Natural Language Processing (2024), 1–42
doi:10.1017/nlp.2024.50

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Textual form features for text readability assessment

Wenjing Pan[1], Xia Li[1,2], Xiaoyin Chen[3] and Rui Xu[4]

[1]School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou, China, [2]Center for Linguistics and Applied Linguistics, Guangdong University of Foreign Studies, Guangzhou, China, [3]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China, and [4]College of Engineering, Carnegie Mellon University, Silicon Valley, USA
Corresponding author: Xia Li; Email: xiali@gdufs.edu.cn

(Received 2 August 2023; revised 4 June 2024; accepted 4 June 2024)

Abstract

Text readability assessment aims to automatically evaluate the degree of reading difficulty of a given text for a specific group of readers. Most of the previous studies considered it as a classification task and explored a wide range of linguistic features to express the readability of a text from different aspects, such as semantic-based and syntactic-based features. Intuitively, when the external form of a text becomes more complex, individuals will experience more reading difficulties. Based on this motivation, our research attempts to separate the textual external form from the text and investigate its efficiency in determining readability. Specifically, in this paper, we introduce a new concept, namely textual form complexity, to provide a novel insight into text readability. The main idea is that the readability of a text can be measured by the degree to which it is challenging for readers to overcome the distractions of external textual form and obtain the text's core semantics. To this end, we propose a set of textual form features to express the complexity of the outer form of a text and characterize its readability. Findings show that the proposed external textual form features can be used as effective evaluation indexes to indicate the readability of text. It brings a new perspective to the existing research and provides a new complement to the existing rich features.

Keywords: Readability assessment; textual form features; natural language processing in computer-assisted language learning

1. Introduction

Text readability refers to the textual elements that influence the comprehension, reading fluency, and level of interest of the readers (Dale and Chall 1949). These textual components involve several dimensions of text properties, including lexical, syntactic, and conceptual dimensions (Xia *et al.* 2016). In order to predict the readability of a text automatically, text readability assessment task has been proposed and widely used in many fields, such as language teaching (Lennon and Burdick 2004; Pearson *et al.* 2022), commercial advertisement (Chebat *et al.* 2003), newspaper readership (Pitler and Nenkova 2008), health information (Bernstam *et al.* 2005; Eltorai *et al.* 2015; Partin *et al.* 2022), web search (Ott and Meurers 2011), and book publishing (Pera and Ng 2014). In this section, we will first introduce our proposed concept of textual form complexity, then present related work on text readability assessment with handcrafted features, and finally present a novel insight into text readability assessment and our study.

1.1 Concept of textual form complexity

Text complexity refers to the sophisticated level of a text, which manifests in various aspects such as vocabulary level, sentence structure, text organization, and inner semantics

(Frantz *et al.* 2015). Derived from this, a new concept named textual form complexity is proposed in this paper. Among all the aspects of text complexity, textual form complexity covers the outer form facets in contrast to the inner semantic facets. In this section, we will describe this concept of textual form complexity in detail.

Previous studies have not given a consistent and formal definition of text complexity. However, one perspective (Siddharthan 2014) suggests defining text complexity as the linguistic intricacy present at different textual levels. To offer a more comprehensive explanation of this definition, we cited the viewpoints of several experts as follows:

- ***Lexical complexity*** *means that a wide variety of both basic and sophisticated words are available and can be accessed quickly, whereas a lack of complexity means that only a narrow range of basic words are available or can be accessed (Wolfe et al. 1998).*
- ***Syntactical complexity*** *refers to the sophistication of syntactic structure seen in writing or speaking that arises from our ability to group words as phrases and embed clauses in a recursive, hierarchical fashion (Friederici et al. 2017).*
- ***Grammatical complexity*** *means that a wide variety of both basic and sophisticated structures are available and can be accessed quickly, whereas a lack of complexity means that only a narrow range of basic structures are available or can be accessed (Wolfe et al. 1998).*
- ***Discourse complexity*** *refers to the difficulty associated with the text's discourse structure or coreference resolution (Cristea et al. 2000).*
- ***Graphical complexity*** *refers to the challenges in interpreting the visual aspects of text, such as letter combinations, especially for individuals with dyslexia (Gosse and Van Reybroeck 2020).*

On the basis of these viewpoints from experts, text complexity is categorized with regard to different aspects, such as lexical, syntactical, grammatical, discourse, and graphical aspects. Different from this way of categorization, we offer a new perspective. In our view, the complexity of a text is affected by the interplay of multiple factors. Among these factors, there are inner semantic-related factors, such as content, vocabulary richness, and implied information or knowledge; and there are outer form-related factors, such as expressions, structural characteristics, and layout. In this paper, we separate out these outer form-related factors and introduce a new concept, namely textual form complexity, to measure the degree of sophistication of the external forms.

Figure 1 depicts a graph that helps illustrate our proposed concept of textual form complexity. As shown in the picture, four texts are given in terms of their outer form complexity and inner semantics. As can be observed, we have the following discussions: First, there are multiple ways to convey the same semantic, and different ways of expression correspond to different textual outer forms of complexity. For example, given an inner semantic with the meaning *"I don't like him," text 1* and *text 2* are two distinct ways to convey this meaning. Although both of them share the same inner semantic, *text 1* expresses it in a simple manner and in a simple textual form, whereas *text 2* expresses it in a more complex manner and a more complex textual form. Second, whether the inner semantic is simple or complex, it can be represented by either a simple or a complex textual form. For instance, although the inner semantic of *"I don't like him,"* is easy, the complex textual form of *text 2* makes it difficult to understand; in contrast, although the inner semantic of *"He is good, but Amy doesn't like him"* is a little difficult, the simple textual form of *text 3* makes it easy to understand. Based on this observation, we argue that the outer textual form of a text plays a crucial role in determining its reading difficulty, indicating that the proposed concept of textual form complexity is worth further research.

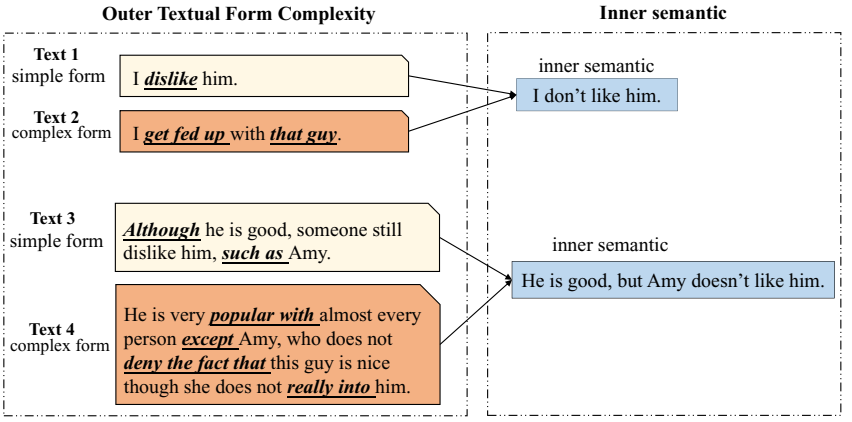

Figure 1. Diagram of the distinctions between the outer textual form complexity and the inner semantic. Four texts are presented in terms of their inner semantic and outer textual form complexity. While text 1 and text 2 convey the same inner semantics, text 2 has a more sophisticated outside textual structure than text 1. Although the inner semantics of text 3 and text 4 are more difficult to comprehend, text 3 expresses them in a simple form, whereas text 4 expresses them in a more complex form.

1.2 Text readability assessment with handcrafted features

Text readability assessment is a significant research field in text information processing. Early work focused on the exploitation of readability formulas (Thorndike 1921; Lively and Pressey 1923; Washburne and Vogel 1926; Dale and Chall 1948; Mc Laughlin 1969; Yang 1971). Some of the readability formulas approximate syntactic complexity based on simple statistics, such as sentence length (Kincaid *et al.* 1975). Other readability formulas focus on semantics, which is usually approximated by word frequency with respect to a reference list or corpus (Chall and Dale 1995). While these traditional formulas are easy to compute, they are too simplistic in terms of text difficulty assumptions (Bailin and Grafstein 2001). For example, sentence length is not an accurate measure of syntactic complexity, and syllable count does not necessarily indicate the difficulty of a word (Schwarm and Ostendorf 2005; Yan *et al.* 2006).

In recent years, researchers have tackled readability assessment as a supervised classification task and followed different methods to construct their models. The methods can be divided into three categories: language model-based methods (Si and Callan 2001; Schwarm and Ostendorf 2005; Petersen and Ostendorf 2009), machine learning-based methods (Heilman *et al.* 2007, 2008; Xia *et al.* 2016; Vajjala and Lučić, 2018), and neural network-based methods (Azpiazu and Pera 2019; Deutsch *et al.* 2020; Lee *et al.* 2021). Based on the fact that text readability involves many different dimensions of text properties, the performance of classification models highly relies on handcrafted features (Schwarm and Ostendorf 2005; Yan *et al.* 2006; Petersen and Ostendorf 2009; Vajjala and Meurers 2012).

All along, multiple linguistic features have been proposed to improve the performance of text readability assessment, such as lexical features to capture word complexity (Lu 2011; Malvern and Richards 2012; Kuperman *et al.* 2012; Collins-Thompson 2014), syntactical features to mimic the difficulty of the cognitive process (Schwarm and Ostendorf 2005; Lu 2010; Tonelli *et al.* 2012; Lee and Lee 2020), and discourse-based features to characterize text organization (Pitler and Nenkova 2008; Feng *et al.* 2010).

Although the existing studies have proposed a number of effective handcrafted features for text readability assessment, there are still three points that can be improved. First, the majority of previous studies focused on extracting features based on the content and shallow and deep semantics of the texts; few of them investigated text readability from the perspective of outer

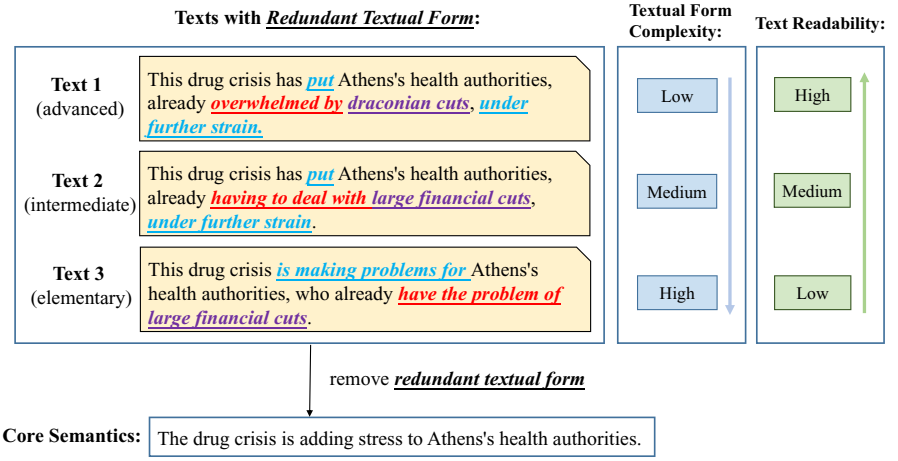

Figure 2. An instance of the novel insight into text readability. Three parallel texts with the same core semantics from the different readability levels of OneStopEnglish corpus are given. The exterior textual form within the texts is marked with underlines. As can be seen, textual form complexity is positively correlated with the amount of exterior textual form, and the exterior textual form negatively affects text readability.

textual formal complexity. Second, most of the existing features focus on a single linguistic dimension of texts; namely, one feature tends to correspond to one single linguistic dimension (e.g., the length of a word corresponds to the lexical dimension) (Collins-Thompson 2014). These single linguistic dimensional features may be limited in their ability to acquire more textual information. If a feature can reflect multiple linguistic dimensions, then it can capture more information about text readability. Third, most of the existing studies focus on a specific language (Chen and Meurers 2016; Okinina *et al.* 2020; Wilkens *et al.* 2022), few of them explicitly exploit language-independent features for multilingual readability assessment.

To this end, in this paper, we provide a new perspective on text readability by employing the proposed concept of textual form complexity and introducing a set of textual formal features for the text readability assessment based on the concept.

1.3 A novel insight into text readability assessment

In traditional viewpoints (Feng *et al.* 2010), text readability evaluates how easy or difficult it is for a reader to understand a given text. In other words, it evaluates the degree of difficulty in comprehending a specific text. In this study, we provide a new perspective on text readability by employing the proposed concept of textual form complexity. As described in Section 1.1, outer textual forms may be carriers of internal core semantics. However, complex outer textual forms may distract the reader and make it more difficult to understand the meaning of the text, reducing the text's readability. Intuitively, the more exterior textual forms a given text has, the more complex and less readable it is. This perception reflects the implicit association between textual form complexity and text readability, and the purpose of our research is to determine if and to what extent this correlation can contribute to text readability assessment.

In this work, it is hoped that a novel insight will contribute to a new understanding of text readability assessment. The insight is illustrated in Figure 2. It can be seen that the three provided texts have the same inner core semantics when the exterior textual forms are removed. Due to the different amounts of external textual form, the textual form complexity and readability of the three texts are different. Textual form complexity is positively correlated with the amount of exterior textual form, which results in the ranking of the textual form complexity of the three texts

as *text 1* < *text 2* < *text 3*. Due to the fact that the exterior textual form causes distractions and makes comprehension more difficult, it negatively affects text readability. That is, text readability is negatively connected with the quantity of exterior textual form, which promotes the ranking of the text readability of the three texts as *text 3* < *text 2* < *text 1*.

In summary, using the exterior textual form as a bridge, we can make the following inference: the higher the textual form complexity, the lower the text's readability. In response to this inference, we propose to construct a set of textual formal features to express the complexity of textual forms and use these features to evaluate the readability of texts. Specifically, we first use a text simplification tool (Martin *et al.* 2020) to simplify a given text, so as to reduce the complex exterior text forms and obtain the simplified text. Then, we design a set of textual form features based on the similarity, overlap, and length ratio of pre-simplified and post-simplified texts to quantify the differences between them as a measure of the textual exterior form information.

In contrast to existing features based on internal semantics, our proposed textual formal features may benefit from their ability to represent multiple dimensions of the language to obtain more information to assess the readability of the text. This is because textual forms represent a text's outer qualities and manifest in various linguistic dimensions, such as lexical, syntactical, and grammatical dimensions. In addition, the extraction procedure for the features involves multiple levels, such as word-level deletion, phrase-level substitution, and sentence-level reconstruction, allowing the proposed features to reflect different dimensions of linguistic information. To sum up, two studies are proposed in this work:

- *Study 1: How to build textual form features to capture text characteristics in terms of textual form complexity?*
- *Study 2: To what extent can the proposed textual form features contribute to readability assessment?*

2. Literature review

In this section, we provide a comprehensive review of related works from two perspectives: text complexity and readability assessment. Concerning text complexity, we explore relevant studies focusing on various dimensions individually in Section 2.1. Regarding readability assessment, we conduct a separate review of studies based on different methods in Section 2.2 and provide an in-depth and detailed review of the linguistic features applied for this task in Section 2.3.

2.1 Different dimensions of text complexity

Text complexity refers to the level of difficulty or intricacy present in a given text. It encompasses both linguistic and structural aspects of the text, such as vocabulary richness, sentence length, syntactic complexity, and overall readability. Over time, researchers have devised numerous tools and systems to quantify text complexity in different dimensions. In the following sections, we will separately review the relevant research on text complexity in the lexical, syntactic, and discourse dimensions.

From the lexical dimension, some studies focus on developing tools for analyzing the lexical complexity of a text, while others focus on predicting the difficulty level of individual words. In early studies, Rayner and Duffy (1986) investigate whether lexical complexity increases a word's processing time. To automatically analyze lexical complexity, there is a widely used tool called Range,[A] which focuses on measuring the vocabulary and word diversity. Based on this tool, Anthony (2014) developed a tool named AntWordProfiler, which is an extension of Range and

[A]https://www.lextutor.ca/cgi-bin/range/texts/

can be applied to multiple languages. Both tools are based on unnormalized type-token ratio (TTR), and thus, they are all sensitive to the length of the text. Recently, researchers have been engaged in the automatic classification of word difficulty levels (Gala *et al.* 2013; Tack *et al.* 2016; Tack *et al.* 2016). For example, Gala *et al.* (2013) investigate 27 intra-lexical and psycholinguistic variables and utilize the nine most predictive features to train a support vector machine (SVM) classifier for word-level prediction. Alfter and Volodina (2018) explore the applicability of previously established word lists to the analysis of single-word lexical complexity. They incorporate topics as additional features and demonstrate that linking words to topics significantly enhances the accuracy of classification.

As another dimension of text complexity, syntactic complexity measures the complexity of sentence structure and organization within the text. In earlier studies, Ortega (2003) evaluates the evidence concerning the relationship between syntactic complexity measures and the overall proficiency of second-language (L2) writers. They conclude that the association varies systematically across different studies, depending on whether a foreign language learning context is investigated and whether proficiency is defined by program level. Szmrecsanyi (2004) compares three measures of syntactic complexity—node counts, word counts, and a so-called "Index of Syntactic Complexity"—with regard to their accuracy and applicability. They conclude that node counts are resource-intensive to conduct and researchers can confidently use this measure. Recently, Lu and Xu (2016) develop a system, namely L2SCA, to analyze the syntactic complexity of second-language (L2) writing. The correlation between the output of the L2SCA system and manual calculation is 0.834-1.000. Although L2SCA is designed for L2 texts, studies have shown that it is also suitable for analyzing other textual materials (Gamson *et al.* 2013; Jin *et al.* 2020). Besides, Kyle (2016) develops another system, TAASSC. The system analyzes syntactic structure at clause and phrase levels and includes both coarse-grained and fine-grained measures.

