Spring 4-18-2024

Determining Reading Complexity using Regression Models

Joshua Talamayan
joshua.talamayan@student.shu.edu

Determining Reading Complexity Using Regression Models

by

Joshua Talamayan

Thesis Committee

Manfred Minimair, Ph. D., Chair

Nathan Kahl, Ph. D., Committee Member

Kobi Abayomi, Ph. D., Committee Member

A thesis submitted in partial fulfillment of the requirements for the degree of

M.S. Degree in Data Science

Department of Mathematics and Computer Science

Seton Hall University

South Orange, NJ

2024

SETON HALL UNIVERSITY

Department of Mathematics and Computer Science

APPROVAL FOR SUCCESSFUL DEFENSE

Joshua Talamayan has successfully defended and made the required modifications to the text of the master's thesis for the **M.S. Degree in Data Science** during this **Spring Semester 2024**.

DISSERTATION COMMITTEE

Mentor Date

Dr. Manfred Minimair_____

Committee Member Date

Dr. Nathan Kahl_____

Committee Member Date

Dr. Kobi Abayomi_____

Table Of Contents

List Of Tables

List Of Figures

Abstract

Reading is an essential skill when pursuing academic success. In order to assist students in developing their critical reading skills, it is essential to offer texts that are both within their capability, yet challenging enough to push them beyond their boundaries. At the moment, the majority of excerpts are matched to readability scores utilizing tools such as the Flesch-Kincaid Grade Level scoring in order to determine reader level. The core issue with reading scores structured around the Flesch-Kincaid Grade Level scoring is that the scoring process is not readily available to the public while also lacking validation studies. In this project, reading complexity will be determined by applying exploratory data analysis (EDA) and training a readability score that utilizes regression models. This reading score will be utilized in order to determine a passage's reading complexity. A positive target value will denote a simple excerpt of text while a negative target value will denote a more difficult or complex excerpt of text. The focus of this M.S. thesis is to establish an understanding of the 'Target' score in order to establish readability scores as a trusted resource. We will delve deeper into the 'Target' variable and textual excerpts by use of EDA to gather insight on how data should be trained by use of regression models. The regression models output readability and accuracy scores that determine how accurately each regression model is able to determine a proper readability score. The accuracy of a given regression model determines the validity of the regression model's ability to determine readability.

KEYWORDS

regression models, reading complexity, readability, reading scores, excerpt

Acknowledgements

This thesis would not have been possible without the guidance of my advisor, Professor Manfred Minimair. I am grateful for the opportunity to participate in Professor Minimair's research in neural networks and natural language processing in 2017. Thanks to him, I was able to harbor my passion in data science and delve deeper into the field of data science and possibilities provided through its applications. Both the research and regular classes conducted by Professor Minimair became a motivation to the construction of my thesis. I would also like to thank him for allowing me the opportunity to be in the M.S. Data Science Program at Seton Hall, which offers many extensive courses in advanced statistics, data science, and project management.

By extension, I would like to thank Professor Abayomi for his guidance in statistics and operations. I was able to learn more about statistics and its application to data science under his guidance, as well as the different aspects of mathematics such as operations research. Furthermore, I would like to thank Professor Kahl for his guidance in the data mining course at Seton Hall which allowed me to understand data science concepts used in this thesis to a deeper level. Both Professor Abayomi and Professor Kahl are also a part of the thesis committee for this thesis, both giving input into the ideology and creation of the data science project and thesis. I would also like to thank Professor Clachar for her course in machine learning, providing guidance in understanding time series analysis.

I would like to thank Sean Ulep for his guidance and willingness to aid in the presentation of my thesis, and for being a great friend of mine who was there for me in my childhood. Of my college friends, I would like to thank Alex Weeks, Jharenson Calderon, Adriano Soares, Alec Goncalves, and Garrett Denton for being good, supportive friends to me in college, creating endless memorable moments with me at Seton Hall. There were many moments at Seton Hall where my course load felt overwhelming in difficulty, but my friends supported and motivated me through these times. Lastly, I would like to mention Joshua Steier, who not only helped me get into data science research under Professor Minimair, but also got me into the data science club at Seton Hall, opening up many doors of opportunities.

Finally, I want to thank my family for supporting me throughout my life. My father, my life mentor, has been a consistent presence to me towards making good decisions and the one who raised me to be the person I am today. The motherly figure in my life, my stepmother, has been supportive of me for all these years growing up. My three sisters, Alex, Sabina, and Isabela, allowed me to be childish, creating fun and loving memories with them. Lastly, I would like to thank my girlfriend, Annie Tsai, for being supportive of my decisions during the pandemic and being there for me mentally and emotionally.

Chapter 1

Introduction

The CommonLit Readability Prize is a competition that focuses on identifying the reading level of excerpts based on a 'target' variable. The company that created the competition CommonLit Incorporated, is a nonprofit education technology organization with millions of teachers and students, providing reading and writing lessons for grades three through twelve. This thesis' objective is centered around rating the complexity of literary passages for grades 3 to 12 based on a 'target' variable in the aforementioned competition (Sha, 2018.; Kurdi, 2020). The 'target' variable is a readability score that needs to be "reverse engineered" by employing machine learning models. By design, the competition hides how the readability score is computed while encouraging a way to approximate and generalize the readability score (Sha,2018.; Kurdi, 2020) within the 'target' variable. The machine learning models that won the competition are determined by the inclusion of text cohesion and semantics (Sung et al., 2015) when regarding the calculation of the readability score and by how competitors approximate the readability score of the 'target' variable. The competition started on May 3, 2021 and ended on August 2, 2021. The winning code by Mathis Lucka of the competition utilized PyTorch (Paszke et al., 2019) and the use of an ensemble method (Dietterich, 2000), which created a final model based on the average of the multiple models that the model creates. The ensemble method (Dietterich, 2000) was utilized on Ridge Regression (McDonald, 2009) in the winning code by Mathis Lucka. In contrast, the work done in this project utilizes multiple regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010) and compares them by use of mean

squared error (MSE) (Hodson et al., 2021). The MSE is an evaluation metric which is used to evaluate the quality of a machine learning model. Using machine learning algorithms, the regression model (González-Garduño & Søgaard, 2017; Feng et al., 2010) will determine the reading score of excerpts by determining the complexity and translating it back into the given target variable to output a readability score. This project's motivation stems from a belief that aiding in the education of others will help build a more informed society. From a data science perspective, learning how to work more deeply with natural language processing (Balyan et al., 2020) will become part of an essential toolkit when working on future data science projects dealing with text. To begin, an understanding of the usages of regression models in determining the target score must be firmly established. Regression models are a type of analysis that is used for estimating the relationship between a given dependent variable and any number of independent variables. Generally, regression models are used to explain a phenomenon or to predict possible outcomes in the future. Regression models may also be used to decide what to do based on given outcomes. In the context of this project, regression models are employed in order to understand how the 'target' variable is calculated for readability. Next, the usage of a mean square error (Hodson et al., 2021) as an evaluation metric to determine the accuracy of each regression model will need to be thoroughly described and understood. After the background knowledge for both the regression models and evaluation metric is set, the data can be read in and preprocessed. For the CommonLit Readability Prize competition, the data and its features are described in Chapter 3. After data and features are described, the data is analyzed, which then assists in the determination of the features that are essential for training the 'target' variable. Feature engineering (Nargesian et al., 2017), a process in machine learning that is

utilized to increase the quality of results within a machine learning model by extracting the most important features, is not particularly necessary for the training and testing of datasets due to the limited features in each of the datasets. The limited features that are necessary to be utilized for analysis in this project are the 'target' variable, standard error, and 'excerpt' variable. Next, the data is read in and the 'excerpt' variable is preprocessed in order to easily evaluate the data utilizing regression models. Preprocessing the 'excerpt' variable allows for the texts within the created variable 'excerpt_preprocessed' to become cleaned into a readable format for training purposes. As the data is being read in, any IDs that contain a target score and standard error score of 0.0 are removed in order to reduce redundancy and ensure that all data entries with no proper data are removed. In the preprocessing of the 'excerpt' variable, the differences in the structure of the passage of text for the 'excerpt' variable and the 'excerpt_preprocessed' variable are compared. EDA, which aids in understanding and visualizing the data, is applied to look deeper into the 'target' variable, 'excerpt' variable, and 'excerpt_preprocessed' variable to try and find comparisons in the data (Aldera et al., 2021) and give insight into how the 'target' variable's readability score is established. To look deeper in the data, the 'target' variable and its standard error are examined to get a brief understanding of the complexity of the excerpts that will be trained. Alongside the exploration of the 'target' variable, in order to view how each of the attributes contribute towards determining a target score, multiple other attributes are also observed, such as word count, average word length, sentence count, and character length amongst the 'excerpt' and 'excerpt_preprocessed' variables (Balyan et al., 2020). After exploring both the 'excerpt' and 'excerpt_preprocessed' variables, these variables are compared side by side to determine the effect of each attribute on the target score. It is important to see how

‘excerpt’ and ‘excerpt_preprocessed’ variables affect the ‘target’ variable and its output readability score in order to gain insight into how the readability score is engineered for the competition. Knowing how the readability score is engineered will help understand how each of the regression models perform. Additionally, multiple regression analyses (González-Garduño & Søgaard, 2017; Feng et al., 2010) are run to determine which method has the highest success rate for determining the target score of excerpts, in which the most successful methodology will be implemented towards the hope that it will contribute towards an end goal of readily usable readability scores.. In training the data, it becomes apparent which regression models end up being the most useful based on a MSE accuracy score, in which a lower MSE score indicates a more accurate result, and vice versa. Afterwards, the results of the regression models are output, showcasing the differences of each result. In this specific project, both regression models and a MSE (Hodson et al., 2021) become the focal points for establishing reading complexity. This current iteration of the readability project retains open issues that need to be determined and addressed alongside what has yet to be explored, as well as what could be improved upon in any future work done on the topic of readability. Lastly, understanding this project’s ethical implications and ramifications (Newson & Wrigley, 2017) are important to the real world application of the work done in this project. Understanding the ethicalities of accessible reading ease scoring and how it applies to education of younger generations will aid critical choices made to the development of the creation of accessible reading ease scoring formulas that are meant to be used as educational aid, providing healthy working practices in the field of machine learning and readability. Determining the impacts of readability on real world problems and infrastructures will direct a focus towards the importance of the development of machine

learning (Sung et al., 2015; Balyan et al., 2020) and natural language processing (Chan et al., 2021; Sha, 2018; Balyan et al., 2020; Romanov et al., 2019; Wang & Hu, 2021).

.

Chapter 2

Background Knowledge

We will review prior background knowledge used to build the proposed analysis of this thesis. Relevant material mentioned throughout the thesis will also be covered.

2.1 Text Analysis

This section will go over different aspects of text analysis involving readability, text preprocessing, and exploratory data analysis used throughout this data science project.

2.1.1 Readability

As noted in Chapter 1, a ‘target’ variable is utilized in order to determine text readability, similar to other readability tools such as the Flesch-Kincaid Grade Level scoring which determines a reader’s level. Since there is no indication of how the ‘target’ variable is created from The CommonLit Readability Prize, it is not determined how the readability score of the ‘target’ variable is created. Improving the accuracy of a readability score in regards to text enables a larger audience to gain an understanding of difficult topics, allowing individuals to strive further in competitive subjects such as mathematics and science. In fact, a study by Akbaşlı et al. (Atmazaki et al., 2018) notes countries that hosted high readability scores also had a correlation with high mathematics and science scores.

In order to understand how the ‘target’ variable is structured, it is important to know what aspects make texts or excerpts readable, and the importance of each of those aspects. Some

aspects of readability that will be focused on in this section include vocabulary, sentence length, and word length. In fact, these are the most commonly associated aspects in regards to establishing readability scores. However, it must be established that vocabulary, sentence length, and word length do not summarize all aspects of readability score.

Regarding vocabulary, Zamanian et al., states that "not only do humans tend to use some words much more often than others, they recognize more frequent words rapidly than less frequent, prefer them, and understand and learn them more readily (Zamanian & Heydari, 2012, p.44)". It is not surprising therefore, that this variable has such a central role in the measurement of readability. Word difficulty concerning vocabulary can be attributed to its frequency, whereas infrequent words tend to be more difficult than words that commonly appear in texts. When words appear more frequently, the reader will become more familiar with the words, which in turn allows the reader to understand the word's usage (Breland, 1996).

Sentence length is an important factor for readability in two different aspects: context and the quantity of syllables in a sentence (Zamanian & Heydari, 2012). Generally, sentences should be long enough to provide context to the reader while also being short enough for the reader to be able to process the information. Word length is measured by the amount of syllables within a word, in which readability becomes more difficult the more syllables a word contains. To simplify passages that are difficult to read in regards to readability score, the replacement of high-syllable words with their shorter-syllabled synonyms to convey the same meaning within a passage is necessary.

The general aspects commonly utilized to create modern readability formulas that classify reading materials include both word and sentence lengths (Zamanian & Heydari, 2012).

Most modern readability formulas focus on limited aspects of readability such as sentence length and word complexity. Modern readability formulas do not consider the target audience, “text cohesion, complexity of ideas, and required schemata (Zamanian & Heydari, 2012, p.47)”. With that in mind, readability formulas should only be used as a guideline to how difficult a passage of text may be.

2.1.2 Text Preprocessing

In terms of natural language processing (Chan et al., 2021; Sha, 2018; Balyan et al., 2020; Romanov et al., 2019; Wang & Hu, 2021), the main idea of text preprocessing (Eke et al., 2021) involves cleaning text data that is used to feed the data into a training model. Text preprocessing (Eke et al., 2021) is a necessary step when working with text since text could be noisy (i.e. contains symbols/punctuation marks/typos, usage of stop words). The end goal for preprocessing text (Eke et al., 2021) is to remove as many noisy details from the text as possible and only include important or meaningful words for the training model. Removing noisy data from text will lead to higher quality data, which will lead to a more reliable training model. Generally when preprocessing text, the following steps are helpful:

- Expanding contractions
- Making all text lowercase
- Removing punctuation
- Removing stop words
- Removing blank spaces
- Tokenization
- Lemmatization

The above mentioned methods all aid in the simplification of text data to create a quality data set that can be used in a training model. It is important to note that the above mentioned methods are not all necessarily possible methods to preprocessing text data (Eke et al., 2021). There may be a need for other text preprocessing (Eke et al., 2021) methods in informal texts such as emails or text messages, such as removal of emoticons or emojis.

Expanding contractions is the process by which words such as “don’t” are expanded into “do not”, or “didn’t” are expanded into “did not”. Keeping contractions within a text dataset allows for unnecessary noise within the text data. Expanding contractions allows for better text analysis.

Making all text lowercase simplifies the data and makes training the data more consistent when running natural language processing tasks (Hickman et al., 2020). It is important to make all text the same case because lowercase and uppercase letters are treated differently when being trained. Using all lowercase words is preferred for text preprocessing (Eke et al., 2021).

Punctuations within text add ambiguity, which creates more noise within the text data. Similar to lowercase and uppercase words, words are treated differently due to punctuation. For example, “data” is treated differently than “data?” when being trained. Normally when removing punctuations within text data, one must be wary of contraction words such as “don’t” or “didn’t” within the data, since text analyzing words such as “don t” or “didn t” will not have any meaning in regards to training the data (Hickman et al., 2020).

Stop words (Mohan, 2015) are words that do not add any information to text data. The words that are considered stop words (Mohan, 2015) are generally the most common words in a given spoken language and don’t give information concerning the tone of a text. A few examples

of stop words are words such as “a”, “the”, and “this”. Stop words (Mohan, 2015; Kannan & Gurusamy, n.d.) are removed if training the data does not account for sentence structure.

Removing blank spaces (Kannan & Gurusamy, n.d.) in text data is necessary to save on memory for running the program that is used to text preprocess and train the data. Alongside saving memory, the text that is being analyzed becomes easier to read. The removal of blank spaces (Kannan & Gurusamy, n.d.) tidies up the text for those who are analyzing the text data.

Tokenization (Kannan & Gurusamy, n.d.) is the process by which text gets split up into smaller units. The most common use of tokenization is known as word tokenization (Kannan & Gurusamy, n.d.), where large groups of text (i.e. sentences or paragraphs) get split into individual words. The smaller units that are output by tokenization are called tokens (Kannan & Gurusamy, n.d.). Word tokenization occurs before lemmatization, since the words become easier to process within lemmatization (Plisson, Lavrac, & Mladenic, n.d.) after they are tokenized. Tokenization (Kannan & Gurusamy, n.d.) is one of the most essential pieces to text preprocessing as it allows a machine to count the number of words within a given text and count word frequency.

