International Journal of Artificial Intelligence in Education (2020) 30:337–370
https://doi.org/10.1007/s40593-020-00201-7

ARTICLE

Applying Natural Language Processing and Hierarchical Machine Learning Approaches to Text Difficulty Classification

Renu Balyan[1] · Kathryn S. McCarthy[2] · Danielle S. McNamara[3]

Published online: 25 June 2020

Abstract

For decades, educators have relied on readability metrics that tend to oversimplify dimensions of text difficulty. This study examines the potential of applying advanced artificial intelligence methods to the educational problem of assessing text difficulty. The combination of hierarchical machine learning and natural language processing (NLP) is leveraged to predict the difficulty of practice texts used in a reading comprehension intelligent tutoring system, iSTART. Human raters estimated the text difficulty level of 262 texts across two text sets (Set A and Set B) in the iSTART library. NLP tools were used to identify linguistic features predictive of text difficulty and these indices were submitted to both flat and hierarchical machine learning algorithms. Results indicated that including NLP indices and machine learning increased accuracy by more than 10% as compared to classic readability metrics (e.g., Flesch-Kincaid Grade Level). Further, hierarchical outperformed non-hierarchical (flat) machine learning classification for Set B (72%) and the combined set A + B (65%), whereas the non-hierarchical approach performed slightly better than the hierarchical approach for Set A (79%). These findings demonstrate the importance of considering deeper features of language related to text difficulty as well as the potential utility of hierarchical machine learning approaches in the development of meaningful text difficulty classification.

Keywords Text difficulty · Machine learning · Hierarchical classification · Natural language processing

✉ Renu Balyan
renu.balyan@asu.edu

Kathryn S. McCarthy
kmccarthy12@gsu.edu

Danielle S. McNamara
dsmcnama@asu.edu

Extended author information available on the last page of the article

Introduction

Text remains a crucial learning tool in most classrooms today (Fuchs et al. 2014). In the classroom, students are often tasked to read texts and textbooks in order to learn new information. As such, many educators and text publishers attempt to level texts such that text difficulty is appropriate for the students (National Governors Association Center for Best Practices 2010). Readability formulas have been used for well over a century as a means to evaluate text difficulty. Indeed, teachers have long relied on readability metrics to select classroom materials (e.g., Fry 2002; Chall 1988). In many ways, this practice is well grounded: theories of learning suggest that learning occurs most readily when tasks are tailored to students' ability (e.g., Bjork 1994; Vygotsky 1978). Vygotsky's well-known *zone of proximal development* posits that tasks that are challenging, but potentially achievable with adequate support, are more effective for learning than tasks that are too easy or too difficult. With this in mind, many researchers and publishers have developed ways to estimate text difficulty in order to *match* reading assignments to students' estimated skill levels (Benjamin 2012).

Finding texts that are matched to course content, students' interest, and their current reading skill is challenging. Given the abundance of materials available and the varying needs of each of their students, instructors simply do not have the time and resources to engage in careful evaluation of text difficulty. One approach has been to rely on publishers' anthologies (e.g., basal readers), which define texts according to their targeted grade level. Though grade level anthologies offer instructors a quick way to find texts that are relevant to the "average" student at a given grade level, the criteria for how these texts are selected are often unclear and unsystematic. For example, Scholastic, a leading publisher for children's books offers a variety of systems to identify grade-level appropriate texts. These systems vary in terms of focus (e.g., interest, skill) and grain-size. It is of note that Scholastic acknowledges that not all of their books have been levelled using the same systems,[1] leaving it up to the instructors to make these cross-system comparisons.

The problem of selecting appropriately-challenging texts is magnified in educational technologies that rely on text materials. In order to deliver texts that are well-matched to ability, there needs to be a large body of texts for the system to draw upon to meet the needs of each student and adapt to those needs as the students' skills change across instruction. Thus, the constraints faced by an individual instructor needing to find appropriate texts for a classroom of students is further amplified when scaling up instruction through automation. As such, developing valid and facile means of assessing text difficulty remains an important issue across many areas in education.

The most common approach has been the use of readability formulas (Bormuth 1966, 1969) such as Flesch-Kincaid Reading Ease or Grade Level (Flesch 1948; Kincaid et al. 1975; Klare 1974), Dale Chall (Dale and Chall 1948), Gunning Fox (Gunning 1969), or Lexile (Lennon and Burdick 2004; Stenner et al. 1988). These measures are relatively easy to calculate (e.g., number of words per sentence and number of syllables per word multiplied by constants) and are even embedded in basic word processing software. Indeed, most readability formulas have been driven by ease of computation, overlooking key aspects of language related to comprehension and

[1] https://www.scholastic.com/teachers/articles/teaching-content/leveled-reading-systems-explained/

learning processes (Duran et al. 2007; Graesser et al. 2004; McNamara et al. 2012; McNamara et al. 2014; McNamara et al. 1996). Texts that contain many short sentences and words are typically rated as "easy" by readability metrics. However, these metrics of readability can be poor at predicting comprehension. For example, Begeny and Greene (2014) assessed passages from the Dynamic Indicators of Basic Early Literacy Skills (DIBELS) test, a test for assessing the acquisition of early literacy skills, using eight common readability formulas. They then asked 360 students from different grades to read the passages. They found that the readability formulas were at or below chance in identifying the appropriate grade level. One consequence of imprecise, or even inaccurate, measures of text difficulty is that students may be assigned texts that are too difficult or too easy (McNamara et al. 1996). Such a mismatch can lead to sub-optimal learning and, potentially, frustration from both the students and teachers when students do not perform as well as they "should".

One reason for this disconnect between readability and reading comprehension is that readability algorithms rely on superficial aspects of language instead of discourse-level features (Duran et al. 2007; Graesser et al. 2004). For example, content-driven texts may include short sentences composed of monosyllabic, but complex and topic-specific words (e.g., "quark"; Si and Callan 2001) or there may be coherence gaps between the ideas conveyed across sentences, rendering the text more difficult to understand (Graesser et al. 2004).

One means of improving evaluation of text difficulty is to go beyond simple word-based metrics to include linguistic and semantic indices related to discourse comprehension. Advances in NLP have allowed researchers to extract rich information about the linguistic features of a text that reflect complex dimensions such as narrativity, syntactic complexity, and cohesion (e.g., Crossley et al. 2016a; Crossley et al. 2016b; Duran et al. 2007; McNamara et al. 2014). These tools include part-of-speech (POS) taggers, parsers, sentiment analyzers, and semantic role labellers. More recently, and of particular relevance for the current study, a large number of NLP tools have been developed that are driven by cognitive theory to more closely mirror human judgements of text difficulty. These features include lexical sophistication (Kyle and Crossley 2015; Kyle et al. 2018), syntactic complexity (Kyle 2016), and cohesion (Crossley et al. 2016a; Crossley et al. 2018; McNamara et al. 2014), as well as clause-level features and rhetorical features (McNamara et al. 2013).

One such NLP tool, Coh-Metrix (McNamara et al. 2014), assesses more than 200 measures of cohesion, language, and readability.[2] Coh-Metrix integrates a number of sophisticated tools (such as advanced syntactic parsers, POS taggers, and distributional models) and psycholinguistic databases (Salsbury et al. 2011) to generate indices of language, text, and readability (Duran et al. 2007). Coh-Metrix reports standard readability metrics (e.g., Flesch-Kincaid) as well as other word-level indices (e.g., familiarity, concreteness) drawn from the MRC Psycholinguistics Database (Coltheart 1981; Gilhooly and Logie 1980; Paivio et al. 1968; Toglia and Battig 1978). In addition, Coh-Metrix returns indices that evaluate the degree to which information is being connected from sentence-to-sentence, sentence-to-paragraph, and paragraph-to-paragraph. Such connections afford a more coherent mental

[2] For more on NLP, see McNamara et al. (2018). For a more thorough discussion of Coh-Metrix, see Graesser et al. (2004) and McNamara et al. (2014).

representation in the mind of the reader. Indeed, indices of cohesion predict ease of comprehension (Allen et al. 2016; Allen et al. 2015; Graesser et al. 2004; Millis et al. 2007; Ozuru et al. 2005). Thus, one aspect of this work is the demonstrate that the automated assessment of text difficulty can be enhanced through examination of these theoretically-motivated features of language.

Machine Learning

The central purpose of the current study is to examine the effectiveness of different types of machine learning algorithms to predict text difficulty, in addition to evaluating the effectiveness of using deeper linguistic features such as cohesion. Classic readability formulas rely on General Linear Modeling (GLM), which involves a set of a priori statistical assumptions about the nature of the data set that may or may not be appropriate depending on the circumstance. In contrast, many machine learning (ML) approaches do not make similar statistical assumptions. Recent work has demonstrated the utility of machine learning techniques in predicting text difficulty. For example, Pitler and Nenkova (2008) compared linear regression and support vector classification approaches. Relevant to the current study, they found that the combination of lexical, syntax and coherence features was more predictive than including only surface level features. Feng et al. (2010) further demonstrated that classification models performed better than regression models. Brunato et al. (2018) used similar approaches but for sentence-level text difficulty, and Vajjala and Meurers (2012) applied classification models to improve prediction accuracy using insights from second language acquisition (SLA). Tanaka-Ishii et al. (2010) used a slightly different method from other researchers, treating text readability as a ranking problem rather than classification or regression. Others (e.g., Collins-Thompson 2014; François and Miltsakaki 2012; Heilman et al. 2008; Kate et al. 2010; Kotani et al. 2011; Pilán et al. 2014; Pilán et al. 2016; Schwarm and Ostendorf 2005; Sung et al. 2015)[3] have demonstrated benefits of incorporating theoretically-motivated linguistic features in the context of machine learning and text classification. The present study builds on this body of work by considering the possible advantages of hierarchical machine learning.

Flat Vs. Hierarchical Approaches to Machine Learning

The previously mentioned machine learning studies focused on the differences between regression and methods such as classification or ranking. By contrast, in this study, we examine the advantages of using *flat* versus *hierarchical* classification approaches for text readability. Hierarchical approaches to machine learning involve a series of classifications.

Non-hierarchical, flat classification is the simplest and the most direct approach to machine learning. It uses either a single or ensemble classifier, and all the class variable instances in the training dataset. Imagine, for example, categorizing 100 supermarket

[3] We were unable to locate downloadable software or corpora associated with these studies. Thus, we could not compare our algorithms to those used in these studies. Notably, that was not the purpose of this study nor does this affect the validity of the previous studies.

items into 10 classes. In a flat classification, a "rater" would pick up each of the 100 items one at a time and consider the item based on a set of features. The rater would then use this exploration of features to place the item in one of the 10 categories. Alternatively, data can be classified using a *binary hierarchical classification* (see Fig. 1), which is based on the divide-and-conquer strategy (Casasent and Wang 2005; Kumar and Ghosh 1999; Wang and Casasent 2009). Using the supermarket example, the items would first be broken into two macro-classes (e.g., meat and produce). At the next level, the produce macro-class would be further divided into two smaller macro-classes of fruit and vegetable, and then fruit macro-class would be divided as stone fruit, citrus, or berries until all items were in divided in one of the 10 final classes.

In binary hierarchical classification, the classes are divided into two smaller macro-classes at each node. Only $\log_2 K$ classifiers need to be traversed in order to move from the top to a bottom decision node. One can use multiple hierarchical classification as well if there are large number of categories. Given the limited number of categories in our targeted corpora (see *Corpus*), it made more sense to use binary classification approach instead of multi-classification.