Despite lexical and syntactic dimensions, discourse is also an integral dimension of text complexity (Graesser *et al.* 2011). In early work, Biber (1992) employs a theory-based statistical approach to examine the dimensions of discourse complexity in English. They analyze the distribution of 33 surface linguistic markers of complexity across 23 spoken and written registers. The study reveals that discourse complexity is a multidimensional concept, with various types of structural elaboration reflecting different discourse functions. Recently, Graesser *et al.* (2014) propose a classical discourse-level text analysis system, namely Coh-Metrix, which is designed to analyze text on multiple levels of language and discourse. This system regards cohesion as a textual notion and coherence as a reader-related psychological concept (Graesser *et al.* 2004). It contains measures such as cohesion features, connective-related features, and LSA-based features (Landauer *et al.* 1998). Besides, Crossley *et al.* (2016, 2019) also focus on analyzing text cohesion and develop a system named TAACO. In addition to the overall cohesion emphasized in previous systems, this system adds local and global cohesion measures based on LDA (Blei *et al.* 2003) and word2vec (Mikolov *et al.* 2013).

In recent years, educators have been trying to measure the complexity of texts so that they can provide texts that match the level of knowledge and skills of their students. Wolfe *et al.* (1998) refer to this as the zone of proximal text difficulty. That is, the text should challenge the reader in a way that inspires them to improve their existing knowledge and skills. Besides, the Common Core State Standards (CCSS) set a goal for students to comprehend texts of progressively increasing complexity as they advance through school (Association *et al.* 2010). To assist educators in determining the proper degree of complexity, CCSS standards introduce a triangle model to measure text complexity through a combination of qualitative analysis, quantitative measurements, and reader-task considerations (Williamson *et al.* 2013). Frantz *et al.* (2015) argue that the syntactic complexity, as one aspect of text complexity, should be explicitly included as a quantitative measure in the model. In addition, Sheehan *et al.* (2014) design a text analysis system namely TextEvaluator to help select reading materials that are consistent with the text complexity goals

outlined in CCSS. This system expands the text variation dimension in traditional metrics and addresses two potential threats, genre bias and blueprint bias.

2.2 Readability assessment methods

Text readability assessment refers to the process of determining the comprehensibility of a given text. It is a significant research field in text information processing. In the majority of studies, researchers treat readability assessment as a supervised classification task, aiming to assign a readability label to each text. They follow different methods to construct their models, which can be divided into three categories: machine learning-based methods, neural network-based methods, and language model-based methods. The following section will review the relevant studies based on the three methods, respectively.

The machine learning-based method involves utilizing machine learning algorithms and feature engineering to assess the readability of a given text. These approaches typically utilize a set of text features as inputs and train a machine learning model to predict the readability labels. Following this method, Vajjala and Lučić (2018) introduce a Support Vector Machine (SVM) classifier trained on a combination of 155 traditional, discourse cohesion, lexico-semantic, and syntactic features. Their classification model attains an accuracy of 78.13% when evaluated on the OneStopEnglish corpus. Xia *et al.* (2016) adopt similar lexical, syntactic, and traditional features as Vajjala and Lučić (2018) and further incorporate language modeling-based and discourse cohesion-based features to train a SVM classifier. Their approach achieves an accuracy of 80.3% on the Weebit corpus. More recently, Chatzipanagiotidis *et al.* 2021) collect a diverse set of quantifiable linguistic complexity features, including lexical, morphological, and syntactic aspects, and conduct experiments using various feature subsets. Their findings demonstrate that each linguistic dimension provides distinct information, contributing to the highest performance achieved when using all linguistic feature subsets together. By training a readability classifier based on these features, they achieve a classification accuracy of 88.16% for the Greek corpus. Furthermore, Wilkens *et al.* (2022) introduce the FABRA readability toolkit and propose that measures of lexical diversity and dependency counts are crucial predictors for native texts, whereas for foreign texts, the most effective predictors include syntactic features that illustrate language development, along with features related to lexical gradation.

With the rise of deep learning, many neural network-based methods have been proposed and applied to readability assessment. For example, Azpiazu and Pera (2019) introduce a neural model called Vec2Read. The model utilizes a hierarchical RNN with attention mechanisms and is applied in a multilingual setting. They find that no language-specific patterns suggest text readability is more challenging to predict in certain languages compared to others. Mohammadi and Khasteh (2019) employ a deep reinforcement learning model as a classifier to assess readability, showcasing its ability to achieve multi-linguality and efficient utilization of text. Moreover, Deutsch *et al.* (2020) evaluate the joint use of linguistic features and deep learning models (i.e., transformers and hierarchical attention networks). They utilize the output of deep learning models as features and combine them with linguistic features, which are then fed into a classifier. However, their findings reveal that incorporating linguistic features into deep learning models does not enhance model performance, suggesting these models may already represent the readability-related linguistic features. This finding connects to the studies of BERTology, which aim to unveil BERT's capacity for linguistic representation (Buder-Gröndahl, 2023). Research in this area has shown BERT's ability to grasp phrase structures (Reif *et al.* 2019), dependency relations (Jawahar *et al.* 2019), semantic roles (Kovaleva *et al.* 2019), and lexical semantics (Soler and Apidianaki 2020). In recent studies, hybrid models have also been explored for their applicability in readability assessment. Wilkens *et al.* (2024) systematically compare six hybrid approaches alongside standard machine learning and deep learning approaches across four corpora, covering different languages and target audiences. Their research provides new insights into the complementarity between linguistic features and transformers.

The language model-based methods can be divided into early ones based on statistical language models and recent ones based on pre-trained language models. In early work, Si and Callan (2001) are the first to introduce the application of statistical models for estimating the reading difficulty. They employ language models to represent the content of web pages and devise classifiers that integrated language models with surface linguistic features. Later on, many other methods based on statistical models are proposed. For instance, Schwarm and Ostendorf (2005) propose training n-gram language models for different readability levels and use likelihood ratios as features in a classification model. Petersen and Ostendorf (2009) train three language models (uni-gram, bi-gram, and tri-gram) on external data resources and compute perplexities for each text to indicate its readability. In recent times, pre-trained language models have gained considerable attention in academia, contributing significantly to advancements in the field of natural language processing. Concurrently, some researchers are also investigating their applicability in readability assessment. For example, Lee *et al.* (2021) investigate appropriate pre-trained language models and traditional machine learning models for the task of readability assessment. They combine these models to form various hybrid models, and their RoBERTA-RF-T1 hybrid achieves an almost perfect classification accuracy of 99%. Lee and Lee (2023) introduce a novel adaptation of a pre-trained seq2seq model for readability assessment. They prove that a seq2seq model—T5 or BART—can be adapted to discern which text is more difficult from two given texts.

2.3 Linguistic features for readability assessment

This section provides an overview of the linguistic features used in readability assessment. We review these features from five dimensions: shallow features, lexical features, syntactical features, discourse-based features, and reading interaction-based features.

The shallow features refer to the readability formulas exploited in the early research (Thorndike 1921; Lively and Pressey 1923; Washburne and Vogel 1926; Dale and Chall 1948; Mc Laughlin 1969; Yang 1971). These formulas often assess text readability through simple statistical measures. For example, the Flesch-Kincaid Grade Level Index (Kincaid *et al.* 1975), a widely used readability formula, calculates its scores based on the average syllable count per word and the average sentence length in a given text. Similarly, the Gunning Fog index (1952) uses the average sentence length and the proportion of words that are three syllables or longer to determine readability. Beyond syntax, some formulas delve into semantic analysis, typically by comparing word frequency against a standard list or database. As an illustration, the Dale-Chall formula employs a combination of average sentence length and the percentage of words not present in the "simple" word list (Chall and Dale 1995). While these traditional formulas are easy to compute, they exhibit shortcomings in terms of their assumptions about text difficulty (Bailin and Grafstein 2001). For instance, sentence length does not accurately measure syntactic complexity, and syllable count may not necessarily reflect the difficulty of a word (Schwarm and Ostendorf 2005; Yan *et al.* 2006). Furthermore, as highlighted by Collins-Thompson (2014), the traditional formulas do not adequately capture the actual reading process and neglect crucial aspects such as textual coherence. Collins-Thompson (2014) argues that traditional readability formulas solely focus on surface-level textual features and fail to consider deeper content-related characteristics.

The lexical features capture the attributes associated with the difficulty or unfamiliarity of words (Lu 2011; Malvern and Richards 2012; Kuperman *et al.* 2012; Collins-Thompson 2014). Lu (2011) analyzes the distribution of lexical richness across three dimensions: lexical density, sophistication, and variation. Their work employs various metrics from language acquisition studies and reports noun, verb, adjective, and adverb variations, which reflect the proportion of words belonging to each respective grammatical category relative to the total word count. Type-Token Ratio (TTR) is the ratio of number of word types to total number of word tokens in a text. It has been widely used as a measure of lexical diversity or lexical variation (Malvern and Richards 2012). Vajjala and Meurers (2012) consider four alternative transformations of TTR and the Measure of

Textual Lexical Diversity (MTLD; Malvern and Richards (2012)), which is a TTR-based approach that is not affected by text length. Vajjala and Meurers (2016) explore the word-level psycholinguistic features such as concreteness, meaningfulness, and imageability extracted from the MRC psycholinguistic database (Wilson 1988) and various Age of Acquisition (AoA) measures released by Kuperman *et al.* (2012). Moreover, word concreteness has been shown to be an important aspect of text comprehensibility. Paivio *et al.* (1968) and Richardson (1975) define concreteness based on the psycholinguistic attributes of perceivability and imageability. Tanaka *et al.* (2013) incorporate these attributes of word concreteness into their measure of text comprehensibility and readability.

The syntactical features imitate the difficulty of cognitive process for readers (Schwarm and Ostendorf 2005; Lu 2010; Tonelli *et al.* 2012; Lee and Lee 2020). These features capture properties of the parse tree that are associated with more complex sentence structure (Collins-Thompson 2014). For example, Schwarm and Ostendorf (2005) examine four syntactical features related to parse tree structure: the average height of parse trees, the average numbers of subordinate clauses (SBARs), noun phrases, and verb phrases per sentence. Building on this, Lu (2010) expands the analysis and implements four specific metrics for each of these phrase types, which are respectively the total number of phrases in a document, the average number of phrases per sentence, and the average length of phrases both in terms of words and characters. In addition to average tree height, Lu (2010) introduces metrics based on non-terminal nodes, such as the average number of non-terminal nodes per parse tree and per word. Tonelli *et al.* (2012) assess syntactic complexity and the frequency of particular syntactic constituents in a text. They suggest that texts with higher syntactic complexity are more challenging to comprehend, thus reducing their readability. This is attributed to factors such as syntactic ambiguity, dense structure, and high number of embedded constituents within the text.

The discourse-based features illustrate the organization of a text, emphasizing that a text is not just a series of random sentence but reveals a higher-level structure of dependencies (Pitler and Nenkova 2008; Feng *et al.* 2010; Lee *et al.* 2021). In previous research, Pitler and Nenkova (2008) investigate how discourse relations impact text readability. They employ the Penn Discourse Treebank annotation tool (Prasad, Webber, and Joshi 2017), focusing on the implicit local relations between adjacent sentences and explicit discourse connectives. Feng *et al.* (2009) propose that discourse processing is based on concepts and propositions, which are fundamentally composed of established entities, such as general nouns and named entities. They design a set of entity-related features for readability assessment, focusing on quantifying the number of entities a reader needs to track both within each sentence and throughout the entire document. To analyze how discourse entities are distributed across a text's sentences, Barzilay and Lapata (2008) introduce the entity grid method for modeling local coherence. Based on this approach, Guinaudeau and Strube (2013) refine the use of local coherence scores, applying them to distinguish documents that are easy to read from those difficult ones.

Recently, various reading interaction-based features have been proposed and applied in the task of readability assessment (Gooding *et al.* 2021; Baazeem, Al-Khalifa, and Al-Salman 2021). Dale and Chall (1949) suggest that readability issues stem not only from the characteristics of the text but also from those of the reader. Similarly, Feng *et al.* (2009) argue that readability is not just determined by literacy skills, but also by the readers' personal characteristics and backgrounds. For example, Gooding *et al.* (2021) measure implicit feedback through participant interactions during reading, generating a set of readability features based on readers' aggregate scrolling behaviors. These features are language agnostic, unobtrusive, and are robust to noisy text. Their work shows that scroll behavior can provide an insight into the subjective readability for an individual. Baazeem *et al.* (2021) suggest the use of eye-tracking features in automatic readability assessment and achieve enhanced model performance. Their findings reveal that eye-tracking features are superior to linguistic features when combined, implying the ability of these features to reflect the text readability level in a more natural and precise way.

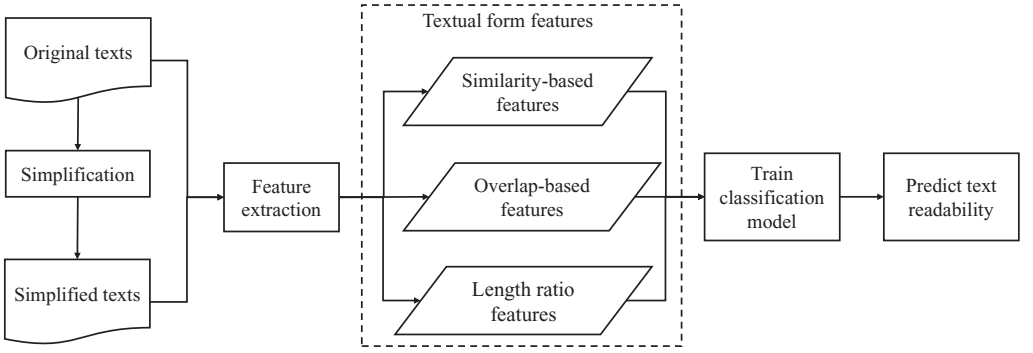

Figure 3. The flowchart of our method. To obtain the simplified texts, we begin by simplifying the original texts. The original and simplified texts are then passed to feature extraction as inputs to get the three subsets of textual form features. Using these features, we then train a classification model to predict text readability.

In addition to the linguistic features for readability assessment reviewed above, this paper proposes a set of novel features which characterize text readability from the perspective of outer textual form complexity. In subsequent sections, we will introduce the proposed features and describe our readability assessment method in detail.

3. Method

3.1 Overall framework

Based on the concept of textual form complexity, we propose to construct a set of textual formal features to help characterize the readability of the text. The overall framework of our method is presented in Figure 3. Our method consists of three components. The first component is the text simplification stage. In this stage, we use the simplification tool to simplify each text to its simplified version. These original and simplified texts are used to extract the textual formal features in the second stage. The process of detailed simplification will be introduced in Section 3.2. The second component is the feature extraction stage. In this stage, we extract the textual form features with the input of the original texts and their simplified versions. The textual form features are extracted by measuring the distinction between the original and the simplified texts from three aspects, which correspond to similarity-based features, overlap-based features, and length ratio-based features. The details of the extraction of the textual form features will be introduced in Section 3.3. The third component is the model training stage. In this stage, we first train a classification model with the extracted features and use the model to predict the readability label of the texts. The details of model prediction will be introduced in Section 3.4.

3.2 Simplification operations to reduce the exterior textual form

In order to provide a more formal description for the proposed idea of textual form complexity, we study certain criteria for writing simple and clear texts, such as the Plain Language Guidelines (Caldwell *et al.* 2008; PLAIN 2011) and the "Am I making myself clear?" guideline (Dean 2009). The basic principles shared by these guidelines are: "write short sentences; use the simplest form of a verb (present tense and not conditional or future); use short and simple words; avoid unnecessary information; etc." Basically, the principle is to simplify the complexity of the textual form by reducing unnecessary exterior textual forms. Consequently, we conclude that the complexity of textual form is a measure of the amount of exterior textual form in a given text.

Table 1. Linguistic obstacles and simplification operations to remove them

Type	Linguistic obstacles	Simplification operations
Lexical	low frequency	replacement by more frequent words and phrases
	lengthy words	replacement by shorter words
Syntactic	long sentence	sentence splitting or removal of non-essential information
	apposition	sentence splitting or removal of apposition (if non-essential)
	relative clause	sentence splitting or removal of subordinate clause (if non-essential)

This table is presented by Štajner *et al.* (2022)
We quote this table to show the functions of simplification operations in reducing the exterior textual form.

In the field of automatic text simplification, these guidelines also serve as guidance on the simplification operations (Štajner *et al.* 2022), such as splitting sentences, deletion of non-essential parts, and replacement of simple words. Table 1 presents some typical examples. It shows that simplification operations can help reduce the external textual form by reducing the linguistic obstacles within the texts. Furthermore, text simplification, according to Martin *et al.* (2020), is the process of reducing the complexity of a text without changing its original meaning. Based on the above theoretical foundations, we assume that text simplification can reduce the outer textual form while retaining the inner core semantics. If we can measure what the simplification operations have done, we can quantify the exterior textual form. Motivated by this, we propose to extract the textual form features by measuring the reduced textual forms during the simplification process.

The technique used in the simplification process is a multilingual unsupervised sentence simplification system called MUSS (Martin *et al.* 2020). The architecture of this system is based on the sequence-to-sequence pre-trained transformer BART (Lewis *et al.* 2020). For different languages, we utilize the corresponding pre-trained model, that is"*muss_en_mined*" for English, "*muss_fr_mined*" for French, and "*muss_es_mined*" for Spanish. Specifically, we first utilize the NLTK tool for sentence segmentation (Loper and Bird 2002). The pre-trained model is then applied to each document, sentence by sentence, to simplify it. Finally, we incorporate the simplified sentences into a single document as the corresponding simplified text.

