Language Understanding $\lambda(\mu)$ Laboratory

Language Understanding Laboratory, Myanmar

Summer Internship (2025) Report

Readability for Myanmar Language: Dataset Creation and Grade-Level Classification Models

Submitted by

Readability Team

Under the supervision of

Prof. Ye Kyaw Thu

Language Understanding (LU) Lab, Myanmar

National Electronic & Computer Technology Center (NECTEC), Thailand

With mentorship from

Hlaing Myat Nwe	**Khaing Hsu Wai**	**Thura Aung**
JAIST & SIIT	Akita University, Japan	KMITL, Thailand

Team Members

Kaung Khant Si Thu
Assumption University of Thailand, Thailand

Hsu Yee Mon
Chiang Mai University, Thailand

Seng Pan
Sirindhorn International Institute of Technology, Thammasat University, Thailand

Thiha Nyein
Rangsit University, Thailand

Yu Myat Moe
Rangsit University, Thailand

Abstract

Readability classification is about measuring how easy or difficult a text is to understand. While a variety of methods and tools have been developed for English and other major languages, research on the *Myanmar language* remains very limited. Since Myanmar is a low-resource language with unique writing characteristics, the development of a readability classification system is both necessary and timely. It will provide a foundation for future research in Myanmar language processing and support broader applications in linguistics and natural language understanding.

This internship project was carried out at the *Language Understanding Laboratory, Myanmar*, with the goal of developing a *Myanmar readability classification system*. The internship provided students with practical experience in *natural language processing (NLP)* and *machine learning*, covering the main stages of dataset construction, model design, and performance evaluation under the guidance of mentors.

The work included *data collection and annotation*, where Myanmar texts from textbooks and online resources were labeled according to grade-level readability, creating the dataset for training and testing. We then applied *preprocessing and feature extraction*, adapting tokenization, normalization, and sentence segmentation for Myanmar text, and using features such as word frequency, sentence length, and syllable counts to represent text complexity. Next, *machine learning models* including logistic regression, SVM, and decision trees were trained and compared to identify effective approaches for readability prediction in a low-resource setting. Finally, we conducted *evaluation and validation*, assessing performance on both readability score prediction and grade-level classification. The results indicated that some models achieved promising outcomes even with limited training data, confirming the feasibility of readability classification for Myanmar.

Contents

Chapter 1

Introduction

1.1 Background

Readability assessment has long been recognized as a key component in education and language learning. It provides a means of determining whether a given text is appropriate for the linguistic proficiency of its intended audience. Accurate readability measurement enables the design of effective curricula, the development of tailored learning materials, and the improvement of reading comprehension support systems.

While numerous readability formulas and computational models have been developed for English and other widely studied languages, research on *Myanmar* remains scarce. Myanmar presents unique challenges for automatic readability assessment due to its complex orthography, syllable-based script, and limited availability of annotated corpora. As a result, there are few reliable tools that can automatically assess and categorize text difficulty for Myanmar learners.

This internship project addresses this gap by developing a *Myanmar Text Readability Classification System*. The system is designed to categorize texts into four educational levels: Primary, Lower Secondary, Upper Secondary, and Advanced. By doing so, it establishes a foundation for future research in Myanmar language processing and contributes to the broader field of readability studies in low-resource languages.

1.2 Related Works

Text readability research has progressed from traditional formulas to data-driven methods, and more recently to evaluating Large Language Models (LLMs). A key foundation of this research is the development of corpora aligned with readability levels, often derived from resources such as school textbooks, graded readers,

or categorized texts like simplified newspapers, combined with human judgments of difficulty [5]. Early measures, such as the Flesch Reading Ease [3], estimated readability using simple surface features like sentence length and word or character counts. Other formulas, such as Dale–Chall [2], extended this approach by incorporating lists of easy words to better capture lexical difficulty. While straightforward to apply, these formulas were primarily designed for English, which limits their usefulness for other languages—for example, Thai, where word segmentation poses challenges for readability assessment [1].

More recent approaches use linguistic analysis and machine learning. Examples include regression models for Japanese using sentence length and word origins [4], and SVM classifiers for Swedish [5] and Thai [1] that leverage syntactic and lexical features. The LLM Readability Benchmark [1] evaluates how well LLMs replicate human assessments of readability. Early results show that smaller open-source models, such as Google's Gemma2, can perform competitively with larger proprietary systems like GPT-4o, marking a new direction in readability research.

1.3 Objectives

The objective of this internship project was to develop a *Myanmar language readability classification system*. This work aimed to address the lack of resources and tools for estimating text difficulty in the Myanmar language, which is a low-resource language with unique writing characteristics.

The main objectives of the project are to:

- gain practical experience in *natural language processing (NLP)* and *machine learning*,
- construct an annotated dataset of Myanmar texts labeled by readability levels,
- investigate preprocessing strategies and feature extraction methods for Myanmar text,
- design and evaluate machine learning models for readability prediction, and
- establish a foundation for future research in Myanmar language processing.

1.4 Report Structure

This report is organized as follows.

- Chapter 1 introduces the background, objectives, and structure of the report.
- Chapter 2 describes the dataset preparation process, including collection, preprocessing, and cleaning.

[1] `https://github.com/alexteghipco/LLMReadabilityBenchmark`

- Chapter 3 presents the experiments, covering feature extraction methods, model development, and evaluation procedures.
- Chapter 4 discusses the results, provides analysis and observations, and highlights the limitations of the current approaches.
- Chapter 5 outlines possible directions for future research and development.
- Chapter 6 concludes the report with a summary of the work and its contributions.

Chapter 2

Dataset Preparation

Developing a Myanmar language readability classification system requires a carefully curated and systematically processed dataset. This chapter describes the steps involved in preparing the dataset, including data collection from educational resources, preprocessing of the text, and thorough the data cleaning. The process ensured that the final corpus was well-organized, annotated, and ready for readability analysis.

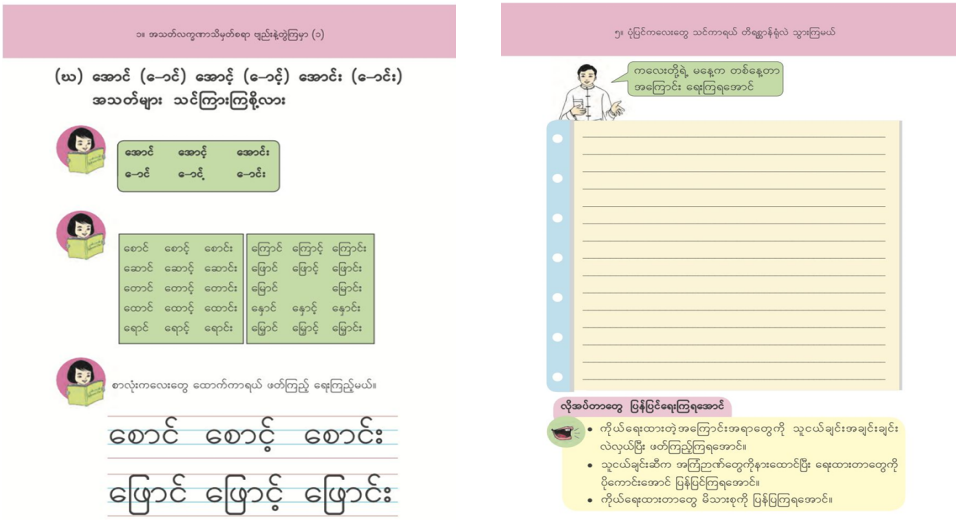
2.1 Data Collection

The data collection process began with a survey of available Myanmar-language educational resources to identify suitable text materials. After an extensive review and comparison of multiple online platforms, the website `https://edu4mm.com/text-books/` was selected as the primary source. This platform provides a comprehensive set of Myanmar curriculum textbooks of high quality and full grade-level coverage.

The key factors for selecting this source included:

- Curriculum alignment: Materials closely follow the official Myanmar education curriculum.
- Content authenticity: Texts are from credible, authoritative educational publications.
- Accessibility: Resources are readily available in digital format.
- Grade-level coverage: The site offers textbooks spanning all intended grade levels.

Ma Khaing Hsu Wai takes the lead in sourcing and downloading textbooks across different educational levels. A systematic approach is used to ensure representative sampling of texts from all educational levels. The collection strategy focuses on four distinct readability categories aligned with the Myanmar educa-



(a) Grade 2 (b) Grade 3

Figure 2.1: Sample textbook pages from Grade 2 and 3 educational materials.

tional framework:

1. Primary Level (မူလတန်းအဆင့်) - Grades 1-5
2. Lower Secondary Level (အလယ်တန်းအဆင့်) - Grades 6-9
3. Upper Secondary Level (အထက်တန်းအဆင့်) - Grades 10-12
4. Advanced Level (တက္ကသိုလ်အဆင့်) - University level

Figure 2.1 shows sample pages from Grade 2 and Grade 3 textbooks, illustrating the type of introductory material collected.

2.1.1 Collection Methodology and Quality Assurance

A comprehensive collection strategy is implemented to ensure balanced representation at all education levels. The team establishes specific selection criteria for choosing textbooks, including curriculum alignment, content authenticity, up-to-date publication relevance, and pedagogical appropriateness for the target grade. Each candidate textbook is carefully reviewed to verify its suitability for readability analysis and its alignment with the established standards.

A total of 25 textbooks containing 1,679 pages across diverse educational levels, is collected to ensure balanced representation in the corpus. This diversity ensures that the corpus includes various types of discourse and subject-specific vocabularies commonly encountered in Myanmar educational texts. By creating a broad and representative dataset, the team establishes a strong foundation for

building the readability assessment system.

2.2 Preprocessing

The preprocessing step involves extracting text from the textbook PDFs using optical character recognition (OCR), adding linguistic annotations, segmenting the text into meaningful units, performing manual corrections, and organizing the data into a structured format. The following subsections detail each of these preprocessing steps.

2.2.1 OCR Processing

Mg Kaung Khant Si Thu enhances an existing command-line interface (CLI) OCR tool, originally developed by Mg Kaung Hset Hein to extract the text from scanned textbook pages. The original tool utilizes the Google Drive API to perform batch OCR on images. The enhancement involved designing and implementing a graphical user interface (GUI) and integrating additional functionality to improve usability and performance, making the tool accessible to non-technical users.