Many important real-world classification problems are naturally treated as hierarchical classification problems, where the classes to be predicted are naturally organized in a class hierarchy. As a result, classification problems where classes are arranged in a hierarchy can be expected to perform better with hierarchical approach as compared to using a flat classification. Hierarchical classification has been used in protein classification (Cerri et al. 2015; Triguero and Vens 2016; Zimek et al. 2008), text classification (Cesa-Bianchi et al. 2006; Mayne and Perry 2009), essay scoring (McNamara et al. 2015), image annotation (Dimitrovski et al. 2011), automatic target recognition (Casasent and Wang 2005). A few studies have demonstrated that hierarchical approaches outperform flat classifiers (Kumar et al. 2002; Schwenker 2000).

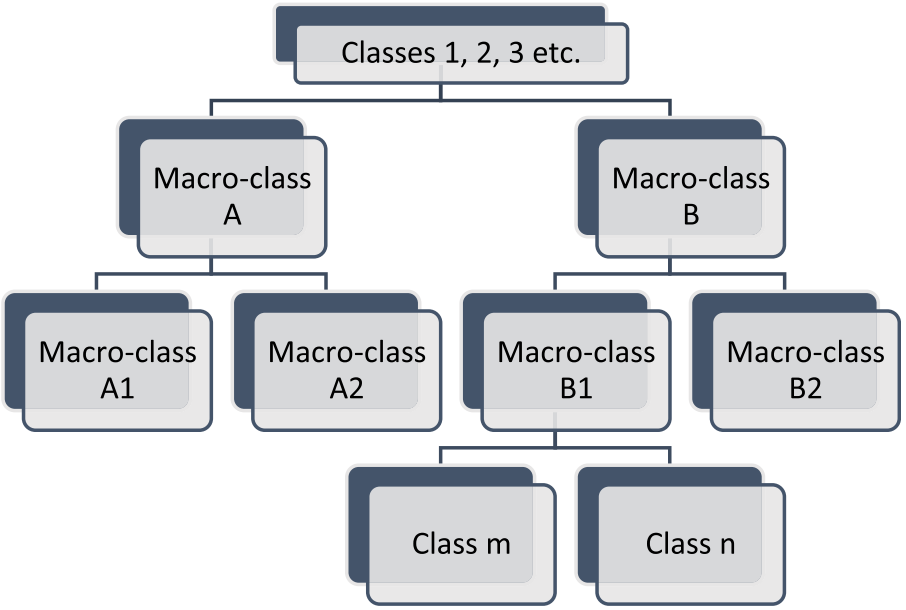


Fig. 1 Hierarchical Classification Structure

To our knowledge, hierarchical approaches have not been applied to classification of text difficulty. Such approaches may be well suited to text difficulty for several reasons. First, the features of texts that make them appropriate for middle school readers might be different than what makes a text more or less difficult for high school readers – that is, there are potential qualitative differences in different categories of texts. Second, such approaches may better reflect processes involved in human judgements of text difficulty. Imagine an instructor identifying the difficulty of a given text. The instructor might first determine if the text is appropriate for young children or adolescents, thus determining a grade band before "drilling down" into a specific grade. Text difficulty involves classifying items into a number of classes that naturally form a hierarchy, rather than simple dichotomous identification. As such, a hierarchical approach may be particularly apt for this complex task.

The Current Study

The current study leverages advances in NLP and ML to examine how the two in combination might predict human ratings of text difficulty. It is an extension of research into hierarchical ML approaches conducted in Balyan et al. (2018). In this work by Balyan et al. (2018), the potential utility of a hierarchical approach was demonstrated. In the current work, we systematically investigate how features derived from natural language processing can improve upon simple readability metrics and how the combination of NLP and ML approaches can support more authentic and accurate estimations of text difficulty. Notably, the aim of this work was not to produce *new* readability formulas, but to demonstrate new ways of estimating grade level that are considerate of complex aspects of language implicated by theories of discourse comprehension. The work examines the accuracy of ML classification approaches in predicting human ratings of text difficulty. Importantly, we used human comparison ratings of text difficulty rather than pre-labelled data (e.g., texts with grade difficulty assigned from unknown origins).

In previous studies, the corpora were either randomly selected from large corpora or were not publicly available. Thus, rather than attempting to replicate results with imperfect comparisons, we selected a new corpus that reflected the type of text set that educators might encounter in their practice. Our selected corpora were two sets of expository texts appropriate for a wide range of readers (elementary through college). These text sets were pre-existing and afforded us the opportunity to test our premises on authentic data sets.

We used NLP tools to identify critical linguistic indices. These indices were then used as predictors to train the models, which were in turn trained using both flat and hierarchical ML classification approaches. Our objective was to examine ML classification accuracy as compared to and in combination with classic readability metrics. We compared a variety of classification algorithms as there is no universally best learning algorithm that fits all, and the "best" models differ depending on the classification task at hand (Caruana and Niculescu-Mizil 2006). For example, SVMs are designed to perform better on high-dimensional data, but are considered complex and it takes a long time to train a model. Tree-based methods are not influenced by outliers and multicollinearity, and do not make any assumptions on the distribution of data but

these methods do not have a very high predicting power. The performance of an algorithm may depend on whether the problem is linearly or non-linearly separable, the type of kernel used, or the type of hyper-parameters used while training the model. Therefore, rather than selecting a single classifier, we explored several machine learning classification algorithms in Weka (version 3.8.1).

It was hypothesized that the NLP indices motivated by cognitive theories of discourse comprehension would be better predictors of text difficulty compared to classic readability metrics (e.g., Flesch-Kincaid Grade Level). It was also predicted that the combination of NLP and ML and, more specifically, a hierarchical ML approach would yield classifications more similar to the human ratings than a flat or non-hierarchical classification. Flat classification approaches make a single decision involving all the categories in the data. It is difficult to make a single decision on multiple categories that may potentially be unbalanced (Babbar et al. 2013) as compared to making decision on a step-by-step dichotomous data, which can be more accurate.

General Methods

Corpus

We conducted our experiments on existing real-world text sets. The corpus included the two text sets within Interactive Strategy Training for Active Reading and Thinking (iSTART) - an intelligent tutoring system (ITS) that supports successful reading comprehension of complex informational texts through self-explanation training (McNamara et al. 2004; Snow et al. 2016). The text sets were collected from a variety of open-source resources in the context of other iSTART development projects. These text sets were ideal for this set of experiments as they were designed to have a variety of levels of difficulty (see Perret et al. 2017; Snow et al. 2016). Set A included texts that varied widely in genre and difficulty, whereas Set B comprised texts used to provide information typical in high school and college science courses. One limitation to the corpus is that it is relatively small. Another limitation of note is that the corpus contains an unbalanced number of instances in each class. That is, we did not actively select the same number of texts at each level of difficulty. While class imbalance can affect the accuracy of ML models, the corpus reflects an authentic challenge. Specifically, the library of texts was selected because it is embedded within a real-world learning environment, and not for the purposes of developing a classification model.

The first set, Set A, was comprised of texts from the iSTART StairStepper module ($n = 162$), including expository, informational texts about science, history, pop culture, and sports. These texts were collected from publicly available websites (see Perret et al. 2017) and contained, on average, 389 words and 27 sentences.

The second set of texts, Set B ($n = 100$) was comprised of the texts from iSTART's main text library. These texts are complex, informational texts about scientific phenomena compiled in the original development of iSTART (Jackson and McNamara 2013). The texts were culled from various sources, primarily science textbooks. Each text in Set B had an average 380 words and 25 sentences. The number of texts in each genre are shown in Table 1.

Table 1 Frequency of Texts by Topic

Text / Topic	Science	Social Sciences	Sports	Pop Culture	Total
Set A	47	75	19	13	162
Set B	100	0	0	0	100

Human Ratings

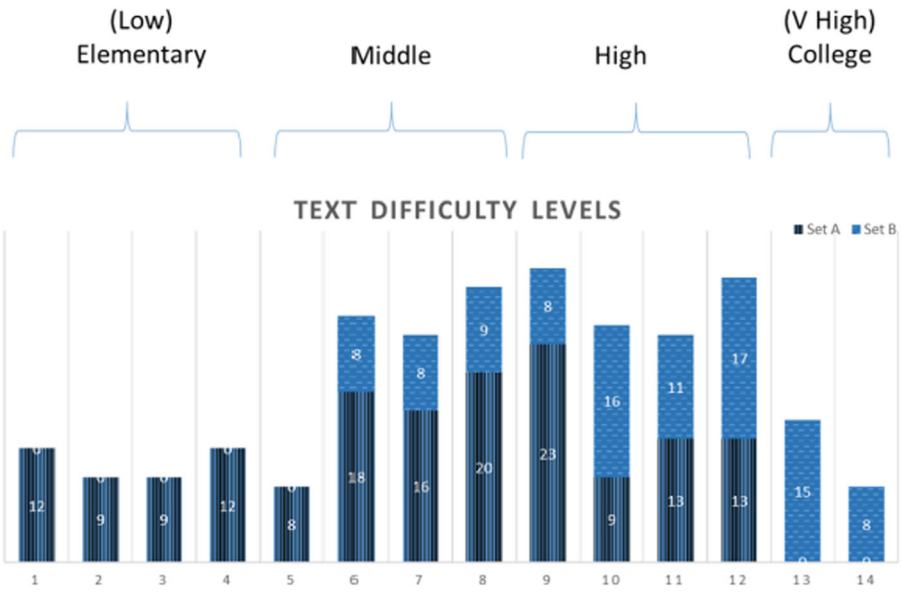
Some of the texts in the corpus were pre-labelled with a grade level difficulty by the original sources. However, we found no relations between these pre-labelled levels and common readability measures. In addition, we found several examples of passages that were inconsistently labelled across sources. In order to establish accurate benchmarks, we employed human comparison ratings to evaluate the difficulty of each text.

The difficulty of the texts in the two sets were rated separately, but followed the same procedure (see Johnson et al. 2017). We developed a qualitative approach drawn from methods of discourse analysis (e.g., Gee 2004; van Dijk 1985) that assess the text beyond word and sentence level information to consider the text as a whole. This approach was similar to an unsupervised clustering task, in which four raters (members of the research lab) iteratively sorted the texts as a team. As a group, the four raters first did a "rough sort" of entire set of texts into three broad sets (easy, medium, and difficult). In this first sort, we were not doing direct comparisons of each text, but rather an approximation of text difficulty. The group of raters then read each text more carefully and separated each of the three levels into "easier" and "more difficult", yielding six levels. This process continued until there was unanimous agreement amongst the raters that the set of texts could not be separated any further. Adjacent levels were combined and resorted by each rater until there was agreement that 1) each level was distinguishably different from the adjacent level and that 2) every text in a given level was of comparable difficulty. Disagreements were resolved through discussion. Thus, agreement was reached across all four raters for all difficulty levels. This process resulted in 12 levels for Set A and 9 levels for Set B. Because Set B was designed to include more difficult texts than Set A, the levels across the text sets needed to be aligned. Two of the original raters determined the comparable difficulty across the two sets. Raters read the easiest texts from Set B and compared this group of texts to the levels in Set A. It was agreed that the easiest texts in Set B were equivalent to the sixth level of difficulty in Set A. Further reading and discussion confirmed that each increasing level of difficulty in Set A matched those in Set B. Consequently, Set A texts ranged in difficulty from level 1 to level 12 and Set B texts ranged in difficulty from level 6 to 14. The combined Set A and Set B corpus resulted in 262 texts categorized into 14 difficulty levels (1–14).