Lemmatization (Plisson, Lavrac, & Mladenic, n.d.) is the process by which a word is simplified to its root form. The process of lemmatization (Plisson, Lavrac, & Mladenic, n.d.) accounts for the prefix or suffix of a word back to their root form. For example, words such as “running” or “runs” would become simplified into their root form, “run”. Lemmatization (Plisson, Lavrac, & Mladenic, n.d.) looks at the full language’s vocabulary and applies morphological analysis (Álvarez & Ritchey, 2015) to words. For instance, the word “were” is derived from the word “be”, therefore any text that has the word “were” would change into the word “be”. Lemmatization (Plisson, Lavrac, & Mladenic, n.d.) aids in finding word frequency

among multiple texts and reducing the amount of words that need to be analyzed within text analysis (Chan et al., 2021).

2.1.3 EDA

EDA (Aldera et al., 2021) is an integral part of text analysis (Chan et al., 2021). In terms of text analysis (Chan et al., 2021), the process of EDA aids in summarizing text data and looking into the main characteristics of text (Morgenthaler, 2009). In this data science project, the ‘target’ variable, ‘standard_error’ variable, ‘excerpt’ variable, and ‘excerpt_preprocessed’ variable are observed in the EDA. Different aspects of each of these aforementioned variables are observed closely, allowing for the formulation of hypotheses and generating insight based on trends seen within the data. Observing the data through EDA (Aldera et al., 2021; Morgenthaler, 2009) helps to potentially discover hidden patterns or anomalies found within the data. EDA is a tool that aids in creating proper assumptions of the data, since there will be better understanding of patterns, outliers, and pairings of interesting relations within the data.

Generally, EDA is summarized through visualizations, such as tables, charts, and graphs (Glymour, 1998). When properly executed, EDA (Morgenthaler, 2009) provides insight into what type of statistical tools and techniques should be used for text analysis. EDA gives a blueprint of how the data may be manipulated in order to find solutions to hypotheses and create assumptions of the ‘target’, ‘excerpt’, and ‘excerpt_preprocessed’ variables.

2.2 Regression Models

This section will go over the multiple different types of regression models that were utilized to predict readability scores.

2.2.1 Introduction To Regression Models

Regression models utilize a mathematical equation as a model in order to predict a continuous or discrete outcome (Killada, 2017). Regressions are used to find a quantifiable relationship between independent variables and a dependent variable (Killada, 2017). A standard regression model is represented by the following formula:

$$Y_i = f(X_i, \beta) + e_i$$

Where Yi is represented as the dependent variable, Xi is represented as the independent variable, □ represents the unknown parameters, and ei represents the error terms.

A total of seven regression models were utilized to determine the 'target' variable. The following regression models were tested and compared using MSE:

- Random Forest Regression
- Gradient Boosting Regression
- Support Vector Regression
- AdaBoost Regression
- XGBoost Regression
- Ridge Regression
- Linear Regression

Listed below are the libraries/packages used for each of the seven regression models:

Figure 1

List of regression model libraries/packages

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.svm import SVR
from sklearn.ensemble import AdaBoostRegressor
import xgboost as xgb
from sklearn.linear_model import Ridge
from sklearn.linear_model import LinearRegression
```

In the upcoming sections, the mathematical formulations, advantages, and disadvantages of each regression models is discussed.

2.2.2 Ensemble Learning

Ensemble learning (Dietterich, 2000) is a technique used in machine learning that looks to improve predictive performance by combining predictions from multiple models. Four of the seven regression models utilized in this data science project (random forest regression, gradient boosting regression, AdaBoost regression, and XGBoost regression) use ensemble learning methods in order to produce an output value. There are three methods of ensemble learning: bagging, stacking, and boosting (Dietterich, 2000). Bagging and boosting ensemble learning (Dietterich, 2000) were utilized in this project, and therefore will be the two that are discussed in section 2.2.2.

Bagging (Dietterich, 2000) is the process that involves fitting many decision trees among different samples within the same dataset and taking the average of the predictions of the different samples. Bagging (Dietterich, 2000) is also known as bootstrapping aggregation.

Bootstrapping (Dietterich, 2000) consists of generating samples based on the initial dataset using the replacement method. The generated samples are produced by randomly selecting samples within the initial dataset. Bootstrapping (Dietterich, 2000) utilizes resampling techniques in order to randomize the selection procedure. Aggregation (Dietterich, 2000) combines all seven models together for a final prediction while considering all possible outcomes. Generally, regression models that utilize bagging ensemble learning start with high variance. Bagging mainly focuses on achieving an ensemble model with less variance. Random forest regression is an example of the bagging ensemble learning.

Boosting (Mohan, Chen, & Weinberger, 2011) is the process which involves adding multiple weak learners sequentially, combining the weak learners, which creates a strong learner with enhanced performance. As the weak learners are fitted into the model, weak learners are added sequentially. Sequentially added, weak learners will bring attention to observations that were handled poorly in previous models within the sequence, focusing on observations that previously fit poorly. The base models used in boosting ensemble methods (Mohan, Chen, & Weinberger, 2011; Dietterich, 2000) often have low variance and high bias. At the end of the boosting ensemble learning process, a strong learner with a lower bias is produced, which produces a stronger predictive model. Gradient boosting regression, AdaBoost regression, and XGBoost regression are examples of boosting ensemble learning.

2.2.3 Random Forest Regression

Random forest regression (Ho, 2016) models are considered to be a supervised learning algorithm in machine learning that utilizes ensemble learning for classification and regression purposes. Random forests typically construct multiple decision trees during training, where each

tree is built using a random subset of features and a random subset of data points, and then the final prediction is obtained by averaging or taking a majority vote of the predictions of all the trees. In this project, Scikit learn, a machine learning Python library, is used to construct multiple decision trees during training, using a parameter called *n_estimators*. The default amount of decision trees created by *n_estimators* is 100. If the quantity of decision trees utilized for random forest regression prove insufficient, then that would lead to results with high variance, which would then lead further to unreliable output results.. On the other hand, if there are a surplus of decision trees utilized for random forest regression, that could lead to excessive noise in the data which would hinder the production of reliable output results.

The type of decision trees that random forest regression utilizes are known as classification and regression trees (CART) (Segal, 2004), which are standard decision trees. Many trees are created and the predictions are combined in order to create a final prediction. Bootstrapping (Doğan, 2017) is used to create multiple different samples from the original data, in which these samples will maintain the same size while containing differing distributions of observation. Bootstrapping is important for random forest regression since the bootstrapping procedure decreases variance and maintains bias within the data. Furthermore, features are randomized in order to minimize the correlation between the multiple decision trees.

Random forest regressions operate by taking a subset of the data for the training dataset. Next, the algorithm creates clusters utilizing the data and categorizes the data into multiple groups and subgroups, thus producing randomized decision trees. The variables within each of the decision trees are random as well. Afterwards, the rest of the dataset, known as the test

dataset, is utilized in order to make a prediction about which decision tree would be able to best fit the dataset.

2.2.4 Gradient Boosting Regression

Gradient boosting (Mohan, Chen, & Weinberger, 2011) is a machine learning technique that utilizes ensemble learning and boosting algorithms and is typically known to be an optimization algorithm. Gradient boosting algorithms can be used for both classification and regression problems. Gradient boosting regression (Mohan, Chen, & Weinberger, 2011) calculates the difference between the current prediction made within the training model and the known correct target value. The difference calculated is known as the residual. Gradient boosting regression (Mohan, Chen, & Weinberger, 2011) trains weak models that map features to the calculated residual.

Decision trees are used as the weak learners within gradient boosting. Gradient boosting is an additive regression model since weak learners are added on one at a time while retaining existing decision trees within the model. Boosting algorithms (Mohan, Chen, & Weinberger, 2011) combine weak learners, in order to create strong "learners". As the boosting algorithm combines weak learners to create strong learners, the model becomes a stronger predictor of the actual value. Weak learners are generally defined as models that are not highly correlated with the classification, while strong learners highly correlate with the classification. Generally, a weak learner in decision trees are called decision stumps (Ayinde et al., 2013), which are decision trees with a single split (consisting of one node and two leaves).

The boosting algorithms used in gradient boosting regression typically use a loss function which is optimized with gradient descent. Gradient boosting regressions start off by making an

initial guess within the dataset. Due to the nature of boosting algorithms, gradient boosting regression generally requires more computation time in order to train the data.

2.2.5 Support Vector Regression

Support vector machines (Awad & Khanna, 2015) are supervised learning models in machine learning that utilize associated learning algorithms for classification and regression purposes. Support vector regressions are built based on the concept of support vector machines, using the same principles as support vector machines strictly for regression problems. There are three important hyperparameters for Support vector regression: hyperplane, kernel, and boundary lines.

First, Support vector regression models attempt to try and fit the best line within a threshold value in order to represent the relationship between the input and output variable, while also minimizing errors within the threshold value. The line that is used to fit the data is known as the hyperplane, which is used to predict a continuous output. The data points that are closest to the hyperplane (Ayinde et al., 2013) are known as support vectors. The support vectors are utilized in order to create a predicted output for the support vector regression algorithm.

Secondly, kernels enable the possibility of mapping data into a higher-dimensional space, potentially making it linearly separable. Kernels are able to organize the data within a datasets as input and transform the data into a readable form that is able to be used to find the hyperplane within higher dimensional spaces.

Thirdly, boundary lines are two lines that are drawn around the hyperplane. Drawing boundary lines around the hyperplane helps define the margin, which is the distance between the hyperplane and the closest data points, and allows for the classification or regression of new data

points based on their position relative to these boundary lines. Support vectors can be on the boundary line or outside of the boundary lines.

2.2.6 AdaBoost Regression

AdaBoost is a supervised learning algorithm that utilizes ensemble learning that can be employed for classification and regression purposes. AdaBoost is a meta algorithm (Arora, Hazan, & Kale, 2012), in which a meta algorithm is an algorithm that is generally used in conjunction with other algorithms for performance improvement. Similar to gradient boosting regression, AdaBoost utilizes decision trees as weak learners. AdaBoost utilizes boosting algorithms, as mentioned in section 2.2.4, combines weak learners, and combines these weak learners into strong learners, where the weak learners are generally decision stumps.

Weighting in the scope of AdaBoost regression serves to emphasize the importance of certain data points in the training process. When examining weighting, difficult classifications are weighted more and easier classifications are weighted less. The decision stump (Ayinde et al., 2013) that is utilized for AdaBoost regression is the decision stump that yields the best accuracy.

2.2.7 XGBoost Regression

XGBoost regression (Chen & He, 2016) is a variant of gradient boosting regression, and generally is considered to yield better results than gradient boosting regression due to its combination of several advanced techniques and optimizations. XGBoost notably yields highly accurate results quickly and outclasses other regression models in terms of accuracy. XGBoost regression, through an implementation of gradient boosting which modifies the gradient boosting algorithm to create more accurate approximations to find the best model. XGBoost regressions

compute second-order gradients of the loss function which helps provide information about the direction of the gradients. Second-order gradient computation of the loss function also aids in figuring out the minimization of the loss function in order to get the best results. When talking about second-order gradient computation, it's not just about looking at how steep the slope is, but also about how the slope is changing. This helps in figuring out the best way to adjust our model to minimize errors and get better results. XGBoost regression utilizes advanced regularization techniques, known as L1 and L2 regularization methods (Cortes, Mohri, & Rostamizadeh, 2009), which assist in generalizing the model by penalizing large coefficient values, thereby reducing overfitting and improving the model's ability to generalize to unseen data.

2.2.8 Ridge Regression

"Ridge regression is a method of estimating the coefficients of multiple-regression models in scenarios where linearly independent variables are highly correlated (Hilt & Seegrist, 1977, p.4)." Ridge regression performs L2 regularization (Cortes, Mohri, & Rostamizadeh, 2009), and any regressions that make use of L2 regularization are considered a ridge regression. Ridge regression is taken into account as a model tuning method that is used to analyze data that suffers from multicollinearity (Cortes, Mohri, & Rostamizadeh, 2009). Multicollinearity (Cortes, Mohri, & Rostamizadeh, 2009) refers to a situation in which two or more predictor variables in a regression model are highly correlated with each other. Multicollinearity (Cortes, Mohri, & Rostamizadeh, 2009) occurs when least-squares are unbiased and the variances of the data are large, which means that the output predicted values are largely inaccurate compared to the desired values. Under L2 regularization (Cortes, Mohri, & Rostamizadeh, 2009), the parameters are shrunk down, preventing multicollinearity (Farrar & Glauber, 1967) and model complexity is

reduced by shrinking the coefficient. The objective of L2 regularization (Cortes, Mohri, & Rostamizadeh, 2009) used in ridge regression is to make sure overfitting does not occur in the model by lowering variance while increasing bias in the data.

When working with ridge regression, the variables should be standardized by subtracting the mean of the variable and dividing by the variable's standard deviation. It is important to standardize the ridge regression variables as ridge regression is calculated based on standardized variables. Bias and variance increase in ridge regression when λ increases.

2.2.9 Linear Regression

Linear regression is the first type of regression analysis that was studied heavily by statisticians and researchers, and the first to be used in practical applications. Essentially, linear regression is the fundamental method for all previously discussed regression methods. There are two types of linear regressions that are generally used in machine learning, consisting of simple linear regression and multiple linear regression. Simple linear regression consists of one independent variable while multiple linear regression will consist of more than one independent variable. Linear regression in statistics is defined as finding a line that most accurately fits the data points in a given plot. The main goal of a linear regression model is to minimize error by defining optimal values for the intercept and coefficients. Linear regression creates a linear relationship (Kumari, 2018) between the independent variable and dependent variable. The linear relationships are numerically related amongst the independent and dependent variables. By nature, linear regressions have constant variance.

Linear regressions perform well when the data is linearly separable. The hyperplane of a simple linear regression is a straight line. As previously discussed in 2.2.4, hyperplanes are the line that are used to fit the data.

2.3 Evaluation Metric

This section will predominantly talk about mean squared error (MSE) (Hodson et al., 2021) and why it is used for the project in this thesis.

2.3.1 Mean Squared Error

MSE is an evaluation metric that is used to compare the performance of different regression models. MSE measures a quantifiable amount of error within a statistical model, in this case, regression models. A regression model has less errors as data points fall closer to the regression line (Hodson et al., 2021), which implies the regression model will have more precise predictions.

In particular, MSE value depicts how well observed data fits and mirrors expected data. Among other evaluation metrics, MSE is a valid method to compare regression models if the given values outputted from MSE are not large values, since the potentially large values output by MSE could make it difficult to compare which regression model(s) is better as a training model. If the output values of MSE become too large, it is advisable to instead use root MSE (Willmott & Matsuura, 2005) as the evaluation metric. Root MSE takes the square root of the MSE value, which could take large values output by MSE and reduce the values down to smaller values which make it easier to interpret which regression model is best for accurate predictions.

2.3.2 Mean Square Error Formula

Mean squared error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Where n represents the number of data points within the dataset, Yi represents an observed value, and $\widehat{Yi}$ represents the predicted values. The difference between the observed values and predicted values are squared in order to remove negative values, which ensures that the MSE is always a value that is greater than or equal to zero. The closer that a MSE value is to zero, the less errors the regression model has.

The MSE evaluates the average square difference between observed and predicted values. The output of the MSE is an absolute number which indicates how much the predicted results deviate from the actual numbers. The absolute values produced by multiple regression models will help identify which model yields the best results.

Chapter 3

Data Description And Preprocessing

The data set contains 2841 data entries in which 2834 of such entries are in the training set, with the remaining 7 in the test set. Alongside the training and test sets, the competition provides file ‘sample_submission.csv’. This file contains the table formatting required for submission in the competition. The 5 features in the training data set include ‘url_legal’, ‘license’, ‘excerpt’, ‘target’, and ‘standard_error’. The variable ‘url_legal’ contains the URL of the source text, also referred to as the ‘excerpt’ variable in the data The ‘license’ variable contains the license of source material. The ‘excerpt’ variable holds a collection of text which we will be using to determine a target score based on the complexity of each excerpt. The ‘target’ variable is a positive or negative number denoting the determined difficulty of each excerpt. The lower this number is, the more difficult the excerpt is. The opposite stands true. Lastly, the ‘standard_error’ contains the measure of the spread of scores among multiple rates for each excerpt in the dataset. Of the 5 features, the test set only contains 3 of the features; the ‘url_legal’, ‘license’, and ‘excerpt’ variables.

Figure 2

Variable names and short description

`id` - unique ID for excerpt

`url_legal` - URL of source - this is blank in the test set.

`license` - license of source material - this is blank in the test set.

`excerpt` - text to predict reading ease of

`target` - reading ease

`standard_error` - measure of spread of scores among multiple raters for each excerpt. Not included for test data.

The main focus of this project will be on the 'excerpt' and 'target' variables since the goal is to create regression models that will take each of the excerpts as an input while returning the reading complexity of each of the excerpts within the 'target' variable as an output (González-Garduño & Søgaard, 2017; Feng et al., 2010). For instance, during data processing, each excerpt undergoes random forest regression (Romanov, Lomotin, & Kozlova, 2019). The output values are then calculated, resulting in a "target" score. This score signifies the level of reading complexity as determined by the random forest regression model's interpretation of the "excerpt" variable. Table 1 contains an example of this case in the data with the five features mentioned.

Table 1

Example of data instance in training dataset

	id	url_legal	license	excerpt	target	standard_error
0	c12129c31	NaN	NaN	When the young people returned to the ballroom...	-0.340259	0.464009

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

3.1 Reading In And Preprocessing The Data

The training and test data sets are read in to acquire the data, in which the training set contains 2834 data entries and the test set contains 7 data entries. The training set was split 70/30. The submission file for a reference example of the output results in the test data is demonstrated in the previous example above

3.1.1 Training Dataset

For the training dataset, any entries containing a target and standard error value of 0 are removed. These entries are removed to ensure that there are no entries with null values. Null values are considered to be undefined values that further complicate the EDA and training of the regression models. Removing null values will increase the performance of the regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010), allowing for a more accurate comparison of the regression models using MSE (Hodson et al., 2021).

After the entries are removed, the train data is observed to ensure that the data is read in correctly. Table 2 contains an example of the first 5 data entries in the training dataset in order to ensure that the data was read in correctly.

Table 2

Example of the first five data instances in the training dataset

	id	url_legal	license	excerpt	target	standard_error
0	c12129c31	NaN	NaN	When the young people returned to the ballroom...	-0.340259	0.464009
1	85aa80a4c	NaN	NaN	All through dinner time, Mrs. Fayre was somewh...	-0.315372	0.480805
2	b69ac6792	NaN	NaN	As Roger had predicted, the snow departed as q...	-0.580118	0.476676
3	dd1000b26	NaN	NaN	And outside before the palace a great garden w...	-1.054013	0.450007
4	37c1b32fb	NaN	NaN	Once upon a time there were Three Bears who li...	0.247197	0.510845

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

3.1.2 Preprocessing The Data

To prepare the data for analysis, preprocessing steps are applied to both the training and testing datasets, involving the editing of each entry within the 'excerpt' variable. Preprocessing encompasses tasks such as removing special characters, tokenization, and converting text to lowercase to ensure uniformity and suitability for machine learning algorithms. Editing tasks include removing foreign texts and unnecessary blanks, converting the text to lowercase, tokenizing the words utilizing the Natural Language Tool Kit (NLTK) library (Wang & Hu, 2021), removing any non-English verbiage, and utilizing a word lemmatizer from the NLTK library (Eke, Norman, & Shuib, 2021). Word lemmatization is generally used to simplify words down to the base, or dictionary form which would enhance the performance of the regression models.

The variable then becomes defined as the ‘excerpt_preprocessed’ variable. This variable is now the newly cleaned ‘excerpt’ variable which will be used for training the regression

models. Figure 3 and Figure 4 visualize the first data entry in the training set, comparing changes in the ‘excerpt’ variable with the aim of creating the ‘excerpt_preprocessed’ variable.

Figure 3

Example of passage of text in the first data entry (‘excerpt’)