The improved OCR application is developed in Python. The following libraries and frameworks are used:

- **tkinter** – to implement the GUI, providing a simple user-friendly interface.
- **PIL** (Pillow) – for image loading, processing, and previewing of pages.
- **threading** – to enable non-blocking GUI operations, ensuring a responsive user experience.
- **queue** – for thread-safe data communication between concurrent processes.
- **pdf2image** – for the direct conversion of PDF files into suitable image formats for OCR processing.

By integration a GUI and these supporting libraries, significantly extended the functionality of the original CLI-based tools, improving accessibility for non-technical users while maintaining the batch OCR capabilities provided by the Google Drive API. The OCR application successfully processed all 25 textbooks, extracting raw text data from 1,679 pages at different grade levels.

2.2.2 Data Labeling

After OCR extraction, the next step is to organize the data into a structured format for analysis. Ma Hlaing Myat Nwe designed the structure of the dataset, which included four essential columns for each record:

- Title: an identifier for the source document or text (e.g., textbook name and page number).

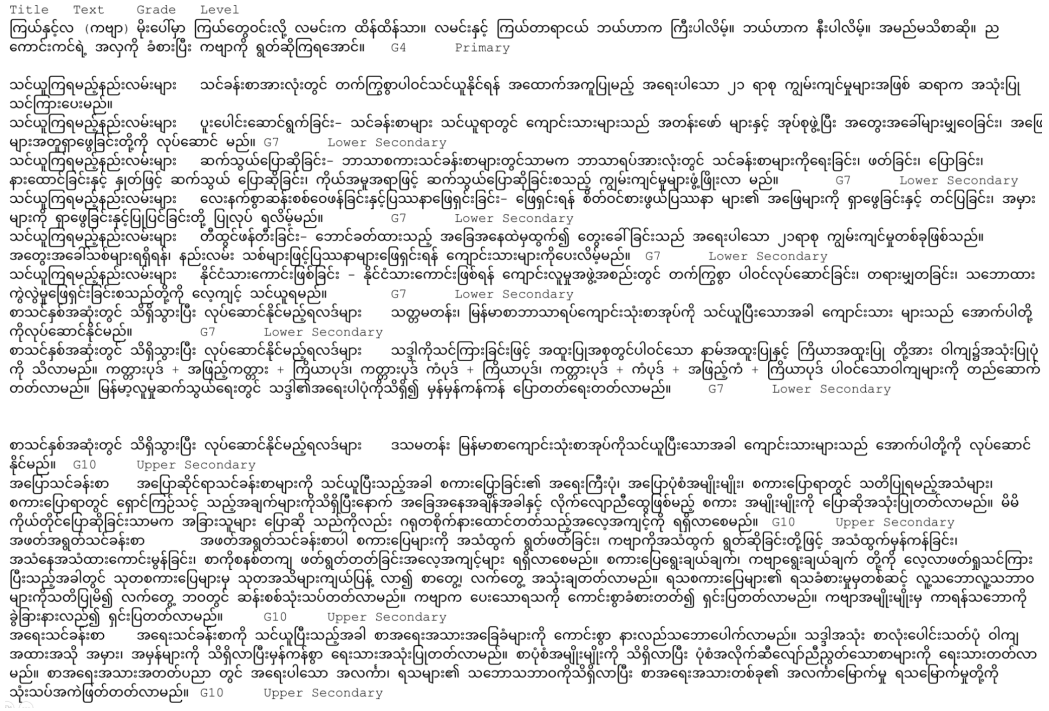


Figure 2.2: Sample data labeling

- Text: the processed text content of the segment (paragraph or sentence).
- Grade: the specific grade level of the material (1 through 12, corresponding to the school grade or an equivalent level).
- Level: the educational category of the text (Primary, Lower Secondary, Upper Secondary, or Advanced).

The finalized dataset was stored in TSV format, ensuring consistency and ease of use for subsequent analysis. An example of the labeling format is shown in Figure 2.2.

2.2.3 Text Segmentation and Manual Verification

Following data annotation, the team applies text segmentation and conducted manual reviews to ensure the quality of the transcribed text. This stage involves two major steps:

1. Word Segmentation: The raw text is segmented into words using established Myanmar language segmentation rules. Here, we use the myWord tool[1]. This step produced initial word-separated text that could be further analyzed.

[1]https://github.com/ye-kyaw-thu/myWord

2. Manual Verification Process: We systematically review the segmented OCR output to improve accuracy. We check each portion of text to:
    - Correct OCR errors: verify proper spelling and accurate character recognition, making correction where needed.
    - Ensure completeness: confirm that no paragraphs or sections were missing from the OCR output.
    - Fix encoding issues: resolve any encoding problems or formatting inconsistencies (such as garbled characters or incorrect line breaks).
    - Validate segmentation: make sure word boundaries and sentence breaks were correctly applied during automatic segmentation.

Through this meticulous manual verification process, we correct many errors that OCR process or segmentation had introduced. This ensures that the textual data closely matched the content of the original textbooks, in both accuracy and structure.

2.2.4 POS Tagging Integration

An important part of preprocessing is the integration of Part-of-Speech (POS) tagging to enrich the text with linguistic information. The myPOS RDR model (a Myanmar-language POS tagging tool) is applied to the cleaned and segmented text. This tagging step identifies the grammatical category of each word (such as noun, verb, adjective, particle, etc.), providing valuable syntactic and lexical features relevant to readability, for example, ratios of different POS categories in a sentence.

2.3 Data Cleaning

Once the dataset is structured, we carry out a thorough data cleaning phase to eliminate any remaining errors and inconsistencies. During this phase, the previously segmented and verified text is revisited to ensure maximum precision and uniformity. This step involves tasks such as spelling corrections, completing text segments, consistent segmentation structure.

After this comprehensive cleaning, the dataset is reliable, well-organized, and ready for analysis. All text records were correctly labeled with their corresponding grade and level, and the content of each record accurately reflected the original material. This ensured that downstream processes such as feature extraction, model training, and evaluation could be performed on a high-quality corpus without being skewed by data errors or inconsistencies.

2.3.1 Identifying and Removing Cross-Level Sentences

As a final cleaning step, we examine the dataset for any sentences that might be misclassified or duplicated across the different reading levels. Even subtle data issues can affect model performance, so the following checks are performed to uphold data integrity:

- Duplicate sentences across levels: If the same sentence appears in multiple level categories, the duplicate occurrence in the higher level is removed. This ensure that each readability level's subset contain only unique sentences not found in other levels, maintaining mutual exclusivity between levels.
- Anomalous entries (word lists): Any entries that are not natural sentences - for instance, records that turn out to be mere lists of words or abbreviations extracted from the textbooks - are removed. Such items are considered out of scope for readability scoring, since they do not reflect typical continuous text.
- Mix-level sentences: Within the lower secondary and upper secondary data sets, the team flagged sentences that are longer than five words but contained no stacked consonants, no complex stacked characters in Burmese script. If a sentence of that length had no stacked letters - an indicator of simpler vocabulary or structure - it is judged to be easier that expected for those levels. Such sentences are removed from the secondary-level sets to prevent mixing of easier than intended content into higher difficulty categories.

By identifying and removing these out-of-place sentences, we ensure that each level of the dataset is internally consistent and truly representative of its intended difficulty. These data cleaning measures strengthened the quality of the final dataset, which could now be confidently used for feature extraction, model training, and evaluation in the developing of the Myanmar readability assessment system.

Chapter 3

Experiments

3.1 Feature Extraction

Feature extraction in this study is grounded in the multilevel linguistic approach described by [7], which emphasizes the integration of lexical, syntactic, semantic, and structural features to model text readability. By processing the preprocessed TSV files, a diverse set of linguistic features is computed to capture the complexity of Burmese texts across multiple dimensions.

This multilevel methodology enables a more comprehensive assessment of text difficulty, reflecting how various linguistic factors interact to influence readability. The extracted features serve as the foundation for subsequent machine learning analyses, aligning with best practices in readability research [7].

3.1.1 Lexical Features

Lexical features describe properties of individual words and their distribution within the text. These indicators reflect the variety, length, and frequency of words used, which strongly influence perceived readability.

- **AvgWordLen**: The mean number of characters per word. Longer words in Burmese often correspond to more complex concepts or morphological structures, making this feature a useful proxy for lexical difficulty.
- **SyllablePerWord**: The average number of syllables per word. Since Burmese syllables can carry tones and meaning units, words with more syllables may indicate a higher cognitive load for readers.
- **WordCount**: The total number of words in a passage. Word count provides an overall measure of text length, which can influence how demanding the passage is to process.
- **RareWordRatio**: The proportion of words occurring no more than two

times in the reference corpus. A higher ratio suggests that the text contains less familiar or domain-specific vocabulary, which may lower readability for general readers.

- **CommonWordRatio**: The proportion of words drawn from a curated list of high-frequency Burmese words. This feature highlights the extent to which the text relies on commonly used vocabulary, which generally improves accessibility.
- **SyllableCount**: The total number of syllables across the entire passage. This metric complements word count by capturing overall phonological load and providing a measure of text length in spoken units.

3.1.2 Sentence and Structural Features

These features capture sentence-level complexity and structural markers of Burmese orthography. Sentence length and structural patterns are often linked to the syntactic complexity of a passage.

- **AvgSentLen**: The mean number of words per sentence, where sentences are delimited by the punctuation marker "။". Longer sentences may involve more clauses and dependencies, reflecting greater syntactic complexity.
- **SentCount**: The number of sentences in a passage. Alongside average length, this helps to distinguish between texts that are concise with many short sentences versus those with fewer, more extended sentences.
- **StackedWordRatio**: The proportion of words containing stacked consonants (consonant clusters written vertically in Burmese script). Such orthographic features often appear in more advanced vocabulary and may pose an additional decoding challenge for readers.

3.1.3 Part-of-Speech Features

Part-of-speech (POS) features are extracted using myPOS[1] model. These features quantify the grammatical composition of the text and indicate the relative distribution of functional and content words, which are central to syntactic and semantic interpretation.

- **VerbCount**, **NounCount**, **AdverbCount**, **AdjCount**, **ParticleCount**, **PronounCount**, **ConjunctionCount**: The absolute frequencies of each POS category. For example, a higher noun or verb count suggests a focus on entities and actions, while frequent particles or conjunctions indicate more complex sentence linking.