Correlational analyses indicated that these human ratings of text difficulty were strongly related to Flesch-Kincaid Grade level for set A ($r = 0.79$). In Set B, where the text difficulty range was higher, this correlation was less strong ($r = 0.41$). These

correlations indicate that the expert ratings were consistent with FKGL, but not redundant. This suggests that our human judgements of difficulty depended on different, or at least additional, features of the text not considered in simple readability formulas, particularly for more complex texts.

In exploratory experiments, we used different ML algorithms, with readability measures such as Flesch Kincaid Grade Level and Flesch Kincaid Reading Ease and several other linguistic features considering all text difficulty levels (1–12 for Set A and 6–14 for set B). The classification accuracy for Set A for the ML algorithms was quite low, ranging from 13.33% to 31.67% for readability formulas and 25.97% to 33.95% when additional linguistic features were used. The accuracy range for Set B was between 8.11% and 18.92% for readability formulas, which improved slightly when additional linguistic features were used, ranging between 29.17% and 35.42%. The classification accuracy further decreased when we combined the two data sets with accuracy ranging from 19.44% to 26.39%. Consequently, we clustered the fine-grained text difficulty levels (1–14) into more coarse-grained levels. The researchers re-read the texts in each difficulty level to identify intuitive breaks in the text set. This resulted in four difficulty levels: low (1–4), middle (5–8), high (9–12), and very high (13 and 14). Set A included low, middle, and high difficulty texts, while Set B included middle, high, and very high difficulty texts. The distribution and partitions made across the difficulty levels across the texts is shown in Fig. 2.



Dataset	Easy	Medium	Difficult	Very Difficult
Set A	42	62	58	-
Set B	-	25	52	23

Fig. 2 Text difficulty levels across the two sets of texts

Feature Extraction and Selection

Following the corpus collection and human rating of difficulty levels of those texts, the next step included extraction of linguistic features/indices of the two text sets. We employed Coh-Metrix (McNamara et al. 2014) to extract the linguistic features of texts. Coh-Metrix returns over 200 linguistic features, some of which are not theoretically-relevant for the informational texts in this corpus. The features when extracted from the Coh-Metrix vary significantly in their scales and therefore the data were normalized (i.e., rescaled the values using Min-Max method in the range of [0,1]) before any machine learning algorithms were used. Further, to avoid overfitting, we used feature selection methods to reduce the dimensionality of features (i.e., the number of indices). We removed features having zero variance (ZV) or nearly zero variance (NZV) and applied a recursive feature elimination (RFE) approach. We tried multiple classifiers for the RFE including random forest, Naïve Bayes and bagged tree to determine the common features selected by the classifiers. Because RFE can be negatively affected by multicollinearity (Lieberman and Morris 2014), we also removed highly correlated (Pearson's $r > .85$) features before applying RFE.

Applying these feature selection approaches resulted in a set of eight linguistic indices (see Table 2). Notably, these features have been shown to correlate with text difficulty in prior work (Crossley et al. 2012; Crossley and McNamara 2009; Klare 1984; McCarthy and Jarvis 2010). These indices relate to lexical sophistication, readability, lexical diversity, and syntactic complexity, and are briefly described in Table 2. Age of acquisition was found to be the most important whereas familiarity was the least important for all the data sets when classifying texts.

A series of analysis of variance (ANOVA) tests confirmed that each of the eight linguistic features differed as a function of the human ratings of text difficulty, both for the two sets of data individually and for the combined dataset (Table 3).

The omnibus ANOVAs indicated a significant effect of text difficulty on the majority of indices (indicated in bold). The exceptions are lexical diversity in Set B, and familiarity in Set B and the combined set. Post hoc tests revealed that the significant effects were driven by differences between the 'low' level compared to the 'high' and 'very high' levels. Notably, few indices showed differences between the 'middle' and 'high' levels. In sum, these tests confirm that these linguistic features are appropriate for determining text difficulty.

Machine Learning Algorithms

In this study, we compared some of the most commonly used classification algorithms. A summary of these algorithms is given in Table 4 (see also Appendix B and Balyan et al. 2017). While determining the accuracy of models, the model parameters were tuned wherever applicable using the Grid Search approach.

Experiments and Results

After the confirmatory ANOVAs (Table 3), we used the NLP features in a series of classification experiments to predict human ratings of text difficulty. As our data

involved categorical human ratings, we adopted supervised machine learning classification approaches rather than regression. We conducted the following experiments:

Experiment 1 (FKGL)

Flat (non-hierarchical) classification comparing a ZeroR baseline classifier to more than 40 different classifiers, using FKGL as the sole predictor. (For clarity, we report only the results of eight classifiers that achieved the highest accuracy).

Experiment 2 (FKGL+)

Flat (non-hierarchical) classification comparing ZeroR to the other classifiers using FKGL in conjunction with additional linguistic features derived from the feature selection (Table 2) to investigate the benefits of including the additional linguistic features.

Experiment 3

Hierarchical classification using the most accurate single classifiers (not ensembles) obtained from Experiments 1 and 2 to examine the potential advantages of a hierarchical approach.

The classifiers used for the experiments are implemented using Weka tool version 3.8.1 and R packages. We used 10-fold stratified cross validation to compute the performance metrics of the classification models. Using only a single split test and train data (or a holdout method) can lead to high variance and biased results. The results may depend heavily on the data points included in the training and test set. Therefore, we used multiple (10) data splits for training and test data, hence called 10-fold cross validation. Using this method ignores the fact how the data is divided as every data point gets to be test data point once and training data point 9 times, thus reducing the variance as number of folds increase. The classification accuracy in this study is the proportion of true results (both true positives and true negatives) among the total number of examined cases. We also report accuracy and the F-scores for clarity.

Experiment 1: Non-hierarchical Classification Using FKGL The first set of experiments considered only FKGL to predict text difficulty. Table 5 shows the best classifiers of the 40 that were tested. Note that these classifiers are consistent with those shown to be the most accurate in a number of other text classification applications (Aggarwal and Zhai 2012; Hartmann et al. 2019; Kowsari et al. 2019; Sun and Lim 2001). The baseline classifier (ZeroR) classification accuracy for Set A was lower (0.38) than that of Set B (0.52) and the combined set (0.42). The accuracy for the different classifiers varied across data sets (Set A: 0.66–0.71; Set B: 0.45–0.60; Combined: 0.45–0.56), but accuracy of the classifiers improves significantly over the baseline classifier (ZeroR).

For Set A, several algorithms including Naïve Bayes, linear discriminant analysis (LDA) and AdaBoost achieved the highest classification accuracy (0.71) and highest kappa (0.56).

For Set B, the classification accuracy varied from 0.45 to 0.60. Notably, the highest accuracy obtained for the classifiers in Set B was less than the lowest classification

Table 2 Description of the Linguistic Features

Linguistic Feature	Description
Flesch-Kincaid Grade Level (FKGL)	FKGL is a simple measure of readability computed using average number of syllables per word and average number of words per sentence. Number of words in a sentence correlates with the effort required to read the sentence. While, number of syllables in a word is inversely related with word frequency, and affects reading difficulty (Zipf 1949; Dufty et al. 2006). Lower grade levels correspond to easier texts.
L2 readability score	L2 readability score predicts the readability of texts for second-language learners (Crossley et al. 2012). L2 readability score considers content word overlap, sentence syntactic similarity, and word frequency. In contrast to FKGL, higher scores indicate easier texts.
Syntactic complexity	Syntactic complexity of a sentence is determined by considering mean number of words before the main verb, and higher number of higher-level constituents per word in the sentence. Sentences having less syntactic complexity are easier to process and comprehend (Crossley et al. 2012; Perfetti et al. 2005).
Uncommon or rare words	Uncommon or rare words in a text refers to how rarely a word occurs in the English language. More uncommon or rare words in a text make the text more difficult. The text difficulty is expected to increase if there are words that readers have never or rarely encountered. This index is computed from CELEX (Baayen et al. 1995), a 17.9 million words corpus.
Lexical diversity	Lexical diversity refers to the variety of words used in a text. Lexical diversity is usually measured using type–token ratios (TTR), which is related to text length. In order to consider indices regardless of text length, we consider MTLD (measure of textual, lexical diversity; McCarthy 2005) and D values (Malvern et al. 2004; McNamara et al. 2013) computed by our NLP tool. The index for MTLD approach is calculated as the mean length of sequential word strings that maintain a criterion level of lexical variation or a given TTR value (McCarthy 2005).
Word familiarity	Word familiarity refers to how familiar or easily an adult recognizes a word. For example, the words 'cat', 'dog', 'table', 'fan' have a higher average familiarity as compared with the words 'cortex', 'dogma', and 'wigwam'. Word familiarity ratings are computed using the MRC Psycholinguistic Database, which provides ratings for several thousands of words along several psychological dimensions. Sentences that contain words that are more familiar are processed more quickly (McNamara et al. 2013).
Word imageability	Word imageability refers to the ease with which one can construct a mental image of a word in one's mind. High-imagery words include terms such as 'airplane' or 'hammer', whereas words like 'dogma' or 'quantum' are much less imageable (Paivio et al. 1968).
Age of Acquisition	Age of Acquisition refers to the age at which a word first appears in a child's vocabulary (Paivio et al. 1968).

accuracy for Set A. Additionally, most of the classifiers failed to predict any instance of 'Medium' class except the Random Forest or the ensemble classifiers (such as Bagging and Boosting) that used Random Forest as the base classifier.

The baseline classification accuracy for the combined dataset was 0.42. The classification accuracy for rest of the classifiers varied between 0.45 and 0.56. The two classifiers with the highest accuracy (SVM, Naïve Bayes) did not predict any instance of the 'Very Difficult' class. Other well-known classifiers (e.g., BayesNet, neural

Table 3 Means and ANOVA Results for the selected linguistic features (significant F-tests appear in bold, significance level of 0.05)

	Set A				Set B				Set A + Set B				
Feature	Low	Middle	High	F (2,159)	Middle	High	Very High	F (2,97)	Low	Middle	High	Very High	F (3,258)
FKGL	5.87	8.83	10.83	**113.40**	9.17	9.25	11.56	**17.03**	5.87	8.93	10.08	11.56	**76.78**
L2 Readability	20.64	14.41	12.46	**46.17**	18.12	17.25	14.34	**4.35**	20.64	15.47	14.72	14.34	**16.80**
Syntactic Complexity	0.73	0.68	0.69	**13.18**	0.72	0.71	0.68	**15.71**	0.73	0.70	0.70	0.68	**11.53**
Uncommon /rare Words	49.52	94.87	107.41	**21.37**	57.92	62.42	81.91	**8.39**	49.52	84.25	86.15	81.91	**8.64**
Lexical Diversity (MTLD)	66.41	78.46	81.06	**9.69**	56.41	57.32	57.15	0.03	66.41	72.12	69.84	57.15	**3.92**
Familiarity	588.88	587.25	585.71	**4.53**	588.29	588.51	588.21	0.04	588.88	587.55	587.03	588.21	1.45
Imageability	355.11	347.74	336.18	**22.46**	333.54	329.06	317.99	**5.78**	355.11	343.66	332.81	317.99	**36.58**
Age of Acquisition (AoA)	5.13	5.53	5.82	**128.40**	5.69	5.97	6.39	**25.07**	5.13	5.57	5.89	6.39	**130.00**