```
#Excerpt
print(train_df.iloc[0,3])
```

```
When the young people returned to the ballroom, it presented a decidedly changed appearance. Instead of an interior scene, it was a winter landscape.
The floor was covered with snow-white canvas, not laid on smoothly, but rumpled over bumps and hillocks, like a real snow field. The numerous palms and evergreens that had decorate
d the room, were powdered with flour and strewn with tufts of cotton, like snow. Also diamond dust had been lightly sprinkled on them, and glittering crystal icicles hung from the
branches.
At each end of the room, on the wall, hung a beautiful bear-skin rug.
These rugs were for prizes, one for the girls and one for the boys. And this was the game.
The girls were gathered at one end of the room and the boys at the other, and one end was called the North Pole, and the other the South Pole. Each player was given a small flag wh
ich they were to plant on reaching the Pole.
This would have been an easy matter, but each traveller was obliged to wear snowshoes.
young people returned ballroom presented decidedly changed appearance instead interior scene winter landscape floor covered snow white canvas laid smoothly rumpled bump hillock lik
e real snow field numerous palm evergreen decorated room powdered flour strewn tuft cotton like snow also diamond dust lightly sprinkled glittering crystal icicle hung branch end r
oom wall hung beautiful bear skin rug rug prize one girl one boy game girl gathered one end room boy one end called north pole south pole player given small flag plant reaching pol
e would easy matter traveller obliged wear snowshoe
```

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 4

Example of passage of text in the first data entry (‘excerpt_preprocessed’)

```
#Exerpt Preprocessed
print(train_df.iloc[0,6])
```

```
young people returned ballroom presented decidedly changed appearance instead interior scene winter landscape floor covered snow white canvas laid smoothly rumpled bump hillock lik
e real snow field numerous palm evergreen decorated room powdered flour strewn tuft cotton like snow also diamond dust lightly sprinkled glittering crystal icicle hung branch end r
oom wall hung beautiful bear skin rug rug prize one girl one boy game girl gathered one end room boy one end called north pole south pole player given small flag plant reaching pol
e would easy matter traveller obliged wear snowshoe
```

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

3.2 Understanding The Target Variable

The ‘target’ variable provides a readability score which then determines text complexity (Balyan, McCarthy, & McNamara, 2020). In this competition, the ‘target’ variable is used to determine the readability score of excerpts. It is not officially determined how the competition organizers generated the ‘target’ variable readability scores which is meant to be predicted

during the competition. The ‘target’ variable’s scoring system is meant to be a more intuitive and accessible approach in determining readability scores when put in comparison to other readability score methods such as the Flesch-Kincaid Grade Level scoring method (Grabeel et al., 2018). The target score in the ‘target’ variable can be a positive or negative numeric value, where readability ease is determined to be easier if the readability score in the ‘target’ variable is a positive numeric value. On the other hand, if the target score in the ‘target’ variable is a negative value, the text is likely to be more complex.

Insight is gathered on how the target score is calculated in chapter 4, where the ‘target’ variable is further discussed and visualized in detail. In chapter 4, the ‘target’ variable is compared to different attributes such as word count and average word length (Chau et al., 2020) within the ‘excerpt’ and ‘excerpt_preprocessed’ variables. Further insight on target score distribution versus attributes in ‘excerpt’ and ‘excerpt_preprocessed’ can be found in Figure 9, Figure 10, and Figure 11. In each set of tables, the target score distribution is determined based on multiple attributes of ‘excerpt’ and ‘excerpt_preprocessed’. A comparison of the aforementioned variables, ‘excerpt’ and ‘excerpt_preprocessed’, and how target distribution is further shown in Figure 12, comparing how target scoring differed before and after the ‘excerpt’ variable became preprocessed for training purposes. EDA will aid in understanding how the ‘target’ variable scores are determined. EDA typically aids in the analysis of data by offering the user more detail of the data by utilizing graphs to visualize the data and exposing patterns in the data.

Chapter 4

Exploratory Data Analysis

The ‘target’, ‘standard_error’, and ‘excerpt’ variables are observed from the training set for the exploratory data analysis (EDA) (Aldera et al., 2021). The ‘excerpt_preprocessed’ variable will also be observed, which stems from the ‘excerpt’ variable. ‘excerpt_preprocessed’ is a cleaned version of the ‘excerpt’ variable, which removes excess redundancy in the passages of text by use of data wrangling (Sarkar & Roychowdhury, 2019). Data wrangling is a form of data cleaning, in which each of the passages of text are transformed into a usable format for analysis. The ‘excerpt_preprocessed’ variable will be explored and the differences between the ‘excerpt’ and ‘excerpt_preprocessed’ variables will be compared, noting how they compare and contrast in regards to their effect on target scoring. The exploratory analysis may be found below within a Kaggle repository: https://www.kaggle.com/code/joshuatalamayan/dasc-project-commonlit-readability-eda.

The main purpose of the EDA is to look at the distribution of the ‘target’ variable and its standard error for the purpose of discovering how scoring for each excerpt is acquired, while also looking at the ‘excerpt’ variable as well as the preprocessed version of the ‘excerpt’ variable in order to determine the changes between both the ‘excerpt’ and ‘preprocessed_excerpt’ and how these changes could affect the accuracy of the regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010).

Understanding the distribution and standard error of the 'target' variable illuminates the types of excerpts that require training, as well as how the distributions may impact each regression model and how each regression model is configured for training. It is important to know how the distributed values of the ‘target’ variable relates to the ‘excerpt’ and ‘excerpt_preprocessed’ variables in this EDA (Aldera et al., 2021). Being able to characterize the ‘target’ variable based on attributes of the ‘excerpt’ and ‘excerpt_preprocessed’ will give us better insight into how a target score is determined. For example, attributes in ‘excerpt’ and ‘excerpt_preprocessed’, such as the word count or average word length, could be compared to a given ‘target’ variable score to determine how such attributes affect the target score. Alongside determining the target score for both the ‘excerpt’ and ‘excerpt_preprocessed’, the target score can be compared amongst the ‘excerpt’ and ‘excerpt_preprocessed’ and how each of the attributes in both the ‘excerpt’ and ‘excerpt_preprocessed’ differ for each of the target scores. Between the ‘excerpt’ and ‘excerpt_preprocessed’ variable, certain attributes such as word count and average word length will change in the passage, generally decreasing in the ‘excerpt_preprocessed’ variable since the text within the ‘excerpt_preprocessed’ is cleaned. The preprocessing done in the ‘excerpt_preprocessed’ variable will change how the target score is calculated in each of the regression models, providing a different result than that of the ‘excerpt’ variable.

4.1 ‘Target’ Variable

In this section, analyzing the distribution of the 'target' variable and 'standard_error' aids in determining the range of the 'target' variable and provides valuable insights into its

distribution. The distribution of the ‘target’ variable will determine how the visualization of attributes in the ‘excerpt’ and ‘excerpt_preprocessed’ variables will be distributed since both the ‘excerpt’ and ‘excerpt_preprocessed’ variables will be visualized utilizing the target scores. After running the distribution for the ‘target’ variable, the ‘target’ variable had a minimum value of -3.676267773 and a maximum value of 1.711389827. The graph itself is a bell curve (Glymour, 1998) in which frequency indicates the instances that a target value range is shown within the data entry which is shown in Figure 6. In Figure 5, it is important to note that most of the values within the data are negative values and the value of the ‘target’ variable is -0.203078529 at the 75th percentile. This indicates that a majority of the data in the training dataset are negative values, meaning that a majority of the excerpts are more complex in terms of readability.

Figure 5

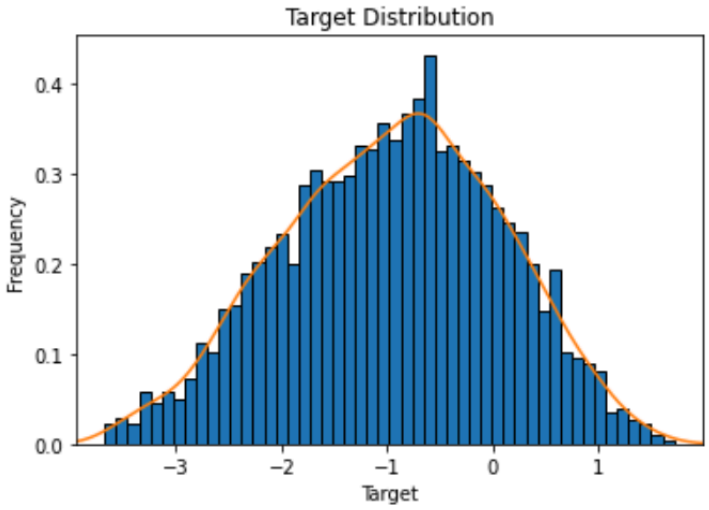
Statistical breakdown of 'Target' variable

```
target Variable
----------
Mean: -0.9596573929279933
Median: -0.91332221
Standard Deviation: 1.033604425872553
Minimum Value: -3.676267773
25th Percentile: -1.691501234
50th Percentile: -0.91332221
75th Percentile: -0.203078529
Maximum Value: 1.711389827
Skew: -0.13020966605362416
```

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 6

Distribution of 'target' variable



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

4.1.1 Standard Error

Alongside the ‘target’ variable, the standard error of the ‘target’ variable is visualized and analyzed in order to determine the accuracy of the mean value of each target variable in relation to the population mean value (Rose & Day, 1990). After running the code to visualize the distribution of the ‘standard_error’, the ‘standard_error’ variable had a minimum value of 0.428232657 and a maximum value of 0.649671297. Below is a visual representation of the statistics of the ‘standard_error’ variable. Shown in Figure 7, the 25th percentile, 50th percentile, and 75th percentile values of standard error do not vary heavily, indicating that the reading scores do not vary by much within the 25th percentile to the 75th percentile, indicating most excerpts in the data are of a similar readability score. where the value of the 25th percentile is 0.468552731 and the 75th percentile is 0.506303911.

Using the formula for the confidence interval (Doğan, 2017), $CI = \bar{x} \pm (z \times s/\sqrt{n})$, where CI is the confidence interval, $\bar{x}$ is the sample mean, s is the sample standard deviation, and n is the sample size, the lower and upper limits of the ‘target’ variable are determined. It is important to know these limits as they provide an indication of the range within which the true mean of the 'target' variable is likely to fall, thus quantifying the uncertainty associated with the estimation process.

Based on the known values of the ‘standard_error’ variable, the confidence interval is 0.490 at the lower limit and 0.493 at the higher limit. This means that there is a 95% confidence rate that the population mean of the ‘target’ variable is between -0.998 (lower limit) and -0.921 (upper limit).

Figure 7

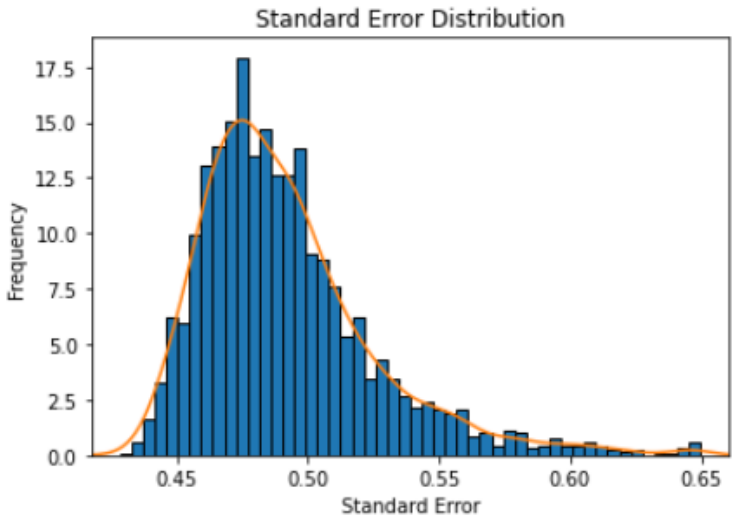
Statistical Breakdown of 'standard_error' variable

