COLUMBUS STATE UNIVERSITY
TSYS School of Computer Science

THE GRADUATE PROGRAM IN APPLIED COMPUTER SCIENCE

AUTOMATED ESSAY EVALUATION USING NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

A THESIS SUBMITTED
IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

BY
HARSHANTHI GHANTA

COLUMBUS, GEORGIA
2019

AUTOMATED ESSAY EVALUATION USING NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

By

Harshanthi Ghanta

Committee Chair:

Dr. Shamim Khan

Committee Members:

Dr. Rania Hodhod

Dr. Hyrum D. Carroll

ABSTRACT

The goal of automated essay evaluation is to assign grades to essays and provide feedback using computers. Automated evaluation is increasingly being used in classrooms and online exams. The aim of this project is to develop machine learning models for performing automated essay scoring and evaluate their performance. In this research, a publicly available essay data set was used to train and test the efficacy of the adopted techniques. Natural language processing techniques were used to extract features from essays in the dataset. Three different existing machine learning algorithms were used on the chosen dataset. The data was divided into two parts: training data and testing data. The inter-rater reliability and performance of these models were compared with each other and with human graders. Among the three machine learning models, the random forest performed the best in terms of agreement with human scorers as it achieved the lowest mean absolute error for the test dataset.

Keywords: Automated essay evaluation, machine learning, natural language processing, feature extraction

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor Dr. Shamim Khan for extensive guidance and support of my research. I would like to thank the thesis committee members Dr. Rania Hodhod and Dr. Hyrum D. Carroll. Finally, I must express my profound gratitude towards my parents and friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

CHAPTER 1 Introduction

Manually evaluating essays is time-consuming. Also, human graders can unintentionally be biased while grading (Zupanc & Bosnić, 2018). It can lead to inefficient grading and inconsistent feedback. On the other hand, if we choose an unbiased training dataset the automated system for essay scoring can avoid these limitations (Bolukbasi, T, 2016). As a result, the development and application of automated essay evaluation systems are growing.

Figure 1 (Hearst, M, 2000) shows how writing evaluation systems have evolved over the decades. This timeline is not comprehensive. This figure was based on the research and development at Educational Testing Services.

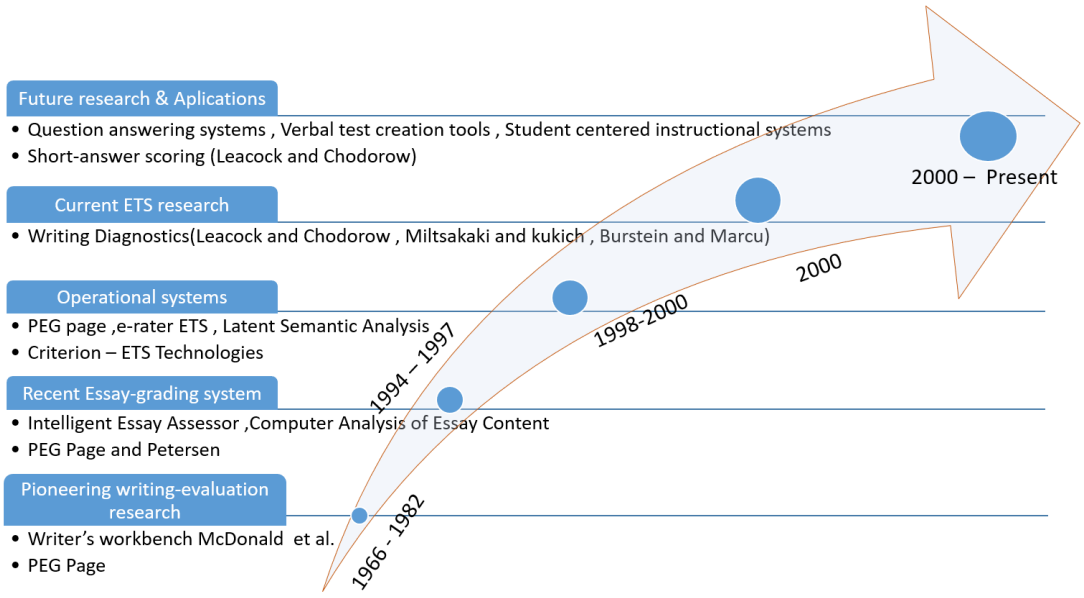

Figure 1. A timeline of research developments in writing evaluation. Based on debate of automated essay grading (Hearst, M, 2000).

Automated essay scoring is the process of evaluating essays by computers where grading models are learned using essay datasets scored by different human graders

(Shankar et al., 2018). It is a method of educational assessment and an application of natural language processing (NLP). Several factors that contribute to the growing interest in automated essay scoring, such as cost, accountability, standards, and technology. Rising education costs have led to pressure to hold the educational system accountable for results by imposing standards. The advance of information technology promises to measure educational achievement at a reduced cost. A major component of secondary education is learning to write effectively, a skill that is bolstered by repeated practice with formative guidance. However, providing focused feedback to every student on multiple drafts of each essay throughout the school year is a challenge for even the most dedicated teachers (Dronen, Foltz, & Habermehl, 2014). Automated essay scoring can enable students to practice by taking tests and write essays over and over to improve the quality of their answers.

English proficiency tests such as GRE and TOEFL use the e-rater (Writing evaluation) automated writing evaluation engine. The produced scores in these tests are the combined average of the automated score and a human grader score. Some of the features used in the e-rater engine relate to writing quality are an error in grammar, usage, mechanics, style, discourse structure, sentence variety, source use, and discourse coherence quality (Shankar, Ravibabu, 2018).

Other features used to evaluate the essays are lexical diversity, sentence count, word frequency, word count, average length, structure, and organization of an essay. These features are used for achieving accuracy in grading (Shankar & Ravibabu, 2018). The following are features used in many reported types of research.

Lexical diversity

Lexical diversity (LD) refers to the variety of words used in a text; LD indices generally measure the number of unique words occurring in the text in all instances of words by tokens. When the number of word types equal to the total number of tokens, all the words are different (McNamara, Crossley, & Roscoe, 2012).

Word frequency

Word frequency refers to how often a word occurs in the English language and is an important indicator of lexical knowledge. The presence of more uncommon words in a text suggests that the writer possesses a large vocabulary (McNamara, Crossley, & Roscoe, 2012).

Syntactic complexity

Sentences that contain a higher number of words before the main verb, a higher number of high-level constituents (sentences and embedded sentence constituents) per word in the sentence, and more modifiers per noun phrase are more syntactically complex and more difficult to process and comprehend (Perfetti, Landi, & Oakhill, 2005).