Table 4 Brief description of classification algorithms used in the study

Algorithm	Description
ZeroR	ZeroR was used as the baseline classifier for our experiments. It uses the simplest classification approach which relies only on the target and ignores all predictors. It predicts the majority class in the data. There is no predictability power in ZeroR. Thus, it is useful for determining a baseline performance as a benchmark for other classification methods (Witten et al. 1999).
Naïve Bayes	Naïve Bayes is based on the Bayes' theorem of posterior probability. It is a probabilistic learning method, which assumes that the effect of an attribute value on a given class is independent of other attributes values (McCallum and Nigam 1998).
Logistic Regression	Logistic regression is a statistical model that (in its basic form) uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model; a form of binary regression (George-Nektarios 2013).
Support Vector Machines (SVM)	SVM constructs a hyperplane that separates the data into classes. SVMs are efficient for high-dimensional feature spaces and are among the best supervised learning algorithms (Dumais et al. 1998; Joachims 1998).
Linear Discriminant Analysis (LDA)	LDA, or normal discriminant analysis (NDA), or discriminant function analysis is a generalization of Fisher's linear discriminant, a method used in statistics, pattern recognition and machine learning to find a linear combination of features that characterizes or separates two or more classes of objects or events. (Martínez and Kak 2001; Mika et al. 1999).
MultiClass	Multiclass is a metaclassifier for handling multi-class datasets with 2-class classifiers. This classifier is also capable of applying error correcting output codes for increased accuracy (Rojas 1996; Zhang 2000).
Boosting	Boosting is meta-algorithm that incrementally builds an ensemble by iteratively training weak learners or classifiers. While training new models, it emphasizes instances that are misclassified by the previous models. Thus, each model is trained on weighted data from the previous model performance. The final result is the weighted sum of the results of all of the classifiers. LogitBoost is used for performing additive logistic regression, and AdaBoost boosts a nominal class classifier using the AdaBoost M1 algorithm (Freund and Schapire 1996; Krogh and Vedelsby 1994).
Bagging	Bagging is an ensemble classifier that uses bootstrap aggregation (or "bagging") to reduce variance. This implementation works for both classification and regression, depending on the base learner. In the case of classification, predictions are generated by averaging probability estimates, not by voting (Breiman 1996; Ho 1995; Schapire and Singer 1999; Schölkopf and Smola 2002).

network) also failed to predict any instance of the 'Very Difficult' class. The ensemble classifiers that did predict instances for the 'Very Difficult' class (e.g., Bagging and Boosting) had low precision (0.19–0.62), recall (0.09–0.22) and F-scores (0.15–0.32). The F-score (F_1) is computed as the harmonic mean of precision and recall.

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Experiment 2: Non-hierarchical Classification Using FKGL plus Linguistic Features The second set of experiments was conducted using the same set of classifiers used in Experiment 1 to examine the extent to which classification accuracy was improved when linguistic features (see Table 2) were added as predictors. We refer to this set of linguistic features as FKGL+.

The classification accuracy results for Set A improved from 0.61–0.71 to 0.75–0.82 when additional linguistic features were used as predictors. Some classifiers such as SVM, LDA, and logistic showed an improvement of over 14% in the classification accuracy and the F-scores. The classification accuracy for some classifiers (AdaBoost and MultiClass) for Set B like Set A also showed improvement in accuracy of over 33%. However, the classification accuracy for the Set B classifiers was still low as compared to the classification accuracies and F-scores of Set A. SVM classifier obtained one of the highest accuracies for this set but it did not predict any instance of the 'Middle' class.

Like the individual data sets, the classification accuracy for the combined set also improved and for some classifiers (mostly the ensembles) with improvement over 20%. However, for Bagging the classification accuracy and F-scores improvement was approximately 51%. The SVM classifier achieved the third highest classification accuracy and was able to identify instances of all four classes. The summary results of some of the top performing classifiers are shown in Table 5. Since we are dealing with multi-class classification in this study with 3 classes in set A and set B, and 4 classes in the combined set (Set A + Set B), there is a separate F-score for each class in each data set that is returned by every classifier. We have not reported individual F-score for each class, but specify a range of F-scores returned for a dataset for every classifier in Table 5 to avoid rendering the table overly complex. Precision, recall, F-scores for all of the classes for the ML classifier is provided in Appendix A.

Consistent with Experiment 1, classification accuracies and F-scores were higher for Set A as compared to those for Set B and the combined data set. Importantly, Experiment 2 demonstrated that the inclusion of the theoretically-motivated linguistic features improved the accuracy of the classifiers for both sets of texts individually as well as the combined set. For clarity, the improvement in classification accuracy when using FKGL+ over FKGL is illustrated in Fig. 3. The details for the summarized results of Table 5 for the precision, recall for the classifiers and the rationale for these range of values for the accuracy and F-score is provided in Appendix A.

Experiment 3: Hierarchical Classification Experiment 3 examines the potential advantages of using a hierarchical, rather than flat classification approach. It was predicted that this approach would be more accurate because there were more than two classes that were ordinally ranked. From a more theoretical perspective, easier texts might be distinct from one another based on more superficial aspects of texts (e.g., word difficulty) whereas more difficult texts might vary along more complex dimensions of language (e.g., syntax). This multidimensionality of language might be better captured in the multiple runs to separate the different classes.

Table 5 Accuracy and F-Scores for classifiers using FKGL and FKGL+ as predictor variable

Classifier	Features	Data Source		
		Set A Accuracy (F-score)	Set B Accuracy (F-score)	Set A + Set B Accuracy (F-score)
ZeroR (Baseline)		0.38	0.52	0.42
Naïve Bayes	FKGL	0.71 (0.62–0.74)	0.59 (0.00–0.70)	0.56 (0.00–0.69)
	FKGL+	0.77 (0.72–0.86)	0.57 (0.35–0.68)	0.58 (0.47–0.66)
Logistic	FKGL	0.70 (0.61–0.75)	0.60 (0.00–0.62)	0.56 (0.15–0.69)
	FKGL+	0.82 (0.76–0.87)	0.63 (0.48–0.68)	0.62 (0.55–0.72)
SVM	FKGL	0.68 (0.59–0.74)	0.57 (0.00–0.69)	0.56 (0.00–0.68)
	FKGL+	0.80 (0.76–0.88)	0.63 (0.00–0.73)	0.64 (0.39–0.77)
LDA	FKGL	0.71 (0.62–0.77)	0.59 (0.00–0.70)	0.55 (0.07–0.69)
	FKGL+	0.81 (0.76–0.89)	0.64 (0.50–0.71)	0.61 (0.54–0.68)
MultiClass	FKGL	0.70 (0.56–0.77)	0.45 (0.31–0.54)	0.47 (0.17–0.51)
	FKGL+	0.79 (0.72–0.88)	0.63 (0.50–0.70)	0.60 (0.49–0.69)
LogitBoost	FKGL	0.66 (0.56–0.76)	0.56 (0.26–0.66)	0.54 (0.32–0.64)
	FKGL+	0.75 (0.67–0.86)	0.63 (0.43–0.68)	0.64 (0.37–0.73)
AdaBoost	FKGL	0.71 (0.62–0.77)	0.45 (0.31–0.54)	0.50 (0.17–0.63)
	FKGL+	0.76 (0.70–0.89)	0.60 (0.44–0.66)	0.64 (0.41–0.75)
Bagging	FKGL	0.70 (0.62–0.76)	0.53 (0.17–0.64)	0.45 (0.19–0.53)
	FKGL+	0.77 (0.73–0.86)	0.66 (0.43–0.76)	0.68 (0.44–0.77)

Ensemble classifiers implementation is complex because for each ensemble the output of one model is given as input to another model. This process needs to be implemented for each level of the hierarchical model, making the training process cumbersome and time consuming. For simplicity and ease of implementation for hierarchical classification, we considered the classification accuracy and F-scores for single classifiers only (Naïve Bayes, logistic, SVM and LDA) in Table 5.

We conducted three experiments using hierarchical classification for each of the data sets by using combinations of different class/category texts. For example, Set A data were classified into three classes: elementary, middle, and high. For the first run, we first classified texts as ‘elementary’ and ‘other’. At the second level in this experiment the ‘others’ class was classified into ‘middle’ and ‘high’. For the second run, the texts were first classified as ‘middle’ and ‘other’ and then the ‘other’ texts were further classified as ‘elementary’ and ‘high’. Finally, for the third run, the texts were first classified as ‘high’ and ‘other’ and then the ‘other’ texts were reclassified as ‘elementary’ and ‘middle’. A summary of these experimental combinations for different data sets is provided in Table 6.

The classifier accuracy of each level is determined by comparing the classifier output with the actual output at that level. This accuracy was used to select the best classifier for a particular level in an experiment or run. The final accuracy

(Table 7) is computed by applying the best classifier at each hierarchy level. The final accuracy is computed separately and not using the accuracy of individual levels. When the whole test set has been divided into the relevant classes, the predictions for each class are compared to the human ratings.

It was observed that out of the four single classifiers discussed above, SVM and LDA performed the best for binary classification used in the hierarchical approach. The classification accuracy of all the three experiments for the hierarchical approach is summarized in Table 7. This table shows the classification results obtained for the final model. We observed that hierarchical classification significantly improves the accuracy of the model for both the Set B (from 0.64 to 0.72) and for the combined set (A + B; from 0.61 to 0.65) when compared with accuracy for these sets for the non-hierarchical approach. In contrast, the accuracy decreases slightly for Set A (from 0.81 to 0.79). The SVM/LDA values mentioned in the "classifier" column in Table 7 indicate that both the classifiers (LDA and SVM) performed equally, hence it did not matter which of these classifiers was chosen at that level.

he classification results for each level and the final accuracy in the three experiments appear in Table 7. We considered all four single classifiers at each level, but report the results of only the best performing model. The highest accuracy for each data set in each experiment is indicated in bold. The confusion matrix and text difficulty level-wise accuracy are provided in Appendix D.

The classification results (Table 7) indicate that the model classification accuracy depends on the data combinations considered at a specific level of hierarchy. Thus, it is important to carefully choose the class combinations for a particular hierarchy level for appropriate classification of the data. The results in Table 7 indicate that the first experiment (or run) resulted in highest accuracy for Set A, while the first and third experiment resulted in highest accuracy for the Set B and the second one for the combined (A + B) data (Table 7).

In sum, the linguistic features of the text differed across the human ratings of text difficulty. These features were particularly salient between the 'elementary' and 'high' levels for Set A, and 'middle' and 'college' levels for Set B. Consistent with extant work on text difficulty, the results show that lower level texts ('elementary' for Set A and 'middle' for Set B) are generally less lexically and syntactically sophisticated than higher-level texts ('high' for Set A and

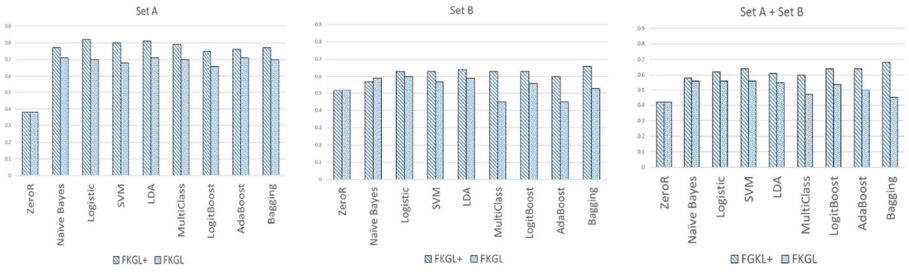


Fig. 3 Comparison of classifier accuracies using only readability (FKGL) and readability in combination with additional linguistic features (FKGL+). (Note that ZeroR uses no features, so the experiment is identical for FKGL and FKGL+, both bars are included only for ease of comparison)

'college' for Set B). Lower-level texts also contain less uncommon (rare words) and more concrete words than the higher-level texts. The hierarchical classification achieved classification accuracy of 79% for Set A, 72% for Set B, and 65% for the combined set A + B. The accuracy improved for Set B and the combined set (A + B) but decreased slightly for the Set A when the hierarchical approach was used.