```
standard_error Variable
----------
Mean: 0.4916085590769504
Median: 0.484748101
Standard Deviation: 0.033576684547630714
Minimum Value: 0.428232657
25th Percentile: 0.468552731
50th Percentile: 0.484748101
75th Percentile: 0.506303911
Maximum Value: 0.649671297
Skew: -0.13020966605362416
```

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 8

Distribution of 'standard_error' variable



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

It is important to note how the distribution of the ‘target’ variable interacts with the attributes of the ‘excerpt’ and ‘excerpt_preprocessed’ variables in order to gain insight for the training of the regression models. Knowing how attributes such as word count, word length, etc. of an excerpt affect the ‘target’ variable will give an idea of the most and least important factors that affect the target score. This will in turn allow for a more in depth understanding of the nuances occurring within the ‘excerpt’ and ‘excerpt_preprocessed’ variables that affect the target scores.

4.2 ‘Excerpt’ And ‘Excerpt_Preprocessed’ Variable

The attributes of the ‘excerpt’ and ‘excerpt_preprocessed’ variable will be analyzed, discovering how they affect target scoring of the ‘target’ variable:

- Word Count = Total number of words in the excerpt
- Average Word Length = Average character length per word in excerpt
- Sentence Count = Total number of sentences in excerpt
- Character Length = Total number of characters in excerpt
- Maximum Sentence Length = Maximum character length of sentence in excerpt
- Minimum Sentence Length = Minimum character length of sentence in excerpt
- Average Sentence Length = Average character length of sentence in excerpt

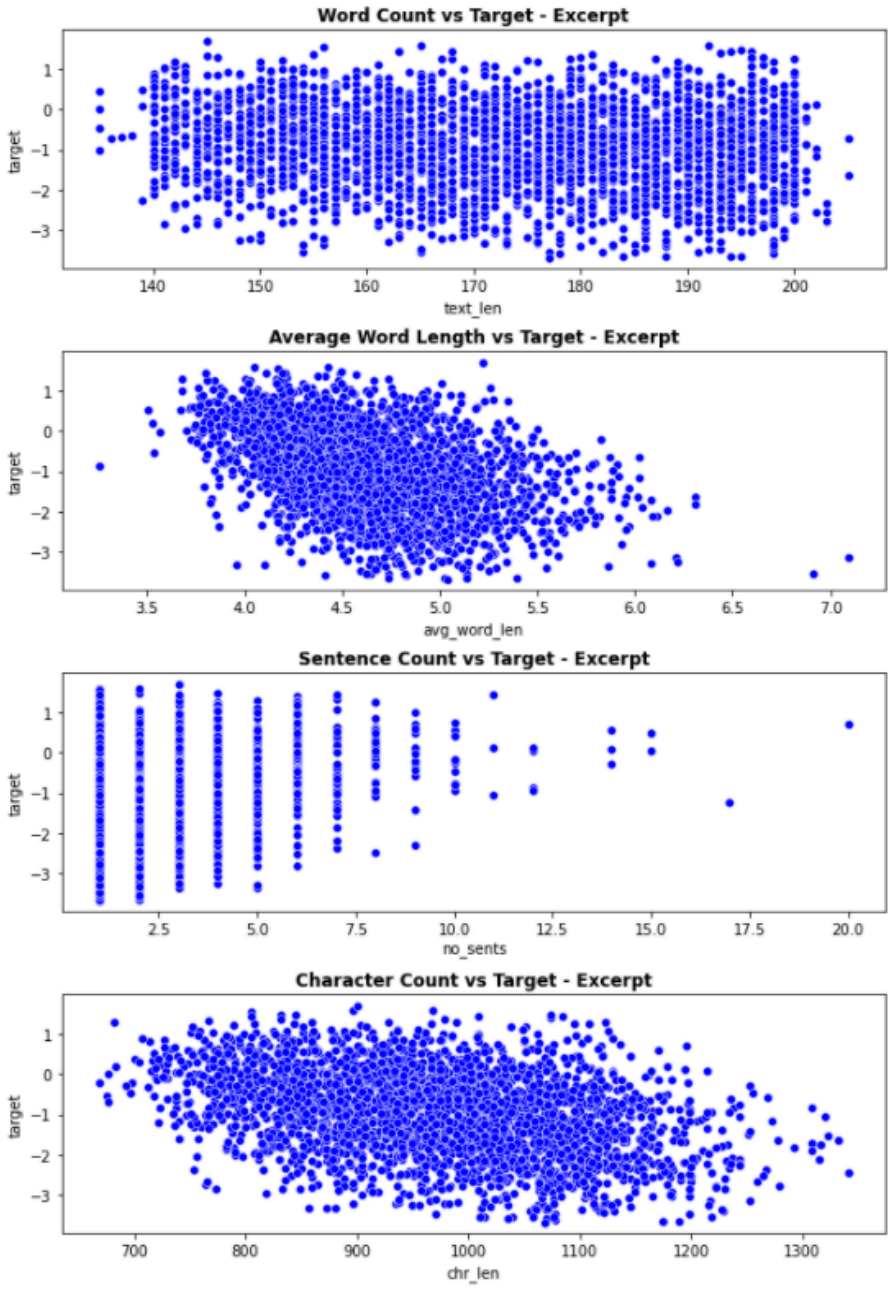
The above-mentioned attributes will be observed for the ‘excerpt’ variable in Figure 9 and Figure 10. The following attributes are observed in order to see how the following attributes affect target score. Of the above-mentioned attributes, the word count, average word length, sentence count, and character count for ‘excerpt_preprocessed’ are visualized in Figure 11. Only a limited number of attributes for ‘excerpt_preprocessed’ are observed because the preprocessing

for the ‘excerpt_preprocessed’ variable rendered the observations of other attributes involving sentence lengths unnecessary. Furthermore, we will observe the frequency of appearance in the top 20 words for both the ‘excerpt’ and ‘excerpt_preprocessed’ variable, in order to determine which words will most affect the regression model training. The distributions amongst the ‘excerpt’ and ‘excerpt_preprocessed’ variables and how they differ will indicate how the preprocessing affected the ‘excerpt_preprocessed’ in comparison to the ‘excerpt’ variable. Visualizing the word distribution will give us an idea of which words will affect the regression model training.

In conclusion, by analyzing attributes such as word count, average word length, sentence count, and character length of both the 'excerpt' and 'excerpt_preprocessed' variables, we gain insight into how these factors influence the target scoring of the 'target' variable. Additionally, observing the frequency of appearance in the top 20 words for both variables allows us to identify key words that may significantly impact regression model training. Furthermore, comparing the distributions of these variables highlights how preprocessing affects readability and other factors, ultimately contributing to our understanding of how text characteristics influence readability and regression model outcomes.

Figure 9

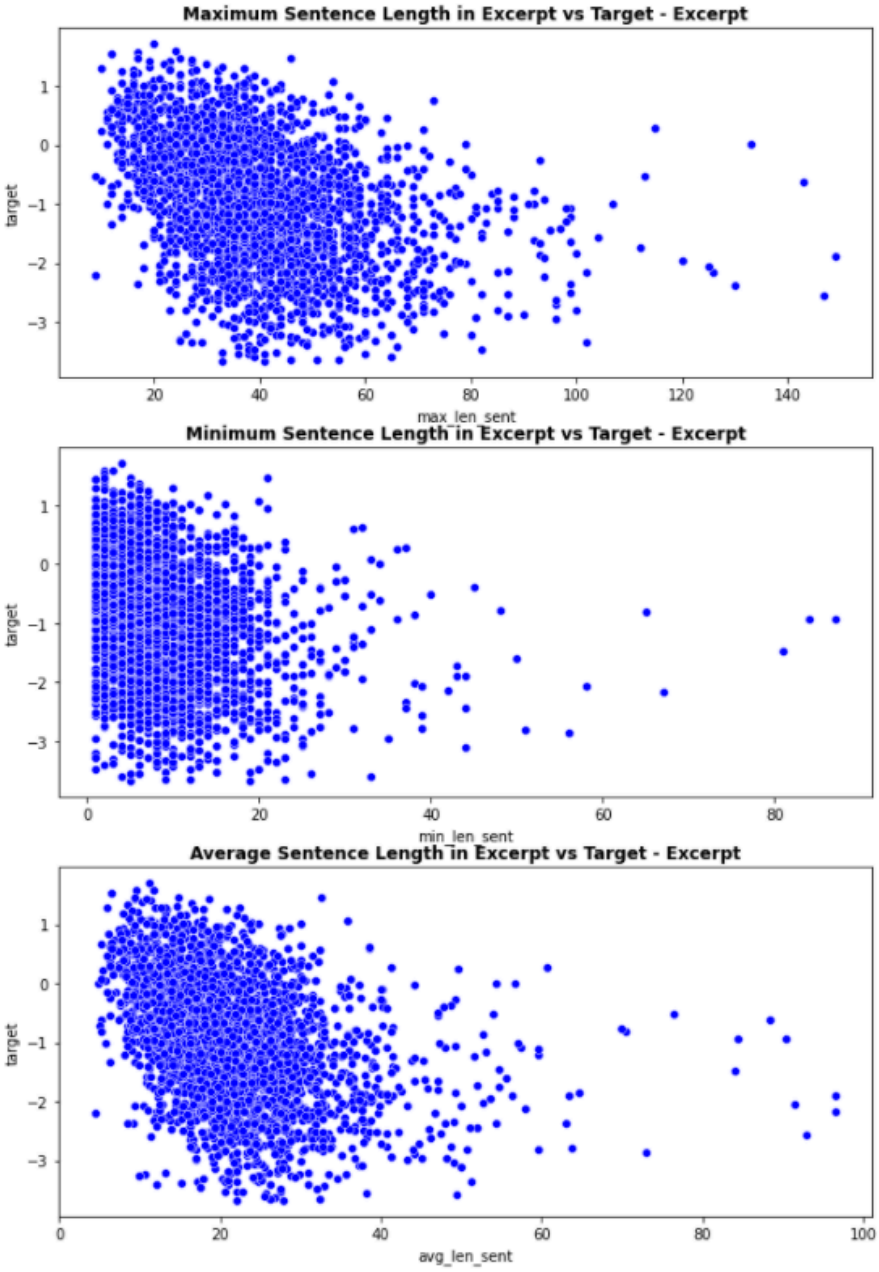
Scatterplot distribution of 'excerpt'



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 10

Scatterplot distribution of 'excerpt' (continued)



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Analysis of the ‘excerpt’ variable is focused mainly on Figure 9 and Figure 10. Figure 9 is a series of tables that is graphed for both the ‘excerpt’ variable and ‘excerpt_preprocessed’. Figure 10 is a series of tables that is graphed specifically for ‘excerpt’. Variable ‘excerpt_preprocessed’ cannot graph the series of tables in Figure 10 because the sentence count for ‘excerpt_preprocessed’ are all in one sentence, and the minimum, average, and maximum sentence lengths will all uniformly be the same value, making EDA (Aldera et al., 2021) unnecessary for sentence lengths.

Taking a look at ‘excerpt’ variable’s word count to the target score, the scatterplot distribution is observed to be uniform (Jerrum et al., 1986), which indicates that word count does not have much impact on whether a target score could be low or high. The average word length is observed to be a right skewed scatter plot distribution (Reimann et al., 2020). This right skewed distribution indicates that longer average word length in a passage of text denotes a lower target score.

Interestingly, as the sentence count increases, the more likely it is that the target score will be a higher value. Sentence count vs. target scatter plot is left skewed (Reimann et al., 2020). A left skew indicates that the reading complexity decreases as the quantity of sentences in the ‘excerpt’ variable passage increases. The more sentences per passage will likely indicate that the sentences in the passage tend to have shorter sentence lengths and have a decreased word count per sentence. Along with the average word count, the character count vs. target score scatter plot distribution is right skewed, indicating that the target score is likely to be lower as character count increases per passage. The lower target score indicates that the passage of text is more difficult to read.

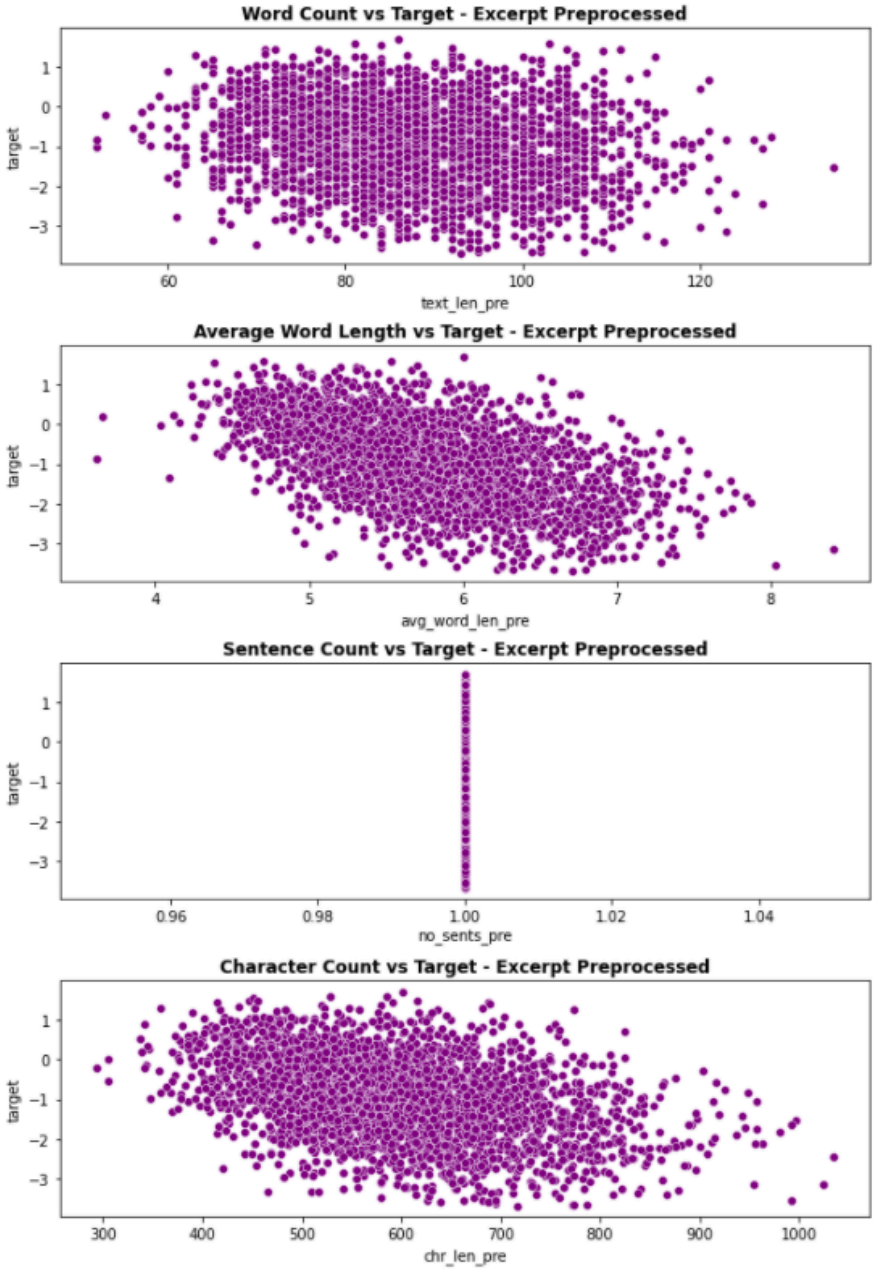
For further analyses of the ‘excerpt’ variable, the maximum sentence length, minimum sentence length, and average sentence length in Figure 10 are accounted for while exploring comparisons of aforementioned attributes within the ‘excerpt’ variable to target score. The sentence lengths are based on the character count per passage of text in each excerpt. The maximum and average sentence length scatter plot distributions (Reimann et al., 2020) are right skewed, meaning that the higher the number of characters in a maximum or average sentence will have a tendency to have lower target scores. The correlation between a higher number of characters in maximum or average sentences and lower target scores can be attributed to the likelihood that longer sentences contain more complex and potentially challenging language. Longer sentences may indicate denser content or more intricate ideas, which could make comprehension more difficult for readers. As a result, texts with longer sentences may tend to have lower target scores, as they may be perceived as more challenging to understand. The minimum sentence length starts off uniform (Jerrum et al., 1986), where a considerable portion of the data entries have a minimum sentence length below 20 characters long. As the minimum character length becomes longer than 20 characters, the scatter plot graphs become right skewed, leading to lower target scores. The correlation between longer minimum character lengths and lower target scores could be due to the fact that longer minimum character lengths may indicate more complex or verbose language. Texts with longer minimum character lengths might contain longer words, sentences, or phrases, which could make them more challenging to comprehend for readers. Consequently, such texts may tend to receive lower target scores as they may be perceived as more difficult to understand.