Syntactic similarity

Syntactic similarity refers to the uniformity and consistency of syntactic constructions in the text at the clause, phrase, and word level. More uniform syntactic constructions result in less complex syntax that is easier for the reader to process (Crossley, Greenfield, & McNamara, 2008). The feature extraction tool Coh-Metrix (Metrix) calculates the mean level of consistency of syntax at different levels of the text (McNamara, Crossley, & Roscoe, 2012).

Lexical overlap

Lexical overlap refers to the extent to which words and phrases overlap across sentences and text, thus making a text more cohesive and facilitating text comprehension (Kintsch & van Dijk, 1978). Coh-Metrix considers four forms of lexical overlap between sentences: noun overlap, argument overlap, stem overlap, and content word overlap (McNamara, Crossley, & Roscoe, 2012).

Semantic overlap

Semantic overlap refers to the extent to which phrases overlap semantically across sentences and text. Coh-Metrix measures semantic overlap using LSA, a mathematical and statistical technique for representing deeper world knowledge based on large corpora of texts. LSA cosines represent the semantic similarity between the words in sentences and paragraphs, an important indicator of cohesion (Landauer, McNamara, Dennis, & Kintsch, 2007).

In this research, features like readability, lexical diversity, word frequency, syntactic similarity, lexical overlap and semantic overlap were used to predict essay scores. Table 1 consists of 35 features that were extracted using Coh-Metrix and were used in the project:

Table 1 Thirty-five features extracted using the Coh-Metrix readability library

Readability grades	**Sentence information**	**Word usage**	**Sentence begins with**
1)Kincaid *2) ARI* *3) ColemanLiau* *4) FleschReadingEase* *5) GunningFogIndex* *6) LIX* *7) SMOGIndex* *8) RIX* *9) DaleChallIndex*	*1) characters_per_word* *2) syll_per_word* *3) words_per_sentence* *4)sentences_per_paragraph* *5) type_token_ratio* *6) characters* *7) syllables* *8)words* *9) wordtypes* *10) sentences* *11) paragraphs* *12) long_words* *13) complex_words* *14) complex_words_dc*	*1) tobeverb* *2) auxverb* *3) conjunction* *4) pronoun* *5) preposition* *6) nominalization*	*1) pronoun* *2) interrogative* *3) article* *4) subordination* *4) conjunction* *5) preposition*

1.1 Project Motivation and Goal

The reason for the lack of reliability in some of the automated essay scoring is that they use very basic features like word count, paragraph count, and sentence length. This causes automated essay scoring systems to focus more on the size and the structure of the essay rather than the content and quality of the essay. One positive development in the field of automated essay evaluation is the growing amount of data available to work with, which makes machine learning an attractive option to solve this problem. However, a grading model must learn from data that represents a noisy relationship between essay attributes and its grade (Zupanc & Bosnić, 2018).

The goal of this project is to combine quantitative features with essay content to improve the reliability of the automated essay scoring. Figure 2 gives a general outline of

the flow of the project.

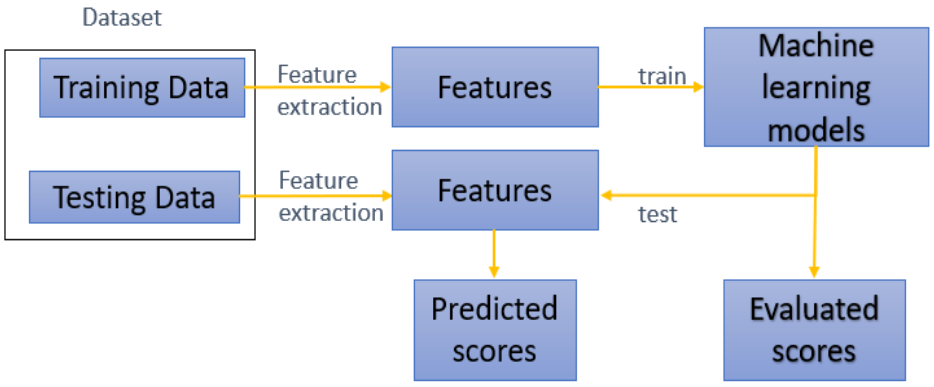

Figure 2 Process flow of the machine learning models

1.2 Organization of the Thesis

The rest of the work is organized as follows: Chapter 2 discusses the related works. Chapter 3 introduces the methodology and datasets used. Chapter 4 shows the experimental results and discussion. Finally, Chapter 5 provides the conclusion and possible future work.

CHAPTER 2 Related Works

The research on writing evaluation and implementation has started many decades ago and continued for more advanced automated evaluation systems. The debate article by (Hearst, 2000) presents the research work on the essay grading or writing evaluation. It explains the evolution of the automated evaluation tools from PEG Writer's workbench to Short-answer scoring systems developed in the time range 1960 to 2000. Some of the operational automated evaluation systems developed by 2000 were PEG, e-rater, Latent Analysis, Criterion.

(Burstein, Kukich, Wolfe & Chodorow, 1998) built an electronic essay rater that uses features discourse marking, syntactic information, and topical content. In their paper, they compared two content vectors to predict scores, essay content, and essay argument content. Electronic essay rater obtained an average of 82% accuracy between argument content scores and human raters, 69% when compared between essay content and human raters. Including the discourse marking feature, e-rater attained 87%-94% agreement with human raters across 15 sets of essay responses.

Shankar et al. (2018) discussed how it is hectic for graders to provide feedback with stable interface, mindset and time bounds. In their paper, they have used automated essay evaluation using "features like bag of words, sentence, and word count along with their average length, structure, and organization of an essay to achieve maximum accuracy in grading." They have also used a sequential forward feature selection algorithm to compare accuracy between different features to select the best subset of

features in order to evaluate the essays. This approach has succeeded with small datasets; however, they need improvement in correcting grammatical errors.

(Crossley et al., 2016) discusses how to obtain the quality of essay automatically by combining the NLP and machine learning approaches that assess the text features and by assessing the individual differences in writers by collecting the information from standardized test scores and survey results. Based on the lexical properties of student essays, they are predicting the vocabulary scores by using indices related to surface-level and discourse-level of student's essays. ReaderBench (Dascula et al, 2014) is an open-source framework used as an automated text analysis tool that calculates indices related to *linguistic* and *rhetorical* features of the text. It was tested on 108 university students' essays and obtained 32.4% variance in vocabulary scores, with multiple paragraph essays the model obtained improved scores

Crossley et al. (2016) aim to assess the individual difference among the students by calculating the lexical properties of their essays. They have tried correlation and regression analyses which revealed that indices with length and diversity of words in the essays combined to account between 32% and 56% of variances. If a model consists of three or more paragraphs. In this paper when an essay contains three or more paragraphs, the vocabulary knowledge and comprehension skills are better characterized by their model. (Crossley et al., 2016) had considered a larger number of ReaderBench indices that tap into discourse-level information. Required further research as the results are preliminary. Further research should specifically test these assumptions and consider developing separate *stealth assessments* for single and multiple paragraph essays. In the paper, they had considered the vocabulary knowledge and compressive skills whereas,

further it can consider a wide variety of factors related to writing such as students' attitudes and self-efficacy towards writing, their motivation level on a day, or their level creativity. Instead, more variable factors, such as daily motivation, maybe better captured by analyses that focus on changes in students' writing (i.e., a comparison of their writing on a day to their style and quality of writing more generally), rather than on properties of individual texts. To delivery personalized instructions and feedback for student users, NLP techniques are used by the researchers and system developers to build stealth assessments. The paper has utilized NLP framework RederBench to investigate the efficacy of NLP techniques to inform stealth assessment of vocabulary knowledge. The model has succeeded in obtaining individual differences among student writers. Overall, the results showed increased accuracy in automatically essay scores by combining both approaches.