Discussion

In order to provide individualized reading support for their students, educators often rely on readability metrics (National Governors Association Center for Best Practices 2010; Graesser et al. 2011). While easy to use, these metrics overlook important aspects of text that relate to a students' ability to understand and learn from the text. The current study takes advantage of advances in AI, namely natural language processing and hierarchical machine learning to assess text difficulty. In the context of iSTART, the ability to rapidly and accurately predict human ratings of text difficulty affords the opportunity to provide adaptive instruction with a larger corpus of practice items.

These experiments demonstrate that including additional linguistic features of text increased classification accuracy as compared to using FKGL alone. This supports theoretical predictions that text complexity emerges from deeper aspects of language than merely number of syllables and sentence length as well as existing studies demonstrating gains of including additional linguistic aspects of text beyond traditional metrics (see Collins-Thompson 2014). Thus, it is important to encourage researchers and instructors to look beyond traditional readability when considering what texts and tasks might be best suited for their students. The ability to provide better metrics in the same amount of time means that instructors can use these indices to make more informed decisions. This also means that researchers can offer more accurate feedback in the context of automated systems. For example, teachers can upload their own texts into the iSTART library, which in turn can be classified regarding difficulty. Asking a teacher to manually decide if the text they are adding is a level 7, 8, or 9 as compared to the other dozens of texts in the extant library would be time consuming for the instructor and lend to inconsistent classifications across

Table 6 Hierarchical classification experiments summary

Experiment	Set A	Set B	Set A+ Set B
Run 1	E + (M/H)	M + (H/C)	(E/M) + (H/C)
Run 2	M + (E/H)	H + (M/C)	(E/H) + (M/C)
Run 3	H + (E/M)	C + (M/H)	(E/C) + (M/H)

E: Elementary, M: Middle, H: High, C: College

Table 7 Level-based classification and Final accuracy results for Set A, Set B, and Combined Set

	Expt.	Partition	Classifier	Accuracy	Final Accuracy
Set A	Run 1	E + Others	LDA	0.94	**0.79**[*]
		M + H	SVM	0.81	
	Run 2	M + Others	SVM	0.75	0.75
		E + H	SVM	1.00	
	Run 3	H + Others	SVM	0.83	0.76
		E + M	SVM	0.90	
Set B	Run 1	M + Others	LDA	0.82	**0.72**[+]
		H + C	LDA	0.86	
	Run 2	H + Others	LDA	0.64	0.62
		M + C	LDA	0.89	
	Run 3	C + Others	LDA	0.87	**0.72**[+]
		M + H	LDA	0.80	
Combined Set (A + B)	Run 1	(E + M) + (H + C)	SVM	0.79	0.64
		E + M	LDA	0.82	
		H + C	SVM/LDA	0.87	
	Run 2	(E + H) + (M + C)	SVM	0.70	**0.65**[#]
		E + H	SVM	0.93	
		M + C	LDA	0.94	
	Run 3	(E + C) + (M + H)	SVM	0.82	0.62
		E + C	SVM/LDA	1.00	
		M + H	SVM	0.73	

[*] Highest accuracy; E: Elementary, M: Middle, H: High

[+] Highest accuracy; M: Middle, H: High, C: College

[#] Highest accuracy; E: Elementary, M: Middle, H: High, C: College

instructors As we continue to improve our text difficulty algorithms, we will be able to include these texts in the adaptive system and have confidence that students are receiving the right type of texts at the right time. The algorithm ensures that the difficulty levels assigned in the system remain consistent over time without needing to rely on the teacher to evaluate a given text in the context of the entire library. These classification approaches allow us to continue to grow the iSTART library, including texts added by teachers, while still providing an adaptive tutoring environment in which students are presented with skill-level appropriate readings (McCarthy et al. in press).

This research also makes multiple theoretical contributions to the literature. At the most basic level, this study provides comparisons of different ML approaches when used for the classification of expository texts. Important to this study, we assessed benchmark text difficulty ratings by hand, rather than relying on pre-existing labels of unknown or inaccurate origins. Such an approach allows for a more authentic assessment of text difficulty. These experiments demonstrated that including additional linguistic features as produced by NLP tools improved ML classification accuracy by more than 10% as compared to simple readability metrics. These findings are consistent with previous work using regression approaches (e.g., Duran et al. 2007; Graesser et al. 2004) and further emphasize

the importance of using linguistic features beyond word or syllable ratios when considering text complexity.

The study also contributed to the literature by introducing hierarchical machine learning as a means of evaluating text difficulty. When classifying Set B and the combined data (A + B) the hierarchical classification approach was more accurate. While the classifier was able to classify the text difficulty better for Set A when using non-hierarchical classification. The differences in the accuracy of these approaches suggest that there are potential differences in the nature of the texts in each text set. As seen in Table 3, the differences between the 'middle' and 'high' text sets were reduced when the sets were combined. The Set B texts are scientific texts appropriate for high school and college students, whereas the Set A texts were designed to include a broader range of reading skills and topics. Given that the two sets were developed for different purposes and scored independently of one another, it is appropriate to assume that they are not perfectly comparable, which is typical given the variety of educational situations and pedagogical objectives. However, one of the drawbacks of the hierarchical classification is that the approach is more resource intense. Several ML models (same or different for each level) on each text must be run while testing the models in order to classify unseen texts and to train any new models.

Limitations and Future Directions

A benefit to this study was that we use a pre-existing corpus, which provides ecological validity for our approach. That is, these models are built upon making decisions about real-world text sets that may have varying numbers of instances of particular cases. However, a resulting limitation of this choice of corpus is that the set is relatively small and unbalanced. Future work should examine the utility of these approaches using larger corpora to examine the generalizability of our findings. We did not find any study that used word embedding for classifying text difficulty except one by Jiang et al. 2018, although it has been successfully used for several other NLP tasks such as text classification, text summarization and sentiment analysis. Therefore, we also plan to use word embedding for the study in future and compare how this performs compared to other approaches used. We also plan on exploring how ordinal logistic regression would perform compared to the hierarchical classification approach.

As a result of these findings, separate algorithms to classify text difficulty will be implemented in iSTART depending on the module and target population. These experiments demonstrate the utility of hierarchical classification approaches for predicting text difficulty. However, these findings also highlight that this approach was not universally more accurate than a flat classification. As such, these findings highlight that improving the accuracy and efficacy of educational technologies may require relying on different approaches depending on the specific aspect of the target texts and population.

Acknowledgements The authors would like to recognize the support of the Institute of Education Sciences, U.S. Department of Education, through Grants R305A180261, R305A190050 and R305A180144, and the Office of Naval Research, through Grant N000141712300, to Arizona State University. The opinions expressed are those of the authors and do not represent views of the Institute, the U.S. Department of Education, or the Office of Naval Research.

Appendix A

Table 8 The evaluation Metrics (Set A) using FKGL

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL	0.00	0.00	0.00	0.38	0.0	Easy
		0.38	1.00	0.55			Medium
		0.00	0.00	0.00			Difficult
AdaBoost (RandomForest)	FKGL	0.65	0.67	0.66	0.61	0.41	Easy
		0.54	0.53	0.54			Medium
		0.66	0.66	0.66			Difficult
BayesNet	FKGL	0.67	0.71	0.69	0.64	045	Easy
		0.55	0.45	0.50			Medium
		0.68	0.78	0.73			Difficult
LogitBoost (Decision Stump)	FKGL	0.73	0.79	0.76	0.66	0.49	Easy
		0.59	0.53	0.56			Medium
		0.67	0.71	0.69			Difficult
Neural network	FKGL	0.76	0.69	0.73	0.67	0.50	Easy
		0.59	0.60	0.59			Medium
		0.71	0.74	0.72			Difficult
SMO (puk kernel)	FKGL	0.77	0.71	0.74	0.68	0.51	Easy
		0.59	0.60	0.59			Medium
		0.72	0.74	0.73			Difficult
Logistic	FKGL	0.79	0.71	0.75	0.70	0.54	Easy
		0.61	0.61	0.61			Medium
		0.73	0.78	0.75			Difficult
MultiClassClassifier (Logistic)	FKGL	–	–	–	0.70	0.54	Easy
		0.68	0.48	0.56			Medium
		0.69	0.88	0.77			Difficult
Bagging (HoeffdingTree)	FKGL	0.79	0.71	0.75	0.70	0.55	Easy
		0.62	0.63	0.62			Medium
		0.74	0.78	0.76			Difficult
Naïve Bayes	FKGL	0.81	0.69	0.74	0.71	0.56	Easy
		0.63	0.61	0.62			Medium
		0.73	0.83	0.77			Difficult
AdaBoost (Hoefdding Tree)	FKGL	0.81	0.71	0.76	0.71	0.56	Easy
		0.64	0.60	0.62			Medium
		0.72	0.83	0.77			Difficult
Trees (Hoefdding)	FKGL	0.81	0.71	0.76	0.71	0.56	Easy
		0.64	0.60	0.62			Medium
		0.72	0.83	0.77			Difficult
LDA	FKGL	0.79	0.71	0.75	0.71	0.56	Easy
		0.63	0.61	0.62			Medium
		0.73	0.81	0.77			Difficult

Table 9 The evaluation Metrics (Set B) using FKGL

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL	0.00	0.00	0.00	0.52	0.0	Medium
		0.52	1.00	0.684			Difficult
		0.00	0.00	0.00			Very Difficult
Trees (Random Forest)	FKGL	0.31	0.32	0.31	0.45	0.12	Medium
		0.55	0.52	0.54			Difficult
		0.40	0.44	0.42			Very Difficult
AdaBoost (RandomForest)	FKGL	0.31	0.32	0.31	0.45	0.12	Medium
		0.55	0.52	0.54			Difficult
		0.40	0.44	0.42			Very Difficult
MultiClassClassifier (RandomForest)	FKGL	0.31	0.32	0.31	0.45	0.12	Medium
		0.55	0.52	0.54			Difficult
		0.40	0.44	0.42			Very Difficult
Bagging (J48)	FKGL	0.27	0.12	0.17	0.53	0.19	Medium
		0.58	0.71	0.64			Difficult
		0.52	0.57	0.54			Very Difficult
BayesNet	FKGL	0.00	0.00	0.00	0.55	0.18	Medium
		0.58	0.85	0.69			Difficult
		0.46	0.48	0.47			Very Difficult
LogitBoost (Decision Stump)	FKGL	0.36	0.20	0.26	0.56	0.25	Medium
		0.60	0.73	0.66			Difficult
		0.57	0.57	0.57			Very Difficult
SMO (puk kernel)	FKGL	0.00	0.00	0.00	0.57	0.18	Medium
		0.56	0.90	0.69			Difficult
		0.63	0.44	0.51			Very Difficult
Trees (Hoefdding)	FKGL	0.00	0.00	0.00	0.59	0.23	Medium
		0.57	0.90	0.70			Difficult
		0.67	0.52	0.59			Very Difficult
AdaBoost (Hoefdding Tree)	FKGL	0.00	0.00	0.00	0.59	0.24	Medium
		0.58	0.89	0.68			Difficult
		0.65	0.57	0.61			Very Difficult
Naïve Bayes	FKGL	0.00	0.00	0.00	0.59	0.24	Medium
		0.58	0.89	0.70			Difficult
		0.65	0.57	0.61			Very Difficult
Neural network	FKGL	0.00	0.00	0.00	0.59	0.25	Medium
		0.58	0.87	0.70			Difficult
		0.61	0.61	0.61			Very Difficult
Logistic	FKGL	0.00	0.00	0.00	0.60	0.25	Medium
		0.58	0.90	0.71			Difficult
		0.68	0.57	0.62			Very Difficult
LDA	FKGL	0.00	0.00	0.00	0.59	0.24	Medium
		0.58	0.89	0.70			Difficult
		0.65	0.57	0.61			Very Difficult