We show the simplification process in Figure 4. Supposing the document set of original texts is $D_{orig} = \{d_1, d_2, \ldots, d_i, \ldots, d_n\}$. For a given text d_i $(1 \leq i \leq n)$, the original sentence set of d_i is denoted as $S^i_{orig} = \{s_1, s_2, \ldots s_j, \ldots, s_k\}$. After the simplification process, the simplified text of d_i is obtained, which is denoted as $d_i{}'$. Due to the sentence-by-sentence manner, we also obtain the parallel simplified sentence $s_j{}'$ for each sentence in the text d_i. If a sentence is split into multiple sub-sentences after simplification, we connect the sub-sentences as the corresponding simplified sentence $s_j{}'$. Then the simplified sentence set of the simplified document $d_i{}'$ is obtained, which is denoted as $S^i_{simp} = \{s_1{}', s_2{}', \ldots, s_j{}', \ldots, s_k{}'\}$. We denote the parallel sentence pairs as $(s_j, s_j{}')$ $(s_j \in S^i_{orig}, s_j{}' \in S^i_{simp}, 1 \leq i \leq n, 1 \leq j \leq k)$, where n is the total number of the texts in the corpus and k is the total number of the sentences in each text.

3.3 Textual form features

On the basis of our innovative insight into text readability, we offer a set of textual form features for assessing text readability. The retrieved features are based on the assumption that the differences between the original text and the simplified text can reflect the degree to which the complex exterior textual form is reduced during the simplification process. In other words, the distinctions measure the quantity of the outer textual form and hence serve as an indicator of the complexity

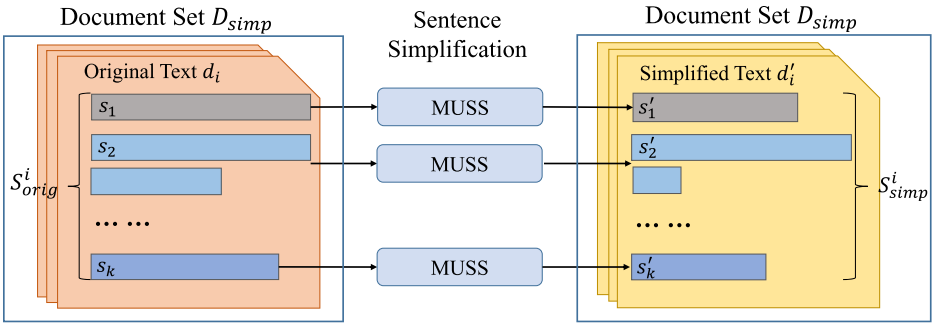

Figure 4. The symbolism of the simplification process. We use the sentence-level simplification model MUSS to simplify each sentence s_j and obtain a simplified sentence s_j' for each text d_i in the original document set D_{orig}. Finally, we combine the simplified sentences in each text d_i' to produce a parallel simplified document set D_{simp}.

of the textual form. To achieve this, we extract features from three dimensions, namely similarity, overlap, and length ratio, to reflect the difference between the original text and the simplified text and to quantify how many exterior forms have been reduced. The proposed textual form features are divided into three subsets:

- **Similarity-based features.** The similarity-based features are based on the classical similarity measurements, including cosine similarity, Jaccard similarity, edit distance, BLEU score, and METEOR score. In particular, we propose an innovative feature *BertSim*, which determines the distance between BERT-based text representations.
- **Overlap-based features.** The overlap-based features are proposed based on the intuition that word-level overlap can represent word-level change through simplification (e.g., the number of complex word deletion) and capture lexical-dimension textual form complexity (e.g., vocabulary difficulty). We investigate the overlap from the perspective of part-of-speech (i.e., noun, verb, and adjective).
- **Length ratio-based features.** The length ratio-based features are based on the observation that text is usually shortened after being simplified, we propose to use the sentence-level length ratio to characterize the syntactical dimension of textual form complexity.

In the following subsections, we will describe these three types of textual form features in detail.

3.3.1 Similarity-based features

According to the information theory lin1998information, similarity refers to the commonalities shared by two different texts. The greater the commonality, the higher the similarity, and vice versa. Over the past three decades (Wang and Dong 2020), several similarity measurements have been proposed. These similarity measurements vary in how they represent text and compute distance. In this work, we use three classical similarity measurements to capture textual form complexity for readability assessment. In addition, we also propose an innovative feature based on the similarity of BERT embeddings (Imperial 2021). The detailed descriptions of them are as follows:

- **CosSim**.

Cosine Similarity measures the cosine of the angle between two vectors in an inner product space (Deza and Deza 2009). For a given document, the *CosSim* feature is calculated as the average cosine similarity between the vector representations of sentences in the text. It can be considered a

sentence-level similarity between the texts before and after simplification. Specifically, for a given text d_i, the k parallel sentence pairs (s_j, s_j') are obtained through simplification operations, where j=1,2. . .,k. We use the one-hot encoding as the vector representations of these sentences, which is a well-known bag-of-words model for text representation (Manning and Schütze 1999). Then, we calculate the cosine similarity between the one-hot vectors of s_j and s_j' and average the k values in text d_i. The feature *CosSim* calculation formula is as follows:

$$CosSim(d_i) = \frac{1}{k}\sum_{j=1}^{k} \frac{\vec{h_o^j} \cdot \vec{h_s^j}}{\left\|\vec{h_o^j}\right\| \cdot \left\|\vec{h_s^j}\right\|} \tag{1}$$

where $\vec{h_o^j}$ is the one-hot vector of sentence s_j in original text d_i, $\vec{h_s^j}$ is the one-hot vector of sentence s_j' in simplified text d_i', $\left\|\vec{h^j}\right\|$ denotes the modulus of vector $\vec{h^j}$, k is the number of sentence pair (s_j, s_j') for the given text d_i.

- **JacSim**.

Jaccard similarity is a similarity metric used to compare two finite sets. It is defined as the intersection of two sets over the union of two sets (Jaccard 1901). Applied to text similarity measurement, Jaccard similarity is computed as the number of shared terms over the number of all unique terms in both strings (Gomaa and Fahmy 2013). The feature *JacSim* measures the average Jaccard similarity of the sentence pairs for the given text. The extraction procedure is the same as for the feature *CosSim*, and the calculation formula is as follows:

$$JacSim(d_i) = \frac{1}{k}\sum_{j=1}^{k} \frac{\left|W_{s_j} \cap W_{s_j'}\right|}{\left|W_{s_j} \cup W_{s_j'}\right|} \tag{2}$$

where W_{s_j} denotes the word set of sentence s_j in original text d_i, $W_{s_j'}$ denotes the word set of sentence s_j' in simplified text d_i', k is the number of sentence pairs (s_j, s_j') for the given text d_i.

- **EditDis**.

Edit distance is a metric used in computational linguistics to measure the dissimilarity between two strings. It is determined by the minimal number of operations required to transform one string into another (Gomaa *et al.* 2013). Depending on the definitions, there are different sets of string operations at different edit distances. In our work, we used the Levenshtein distance, which incorporates operations such as the deletion, insertion, or substitution of a character in a string (Levenshtein *et al.* 1966). The feature *EditDis* calculates the Levenshtein distance between the original text and its simplified version. Because the Levenshtein distance is calculated character by character, the feature *EditDis* can capture the character-level textual form complexity for a given text.

- **BertSim**.

The *BertSim* feature measures the text's readability based on the similarity between the BERT-based representations of the texts before and after simplification at the document level. BERT is a pre-trained language model that has shown effectiveness in a variety of NLP tasks. It possesses the inherent capability to encode linguistic knowledge, such as hierarchical parse trees (Hewitt and Manning 2019), syntactic chunks (Liu *et al.* 2019), and semantic roles (Ettinger 2020). We believe such knowledge can be incorporated to capture the text's multidimensional textual features in order to more comprehensively represent the textual form complexity. During

the feature extraction process, we leveraged the pre-trained models "bert-base-cased[B]" and "bert-base-multilingual-uncased[C]" for English and non-English texts. Steps for feature extraction are as follows:

Step 1: BERT-based sentence representation

For each sentence of a given text, we use the BERT tokenizer to split and get the token sequence $[t_1, t_2, \ldots, t_l, \ldots, t_L]$, where t_l is the l^{th} token for the sentence and $L = 510$ is the maximum sentence length supported by BERT. If the token length in the sentence is greater than L, the sequence will be truncated; otherwise, it will be padded with the $[PAD]$ token. The final token sequence of the sentence is concatenated by the $[CLS]$ token, original sentence token sequence, and the $[SEP]$ token, which is denoted as $T_j = [[CLS], t_1, t_2, \ldots\ldots t_L, [SEP]]$. As described in the work of BERT (Kenton and Toutanova 2019), we use the final token embeddings, segmentation embeddings, and position embeddings as input to the BERT to get the BERT encoding of the sentence. For a given sentence s_j, the output hidden states of the sentence are $\mathbf{H} \in \mathbb{R}^{L \times d}$, where $d = 768$ is the dimension of the hidden state. In our work, instead of using the output vector of token [CLS], we use the mean pooling of $\mathbf{H} \in \mathbb{R}^{L \times d}$ as the sentence representation, which is denoted as $\vec{v_{s_j}} \in \mathbb{R}^d$.

Step 2: BERT-based text representation and similarity calculation

After getting the BERT-based representations of the text's sentences, the BERT-based representation of the text can be obtained by taking the average of the sentence representations. Given the original text d_i and the simplified text d_i', we first get the representations of all sentences in them, which are denoted as $V_{d_i} = \{\vec{v_{s_1}}, \vec{v_{s_2}}, \ldots, \vec{v_{s_k}}\}$ for the text d_i and $V_{d_i'} = \{\vec{v_{s_1}}', \vec{v_{s_2}}', \ldots, \vec{v_{s_k}}'\}$ for the text d_i', where k is the sentence number, $\vec{v_{s_k}} = [v_1^k, v_2^k, \ldots, v_{768}^k]$ is the vector of the k^{th} sentence in d_i and $\vec{v_{s_k}}' = [v_1^{k\prime}, v_2^{k\prime}, \ldots, v_{768}^{k\prime}]$ is the vector of the k^{th} sentence in d_i'. The final representation $\vec{v_{d_i}} = [v_1, v_2, \ldots, v_{768}]$ for text d_i and $\vec{v_{d_i'}} = [v_1', v_2', \ldots, v_{768}']$ for text d_i' are obtained by averaging the values of each dimension of the k number of sentences. The inner product of the two text representations is then utilized as the *BertSim* feature value. The equations are as follows:

$$BertSim(d_i) = \sum_{i=1}^{768} v_i \cdot v_i' \quad (i = 1, 2, \ldots, 768) \tag{3}$$

$$v_i = \frac{1}{k} \sum_{j=1}^{k} v_i^j \quad (j = 1, 2, \ldots, k, i = 1, 2, \ldots, 768) \tag{4}$$

$$v_i' = \frac{1}{k} \sum_{j=1}^{k} v_i^{j\prime} \quad (j = 1, 2, \ldots, k, i = 1, 2, \ldots, 768) \tag{5}$$

where k is the number of sentences in text, v_i is the average value of i^{th} dimension in the representations of the k sentences of text d_i, and v_i' is the average value of i^{th} dimension in the representations of the k sentences of text d_i'. The detailed procedure for step 2 is presented in Figure 5.

[B]https://huggingface.co/bert-base-uncased
[C]https://huggingface.co/bert-base-multilingual-uncased

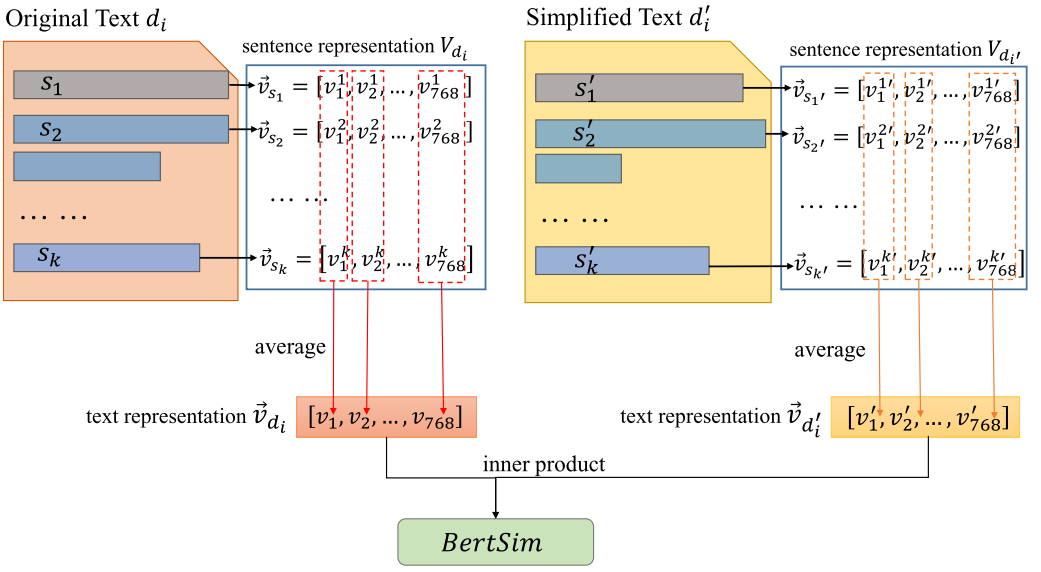

Figure 5. The extraction process of the feature *BertSim* in step 2. The sentence representations $\vec{v_{s_k}}$ and $\vec{v_s^i}$ are obtained in step 2. For each text d_i, we average the value of each dimension in the sentence representations as the text representation. The text representation for the original text is $\vec{v_{d_i}}$, and the text representation for the simplified text is $\vec{v_{d_i'}}$. At last, the inner product of $\vec{v_{d_i}}$ and $\vec{v_{d_i'}}$ is calculated as the final value of the feature *BertSim*.

- **BleuSim**.

The BLEU score is a commonly used evaluation metric for machine translation (Papineni *et al.* 2002). It assesses the frequency of n-gram matches (usually 1-gram to 4-gram) in the machine-generated text compared to reference texts. As the core idea of BLEU is to evaluate the alignment between machine-generated output and human reference, BLEU score can be viewed as a similarity measure between two text segments (Callison-Burch, Osborne, and Koehn 2006). Based on this similarity measure, we construct the feature *BleuSim*. Specifically, *BleuSim* computes the BLEU score for each original-simplified sentence pair for the given text d_i and then averages these scores across all pairs. The calculation formulas are as follows:

$$BleuSim(d_i) = \frac{1}{k}\sum_{j=1}^{k} \text{BLEU}\left(s_j, s_j'\right) \tag{6}$$

$$\text{BLEU}\left(s_j, s_j'\right) = \text{BP} \cdot \exp\left(\frac{1}{4}\log P_1^{(j)} + \frac{1}{4}\log P_2^{(j)} + \frac{1}{4}\log P_3^{(j)} + \frac{1}{4}\log P_4^{(j)}\right) \tag{7}$$

where $\text{BLEU}(s_j, s'_j)$ is the BLEU score for the j-th sentence pair for text d_i, s_j is the original sentence, s'_j is the corresponding simplified sentence, and k is the total number of sentence pairs. BP is the brevity penalty that penalizes shorter simplified sentences to ensure they are not too brief compared to the original sentences. $P_n^{(j)}$ denotes the precision of the n-gram for the j-th sentence pair. Each n-gram length is treated equally with the weight of 1/4.

The precision $P_n^{(j)}$ is the proportion of the matched n-grams in s'_j out of the total number of n-grams in s_j. Its calculation is as follows:

$$P_n = \frac{\sum_{n\text{-gram} \in s_i'} \min\left(\text{Count}_{s_i'}(n\text{-gram}) + 1, \text{MaxCount}_{s_i}(n\text{-gram}) + 1\right)}{\sum_{n\text{-gram} \in s_i'}\left(\text{Count}_{s_i'}(n\text{-gram}) + 1\right)} \tag{8}$$

where $\text{Count}_{s'_i}(n\text{-gram})$ is the count of n-gram in the simplified sentence s'_j. $\text{MaxCount}_{s_i}(n\text{-gram})$ is the maximum count of that n-gram in the original sentence. We use the additive smoothing method to avoid the extreme case where an n-gram is completely absent in s'_j, leading to a zero precision. This smoothing method adds the constant 1 to the count of all n-grams.

- **MeteorSim**.

The METEOR is an automatic metric for the evaluation of machine translation(Banerjee and Lavie 2005). It functions by aligning words within a pair of sentences. By accounting for both precision and recall, METEOR offers a more balanced similarity measure between two text segments. Based on this similarity measure, we construct the feature *MeteorSim*. Specifically, *MeteorSim* computes the METEOR score for each sentence pair for the given text d_i and then averages these scores across all pairs. The calculation formulas are as follows:

$$MeteorSim(d_i) = \frac{1}{k}\sum_{j=1}^{k} \text{METEOR}\left(s_j, s'_j\right) \tag{9}$$

$$\text{METEOR}\left(s_j, s'_j\right) = (1-\rho) \times \frac{P_j \times R_j}{\alpha \times P_j + (1-\alpha) \times R_j} \tag{10}$$

where k is the total number of sentences in text d_i, s_j is the j-th sentence in document d_i, s'_j is the corresponding simplified sentence s_j. ρ is the penalty factor based on the chunkiness of the alignment between s_j and s'_j. It penalizes the score if the word order in the simplified sentence differs significantly from the original sentence. α is a parameter that balances the contributions of precision P_j and recall R_j.