Neural network models have been used for automated essay scoring. For example, Fei et al. (2017) used recurrent and convolution neural networks to model input essays, giving grades based on a single vector representation of the essay. They built a hierarchical sentence-document model using attention mechanisms to automatically decide the relative weights of words and sentences to score different parts of the essay. The attention mechanism outperformed the previous state-of-the-art methods.

(Woods et al., 2017) explains how learning to write effectively is important in secondary education. Which lead to development of automated essay scoring. (Woods et al., 2017) had considered ordinal essay scoring model to generate feedback based on rubric using the predictive realistic essay variants. A similar method was used in Revision Assistant, an educational product that provides rubric-specific, sentence-level feedback to

students. The model performed adequately while preserving characteristics fit for a novel sentence influence evaluation task.

Writing MentorTM is an add-on designed to provide feedback to struggling writers to help them improve their writing uses NLP techniques and resources to generate feedback and features span from many writing sub-constructs. It was used to obtain positive results from users in terms of usability and potential impact on their writing (Madnani, Burstein, Elliot, Klebanov, Napolitano, Andreyev, & Schwartz, 2018).

CHAPTER 3 Methodology

3.1 Introduction

In this project to automate the essay evaluation, machine learning is used.

"Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed." by Arthur Samuel, 1959

- Machine learning involves learning from data to solve problems that are difficult to solve by conventional programs. Figure 4 shows the traditional approach to problem solving and Figure 5 shows the machine learning approach to problem solving.

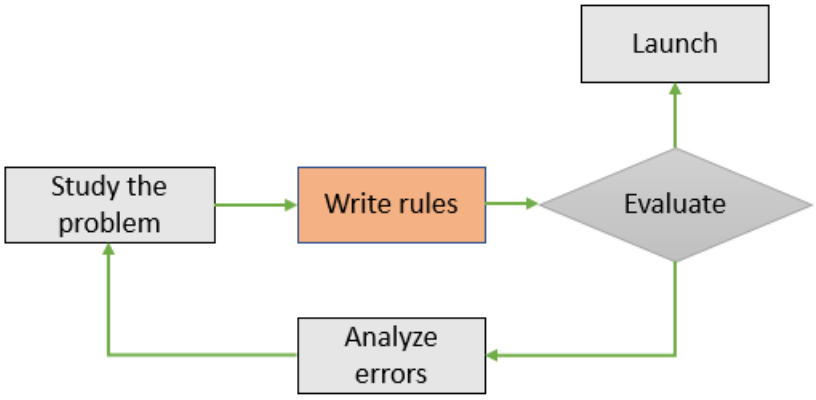

Figure 4 The traditional approach to problem solving.

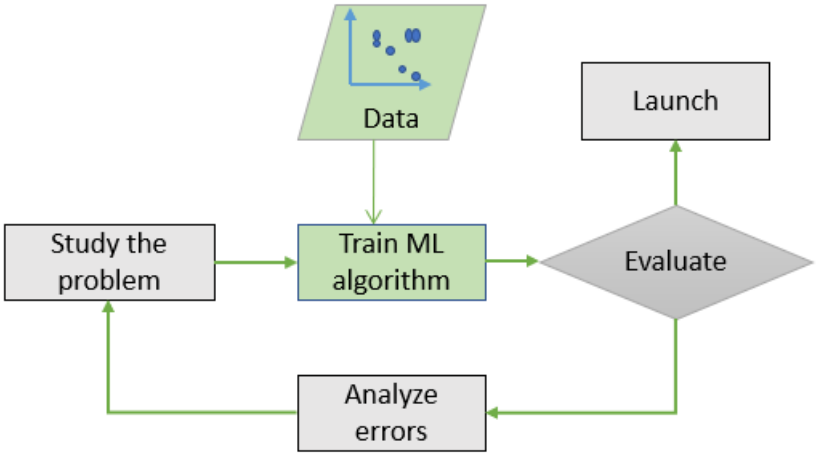

Figure 5 The machine learning approach to problem solving.

Machine learning applications are being used in a variety of fields, some examples are spam filtering, financial fraud detection, voice command systems, NLP like Google translate, self-driving vehicles.

In this project supervised learning model is used, as the training algorithm monitors the performance of the model for each date item used as input and adjusts the model's parameters based on how accurate the prediction is. In this project we are predicting a score, so regression algorithms are used.

To compute the model the Python programming language is used. Python is widely used for machine learning because of the availability of a lot of pre-written code and libraries. The Scikit_Learn python library comes with many machine learning algorithms for regression. I have used the Scikit_Learn library to implement, train, test and evaluate the model.

3.2 Data

Hewlett has sponsored automated student assessment prize competition to get fast, effective and affordable solutions for automated grading of the student-written essay. They have provided access to hand scored essays. For this competition, there are eight essay sets. Selected essays range from an average length of 150 to 550 words per response. All responses were written by students ranging in grade levels from Grade 7 to Grade 10. All essays were hand graded and were double scored.

The Hewlett essay scoring dataset (The Hewlett Foundation: Automated Essay Scoring) was used in this research. The dataset has eight essay sets, which are handwritten by students ranging in grade levels from Grade 7 to Grade 10. All essays were hand graded and were double scored. The training data contained essay sets 1-8. Each essay set had a description and a rubric for the score, in this project all 8-essay set were used. The description of each set is as follows:

Essay set #1 was written by grade level 8 students. The type of essays is persuasive/narrative/expository, the training set size contains 1785 essays, the average length of essays are 350 words. The set is evaluated by two raters, rater1 and rater2 who gave score1 and score2. The rubric range for essay set #1 is 1-6. This set consists of a resolved score which is the sum of the scores of both raters which ranges from 2-12.