Table 10 The evaluation Metrics (Set A + Set B) using FKGL

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL	0.00	0.00	0.00	0.42	0.0	Easy
		0.00	0.00	0.00			Medium
		0.42	1.00	0.59			Difficult
		0.00	0.00	0.00			Very Difficult
Bagging (RandomForest)	FKGL	0.46	0.43	0.44	0.45	0.19	Easy
		0.42	0.43	0.42			Medium
		0.52	0.54	0.53			Difficult
		0.21	0.17	0.19			Very Difficult
MultiClassClassifier (HoeffdingTree)	FKGL	0.64	0.43	0.51	0.47	0.19	Easy
		0.42	0.54	0.47			Medium
		0.49	0.49	0.49			Difficult
		0.25	0.13	0.17			Very Difficult
AdaBoost (Random Forest)	FKGL	0.51	0.52	0.52	0.47	0.21	Easy
		0.43	0.44	0.43			Medium
		0.53	0.53	0.53			Difficult
		0.19	0.17	0.18			Very Difficult
Trees (Hoefdding)	FKGL	0.61	0.64	0.63	0.50	0.26	Easy
		0.43	0.52	0.47			Medium
		0.56	0.52	0.54			Difficult
		0.25	0.13	0.17			Very Difficult
AdaBoost (HoeffdingTree)	FKGL	0.61	0.64	0.63	0.50	0.26	Easy
		0.43	0.52	0.47			Medium
		0.56	0.52	0.54			Difficult
		0.25	0.13	0.17			Very Difficult
BayesNet	FKGL	0.77	0.48	0.59	0.53	0.26	Easy
		0.47	0.39	0.43			Medium
		0.52	0.78	0.63			Difficult
		0.00	0.00	0.00			Very Difficult
Neural network	FKGL	0.67	0.67	0.67	0.53	0.28	Easy
		0.46	0.47	0.46			Medium
		0.54	0.64	0.58			Difficult
		0.00	0.00	0.00			Very Difficult
LogitBoost (REPTree)	FKGL	0.67	0.62	0.64	0.54	0.29	Easy
		0.45	0.38	0.41			Medium
		0.55	0.70	0.61			Difficult
		0.62	0.22	0.32			Very Difficult
Naïve Bayes	FKGL	0.72	0.67	0.69	0.56	0.32	Easy
		0.49	0.51	0.50			Medium
		0.57	0.67	0.61			Difficult
		0.00	0.00	0.00			Very Difficult
Logistic	FKGL	0.72	0.67	0.69	0.56	0.32	Easy
		0.49	0.43	0.45			Medium

Table 10 (continued)

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
		0.55	0.72	0.63			Difficult
		0.50	0.09	0.15			Very Difficult
SMO (puk kernel)	FKGL	0.74	0.62	0.68	0.56	0.33	Easy
		0.49	0.53	0.51			Medium
		0.57	0.69	0.63			Difficult
		0.00	0.00	0.00			Very Difficult
LDA	FKGL	0.72	0.67	0.69	0.55	0.31	Easy
		0.50	0.41	0.45			Medium
		0.54	0.73	0.62			Difficult
		0.33	0.04	0.07			Very Difficult

Table 11 The evaluation Metrics for Set A using FKGL+

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL+	0.38	1.00	0.55	0.38	0.0	Easy
		0.00	0.00	0.00			Medium
		0.00	0.00	0.00			Difficult
AdaBoost (RandomForest)	FKGL+	0.90	0.88	0.89	0.76	0.64	Easy
		0.68	0.73	0.70			Medium
		0.76	0.72	0.74			Difficult
LogitBoost (Decision Stump)	FKGL+	0.86	0.86	0.86	0.75	0.63	Easy
		0.69	06.3	0.67			Medium
		0.74	0.79	0.77			Difficult
SMO (poly kernel)	FKGL+	0.92	0.83	0.88	0.80	0.70	Easy
		0.71	0.81	0.76			Medium
		0.83	0.78	0.80			Difficult
Logistic	FKGL+	0.88	0.86	0.87	0.82	0.73	Easy
		0.77	0.76	0.76			Medium
		0.83	0.86	0.85			Difficult
MultiClassClassifier (Logistic)	FKGL+	0.88	0.88	0.88	0.79	0.69	Easy
		0.75	0.69	0.72			Medium
		0.78	0.85	0.81			Difficult
Bagging (HoeffdingTree)	FKGL+	0.92	0.81	0.86	0.77	0.65	Easy
		0.69	0.79	0.73			Medium
		0.78	0.74	0.76			Difficult
Naïve Bayes	FKGL+	0.92	0.81	0.86	0.77	0.65	Easy
		0.69	0.76	0.72			Medium
		0.77	0.76	0.77			Difficult
LDA	FKGL+	0.92	0.86	0.89	0.81	0.72	Easy
		0.75	0.77	0.76			Medium
		0.81	0.83	0.82			Difficult

Table 12 The evaluation Metrics for Set B using FKGL+

Machine Learning Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL+	0.00	0.00	0.00	0.52	0.0	Medium
		0.52	1.00	0.684			Difficult
		0.00	0.00	0.00			Very Difficult
AdaBoost (RandomForest)	FKGL+	0.56	0.36	0.44	0.60	0.30	Medium
		0.59	0.75	0.66			Difficult
		0.67	0.52	0.59			Very Difficult
MultiClassClassifier (Logistic)	FKGL+	0.52	0.48	0.50	0.63	0.39	Medium
		0.65	0.67	0.66			Difficult
		0.70	0.70	0.70			Very Difficult
Bagging (HoeffingTree)	FKGL+	0.53	0.36	0.43	0.66	0.42	Medium
		0.66	0.77	0.71			Difficult
		0.77	0.74	0.76			Very Difficult
LogitBoost (RandomTRee)	FKGL+	0.53	0.36	0.43	0.63	0.37	Medium
		0.63	0.75	0.68			Difficult
		0.71	0.65	0.68			Very Difficult
SMO (poly kernel)	FKGL+	0.00	0.00	0.00	0.63	0.30	Medium
		0.59	0.94	0.73			Difficult
		0.50	0.63	0.54			Very Difficult
Naïve Bayes	FKGL+	0.38	0.32	0.35	0.57	0.29	Medium
		0.60	0.64	0.62			Difficult
		0.67	0.70	0.68			Very Difficult
Logistic	FKGL+	0.52	0.44	0.48	0.63	0.38	Medium
		0.64	0.71	0.67			Difficult
		0.71	0.65	0.68			Very Difficult
LDA	FKGL+	0.52	0.48	0.50	0.64	0.41	Medium
		0.67	0.67	0.67			Difficult
		0.68	0.74	0.71			Very Difficult

Table 13 The evaluation Metrics for Set A + Set B using FKGL+

ML Algorithm	Features	Precision	Recall	F-measure	Overall Accuracy	Kappa	Class
ZeroR	FKGL+	0.00	0.00	0.00	0.42	0.0	Easy
		0.00	0.00	0.00			Medium
		0.42	1.00	0.59			Difficult
		0.00	0.00	0.00			Very Difficult
Bagging (RandomForest)	FKGL+	0.78	0.76	0.77	0.68	0.52	Easy
		0.63	0.66	0.64			Medium
		0.69	0.76	0.72			Difficult
		0.78	0.30	0.44			Very Difficult
MultiClassClassifier (Logistic)	FKGL+	0.69	0.69	0.69	0.60	0.40	Easy
		0.53	0.45	0.49			Medium
		0.60	0.73	0.66			Difficult
		0.77	0.44	0.56			Very Difficult
AdaBoost (Random Forest)	FKGL+	0.79	0.71	0.75	0.64	0.47	Easy
		0.58	0.67	0.62			Medium
		0.66	0.67	0.66			Difficult
		0.64	0.30	0.41			Very Difficult
LogitBoost (DecisionStump)	FKGL+	0.78	0.69	0.73	0.64	0.46	Easy
		0.61	0.62	0.62			Medium
		0.64	0.71	0.67			Difficult
		0.47	0.30	0.37			Very Difficult
Naïve Bayes	FKGL+	0.58	0.76	0.66	0.58	0.40	Easy
		0.54	0.52	0.53			Medium
		0.63	0.62	0.62			Difficult
		0.60	0.39	0.47			Very Difficult
Logistic	FKGL+	0.68	0.76	0.72	0.62	0.45	Easy
		0.59	0.52	0.55			Medium
		0.63	0.68	0.65			Difficult
		0.68	0.57	0.62			Very Difficult
SMO (puk kernel)	FKGL+	0.75	0.79	0.77	0.64	0.46	Easy
		0.60	0.54	0.57			Medium
		0.63	0.75	0.68			Difficult
		0.75	0.26	0.39			Very Difficult
LDA	FKGL+	0.59	0.62	0.61	0.61	0.43	Easy
		0.57	0.52	0.54			Medium
		0.65	0.70	0.68			Difficult
		0.62	0.57	0.59			Very Difficult

Appendix B

Table 14 Machine Learning algorithms used in the study (Weka 3.8.1)

BayesNet	Naïve Bayes
LDA	Logistic
SimpleLogistic	ZeroR
MultiLayerPerceptron	SMO (puk kernel, poly kernel, RBFKernel)
Trees (DecisionStump, HoeffdingTree, J48, LMT, RandomTree, RandomForest, REPTree)	LogitBoost (DecisionStump, Hoeffding Tree, J48, LMT, RandomTree, RandomForest, REPTree)
AdaBoostM1 (DecisionStump, HoeffdingTree, J48, LMT, RandomTree, RandomForest, REPTree)	MultiClassClassifier (DecisionStump, Hoeffding Tree, J48, LMT, RandomTree, RandomForest, REPTree)
Bagging (DecisionStump, HoeffdingTree, J48, LMT, RandomTree, RandomForest, REPTree)	

Appendix C

Table 15 Maximum and Minimum un-normalized values for Linguistic indices (Set A)

Linguistic Indices	Minimum	Maximum
FKGL (Flesch Kincaid Grade Level)	01.88	13.99
L2 Readability	05.66	38.23
Syntactic Complexity	0.580	0.820
Uncommon /rare Words	14.00	317.0
Lexical Diversity (MTLD; measure of textual lexical diversity)	30.71	130.2
Familiarity	570.9	597.8
Imageability	311.9	395.9
Age of Acquisition (AoA)	04.61	06.28