For the j-th sentence pair, precision P_j quantifies the ratio of words in the simplified sentence s'_j that align to the words in the original sentence s_j; recall R_j quantifies the ratio of words in the original sentence s_j that find alignments in the simplified sentence s'_j. According to Banerjee and Lavie (2005), the word alignment process is executed by considering not only exact word matches but also the presence of synonyms and stem matches. The calculation formulas are as follows:

$$P_j = \frac{\sum_{w \in W_{s'_j}} \text{align}\left(w, W_{s_j}\right)}{\left|W_{s'_j}\right|} \tag{11}$$

$$R_j = \frac{\sum_{w \in W_{s_j}} \text{align}\left(w, W_{s'_j}\right)}{\left|W_{s_j}\right|} \tag{12}$$

where $W_{s'_j}$ is the set of words in the simplified sentence s'_j, W_{s_j} is the set of words in the original sentence s_j, $\text{align}(w, W)$ is a function that returns 1 if the word w aligns with any word in the set W, otherwise 0. $|W_{s_j}|$ and $|W_{s'_j}|$ denote the number of words in the sentence s_j and s'_j, respectively.

The six similarity-based features listed above characterize textual form complexity at various levels and assess text readability across multiple linguistic dimensions. The features *CosSim* and *JacSim* measure the word-level text similarity before and after simplification. *JacSim* captures vocabulary change through a set operation, whereas *CosSim* captures the difference in word distribution through one-hot encoding. For these two features, the lower the value, the greater the textual form complexity and the lower the text's readability. The feature *EditDis* measures the character-level text distance before and after simplification. Because text distance indicates how much external textual form exists, the higher the value of the feature *EditDis*, the

more complex the textual form, and the less readable the text. Regarding the feature *BertSim*, given that text representations are rich in syntactic and discourse knowledge as noted by Rogers *et al.* (2020), it is capable of capturing textual form complexity at the text level and assessing readability from the perspective of syntactic. The classical machine translation metrics-based features, *BleuSim* and *MeteorSim*, differ in their approaches: *BleuSim* functions at the n-gram level, evaluating the occurrence of specific phrases up to a length of n, while *MeteorSim* expands on this by including synonyms and stemming in its evaluation. Similar to *CosSim* and *JacSim*, lower values of these three features indicate greater textual form complexity and reduced readability.

3.3.2 Overlap-based features

Text simplification involves some lexical simplification operations, such as word substitution and deletion (Qiang *et al.* 2020). We believe that measuring the simplification change at the word level within the categories of different parts of speech can help in capturing the lexical-textual form complexity of the text in more granular aspects. This is because the greater the change, the more complex the textual form and the less readable the text. Based on this, in this work, we propose using the overlap of the words in the texts before and after simplification in terms of noun, verb, and adjective part-of-speech categories to capture this change. We only focus on these three parts of speech since other parts of speech are less frequent, resulting in a large number of zeros and a limited capacity to capture overlap information. The following describes the extraction process for the three overlap-based features:

- **NounOlap** , **AdjOlap** , **VerbOlap**.

The extraction process for the three features contains two steps. Firstly, the NLTK tool is initially employed for word segmentation and pos-tagging. As the NLTK tag categories are fine-grained, the tags are regrouped into three coarse-grained categories: noun, verb, and adjective. The detailed groupings of tags are listed in Appendix A. Secondly, we collect the word sets of each sentence according to the pos-tagging of each word. Assuming that the noun, verb, and adjective word sets are $W_{noun}^{s_j}$, $W_{verb}^{s_j}$, and $W_{adj}^{s_j}$ for sentence s_j and $W_{noun}^{s_j'}$, $W_{verb}^{s_j'}$, and $W_{adj}^{s_j'}$ for sentence s_j', the three overlap-based features are calculated as follows:

$$NounOlap(d_i) = \frac{1}{k}\sum_{j=1}^{k} \frac{\left| W_{noun}^{s_j} \cap W_{noun}^{s_j'} \right|}{\left| W_{noun}^{s_j} \cup W_{noun}^{s_j'} \right|} \tag{13}$$

$$AdjOlap(d_i) = \frac{1}{k}\sum_{j=1}^{k} \frac{\left| W_{adj}^{s_j} \cap W_{adj}^{s_j'} \right|}{\left| W_{adj}^{s_j} \cup W_{adj}^{s_j'} \right|} \tag{14}$$

$$VerbOlap(d_i) = \frac{1}{k}\sum_{j=1}^{k} \frac{\left| W_{verb}^{s_j} \cap W_{verb}^{s_j'} \right|}{\left| W_{verb}^{s_j} \cup W_{verb}^{s_j'} \right|} \tag{15}$$

where $W_{pos} \cap W_{pos}$ and $W_{pos} \cup W_{pos}$ are the intersection and union of the two word sets, respectively, $pos \in \{noun, adj, verb\}$ is the part-of-speech corresponding to the word sets $\{W_{noun}, W_{adj}, W_{verb}\}$. The operation $|W|$ represents the number of the elements in the set W.

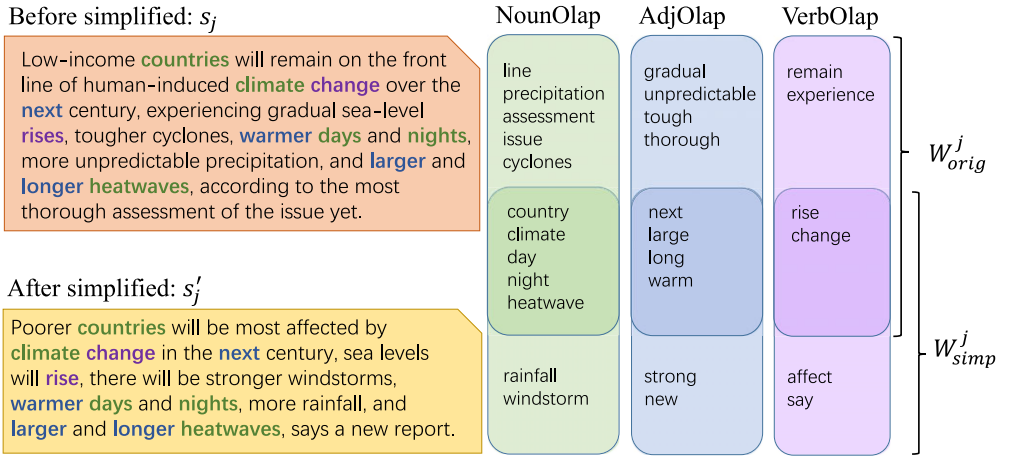

Figure 6. An instance of the overlap at the word level between the sentences s_j (before simplified) and s_j' (after simplified). We investigate the overlap from the perspective of part-of-speech. The common nouns, adjectives, and verbs in two sentences are counted separately.

An instance of the overlap at the word level between the sentences is presented in Figure 6. Nouns, adjectives, and verbs in sentences s_j (before simplified) and s_j' (after simplified) are counted individually. There are common nouns, adjectives, and verbs between the two sentences. The overlap-based features quantify the proportion of common words within the part-of-speech category, indicating the number of words that remain unchanged during the simplification process. Due to the fact that text simplification aims at reducing redundant textual form, the degree of change reveals the complexity of the textual form. Therefore, the lower the feature values, the greater the complexity of the textual form and the lower the readability.

3.3.3 Length ratio-based feature

We introduce a feature named *LenRatio* that assesses changes in text length due to simplification. On the one hand, simplification may condense the sentence by removing non-essential elements, thereby shortening its length. On the other hand, it can also lead to an increase in length, especially when a complex term with rich meaning is paraphrased using more simple and straightforward words to retain the original connotation. Regardless of whether the simplification results in a longer or shorter text, the change in length mirrors the change of the exterior textual form through simplification and thus characterizes the readability of the original text, as described in Section 3.2. We suggest employing the length ratio to quantify this change. The detailed extraction process is introduced in the following:

- **LenRatio**.

For a given text d_i, the *LenRatio* feature of the text is calculated as:

$$LenRatio(d_i) = \frac{1}{k}\sum_{j=1}^{k}\frac{\text{len}\,(s_j')}{\text{len}\,(s_j)} \tag{16}$$

where s_j is the j-th sentence in text d_i, s_j' is the simplified sentence of s_j, k is the number of the sentences in text d_i, len (s_j) and len (s_j') are the length of the original and simplified sentences, which are defined as the total number of words in the sentence.

Table 2. The hyperparameter settings in RandomForest and SMO

Model	Parameter	Parameter	Value settings
RandomForest	I	The number of trees in the random forest.	100
	P	Size of each bag, as a percentage of the training set size.	100
	K	Number of attributes to randomly investigate, 0 for all.	0
	M	Set minimum number of instances per leaf.	1
	depth	The maximum depth of the tree, 0 for unlimited.	0
	S	Seed for random number generator.	1
SMO	c	The complexity parameter C.	1
	epsilon	The epsilon for round-off error.	$1 \times e^{-12}$
	numDecimalPlaces	The number of decimal places used for the output of numbers.	2
	randomSeed	Random number seed for the cross-validation.	1
	kernel	The kernel to use.	PolyKernel

3.4 Model prediction

In our work, we utilize two types of classification models for readability prediction: the tree-based classifier Random Forest (Breiman 2001), and the function-based classifier Support Vector Machine (SVM) (Hearst *et al.* 1998). For the tree-based model, Random Forest generates multiple decision trees using a randomly selected subset of training samples and employs bagging (Breiman 1996) as the ensemble learning method. The model predicts the class through a majority vote and ensures low correlation among decision trees by creating a random selection of attributes. For the function-based model, SVM works by identifying the optimal hyperplane that most effectively separates different classes in the feature space, aiming to maximize the margin between the closest points of the classes. The model is especially advantageous in high-dimensional spaces and in situations where the number of dimensions surpasses the number of samples. We employ the Sequential Minimal Optimization method (Platt 1999) for training the SVM model. Thus, we refer to this model as SMO in the following sections.

We use the WEKA 3.9.6[D] (Waikato Environment for Knowledge Analysis) (Witten and Frank 2002) tool to build the models. The models' hyperparameter settings are shown in Table 2. Following previous work, we use accuracy, F1 score, precision, and recall as metrics to evaluate the model's performance.

4. Corpora

In order to evaluate to what extent our proposed textual form features can contribute to readability assessment, we first carry out preliminary validation experiments using two parallel corpora, which are widely used in both readability assessment and text simplification tasks. The main experiments in our work are conducted on three traditional readability corpora. Among these corpora, two are in English, and the third is a multilingual corpus in Spanish and French. This section provides an overview of these five corpora.

[D]https://www.weka.io/

Table 3. Statistics for the OneStopEnglish corpus

Level	Number of documents	Total number of words	Number of words per document
Elementary	189	100800	533.33
Intermediate	189	127934	676.90
Advanced	189	155253	820.49

4.1 Parallel corpora for preliminary validation

Parallel corpora consist of text collections in which every complex sentence from a source text is matched with a more straightforward and comprehensible version of that same sentence. In our work, we employ two such parallel corpora, widely utilized in both readability assessment and text simplification research. Because the sentences in the corpora are manually simplified to their simplified versions, it allows us to extract the proposed features and preliminarily validate their effectiveness, bypassing the need for automated simplification processes.

OneStopEnglish.[E] This corpus is created by Vajjala and Lučić (2018). It contains 189 aligned documents at three reading levels: elementary (ELE), intermediate (INT), and advanced (ADV). The content of the corpus comes from the language learning resource website onestopenglish.com from 2013 to 2016 and was rewritten by English teachers into the three levels for L2 English adult learners. The statistics of the corpus are presented in Table 3. The sentence alignment process, as described by Vajjala and Lučić (2018), yields 1,674 sentence pairs for the ELE-INT level, 2,166 sentence pairs for the ELE-ADV level, and 3,154 sentence pairs for the INT-ADV level. In our experiment, we regard the elementary-level documents as the simplified versions of the documents at intermediate and advanced levels. Specifically, we utilize the sentence pairs for the ELE-INT level to extract the textual form features of the intermediate-level documents and utilize the sentence pairs for the ELE-ADV level to extract the features of advanced-level documents. The prediction models are trained on the documents from the INT and ADV levels.

NewsEla. This corpus is created by Xu *et al.* (2015) to study how professional editors conduct text simplification. It consists of 1,130 news articles. Each article is rewritten 4 times for children at different grade levels by editors at the Newsela company. The statistics of the corpus are presented in Table 4. Xu *et al.* (2015) design a sentence alignment algorithm that pairs each sentence in a simplified text version with the closest matching sentence in a more complex version, utilizing overlapping word lemmas. In our experiments, we employ the aligned sentence pairs and follow a similar procedure to the experiments on the OneStopEnglish corpus. Specifically, we regard the articles at the Simp-4 level as simplified versions of those at the other four levels for feature extraction and build our classification model based on articles from these four levels.

4.2 English and multilingual corpora for readability assessment

Weebit. This corpus combines articles extracted from WeeklyReader[F] and the BBC Bitesize website.[G] The articles are classified into five readability levels according to the different targeted age groups. Levels 2, 3, and 4 consist of articles from the educational newspaper Weekly Reader, which covers a wide range of nonfiction topics, from science to current affairs. KS3 and GCSE consist of articles from the BBC Bitesize website, which contain educational material categorized into topics

[E] https://zenodo.org/record/1219041.
[F] http://www.weeklyreader.com
[G] http://www.bbc.co.uk/bitesize

Table 4. Statistics for the NewsEla corpus

Level	Number of documents	Total number of tokens	Number of words per document
Original	1130	1,301,767	1,152.01
Simp-1	1130	1,126,148	996.59
Simp-2	1130	1,052,915	931.78
Simp-3	1130	903,417	799.48
Simp-4	1130	764,103	676.2

Note: Simp-4 denotes the most simplified level and Simp-1 denotes the least simplified level.

Table 5. Statistics for the Weebit corpus

Level	Age group	Number of documents	Number of sentences per document
Level2	7–8	625	23.41
Level3	8–9	625	23.28
Level4	9–10	625	28.12
KS3	10–14	625	22.71
GCSE	14–16	625	27.85

Table 6. Statistics for the Cambridge English exam data

Level	Exam	Number of documents	Number of sentences per document
A2	KET	60	14.75
B1	PET	60	19.48
B2	FCE	71	38.07
C1	CAE	67	45.76
C2	CPE	69	39.97

that roughly match school subjects in the UK. Each level of the corpus has 625 documents, for a total of 3,125 documents. The statistics of the corpus are presented in Table 5.

Cambridge. The Cambridge Exam Corpus[H] was created by Xia *et al.* (2016). This corpus categorizes document readability according to Cambridge English Exam levels. The tests are designed for second-language (L2) learners at Common European Framework of Reference levels A2-C2 (CEFR) (Xia *et al.* 2016). The five readability levels KET, PET, FCE, CAE, and CPE correspond to CEFR levels A1 (breakthrough), B1 (threshold), B2 (vantage), C1 (effective operational proficiency), and C2 (mastery), respectively. The statistics of the corpus are presented in Table 6.

VikiWiki-Es/Fr. The VikiWiki corpus was created by Azpiazu and Pera (2019). It contains all documents retrieved from Vikidia, together with their corresponding Wikipedia entries. The articles are uniformly distributed between two levels of readability: simple and complex. There are a total of 70,514 documents in six different languages, with 8,390 in Spanish and

[H] http://www.cl.cam.ac.uk/oemx223/cedata.html

23,648 in French. At the simple level, there are on average 16 sentences and 303 words per document, and at the complex level, there are on average 217 sentences and 6,036 words per document. In our work, VikiWiki-Fr (French) and VikiWiki-Es (Spanish) are used for multilingual experiments.

For the three corpora mentioned above, we adopt a 10-fold cross-validation approach in our experiments. Specifically, the documents in the corpora are divided into 10 folds. The classification model is run 10 times on different folds of documents. Each time, the model is trained on nine folds and tested on the 10-th fold. The results of the 10 runs are averaged to generate the performance evaluation.

5. Experimental design

We design four experiments to evaluate the efficacy of our proposed textual form features in assessing readability. *Experiment 1* aims to examine the effectiveness of these features in characterizing text readability. *Experiment 2* seeks to investigate the contributions of the proposed textual form features in making predictions by examining their importance. *Experiment 3* aims to explore the applicability of these features in different languages. *Experiment 4* focuses on evaluating the impact that simplification tools on the performance of these features.

5.1 Experiment 1: examining the feature effectiveness in characterizing text readability

In this experiment, we first conduct a preliminary validation of the feature effectiveness on parallel corpora. Then, we evaluate our proposed textual form features by comparing them with other types of features using two traditional readability corpora. Finally, to explore the contributions of the proposed features to the field of readability, we integrate our proposed features into state-of-the-art models and assess whether their inclusion improves the model's performance.

5.1.1 Preliminary validation of feature effectiveness on parallel corpora.

We conduct experiments on two parallel corpora, OneStopEnglish and NewsEla, to preliminary validate the effectiveness of our proposed textual form features on parallel corpora. As mentioned in Section 4.1, the parallel corpora contain numerous aligned sentence pairs before and after manual simplification. We choose parallel corpora for our validation experiments for two main reasons. Firstly, the texts within these corpora are manually simplified, which guarantees a high quality of simplification and, consequently, ensures the extraction of high-quality features. Second, because these corpora already contain pre- and post-simplification sentence pairs, we may simply extract our proposed features without the need for simplification tools.

Figure 7 illustrates the experimental design we used. As shown in the figure, in a parallel corpus, each document is represented in different versions that correspond to different levels of readability. For example, Level 1 corresponds to the most complex version, while Level 5 corresponds to the most simplified version. We use the most simplified version to facilitate the process of extracting features, while employing documents from other levels for model training and prediction. More specifically, for each document in the training set, we find its corresponding simplified version in Level 5 and utilize the aligned sentence pairs in the two documents to extract our proposed textual form features. In addition, we utilize the Coh-Metrix tool to extract fundamental linguistic features, which will be detailed in the following description of experiments. Next, we implement a step-by-step method to build the feature sets by incorporating our proposed textual form features into the existing baseline features. Finally, we use the feature sets to train classification models and examine whether the inclusion of our proposed features leads to an enhancement in the model's performance.