4.2.1 Excerpt Preprocessed

Analysis of the ‘excerpt_preprocessed’ variable is focused mainly on Figure 11. As previously mentioned, analysis of the ‘excerpt_preprocessed’ variable will be on the attributes of word count, average word length, sentence count, and character count in comparison to the target score in the ‘target’ variable. When observing the scatter plot distributions (Reimann et al., 2020) for both ‘excerpt’ and ‘excerpt_preprocessed’, it becomes apparent how the preprocessing of the ‘excerpt’ variable affects the target score based on the word length in a given passage of text.

Figure 11

Scatterplot distribution of 'excerpt_preprocessed'



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

An apparent observation to be made on Figure 11 would be the attribute sentence count and how all passages in ‘excerpt_preprocessed’ become one singular sentence, creating a linear scatter plot graph (Husson & Pagès, 2005). This is because all punctuations were removed for preprocessing purposes, leading to each excerpt becoming one sentence each. Due to the removal of stop words in preprocessing, word count and character count decrease due to removal of irrelevant text in each passage of text in the ‘excerpt’ to create ‘excerpt_preprocessed’.

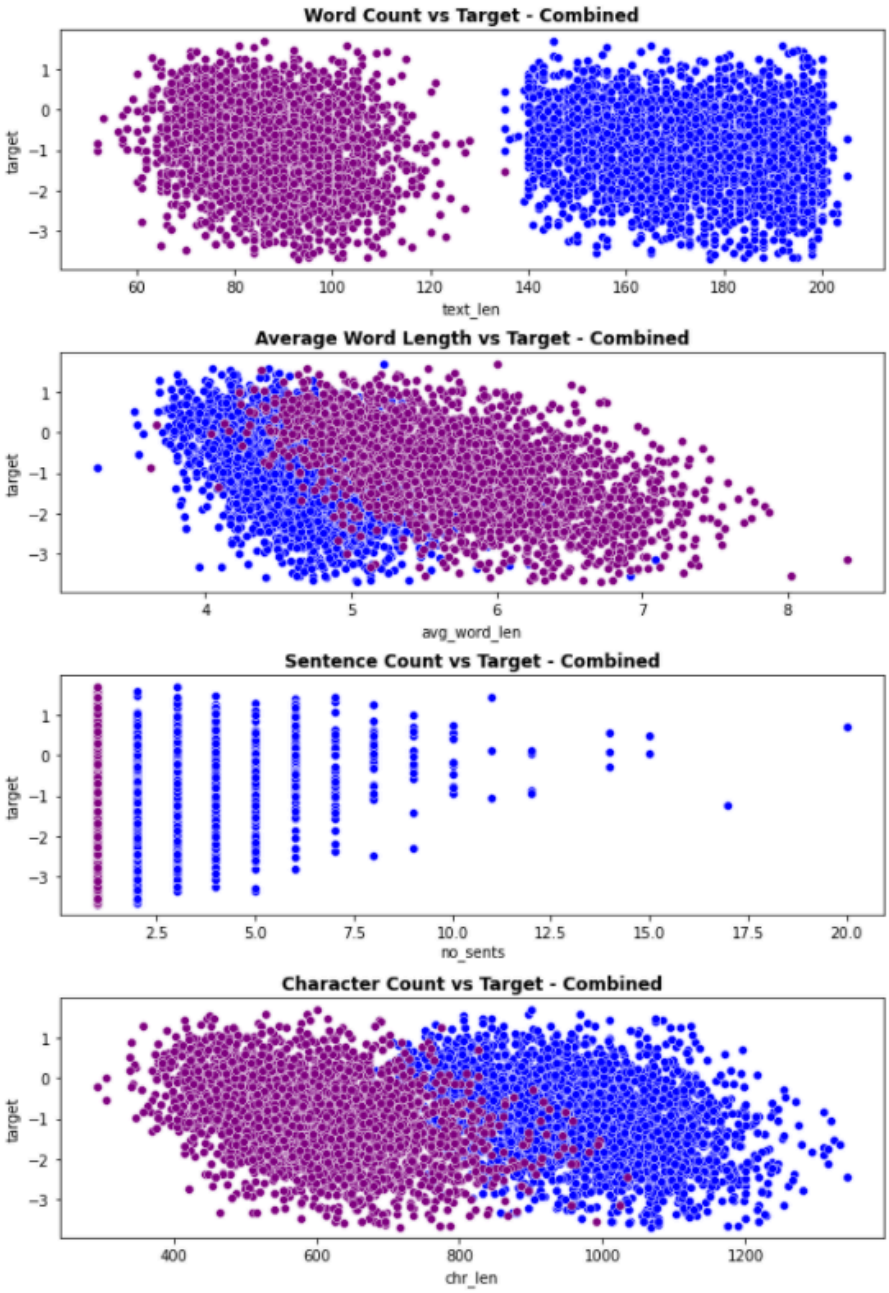
The scatter plot distribution for word count in ‘excerpt_preprocessed’ seems to be uniform (Jerrum et al., 1986), while the scatter plot distribution for character count seems to have a right skewed distribution (Reimann et al., 2020), meaning that the higher the character count could be indicative of a lower target score. The average word length for ‘excerpt_preprocessed’ is a skewed right scatter plot distribution, meaning that the longer the average word length is, the more indicative it is of a lower target score.

4.2.2 Comparing Excerpt Vs. Excerpt Preprocessed

We want to compare the excerpt and the preprocessed version of excerpt. Both ‘excerpt’ and ‘excerpt_preprocessed’ variables for each of the attributes for excerpt are graphed for comparison purposes. We want to know how each of the attributes affects the ‘target’ variable. Looking at the ‘target’ variable, we also want to know how preprocessing the data changes the target variable's distribution in a scatterplot as shown in Figure 12.

Figure 12

Scatterplot distribution comparison between 'excerpt' and 'excerpt_preprocessed'



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Based on the results of the comparison between the ‘excerpt’ variable and ‘excerpt_preprocessed’ variable, the following was found:

1. The average word and character count for the ‘excerpt_preprocessed’ variable was lower on average compared to the ‘excerpt’ variable. This is likely due to the removal of unnecessary words and characters done in the data wrangling (Sarkar & Roychowdhury, 2019) for the ‘preprocessed_excerpt’ variable.
2. The average word length for the ‘excerpt_preprocessed’ variable was on average, longer, compared to the ‘excerpt’ variable. It becomes apparent that word length affects the target score when observing the differences in word length between ‘excerpt’ and ‘excerpt_preprocessed’, as observed in Figure 12.
3. The sentence count for the ‘excerpt_preprocessed’ variable lowers to 1, since the only thing retained in the preprocessed excerpt is relevant text used for the training model.

The ‘excerpt_preprocessed’ variable is cleaned so that each excerpt is condensed into a singular sentence, which makes comparing analysis between sentence count vs. ‘target’ redundant when working with the ’excerpt_preprocessed’ variable. Therefore, there are no comparisons of sentence count vs ‘target’ amongst the ‘excerpt’ and ‘excerpt_preprocessed’ variables, and the analysis only pertains to the ‘excerpt’ variable.

4.2.3 Word Frequency

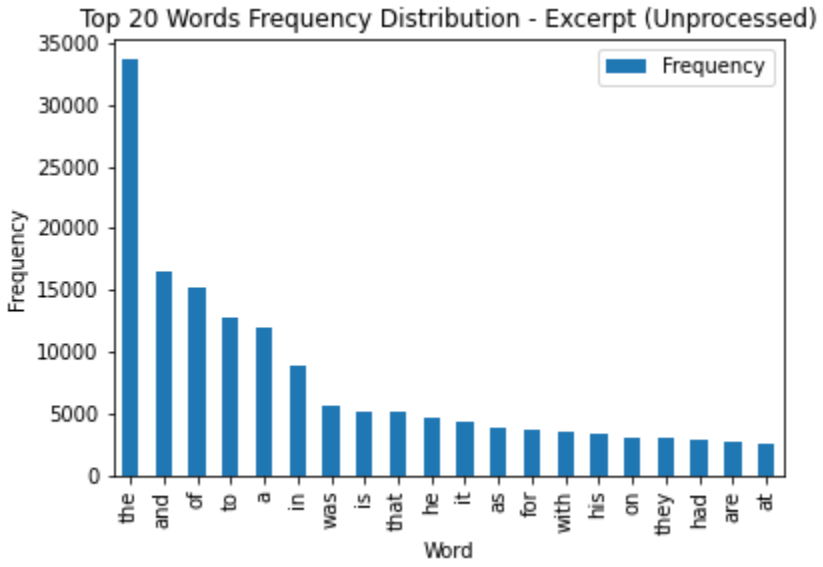
Analyzing the word frequency of ‘excerpt’ and ‘excerpt’ preprocessed variables aids in visualizing the characteristics and complexity of the texts being examined within the dataset. Taking note of the most common words that appear within each of the data entries helps determine how difficult each of the texts are within the dataset. If a text within the dataset

contains a higher frequency of common or simpler words, the text itself may be less complex. On the other hand, if the text within the dataset contained a higher frequency of technical or rare words, the text may be more complex in terms of its vocabulary.

Before comparing the ‘excerpt’ and ‘excerpt_preprocessed’ variables in terms of word frequency, it is important to visualize the ‘excerpt’ variable prior to cleaning in order to visualize the quantity of noise in the data. In Figure 13, it is shown that every word within the top 20 for word frequency are stop words. Therefore, for a deeper insight of the exploratory data analysis, the ‘excerpt’ variable will be cleaned with the exception of lemmatization (Plisson, Lavrac, & Mladenic, n.d.) in order to visualize which meaningful words impact the training of the regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010).

Figure 13

Top 20 word distribution (‘excerpt’) (unprocessed)



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

A variable ‘clean_text’, which is shown in Figure 14, is created for the purpose of cleaning both the ‘excerpt’ and ‘excerpt_preprocessed’ variables. Variable ‘clean_text’ removes non-alphabetic characters, converts all characters into lower case, tokenizes the list of words, and removes stop words. For both the ‘excerpt’ and ‘excerpt_preprocessed’ variable, the top 10 words remain the same with varying degrees of change in some words such as “time”, whereas there is a larger frequency of the word “time” in the ‘excerpt_preprocessed’ variable. It is possible for frequency of words to increase in the ‘excerpt_preprocessed’ variable due to lemmatization (Plisson et al., n.d.) normalizing words within the passages of texts. However, on average, word frequency across all passages of text dropped due to passages of text uniformly

dropping in word count from the ‘excerpt’ variable to the ‘excerpt_preprocessed’ variable due to the removal of stop words.

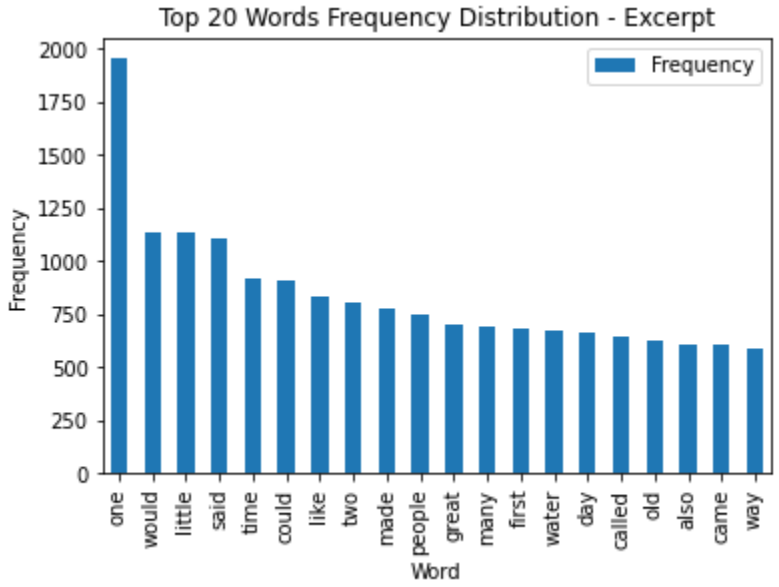
Figure 14

Example code for clean_text