Essay set #2 was written by students of grade-level 10, the type of essays is persuasive/narrative/expository, the training set size contains 1800 essays, the average length of essays are 350 words. The set was evaluated by two raters from two domains; domain1 is evaluated on writing application, i.e. the rubric is based on the ideas and content, the organization, style and voice of the essay which is evaluated on the rubric

range 1-6 by rater1 and rater2 of domain1. Domain2 was evaluated on the language conventions of the written essay, the rubric range is 1-4 graded by both rater1 and rater2 of domain2. In this project, domain1 scores are taken into consideration.

Essay set #3 was written by students from grade level 10, the type of essays is source dependent responses, the size of the training set is 1726 essays, the average length of essays is 150 words. The set is evaluated by two raters; rater1 and rater2 based on a rubric range 0-3, the resolved score of both raters is the average of the rater1 and rater2 and is in the range is 0-3.

Essay set #4 consists of grade level 10 students' essays of type source dependent responses, the training set size is 1772 essays and the average length of essays are 150 words. The set is evaluated by rater1 and rater2 on the rubric range 0-3, has a resolved score which is the best score of the rater1 and rater2 and the resolved score range is 0-3.

Essay set #5 contains students' essays in grade level 8, the type of essays are source dependent responses, the set consists of 1805 training essays with an average essay length of 150 words. The essay set #5 is graded by both the raters, rater1 and rater2. The score range of the rater1 and rater2 is 0-4, has a final score, which is the average of the rater1 and rater2, final has a rubric range 0-4.

Essay set #6 was written by students of grade level 10, the type of essays is source dependent responses, the training set size is 1800 essays and the average length of essays is 150 words. The set is evaluated by rater1 and rater2 based on the rubric range 0-4, which has a final score equals to the average of rater1 and rater2 with final score range 0-4.

Essay set #7 was written by grade level 7 students. The type of essays is persuasive/narrative/expository, the training set size contains 1730 essays, the average length of essays is 250 words. The set is evaluated based on different parameters like ideas, organization, style and conventions. Evaluation was done by two raters; rater1 and rater2 who gave score1 and score2. The rubric range for essay set #7 is 0-15. This set consists of a resolved score which is the sum of the scores of both raters with the range 0-30.

Essay set #8 was written by grade level 10 students the type of essays is persuasive/narrative/expository, the training set size contains 918 essays, the average length of essays is 650 words. The set is evaluated on different parameters like the ideas and content, organization, voice, word choice, sentence Fluency, style and conventions by three raters; rater1, rater2 and rater3 who gave score1, score2 and score3. The rubric range for essay set #8 is 0-30. This set consists of a resolved score which is composite of the scores of three raters and lies in the range 0-60.

In this research, the conducted experiments considered rater1 scores, rater2 scores and the average of rater1 and rater2 scores for all essay sets by preprocessing all the scores of the essay sets in the ranges 0-10. Figure 3 shows the histogram plot of both rater1 and rater2 with 0-10 score range on x-axis and the frequency of essays on y-axis (rater1 is represented in yellow and rater2 is represented in blue).

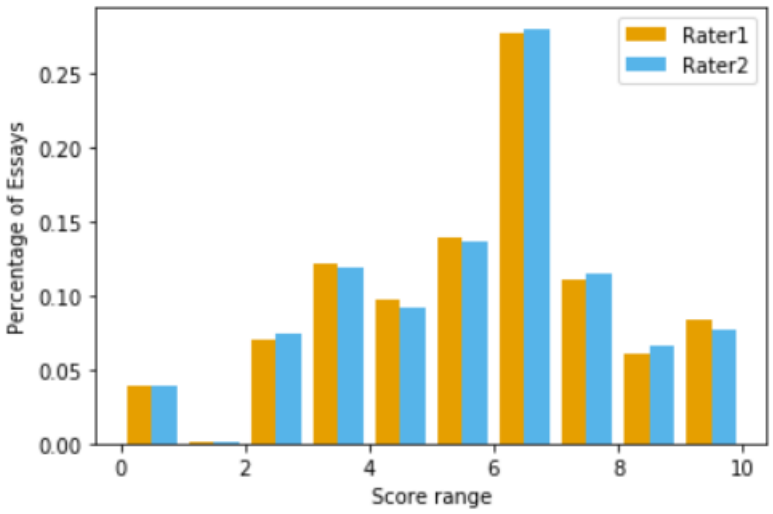

Figure 3 Distribution of essay scores for rater1 and rater2

3.3 Inter-Rater Reliability

Inter-rater reliability (IRR) is the extent of agreement between raters in terms of their evaluation of the same input data. Since the performance of the machine learning models in this project is evaluated based on their agreement with human graders, it is important to consider how well the human graders agree among themselves in their evaluation of essays used as input data. An IRR value indicating close agreement between the two human graders whose scores were used in this project would increase the reliability of the data used and consequently, any conclusion drawn on the relative performance of the machine learning models.

There are different approaches to checking IRR. In this project, Cohen's kappa statistic was used. Cohen's kappa (McNamara, D. S, et al, 2015) is a commonly used measure for IRR when data raters give scores for the same data items. The kappa statistic value ranges from 0 to 1; zero is considered as no agreement and one is considered as perfect agreement. In the current project, the data used had a 0.58 kappa value for IRR

between rater1 and rater2. This is considered as indicating moderate agreement in the kappa statistics.

3.4 Data and Feature Extraction

Coh-Metrix is a computational tool that produces indices of the linguistic and discourse representations of a text. These values are used in many ways to investigate the cohesion of the explicit text and the coherence of the mental representation of the text. Cohesion consists of characteristics of explicit text that play some role in helping the reader mentally connect the ideas in the text (Graesser, McNamara & Louwerse, 2003). The definition of coherence is the subject of much debate. Theoretically, the coherence of a text is defined by the interaction between linguistic representations and knowledge representations. When the spotlight is put on the text, however, coherence can be defined as characteristics of the text (i.e., aspects of cohesion) that are likely to contribute to the coherence of the mental representation. Coh-Metrix provides indices of such cohesion characteristics.

Coh-Metrix features were extracted from the dataset. This research used the Appendix-A listed features (Metrix) by importing the readability library from python. Appendix-A provides the list of indices in Coh-Metrix version 3.0. The first column provides the label that appears in the output in the current version. The second column provides the label used in prior versions of Coh-Metrix. The third column provides a short description of the index.

The data is stored in an Excel sheet with the following parameters: *essay_id*, *essay_set*, *essay*, *rater1*, *rater2*, and *domain1 score* which is the resolved score. The Python code reads the data from the Excel sheet, takes all the essays and convert them

into essay of vectors. Using the readability measures, each essay in the essay vector would have thirty-five extracted features as shown in Table1.

3.5 Data Preprocessing

Human graded scores, which are considered as target values were also extracted from the datasets. As mentioned earlier in 3.2, each essay set has a different score range, the score of each essay set is processed to score range 1-10. Principal component analysis technique was used to reduce dimensionality.