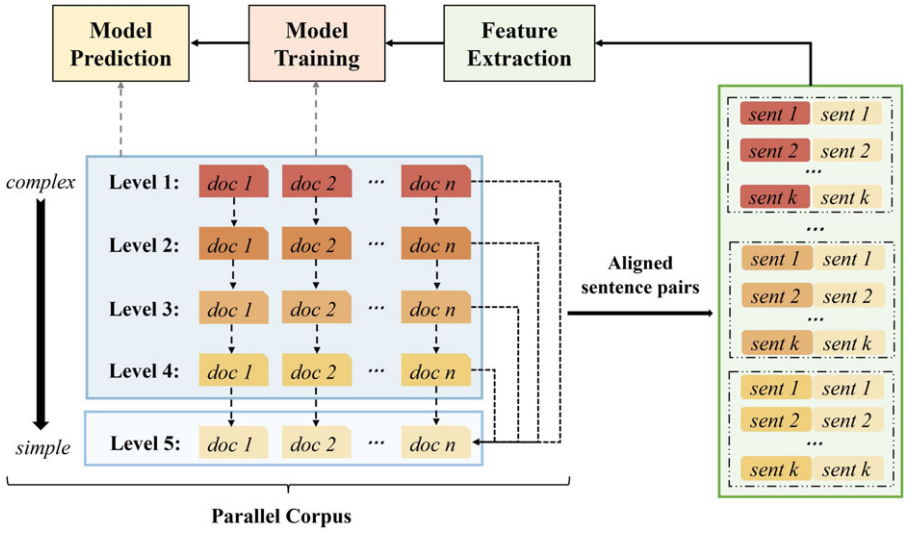

Figure 7. Diagram of the experimental design for validation on the parallel corpus. We use documents from the most simplified level in the parallel corpus to facilitate feature extraction, and we use documents from the other levels for training the model and making predictions. We select the most simple version of each document in the training set and use the aligned sentence pairs in the two texts to extract our proposed textual form features.

5.1.2 Investigating and comparing feature capabilities in characterizing text readability.
In this experiment, we test different types of features and their combinations in terms of their ability to characterize text readability. In addition to our proposed textual form features, we also select two other types of features for comparison: current representative handcrafted features (inner semantic-based features) and neural features. In the following, we will first introduce the two types of features and then describe our experimental procedure in detail.

Existing handcrafted features mainly focus on the inner semantic-based textual characteristics, while our proposed textual form features concentrate on the external complexity of the texts. To analyze the effects of inner and outer textual aspects on text readability, we utilize the Coh-Metrix tool to extract features as a representative of existing inner semantic-based features. According to Crossley *et al.* (2008), Coh-Metrix is a synthesis of the advances in conventional readability features. It was designed with the goal of providing a means to match textbooks appropriately to the targeted students (Graesser *et al.* 2004). It is a computational tool that measures text difficulty from an inner textual perspective, such as discourse cohesion, semantic and conceptual features. The features are categorized into eleven groups based on their linguistic properties. A total of 108 features are extracted. For convenience, we denote the extracted 108 features as Coh-Metrix features in the following sections.

In addition to handcrafted features, neural networks have recently shown powerful capability in capturing high-level textual linguistic information. Among the neural networks, the representative one is the pre-trained language model BERT. Thus, we leverage BERT to extract neural features. The experimental aim is to investigate the strengths and weaknesses of handcrafted and neural features. During the process of extracting neural features, we initially employ the BERT tokenizer to split the text into a sequence of tokens. The length of the input sequences is set to 512. If the length of the token sequence is less than 512, we pad it to 512 with the token [*PAD*]; If it is longer than 512, the part that exceeds the given length is truncated. Next, we append the token [*CLS*] to the beginning of the sequence and the token [*SEP*] at the end of the sequence. By inputting the token sequence, we acquire the associated token embeddings, segmentation embeddings, and position embeddings. Subsequently, we perform element-wise addition (Kenton and

Toutanova 2019) on the three embeddings and feed them into the pre-trained BERT model. The output of token [*CLS*] is used as the extracted neural features, which are 768-dimensional vectors.

The experimental procedure contains two main steps:

Step 1: Extraction of features. The two corpora used in this experiment are Weebit and Cambridge. Both corpora contain five readability levels. On each corpus, we extract three types of features for texts at all five levels. Specifically, we extract our proposed eight textual form features to capture the texts' outer form characteristics following the methods described in Section 3.3. As described earlier, we utilize the Coh-Metrix tool to extract inner semantic-based features and leverage the pre-trained BERT model to extract neural features.

Step 2: Training the model and making predictions. Using the extracted features, a classification model is trained to predict the readability label for the given text. Our prediction models consist of Random Forest and SMO. The specific details of the model setup can be found in section 3.4. In addition to using individual types of features for model training, we also investigate two feature combinations: 1) The integration of inner semantic-based features and outer textual form features; 2) The combination of handcrafted features and neural features. As for the incorporation of neural features, we concatenate the extracted 768-dimensional vectors with all the extracted handcrafted features, which include 108 inner semantic-based features and ten outer textual form features. The resulting vectors are then considered a combination of handcrafted and neural features, with a dimensionality of 884. Finally, we leverage the trained models to make predictions about the readability of text and evaluate the model's performance in terms of accuracy, F1 score, precision, and recall.

5.1.3 Examining the feature contributions to enhancing the performance of SOTA models.

Current state-of-the-art (SOTA) methods for readability assessment predominantly employ pre-trained language models based on deep learning approach (Deutsch *et al.* 2020; Lee *et al.* 2021; Liu and Lee 2023). The purpose of this experiment is to validate our proposed features and determine their ability to improve the SOTA performance in the field of readability assessment.

We choose the hybrid models (Lee *et al.* 2021; Liu and Lee 2023) from the current state-of-the-art models to ensure a more efficient comparison. These hybrid models combine handcrafted features with fine-tuned pre-trained models, including BERT, RoBERTa, XLNet, and BART. Since the model of Liu and Lee (2023) is designed for sentence-level readability assessment that differs from our task setup, we choose to make comparisons with the hybrid models for passage-level readability assessment from their previous work (Lee *et al.* 2021). This work utilizes corpora that align with ours, specifically Weebit and Cambridge. In this experiment, we integrate our proposed features into their hybrid models and compare the results with those reported in their paper. More precisely, we first reran their code to fine-tune the pre-trained models and obtain the transformer predictions. Next, we concatenate our proposed ten features with their extracted, handcrafted features, namely T1 and P3 feature sets. Finally, we train classification models using the combined handcrafted features and transformer predictions, in accordance with their approach.

5.2 Experiment 2: investigating the feature contributions in model prediction

Beyond gaining good model performance for readability prediction, it is also important to find out which features are indicative of these predictions and the extent to which these features contribute to the model's performance (Zien *et al.* 2009). Therefore, we design this experiment to further analyze the impact of our proposed textual form features on the model's prediction. This experiment is a continuation of *Experiment 1* and focuses on all the handcrafted features. To be specific, we train the model using a combination of 108 inner semantic-based features and ten outer textual form features and analyze their contributions to the Weebit and Cambridge corpora.

We employ the mean impurity decrease method to analyze the contributions of features. We use this strategy because Mean Decrease in Impurity (MDI) is a measurement of feature importance in decision tree models, such as Random Forest. More precisely, we first calculate the MDI score for each feature by averaging the total decrease in node impurities over all the trees from splitting on the respective features (Han, Guo, and Yu 2016). Regarding the node impurity, we employ the Gini index, which is calculated using the formula below:

$$\textbf{Gini} = \sum_{i=1}^{C} p(i) * (1 - p(i)) \tag{17}$$

where C is the total number of classes, i represents the i^{th} class, and $p(i)$ is the probability of picking an instance with class i. Then, we rank the importance of the total of 118 handcrafted features according to their MDI scores.

5.3 Experiment 3: exploring the applicability of the proposed features across different languages

The goal of this experiment is to explore the applicability of our proposed features to languages other than English. Previous studies (Scarton *et al.* 2017; Finnimore *et al.* 2019; Stodden and Kallmeyer 2020) have shown that text simplification exhibits a certain level of consistency across different languages. Therefore, it is reasonable to assume that our proposed text-simplification-based textual form features could be effective across various languages. To test this hypothesis, we apply our proposed features to texts written in Spanish and French and evaluate their efficacy in this multilingual setting. This experiment utilizes the VikiWiki-Es/Fr corpus.

As mentioned in Section 4.2, this corpus contains two sub-corpora: VikiWiki-Es (Spanish) and VikiWiki-Fr (French). Each sub-corpus has two readability levels, namely simple and difficult. Specifically, we first extract our proposed ten textual form features from texts at two levels of each sub-corpus. Then, we train classification models with our proposed features. At last, we evaluate the model's performance in terms of accuracy and compare our approach with both traditional and state-of-the-art strategies on this corpus. The compared strategies were implemented by Azpiazu and Pera (2019), and the brief introductions are as follows:

- **Flesch.** In this strategy, the traditional Flesch formula (Kincaid *et al.* 1975) is used to compute the readability score of each text. Due to the multilingual setting, this strategy considers the language-specific adaptations of the formula (Kandel and Moles 1958; Fernández Huerta, 1959). The employed framework categorizes a given text as either simple or complicated by mapping its Flesch score onto a binary value. This mapping is determined by the text's distance from the average score of texts in each respective category (Azpiazu 2017).
- **LingS3+RandomForest.** This is a current state-of-the-art feature engineering strategy. It adopts the classical machine learning method. In this strategy, the Random Forest classification model is trained based on a set of linguistic features introduced by Azpiazu (2017). We denoted this set of features as LingS3 in our paper. Due to the unavailability of the feature set of LingS3, we present their reported results (Azpiazu 2017).
- **Vec2Read.** This is a state-of-the-art strategy based on deep learning methods. It employs a multi-attentive recurrent neural network architecture for evaluating the readability of texts in multiple languages (Azpiazu and Pera 2019). In this strategy, a recurrent neural network is leveraged to capture text structure, and a multi-attention mechanism is utilized to incorporate textual information related to syntax and morphology.

Table 7. Simplification examples from the Cambridge corpus using MUSS and ReWordify tools

Simp.tool	Example Text
Example 1	
Original	I actually look rather good in jeans. So I rarely wear them. I don't want to get mixed up with the wrong sort of person.' So runs the logic of Isabel, 'our heroine' in this refreshingly zany novel.
MUSS	I look pretty good in jeans, so I don't want to get mixed up with the wrong sort of person." So runs the logic of Isabel, "our heroine" in this refreshingly zany novel.
ReWordify	I actually look rather good in jeans. So I rarely wear them. I don't want to get mixed up with the wrong sort of person.' So runs the logic of Isabel, 'our female hero' in this refreshingly wacky novel.
Example 2	
Original	An art student of 31, devotee of historical romances, she seeks Mr Right but is concerned lest her knobbly knees ruin her chances. Hope, however, springs eternal—which proves to be her salvation, and that of other people in the story.
MUSS	An art student of 31, she seeks Mr Right but is concerned lest her knobbly knees ruin her chances. She is also a fan of historical romances, but those aren't the only things in her life.
ReWordify	An art student of 31, fan of historical romances, she looks for Mr Right but is concerned lest her knobbly knees ruin her chances. Hope, however, springs never-ending—which proves to be her (saving or protecting someone from sin or harm), and that of other people in the story.

Note: "Simp.tool" denotes the simplification tool.

5.4 Experiment 4: analyzing the impact of simplification tools on the performance of proposed features

The purpose of this experiment is to assess how different simplification tools affect the capabilities of features to enhance model performance.

In addition to the MUSS simplification tool (Martin *et al.* 2020) mentioned in Section 3.2, there are several more text simplification tools that can be utilized. Among these tools, we select Rewordify[I] for comparison, which is a free and efficient online tool. Morales *et al.* Morales *et al.* (2019) emphasize that this tool enhances reading, learning, and teaching experiences for English texts. It focuses on lexical-level simplification and is capable of contextually modifying lexical expressions within texts.

Table 7 presents a comparison of text simplifications by MUSS and ReWordify using two examples from the Cambridge corpus. According to the table, MUSS frequently condenses and modifies the original sentence phrasing, whereas ReWordify preserves the sentence structure, emphasizing the simplification of individual words and phrases. For example, in the second example, MUSS simplifies and condenses the description by removing phrases like "*devotee of*" ; ReWordify clarifies and simplifies specific expressions, such as changing "*springs eternal*" to "*springs never-ending.*" These examples suggest that MUSS focuses on structural simplification, which can involve condensing information and modifying sentence structure. On the other hand, ReWordify primarily focuses on simplifying the vocabulary by substituting complex words with simpler ones. Both methods aim to make the text more readable and understandable, but they employ different strategies to achieve this goal, each with its own advantages and target use cases.

In this experiment, we use the Rewordify tool for feature extraction. Following the same experimental design as conducted on the Weebit and Cambridge corpora, we train Random Forest and SMO classification models using the extracted features. Finally, we compare the performance of

[I]https://rewordify.com/

Table 8. Performance comparison of RandomForest and SMO trained with baseline Coh-Metrix features and added textual form features on OneStopEnglish and NewsEla parallel corpora

		RandomForest				SMO			
Parallel corpora	Feature set	ACC	F1	P	R	ACC	F1	P	R
OneStopEnglish	Coh-Metrix(108)	0.794	0.793	0.795	0.794	0.894	0.894	0.894	0.894
	Coh-Metrix(108) + LenRatio(1)	0.804	0.804	0.804	0.804	0.902	0.902	0.902	0.902
	Coh-Metrix(108) + Overlap(3)	0.802	0.802	0.802	0.802	0.902	0.902	0.903	0.902
	Coh-Metrix(108) + Similarity(6)	0.915	0.915	0.915	0.915	0.947	0.946	0.947	0.947
	Coh-Metrix(108) + TextFormFeat(10)	**0.918**	**0.918**	**0.919**	**0.918**	**0.947**	**0.947**	**0.947**	**0.947**
NewsEla	Coh-Metrix(108)	0.838	0.840	0.842	0.839	0.848	0.849	0.850	0.848
	Coh-Metrix(108) + LenRatio(1)	0.856	0.857	0.859	0.856	0.876	0.876	0.877	0.876
	Coh-Metrix(108) + Overlap(3)	0.855	0.856	0.858	0.855	0.876	0.877	0.877	0.876
	Coh-Metrix(108) + Similarity(6)	**0.888**	**0.889**	**0.890**	**0.888**	0.896	0.896	0.895	0.896
	Coh-Metrix(108) + TextFormFeat(10)	0.885	0.885	0.886	0.885	**0.896**	**0.896**	**0.896**	**0.896**

Note: The best performance on the corresponding parallel corpora is highlighted in bold.

models in three different settings: 1) without using any simplification pipeline, 2) utilizing MUSS for simplification, and 3) employing Rewordify for simplification.

6. Results and discussion

This section presents the findings and discussion of the four experiments mentioned in Section 5. For convenience, we abbreviate the names of the feature sets used in our experiments: TextFormFeat (10) refers to our proposed ten textual form features; Similarity (6) refers to the six similarity-based features, including *CosSim*, *JacSim*, *EditDis*, *BertSim*, *BleuSim* and *MeteorSim*; Overlap (3) refers to the three overlap-based features, including *NounOlap*, *VerbOlap* and *AdjOlap*; LenRatio (1) refers to the length-ratio feature *LenRatio*; Coh-Metrix (108) refers to the 108 Coh-Metrix features; BERTFeat (768) refers to the 768-dimensional neural features extracted from the pre-trained language model BERT. The four evaluation metrics are also represented in their abbreviated forms, where ACC, F1, P, and R denote Accuracy, F1 score, Precision, and Recall, respectively.

6.1 Discussion of feature effectiveness

This section presents the experimental results of *Experiment 1*. In the following, we first present the preliminary validation results on parallel corpora. Then, we make a discussion of different types of readability features based on the experimental results. Finally, we present the experimental results of integrating our proposed features into the existing SOTA models.

The validation experiments preliminarily confirm the efficacy of our proposed features. From Table 8, it can be seen that adding our proposed textual form feature sets leads to a significant improvement in model performance. Specifically, on the OneStopEnglish corpus, Coh-Metrix (108) + TextFormFeat (10) achieves the best performance in both classification models; RandomForest reaches an accuracy of 0.918 and F1 score of 0.918, while SMO achieves an accuracy of 0.947 and F1 score of 0.947. In comparison to using Coh-Metrix (108) alone, when adding our proposed TextFormFeat (10), the greatest enhancement observed in RandomForest is 12.5% (F1 score), and in SMO is 5.3% (F1 score). On the NewsEla corpus, adding Similarity (6) results

Table 9. Performance comparison of RandomForest with different types of feature sets on Weebit and Cambridge

		Weebit				Cambridge			
Feature type	Feature set	ACC	F1	P	R	ACC	F1	P	R
inner	Coh-Metrix(108)	0.744	0.744	0.743	0.744	0.731	0.727	0.730	0.731
outer	TextFormFeat(10)	0.547	0.545	0.544	0.547	0.514	0.514	0.519	0.514
inner+outer	Coh-Metrix(108) + LenRatio(1)	0.763	0.763	0.764	0.763	0.731	0.728	0.730	0.731
	Coh-Metrix(108) + Overlap(3)	0.755	0.754	0.754	0.755	0.767	0.765	0.766	0.767
	Coh-Metrix(108) + Similarity(6)	**0.771**	**0.770**	**0.770**	**0.771**	**0.795**	**0.794**	**0.794**	**0.795**
	Coh-Metrix(108)+ TextFormFeat(10)	0.768	0.767	0.767	0.768	0.773	0.771	0.772	0.773
neural	BERTFeat(768)	0.794	0.794	0.795	0.794	0.598	0.589	0.588	0.598
neural+inner +outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	**0.816**	**0.816**	**0.817**	**0.816**	**0.689**	**0.681**	**0.685**	**0.689**

Note: The 'inner' and 'outer' types belong to handcrafted features, while the 'neural' type belongs to neural features. The upper part of the table presents a comparison of solely handcrafted features, with the best performance among these handcrafted-only feature sets highlighted in **bold**. The lower part of the table includes neural features into comparison, with the best performance highlighted in **bold**.

in the highest performance for RandomForest, attaining an accuracy of 0.888 and an F1 score of 0.889; and for SMO, adding TextFormFeat (10) yields the best results, with an accuracy of 0.896 and F1 score of 0.896. These results indicate that different models may have varying preferences for the features; however, overall, the inclusion of our proposed features leads to the performance enhancement for both models on two corpora, thus giving a preliminary validation of the efficacy of our proposed features.