Table 16 Maximum and Minimum un-normalized values for Linguistic indices (Set B)

Linguistic Indices	Minimum	Maximum
FKGL (Flesch Kincaid Grade Level)	06.15	14.82
L2 Readability	03.36	33.87
Syntactic Complexity	0.610	0.780
Uncommon /rare Words	18.00	120.0
Lexical Diversity (MTLD; measure of textual lexical diversity)	29.71	108.9
Familiarity	572.4	600.1
Imageability	287.3	365.7
Age of Acquisition (AoA)	05.07	06.92

Table 17 Maximum and Minimum un-normalized values for Linguistic indices (Set A + Set B)

Linguistic Indices	Minimum	Maximum
FKGL (Flesch Kincaid Grade Level)	01.88	14.82
L2 Readability	03.36	38.23
Syntactic Complexity	0.580	0.820
Uncommon /rare Words	14.00	317.0
Lexical Diversity (MTLD; measure of textual lexical diversity)	29.71	130.2
Familiarity	570.9	600.1
Imageability	287.3	395.9
Age of Acquisition (AoA)	04.61	06.92

Appendix D

Table 18 Text Difficulty Level-wise Performance metrics

	Class	Sensitivity	Specificity	PPV	NPV	Final Accuracy	Kappa
Set A	Easy	0.75	1.00	1.00	0.92	0.79	0.68
	Medium	0.92	0.72	0.67	0.93		
	Difficult	0.70	0.95	0.89	0.84		
Set B	Medium	0.40	1.00	1.00	0.83	0.72	0.52
	Difficult	0.85	0.58	0.68	0.79		
	Very Difficult	0.78	0.90	0.70	0.93		
Combined Set (A + B)	Easy	0.83	0.92	0.67	0.97	0.65	0.47
	Medium	0.54	0.84	0.64	0.78		
	Difficult	0.73	0.70	0.65	0.78		
	Very Difficult	0.33	0.99	0.67	0.95		

Table 19 Confusion Matrix for Set A

Prediction	Reference		
	Easy	Medium	Difficult
Easy	12	0	0
Medium	4	22	7
Difficult	0	2	16

Table 20 Confusion Matrix for Set B

Prediction	Reference		
	Medium	Difficult	Very Difficult
Medium	4	0	0
Difficult	6	17	2
Very Difficult	0	3	7

Table 21 Confusion Matrix for the Combined Set (A + B)

Prediction	Reference			
	Easy	Medium	Difficult	Very Difficult
Easy	10	4	1	0
Medium	0	14	7	1
Difficult	2	8	24	3
Very Difficult	0	0	1	2

References

Allen, L. K., Jacovina, M. E., & McNamara, D. S. (2016). Cohesive features of deep text comprehension processes. In J. Trueswell, A. Papafragou, D. Grodner, & D. Mirman (Eds.), *Proceedings of the 38th annual meeting of the cognitive science Society in Philadelphia, PA* (pp. 2681–2686). Austin, TX: Cognitive Science Society.

Allen, L. K., Snow, E. L., & McNamara, D. S. (2015). Are you reading my mind? Modeling students' reading comprehension skills with natural language processing techniques. In J. Baron, G. Lynch, N. Maziarz, P. Blikstein, A. Merceron, & G. Siemens (Eds.), *Proceedings of the 5th International Learning Analytics & Knowledge Conference (LAK'15)* (pp. 246–254). Poughkeepsie, NY: ACM.

Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In C. Aggarwal & C. Zhai (Eds.), *Mining text data*. Boston, MA: Springer.

Baayen, R. H., Piepenbrock, R., & Gulikers, L. (1995). The CELEX lexical database (release 2). Distributed by Linguistic Data Consortium, University of Pennsylvania.

Babbar, R., Partalas, I., Gaussier, E., & Amini, M. R. (2013). On flat versus hierarchical classification in large-scale taxonomies. In *Advances in Neural Information Processing Systems*. 1824–1832.

Balyan, R., McCarthy, K. S., & McNamara, D. S. (2017). Combining machine learning and natural language processing to assess literary text comprehension. In X. Hu, T. Barnes, A. Hershkovitz, & L. Paquette (Eds.), *Proceedings of the 10th International Conference on Educational Data Mining (EDM)* (pp. 244–249). Wuhan: International Educational Data Mining Society.

Balyan, R., McCarthy, K. S., & McNamara, D. S. (2018). Comparing machine learning classification approaches for predicting expository text difficulty. In *Proceedings of the 31st Annual Florida Artificial Intelligence Research Society International Conference* (FLAIRS). AAAI Press.

Begeny, J. C., & Greene, D. J. (2014). Can readability formulas be used to successfully gauge difficulty of reading materials? *Psychology in the Schools, 51*(2), 198–215.

Benjamin, R. (2012). Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. *Educational Psychology Review, 24*(1), 63–88.

Bjork, R. A. (1994). Memory and metamemory considerations in the training of human beings. In J. Metcalfe & A. Shimamura (Eds.), *Metacognition: Knowing about knowing* (pp. 185–205). Cambridge, MA: MIT Press.

Bormuth, J. R. (1966). Readability: A new approach. *Reading research quarterly*, pp. 79–132, 1.

Bormuth, J. R. (1969). *Development of Readability Analysis.* (final report, project no. 7-0052, contract no. OEC-3-7-070052-0326). Retrieved from ERIC database. (ED029166).

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

Brunato, D., De Mattei, L., Dell'Orletta, F., Iavarone, B., & Venturi, G. (2018). Is this sentence difficult? Do you agree?. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 2690-2699).

Caruana, R., & Niculescu-Mizil, A. (2006, June). An empirical comparison of supervised learning algorithms. In proceedings of the 23rd international conference on machine learning (pp. 161-168). ACM.

Casasent, D., & Wang, Y.-C. F. (2005). A hierarchical classifier using new support vector machine for automatic target recognition. *Neural Networks, 18*(5–6), 541–548.

Cerri, R., Barros, R. C., & de Carvalho, A. C. (2015, July). Hierarchical classification of gene ontology-based protein functions with neural networks. In 2015 international joint conference on neural networks (IJCNN) (pp. 1-8). IEEE.

Cesa-Bianchi, N., Gentile, C., & Zaniboni, L. (2006). Incremental algorithms for hierarchical classification. *Journal of Machine Learning Research, 7*(Jan), 31–54.

Chall, J. S. (1988). The beginning years. In B. L. Zakaluk & S, J. Samuels (Eds.) readability: Its past, present, and future. Newark, DE: International Reading association.

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.

Coltheart, M. (1981). The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology, 33*(4), 497–505.

Crossley, S. A., Allen, D., & McNamara, D. S. (2012). Text simplification and comprehensible input: A case for an intuitive approach. *Language Teaching Research, 16*, 89–108.

Crossley, S. A., Allen, L. K., Snow, E. L., & McNamara, D. S. (2016a). Incorporating learning characteristics into automatic essay scoring models: What individual differences and linguistic features tell us about writing quality. *Journal of Educational Data Mining, 8*(2), 1–19.

Crossley, S. A., Kyle, K., & Dascalu, M. (2018). The tool for the automatic analysis of cohesion 2.0: Integrating semantic similarity and text overlap. *Behavioral Research Methods.* 1-14.

Crossley, S. A., Kyle, K., & McNamara, D. S. (2016b). 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. https://doi.org/10.3758/s13428-015-0651-7.

Crossley, S. A., & McNamara, D. S. (2009). Computationally assessing lexical differences in second language writing. *Journal of Second Language Writing, 17*, 119–135.

Dale, E., & Chall, J. S. (1948). A formula for predicting readability. *Educational Research Bulletin, 27*(1), 11–28.

Dimitrovski, I., Kocev, D., Loskovska, S., & Džeroski, S. (2011). Hierarchical annotation of medical images. *Pattern Recognition, 44*(10–11), 2436–2449.

Dufty, D. F., Graesser, A. C., Louwerse, M., & McNamara, D. S. (2006). Assigning grade level to textbooks: Is it just readability? In *Proceedings of the 28th Annual Conference of the Cognitive Science Society* Austin, TX: Cognitive science society. In R. Sun and N. Miyake, Eds. 1251–1256.

Dumais, S. T., Platt, J., Heckerman, D., & Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management* (Bethesda, Maryland, USA, November 02–07, 1998). CIKM'98. ACM, New York, NY, 148–155.

Duran, N., Bellissens, C., Taylor, R., & McNamara, D. S. (2007). Quantifying text difficulty with automated indices of cohesion and semantics. In D. S. McNamara & G. Trafton (Eds.), *Proceedings of the 29th annual meeting of the cognitive science society* (pp. 233–238). Austin, TX: Cognitive Science Society.

Feng, L., Jansche, M., Huenerfauth, M., & Elhadad, N. (2010, August). A comparison of features for automatic readability assessment. In *Proceedings of the 23rd international conference on computational linguistics: Posters*, 276–284. Association for Computational Linguistics.

Flesch, R. F. (1948). A new readability yardstick. *Journal of Applied Psychology, 32*(3), 221–233.

François, T., & Miltsakaki, E. (2012). Do NLP and machine learning improve traditional readability formulas? In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations,* pages 49–57, Montreal, Canada, Association for Computational Linguistics.

Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In icml (Vol. 96, pp. 148-156).

Fry, E. (2002). Readability versus leveling. *Reading Teacher, 56*(3), 286–291.

Fuchs, E., Niehaus, I., & Stoletzki, A. (2014). *Das Schulbuch in der Forschung. Analysen und Empfehlungen für die Bildungspraxis*. Göttingen: V&R unipress.

Gee, J. P. (2004). An introduction to discourse analysis: Theory and method. Routledge.

George-Nektarios, T. (2013). Weka classifiers summary. Athens University of Economics and Bussiness Intracom-Telecom, Athens.

Gilhooly, K. J., & Logie, R. H. (1980). Age-of-acquisition, imagery, concreteness, familiarity, and ambiguity measures for 1,944 words. *Behavior Research Methods & Instrumentation, 12*(4), 395–427.

Graesser, A. C., McNamara, D. S., & Kulikowich, J. M. (2011). Coh Metrix: Providing multilevel analyses of text characteristics. *Educational Researcher, 40*, 223–234.

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

Gunning, R. (1969). The fog index after twenty years. *Journal of Business Communication, 6*(2), 3–13.

Hartmann, J., Huppertz, J., Schamp, C., & Heitmann, M. (2019). Comparing automated text classification methods. *International Journal of Research in Marketing, 36*(1), 20–38.

Heilman, M., Collins-Thompson, K., & 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*. Association for Computational Linguistics, Columbus, OH, USA, 71–79.

Ho, T. K. (1995). Random decision forests. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition* (Montreal, QC, august 14–15, 1995). ICDAR'95, IEEE computer society Washington, DC, USA, 278–282.

Jackson, G. T., & McNamara, D. S. (2013). Motivation and performance in a game-based intelligent tutoring system. *Journal of Educational Psychology, 105*, 1036–1049.

Jiang, Z., Gu, Q., Yin, Y., & Chen, D. (2018, August). Enriching word Embeddings with domain knowledge for readability assessment. In *Proceedings of the 27th International Conference on Computational Linguistics,* 366–378.

