology Review, vol. 24, no. 1, pp. 63–88, 2012.

[8] J. Hartley, “Is time up for the Flesch measure of reading ease?" Scientometrics, vol.107, no. 3, pp. 1523–1526, 2016.

[9] S.E. Petersen and M. Ostendorf, “A machine learning approach to reading level assessment,” Computer Speech & Language, vol. 23, no. 1, pp. 89–106, 2009.

[10] S. A. Crossley, D. B. Allen, and D. S. McNamara, “Text readability and intuitive simplification: A comparison of readability formulas." Reading in a Foreign Language, vol. 23, no. 1, pp. 84–101, 2011.

[11] I. Pilán, E. Volodina, and R. Johansson, “Rule-based and machine learning approaches for second language sentence-level readability,” in Proceedings of the Ninth Workshop on Innovative Uses of NLP for Building Educational Applications, 2014, pp. 174–184.

[13] L. Feng, M. Jansch, M. Huenerfauth, and N. Elhadad, “A Comparison of Features for Automatic Readability Assessment,” in Proceedings of the 23rd international conference on computational linguistics: Posters.Association for Computational Linguistics, 2010, pp. 276–284.

[14] S.E. Schwarm and M. Ostendorf, “Reading level assessment using support vector machines and statistical language models,” in Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics.    Association for Computational Linguistics, 2005, pp. 523–530.

[15] R. J. Kate, X. Luo, S. Patwardhan, M. Franz, R. Florian, R. J. Mooney, S. Roukos, and C. Welty, “Learning to predict readability using diverse linguistic features,” in Proceedings of the 23rd international conference on computational linguistics. Association for Computational Linguistics, 2010, pp. 546–554.

[16] Y.-T. Sung, J.-L. Chen, J.-H. Cha, H.-C. Tseng, T.-H. Chang, and K.-E. Chang, “Constructing and validating readability models: the method of integrating multilevel linguistic features with machine learning,” Behavior Research Methods, vol. 47, no. 2, pp. 340–354, 2015.

[17] M. Heilman, K. Collins-Thompson, J. Callan, and M. Eskenazi, “Combining lexical and grammatical features to improve readability measures for first and second language texts,” in Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, 2007, pp. 460–467.

[18] T. François, “An analysis of a French as a Foreign Language corpus for readability assessment,” in Proceedings of the third workshop on NLP for computer-assisted learning, 2014, pp. 13–32.

[19] N. Karpov, J. Baranova, and F. Vitugin, “single-sentence readability prediction in Russian,” International Conference on Analysis of Images, Social Networks, and Texts.    Springer, 2014, pp. 91–100.

[20] J. Hancke, S. Vajjala, and D. Meurers, “Readability classification for German using lexical, syntactic, and morphological features,” in Proceedings of COLING 2012, 2012, pp. 1063–1080.

[21] Y.-T. SUNG, W.-C. LIN, S. B. Dyson, K.-E. CHANG, and Y.-C. CHEN, “Leveling l2texts through readability: Combining multilevel linguistic features with the cefr,” The Modern Language Journal, vol. 99, no. 2, pp. 371–391, 2015.

[22] A. K. Al Tamimi, M. Jaradat, N. Al-Jarrah, and S. Ghanem, “Aari: automatic Arabic readability index.”Int. Arab J. Inf. Tech., vol. 11, no. 4, pp. 370–378, 2014.

[23] C. Scarton, C. Gasperin, and S. Aluisio, “Revisiting the readability assessment of texts in Portuguese,” in Ibero-American Conference on Artificial Intelligence. Springer, 2010, pp. 306–315.

[24] Y. Kim, “Convolutional neural networks for sentence classification,”arXiv preprintarXiv: 1408.5882, 2014.

[25] P. Liu, X. Qiu, and X. Huang, “Recurrent neural network for text classification with multi-task learning,”arXiv preprint arXiv:1605.05101, 2016.

[26] P. Zhou, Z. Qi, S. Zheng, J. Xu, H. Bao, and B. Xu, “Text classification improved by integrating bidirectional lstm with two-dimensional max pooling,”arXiv preprintarXiv:1611.06639, 2016.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,”arXiv preprint arXiv:1301.3781 (2013).

[28] A. Grissom II, H. He, J. Boyd-Graber, J. Morgan, and H. Daumé III, “DonâĂŹt until the final verb wait: Reinforcement learning for simultaneous machine translation,” in Proceedings of the 2014 Conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1342–1352.