3.5.1 Principal Component Analysis (PCA)

"Principal component analysis (PCA) is a technique for reducing the dimensionality of datasets, increasing interpretability but at the same time minimizing information loss." (Cadima, 2016). PCA is used to get deeper insight into data. In PCA the axes are ranked in order of importance. Cells that are highly correlated are clustered together. If we have 4 or more-dimension data, we make 2 dimensional PCA plot.

Out of the 35 extarcted features, using the principal component analysis (PCA) 11 of the individual features reflect essential characteristics in essay writing and are aligned with human scoring criteria. The features and features distribution is shown in Figure 6 and Figure 7.

Using scikit learn the pca.fit() function, the fit function is used to transform the features, and to reduces the dimensionality.

No.	Name
1	-0.223sentence info.characters...
2	-0.509sentence info.characters_per_word...
3	0.688sentence beginnings.article...
4	0.797sentence beginnings.pronoun...
5	0.956sentence beginnings.interrogative...
6	-0.964sentence beginnings.subordination...
7	-0.998sentence beginnings.conjunction...
8	-0.897word usage.nominalization...
9	0.712word usage.auxverb...
10	0.689sentence beginnings.article...
11	-0.785sentence info.type_token_ratio...

Figure 6. Eleven features extracted using PCA

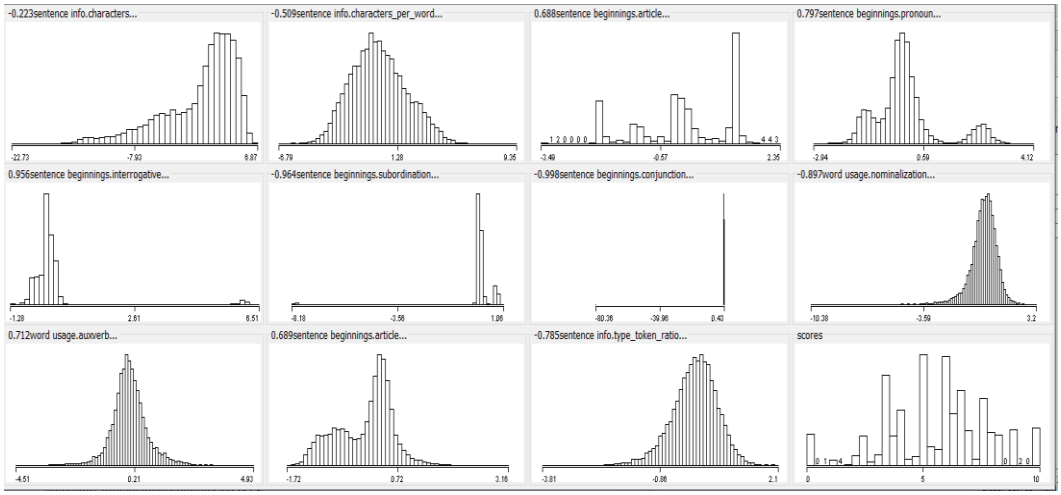

Figure 7 Distribution of eleven features extracted using PCA and score of all the essay instances

CHAPTER 4 Experimental Results and Discussions

In this project the final output is a score. From research and literature study the below algorithms achieved better results for similar problems. Machine learning algorithms used in this project are Linear Regression, Random Forest Regression and Support Vector Regression.

Linear Regression is a supervised machine learning algorithm, which performs the regression task (Burstein, J, et al., 1999). This model is used to predict a target value based on independent variables. The other algorithm used is support vector regression (SVR) which is a modified version of the support vector machine algorithm (Peng, X, et al, 2010). The support vector machine is used for classification problems, but for regression problems we need a real number as an output which makes it difficult to predict the information as we have numerous possibilities. For support vector regression, we made a small modification on the error function which helps to produce the real values as the output.

Usually, ensemble algorithms work better than standalone algorithms on hard tasks like automated essay scoring (Wijaya, E., et al, 2008). Decision Trees are also the fundamental components of Random Forests, which are among the most understandable machine learning algorithms available today (Chen, H., et al, 2013). Random forest regression was used as one of the most common ensemble algorithms which combine the multiple decisions from the decision trees to conclude a prediction score. Each decision tree is trained on a subset of the data.

4.1 Training Model

The dataset was split using the train and test split function in scikit_learn, the split function divides the data into training and testing sets, the size of the training set is 2/3 of the total data and the size of the testing set is 1/3 of the data. The features of the training set were stored in X_train and the scores or target values of the corresponding features was stored in the y_train. The test set features are stored in X_test and the scores are stored in y_test.

The train the model X_train and y_train values are used, along with the selected algorithm. Initially, used the linear regression algorithm which trains the model by fitting the training data.

```
Linear_reg = LinearRegression().fit(X_train, y_train
```

Similarly, for random forest regression and support vector regression algorithms, the following functions were used to train and fit the models.

```
Random_reg = RandomForestRegressor().fit(X_train, y_train)
```

```
SVR_reg = svm.LinearSVR().fit(X_train, y_train)
```

4.2 Testing Model

To test a model, A prediction function of scikit learn was used, which takes in the parameter X_test and produces Linear_pred, Random_pred, SVR_pred as the respective outputs.

```
Linear_pred = Linear_reg.predict(X_test)
Random_pred = Random_reg.predict(X_test)
SVR_pred = SVR_reg.predict(X_test)
```

Once the model is tested, the result can be obtained based on different parameters. In this project, the mean absolute error and mean squared errors were used to obtain the performance of the machine learning models.

4.3 Results

Table 2 shows the score for the essays, the average of rater1 and rater2 dataset allows for better performance when compared to rater1 or rater2 datasets alone. Among the models in the dataset, the average of rater1 and rater2 in the random forest regression model provided the best results.

Table 2 Results for different models and raters

Algorithm	Raters	Mean absolute error	Mean Square error
LinearRegression	Average of Rater1 and Rater2	1.42	3.18
RandomForestRegressor	Average of Rater1 and Rater2	1.22	2.52
LinearSVR	Average of Rater1 and Rater2	1.83	6.09
LinearRegression	Rater1	1.54	3.75
RandomForestRegressor	Rater1	1.37	3.11
LinearSVR	Rater1	1.78	5.51
LinearRegression	Rater2	1.54	3.72
RandomForestRegressor	Rater2	1.37	3.07
LinearSVR	Rater2	1.83	5.83

The results are shown in Figure 8 and Figure 9. Figure 8 represents the histogram and density plot for the three models: linear regression in green, random forest regression in blue and support vector regression in red. From the combined density and histogram plots the best performing model is the random forest model. There are some essays where linear regression model is working better, but the linear regression works well when the data has a linear connection with the score. While random forest can be used for data with the non-linear and linear connection of the score. Which explains the better performance of the random forest model than linear regression model. For this dataset SVR did not work as expected. Future testing is needed to understand why SVR didn't work as expected.