[1] https://github.com/ye-kyaw-thu/myPOS

- **TotalPOSWords**: The total number of words that received a POS tag. This ensures consistency in interpreting ratios and densities of POS categories.
- **ContentWordCount**: The number of content words (verbs, nouns, adjectives, adverbs). Content words carry the bulk of semantic meaning, and their proportion relative to total words serves as an indicator of lexical richness and conceptual density.

3.1.4 Stopword Features

Stopword-based features are derived using myStopword[2] list, which contains common Burmese function words such as particles and connectors. While individually uninformative, their distribution offers insight into text flow and grammatical style.

- **StopwordRatio**: The proportion of stopwords relative to the total word count. High ratios often indicate simpler sentence structures with more function words guiding cohesion, whereas lower ratios may suggest dense, information-heavy text.
- **ContentWordDensity**: The proportion of content words compared to the total words. This measure complements stopword ratio by quantifying the amount of semantically rich vocabulary per unit of text.
- **StopwordCount**: The raw number of stopwords in the text. Useful as a simple length-normalized measure when comparing across passages of varying sizes.
- **ContentRareWordRatio**: The fraction of rare words (frequency ≤ 2 in corpus) occurring specifically among content words. This highlights texts that use uncommon, conceptually important vocabulary, which may significantly affect comprehension difficulty.

3.1.5 Grade-Level Word Ratios

To model readability in an educational context, features are also derived from grade-specific vocabulary lists from Grade 1–12 (G1-12). Each list contains words that predominantly appear in textbooks of that grade level.

- **G1_WordRatio** to **G12_WordRatio**: Each feature represents the proportion of words in the passage that belong exclusively to the vocabulary of a particular grade. For instance, a high G1 ratio indicates reliance on simple, foundational words, while a high G10 or G12 ratio suggests advanced vocabulary appropriate for higher-grade texts. These features help in aligning the measured difficulty of a passage with the expected reading level of

[2]https://github.com/ye-kyaw-thu/myStopword

Burmese students.

3.1.6 Example

To illustrate the computation of these features, consider the following paragraph which has been manually segmented to match the input format required by feature extraction methods which ensures that each word can be accurately processed for lexical, sentence & structural, part of speech, and stopword features calculation.

Paragraph: လက်ဖက် ဆိုသည်မှာ ကစော်ဖောက် ထားသော သို့မဟုတ် အချဉ်ဖောက် ထားသော လက်ဖက်ပင် ၏ အရွက် ဖြစ်သည် ။ မြန်မာ နိုင်ငံ သည် ကျ် ဒေသ တွင် သာ သီးသန့် တွေ့ရှိ ရသော အစားအစာ တစ်မျိုး ဖြစ်သည့် လက်ဖက် ကို သောက်စရာ အဖြစ် သာမက စားသောက်ကုန် အဖြစ် ပါ သုံးဆောင် ကြသည့် နိုင်ငံ အနည်းငယ် ထဲ မှ တစ်နိုင်ငံ ဖြစ်သည် ။ လက်ဖက် သည် မြန်မာ့ လူ့အဖွဲ့အစည်း တွင် အရေးပါသော ကဏ္ဍ မှ ပါဝင် သည့် အမျိုးသား အစားအစာ တစ်ခု အဖြစ် သတ်မှတ် ခံရ ပြီး၊ အိမ် သို့ အလည်လာသော ဧည့်သည် များ ကို တည်ခင်း ဧည့်ခံ သည့် မြန်မာ့ ရိုးရာ ဧည့်ဝတ် ပြုမှု အထိမ်းအမှတ် တစ်ခု အဖြစ် လည်း ဆက်လက် တည်ရှိ နေသည် ။

Noun: လက်ဖက်၊ လက်ဖက်ပင်၊ အရွက်၊ မြန်မာ၊ နိုင်ငံ၊ ဒေသ၊ အစားအစာ၊ တစ်မျိုး၊ စားသောက်ကုန်၊ နိုင်ငံ၊ တစ်၊ မြန်မာ့လူ့အဖွဲ့အစည်း၊ ကဏ္ဍ၊ အမျိုးသား၊ အစားအစာ၊ တစ်ခု၊ အိမ်၊ ဧည့်သည်၊ ရိုးရာ၊ ဧည့်ဝတ်ပြုမှု၊ အထိမ်းအမှတ်။

Verb: ကစော်ဖောက်၊ အချဉ်ဖောက်၊ ဖြစ်သည်၊ တွေ့ရှိ၊ သုံးဆောင်၊ ပါဝင်၊ သတ်မှတ်၊ အလည်လာ၊ တည်ခင်း၊ ဧည့်ခံ၊ တည်ရှိ၊ နေသည်။

Adjective: အရေးပါသော၊ အလည်လာသော။

Adverb: သီးသန့်၊ ဆက်လက်။

Pronoun: ကျ်

Conjunction: သို့မဟုတ်၊ ပြီး၊ လည်း။

Particle: ဆိုသည်မှာ၊ ထားသော၊ ၏၊ သည်၊ တွင်၊ သာ၊ ရသော၊ ဖြစ်သည့်၊ ကို၊ အဖြစ်၊ သာမက၊ ပါ၊ ကြသည့်၊ ထဲ၊ မှ၊ ပြီး၊ သို့၊ များ၊ လည်း။

The numerical values of these readability features are presented in Table 3.1.

3.1.7 Feature Extraction Pipeline and Dataset Overview

The Feature Extractor notebook provides the full pipeline for transforming raw Burmese text into a structured set of linguistic features. It loads segmented and POS-tagged corpus data, applies stopword filtering, computes grade-level vocabulary ratios, and extracts lexical, syntactic, and semantic features for each text. This automated process ensures consistency and reproducibility, resulting in the final

Table 3.1: Features of Sample Burmese Text.

Feature	Value
Lexical Features	
Average Word Length	5.60
Syllables Per Word	2.07
Word Count	82
Syllable Count	170
Rare Word Ratio	0.32
Common Word Ratio	0.45
Syntactic Features	
Average Sentence Length	20.50
Sentence Count	4
Stacked Word Ratio	0.01
Part-of-Speech (POS) Counts	
Content Word Count	38
Noun Count	22
Verb Count	12
Adjective Count	2
Adverb Count	2
Pronoun Count	1
Conjunction Count	3
Particle Count	18
Total POS Words	70
Grade Level Word Ratios	
Grade 1 Word Ratio	0.34
Grade 2 Word Ratio	0.15
Grade 3 Word Ratio	0.11
Grade 4 Word Ratio	0.02
Grade 5 Word Ratio	0.01
Grade 6 Word Ratio	0.04
Grade 7 Word Ratio	0.05
Grade 8 Word Ratio	0.00
Grade 9 Word Ratio	0.00
Grade 10 Word Ratio	0.00
Grade 11 Word Ratio	0.05
Grade 12 Word Ratio	0.00
Stopword Metrics	
Stopword Count	20
Stopword Ratio	0.24
Content Word Density	0.76
Content Rare Word Ratio	0.42

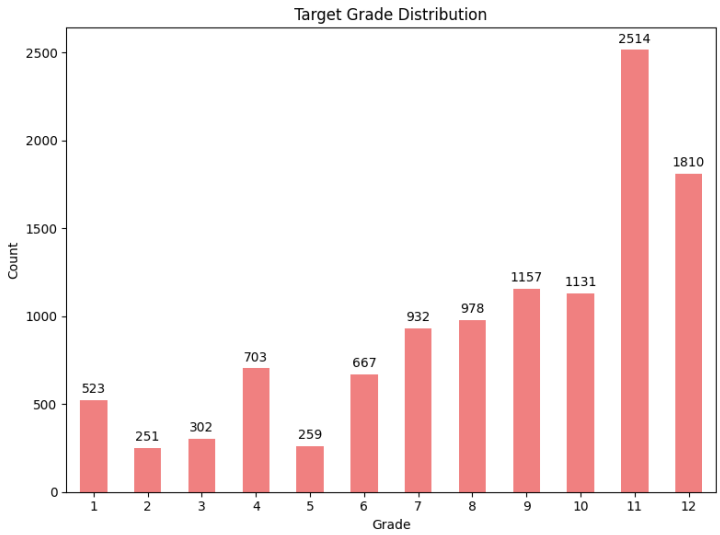


Figure 3.1: Grade Distribution

dataset, `readability_features.csv`, which captures the linguistic complexity of each sample and is ready for downstream analysis and machine learning.

The chart in Figure 3.1 displays the distribution of text samples across grade levels in the dataset used for readability scoring experiments. It reveals a significant imbalance, with higher grades (especially Grade 11 and 12) containing substantially more samples than lower grades. This skewed distribution may influence model training and evaluation, potentially making the classifier more accurate for grades with more data. When interpreting results from SVM and regression models, it is important to consider this imbalance, as it can affect both overall accuracy and per-grade performance.

3.2 Regression Models

The section on regression models provides a clear baseline for readability research, contrasting traditional regression approaches with modern transformer-based models. Regression models, which utilize interpretable and handcrafted features, offer transparency in estimating Burmese readability. The experiment aims to determine whether these traditional techniques can adequately capture Burmese readability patterns.

3.2.1 Experimental Setup

In the experimental setup, five regression models were evaluated:

Table 3.2: Base Regression Models Performance Comparison

	MAE	RMSE	MSE	MedAE	R^2	Expl.Var	MaxErr	Duration
RandomForest	**0.84**	**1.44**	**2.07**	**0.34**	**0.80**	**0.80**	8.57	5.04
LinearRegression	1.91	2.40	5.76	1.61	0.44	0.44	11.30	**0.03**
Ridge	1.91	2.40	5.76	1.61	0.44	0.44	11.25	0.05
ElasticNet	2.41	2.92	8.54	2.17	0.17	0.17	**7.47**	0.05
SVR	2.30	3.08	9.46	1.80	0.08	0.14	10.56	1.62

Table 3.3: GridSearchCV-utilized Regression Models Performance Comparison

	MAE	RMSE	MSE	MedAE	R2	Expl.Var	MaxErr	Duration
SVR	1.16	1.68	2.83	0.72	0.72	0.72	**8.27**	**128.8**
RandomForest	**0.84**	**1.43**	**2.05**	**0.35**	**0.80**	**0.80**	8.35	137.0

1. **Linear Regression:** serves as the simplest baseline, assuming linear relationships between features and readability.
2. **Ridge Regression:** incorporates L_2 regularization to address multicollinearity among linguistic features.
3. **ElasticNet:** combines L_1 and L_2 penalties, balancing feature selection and coefficient shrinkage.
4. **Support Vector Regression (SVR):** explores nonlinear mappings via kernel functions, aiming to capture more complex relationships.
5. **Random Forest Regressor:** an ensemble of decision trees, robust to nonlinearities and feature interactions.