```
def clean_text(text):
    text = re.sub("[^a-zA-Z]", " ", text)
    text = text.lower().split()
    return [word for word in text if word not in stopwords.words('english')]
```

Figure 15

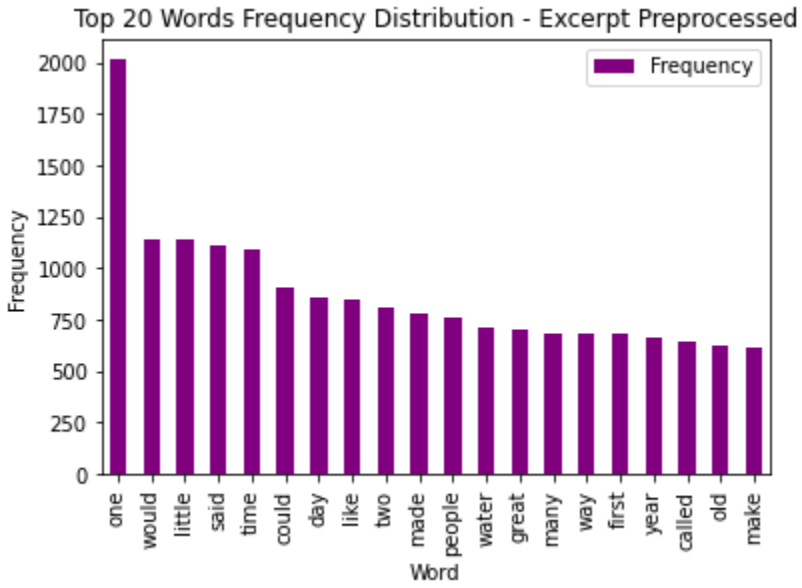
Top 20 word distribution (‘excerpt’)



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 16

Top 20 word distribution (‘excerpt_preprocessed’)



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

4.3 Conclusion For Exploratory Data Analysis

In conclusion, it is noted that the ‘target’ variable has a wide range, with a minimum value of -3.676267773, and a maximum value of 1.711389827. The value of the ‘target’ variable is -0.203078529 at the 75th percentile as indicated in Figure 5, indicating that over 75 percent of target scores had a negative target score.

Most of the scatter plot distribution (Reimann, Blech, & Gaschler, 2020) graphs for the ‘excerpt’ variable and the ‘excerpt_preprocessed’ variable in Figure 9, Figure 10, and Figure 11 were right skewed or uniform. Right skewed scatter plot distributions (Reimann, Blech, & Gaschler, 2020) indicated that certain attributes within ‘excerpt’ or ‘excerpt_preprocessed’ lead to lower target scores. Uniform scatter plot distributions (Jerrum, Valiant, & Vazirani, 1986)

meant a given attribute within the ‘excerpt’ or ‘excerpt_preprocessed’ did not have a high impact on target score. In Figure 12, it is observed that long average word length tended to have low target scores amongst both the ‘excerpt’ and ‘excerpt_preprocessed’ variables. This observation could be attributed to the fact that longer average word lengths may indicate more complex or technical language, which could potentially make the text more challenging to comprehend for readers. The word count and character count for ‘excerpt_preprocessed’ variable were uniformly lower than the ‘excerpt’ variable, while average word length was uniformly longer for ‘excerpt_preprocessed’.

Figure 13 showcases the amount of noise within the ‘excerpt’ variable. Based on the top 20 words frequency distribution that were run in Figure 15 and Figure 16, all words within the top 20 list ranged from words that are lengths three to six, with “one” being the most used word in the preprocessed excerpt. For both the ‘excerpt’ and ‘excerpt_preprocessed’ variables, most words had similar frequency amongst the top 20 words distribution.

The main insight acquired from the EDA regarding the regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010) is how the impact of the changes in the ‘excerpt_preprocessed’ variable in comparison to the ‘excerpt’ variable could aid in the accuracy of the regression models. The preprocessed excerpt has lower word and character count, which will aid in the accuracy of the regression models.

It is determined that standard setups for each of the regression models can be run when training the data based on the knowledge that the data is normalized (Singh & Singh, 2020) and there are no anomalies in the 'target' or 'excerpt_preprocessed' variables.

Chapter 5

Training And Results

In the previous chapter, insight was gathered about the ‘target’, ‘standard_error’, ‘excerpt’, and ‘excerpt_preprocessed’ variables. The ways in which ‘standard_error’ could potentially affect ‘target’ variable scoring were observed in the EDA. Different attributes of ‘excerpt’ and ‘excerpt_preprocessed’ were also explored (Aldera et al., 2021) for their effects on the ‘target’ variable scoring. All this insight is gathered to understand the relationships and potential influences of the 'target', 'standard_error', 'excerpt', and 'excerpt_preprocessed' variables. Through EDA, we observe how 'standard_error' could potentially impact the scoring of the 'target' variable, while also exploring various attributes of 'excerpt' and 'excerpt_preprocessed' (Aldera et al., 2021) to assess their effects on the scoring of the 'target' variable. Within the training of the data, the EDA will be investigated to see if it had any weight on the MSE (Hodson et al., 2021) results for each of the regression models (González-Garduño & Søgaard, 2017; Feng et al., 2010).

This chapter pertains to the set up of the training model and discusses the results of the training model for each of the regression models. TF-IDF vectorizer (Dey, Jenamani, & Thakkar, 2017) is used to convert each of the excerpts into a number that each of the regression models utilizes to train the model since regression models can only take in numerical values. TF-IDF vectorizer is the only variable that is used in training the data. The evaluation metric method (Kong, 2019) of MSE will be discussed in correspondence to the training of each regression model. Further perspective will be offered as to why the MSE was used and its significance to

the results offered during training. The MSEs for each regression model are graphed to visually represent the accuracy of each model. This graphical representation helps to provide an understanding of the predictability levels of each regression model.

Afterwards, the regression models are ranked based on descending MSE to indicate which regression model is most likely to yield the most efficiently accurate results amongst the seven regression models tested for this project. A lower MSE corresponds to more accurate results, as it indicates that the predicted values from the regression model are closer to the actual observed values. Likewise, the opposite will also hold true.

After the MSE is calculated and analyzed, the results of the target score for the excerpts within the test dataset for each regression model are loaded in. Each of the results are loaded into a .csv file. The target scores are then compared from each regression model in comparison to their MSE in order to try and find correlations as well as any anomalies within the target scores presented by each regression model.

5.1 Training Model Set Up

The data is being trained using the following regression models:

- Random Forest Regression
- Gradient Boosting Regression
- Support Vector Regression
- AdaBoost Regression
- XGBoost Regression
- Ridge Regression
- Linear Regression

When running each of the regression models, the training model takes the 'excerpt_preprocessed' variable as the input variable and will set the 'target' variable score as the output.The 'excerpt_preprocessed' variable becomes the independent variable and the 'target' variable becomes the dependent variable. Each of the 'excerpt_preprocessed' variables become numerical values due to TF-IDF vectorizer (Dey, Jenamani, & Thakkar, 2017), which creates numbers based on word frequency, which looks at the frequency of words within each of the excerpts individually and across each set of documents (amongst all excerpts), and creates a score between 0 and 1, where a score of 0 would indicate that the text contains many words that are common and a score of 1 would mean that the text contains many words that are infrequent. It is important to note that TF-IDF vectorizer is the only variable that is used to train the regression models, meaning that word frequency is the only aspect of the excerpts that is being trained in each of the models. TF-IDF looks at the rarity of words, where infrequent word usage would likely indicate that a text is more difficult. In the training model, word infrequency is the driver of whether or not a text is difficult. A 10 k-fold cross validation was also run on the dataset with the data randomized which yielded similar results to the initial test run on this dataset.

Random forest regression (Segal, 2004), gradient boosting regression (Mohan et al., 2011), support vector regression (Awad & Khanna, 2015), AdaBoost regression (Kummer & Najjaran, 2014), XGBoost regression (Chen & He, 2016), ridge regression (McDonald, 2009), and linear regression (Kumari, 2018) are run with their default settings for the training of the dataset. XGBoost regression was set up with the parameters set as 'reg:squarederror', *colsample_bytree* set to 0.3, *learning_rate* set to 0.1, *max_depth* set to 5, *alpha* set to 10, and

n_estimators set to 100. The parameter *colsample_bytree* indicates the number of features that are used within each tree. The parameter *colsample_bytree* is set to 0.3 to match the test size used in the training. Next, the parameter *learning_rate* indicates the amount that the weights are updated to during the training utilizing the XGBoost regression. Third, the parameter *max_depth* indicates the maximum depth of each tree. The parameter *alpha* indicates how accurately the XGBoost regression will converge during the training. How XGBoost regression converges during the training will indicate how the mean squared score will look, with a lower mean squared score indicating a more accurate result. Lastly, the parameter *n_estimators* indicates the number of trees in the ensemble. The n_estimators for XGBoost regression were tested at 10, 100, and 1000, where 10 yielded poor MSE results while 100 and 1000 produced similar MSE (Hodson et al., 2021) results. The n_estimators were kept at 100 to retain low runtime.

For each model, the MSE is run to determine accuracy and used a training size of 70% (1984 instances) and a test size of 30% (850 instances). The MSE is run on the test set. The closer the MSE score (Hodson et al., 2021) is to the value of 0, the more accurate the result of each regression is. MSE is used to determine the accuracy of the models since all the models run are regression models. Accuracy in this case will indicate which regression model was best at providing a target score based on the excerpts, which were pre processed into 'excerpt_preprocessed', given in the training model. The MSE indicates how close the regression model line is to the target scoring, which will give a clear indication of which regression model potentially yields the best accuracy based on the MSE. The MSE of each regression model is then graphed to visually represent the accuracy of each regression model. The graph will then indicate which of the regression models performed the best given the MSE score.

5.2 Comparing Regression Model Accuracy And Results

5.2.1 Mean Squared Error Rankings

The results of the MSE for each regression model are shown in Figure 17 and visualized in Figure 18.

Figure 17

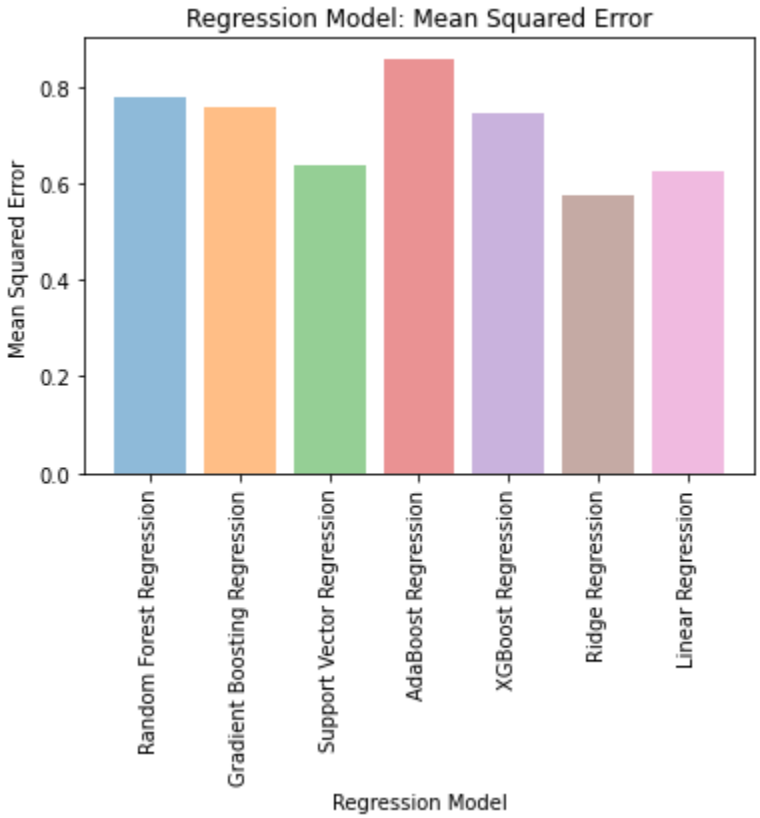
Mean squared error values of regression models