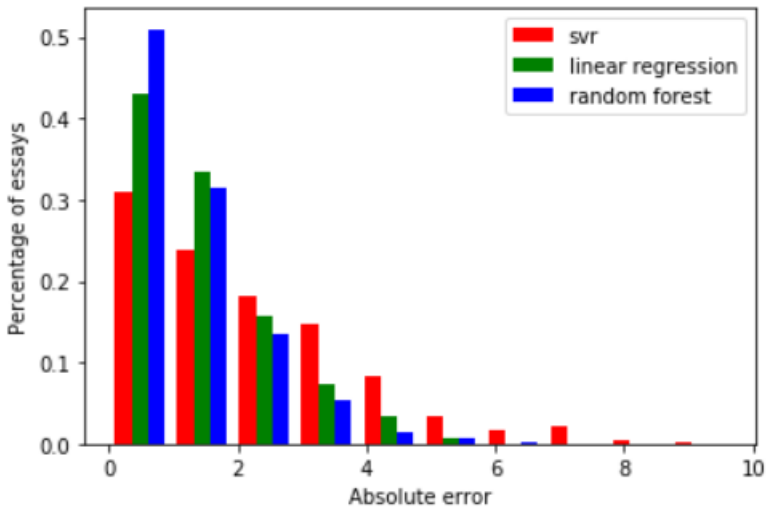

Figure 8 Histogram plot of absolute error for all three models

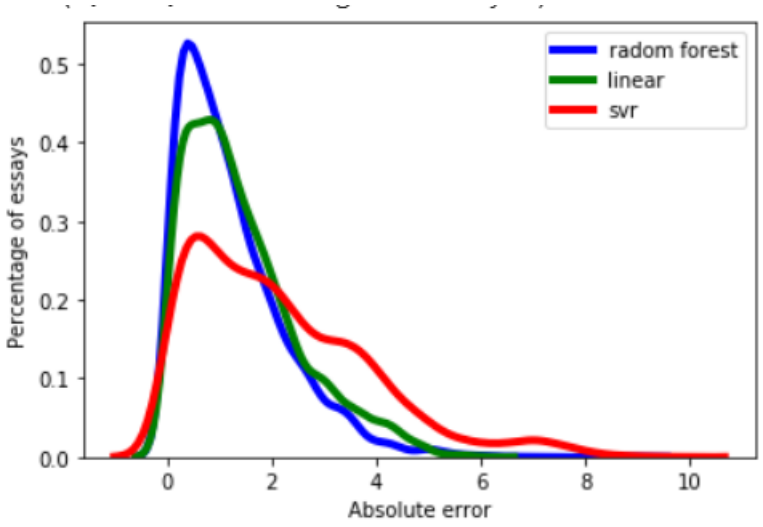

Figure 9 Density plot of absolute error for all three models

Figure 9 shows random forest performs well with an absolute error of zero for more than fifty-five percent of essays, and 10 percent of essays with absolute error more than three. Whereas the linear regression and SVR had forty and less than thirty percent of essays with an absolute error of zero respectively. The linear regression model performed better compared to the SVM model. Out of the three models, random forest outperformed other models as shown in the graph.

With respect to individual essays' scoring, 30 random essays were chosen to be scored by the three algorithms and then compared to the human raters. The results are shown in Table 3.

Table 3 Representing thirty random essays with human scores and predicted scores of all three models.

Essay Id	Average human score	Linear regression	Random forest regression	Support vector regression
8	8.0	7.04	7.8	10.62
15	4.0	5.51	4.8	5.78
65	5.0	4.66	5.7	5.06
83	6.0	6.93	5.7	6.98
4014	6.0	5.09	6.5	7.64
4027	5.0	4.92	5.75	6.55
4063	4.0	7.24	4.87	11.02
4124	3.0	5.01	2.98	4.15
4359	5.0	5.92	5.2	13.09
6383	6.67	5.46	6.67	3.35
6463	6.67	5.45	5.67	4.23
6480	6.67	5.6	6.43	3.84
7517	3.33	4.25	3.0	1.67
7526	8.33	4.74	7.22	4.69
7561	8.33	6.32	7.62	5.65
7674	5.0	5.21	5.83	4.47
9344	0.0	4.66	3.87	4.02
9367	8.33	5.24	7.04	4.17
9394	3.33	3.69	3.37	1.74

9428	8.33	5.91	8.17	6.23
9473	8.33	5.71	5.24	5.03
13068	5.0	5.51	6.08	4.31
13421	7.5	6.44	7.77	6.58
15568	8.75	6.35	6.88	5.56
15596	7.5	6.14	7.38	5.05
15790	10.0	6.48	9.88	7.4
19223	5.67	5.12	6.08	6.3
19243	8.0	6.3	7.6	8.53
21546	5.83	5.23	5.55	14.67
21574	6.67	7.3	6.67	26.35

Table 3 represents the thirty random essays and their essay id, average of rater1 and rater2 represented as an average human score and the predicted scores of all three models used in the project. The score range of the essays is between 0 and 10. From the table, linear regression and random forest both works well. While SVR gives unpredictable results, as we can observe from Table 3 the essay with essay id 21574 had predicted a score of 26.35 which is 16.35 points above the score range. Further testing needed to be done to understand SVR model behavior, because of the time constraint this testing is left for future work.

For all the three models, Cohen's kappa statistic is calculated by rounding the average human scores and the machine predicted scores to the nearest integer. The results calculate the inter-rater reliability between the average of human raters scores and the

system generated scores. The best performing model is the random forest with 0.14, which is followed by the linear regression model with 0.07 and support vector regression obtained 0.04 respectively. The inter-rater reliability performance of the models aligns with the performance of the mean absolute error of the models.

CHAPTER 5 Conclusion and Future Work

Automatic essay scoring helps users get an instantaneous score. It helps teachers reduce their work in grading essays. In this project, machine learning algorithms were used to generate automated essay scores by training and testing more than 12,000 human graded essays from the dataset. The models used in the project are linear regression, random forest regression and support vector regression. The models were tested on data evaluated by rater1, rater2, and the average score of rater1 and rater2. The average of the rater1 and rater2 score values was used as the benchmark score for comparing the performance of all the three machine learning models. The random forest regression model was found to outperform the linear regression and support vector regression models. It obtained a minimum absolute error of 1.22 as shown in Table 2.

5.1 Future Work

There are several ways in which the present project can be enhanced. The current work generates a score, but it can be implemented to provide feedback along with the score. The essays can be classified based on the content types (e.g. Expository, Descriptive, Narrative, Compare-&-contrast, Persuasive/argumentative). The present project focuses on English essays but, can be extended to develop the automated essay score in other languages (e.g. Hindi, Chinese).