All models were implemented using scikit-learn [3] and trained under identical conditions, with an 80/20 train-test split.

- **R-squared (R^2):** proportion of variance in readability scores explained by the model.
- **Mean Absolute Error (MAE):** average magnitude of prediction error.
- **Root Mean Squared Error (RMSE):** penalizes larger errors, providing a stricter evaluation.
- **Additional metrics:** Median Absolute Error (MedAE), Maximum Error (MaxErr), and training duration.

3.2.2 Results and Comparison

Table 3.2 summarizes the performance of these baseline models across several metrics, including MAE, RMSE, R^2, and training duration.

To further optimize model performance, GridSearchCV was applied to SVR and Random Forest, as shown in Table 3.3. This table highlights the improve-

[3] https://scikit-learn.org/stable/

ments achieved through hyperparameter tuning, particularly for SVR, which saw a notable reduction in error metrics.

The results indicate that the Random Forest Regressor consistently outperformed other models, achieving the lowest MAE and RMSE, and the highest R^2 score (0.80), as detailed in Table 3.2. In contrast, Linear Regression and Ridge Regression showed moderate performance, while ElasticNet and SVR lagged behind. Table 3.3 further demonstrates that, even with hyperparameter optimization, Random Forest remains the most effective model.

3.2.3 Analysis and Observations

Several insights emerge from the regression experiments:

- **Feature effectiveness:** sentence length and syllable counts were strong predictors of readability, aligning with traditional readability indices in other languages. However, their linear contribution alone was insufficient for accurate modeling.
- **Linear vs. nonlinear performance:** Random Forest's superior results highlight the importance of modeling nonlinear feature interactions. The strong gap between Random Forest ($R^2 = 0.80$) and Ridge/Linear Regression ($R^2 = 0.44$) suggests that linear assumptions are too restrictive for Burmese readability prediction.
- **ElasticNet vs Ridge:** ElasticNet performed worse despite feature selection, suggesting that feature sparsity is less critical than regularization when handling Burmese linguistic features.
- **SVR limitations:** SVR struggled despite its nonlinear capabilities, likely due to sensitivity to hyperparameters and difficulties in scaling Burmese text features.
- **Burmese-specific challenges:** the lack of explicit word boundaries and stacked syllable structure complicate feature engineering, making handcrafted features less expressive compared to models that can capture contextual meaning.

3.2.4 Limitations of Regression Approaches

While regression models provide interpretable baselines, their limitations are evident:

- **Dependence on handcrafted features:** effectiveness is bounded by the quality of manually engineered linguistic statistics.
- **Lack of semantic understanding:** models cannot account for discourse structure, contextual cues, or meaning.

- **Performance ceiling:** accuracy plateaus below what is expected from deep contextual models such as transformers.

Thus, regression serves as a baseline, providing a reference point before introducing more advanced classification and embedding-based approaches, which will be discussed in subsequent sections.

3.3 Support Vector Machine Classifier

In addition to regression approaches, readability assessment can be formulated as a classification problem, where each text is assigned to a discrete grade-based category (e.g., G1–G12). Drawing on the methodology of the previous work [1], Support Vector Machine (SVM) classifiers are utilized to distinguish Burmese texts across multiple readability levels. SVMs are well-suited for high-dimensional linguistic data, and their effectiveness has been demonstrated in Thai text readability research through the use of word segmentation and feature engineering. By leveraging kernel methods and feature scaling, SVMs can robustly separate texts into appropriate readability categories based on extracted linguistic features.

3.3.1 Dataset and Features

The dataset used for this experiment consists of Burmese texts labeled by grade-level readability (G1 to G12). Each text was transformed into a feature vector comprising:

- **Word ratio features:** proportion of grade-specific vocabulary across G1–G12 levels.
- **Surface-level features:** sentence length, token counts, and syllable distributions.
- **Lexical richness features:** type-token ratios and frequency-based statistics.

This enhanced feature set builds upon the regression experiments, with the addition of grade-specific word usage ratios to better discriminate among discrete readability categories.

3.3.2 Experimental Setup

The experimental pipeline included the following steps:

1. **Train/test split:** The dataset was split into training and test sets with stratification to preserve grade distribution.
2. **Feature scaling:** Standardization was applied to ensure comparability of feature magnitudes.

Table 3.4: GridSearchCV-utilized Classification Model Performance Comparison

Model	Test Accuracy	CV	Precision	Recall	F1-score	Duration
SVM	0.84	0.85	0.82	0.85	0.83	103.05

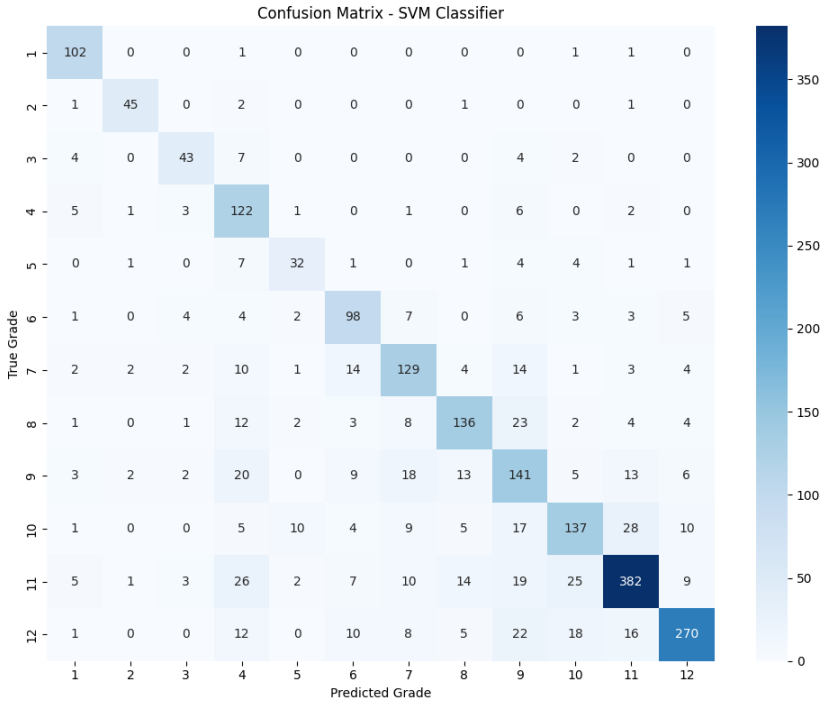


Figure 3.2: Confusion Matrix - SVM

3. **Model training:** A Support Vector Classifier (`SVC`) was trained with hyperparameter tuning to identify the best kernel type (linear, RBF, or polynomial), regularization parameter C, and kernel-specific parameters (e.g., γ).
4. **Evaluation:** Performance was measured using accuracy, precision, recall, F1-score, and confusion matrix analysis.

3.3.3 Results and Comparison

The results of the SVM classifier are summarized in Table 3.4. A confusion matrix in Figure 3.2 illustrates the distribution of predictions across grade levels, and feature correlation plot (Figure 3.3) provides insights into the most predictive attributes.

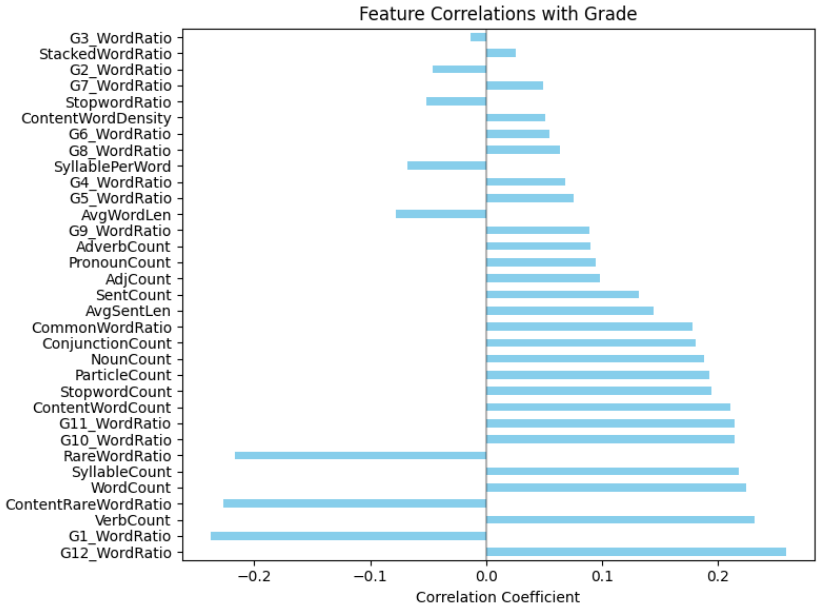


Figure 3.3: Feature Correlations - SVM

3.3.4 Analysis and Observations

- **Grade-level separation:** SVM effectively distinguished between texts at extreme readability levels (e.g., G1 vs. G12), but showed more overlap in mid-level grades (e.g., G6–G8).
- **Impact of word ratio features:** The inclusion of G1–G12 word ratios substantially improved discrimination, indicating that grade-specific vocabulary usage is a strong readability signal.
- **Kernel performance:** The RBF kernel outperformed the linear kernel in capturing nonlinear boundaries between grade clusters.
- **Limitations:** SVMs were computationally more expensive on larger feature sets, and interpretability of feature contributions was lower compared to tree-based models.

3.3.5 Limitations

While the SVM classifier demonstrates strong performance on Burmese readability classification, several limitations were identified:

- **Scalability:** Training time grows significantly with dataset size, especially with nonlinear kernels.
- **Interpretability:** Unlike regression coefficients or tree-based feature importance, SVM decision boundaries are less transparent.