```
Model: Random Forest Regression
Mean Squared Error: 0.7780480810840206
Model: Gradient Boosting Regression
Mean Squared Error: 0.7589500331668939
Model: Support Vector Regression
Mean Squared Error: 0.6362544439858806
Model: AdaBoost Regression
Mean Squared Error: 0.8584656147226847
Model: XGBoost Regression
Mean Squared Error: 0.7460510416299314
Model: Ridge Regression
Mean Squared Error: 0.5745997882893428
Model: Linear Regression
Mean Squared Error: 0.6241485389666971
```

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Figure 18

Mean squared error of regression models bar chart



Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Based on the mean squared error, the regression models from best to worst are:

1) Ridge Regression
2) Linear Regression
3) Support Vector Regression
4) XGBoost Regression
5) Gradient Boosting Regression
6) Random Forest Regression
7) AdaBoost Regression

The training model did not take much time to compute with an average running speed of approximately 3 to 5 minutes.

5.2.2 Regression Model Output And Results

The tables below consist of all the results of the regression models for the outputted 'target' variable of the test sets' 7 excerpts. The 'excerpt' variable is the independent variable and the 'target' variable is the dependent variable.

Table 3

Random forest regression results

	id	target
0	c0f722661	-0.585648
1	f0953f0a5	-0.067364
2	0df072751	-0.446015
3	04caf4e0c	-1.729682
4	0e63f8bea	-1.696023
5	12537fe78	-0.299432
6	965e592c0	-0.285125

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 4

Gradient boosting regression results

:

	id	target
0	c0f722661	-1.148737
1	f0953f0a5	-0.134280
2	0df072751	-0.644293
3	04caf4e0c	-1.452009
4	0e63f8bea	-1.612390
5	12537fe78	-0.695474
6	965e592c0	0.030733

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 5

Support Vector Regression Results

	id	target
0	c0f722661	-1.161264
1	f0953f0a5	-0.500954
2	0df072751	-0.871783
3	04caf4e0c	-1.745278
4	0e63f8bea	-1.469826
5	12537fe78	-0.620321
6	965e592c0	-0.139143

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 6

AdaBoost regression results

	id	target
0	c0f722661	-1.427776
1	f0953f0a5	0.140642
2	0df072751	-1.427776
3	04caf4e0c	-1.523621
4	0e63f8bea	-1.486564
5	12537fe78	-1.294774
6	965e592c0	-0.233322

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 7

XGBoost regression results

	id	target
0	c0f722661	-0.824621
1	f0953f0a5	-0.431922
2	0df072751	-0.586349
3	04caf4e0c	-1.645461
4	0e63f8bea	-1.730322
5	12537fe78	-0.384243
6	965e592c0	0.322650

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 8

Ridge Regression Results

	id	target
0	c0f722661	-1.216477
1	f0953f0a5	-0.493634
2	0df072751	-0.817237
3	04caf4e0c	-1.947372
4	0e63f8bea	-1.512607
5	12537fe78	-0.526674
6	965e592c0	-0.036794

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Table 9

Linear Regression Results

	id	target
0	c0f722661	-1.324096
1	f0953f0a5	-0.615544
2	0df072751	-0.537346
3	04caf4e0c	-2.313807
4	0e63f8bea	-1.605459
5	12537fe78	-0.429681
6	965e592c0	-0.072484

Note. Data sourced from "CommonLit Readability Prize" by Malatinszky, Heintz, asiegel, Harris, Choi, Maggie, Culliton, & Crossley, 2021, Kaggle. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

Overall, the regression models had reasonable results, with the lowest mean squared error score being 0.5745997882893428 from the ridge regression model and the highest mean squared error score being 0.8362041138266413 from the AdaBoost regression model. These results are reasonable because the mean squared error (MSE) scores range between 0.5745997882893428 and 0.8362041138266413, indicating relatively low error rates in predicting the target variable across different regression models. The fact that the lowest MSE is from the ridge regression model suggests that this model performed particularly well in minimizing the discrepancy between predicted and actual values. While the highest MSE from the AdaBoost regression model indicates slightly higher error, it is still within a reasonable range, considering the complexity of the dataset and the variability inherent in real-world data. Overall, the results suggest that the regression models were effective in capturing the underlying patterns in the data, albeit with varying degrees of accuracy.

There could have been modifications to the data wrangling (Sarkar & Roychowdhury, 2019) of the preprocessed data or different natural language processing models could have been used to determine the scoring for ‘target’ variables. A low sample size likely does not offer enough information for us to come to a proper conclusion for which regression model is the best fit for determining a target score. Additionally, there was a lack of positive target score results, which would make it difficult to determine what excerpts from a different dataset could potentially be a positive target score based on the current results.

Based on the results given by the regression models for the ‘target’ variable, most excerpts were determined to be of a harder reading skill level due to having negative target scores.

Chapter 6

Further Developments And Applications

In this chapter, this project's future applications will be discussed, delving into different perspectives of the project that reside outside of its technical usage. Three main topics will be covered: ethical considerations and societal impact, further project possibilities, and development-related issues.

This chapter will explore the real-world applicability of the project beyond educational contexts. Additionally, we will examine the ethical implications of implementing a readability ease scoring formula, shedding light on potential issues in handling natural language processing projects (Omari & Mohammadian, 2016. By understanding these potential conflicts, we can discuss the project's social impacts (Berger & Packard, 2021) and its broader implications for society.

Next, open issues pertaining to the project will be discussed. First, certain aspects of the project need to be observed and questioned on whether there could potentially be beneficial changes made to the project structure. Topics such as the usage of different natural language processing methods to determine target score will be discussed amongst other open issues. Since the project only deals with 7 different data entries in the test dataset, potentially using a different test dataset with more data entries could be a consideration for future research.

Lastly, the current structure of the project will be discussed. There are a few changes that can be considered in the current iteration of this project. Keeping in mind that regression models

(González-Garduño & Søgaard, 2017; Feng et al., 2010; Romanov et al., 2019) were used to determine the ‘target’ variable’s score, questioning the use of different parameters in the regression models could become a cause of concern when determining changes to target score calculations, since changing the parameters could skew the results to misrepresent a given regression model. This misrepresentation could lead to inaccurate predictions and misinterpretations of the data, affecting the reliability and validity of the model's outcomes. Implementing other regression models into the current iteration of the project holds the potential to enhance accuracy compared to the models currently utilized. Further EDA could be applied to the ‘excerpt’ and ‘excerpt_preprocessed’ variable in order to gain deeper insights into their characteristics and potential influences on the target variable. Looking at different potential attributes of the ‘excerpt’ and ‘excerpt_preprocessed’ variables could yield potentially fruitful insight into the ‘target’ variable as there would be a more in depth understanding into how the ‘excerpt’ and ‘excerpt_preprocessed’ variables could affect the ‘target’ variable.

In this chapter, we will delve into the broader implications of the project, exploring its relevance beyond its technical aspects. We will examine its potential applications in real-world scenarios, consider the ethical considerations associated with its implementation, and discuss its societal impacts. Furthermore, we will address open issues regarding the project's structure and discuss potential enhancements to its methodology. Finally, we will explore the current structure of the project, highlighting areas for improvement and suggesting avenues for further investigation.

6.1 Towards An Implementation

The readability score within the ‘target’ variable that has been established within this project was created to move towards a more accessible readability ease scoring formula that may be implemented within various disciplines. For instance, when examining the academic disciplines that have become more accessible in modernity due to the popularization of college, the implementation of a readability score could potentially open up further avenues of higher and/or alternative routes to education by organizing public texts into appropriate reading grades based upon the readability score. This would essentially filter out the inaccessible “noise”, or inaccessible texts based upon an individual’s current reading comprehension, that a person in pursuit of knowledge would encounter.

In the scope of data science, this thesis project explored how regression models predicted a readability ease scoring formula, in this case being the ‘target’ variable. This is a lesser explored avenue of consideration in regards to predicting a readability ease scoring formula when compared to more well-tuned methodologies such as BERT. The contribution of this exploration in regression models towards data science fleshes out the possibilities regarding readability scores and may further aid in future pursuits regarding readability in data science.

6.2 Ethical Considerations And Impacts On Society

This project focuses on the development of a standardized readability ease scoring formula (Arslan, Tutar, & Kozanhan, 2020). In the scope of this project, the standardized readability ease scoring formula is mainly targeted towards determining readability ease for young children in classrooms from grades 3 to 12. The purpose of the standardized readability

ease scoring in the ‘target’ variable is to separate out passages of texts that are comprehensible, yet challenging enough for children during their critical years of intellectual development, further accelerating their reading capabilities. With this goal in mind, aiding in the development and acceleration of the future generation’s reading ability is a stepping stone to a potential issue stemming from the lack of accessibility to educational resources (Navarrete & Luján-Mora, 2018).

A standardized readability ease scoring formula that is easily accessible would have the potential to be globally implemented into educational infrastructures (Navarrete & Luján-Mora, 2018). This potential increase in educational prowess will aid in dropping criminal activity (Atems & Blankenau, 2021) as well as decrease the amount of people who are in poverty (Atems & Blankenau, 2021; Duarte et al., 2018). In a study by Machin et. al., the study states “Education can increase patience, which reduces the discount rate of future earnings and hence reduces the propensity to commit crimes. Education may also increase risk aversion that, in turn, increases the weight given by individuals to a possible punishment and consequently reduces the likelihood of committing crimes (Machin et al., 2010, p.5)”.

The quote from Machin et al. highlights the potential role of education in reducing criminal behavior through two mechanisms: increased patience and higher risk aversion. By increasing patience, education may lead individuals to consider the long-term consequences of their actions, reducing the impulsive behavior that can lead to criminal activity. Additionally, education may enhance risk aversion, causing individuals to place greater importance on avoiding potential punishments, thereby reducing the likelihood of engaging in criminal

behavior. This suggests that investing in education may have broader societal benefits beyond purely academic or economic outcomes.

Education aids in deductive reasoning (Carreira et al., 2020), a skill that hones an individual's ability to take information that they are taught and transfer the information into knowledge. Information is meant to inform someone, while knowledge is something that is known by someone. Information becomes knowledge when the information is retained. This knowledge is the product obtained through an understanding of the information processed through deductive reasoning. Using deductive reasoning (Carreira et al., 2020), individuals are able to make wiser decisions based on the information that they learn in order to rise out of poverty. When individuals are able to stay out of poverty, they will not need to resort to criminal activity in order to survive within the structures of society (Atems & Blankenau, 2021).

Increase in education also aids in the reduction of world population (Eryong & Xiuping, 2018). Currently, overpopulation is a global issue (Dow, 1966). There is a correlation between the amount of children a woman will have based upon the amount of education that they have completed, where women who are more educated are less likely to have a large number of children (Jiggins, 1994). The correlation between education and a reduction in world population can be attributed to several factors. Firstly, education empowers women with knowledge and skills, enabling them to make informed decisions about their reproductive health and family planning. Women who are more educated are more likely to have access to contraception and family planning services, allowing them to control the timing and number of children they have. Additionally, education often leads to greater economic opportunities for women, which may incentivize smaller family sizes as they prioritize career and personal aspirations. Moreover,

education tends to elevate the status of women in society, challenging traditional gender roles and norms that may promote larger family sizes. Overall, investing in women's education has significant implications for population control, as educated women are more likely to choose smaller family sizes, thereby contributing to the reduction of overpopulation on a global scale.

Alongside a reduction in world population, an increase in education will aid towards a decrease in disease and death (Pappas, 2017). Those who are educated are less prone to heart disease, diabetes, and obesity due to knowledge that was acquired through credible sources (Pappas, 2017; Scrivano et al., 2017). Education plays a pivotal role in mitigating health risks and reducing mortality rates. Educated individuals possess the knowledge and skills to make informed decisions regarding their well-being, thereby diminishing the prevalence of diseases such as heart disease, diabetes, and obesity. Through access to credible information sources, educated individuals can comprehend and apply preventive measures, fostering healthier lifestyle choices. Furthermore, education empowers individuals to critically assess health information, enabling them to discern reliable sources and make informed health-related decisions. Additionally, education often correlates with improved socioeconomic status, which affords better access to healthcare services and promotes healthier living environments. Thus, investing in education not only enhances individual health outcomes but also yields broader benefits for public health and population well-being.

Keeping the aforementioned issues in consideration, an accessible readability ease scoring formula, if implemented, can aid in educational infrastructures around the globe (Kruse et al., 2021). Easy access to education for developing future generations may help combat global issues such as criminal activity, poverty, overpopulation, and disease (Atems & Blankenau, 2021;

Dow, 1966; Pappas, 2017). By equipping individuals with knowledge and skills, education empowers them to make informed decisions, adopt healthier lifestyles, and contribute positively to society, ultimately fostering healthier communities and reducing the burden of social and health-related challenges.

Looking back, the development of an accessible reading ease scoring formula is widely viewed as ethically commendable, as its primary aim is to enhance educational effectiveness in classrooms worldwide (Navarrete & Luján-Mora, 2018). However, its usefulness extends beyond education alone, finding applications across various disciplines. Take, for instance, the Flesch-Kincaid index, a similar reading ease scoring formula widely utilized in search engine optimization (SEO) strategies (Anuyah et al., 2020), among other contexts. Furthermore, these formulas are valuable in professional domains such as technical writing, legal documentation, and medical communication, where clarity and accessibility are essential for effective communication (Anuyah et al., 2020). By promoting clearer communication and understanding across different fields, accessible reading ease scoring formulas play a vital role in improving accessibility and inclusivity in information dissemination, enriching learning experiences, and fostering broader societal engagement with complex subject matter.

Suppose there is a program that aids in the creation of SEO searchable terms and phrases, helping to create headlines or titles that are more likely to show up in the recommended search results. The program mentioned uses a reading ease scoring formula (Anuyah et al., 2020) which is implemented into the SEO program to generate a readability score to determine whether or not certain SEO phrasing is appropriate for search engine usage. Ideally, an individual or company could use the SEO program to generate titles that are optimal for search engine purposes.

However, a big company becomes affiliated with the SEO program and tweaks the readability scoring formula within the SEO program to output unfavorable reading scores for competitors while using the non-tweaked SEO program to generate favorable reading scores for their own usage (Ziewitz, 2019). This particular example could be an example of how reading scores can be ethically abused in a brand competition scenario.

Search engines themselves could manipulate search results based on the reading score of a search (Ziewitz, 2019), where certain reading score brackets see certain results show up in a search result, creating a bias towards information which could pertain to medical knowledge, politics, and other pressing matters. Politicians could manipulate search engines into changing search results for certain demographics (i.e. ethnicity, race, gender), which would fall under a form of gerrymandering. Search results and reading material that may or may not contain credible information becomes potentially dangerous for the common man, as doing research into pressing matters could form a skewed bias within an individual. An increase in education globally could aid in helping individuals form educated decisions based on proof or logic, creating a less skewed bias on topics such as medicine, politics, etc. the more informed an individual becomes. Manipulating bias in groups of individuals based on search engine results and reading ease is unethical. This is due to the potential threat that manipulations hold, creating a chance in which a misinformed point of view in daily topics such as racial bias, stereotyping, and more.

6.3 Open Issues

Looking at the current structure of the project, there are a few aspects that could be worked on in a future iteration of this project. The project's training and test datasets are skewed, where there are 2834 data entries in the training set and 7 data entries in the test set. These two datasets would be combined into one and split the data 70/30 for more consistent test results. Only having 7 data entries in the test set provides too little information to have fruitful insight into how the training model trained each of the excerpts.

Secondly, in the training model, only one variable, the TF-IDF vectorizer (Dey, Jenamani, & Thakkar, 2017), is used to train each of the regression models. In a future iteration of the project, different aspects of the excerpts should also be utilized, such as word count, average word length, sentence count, or character length. This creates more depth in the training model and could potentially provide a more accurate MSE score amongst all regression models since there would be more in-depth training, rather than only looking at word frequency.

Other methods used to determine reading score such as the Flesch-Kincaid Grade Level scoring (Grabeel et al., 2018) can be calculated based on the excerpt that we must determine the grade level of reading that each of the excerpts are and compare them to the target score of each of the excerpts. Alongside the 'excerpt' variable, we could extend the other methods of grading of reading scores to the 'excerpt_preprocessed' variable and compare the differences between each of the different grading methods and the target score. EDA (Aldera et al., 2021) allows us to visually examine the variations in grading methods, providing insights into their effectiveness and helping us identify the best methods in conjunction with the target score. Through visual exploration of the data, we can discern patterns, trends, and relationships, which aids in

understanding the strengths and weaknesses of different grading approaches and informs decision-making regarding their suitability for predicting the target score.

Chapter 7

Related Work

This thesis incorporates natural language processing with regression model analysis, text analysis, and reading comprehension. Related work regarding the analysis of text and reading comprehension are described. Additionally, it is important to explore different NLP techniques in regard to what is possible when constructing future work for the project.

7.1 Natural Language Processing Versus Rule Based Text Analysis

The following related work discusses another NLP technique, known as the Bidirectional Encoder Representation from Transformers (BERT), and its relation to predicting readability. BERT is an option that goes in a separate direction that was possible during the early-stages of this data science project. BERT produces potentially superior results by having a process that takes text and converts it into numbers specifically meant for text analysis, compared to that of regression models and other NLP tools (Chan et al., 2021; Devlin et al., 2018; Gabriel et al., 2019; Yoshimura et al., 2019; Zhang et al., 2019) which normally need some other process outside of their machine learning model to produce results. As such, BERT is an effective tool in predicting readability levels of English and Chinese passages (Chan et al., 2021; Deutsch et al., 2020; Tseng et al., 2019) . BERT takes into account semantic meaning or context of a word while other methods such as TF-IDF does not account for semantic meaning or context of a word, for weighting numbers differently than other NLP techniques. A study that utilizes BERT was done by Chan et al. that measures the readability by using the Flesch Reading Ease test (Chan et al.,

2021; Grabeel et al., 2018). The Flesch Reading Ease test creates a reading score which uses a formula known as Flesch Reading Ease Readability (Grabeel et al., 2018) to calculate reading ease. The BERT score is calculated for each sentence, where a lower average BERT score indicates higher grammatical quality and fluidity, which then represents a higher quality of writing per passage.

Chan et al. uses a dataset consisting of Kickstarter projects. In fact, "Kickstarter is a popular reward-based crowdfunding platform for new ventures to raise small monetary contributions from a large number of individuals online in exchange for tangible and/or social rewards (Chan et al., 2021, p.4)". The dependent variable for the study done by Chan et al. consists of "the total pledged amount a campaign had received as a proxy of crowdfunding outcomes (Chan et al., 2021, p.5)". The independent variable in the study by Chan et al. consisted of BERT, which was utilized to generate a NLP-based readability score (Chan et al., 2021; Devlin et al., 2019). Chan et al. changed BERT's default masking strategy by masking one word at a time to predict the probability of a given word appearing in a set position in context of the rest of the sentence (Chan et al., 2021).