Appendix-A

The Appendix-A shows the list of features of Coh-Metrix, with little description to each feature.

Title			Title
Genre			Genre
Source			Source
UserCode			UserCode
LSASpace			LSASpace
Date			Date
Time			Time
	Label in Version 3.x	**Label in Version 2.x**	**Description**
Descriptive			
1	DESPC	READNP	Paragraph count, number of paragraphs
2	DESSC	READNS	Sentence count, number of sentences
3	DESWC	READNW	Word count, number of words
4	DESPL	READAPL	Paragraph length, number of sentences, mean
5	DESPLd	n/a	Paragraph length, number of sentences, standard deviation
6	DESSL	READASL	Sentence length, number of words, mean
7	DESSLd	n/a	Sentence length, number of words, standard deviation
8	DESWLsy	READASW	Word length, number of syllables, mean
9	DESWLsyd	n/a	Word length, number of syllables, standard deviation
10	DESWLlt	n/a	Word length, number of letters, mean
11	DESWLltd	n/a	Word length, number of letters, standard deviation
Text Easability Principal Component Scores			
12	PCNARz	n/a	Text Easability PC Narrativity, z score
13	PCNARp	n/a	Text Easability PC Narrativity, percentile
14	PCSYNz	n/a	Text Easability PC Syntactic simplicity, z score
15	PCSYNp	n/a	Text Easability PC Syntactic simplicity, percentile
16	PCCNCz	n/a	Text Easability PC Word concreteness, z score
17	PCCNCp	n/a	Text Easability PC Word concreteness, percentile
18	PCREFz	n/a	Text Easability PC Referential cohesion, z score
19	PCREFp	n/a	Text Easability PC Referential cohesion, percentile
20	PCDCz	n/a	Text Easability PC Deep cohesion, z score

21	PCDCp	n/a	Text Easability PC Deep cohesion, percentile
22	PCVERBz	n/a	Text Easability PC Verb cohesion, z score
23	PCVERBp	n/a	Text Easability PC Verb cohesion, percentile
24	PCCONNz	n/a	Text Easability PC Connectivity, z score
25	PCCONNp	n/a	Text Easability PC Connectivity, percentile
26	PCTEMPz	n/a	Text Easability PC Temporality, z score
27	PCTEMPp	n/a	Text Easability PC Temporality, percentile
Referential Cohesion			
28	CRFNO1	CRFBN1um	Noun overlap, adjacent sentences, binary, mean
29	CRFAO1	CRFBA1um	Argument overlap, adjacent sentences, binary, mean
30	CRFSO1	CRFBS1um	Stem overlap, adjacent sentences, binary, mean
31	CRFNOa	CRFBNaum	Noun overlap, all sentences, binary, mean
32	CRFAOa	CRFBAaum	Argument overlap, all sentences, binary, mean
33	CRFSOa	CRFBSaum	Stem overlap, all sentences, binary, mean
34	CRFCWO1	CRFPC1um	Content word overlap, adjacent sentences, proportional, mean
35	CRFCWO1d	n/a	Content word overlap, adjacent sentences, proportional, standard devi
36	CRFCWOa	CRFPCaum	Content word overlap, all sentences, proportional, mean
37	CRFCWOad	n/a	Content word overlap, all sentences, proportional, standard deviation
38	CRFANP1	CREFP1u	Anaphor overlap, adjacent sentences
39	CRFANPa	CREFPau	Anaphor overlap, all sentences
LSA			
40	LSASS1	LSAassa	LSA overlap, adjacent sentences, mean
41	LSASS1d	LSAassd	LSA overlap, adjacent sentences, standard deviation
42	LSASSp	LSApssa	LSA overlap, all sentences in paragraph, mean
43	LSASSpd	LSApssd	LSA overlap, all sentences in paragraph, standard deviation
44	LSAPP1	LSAppa	LSA overlap, adjacent paragraphs, mean
45	LSAPP1d	LSAppd	LSA overlap, adjacent paragraphs, standard deviation
46	LSAGN	LSAGN	LSA given/new, sentences, mean
47	LSAGNd	n/a	LSA given/new, sentences, standard deviation
Lexical Diversity			
48	LDTTRc	TYPTOKc	Lexical diversity, type-token ratio, content word lemmas
49	LDTTRa	n/a	Lexical diversity, type-token ratio, all words
50	LDMTLDa	LEXDIVTD	Lexical diversity, MTLD, all words

51	LDVOCDa	LEXDIVVD	Lexical diversity, VOCD, all words
Connectives			
52	CNCAll	CONi	All connectives incidence
53	CNCCaus	CONCAUSi	Causal connectives incidence
54	CNCLogic	CONLOGi	Logical connectives incidence
55	CNCADC	CONADVCONi	Adversative and contrastive connectives incidence
56	CNCTemp	CONTEMPi	Temporal connectives incidence
57	CNCTempx	CONTEMPEXi	Expanded temporal connectives incidence
58	CNCAdd	CONADDi	Additive connectives incidence
59	CNCPos	n/a	Positive connectives incidence
60	CNCNeg	n/a	Negative connectives incidence
Situation Model			
61	SMCAUSv	CAUSV	Causal verb incidence
62	SMCAUSvp	CAUSVP	Causal verbs and causal particles incidence
63	SMINTEp	INTEi	Intentional verbs incidence
64	SMCAUSr	CAUSC	Ratio of casual particles to causal verbs
65	SMINTEr	INTEC	Ratio of intentional particles to intentional verbs
66	SMCAUSlsa	CAUSLSA	LSA verb overlap
67	SMCAUSwn	CAUSWN	WordNet verb overlap
68	SMTEMP	TEMPta	Temporal cohesion, tense and aspect repetition, mean
Syntactic Complexity			
69	SYNLE	SYNLE	Left embeddedness, words before main verb, mean
70	SYNNP	SYNNP	Number of modifiers per noun phrase, mean
71	SYNMEDpos	MEDwtm	Minimal Edit Distance, part of speech
72	SYNMEDwrd	MEDawm	Minimal Edit Distance, all words
73	SYNMEDlem	MEDalm	Minimal Edit Distance, lemmas
74	SYNSTRUTa	STRUTa	Sentence syntax similarity, adjacent sentences, mean.
75	SYNSTRUTt	STRUTt	Sentence syntax similarity, all combinations, across paragraphs, mean
Syntactic Pattern Density			
76	DRNP	n/a	Noun phrase density, incidence
77	DRVP	n/a	Verb phrase density, incidence
78	DRAP	n/a	Adverbial phrase density, incidence
79	DRPP	n/a	Preposition phrase density, incidence