Table 3.5: Accuracy of using SVM model with difference word embedding.

Word Embedding	Model	Accuracy
TF-IDF 1 gram	SVM	0.8112199465716830
TF-IDF 1 and 2 gram	SVM	0.8005342831700801
CBOW	SVM	0.6509349955476402
Skip gram	SVM	0.6767586821015138
Fast text	SVM	0.6260017809439002

- **Granularity of prediction:** Misclassifications tend to cluster around adjacent grades, indicating challenges in distinguishing fine-grained levels of readability.

These results establish SVM as a strong baseline classifier, paving the way for comparison with more advanced models that incorporate word embeddings in subsequent sections.

3.4 Word Embedding

After feature extraction, we tested several word embedding methods on the 'Text' columns, including TF-IDF (1-gram and 1–2 gram), CBOW, Skip-Gram, and FastText. To compare their effectiveness, we trained an SVM model from scikit-learn using the embeddings from each method. Since the text data was manually segmented, we used the split() function for tokenization. TF-IDF embeddings were generated with scikit-learn, while CBOW, Skip-Gram, and FastText were implemented using the gensim library. From the results presented in Table 3.5, it can be observed that TF-IDF with 1-gram achieved the best performance.

3.4.1 Model

We experimented with several classical machine learning models, including Support Vector Machine (SVM), K-Neighbors Classifier (KNN), Random Forest, XGBoost, and Multi-Layer Perceptron (MLPClassifier).

'Combined Dataset' was created by concatenating the PCA-reduced TF-IDF features (20 components) with the numerical features. SVM, KNN, and Random Forest models were trained using their default parameters from the scikit-learn library, while XGBoost was implemented with the xgboost library using its default parameters. For the MLPClassifier, we used scikit-learn with hidden layers of (512, 256, 128, 64, 3), the Adam optimizer, an initial learning rate of 0.01, and a maximum of 20 iterations.

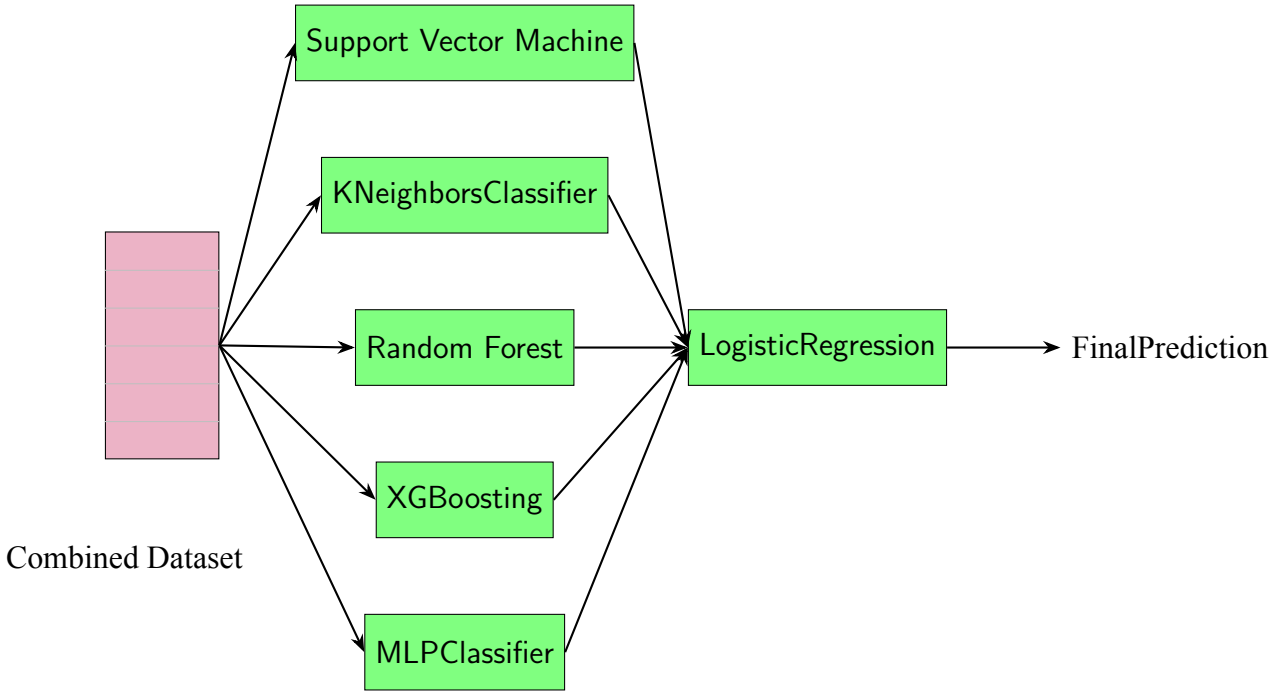


Figure 3.4: A visual representation of the stacking ensemble model.

Finally, we trained a stacking model with Logistic Regression as the final estimator, using the outputs of the above models as inputs as shown in Figure 3.4. The stacking model achieved the best performance among all approaches. The detailed results are presented in Table 3.6.

3.5 Testing Neural Network

We also investigated the performance of neural network models. In this experiment, we trained four types of models: Artificial Neural Network (ANN), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM). To reduce training time, the input dimension was reduced using PCA with 1,000 components, and each model was trained for 10 epochs. Among these models, ANN achieved the best performance on the classification task.

To further analyze the effect of dimensionality reduction, we trained ANN with different PCA settings (100, 500, and 1,000 components) as well as without PCA on the TF-IDF unigram features. The results showed that PCA with 500 components provides performance comparable to using TF-IDF without PCA as shown in Table 3.7.

Table 3.6: Model comparison with stacking ensemble model on Combined dataset.

Model	Accuracy	Precision	Recall	F1-Score
Support Vector Machine	0.896552	0.897894	0.896552	0.896783
KNeighborsClassifier	0.838558	0.843485	0.838558	0.839485
Random Forest	0.911181	0.916013	0.911181	0.911624
XGBoosting	0.924242	0.926847	0.924242	0.924657
MLPClassifier	0.879833	0.893067	0.879833	0.881553
Stacking Model	0.928422	0.929426	0.928422	0.928646

Table 3.7: Evaluation Results of Different Neural Network Models.

Dataset Name	Model	Accuracy	Loss
TF-IDF 1 gram with PCA 1000	ANN	0.808548	0.781864
TF-IDF 1 gram with PCA 1000	CNN	0.769813	1.010523
TF-IDF 1 gram with PCA 1000	RNN	0.477292	1.032233
TF-IDF 1 gram with PCA 1000	LSTN	0.477292	1.032279
TF-IDF 1 gram with PCA 50	ANN	0.667408	0.719744
TF-IDF 1 gram with PCA 100	ANN	0.767141	0.697804
TF-IDF 1 gram with PCA 500	ANN	0.813000	0.744820
TF-IDF 1 gram without PCA	ANN	0.822350	1.119706

The number of rows in training and testing data and number of features used to train models are shown in Table 3.8.

3.6 Shallow Features: Machine Learning Approach

Shallow linguistic features inspired by Thai readability[1] studies for simplicity and effectiveness in analyzing Burmese text. These features help capture key lexical, structural, and cultural elements without requiring complex processing. In the context of readability, “shallow features” are simple, surface-level characteristics of a text that can be calculated easily. These features include metrics like average word length, the ratio of specific word types (like conjunctions), and the frequency of words from a pre-defined list (such as difficult words or grade-level vocabulary). They are termed ’shallow’ because they do not require a deep or complex understanding of grammar (syntax) or meaning (semantics). Instead, they act as powerful indicators of complexity that a machine learning model can learn from.

We chose to test these features in our Burmese readability research for several important reasons:

- **Proven Effectiveness**: The previous Thai study showed that these simple

Table 3.8: Table of Features

Name	Training Count	Testing Count	Features
Numerical Columns	7,656	1,914	34
TF-IDF 1 gram	7,656	1,914	12,330
TF-IDF 1 and 2 gram	7,656	1,914	116,041
CBOW	7,656	1,914	100
Skip gram	7,656	1,914	100
Fast text	7,656	1,914	100
TF-IDF 1 gram with PCA 500	7,656	1,914	500
TF-IDF 1 gram with PCA 20	7,656	1,914	20
Combine Dataset	7,656	1,914	54

features were surprisingly effective in assessing text readability. Since Thai and Burmese have some linguistic similarities (e.g., being tonal languages with complex scripts), this method provides a proven starting point for our research.

- **Computational Simplicity**: Unlike deep methods that require huge amounts of data and powerful computers, shallow features are computationally efficient. They can be extracted quickly and used to train effective models without needing extensive resources, making this a practical and accessible approach.
- **Data Availability**: This method is highly practical for the Burmese language. While it requires a dataset with word segmentation and Part-of-Speech (POS) tags, this level of annotation is much simpler to create than a deep dataset, which would need full grammatical and semantic analysis. Because a POS-tagged corpus is more achievable, this shallow approach allows us to build an effective model without needing the massive, deeply analyzed datasets that are often unavailable for Burmese.

3.6.1 Feature Engineering

Designed shallow features as shown in Table 3.9, that are simple to compute but still capture Burmese text complexity:

Lexical Features

- **Stack Word Ratio**: Ratio of words with stacked consonants to all words, showing complexity.
- **Average Word Length**: Mean number of syllables per word, reflecting lexical difficulty.

Table 3.9: Burmese Shallow Feature

No	Feature	Description
1	stack_word_ratio	ratio of words with stacked consonants to all words
2	avg_word_length	Average number of syllables per word
3	conj_ratio	Ratio of conjunctions in the text
4	ppm_ratio	Ratio of post-positional markers in the text
5	adv_ratio	Ratio of adverbs in the text
6	proverb_ratio	Ratio of proverbs to total words
7-18	G1–G12_WordRatio	Ratio of words unique to each grade level (G1 to G12)
19	Primary_WordRatio	Ratio of words belonging to Primary level (G1-5)
20	LowerSecondary_WordRatio	Ratio of words belonging to Lower Secondary level (G6-9)
21	UpperSecondary_WordRatio	Ratio of words belonging to Upper Secondary level (G10-12)

- **Grade-Level Word Ratios (G1–G12)**: Ratio of words unique to each grade-level word list.
- **Education-Level Word Ratio**: Ratio of words grouped into three broader levels:
  - **Primary_WordRatio (G1-G5)**
  - **Lower Secondary_WordRatio (G6-G9)**
  - **Upper Secondary_WordRatio (G10-12)**

POS Features

- **Conjunction Ratio**: Proportion of conjunctions, indicating sentence connection complexity.
- **Post-positional Marker (ppm) Ratio**: Ratio of post-positional markers, reflecting grammatical structure.
- **Adverb Ratio**: Frequency of adverbs, showing descriptive and modifying usage.