Johnson, A. M., McCarthy, K. S., Kopp, K. J., Perret, C. A., & McNamara, D. S. (2017). Adaptive Reading and writing instruction in iSTART and W-pal. In proceedings of the 30th Florida artificial intelligence research society international conference (FLAIRS). AAAI Press.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of 10th European Conference on Machine Learning* (April 21-23). ECML'98. Springer-Verlag London, UK, 137-142.

Kate, R. J., Luo, X., Patwardhan, S., Franz, M., Florian, R., Mooney, R. J., Roukos, S., & Welty, C. (2010). Learning to predict readability using diverse linguistic features. In *Proceedings of the 23rd International Conference on Computational Linguistics* (COLING '10). Association for Computational Linguistics, Stroudsburg, PA, USA, 546–554.

Kincaid, J. P., Fishburne, R. P., Rogers, R. L., & Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and Flesch Reading ease formula) for navy enlisted personnel. *Research Branch Report* 8–75, Millington, TN: Naval technical training, U. S. Naval Air Station, Memphis, TN.

Klare, G. R. (1974). Assessing readability. *Reading Research Quarterly, 10*, 62–102.

Klare, G. R. (1984). Readability. In P. D. Pearson, R. Barr, M. L. Kamil, P. Mosenthal, & R. Dykstra (Eds.), *Handbook of Reading research* (pp. 681–744). New York: Longman.

Kotani, K., Yoshimi, T., & Isahara, H. (2011). A machine learning approach to measurement of text readability for EFL learners using various linguistic features. *US-China Education Review B, 6*, 767–777.

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information, 10*(4), 150.

Krogh, A., & Vedelsby, J. (1994). Neural network ensembles, cross validation, and active learning. In *Proceedings of 7th International Conference on Neural Information Processing Systems* (Denver, Colorado). NIPS'94. MIT press Cambridge, MA, USA, 231–238.

Kumar, S., Ghosh, J., & Crawford, M. M. (2002). Hierarchical fusion of multiple classifiers for Hyperspectral data analysis. *Pattern Analysis and Applications, Spl. Issue on Fusion of Multiple Classifiers, 5*(2), 210–220.

Kumar, S., & Ghosh, J. (1999). GAMLS: A generalized framework for associative modular learning systems. In *Proceedings of SPIE conference on applications and science of computational intelligence II*, SPIE proceedings, Orlando, FL, 3722, 24–35.

Kyle, K. (2016). *Measuring syntactic development in L2 writing: Fine grained indices of syntactic complexity and usage-based indices of syntactic sophistication* (Doctoral Dissertation). Retrieved from http://scholarworks.gsu.edu/alesl_diss/35.

Kyle, K., & Crossley, S. A. (2015). Automatically assessing lexical sophistication: Indices, tools, findings, and application. *TESOL Quarterly, 49*(4), 757–786. https://doi.org/10.1002/tesq.194.

Kyle, K., Crossley, S. A., & Berger, C. (2018). The tool for the analysis of lexical sophistication version 2.0. *Behavior Research Methods, 50*(3), 1030–1046.

Lennon, C., & Burdick, H. (2004). The lexile framework as an approach for reading measurement and success. (electronic publication on www.lexile.com).

Lieberman, M. G., & Morris, J. D. (2014). The precise effect of multicollinearity on classification prediction. *Multiple Linear Regression Viewpoints, 40*(1), 5–10.

Malvern, D. D., Richards, B. J., Chipere, N., & Durán, P. (2004). *Lexical diversity and language development: Quantification and assessment*. Houndmills: Palgrave Macmillan.

Martínez, A. M., & Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23*(2), 228–233.

Mayne, A., & Perry, R. (2009, March). Hierarchically classifying documents with multiple labels. In 2009 IEEE symposium on computational intelligence and data mining (pp. 133-139). IEEE.

McCallum, A., & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, tech. Rep. WS-98-05, AAAI press.

McCarthy, K. S., Watanabe, M. , Dai, J., & McNamara, D. S. (in press). Personalized learning in iSTART: Past modifications and future design. *Journal of Research on Technology in Education.*

McCarthy, P. M. (2005). An assessment of the range and usefulness of lexical diversity measures and the potential of the measure of textual, lexical diversity (MTLD). Dissertation abstracts international, 66, UMI no. 3199485.

McCarthy, P. M., & Jarvis, S. (2010). MTLD, vocd-D, and HD-D: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior Research Methods, 42*, 381–392.

McNamara, D. S, Allen, L. K., McCarthy, S. & Balyan, R. (2018). NLP: Getting computers to understand discourse. In *Deep Comprehension* (pp. 224-236). Routledge.

McNamara, D. S., Crossley, S. A., & Roscoe, R. D. (2013). Natural language processing in an intelligent writing strategy tutoring system. *Behavior Research Methods, 45*, 499–515.

McNamara, D. S., Crossley, S. A., Roscoe, R. D., Allen, L. K., & Dai, J. (2015). A hierarchical classification approach to automated essay scoring. *Assessing Writing, 23*, 35–59.

McNamara, D. S., Graesser, A. C., McCarthy, P., & Cai, Z. (2014). *Automated evaluation of text and discourse with Coh-Metrix*. Cambridge: Cambridge University Press.

McNamara, D. S., Graesser, A. C., & Louwerse, M. M. (2012). Sources of text difficulty: Across genres and grades. In J. P. Sabatini, E. Albro, & T. O'Reilly (Eds.), *Measuring up: Advances in how we assess reading ability* (pp. 89–116). RandL Education: Lanham, MD.

McNamara, D. S., Kintsch, E., Songer, N. B., & Kintsch, W. (1996). Are good texts always better? Interactions of text coherence, background knowledge, and levels of understanding in learning from text. *Cognition and Instructions, 14*, 1–43.

McNamara, D. S., Levinstein, I. B., & Boonthum, C. (2004). iSTART: Interactive strategy trainer for active reading and thinking. *Behavioral Research Methods, Instruments, and Computers, 36*, 222–233.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., & Mullers, K. R. (1999, August). Fisher discriminant analysis with kernels. In neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. No. 98th8468) (pp. 41-48). IEEE.

Millis, K., Magliano, J. P., Wiemer-Hastings, K., Todaro, S., & McNamara, D. S. (2007). Assessing and improving comprehension with latent semantic analysis. In T. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *Handbook of latent semantic analysis* (pp. 207–225). Mahwah, NJ: Erlbaum.

National Governors Association Center for Best Practices. (2010). *Common Core State Standards. National Governors Association Center for best practices*. Washington, D. C: Council of Chief State School Officers.

Ozuru, Y., Dempsey, K., Sayroo, J., & McNamara, D. S. (2005). Effect of text cohesion on comprehension of biology texts. In B. G. Bara, L. Barsalou, & M. Bucciarelli (Eds.), *Proceedings of the 27th annual conference of the cognitive science society* (pp. 1696–1701). Mahwah, NJ: Erlbaum.

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

Perfetti, C. A., Landi, N., & Oakhill, J. (2005). The acquisition of reading comprehension skill. In M. J. Snowling & C. Hulme (Eds.), *The science of Reading: A handbook* (pp. 227–247). Oxford: Blackwell.

Perret, C. A., Johnson, A. M., MCarthy, K. S., Guerrero, T. A., & McNamara, D.S. (2017). StairStepper: An adaptive remedial iSTART module. In *Proceedings of the 18th International Conference on Artificial Intelligence in Education (AIED),* Wuhan, China: Springer.

Pilán, I., Vajjala, S., & Volodina, E. (2016). A readable read: Automatic assessment of language learning materials based on linguistic complexity. *International Journal of Computational Linguistics and Applications, 7*, 143–159.

Pilán, I., Volodina, E., & Johansson, R. (2014). Rule-based and machine learning approaches for second language sentence-level readability. In Proceedings of the ninth workshop on innovative use of NLP for building educational applications, Baltimore, Maryland USA, 174–184.

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

Rojas, R. (1996). Neural networks - a systematic introduction. Springer-Verlag, Berlin.

Salsbury, T., Crossley, S. A., & McNamara, D. S. (2011). Psycholinguistic word information in second language oral discourse. *Second Language Research, 27*, 343–360.

Schapire, R. E., & Singer, Y. (1999). BoosTexter: A boosting-based system for text categorization. *Machine Learning, 39*(2–3), 135–168.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA: MIT Press.

Schwarm, S. E., & Ostendorf, M. (2005). Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics* (pp. 523-530). Association for Computational Linguistics.

Schwenker, F. (2000). Hierarchical support vector machines for multiclass pattern recognition. In *Proceedings of 4th KES*, Brighton, UK, 2, 561–565.

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

Snow, E. L., Jacovina, M. E., Jackson, G. T., & McNamara, D. S. (2016). iSTART-2: A reading comprehension and strategy instruction tutor. In *Adaptive educational technologies for literacy instruction,* D.S. McNamara and S. A. Crossley, Eds., Taylor and Francis, Routledge: NY, 104-121.

Stenner, A. J., Horabin, I., Smith, D. R., & Smith, M. (1988). *The lexile framework*. Durham, NC: MetaMetrics.

Sun, A. & Lim, E. P. (2001). Hierarchical text classification and evaluation. In proceedings of the IEEE international conference on data mining (ICDM 2001), San Jose, CA, USA, 29 November–2 December 2001; pp. 521–528.

Sung, Y. T., Chen, J. L., Cha, J. H., Tseng, H. C., Chang, T. H., & Chang, K. E. (2015). Constructing and validating readability models: The method of integrating multilevel linguistic features with machine learning. *Behavior Research Methods, 47*(2), 340–354.

Tanaka-Ishii, K., Tezuka, S., & Terada, H. (2010). Sorting by readability. *Computational Linguistics, 36*(2), 203–227.

Toglia, M. P., & Battig, W. F. (1978). *Handbook of semantic word norms*. Lawrence Erlbaum.

Triguero, I., & Vens, C. (2016). Labelling strategies for hierarchical multi-label classification techniques. *Pattern Recognition, 56*, 170–183.

Vajjala, S., & Meurers, D. (2012, June). 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 (pp. 163-173). Association for Computational Linguistics.

van Dijk, T. A. (1985). Semantic discourse analysis. In T. van Dijk (Ed.), *Handbook of discourse analysis* (Vol. 2, pp. 103–136). London: Academic Press.

Vygotsky, L. (1978) Mind in society: The development of higher psychological processes. (M. Cole, V. John-Steiner, S. Scribner, & E. Souberman, Trans.). Cambridge, MA: Harvard University Press.

Wang, Y.-C. F., & Casasent, D. (2009). A support vector hierarchical method for multi-class classification and rejection. *In Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June, 14-19*, 3281–3288.

Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Zhang, G. P. (2000). Neural networks for classification: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30*(4), 451–462.

Zimek, A., Buchwald, F., Frank, E., & Kramer, S. (2008). A study of hierarchical and flat classification of proteins. *IEEE Transactions on Computational Biology and Bioinformatics, 7*(3), 563–571.

Zipf, G. K. (1949). *Human behavior and the principle of least effort*. Reading, MA: Addison-Wesley.

Affiliations

Renu Balyan[1] · **Kathryn S. McCarthy**[2] · **Danielle S. McNamara**[3]

[1] Ira A. Fulton School of Engineering, Arizona State University, Mesa, AZ, USA

[2] Department of Learning Sciences, Georgia State University, Atlanta, GA, USA

[3] Department of Psychology, Arizona State University, Tempe, AZ, USA