We compare the model's performance based on different types of feature sets on two traditional readability corpora for each classification model mentioned in Section 3.4. The experimental results further demonstrate the effectiveness of our proposed features for readability assessment. Tables 9 and 10 report the model performance of RandomForest and SMO, respectively. It is evident that adding our proposed TextFormFeat (10) to Coh-Metrix (108) enhances the performance of both models on the two corpora. For RandomForest, the model's performance in terms of accuracy and F1 score improves by 2.4% and 2.3% on the Weebit corpus and by 4.2% and 4.4% on the Cambridge corpus (line 1 and line 6 in Table 9). For SMO, the model's performance in terms of accuracy and F1 score increases by 2.3% and 2.2% on the Weebit corpus and by 1.2% and 1.5% on the Cambridge corpus (line 1 and line 6 in Table 10). Since only ten features have brought about this considerable improvement, we may safely draw the conclusion that our proposed textual form features are effective. Moreover, it can also be seen that when the three feature subsets Similarity (6), Overlap (3), and LenRatio (1) are added to Coh-Metrix (108) individually, the model performances are all improved. This further demonstrates the effectiveness of each subset of textual form features. Interestingly, for RandomForest, the addition of the feature subset Similarity (6) achieves the best performance on both corpora, demonstrating the effectiveness and superiority of these similarity-based features. Previous research (Martin *et al.* 2020) points out that overlap-based metrics are not the best for evaluating text simplification tasks. Despite this, we observe that incorporating overlap-based features into our model markedly improves readability assessment predictions. While the specific reasons for this enhancement are not fully clear, we hypothesize that it may be due to the differentiation between the use of Overlap as a predictive feature within the learning algorithm and its use as an evaluative metric.

It can also be concluded that both inner semantic-based features and outer textual form features are essential in characterizing text readability. With the inner semantic-based features, both

Table 10. Performance comparison of SMO with different types of feature sets on Weebit and Cambridge corpora

		Weebit				Cambridge			
Feature type	Feature set	ACC	F1	P	R	ACC	F1	P	R
inner	Coh-Metrix(108)	0.761	0.761	0.761	0.761	0.740	0.736	0.738	0.740
outer	TextFormFeat(10)	0.533	0.535	0.541	0.533	0.432	0.431	0.434	0.432
inner+outer	Coh-Metrix(108) + LenRatio(1)	0.763	0.763	0.764	0.763	0.731	0.728	0.730	0.731
	Coh-Metrix(108) + Overlap(3)	0.767	0.768	0.768	0.767	0.737	0.735	0.737	0.737
	Coh-Metrix(108) + Similarity(6)	0.783	0.783	0.783	0.783	0.749	0.747	0.751	0.749
	Coh-Metrix(108) + TextFormFeat(10)	**0.784**	**0.783**	**0.783**	**0.784**	**0.752**	**0.751**	**0.755**	**0.752**
neural	BERTFeat(768)	0.834	0.834	0.834	0.834	0.665	0.662	0.665	0.665
neural+inner +outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	**0.845**	**0.845**	**0.845**	**0.845**	**0.740**	**0.741**	**0.745**	**0.740**

Note: The 'inner' and 'outer' types belong to handcrafted features, while the 'neural' type belongs to neural features. The upper part of the table presents a comparison of solely handcrafted features, with the best performance among these handcrafted-only feature sets highlighted in **bold**. The lower part of the table includes neural features into comparison, with the best performance highlighted in **bold**.

models exhibit good performance on the Weebit and Cambridge corpora. As detailed in Tables 9 and 10, using Coh-Metrix(108) as inner features, RandomForest attains an F1 score of 0.744 on Weebit and 0.727 on Cambridge (line 1 in Table 9), SMO attains F1 scores of 0.761 and 0.736 on the respective corpora (line 1 in Table 10). These results indicate that the inner semantic-based features play a crucial role in readability assessment. Regarding the outer textual form features, it's evident that while the ten features alone may not yield impressive results, they can boost the model when used as a complement to the inner textual features. With RandomForest, combining Coh-Metrix (108) with Similarity (6) results in an F1 score of 0.770 on Weebit and 0.794 on Cambridge. For SMO, the addition of TextFormFeat (10) to Coh-Metrix (108) leads to F1 scores of 0.783 and 0.751 on the respective corpora. Hence, the combination of inner semantic-based and outer textual form features is most likely to improve the model's performance.

In addition to handcrafted features, we also investigate neural features and observe that neural features sometimes are not necessarily effective. In Tables 9 and 10, we can see that neural features perform pretty well on the Weebit corpus; that is, the combination of handcrafted features and neural features (neural+inner+outer) yields the best performance, with RandomForest reaching an accuracy and F1 score of 0.816 (line 8 in Table 9), and SMO attaining an accuracy and F1 score of 0.845 (line 8 in Table 10). Complementary to our findings, recent research, such as that conducted by Wilkens *et al.* (2024), explores the efficacy of combining linguistic features with transformer-based neural features. Their research supports the idea that integrating diverse types of features—traditional linguistic and advanced neural—can significantly improve the performance of readability assessment systems. However, we can also see that neural features are not as effective on the Cambridge corpus. As demonstrated, when trained solely with neural features (BERTFeat (768)), the RandomForest model achieves only an accuracy of 0.598 and F1 score of 0.589 (line 7 in Table 9). Even though concatenating handcrafted features (Coh-Metrix (108)+TextFormFeat (10)), the model's performance improves to an accuracy of 0.689 and an F1 score of 0.681 (line 8 in Table 9), yet it still does not surpass the performance achieved with the four sets of inner and outer features. Likewise, for SMO, neural features also show subpar performance on the Cambridge corpus. These results suggest the instability of the performance of neural features. In contrast, as can be seen from the two tables, handcrafted features (especially the combination of inner semantic-based features and outer textual form features) have a stable and effective performance on both corpora. For example, using Coh-Metrix(108)+TextFormFeat(10)

Table 11. Performance comparison before and after integrating our proposed features into SOTA models

Model	Weebit				Cambridge			
	ACC	F1	P	R	ACC	F1	P	R
BERT + T1_GB	0.895	0.895	0.897	0.897	0.687	0.682	0.732	0.687
BERT + T1_GB + TextFormFeat(10)	**0.974**	**0.974**	**0.975**	**0.974**	**0.791**	**0.788**	**0.791**	**0.819**
RoBERTa+T1_RF	0.902	0.902	0.903	0.903	0.763	0.752	0.792	0.753
RoBERTa+T1_RF + TextFormFeat(10)	**0.952**	**0.952**	**0.953**	**0.952**	**0.779**	**0.773**	**0.771**	**0.779**
BART + T1_RF	0.905	0.905	0.905	0.904	0.727	0.727	0.76	0.727
BART + T1_RF + TextFormFeat(10)	**0.952**	**0.952**	**0.953**	**0.952**	**0.777**	**0.768**	**0.766**	**0.776**
XLNet + P3_RF	0.892	0.892	0.893	0.892	0.687	0.676	0.710	0.687
XLNet + P3_RF + TextFormFeat(10)	**0.948**	**0.948**	**0.949**	**0.948**	**0.782**	**0.778**	**0.777**	**0.782**

Note: T1 and P3 represent the handcrafted feature sets used in hybrid models (Lee *et al.* 2021). RF denotes RandomForest, GB denotes XGBoost. Bold indicates the best performance of the model before and after integrating our proposed TextFormFeat(10).

as inner and outer features, the RandomForest model attains an accuracy of 0.768 on the Weebit corpus and 0.773 on the Cambridge corpus (line 6 in Table 9); meanwhile, the SMO model demonstrates an accuracy of 0.784 on the Weebit corpus and 0.752 on the Cambridge corpus (line 6 in Table 10). Moreover, it is claimed that handcrafted features are better than neural features in terms of their interpretability. Though we cannot give an exact explanation of vectors extracted from neural networks (e.g., the vector of 768 dimensions extracted from BERT), we can gain a good understanding of what information is extracted by handcrafted features (e.g., textual form features capture the outer form characteristics of the text).

The experimental results in Table 11 demonstrate that our proposed features are valid and can enhance the state-of-the-art performance in the field of readability assessment. It is evident that integrating our proposed ten textual form features into the existing four SOTA hybrid models leads to a significant performance improvement. On the Weebit corpus, the model BERT + T1_GB + TextFormFeat (10) achieves the best performance (ACC = 0.974, F1 = 0.974), marking an increase of 7.9% in both ACC and F1 compared to the SOTA model BERT + T1_GB. On the Cambridge corpus, the same model, BERT + T1_GB + TextFormFeat (10), also reaches the highest performance (ACC = 0.791, F1 = 0.788), with an ACC increase of 10.4% and an F1 increase of 10.6% compared to the SOTA model BERT + T1_GB. These results fully demonstrate that our proposed features can enhance the performance of existing SOTA models, contributing significantly to the field of readability.

Additionally, we conduct a performance comparison of models incorporating our features with recent models. We select four existing readability models, which are respectively SVM (Xia *et al.* 2016), BERT + SVM (Deutsch *et al.* 2020), BERT (Martinc *et al.* 2021), and BART + T1_RF hybrid (Lee *et al.* 2021). The reason we choose these works is that they all utilize the Weebit corpus. Since not all studies report all four metrics, we report the accuracy metric, which was used across all works. The results, displayed in Table 12, indicate that our models significantly outperform all existing readability assessment models on the Weebit corpus. Specifically, our BART + T1_RF hybrid + TextFormFeat (10) surpasses the previous best model by 4.7%, while our BERT + T1_GB hybrid + TextFormFeat (10) shows an even more substantial improvement of 6.9% over the same benchmark. The superior performance of our models over established techniques substantiates the effectiveness and applicability of our approach in contemporary readability research.

Table 12. Performance comparison of our models and existing readability assessment models on the WeeBit corpus

	Readability Assessment Model	Accuracy
Existing models	SVM (Xia *et al.* 2016)	0.803
	BERT + SVM (Deutsch *et al.* 2020)	0.838
	BERT (Martinc *et al.* 2021)	0.857
	BART + T1_RF hybrid (Lee *et al.* 2021)	0.905
Our models	BART + T1_RF hybrid + TextFormFeat(10)	0.952
	BERT + T1_GB hybrid + TextFormFeat(10)	**0.974**

Note: As Accuracy is the common evaluation metric reported in the papers of these existing models, we report Accuracy in this table.

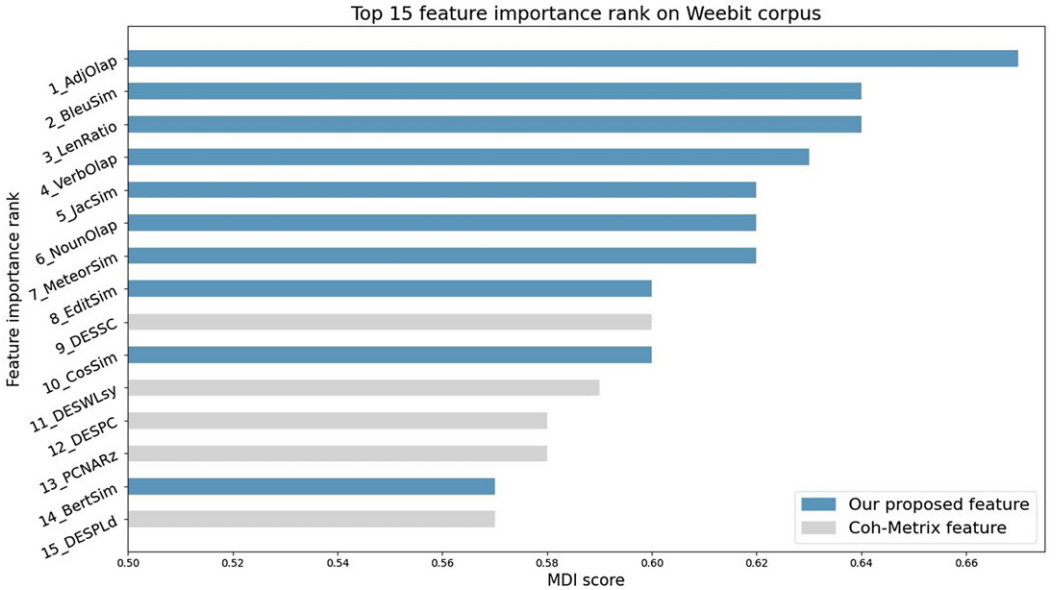

Figure 8. Top 15 feature importance ranks on the Weebit corpus. On the *x* axis, the MDI scores are reported. On the *y* axis, features ranked in the top 15 are reported, where the rank order increases from top to bottom. The blue bars represent our proposed features. The gray bars represent Coh-Metrix features.

6.2 Analysis of feature contributions

This section presents the experimental results in *Experiment 2*. We rank the feature importance of the total 118 features (108 Coh-Metrix features and our proposed ten textual form features) according to their MDI scores. Rank lists of the top 15 important features on Weebit corpus and Cambridge corpus are separately presented in Figures 8 and 9. On the *x* axis, the MDI scores are reported. On the *y* axis, features ranked in the top 15 are reported, where the rank order increases from top to bottom (the rank 1 writes at the top, the rank 15 writes at the bottom). The blue bars represent our proposed textual form features. The gray bars represent the Coh-Metrix features.

It can be concluded from the two figures that our proposed textual form features play a vital role in the model's prediction. As can be seen from the figures, the MDI scores of our proposed ten textual form features rank highly among the total 118 features, mostly occupying the top 15 rank lists on the two corpora.

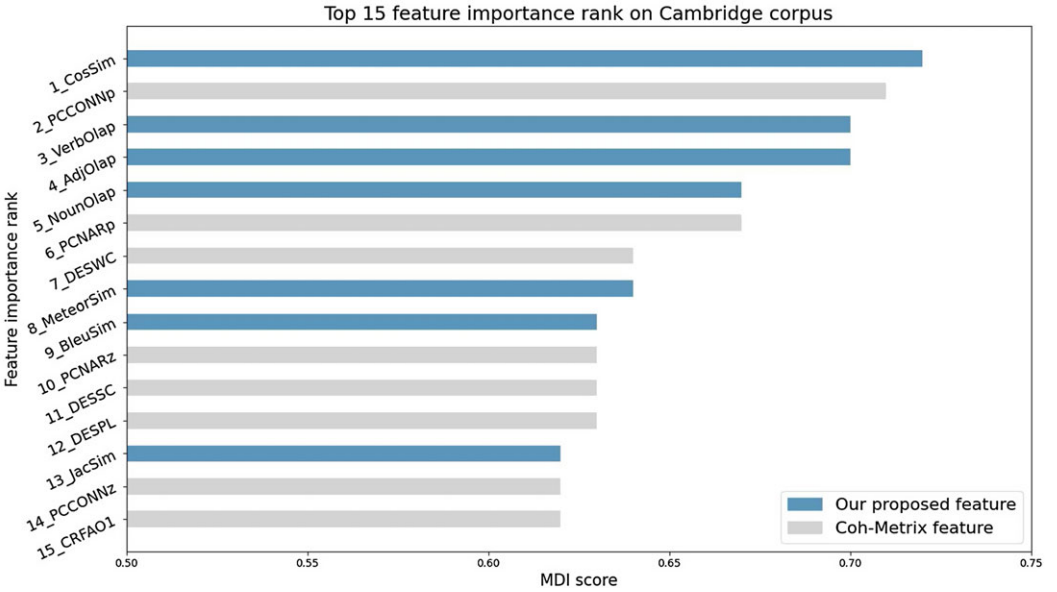

Figure 9. Top 15 feature importance ranks on the Cambridge corpus. On the *x* axis, the MDI scores are reported. On the *y* axis, features ranked in the top 15 are reported, where the rank order increases from top to bottom. The blue bars represent our proposed features. The gray bars represent Coh-Metrix features.

To be specific, it can be seen from Figure 8 that on Weebit corpus, our proposed features occupy the top eight of the importance rank list. And what stands out is that all ten proposed textual form features are in the top 15 ranks on this corpus. Also, in Figure 9, we can see that on Cambridge corpus, seven out of the top 15 features in the rank list are our proposed textual form features, and the top one important feature is our proposed *CosSim*. It is noteworthy that our proposed features *CosSim*, *JacSim*, *BleuSim*, *MeteorSim*, *VerbOlap*, *NounOlap*, and *AdjOlap* are in the top 15 rank lists on both corpora. Taken together, these results suggest that the proposed ten features have made significant contributions to the model's prediction.