Cultural Features

- **Proverb Ratio**: This feature measures the use of proverbs, which is a strong indicator of cultural and linguistic complexity. A text with more proverbs is typically aimed at a more advanced audience.
  Implementation:
  1. **Create a Corpus**: We first built a master list of unique Burmese proverbs.

2. **Syllable Tokenization**: We broke down each text and each proverb into individual syllables.
3. **Scan and Count**: Our program then scanned each text to count how many proverbs from our lists were present.

Finding: The analysis revealed a clear trend: texts for higher education levels contained significantly more proverbs. As the dataset we used, Upper Secondary texts had the highest count, while Primary texts had almost none. This strong correlation makes the Proverb Ratio a highly effective feature for our readability model.

3.6.2 Experimental Setup

To build and validate our readability models, we established a systematic machine learning pipeline. This ensures our results are reliable and our comparisons between different models are fair and accurate.

Handling Data Imbalance

We used the SMOTE (Synthetic Minority Over-sampling Technique) algorithm. This was a critical step to balance the number of texts for each education level (Primary, Lower Secondary, Upper Secondary). By creating synthetic examples of the minority classes, SMOTE ensures the model receives a balanced view of the data, preventing it from becoming biased toward the most common education level.

Feature Scaling & Selection

To ensure our models performed optimally, we applied two key preprocessing steps:

- **Feature Scaling**: We applied scalers to normalize the range of our features. The primary scalers used were 'StandardScaler', which centers data around a mean of 0 and a standard deviation of 1, and 'RobustScaler', which is less sensitive to outliers. The choice of scaler depended on the characteristics of the feature set being used in a particular model.
- **Feature Selection & Dimensionality Reduction**: With a large number of features, simplifying the model is essential to improve performance and interpretability. We used three distinct methods:
  - **RFE (Recursive Feature Elimination)**: This method iteratively removes the least important features to find the optimal subset that works well together. It was through but computationally intensive.

- **SelectKBest**: This is a faster, statistical approach that selects the top 'k' features based on their individual correlation with the readability level.
- **PCA (Principal Component Analysis)**: Instead of selecting features, PCA reduces dimensionality by transforming the original features into a smaller set of new, uncorrelated "principal components" that capture the most variance in the data.

Models Tested

We tested several models to see which one could best understand the relationship between our features and readability. Our experiments were designed to systematically evaluate different combinations of features, preprocessing techniques, and learning algorithms.

- **Ordianl Regression (LogisticIT, LogisticAT, LogisticSE)**: We chose these models because readability levels have a natural order (Primary < Lower Secondary < Upper Secondary). These models are speicifically designed for this type of problem, respecting the inherent ranking in the target variable.
- **Random Forest Classifier**: We also tested this powerful model because it is excellent at finding complex patterns in data and often performs very well.

Below is a detailed breakdown of the seven model configurations we evaluated:

1. **Grade-Level Features with RFE (LogisticIT)**: Used RFE to identify the most potent combination of grade-level features, forcing the model to find the core drivers of readability within this specific feature set to establish a baseline using only grade-specific metrics.
2. **Level-Only Features with SelectKBest (LogisticIT)**: Switched to the faster SelectKBest to quickly identify the strongest individual predictors, and RobustScaler was chosen to handle potential outliers that are more common in general-purpose text metrics. To evaluate the predictive power of general, level-agnostic features.
3. **Combined Features with SelectKBest (LogisticIT)**: With a much larger pool of features, SelectKBest offered an efficient way to filter out noise and focus the LogisticIT model on the most statistically relevant variables from both grade-level and level-only sets to see if a richer, combined feature set improves performance.
4. **Combined Features with PCA (LogisticIT)**: PCA was used here to combat potential multicollinearity in the combined feature set. By transforming features into principal components, this model attempts to learn from the underlying patterns of variance in the data rather than from individual features to test a dimensionality reduction approach instead of feature selection.

Table 3.10: Classification Metric Summary

Model Configuration	Precision	Recall	F1
Random Forest Classifier	0.8354	0.8192	0.8209
LogisticAT	0.7644	0.7289	0.7353
Level-Based Features (LogisticIT)	0.7790	0.7297	0.7382
Combined Features (LogisticIT)	0.7618	0.7199	0.7291
Combined + PCA (LogisticIT)	0.7687	0.7293	0.7375
Grade-Level Features (LogisticIT)	0.7647	0.7244	0.7331
LogisticSE	0.7628	0.7061	0.7085

5. **Alternative Ordinal Model with PCA (LogisticAT)**: The LogisticAT (All-Threshold) model makes different assumptions about the boundaries between classes. We kept the PCA pipeline constant to isolate the effect of the modeling algorithm itself to be able to determine if a different oridianl regression assumption fits the data better.
6. **Second Alternative Ordinal Model with PCA (LogisticSE)**: To test a third ordinal regression variant, we tested the LogisticSE (Stochastic-Explanatory) model to see if its unique approach to ordinal classification would yield better results on our PCA-transformed data similar to the previous experiment.
7. **Random Forest with RFE**: To apply a powerful, non-linear model to capture complex feature interactions, we chose the Random Forest algorithm with the targeted feature pruning of RFE. This approach was designed to be our most sophisticated attempt, allowing the model to find complex patterns within a pre-selected subset of the most impactful features.

To get a reliable measure of performance, we used 5-fold cross-validation. The main metric was accuracy, which tells us the percentage of texts that were classified correctly.

3.6.3 Results and Discussion

After running all the experiments in Shallow method, we summarized the cross-validation accuracy for each model. The results clearly show a top performer.

Important clarification of the different feature sets we tested with LogisticIT model. This allows us to see how different combinations of features affected the performance of the model.

- **Grade-Level Features**: This model was trained using the base features like avg_word_length, conjunction_ratio, etc., combined with the highly specific G1–G12 Word Ratios. It tests if a fine-grained, grade-by-grade vocabulary analysis is effective.

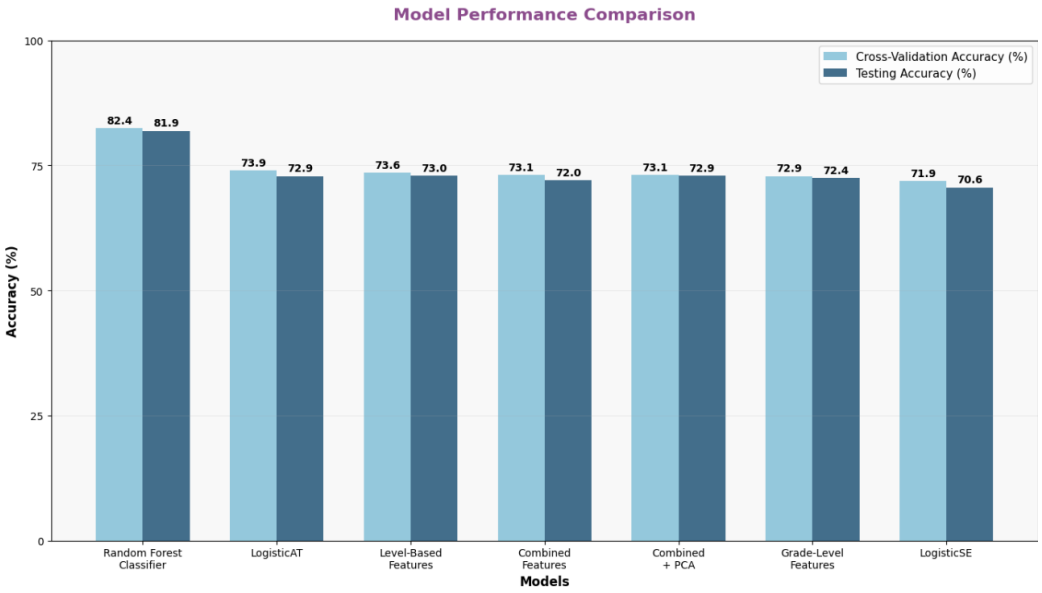


Figure 3.5: Model Comparison of Shallow Method

- **Level-Based Features**: This model used the base features plus the three broader Education-Level Word Ratios (Primary, Lower Secondary, Upper Secondary). This tests more generalized approach to vocabulary complexity.
- **Combined Features**: This model was trained on all available shallow features (the base features, all 12 grade-level ratios, and all 3 education-level ratios). The goal was to see if giving the model all possible information would yield the best result.

Table 3.10 shows the performance of these different configurations, along with the other models we tested.

The chart in Figure 3.5 compares the cross-validation accuracy of various machine learning models tested in the Burmese readability assessment experiments. The Random Forest Classifier was the clear winner, achieving an impressive accuracy of 82.39%. This performance suggests that the relationship between the shallow features and text readability is complex, and the Random Forest model was best able to capture these patterns. The ordinal regression models performed consistently well, with accuracies around 72–74%. This confirms that the shallow features we designed are good predictors of readability, even with simpler models.

Table 3.11: Train/Test distribution of samples by Level.

Level	Train	Test	Total
Primary	1548	387	1935
Lower Secondary	2845	711	3556
Upper Secondary	4195	1049	5244
Total	8588	2147	10735

3.7 Traditional Readability Formula

Readability formulas have been widely applied to assess the ease with which a text can be read, often expressed in terms of a corresponding grade level or reading ease score. These formulas help educators, writers, and researchers evaluate and create content to suit specific audiences.

The most commonly used readability formulas include the Flesch Reading Ease Score [3], which calculates readability based on sentence length and syllable count; the Dale-Challs Formula (1948) [2], which evaluates text difficulty using vocabulary familiarity combined with sentence length; the Lix Score [5], which measures readability by combining average sentence length with the percentage of long words (words with more than six letters); and the Lorge Formula [6], which is another notable readability formula based on word and sentence length [5].