80	DRPVAL	AGLSPSVi	Agentless passive voice density, incidence
81	DRNEG	DENNEGi	Negation density, incidence
82	DRGERUND	GERUNDi	Gerund density, incidence
83	DRINF	INFi	Infinitive density, incidence
Word Information			
84	WRDNOUN	NOUNi	Noun incidence
85	WRDVERB	VERBi	Verb incidence
86	WRDADJ	ADJi	Adjective incidence
87	WRDADV	ADVi	Adverb incidence
88	WRDPRO	DENPRPi	Pronoun incidence
89	WRDPRP1s	n/a	First person singular pronoun incidence
90	WRDPRP1p	n/a	First person plural pronoun incidence
91	WRDPRP2	PRO2i	Second person pronoun incidence
92	WRDPRP3s	n/a	Third person singular pronoun incidence
93	WRDPRP3p	n/a	Third person plural pronoun incidence
94	WRDFRQc	FRCLacwm	CELEX word frequency for content words, mean
95	WRDFRQa	FRCLaewm	CELEX Log frequency for all words, mean
96	WRDFRQmc	FRCLmcsm	CELEX Log minimum frequency for content words, mean
97	WRDAOAc	WRDAacwm	Age of acquisition for content words, mean
98	WRDFAMc	WRDFacwm	Familiarity for content words, mean
99	WRDCNCc	WRDCacwm	Concreteness for content words, mean
100	WRDIMGc	WRDIacwm	Imagability for content words, mean
101	WRDMEAc	WRDMacwm	Meaningfulness, Colorado norms, content words, mean
102	WRDPOLc	POLm	Polysemy for content words, mean
103	WRDHYPn	HYNOUNaw	Hypernymy for nouns, mean
104	WRDHYPv	HYVERBaw	Hypernymy for verbs, mean
105	WRDHYPnv	HYPm	Hypernymy for nouns and verbs, mean
Readability			
106	RDFRE	READFRE	Flesch Reading Ease
107	RDFKGL	READFKGL	Flesch-Kincaid Grade Level
108	RDL2	L2	Coh-Metrix L2 Readability

References

About the e-rater® Scoring Engine. (n.d.). Retrieved December 15, 2019, from https://www.ets.org/erater/about.

Allen, L., Dascalu, M., Mcnamara, D., Crossley, S., & Trausan-Matu, S. (2016). Modeling Individual Differences Among Writers Using Readerbench. EDULEARN16 Proceedings. doi:10.21125/edulearn.2016.2241

Automated Scoring of Writing Quality. (n.d.). Retrieved December 5, 2019, from https://www.ets.org/research/topics/as_nlp/writing_quality/.

Bolukbasi, T., Chang, K. W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In Advances in neural information processing systems (pp. 4349-4357).

Burstein, J., Kukich, K., Wolfe, S., Lu, C. and Chodorow, M. (1998) 'Enriching Automated Essay Scoring Using Discourse Marking', in E. Hovy (ed.) Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (ACL-COLING'98) Workshop on Discourse Relations and Discourse Markers, pp. 15–21, Montréal, Canada.

Burstein, J., & Chodorow, M. (1999, June). Automated essay scoring for nonnative English speakers. In Proceedings of a Symposium on Computer Mediated Language

Assessment and Evaluation in Natural Language Processing (pp. 68-75). Association for Computational Linguistics.

Chen, H., & He, B. (2013, October). Automated essay scoring by maximizing human-machine agreement. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 1741-1752).

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

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

Dascalu, M., Dessus, P., Bianco, M., Trausan-Matu, S., & Nardy, A. (2014). Mining texts, learner productions and strategies with ReaderBench. In Educational Data Mining (pp. 345-377). Springer, Cham.

Dronen, N., Foltz, P. W., & Habermehl, K. (2014). Effective sampling for large-scale automated writing evaluation systems. arXiv preprint arXiv:1412.5659. Fedorov, V. V. (1972). Theory of optimal experiments. New York: Academic Press.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring. In Proceedings of CONLL. pages 153–162.

Graesser, A. C., McNamara, D. S., & Louwerse, M. M (2003). What do readers need to learn in order to process coherence relations in narrative and expository text. In A.P. Sweet and C.E. Snow (Eds.), *Rethinking reading comprehension*. New York: Guilford Publications.

Hearst, M. (2000). The debate on automated essay grading. IEEE Intelligent Systems and Their Applications,15(5), 22-37. doi:10.1109/5254.889104

Jolliffe, lan T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philos Trans A Math Phys Eng Sci*. doi: 10.1098/rsta.2015.0202

Kintsch, W., & van Dijk, T. (1978). Toward a model of text comprehension and production. Psychological Review, 85, 363– 394.

Landauer, T. K., McNamara, D. S., Dennis, S., & Kintsch, W. (Eds.). (2007). Hand-book of latent semantic analysis. Mahwah, NJ: Lawrence Erlbaum.

learn. (n.d.). Retrieved December 5, 2019, from https://scikit-learn.org/stable/index.html.

Madnani, N., Burstein, J., Elliot, N., Klebanov, B. B., Napolitano, D., Andreyev, S., & Schwartz, M. (2018). Writing Mentor: Self-Regulated Writing Feedback for Struggling Writers. In Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations (pp. 113- 117).

McNamara, D., Crossley, S., & Roscoe, R. (2012). Natural language processing in an intelligent writing strategy tutoring system. Behavior Research Methods. Advance online publication. doi:10.3758/s13428- 012-0258-1

Metrix. (n.d.). Retrieved December 5, 2019, from http://www.cohmetrix.com/.

Peng, X., Ke, D., Chen, Z., & Xu, B. (2010, October). Automated Chinese essay scoring using vector space models. In 2010 4th International Universal Communication Symposium (pp. 149-153). IEEE.

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

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. doi:10.18653/v1/n18-1202

ReaderBench. (n.d.). Retrieved December 5, 2019, from http://www.readerbench.com/.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. IBM Journal of Research and Development, 3(3), 211–229.

Shankar, R. S., & Ravibabu, D. (2018). Digital Report Grading Using NLP Feature Selection. *Soft Computing in Data Analytics Advances in Intelligent Systems and Computing*,615-623. doi:10.1007/978-981-13-0514-6_59

The Hewlett Foundation: Automated Essay Scoring. (n.d.). Retrieved December 5, 2019, from https://www.kaggle.com/c/asap-aes/data.

Wijaya, E., Yiu, S. M., Son, N. T., Kanagasabai, R., & Sung, W. K. (2008). MotifVoter: a novel ensemble method for fine-grained integration of generic motif finders. Bioinformatics, 24(20), 2288-2295.

Woods, B., Adamson, D., Miel, S., & Mayfield, E. (2017). Formative Essay Feedback Using Predictive Scoring Models. Paper presented at the Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mini

Zupanc, K., & Bosnić, Z. (2018). Increasing accuracy of automated essay grading by grouping similar graders. Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics - WIMS 18. doi:10.1145/3227609.3227645