From Figure 8 and 9, we can also conclude that some of the Coh-Metrix features are essential in the model prediction as well. As introduced in Section 5.1, the 108 Coh-Metrix features were divided into 11 categories according to their linguistic properties (McNamara *et al.* 2014). The detailed descriptions of the Coh-Metrix features are presented in Appendix B. From the figures, it can be seen that the highly ranked Coh-Metrix features are mainly distributed in three categories, which are Descriptive, Text Easability Principle Component Scores, and Referential Cohesion. Interestingly, four out of the five Coh-Metrix features that rank among the top 15 on the Weebit corpus are descriptive features. And three descriptive features also appear in the top 15 list on Cambridge corpus as well. As introduced by McNamara *et al.* (2014), descriptive features include the basic statistics of a sentence or paragraph in the given text. This result suggests that the descriptive features are more effective among the 108 Coh-Metrix features and have comparable contributions to our proposed textual form features.

6.3 Results in multilingual setting

This section presents the experimental results of *Experiment 3*. Our strategy is to train the classification model Random Forest based on our proposed textual form features in a multilingual setting (in Spanish and French). We compare our strategy with traditional and state-of-the-art

Table 13. The classification accuracy on the VikiWiki-Es/Fr corpus

Strategies	VikiWiki-Es (Spanish)	VikiWiki-Fr (French)
Flesch	0.687	0.670
LingS3 + RandomForest	0.792	0.842
Vec2read	0.847	**0.884**
TextFormFeat(10) + RandomForest (Ours)	**0.874**	0.870

strategies. Table 13 presents the model performance of the strategies on VikiWiki-Es (Spanish) and VikiWiki-Fr (French).

From the results in Table 13, we can conclude that our proposed textual form features can be applied to different languages. As can be seen from the table, based on our proposed features, our strategy achieves the best performance among all the strategies (accuracy of 0.874 on VikiWiki-Es (Spanish); accuracy of 0.870 on VikiWiki-Fr (French)). What is unexpected is that only ten features have yielded such good model performance on Spanish and French corpora and even better performance than on English corpora. To make a fair comparison, we compare our proposed features with the existing multilingual features (LingS3) based on the same prediction model, Random Forest. Results show that our proposed TextFormFeat (10) yields better model performance than LingS3: the accuracy improves from 0.792 to 0.874 on VikiWiki-Es (Spanish), increasing by 8.2%; and improves from 0.842 to 0.870 on VikiWiki-Fr (French), increasing by 2.8%. These improvements further demonstrate the superiority of our proposed features in multilingual readability assessment. Moreover, our strategy even achieves comparable performance with the neural network-based state-of-the-art strategy Vec2Read (see lines 3-4 in Table 13). This suggests that our proposed textual form features are especially powerful in characterizing text readability in this corpus. Overall, our proposed features have been successfully applied to English, Spanish, and French, which confirms the effectiveness of our proposed features for different languages.

6.4 Results with different simplification tools

This section presents the experimental results of *Experiment 4*. We conduct experiments on the Weebit and Cambridge corpora, comparing model performance when the simplification pipeline in our method is either removed or replaced with different simplification tools. The results for the Weebit and Cambridge corpora are displayed in Table 14 and Table 15, respectively.

Firstly, it is observed that removing the simplification pipeline results in a decrease in model performance on both corpora, suggesting that incorporating the simplification pipeline into the model contributes to improving the model's performance. Specifically, for models based solely on handcrafted features, on the Weebit corpus (in Table 14), the absence of the simplification pipeline leads to an average performance decrease of 2.375% and 2.25% for the RandomForest and SMO models, respectively. On the Cambridge corpus (in Table 15), the most significant performance drop is seen in the RandomForest model, with an average decrease of 5.45%. When neural features are also considered, the results remain consistent. Across both corpora for both models, the performance with a simplification pipeline is better than without. This is particularly evident on the Cambridge corpus (in Table 15), where the RandomForest model with a simplification pipeline outperforms the one without an average of 12.325%, and the SMO model with a simplification pipeline exceeds the performance of the one without an average of 7.725%. In summary, the above results demonstrate the contribution of the simplification pipeline towards improving the model performance.

Table 14. Performance comparison of our method with the use of different simplification tools on Weebit corpus

Model	Feature type	Feature set	Simp. tool	ACC	F1	P	R
RandomForest	inner	Coh-Metrix(108)	-	0.744	0.744	0.743	0.744
	inner + outer	Coh-Metrix(108) + TextFormFeat(10)	MUSS	**0.768**	**0.767**	**0.767**	**0.768**
			Rewordify	0.764	0.764	0.764	0.764
	neural	BERTFeat(768)	-	0.794	0.794	0.795	0.794
	neural + inner + outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	MUSS	0.816	0.816	0.817	0.816
			Rewordify	**0.821**	**0.821**	**0.821**	**0.821**
SMO	inner	Coh-Metrix(108)	-	0.761	0.761	0.761	0.761
	inner + outer	Coh-Metrix(108) + TextFormFeat(10)	MUSS	**0.784**	**0.783**	**0.783**	**0.784**
			Rewordify	0.771	0.771	0.771	0.771
	neural	BERTFeat(768)	-	0.834	0.834	0.834	0.834
	neural+inner +outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	MUSS	0.845	0.845	0.845	0.845
			Rewordify	**0.860**	**0.860**	**0.860**	**0.860**

Note: "-" indicates no simplification pipeline used, "Simp. tool" denotes the simplification tool. Bold highlights the highest performance obtained using different simplification tools for each model.

Secondly, it is evident that the performance of models varies with the use of different simplification tools. In Table 14, on the Weebit corpus, for models based solely on handcrafted features, the MUSS simplification tool performs better, with RandomForest achieving an accuracy of 0.768 and SMO reaching 0.784; when neural features are included, the ReWordify simplification tool shows better performance, with RandomForest attaining an accuracy of 0.821 and SMO achieving 0.860. Table 15 shows that on the Cambridge corpus, the ReWordify simplification tool yields better results for the RandomForest model, with the best performance at an accuracy of 0.786; whereas for the SMO model, the MUSS simplification tool is more effective, with the highest performance at an accuracy of 0.740. Overall, although different simplification tools result in varied model performances, likely due to differences in the quality and preferences of the tools, using these tools for feature extraction enhances the model's performance. This suggests that our method is not limited by the choice of simplification tool and can be effectively applied to other simplification systems, but further investigation is still needed to determine if there is an optimal simplification tool.

7. Conclusion

In this paper, we introduce a new concept, namely textual outer form complexity, to provide a novel insight into text readability. The main idea of this insight is that the readability of a text can be measured by how difficult it is for the reader to overcome the distractions of exterior textual form. Based on this insight, we propose to construct a set of textual form features that reflect the complexity of the text's outer form, thereby facilitating the evaluation of its readability. We

Table 15. Performance comparison of our method with the use of different simplification tools on Cambridge corpus

Model	Feature type	Feature set	Simp. tool	ACC	F1	P	R
RandomForest	inner	Coh-Metrix(108)	-	0.731	0.727	0.730	0.731
	inner + outer	Coh-Metrix(108)+ TextFormFeat(10)	MUSS	0.773	0.771	0.772	0.773
			Rewordify	**0.786**	**0.783**	**0.783**	**0.785**
	neural	BERTFeat(768)	-	0.598	0.589	0.588	0.598
	neural + inner + outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	MUSS	0.689	0.681	0.685	0.689
			Rewordify	**0.720**	**0.711**	**0.716**	**0.719**
SMO	inner	Coh-Metrix(108)	-	0.740	0.736	0.738	0.740
	inner + outer	Coh-Metrix(108) + TextFormFeat(10)	MUSS	**0.752**	**0.751**	**0.755**	**0.752**
			Rewordify	0.746	0.746	0.747	0.746
	neural	BERTFeat(768)	-	0.665	0.662	0.665	0.665
	neural + inner + outer	BERTFeat(768) + Coh-Metrix(108) + TextFormFeat(10)	MUSS	**0.740**	**0.741**	**0.745**	**0.740**
			Rewordify	0.740	0.739	0.742	0.740

Note: "-" indicates no simplification pipeline used, "Simp. tool" denotes the simplification tool. Bold highlights the highest performance obtained using different simplification tools for each model.

propose two studies in this work to achieve this goal. *Study 1* focuses on the method of constructing and extracting features. This study quantifies the differences between pre-simplified and post-simplified texts to construct textual form features. We construct three subsets of textual form features, namely similarity-based, overlap-based, and length ratio-based features, corresponding to the three aspects from which we measure the differences. *Study 2* aims to investigate the extent to which our proposed features can contribute to the field of readability assessment. With regard to this, four experiments are designed and conducted.

Experiment 1 demonstrates the effectiveness of our proposed textual form features for readability assessment. We first perform validation experiments on the OneStopEnglish and NewsEla parallel corpora. Results show that incorporating our proposed features demonstrates a notable improvement across both corpora, confirming their potential in readability assessment. We then compare our proposed features with existing inner-semantic-based features in terms of their ability to characterize text readability. Results show that combining our proposed features with existing inner semantic-based features yields the best model performance, both for RandomForest and SMO models across the Weebit and Cambridge corpora. Notably, this combination achieves an accuracy of 0.784 with the SMO model on the Weebit corpus and 0.795 with the RandomForest model on the Cambridge corpus. Therefore, it is concluded that our proposed features are highly complementary to existing inner semantic-based features. In addition to these handcrafted features, neural features are also investigated in this experiment. Results show that handcrafted features are superior to neural features in terms of stability and interpretability. Finally, the incorporation of our proposed features into state-of-the-art models markedly enhances their performance, further confirming the value of our proposed features in readability assessment.

Experiment 2 gives a detailed analysis of the feature contributions to model prediction. Results show that our proposed textual form features rank highly among all the 118 features in terms of the feature importance score (MDI). Notably, all of the ten proposed features have ranked in the top 15 ranking lists on the Weebit corpus. It is concluded that our proposed textual form features significantly contribute to the prediction of readability among the handcrafted features.

Experiment 3 confirms that our proposed textual form features are applicable to different languages. We test the effectiveness of our proposed textual form features in a multilingual setting (in Spanish and French). Our strategy outperforms the feature-engineered state-of-the-art method and achieves comparable performance with the neural network-based state-of-the-art strategy. *Experiment 4* analyzes the impact of simplification tools on the performance of features. The results indicate that the model's performance on both corpora decreases when the simplification pipeline is removed, suggesting that incorporating the simplification pipeline into the model contributes to improving the model's performance. While the impact on performance differs across various simplification tools, the overall result shows that the inclusion of a simplification tool boosts model performance, suggesting the adaptability of our approach to different simplification systems.

Although our proposed features demonstrate promising results for readability assessment, there are some limitations that need further improvement. One of the limitations is that the quality of the proposed features relies heavily on the performance of the simplification tool. If the simplification tool fails to accurately simplify the given text, the effectiveness of our proposed features could be compromised. Future work will enhance the robustness of proposed features across different simplification tools. Moreover, in the future, we will investigate more fine-grained textual form features and exploit their deeper potential for multilingual readability assessment.

Acknowledgements. This work was supported by the National Natural Science Foundation of China [grant number: 61976062].

References

Alfter D. and Volodina E. (2018). Towards single word lexical complexity prediction. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pp. 79–88.

Anthony L. (2014). *Antwordprofiler (version 1.4. 1)[computer software]*. Tokyo, Japan: Waseda university.

Association N. G., et al. (2010). Common core state standards. Washington, DC..

Azpiazu I.M. (2017). Towards multilingual readability assessment, PhD thesis, Master's thesis. Boise State University, Boise, Idaho.

Azpiazu I.M. and Pera M.S. (2019). Multiattentive recurrent neural network architecture for multilingual readability assessment. *Transactions of the Association for Computational Linguistics* **7**, 421–436.

Baazeem I., Al-Khalifa H. and Al-Salman A. (2021). Cognitively driven arabic text readability assessment using eye-tracking. *Applied Sciences* **11**(18), 8607.

Bailin A. and Grafstein A. (2001). The linguistic assumptions underlying readability formulae: a critique. *Language & Communication* **21**(3), 285–301.

Banerjee S. and Lavie A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72.

Barzilay R. and Lapata M. (2008). Modeling local coherence: An entity-based approach. *Computational Linguistics* **34**(1), 1–34.

Bernstam E.V., Shelton D.M., Walji M. and Meric-Bernstam F. (2005). Instruments to assess the quality of health information on the world wide web: what can our patients actually use? *International Journal of Medical Informatics* **74**(1), 13–19.

Biber D. (1992). On the complexity of discourse complexity: a multidimensional analysis. *Discourse Processes* **15**(2), 133–163.

Blei D.M., Ng A.Y. and Jordan M.I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research* **3**, 993–1022.

Breiman L. (1996). Bagging predictors. *Machine Learning* **24**(2), 123–140.

Breiman L. (2001). Random forests. *Machine Learning* **45**(1), 5–32.

Buder-Gröndahl T. (2023). The ambiguity of bertology: what do large language models represent? *Synthese* **203**(1), 15.

Caldwell B., Cooper M., Reid L.G., Vanderheiden G., Chisholm W., Slatin J. and White J. (2008). Web content accessibility guidelines (wcag) 2.0. *WWW Consortium (W3C)* **290**,1–34.

Callison-Burch C., Osborne M. and Koehn P. (2006). Re-evaluating the role of bleu in machine translation research. In *11th conference of the european chapter of the association for computational linguistics*, pp. 249–256.

Chall J. S. and Dale E. (1995). Readability revisited: the new dale-chall readability formula. Brookline Books..

Chatzipanagiotidis S., Giagkou M. and Meurers D. (2021). Broad linguistic complexity analysis for greek readability classification. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 48–58.

Chebat J.-C., Gelinas-Chebat C., Hombourger S. and Woodside A.G. (2003). Testing consumers' motivation and linguistic ability as moderators of advertising readability. *Psychology & Marketing* **20**(7), 599–624.

Chen X. and Meurers D. (2016). Ctap: A web-based tool supporting automatic complexity analysis. In *Proceedings of the workshop on computational linguistics for linguistic complexity (CL4LC)*, pp. 113–119.

Collins-Thompson K. (2014). Computational assessment of text readability: a survey of current and future research. *ITL-International Journal of Applied Linguistics* **165**(2), 97–135.

Cristea D., Ide N., Marcu D. and Tablan V. (2000). Discourse Structure and Coreference: An Empirical Study. *The 18th International Conference on Computational Linguistics (COLING '00)*, Saarbrücken, Germany, Vol. **1**, pp. 208–214.

Crossley S.A., Greenfield J. and McNamara D.S. (2008). Assessing text readability using cognitively based indices. *Tesol Quarterly* **42**(3), 475–493.

Crossley S.A., Kyle K. and Dascalu M. (2019). The tool for the automatic analysis of cohesion 2.0: integrating semantic similarity and text overlap. *Behavior Research Methods* **51**(1), 14–27.

Crossley S.A., Kyle K. and McNamara D.S. (2016). The tool for the automatic analysis of text cohesion (TAACO): Automatic assessment of local, global, and text cohesion. *Behavior Research Methods* **48**(4), 1227–1237.

Dale E. and Chall J.S. (1948). A formula for predicting readability: instructions. *Educational Research Bulletin*, 37–54.

Dale E. and Chall J.S. (1949). The concept of readability. *Elementary English* **26**(1), 19–26.

Dean C. (2009). *Am i Making Myself Clear? In, Am I Making Myself Clear?*. Harvard University Press.

Deutsch T., Jasbi M. and Shieber S. (2020). Linguistic features for readability assessment. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, Seattle, WA, USA→ Online: Association for Computational Linguistics, pp. 1–17.

Deza M.M. and Deza E. (2009). Encyclopedia of distances. In *Encyclopedia of Distances*. Springer, pp. 1–583.

Eltorai A.E., Naqvi S.S., Ghanian S., Eberson C.P., Weiss A.-P.C., Born C.T. and Daniels A. H. (2015). Readability of invasive procedure consent forms. *Clinical and Translational Science* **8**(6), 830–833.

Ettinger A. (2020). What BERT is not: lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics* **8**, 34–48.

Feng L., Elhadad N. and Huenerfauth M. (2009). Cognitively motivated features for readability assessment. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pp. 229–237.

Feng L., Jansche M., Huenerfauth M. and Elhadad N. (2010). A comparison of features for automatic readability assessment.

Fernández Huerta J. (1959). Medidas sencillas de lecturabilidad. *Consigna* **214**, 29–32.

Finnimore P., Fritzsch E., King D., Sneyd A., Rehman A.U., Alva-Manchego F. and Vlachos A. (2019). Strong Baselines for Complex Word Identification across Multiple Languages. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Volume 1 (Long and Short Papers). Minneapolis, MN: Association for Computational Linguistics, pp. 970–977.

Frantz R.S., Starr L.E. and Bailey A.L. (2015). Syntactic complexity as an aspect of text complexity. *Educational Researcher* **44**(7), 387–393.

Friederici A.D., Chomsky N., Berwick R.C., Moro A. and Bolhuis J.J. (2017). Language, mind and brain. *Nature Human Behaviour* **1**(10), 713–722.

Gala N., François T., Bernhard D. and Fairon C. (2014). Un modèle pour prédire la complexité lexicale et graduer les mots. In *TALN'2014*, pp. 91–102.

Gala N., François T. and Fairon C. (2013). Towards a French lexicon with difficulty measures: NLP helping to bridge the gap between traditional dictionaries and specialized lexicons. In *Proceedings of the 3rd eLex Conference: Electronic Lexicography in the 21st Century: Thinking Outside the Paper*, Tallinn, Estonia.

Gamson D.A., Lu X. and Eckert S.A. (2013). Challenging the research base of the common core state standards: a historical reanalysis of text complexity. *Educational Researcher* **42**(7), 381–391.