Since these formulas were originally designed for languages such as English, we explored their relevance and feasibility for Burmese. Our goal was to identify which features of these formulas remain applicable and explore how to incorporate the distinct characteristics of Burmese to enhance readability assessment in this context. Specifically, we tested the Lix Score [5], Dale-Chall Formula [2], and Flesch Reading Ease Score [3] on Burmese texts to evaluate their effectiveness and adaptability. The sample distribution by level for train and test can be seen in Table 3.11.

Testing Lix Score Formula

The LIX score is calculated as:

$$\text{LIX} = \frac{wl}{s} + 100 \cdot \frac{wd}{wl} \tag{3.1}$$

where:

$wl =$ Number of words in the text

$s =$ Number of sentences in the text

$wd =$ Number of difficult words in the text (words with more than 6 letters)

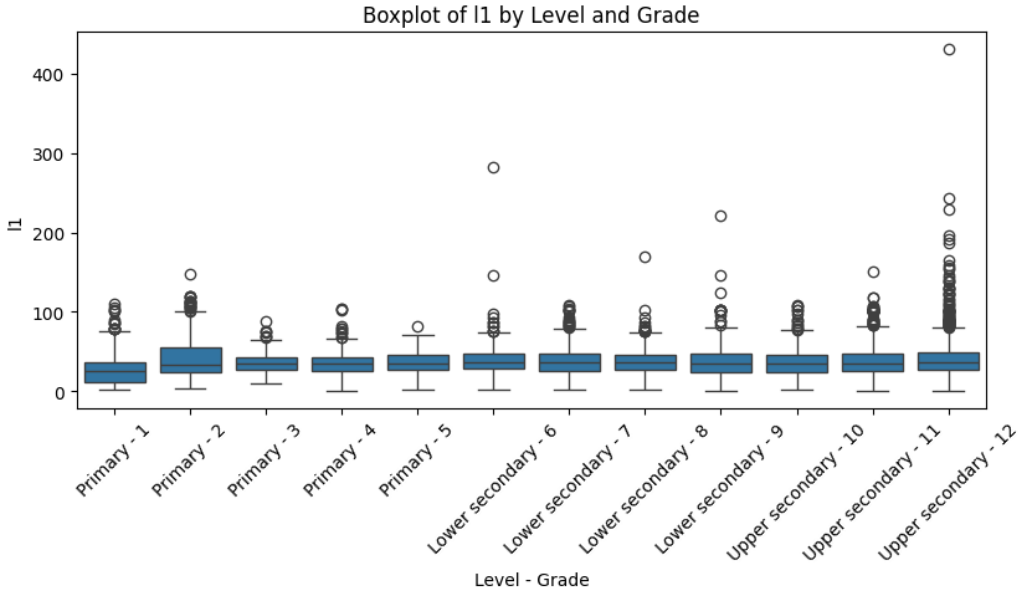


Figure 3.6: Lix Score by Grade and Level Distribution on Training set

Table 3.12: L1 statistics by grade level.

Grade Level	Min	Max	Mean
Primary - 1	2.0	110.0	28.864
Primary - 2	4.0	147.286	43.214
Primary - 3	10.0	88.714	36.261
Primary - 4	1.0	104.0	34.781
Primary - 5	2.0	81.25	36.767
Lower Secondary - 6	1.5	282.386	39.055
Lower Secondary - 7	1.5	108.101	37.770
Lower Secondary - 8	1.5	170.078	36.803
Lower Secondary - 9	1.333	221.462	36.567
Upper Secondary - 10	1.5	108.829	35.530
Upper Secondary - 11	0.5	151.4	37.121
Upper Secondary - 12	1.0	430.732	40.854

Score descriptions L1: Lix Score with its original formula definition

As seen in Table 3.12 and the box plot of Figure 3.6, there are overlaps in the score ranges to be able to differentiate between level.

Applying Linear Regression to adjust the coefficients for wl/s and wd/wl The experimental results of this formula are described in Figure 3.6 and 3.7.

According to figure 3.7 , we can see the training of two features wl/s and wd/s

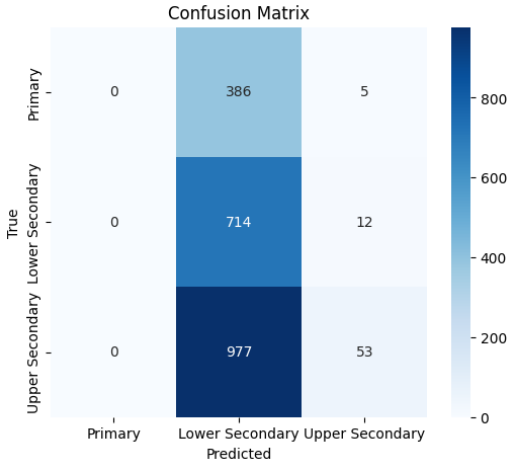


Figure 3.7: Confusion Matrix of the Linear Regression Model Testing

with our own data does not produce any 'primary' level. So , we did not use with the coefficient produced by the model.

Testing Dale-Chall Formula

The Dale-Chall formula estimates the grade level required to understand a text. It is calculated as follows:

$$\text{readability score} = 0.1579 \times \frac{wd}{wl} \times 100 + 0.0496 \times sl \tag{3.2}$$

where:

- sl = Average sentence length.
- wd = Number of different difficult words per 100 words. Difficult words are defined as words not on the Dale 769-word list.

Note: In our testing, the wd feature was modified to use a list of difficult words in Burmese (stacked words) instead of the original Dale 769-word list.

The experimental results of this formula are described in Figures 3.8 and 3.9

3.7.1 Flesch Reading Ease Formula (1948)

The Flesch Reading Ease formula is a widely used method to measure text readability. It produces a score where a lower value indicates a more difficult text.

$$\text{ReadingEase} = 206.835 - 1.015 \times sl - 0.846 \times wl \tag{3.3}$$

where:

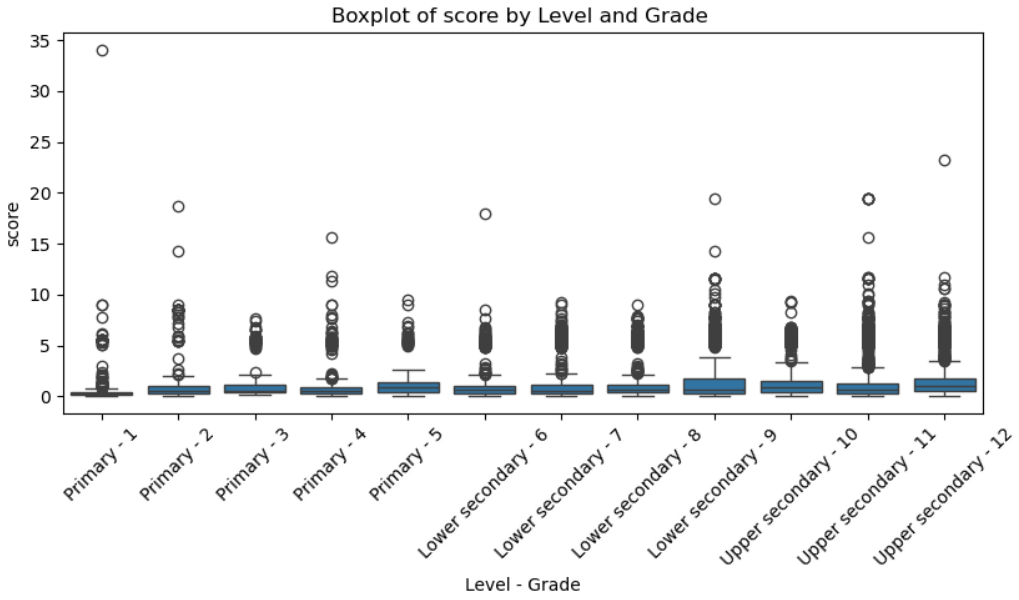


Figure 3.8: Dale-Challs Score by Grade and Level Distribution on Training set

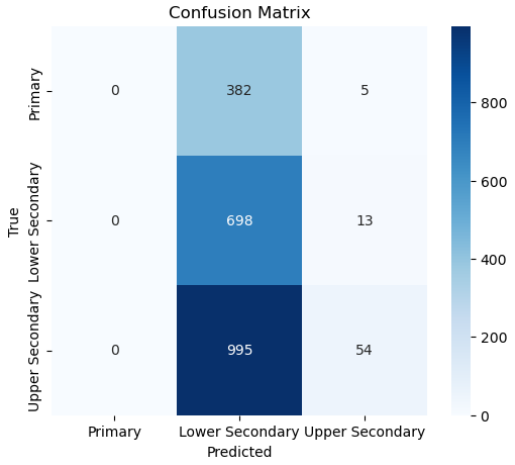


Figure 3.9: Confusion Matrix of Linear Regression on Dale–Chall readability score (1948) Features

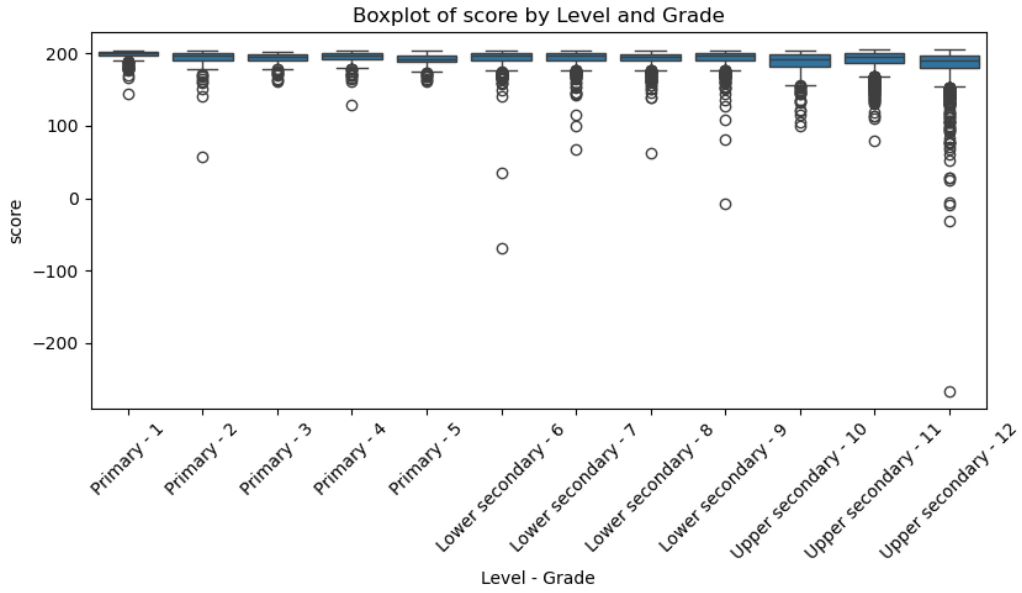


Figure 3.10: Flesch Reading Ease Score by Grade and Level Distribution on Training set

- sl = Average number of words per sentence.
- wl = Number of syllables per 100 words.