Given the results of the study conducted by Chan et al., it was deemed that BERT and Flesch Reading Ease Readability (Grabeel et al., 2018) were not highly correlated. Based on the results of the study conducted by Chan et al. , Higher BERT scores denoted a lower quality of writing, which could potentially increase funding within Kickstarter projects because it seems that lower quality of writing was more generally understood by the general consensus (Chan et al., 2021). Chan et al. concludes that "the average BERT score on determining crowdfunding outcomes (in Kickstarter projects) rise above traditional rule-based readability scores (such as the

Flesch Kincaid Reading Ease Readability scores) (Chan et al., 2021, p.10)". This indicates that predicting crowdfunding outcomes is much more successful with the BERT score method in comparison to standard readability score models such as Flesch Kincaid.

7.2 Machine Learning Usage To Predict Reading Comprehension

The following related work by Sinclair et al. (Sinclair et al., 2021) analyzes different machine learning methodologies and how they associate with reading comprehension. The study conducted by Sinclair et al. notes "with regard to language and literacy education—cornerstones of the educational enterprise—(machine learning) applications have focused primarily on assessment of writing (Burstein, Tetreault, & Madnani, 2013), oral reading fluency (Black et al., 2009; Bock & Aitkin, 1981;Evanini et al., 2015), and spontaneous speech (Sinclair et al., 2021, p.2)." While machine learning does have a focus on certain aspects of speech and writing, application of machine learning to reading comprehension has not been explored in machine learning.

Although machine learning has made significant strides in areas like speech recognition and natural language processing, its application to reading comprehension has been relatively overlooked within the machine learning field. The complexity of reading comprehension tasks, which require not only understanding individual words and sentences but also synthesizing information to derive meaning from a passage, presents a formidable challenge for machine learning algorithms. This lack of exploration may be due to the intricate nature of reading comprehension, which demands a deeper understanding of linguistic nuances and cognitive processes. While machine learning shows promise in addressing reading comprehension

challenges, further interdisciplinary efforts are necessary to fully harness its potential in this domain, drawing insights from fields such as cognitive science and linguistics. Thus, while the application of machine learning to reading comprehension remains relatively uncharted territory, it holds great promise for advancing our understanding of language comprehension and interpretation.

Sinclair et al. worked with two data sets, one which contained text-elicited responses and shaone that contained transcribed oral-elicited responses. Sinclair et al. utilizes natural language processing (Chan et al., 2021; Sha, 2018; Balyan et al., 2020; Romanov et al., 2019; Wang & Hu, 2021) for its data preprocessing, using Core Variable Feature Extraction Feature Extractor (COVFEFE) NLP package (Komeili et al., 2019) to preprocess raw text files. After the data is preprocessed using COVFEFE, Sinclair et al. (Sinclair et al., 2021) inspected the data for data cleaning, cleaning the oral elicited data into a single interpretable dataset (Sinclair et al., 2021), normalizing the data to prevent high dimensionality (Lee & Jemain, 2021). The work done by Sinclair et al. to analyze the datasets were done in "Python (Version 3.6.7) using scikit-learn version 20.3 (Pedregosa et al., 2011) with an iterative process of cross-validation (Sinclair et al., 2021, p.58)". Five different machine learning models were trained for this study, each built with the scikit-learn function ShuffleSplit (Little et al., 2017). ShuffleSplit operates differently from k-fold cross-validation (Little et al., 2017), allowing for splits within the input data for a set amount of times for training by 75% and testing by 25% respectively.

Sinclair et al. utilizes mean absolute error (Sammut & Webb, 2011; Willmott & Matsuura, 2005) as their evaluation metric for their machine learning models, which takes the average of all the individual prediction errors over all instances in the data set (Sammut & Webb,

2011). Mean absolute error differs from MSE in how they measure the average magnitude of error within a data set. Mean absolute error takes the average over the test sample to obtain a 'total error' (Willmott & Matsuura, 2005), then dividing the total error by *n*, where *n* is the number of instances within a dataset. In contrast, MSE takes a different approach, computed by summing the squares of individual errors (Willmott & Matsuura, 2005). This calculation method emphasizes larger errors more prominently due to the squaring operation, thereby making MSE more sensitive to outliers or extreme values present in the dataset. Consequently, as the dataset's variance increases, MSE tends to produce higher error scores compared to the mean absolute error. This characteristic of MSE is valuable in scenarios where it is essential to give greater weight to larger errors, offering insights into the overall variability and dispersion of errors within the model predictions.

The five regressions utilized in the study conducted by Sinclair et al. consisted of gradient boosting regression, random forest regression, support vector regression, linear support vector regression, and linear regression. Linear support vector regression (Klopfenstein & Vaiter, 2021) is similar to support vector regression, however implemented differently, utilizing liblinear from the scikit learn library rather than utilizing libsvm. Noting the similarities in utilization of regression models, the calculation of the mean absolute error is a core difference in how speech and writing is predicted for reading comprehension in comparison to that of MSE.

Chapter 8

Conclusion

The main objective of this data science project was to determine which regression model would best predict the ‘target’ variable for a target score which indicates the readability of excerpts. First, the training and test data are read in, with the training data containing 2834 data entries while the test set contains 7 data entries. After the training and test data was read in, EDA was performed on the ‘target’, ‘excerpt’, and ‘excerpt_preprocessed’ variables in order to determine how the regression models would be trained.

Seven distinct regression models were employed in the training process to identify the most effective predictor of 'target' variable scores. Only the TF-IDF vectorizer was utilized as a variable for training, focusing solely on word frequency within the excerpts for each model. The accuracy of each of the regression models was determined by using the MSE formula. After each of the regression models were trained, based on the MSE values, it was determined that Ridge Regression had best predicted the scores for the ‘target’ variable. Overall, each regression model had reasonably accurate results with the exception of AdaBoost regression.

In future application, more variables could be used to train the regression models. Since there were only 7 samples in the test set, there is a chance that the results of the regression models are skewed. Therefore, using a larger sample in the test set could potentially lead to more accurate MSE scores. Being able to relate the ‘target’ variable to other scores such as the Flesch-Kincaid Grade Level scoring in the EDA may have been able to provide deeper insight into how the ‘target’ variable is calculated.

References

Aldera, S., Emam, A., Al-Qurishi, M., Alrubaian, M., & Alothaim, A. (2021). Exploratory data analysis and classification of a new Arabic online extremism dataset. *IEEE Access, 9*, 161613–161626. https://doi.org/10.1109/ACCESS.2021.3132651

Álvarez, A., & Ritchey, T. (2015). Applications of General Morphological Analysis, *vol. 4*, no. 1.

Anuyah, O., Milton, A., Green, M., & Pera, M. S. (2020). An empirical analysis of search engines' response to web search queries associated with the classroom setting. *Aslib Journal of Information Management*, *72*(1), 88–111. doi: http://dx.doi.org/10.1108/AJIM-06-2019-0143.

Arora, S., Hazan, E., & Kale, S. (2012). The multiplicative weights update method: A meta-algorithm and applications. *Theory of Computing, 8*(1), 121–164. https://doi.org/10.4086/toc.2012.v008a006

Arslan, D., Tutar, M. S., & Kozanhan, B. (2020). Evaluating the readability, understandability, and quality of online materials about chest pain in children. *European Journal of Pediatrics, 179*(12), 1881–1891. https://doi.org/10.1007/s00431-020-03772-8.

Atems, B., & Blankenau, W. (2021). The 'time-release', crime-reducing effects of education spending. *Economics Letters, 209*, 110143. https://doi.org/10.1016/j.econlet.2021.110143.

Atmazaki, A., Ermanto, E., & Putri, D. A. (2018). Development of CTL-Based Reading

Materials. *Proceedings of the International Conference on Language, Literature, and Education (ICLLE 2018)*.

Awad, M., & Khanna, R. (2015). Support vector regression. In *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers* (pp. 67–80). Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4302-5990-9_4.

Ayinde, A. Q., Adetunji, A. B., Bello, M., & Odeniyi, O. A. (2013). Performance evaluation of naive bayes and decision stump algorithms in mining students' educational data. *International Journal of Computer Science Issues (IJCSI)*, *10*(4), 147-151. Retrieved from https://www.proquest.com/docview/1471054666?sourcetype=Scholarly%20Journals

Balyan, R., McCarthy, K. S., & McNamara, D. S. (2020). Applying natural language processing and hierarchical machine learning approaches to text difficulty classification. *International Journal of Artificial Intelligence in Education, 30*(3), 337–370. https://doi.org/10.1007/s40593-020-00201-7.

Berger, J., & Packard, G. (2021). Using natural language processing to understand people and culture. *American Psychologist*. https://doi.org/10.1037/amp0000882.

Black, M., Tepperman, J., Lee, S., & Narayanan, S. S. (2009). Predicting children's reading ability using evaluator-informed features. *Tenth Annual Conference of the International Speech Communication Association*. Signal Analysis and Interpretation Laboratory, University of Southern California.

Breland, H. M. (1996). Word Frequency and Word Difficulty: A Comparison of Counts in Four Corpora. *Psychological Science*, *vol. 7*, no. 2, pp. 96–99.

Briskilal, J., & Subalalitha, C. N. (2022). An ensemble model for classifying idioms and literal texts using BERT and RoBERTa. *Information Processing & Management, 59*(1), 102756. https://doi.org/10.1016/j.ipm.2021.102756.

Burstein, J., Tetreault, J., & Madnani, N. (2013). The e-rater automated essay scoring system. In M. D. Shermis & J. Burstein (Eds.), *Handbook of Automated Essay Scoring: Current Applications and Future Directions* (pp. 55–67). Routledge.

Carreira, S., Amado, N., & Jacinto, H. (2020). Venues for analytical reasoning problems: How children produce deductive reasoning. *Education Sciences, 10*(6), 169. https://doi.org/10.3390/educsci10060169.

Chan, C. S. R., Pethe, C., & Skiena, S. (2021). Natural language processing versus rule-based text analysis: Comparing BERT score and readability indices to predict crowdfunding outcomes. *Journal of Business Venturing Insights, 16*, e00276. https://doi.org/10.1016/j.jbvi.2021.e00276.

Chau, M., Li, T. M. H., Wong, P. W. C., Xu, J. J., Yip, P. S. F., & Chen, H. (2020). Finding people with emotional distress in online social media: A design combining machine learning and rule-based classification. *MIS Quarterly, 44*(2), 933–956. https://doi.org/10.25300/MISQ/2020/14110.

Chen, T., & He, T. (2016). xgboost: eXtreme Gradient Boosting (p. 4).

Cortes, C., Mohri, M., & Rostamizadeh, A. (2009). L2 Regularization for Learning Kernels.

Deutsch, T., Jasbi, M., & Shieber, S. (2020). *arXiv preprint arXiv:2006.00377*.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *arXiv preprint arXiv:1810.04805*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805 [cs]*. Accessed: Jan. 27, 2022. [Online]. Available: http://arxiv.org/abs/1810.04805

Dey, A., Jenamani, M., & Thakkar, J. J. (2017). Lexical TF-IDF: An n-gram Feature Space for Cross-Domain Classification of Sentiment Reviews. In *Pattern Recognition and Machine Intelligence* (pp. 380–386). Cham. https://doi.org/10.1007/978-3-319-69900-4_48.

Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. In *Multiple Classifier Systems* (pp. 1–15). Berlin, Heidelberg. https://doi.org/10.1007/3-540-45014-9_1.

Doğan, C. D. (2017). Applying bootstrap resampling to compute confidence intervals for various statistics with R. *Eurasian Journal of Educational Research, 17*(68), 1–18. https://doi.org/10.14689/ejer.2017.68.1.

Dow, T. E. (1966). Overpopulation: Dilemma for U.S. aid. *Current History*, *51*(300), 65–72.

Duarte, R., Ferrando-Latorre, S., & Molina, J. A. (2018). How to escape poverty through education?: Intergenerational evidence in Spain. *Applied Economics Letters*, *25*(9), 624–627. doi: 10.1080/13504851.2017.1352073.

Eke, C. I., Norman, A. A., & Shuib, L. (2021). Multi-feature fusion framework for sarcasm identification on Twitter data: A machine learning based approach. *PLoS ONE*, *16*(6), 1–32. doi: 10.1371/journal.pone.0252918.

Eryong, X., & Xiuping, Z. (2018). Education and anti-poverty: Policy theory and strategy of poverty alleviation through education in China. *Educational Philosophy & Theory*, *50*(12), 1101–1112. doi: 10.1080/00131857.2018.1438889.

Evanini, K., Heilman, M., Wang, X., & Blanchard, D. (2015). Automated scoring for the TOEFL Junior® Comprehensive Writing and Speaking Test. *ETS Research Report Series*, *2015*(1), 1–11. doi: 10.1002/ets2.12052.

Farrar, D. E., & Glauber, R. R. (1967). Multicollinearity in Regression Analysis: The Problem Revisited. *The Review of Economics and Statistics*, *vol. 49*, no. 1, pp. 92–107. doi: 10.2307/1937887.

Feng, L., Jansche, M., Huenerfauth, M., & Elhadad, N. (2010). A comparison of features for automatic readability assessment (p. 9).

Gabriel, S., Bosselut, A., Holtzman, A., Lo, K., Celikyilmaz, A., & Choi, Y. (2019). *arXiv preprint arXiv:1907.01272*.

Glymour, C. (1998). What went wrong? Reflections on science by observation and the bell curve. *Philosophy of Science*, *65*(1), 1–32.

González-Garduño, A. V., & Søgaard, A. (2017). Using gaze to predict text readability. In

Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications (pp. 438–443). doi: 10.18653/v1/W17-5050.

Grabeel, K. L., Russomanno, J., Oelschlegel, S., Tester, E., & Heidel, R. E. (2018). Computerized versus hand-scored health literacy tools: A comparison of Simple Measure of Gobbledygook (SMOG) and Flesch-Kincaid in printed patient education materials. *Journal of the Medical Library Association*, *106*(1), 38–45. doi: 10.5195/jmla.2018.262.

Hickman, L., Thapa, S., Tay, L., Cao, M., & Srinivasan, P. (2022). Text preprocessing for text mining in organizational research: Review and recommendations. Organizational Research Methods, 25(1), 114–146. https://doi.org/10.1177/1094428120971683

Hilt, D. E., & Seegrist, D. W. (1977). Ridge, a computer program for calculating ridge regression estimates. Department of Agriculture, Forest Service, Northeastern Forest Experiment Station. Retrieved from https://ia803007.us.archive.org/23/items/ridgecomputerpro236hilt/ridgecomputerpro236hilt.pdf

Ho, T. K. (1995). Random decision forests. In Proceedings of the 3rd International Conference on Document Analysis and Recognition (Vol. 1, pp. 278–282). Montreal, QC, Canada. https://doi.org/10.1109/ICDAR.1995.598994

Hodson, T. O., Over, T. M., & Foks, S. S. (2021). Mean squared error, deconstructed. *Journal of Advances in Modeling Earth Systems*, *13*(12), 1–10. doi: 10.1029/2021MS002681.

Husson, F., & Pagès, J. (2005). Scatter plot and additional variables. *Journal of Applied*

Statistics, *32*(4), 341–350. doi: 10.1080/02664760500079043.

Jerrum, M. R., Valiant, L. G., & Vazirani, V. V. (1986). Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, *43*, 169–188. doi: 10.1016/0304-3975(86)90174-X.

Jiggins, J. (1994). *Changing the Boundaries: Women-Centered Perspectives On Population And The Environment*. Island Press.

Kannan, D. S., & Gurusamy, V. (n.d.). Preprocessing Techniques for Text Mining.

Killada, P. (2017). Data Analytics Using Regression Models for Health Insurance Market Place Data (pp. 1–61). Retrieved from https://etd.ohiolink.edu/acprod/odb_etd/ws/send_file/send?accession=toledo1501721348961437&disposition=inline

Klopfenstein, Q., & Vaiter, S. (2021). Linear support vector regression with linear constraints. *Mach Learn*, *vol. 110*, no. 7, pp. 1939–1974. doi: 10.1007/s10994-021-06018-2.

Komeili, M., Pou-Prom, C., Liaqat, D., Fraser, K. C., Yancheva, M., & Rudzicz, F. (2019). Talk2Me: Automated linguistic data collection for personal assessment. *PLOS ONE*, *14*(3), e0212342. doi: 10.1371/journal.pone.0212342.

Kong, F. (2019). Development of metric method and framework model of integrated complexity evaluations of production process for ergonomics workstations. *International Journal of Production Research*, *57*(8), 2429–2445. doi: 10.1080/00207543.2018.1519266.

Kruse, J., et al. (2021). Readability, content, and quality of COVID-19 patient education materials from academic medical centers in the United States. *American Journal of Infection Control*, *49*(6), 690–693. doi: 10.1016/j.ajic.2020.11.023.

Kumari. (2018). Linear regression analysis study. Retrieved from https://www.j-pcs.org/article.asp?issn=2395-5414;year=2018;volume=4;issue=1;spage=33;epage=36;aulast=Kumari.

Kummer, N., & Najjaran, H. (2014). Adaboost.MRT: Boosting regression for multivariate estimation. *Artificial Intelligence Research*, *3*. doi: 10.5430/air.v3n4p64.

Kurdi, M. Z. (2020). Text Complexity Classification Based on Linguistic Information: Application to Intelligent Tutoring of ESL. *Journal of Data Mining & Digital Humanities*, *2020*, 6012. https://doi.org/10.46298/jdmdh.6012.

Lee, L. C., & Jemain, A. A. (2021). An overview of PCA application strategy in processing high dimensionality forensic data. *Microchemical Journal*, *vol. 169*, p. 106608. doi: 10.1016/j.microc.2021.106608.

Little, M. A., et al. (2017). Using and understanding cross-validation strategies. *GigaScience*, *vol. 6*, no. 5. doi: 10.1093/gigascience/gix020.

Liu, Y., et al. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692 [cs]*. Accessed: Jan. 27, 2022. [Online]. Available: http://arxiv.org/abs/1907.11692

Machin, S. J., et al. (2010). The Crime Reducing Effect of Education. *SSRN Electronic Journal*. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1631135

Malatinszky, A., Heintz, A., asiegel, Harris, H., Choi, J. S., Maggie, Culliton, P., & Crossley, S. (2021). CommonLit Readability Prize. *Kaggle*. Retrieved from https://kaggle.com/competitions/commonlitreadabilityprize

McDonald, G. C. (2009). Ridge regression. *WIREs Computational Statistics*, *1*(1), 93–100. doi: 10.1002/wics.14.

Mohan, A., Chen, Z., & Weinberger, K. (2011). Web-search ranking with initialized gradient boosted regression trees. *Proceedings of the Learning to Rank Challenge*, 77–89. Retrieved from https://proceedings.mlr.press/v14/mohan11a.html

Mohan, V. (2015). Preprocessing Techniques for Text Mining - An Overview. Retrieved from https://www.researchgate.net/publication/339529230_Preprocessing_Techniques_for_Text_Mining_-_An_Overview

Morgenthaler, S. (2009). Exploratory data analysis. *WIREs Computational Statistics*, *vol. 1*, no. 1, pp. 33–44. doi: 10.1002/wics.2.

Nargesian, F., Samulowitz, H., Khurana, U., Khalil, E. B., & Turaga, D. (2017). Learning feature engineering for classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, Melbourne, Australia (pp. 2529–2535). doi: 10.24963/ijcai.2017/352.

Navarrete, R., & Luján-Mora, S. (2018). Bridging the accessibility gap in Open Educational Resources. *Universal Access in the Information Society*, *17*(4), 755–774. doi: 10.1007/s10209-017-0529-9.

Newson, A. J., & Wrigley, A. (2017). Is mitochondrial donation germ-line gene therapy? Classifications and ethical implications. *Bioethics*, *31*(1), 55–67. doi: 10.1111/bioe.12312.

Omari, R. M., & Mohammadian, M. (2016). Rule-based fuzzy cognitive maps and natural language processing in machine ethics. *Journal of Information, Communication & Ethics in Society*, *14*(3), 231–253. doi: http://dx.doi.org/10.1108/JICES-10-2015-0034.

Pappas, A. (2017). *The Relationship Between Healthcare and Education and Their Impact on Global Health*. La Salle University.

Paszke, A., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, *vol. 32*. Accessed: Jan. 27, 2022. [Online]. Retrieved from https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *vol. 12*, 2825–2830.

Plisson, J., Lavrac, N., & Mladenic, D. (n.d.). A rule-based approach to word lemmatization (p. 4).

Reimann, D., Blech, C., & Gaschler, R. (2020). Visual model fit estimation in scatterplots and distribution of attention: Influence of slope and noise level. *Experimental Psychology*, *67*(5), 292–302. doi: 10.1027/1618-3169/a000499.

Romanov, A., Lomotin, K., & Kozlova, E. (2019). Application of natural language processing algorithms to the task of automatic classification of Russian scientific texts. *Data Science Journal*, *18*(1), Article 1. doi: 10.5334/dsj-2019-037.

Rose, G., & Day, S. (1990). The population mean predicts the number of deviant individuals. *BMJ*, *301*(6759), 1031–1034. doi: 10.1136/bmj.301.6759.1031.

Sammut, C., & Webb, G. I. (Eds.) (2011). Mean Absolute Error. In *Encyclopedia of Machine Learning*. Springer, Boston, MA. Retrieved from https://link.springer.com/referenceworkentry/10.1007/978-0-387-30164-8_525

Sarkar, T., & Roychowdhury, S. (2019). *Data wrangling with Python: Creating actionable data from raw sources*. Birmingham; Mumbai: Packt Publishing Ltd.

Scrivano, R. M., Scisco, J. L., & Giumetti, G. W. (2017). The impact of applicants' weight and education about obesity on applicant ratings. *Psi Chi Journal of Psychological Research*, *22*(4), 278–285. doi: 10.24839/2325-7342.JN22.4.278.

Segal, M. R. (2004). Machine learning benchmarks and random forest regression. Retrieved from https://escholarship.org/uc/item/35x3v9t4

Sha, H. (2018). *CS 229 Project Report: Text Complexity (Natural Language)*.

Sinclair, J., Jang, E. E., & Rudzicz, F. (2021). Using machine learning to predict children's reading comprehension from linguistic features extracted from speech and writing. *Journal of Educational Psychology*, *113*(6), 1088–1106. doi: 10.1037/edu0000658.

Singh, D., & Singh, B. (2020). Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, *97*, 105524. doi: 10.1016/j.asoc.2019.105524.

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. *Behav Res*, *47*(2), 340–354. https://doi.org/10.3758/s13428-014-0459-x.

Tseng, H. C., Chen, H. C., Chang, K. E., Sung, Y. T., & Chen, B. (2019). International Conference on Innovative Technologies and Learning, 301–308.

Wang, M., & Hu, F. (2021). The application of NLTK library for Python natural language processing in corpus research. *Theory and Practice in Language Studies*, *11*(9), 1041–1049. doi: http://dx.doi.org/10.17507/tpls.1109.09.

Willmott, C., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.*, *vol. 30*, 79–82. doi: 10.3354/cr030079.

Yoshimura, R., Shimanaka, H., Matsumura, Y., Yamagishi, H., & Komachi, M. (2019). Proceedings of the Fourth Conference on Machine Translation, *2*, 521–525. Shared Task Papers, Day 1.

Zamanian, M., & Heydari, P. (2012). Readability of Texts: State of the Art. *TPLS*, *vol. 2*, no. 1, pp. 43–53. doi: 10.4304/tpls.2.1.43-53.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). *arXiv preprint arXiv:1904.09675*.

Ziewitz, M. (2019). Rethinking gaming: The ethical work of optimization in web search engines. *Social Studies of Science*, *49*(5), 707–731. doi: 10.1177/0306312719865607.