[29] J. Gu, G. Neubig, K. Cho, and V. O. Li, “Learning to translate in real-time with neural machine translation,”arXiv preprint arXiv: 1610.00388, 2016.

[30] K. Nguyen, H. Daumé III, and J. Boyd-Graber, “Reinforcement learning for bandit neural machine translation on with simulated human feedback,”arXiv preprintarXiv:1707.07402, 2017.

[31] T. K. Lam, J. Kreutzer, and S. Riezler, “A reinforcement learning approach to interactive predictive neural machine translation,”arXiv preprintarXiv:1805.01553, 2018.

[32] X. Zhang and M. Lapata, “Sentence simplification with deep reinforcement learning,”arXiv preprint arXiv:1703.10931, 2017.

[33] S. Ryan and T. Abekawa, “Framework of automatic text summarization using reinforcement learning,” in Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012, pp. 256–265.

[34] Y. Gao, C. M. Meyer, and I. Gurevych, “April: Interactively learning to summarise by combining active preference learning and reinforcement learning,”arXivpreprint arXiv:1808.09658, 2018.

[35] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, “Deep reinforcement learning for dialogue generation,”arXiv preprint arXiv:1606.01541, 2016.

[36] A.-V. Luong, D. Nguyen, and D. Dinh, "A New Formula for Vietnamese Text Readability Assessment," in the 2018 10th International Conference on Knowledge and Systems Engineering (KSE), 2018, pp.

[37] S. Zeno, T. A. S. Associates, R. T. Millard, and R. Duvvuri, The Educator's Word Frequency Guide. Touchstone Applied Science Associates, 1995.

[38] S T. Yao, L. C. Wei, S. B. Dyson, K. Chang, and Y. Chen, “Leveling L2 texts through readability: combining multilevel linguistic features with the CEFR,” Ae Modern Language Journal, vol. 99, no. 2, pp. 371–391, 2015.

[39] B. J. Esfahani, A. Faron, K. S. Roth, PP. Grimminger, and JC. Luers, “[Systematic readability analysis of medical texts on websites of German university clinics for general and abdominal surgery],” Zentralbl Chir, vol. 141, no. 6, 2016.

[40] A. Schlapbach, F. Wettstein, and H. Bunke, “Estimating the Readability of Handwritten Text - a Support Vector Regression Based Approach,” in Proceedings of the International Conference on Pattern Recognition, IEEE, Tampa, FL, USA, December 2008.

[41] J. M. O. Amp and R. A. R. King, “A Matter of Significance: Can sampling error invalidate cloze estimates of text readability?" [J],” Language Assessment Quarterly, vol. 7, no. 4, pp. 303–316, 2010.

[42] Y. Gecit, "(e) evaluation of high school geography 9 and high school geography 11 textbooks with some formulas of readability,” Educational ences: Theory and Practice, vol. 10, no. 4, pp. 2205–2220, 2010.

[43] R. G. Benjamin, “Reconstructing readability: recent developments and recommendations in the analysis of text difficulty,” Educational Psychology Review, vol. 24, no. 1, pp. 63–88, 2012.

[44] U.-V. Marti and H. Bunke, “Using a statistical language model to improve the performance of a hmm-based cursive handwriting recognition system,” Series in Machine Perception and Artificial Intelligence, vol. 15, no. 1, pp. 65–90, 2001.

[45] D. Jones, E. Gibson, W. Shen, et al., “Measuring human readability of machine-generated text: three case studies in speech recognition and machine translation,” in Proceedings of the (ICASSP ‘05)IEEE International Conference on Acoustics Speech and Signal Processing 2005, IEEE, Philadelphia PA USA, March 2005.

[46] R. Shams and R. E. Mercer, “Classifying spam emails using text and Features,” in Proceedings of the IEEE International Conference on Data Mining, pp. 657–666, IEEE, Dallas TX USA, December 2013.

[47] Y. X. Sun, K. Y. Chen, L. Sun, and C. Hu, “Attention-based deep learning model for text readability evaluation,” in Proceedings of the International Joint Conference on Neural Networks (IJCNN) held as part of the IEEE World Congress on Computational Intelligence (IEEE WCCI), by IEEE, Glasgow, UK, July 2020.

[48] J. H. Lee and Y. Hasebe, "Readability measurement for Japanese text based on leveled corpora," Papers on Japanese Language from an Empirical Perspective, Ljubljana: Academic Publishing Division of the Faculty of Arts, University of Ljubljana, 2016.

[49] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016 (pp. 770–778).

[50] Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J. Google's neural machine translation system: bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144. 2016 Sep 26.