The experimental results of this formula are described in Figure 3.10 and 3.11.

3.7.2 Insights Summary

- By seeing the score distribution across different readability level the current scoring feature options to use as it in Burmese is not still getting optimal result.
- Two Issues that can be improved in the future work : Data imbalance, uniqueness on each level
- Incorporating the Burmese-specific readability features in the previous sections, will provide a more optimal readability score for Burmese data.

3.8 N-gram Perplexity Features

Perplexity features were generated following the approach in [1], using unigram, bigram, and trigram language models trained on documents from six grade levels. For a given text, each model computed the conditional probability of the word se-

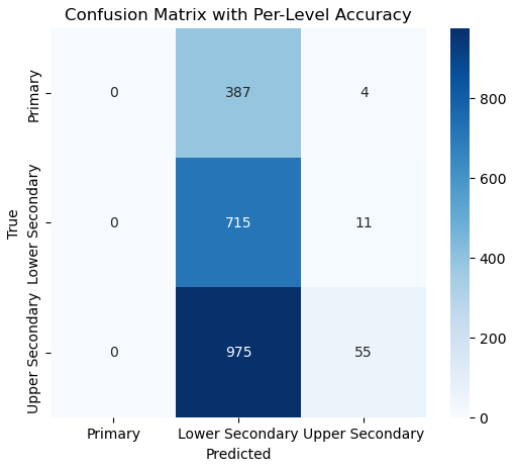


Figure 3.11: Confusion Matrix of Linear Regression on Flesch Reading Ease Features

quence, and perplexity, defined as the inverse probability normalized by sequence length, was used as a feature. Lower perplexity indicates higher predictability, capturing both term frequency and collocation information.

3.8.1 Comparison of N-gram Perplexity Features Across Tests

In this experiment, we evaluated the effectiveness of n-gram perplexity features for readability assessment across different grade levels. To examine their performance, we designed three different tests.

In Test 1, we used unigram, bigram, and trigram perplexities as features and evaluated them across all grade levels, from Grade 1 to Grade 12. Test 2 focused on aggregated, level-specific perplexities: unigram perplexity for the Primary level, bigram perplexity for the Lower Secondary level, and trigram perplexity for the Upper Secondary level. Finally, in Test 3, we combined all n-gram perplexities but tested them separately within each level.

Table 3.13 summarizes the results obtained from the three tests. It highlights how unigram, bigram, and trigram perplexities contributed to readability assessment across different levels.

3.9 Large Language Model with Zero-shot prompts

The two open-source large language models are tested with 100 sample records samples from the dataset using ollama. Table 3.14 summarized the results from

Table 3.13: Accuracy Comparison Across Grades and Tests

Grade	Test 1	Test 2	Test 3
1	0.4412	0.5074	0.1618
2	0.2537	0.3881	0.1642
3	0.1169	0.1299	0.0130
4	0.2253	0.2143	0.0055
5	0.0000	0.1045	0.0149
6	0.4933	0.6267	0.0267
7	0.3738	0.4953	0.0701
8	0.3755	0.5371	0.0524
9	0.3481	0.4333	0.0519
10	0.8674	0.7841	0.9659
11	0.8018	0.7930	0.9632
12	0.9595	0.9381	0.9857
Average Accuracy	0.4381	0.4960	0.2896
Overall Accuracy (by Level)	0.5858	0.6213	0.4909

Table 3.14: Model performance comparison across accuracy, F1-score, and per-class accuracy.

Model	Accuracy	F1-score	Primary	Lower Secondary	Upper Secondary
Gemma	0.624	0.567	0.000	0.844	0.689
LLaMA-3	0.429	0.396	0.000	0.697	0.388

two different open-source llm models namely Gemma [4] and LLaMa-3 [5] using the prompt in Table 3.15.

[4]https://ollama.com/library/gemma
[5]https://ollama.com/library/llama3

Table 3.15: Prompt Used For Testing Readability Score Response by LLM

System	You are an expert Burmese language teacher with over 15 years of experience in curriculum development and readability assessment. You have deep expertise in evaluating text difficulty across different educational levels of the education system in Myanmar.
Instructions	Carefully analyze the Burmese text provided below and assess its readability level. Evaluation Criteria: • Vocabulary Complexity: Simple everyday words vs. academic/technical terms • Sentence Structure: Short simple sentences vs. long complex sentences with multiple clauses • Grammar Patterns: Basic vs. advanced grammatical constructions • Concept Difficulty: Concrete familiar topics vs. abstract complex ideas • Text Organization: Clear logical flow vs. complex argument structure Readability Scale: • Primary Level (Very Easy) – Simple vocabulary known to young students – Short, straightforward sentences – Familiar everyday topics – Clear, direct communication • Secondary Level (Moderate) – Mix of common and academic vocabulary – Medium-length sentences with some complexity – Topics requiring some background knowledge – Generally clear but may need some concentration • Upper Secondary Level (Challenging) – Advanced vocabulary and technical terms – Long, complex sentences with multiple clauses – Abstract or specialized subject matter – Requires high-level reading skills and prior knowledge **IMPORTANT**: Respond only with '<Text>': [readability score]

Chapter 4

Future Work

The current readability classification system provides a solid foundation, but there are several avenues for future work to improve its accuracy and expand its capabilities. These efforts can build upon the promising results achieved with limited data and traditional machine learning models.

4.1 Dataset Enhancing

The existing dataset, while carefully curated, has a significant class imbalance, with higher grades having more samples than lower grades. Future work should prioritize expanding the dataset with more diverse text sources to ensure a more balanced distribution across all grade levels. This will help reduce model bias and improve performance for underrepresented categories.

4.2 Incorporation of Advanced Linguistic Features

The current feature set includes lexical, structural, and Part-of-Speech (POS) features. Future work could explore more complex linguistic features specific to the Myanmar language, such as morphological features and a more detailed analysis of stacked characters, to capture the nuances of text complexity better.

4.3 Deep Learning Model Integration

The project's experiments with machine learning models like Random Forest demonstrated strong performance, but future research can explore more advanced deep learning architectures. Experimenting with transformer-based models, such as BERT or a custom-trained equivalent for the Myanmar language, could lead to

a significant improvement in classification accuracy by leveraging their ability to capture semantic and contextual information.

Chapter 5

Conclusion

This internship project successfully developed and evaluated a Myanmar Text Readability Classification System, a timely and necessary step in a field with very limited existing research. The team successfully created a foundational dataset by systematically collecting, pre-processing, and annotating texts from educational resources. Through a series of experiments, the project demonstrated the feasibility of readability classification for the Myanmar language.

The results of the experiments showed that traditional machine learning approaches, particularly the Random Forest Regressor, achieved strong performance (R2 score of 0.80), which proved to be a robust baseline for readability prediction. Furthermore, the effectiveness of using grade-specific vocabulary ratios as a feature for the Support Vector Machine (SVM) classifier highlighted the importance of linguistic features tailored to the educational context. Although certain limitations were identified, such as the imbalance in the data set and the computational cost of some models, these findings provide clear, actionable directions for future research. In general, the contributions of this project serve as a valuable reference point and a stepping stone for the long-term advancement of Myanmar language processing.

Acknowledgment

We would like to express our deepest gratitude to several individuals and institutions who have made this internship project possible.

First and foremost, we are immensely grateful to our supervisor, Dr. Ye Kyaw Thu, for his continuous guidance, support, and insightful feedback throughout the project. His expertise was invaluable in shaping the direction of our research and helping us navigate the complexities of readability classification for the Myanmar language.

We also extend our sincere thanks to our mentors, Hlaing Myat Nwe, Khaing Hsu Wai, and Thura Aung, for their dedicated mentorship and for providing practical assistance and encouragement. Their willingness to share their knowledge and resources was crucial to the project's success.

This project was conducted at the Language Understanding Laboratory, Myanmar, and we thank the entire lab for providing us with a supportive and inspiring environment to work in.

Finally, we wish to thank our fellow team members: Kaung Khant Si Thu (Assumption University of Thailand), Hsu Yee Mon (Chiang Mai University), Seng Pan (Sirindhorn International Institute of Technology, Thammasat University), Thiha Nyein (Rangsit University), and Yu Myat Moe (Rangsit University). Their collaborative spirit, hard work, and commitment were essential to overcoming challenges and achieving our project goals.

References

[1] Y.-H Chen and Patcharanut Daowadung. Assessing readability of thai text using support vector machines. *Maejo international journal of science and technology*, 9:355–369, 11 2015.

[2] Edgar Dale and Jeanne S. Chall. A formula for predicting readability. *Educational Research Bulletin*, 27(1):11–28, 1948.

[3] Rudolf Franz Flesch. A new readability yardstick. *The Journal of applied psychology*, 32 3:221–33, 1948.

[4] Yoichiro Hasebe and Jae-Ho Lee. Introducing a readability evaluation system for japanese language education. In *Proceedings of the 6th international conference on computer assisted systems for teaching & learning Japanese*, pages 19–22, 2015.

[5] Patrik Larsson. Classification into readability levels : Implementation and evaluation, 2006.

[6] Irving Lorge. Predicting reading difficulty of selections for children. *The Elementary English Review*, 16(6):229–233, 1939.

[7] Yao-Ting Sung, Ju-Ling Chen, Ji-Her Cha, Hou-Chiang Tseng, Tao-Hsing Chang, and Kuo Chang. Constructing and validating readability models: The method of integrating multilevel linguistic features with machine learning. *Behavior research methods*, 47, 04 2014.