Gomaa W.H., Fahmy A.A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications* **68**(13), 13–18.

Gooding S., Berzak Y., Mak T. and Sharifi M. (2021). Predicting text readability from scrolling interactions. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pp. 380–390.

Gosse C. and Van Reybroeck M. (2020). Do children with dyslexia present a handwriting deficit? impact of word orthographic and graphic complexity on handwriting and spelling performance. *Research in Developmental Disabilities* **97**, 103553.

Graesser A.C., **McNamara D.S.**, **Cai Z.**, **Conley M.**, **Li H. and Pennebaker J.** (2014). Coh-metrix measures text characteristics at multiple levels of language and discourse. *The Elementary School Journal* **115**(2), 210–229.

Graesser A.C., **McNamara D.S. and Kulikowich J.M.** (2011). Coh-metrix: providing multilevel analyses of text characteristics. *Educational Researcher* **40**(5), 223–234.

Graesser A.C., **McNamara D.S.**, **Louwerse M.M. and Cai Z.** (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, & Computers* **36**(2), 193–202.

Guinaudeau C. and Strube M. (2013). Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 93–103.

Gunning R., et al (1952). Technique of clear writing.

Han H., **Guo X. and Yu H.** (2016). Variable selection using mean decrease accuracy and mean decrease gini based on random forest. In *2016 7th ieee international conference on software engineering and service science (icsess)*, IEEE, pp. 219–224.

Hearst M.A., **Dumais S.T.**, **Osuna E.**, **Platt J. and Scholkopf B.** (1998). Support vector machines. *IEEE Intelligent Systems and Their Applications* **13**(4), 18–28.

Heilman M., **Collins-Thompson K.**, **Callan J. and Eskenazi M.** (2007). 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*, pp. 460–467.

Heilman M., **Collins-Thompson K. and Eskenazi M.** (2008). An analysis of statistical models and features for reading difficulty prediction. In *Proceedings of the third workshop on innovative use of NLP for building educational applications*, pp. 71–79.

Hewitt J. and Manning C.D. (2019). A structural probe for finding syntax in word representations, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1(Long and Short Papers)*, pp. 4129–4138.

Imperial J.M. (2021). Bert embeddings for automatic readability assessment. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pp. 611–618.

Jaccard P. (1901). Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulleting - Société Vaudoise des Sciences Naturelles* **37**, 547–579.

Jawahar G., **Sagot B. and Seddah D.** (2019). What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 3651–3657.

Jin T., **Lu X. and Ni J.** (2020). Syntactic complexity in adapted teaching materials: differences among grade levels and implications for benchmarking. *The Modern Language Journal* **104**(1), 192–208.

Kandel L. and Moles A. (1958). Application de l'indice de flesch à la langue française. *Cahiers Etudes De Radio-Télévision* **19**(1958), 253–274.

Kenton J.D. M.-W.C. and Toutanova L.K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pp. 4171–4186.

Kincaid J.P., **Fishburne Jr R.P. and Chissom B.S.** (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel, Naval Technical Training Command Millington TN Research Branch. Technical report.

Kovaleva O., **Romanov A.**, **Rogers A. and Rumshisky A.** (2019). Revealing the dark secrets of bert4364–4373, arXiv preprint arXiv: 1908.

Kuperman V., **Stadthagen-Gonzalez H. and Brysbaert M.** (2012). Age-of-acquisition ratings for 30,000 english words. *Behavior Research Methods* **44**(4), 978–990.

Kyle K. (2016). *Measuring syntactic development in L2 writing: Fine-grained indices of syntactic complexity and usage-based indices of syntactic sophistication (Doctoral dissertation)*. Georgia State University.

Landauer T.K., **Foltz P.W. and Laham D.** (1998). An introduction to latent semantic analysis. *Discourse Processes* **25**(2-3), 259–284.

Lee B. W., **Jang Y. S. and Lee J.** (2021). Pushing on text readability assessment: A transformer meets handcrafted linguistic features. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 10669–10686.

Lee B.W. and Lee J. (2020). Lxper index 2.0: Improving text readability assessment model for l2 english students in korea. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*, pp. 20–24.

Lee B.W. and Lee J. (2023). Prompt-based learning for text readability assessment. In *Findings of the Association for Computational Linguistics: EACL*, pp. 1774–1779.

Lennon C. and Burdick H. (2004). *The Lexile framework as an approach for reading measurement and success*. Durham, NC: MetaMetrics, Inc.

Levenshtein V. I., et al. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady*, **10**: Soviet Union, pp. 707–710.

Lewis M., **Liu Y.**, **Goyal N.**, **Ghazvininejad M.**, **Mohamed A.**, **Levy O.**, **Stoyanov V. and Zettlemoyer L.** (2020). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880.

Liu F. and Lee J.S. (2023). Hybrid models for sentence readability assessment. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pp. 448–454.

Liu N.F., Gardner M., Belinkov Y., Peters M.E. and Smith N.A. (2019). Linguistic knowledge and transferability of contextual representations, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies Volume 1 (Long and Short Papers)*, pp. 1073–1094.

Lively B.A. and Pressey S.L. (1923). A method for measuring the vocabulary burden of textbooks. *Educational Administration and Supervision* **9**(7), 389–398.

Loper E. and Bird S. (2002). Nltk: the natural language toolkit**1**, 63–70. arXiv preprint cs/0205028.

Lu X. (2010). Automatic analysis of syntactic complexity in second language writing. *International Journal of Corpus Linguistics* **15**(4), 474–496.

Lu X. (2011). A corpus-based evaluation of syntactic complexity measures as indices of college-level esl writers' language development. *TESOL Quarterly: A Journal for Teachers of English to Speakers of Other Languages and of Standard English as a Second Dialect* **45**(1), 36–62.

Lu X. and Xu Q. (2016). L2 syntactic complexity analyzer and its applications in l2 writing research. *Foreign Language Teaching and Research* **48**(3), 409–420.

Malvern D. and Richards B. (2012). *Measures of lexical richness. Encyclopedia of Applied Linguistics*. Hoboken, NJ: Blackwell Publishing Ltd.

Manning C.D. and Schütze H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Martin L., Fan A., de la Clergerie É., Bordes A. and Sagot B. (2022). MUSS: Multilingual unsupervised sentence simplification by mining paraphrases. In N. Calzolari, F. Béchet, P. Blache, et al. (Eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 1651–1664.

Martinc M., Pollak S., Robnik-Šikonja M. (2021). Supervised and unsupervised neural approaches to text readability. *Computational Linguistics* **47**(1), 141–179.

Mc Laughlin G.H. (1969). Smog grading-a new readability formula. *Journal of Reading* **12**(8), 639–646.

McNamara D.S., Graesser A.C., McCarthy P.M. and Cai Z. (2014). *Automated evaluation of text and discourse with CohMetrix*. Cambridge, UK: Cambridge University Press.

Mikolov T., Sutskever I., Chen K., Corrado G.S. and Dean J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pp. 3111–3119.

Mohammadi H. and Khasteh S.H. (2019). Text as environment: A deep reinforcement learning text readability assessment model. *Computation and Language*. arXiv preprint arXiv:1912.05957, pp.1–24.

Morales S., Mora J. and Alvarez M. (2019). Effectiveness of rewordify in a receptive skill: implication in reading comprehension in EFL A2 ecuadorian learners in tertiary education level. *Education Quarterly Reviews* **2**(4), 684–693.

Okinina N., Frey J.-C. and Weiss Z. (2020). Ctap for italian: Integrating components for the analysis of italian into a multilingual linguistic complexity analysis tool. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC)*, pp. 7123–7131.

Ortega L. (2003). Syntactic complexity measures and their relationship to l2 proficiency: A research synthesis of college-level l2 writing. *Applied Linguistics* **24**(4), 492–518.

Ott N. and Meurers D. (2011). Information retrieval for education: making search engines language aware. *Themes in Science and Technology Education* **3**(1-2), 9–30.

Paivio A., Yuille J.C. and Madigan S.A. (1968). Concreteness, imagery, and meaningfulness values for 925 nouns. *Journal of Experimental Psychology* **76**(1, Pt.2), 1–25.

Papineni K., Roukos S., Ward T. and Zhu W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318.

Partin S., Westfall E., Sanda G., Branham K., Muir K., Bellcross C. and Jain N. (2022). Readability, content, and accountability assessment of online health information for retinitis pigmentosa & retinitis pigmentosa treatment options. *Ophthalmic Genetics* **44**(1), 1–6.

Pearson K., Ngo S., Ekpo E., Sarraju A., Baird G., Knowles J., Rodriguez F. (2022). Online patient education materials related to lipoprotein (a): readability assessment. *Journal of Medical Internet Research* **24**(1), e31284.

Pera M.S. and Ng Y.-K. (2014). Automating readers' advisory to make book recommendations for k-12 readers. In *Proceedings of the 8th ACM Conference on Recommender Systems*, pp. 9–16.

Petersen S.E. and Ostendorf M. (2009). A machine learning approach to reading level assessment. *Computer Speech & Language* **23**(1), 89–106.

Pitler E. and Nenkova A. (2008). Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, pp. 186–195.

Plain Language Action and Information Network (PLAIN). (2011). *Federal Plain Language Guidelines*. Washington, DC: PLAIN.

Platt J.C. (1999). 12 fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods*, pp. 185–208.

Prasad R., Webber B. and Joshi A. (2017). The Penn Discourse Treebank: An Annotated Corpus of Discourse Relations. In: Ide, N., Pustejovsky, J. (eds) *Handbook of Linguistic Annotation*. Dordrecht: Springer, pp. 1197–1217.

Qiang J., Li Y., Zhu Y., Yuan Y. and Wu X. (2020). Lexical simplification with pretrained encoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, pp. 8649–8656.

Rayner K. and Duffy S.A. (1986). Lexical complexity and fixation times in reading: effects of word frequency, verb complexity, and lexical ambiguity. *Memory & Cognition* **14**(3), 191–201.

Reif E., Yuan A., Wattenberg M., Viegas F.B., Coenen A., Pearce A. and Kim B. (2019). Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems 32*, pp. 8594–8603.

Richardson J.T. (1975). Imagery, concreteness, and lexical complexity. *Quarterly Journal of Experimental Psychology* **27**(2), 211–223.

Rogers A., Kovaleva O. and Rumshisky A. (2020). A primer in bertology: what we know about how bert works. *Transactions of the Association for Computational Linguistics* **8**, 842–866.

Scarton C., Aprosio A. P., Tonelli S., Wanton T. M. and Specia L. (2017). Musst: A multilingual syntactic simplification tool. In *Proceedings of the IJCNLP 2017, System Demonstrations*, pp. 25–28.

Schwarm S. E. and Ostendorf M. (2005). Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL'05)*, pp. 523–530.

Sheehan K.M., Kostin I., Napolitano D. and Flor M. (2014). The textevaluator tool: Helping teachers and test developers select texts for use in instruction and assessment. *The Elementary School Journal* **115**(2), 184–209.

Si L. and Callan J. (2001). A statistical model for scientific readability. In *Proceedings of the tenth international conference on Information and knowledge management*, pp. 574–576.

Siddharthan A. (2014). A survey of research on text simplification. *ITL-International Journal of Applied Linguistics* **165**(2), 259–298.

Soler A.G. and Apidianaki M. (2020). Bert knows punta cana is not just beautiful, it's gorgeous: Ranking scalar adjectives with contextualised representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7371–7385.

Štajner S., Sheang K. C. and Saggion H. (2022). Sentence simplification capabilities of transfer-based models. *Proceedings of the AAAI Conference on Artificial Intelligence* **36**(11), 12172–12180.

Stodden R. and Kallmeyer L. (2020). A multi-lingual and cross-domain analysis of features for text simplification. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI)*. Marseille, France: European Language Resources Association, pp. 77–84.

Szmrecsanyi B. (2004). On operationalizing syntactic complexity. In *Le poids des mots. Proceedings of the 7th international conference on textual data statistical analysis. Louvain-la-Neuve*, **2**, pp. 1032–1039.

Tack A., François T., Ligozat A.-L. and Fairon C. (2016). Modèles adaptatifs pour prédire automatiquement la compétence lexicale d'un apprenant de français langue étrangère. In L. Danlos & T. Hamon (Eds.), *Actes de la conférence conjointe JEP-TALN-RECITAL 2016*. volume 2: TALN (Articles longs). Paris, France: AFCP - ATALA, pp. 221–234.

Tanaka S., Jatowt A., Kato M.P. and Tanaka K. (2013). Estimating content concreteness for finding comprehensible documents. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 475–484.

Thorndike E.L. (1921). *The teacher's word book*. New York, NY: Teachers College, Columbia University.

Tonelli S., Tran K.M. and Pianta E. (2012). Making readability indices readable. In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, pp. 40–48.

Vajjala S. and Lučić I. (2018). OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, New Orleans, Louisiana: Association for Computational Linguistics, pp. 297–304.

Vajjala S. and Meurers D. (2012). On improving the accuracy of readability classification using insights from second language acquisition. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, Montréal, Canada: Association for Computational Linguistics, pp. 163–173.

Vajjala S. and Meurers D. (2016). *Readability-based sentence ranking for evaluating text simplification*. Ames, IA: Iowa State University.

Wang J. and Dong Y. (2020). Measurement of text similarity: a survey. *Information-an International Interdisciplinary Journal* **11**(9), 421.

Washburne C. and Vogel M. (1926). *What children like to read: Winnetka graded book list*. Chicago, IL: Public School Publishing Company.

Wilkens R., Alfter D., Wang X., Pintard A., Tack A., Yancey K.P. and François T. (2022). Fabra: French aggregator-based readability assessment toolkit. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 1217–1233.

Wilkens R., Watrin P., Cardon R., Pintard A., Gribomont I. and François T. (2024). Exploring hybrid approaches to readability: experiments on the complementarity between linguistic features and transformers. In Graham Y. and Purver M., (eds), *Findings of the Association for Computational Linguistics: EACL*. St. Julian's, Malta: Association for Computational Linguistics, pp. 2316–2331.

Williamson G.L., Fitzgerald J. and Stenner A.J. (2013). The common core state standards' quantitative text complexity trajectory: figuring out how much complexity is enough. *Educational Researcher* **42**(2), 59–69.

Wilson M. (1988). MRC psycholinguistic database: Machine-usable dictionary, version 2.00. *Behavior Research Methods, Instruments, & Computers* **20**(1), 6–10.

Witten I. H. and Frank E. (2002). Data mining: practical machine learning tools and techniques with java implementations. *Sigmod Record* **31**(1), 76–77.

Wolfe M.B., Schreiner M.E., Rehder B., Laham D., Foltz P.W., Kintsch W. and Landauer T.K. (1998). Learning from text: matching readers and texts by latent semantic analysis. *Discourse Processes* **25**(2-3), 309–336.

Xia M., Kochmar E. and Briscoe T. (2016). Text readability assessment for second language learners. In J. Tetreault, J. Burstein, C. Leacock, & H. Yannakoudakis (Eds.), *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, San Diego, CA: Association for Computational Linguistics, pp. 12–22.

Xu W., Callison-Burch C. and Napoles C. (2015). Problems in current text simplification research: new data can help. *Transactions of the Association for Computational Linguistics* **3**, 283–297.

Yan X., Song D. and Li X. (2006). Concept-based document readability in domain specific information retrieval. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pp. 540–549.

Yang S.-j. (1971). *A readability formula for Chinese language (Doctoral dissertation)*. Madison, WI: University of Wisconsin-Madison.

Zien A., Krämer N., Sonnenburg S. and Rätsch G. (2009). The feature importance ranking measure. In *Joint European conference on machine learning and knowledge discovery in databases*, Springer, pp. 694–709.

A. Part-of-speech tagging in NLTK

Part-of-speech	Tags in NLTK	Word type for tagging
Noun	NN	Noun singular
	NNS	Noun plural
	NNP	Proper noun or proper noun plural
Verb	VB	Base verb
	VBD	Past tense verb
	VBG	Gerund or present participle
	VBN	Past participle verb
	VBP	Third-person singular present tense verb
	VBZ	Present tense verb
Adjective	JJ	Adjective
	JJR	Comparative adjective
	JJS	Superlative adjective

B. Descriptions of the Coh-Metrix features (partial)

Category	Feature	Description
Descriptive	DESPC	Paragraph count, number of paragraphs
	DESSC	Sentence count, number of sentences
	DESWC	Word count, number of words
	DESPL	Paragraph length, number of sentences, mean
	DESPLd	Paragraph length, number of sentences, standard deviation
	DESWLsyd	Word length, number of syllables, standard deviation
	DESWLlt	Word length, number of letters, mean
	DESWLltd	Word length, number of letters, standard deviation
Text Easability PC Scores	PCNARz	Text Easability PC Narrativity, *z* score
	PCCNCp	Text Easability PC Word concreteness, percentile
	PCREFz	Text Easability PC Referential cohesion, *z* score
	PCREFp	Text Easability PC Referential cohesion, percentile
	PCDCp	Text Easability PC Deep cohesion, percentile
	PCCONNp	Text Easability PC Connectivity, percentile
	PCTEMPz	Text Easability PC Temporality, *z* score
	PCTEMPp	Text Easability PC Temporality, percentile
Referential Cohesion	CRFAO1	Argument overlap, adjacent sentences, binary, mean
	CRFSO1	Stem overlap, adjacent sentences, binary, mean

This table is partial quoted from the book written by McNamara *et al.* (2014). Due to limited space, only the features mentioned in the paper are presented.

Cite this article: Pan W, Li X, Chen X and Xu R. Textual form features for text readability assessment. *Natural Language Processing* https://doi.org/10.1017/nlp.2024.50