[51] Schlichtkrull M, Kipf TN, Bloem P, Berg RV, Titov I, Welling M. Modeling relational data. In European Semantic Web Conference 2018, June 3 (pp. 593-607). Springer, Cham.

[52] Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X, Residual attention network for image classification In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017 (pp. 3156-3164).

[53] Đ. T. N. Nguyễn, A.-V. Lương, and Đ. Đinh, "Affection of the part of speech elements in Vietnamese text readability," Acta Linguistica Asiatica, vol. 9, no. 1, 01/30 2019.

[54] Al-Tamimi AK, Jaradat M, Al-Jarrah N, Ghanem S. AARI: automatic Arabic readability index. Int. Arab J. Inf. Technol.. 2014 Jul 1;11(4):370-8.

[55] Benjamin RG. Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. Educational Psychology Review. 2012 Mar;24(1):63-88.

[56] Cha M,Gwon Y, and Kung HT. Language modeling by clustering with word embeddings for text readability assessment. InProceedings of the 2017 ACM Conference on Information and Knowledge Management, Nov. 6, 2017 (pp. 2003–2006).

[57] La L, Wang N, Zhou DP. Improving reading comprehension step by step using the Online-Boost text readability classification system. Neural Computing and Applications 2015 May;26(4):929-39.

[58] Bessou S, Chennai G. Efficient Measuring of Readability to Improve Documents' Accessibility for Arabic Language Learners. arXiv preprint arXiv:2109.08648. 2021 Sep 9.

[59] Falkenjack J, Mühlenbock KH, Jönsson A. Features indicating readability in Swedish text. InProceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013), May 2013, pp. 27–40.

[60] Rahman MM, Siddiqui FH. An optimized abstractive text summarization model using peephole convolutional LSTM. Symmetry. 2019 Oct 14;11(10):1290.

[61] Preethi V. Survey on text transformation using bi-lstm in natural language processing with text data. Turkish Journal of Computer and Mathematics Education (TURCOMAT). 2021 Apr 24;12(9):2577-85.

[62] Azpiazu IM, Pera MS. Multiattentive recurrent neural network architecture for multilingual readability assessment. Transactions of the Association for Computational Linguistics. 2019 Aug 1;7:421-36.

[63] Ströbel M, Kerz E, Wiechmann D, Qiao Y. Text genre classification is based on linguistic complexity contours using a recurrent neural network. In MRC@ IJCAI 2018 Jan 1.

[64] Moró A, Szaszák G. A prosody-inspired RNN approach for the punctuation of machine-produced speech transcripts to improve human readability. In 2017, the 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom) 2017 Sep 11 (pp. 000219–000224). IEEE.

[65] Martinc M, Pollak S, Robnik-Šikonja M. Supervised and unsupervised neural approaches to text readability. Computational Linguistics. 2021 Apr 21;47(1):141-79.

[66] A. N. Gorban, E. M. Mirkes, and I. Y. Tukin, “How deep should be the depth of convolutional neural networks: a backyard dog case study,” Cognitive Computation, vol. 12, no. 1, pp. 388–397, 2020.

[67] BRITT, M. A, ROUET, J. E, GEORGI, M. A., & PERFETTI, C. A. (1994). Learning from historical texts: From causal analysis to argument models. In G. Leinhardt, I. L. Beck, & C. Stainton (Eds.), Teaching and learning in history (pp. 47-84). Hillsdale, NJ: Erlbaum.

[68] FOLTZ, P.w., KiNTSCH, w., & LANDAUER, T. K. (1993, July). An analysis of textual coherence using latent semantic indexing. A paper presented at the Third Annual Conference of the Society for Text and Discourse, Boulder, CO.

[69] KiNTSCH, W., & VIPOND, D. (1979). Reading comprehension and readability in educational practice and psychological theory. In L.-G. Nilsson (Ed.), Perspectives on memory research (pp. 329-365). Hillsdale, NJ: Erlbaum.

[70] KINTSCH, W., & KEENAN, J. (1973). Reading rate and retention as a function of the number of propositions in the base structure of sentences Cognitive Psychology, 5, 257-274.

[71] Mc NAMARA, D. S., KiNTSCH, E., Songer, N. B., & KiNTSCH, W. (1996). Text coherence, background knowledge, and levels of understanding in learning from text. Cognition & Instruction, 14, 1-44.

[72] BRITTON, B. K., & GULGOZ, S. (1991). Using Kintsch's computational model to improve instructional text: Effects of repairing inference calls on recall and cognitive structures. Journal of Educational Psychology, 83, 329-345.