MONASH University

Improved Semantic Features for Automated Essay Scoring with Hybrid Topic Modeling Approach

Jih Soong Tan

Master of Philosophy

A Thesis Submitted for the Degree of Master of Philosophy at
Monash University in 2023
School of Information Technology

Copyright notice

Abstract

Essay writing is one of the crucial elements in all language proficiency tests. However, human evaluation is inconsistent, time-consuming, and labor-intensive. While the popularization of Automated Essay Scoring (AES) models in numerous institutions for scoring essays, current AES models still have the issue of extracting the semantic attributes from the essays, which affects their performance in evaluating essays. Hence, content detection and sentence similarity features are proposed to extract the semantic attributes from the essay.

Content detection is proposed to determine the main content or topics in the written essays. The Latent Dirichlet Allocation (LDA) topic modelling algorithm is proposed for content detection. The topic modelling algorithm determines the topic contents of a particular essay by extracting latent features. The content detection features are based on a topic modelling algorithm to identify how well the content of a particular essay is in relation to all the other essays.

A sentence-prompt similarity measure based on a transformer model is proposed to determine if the essay is written surrounding a given prompt. A transformer model takes account of the contextual meaning of words, which is hypothesized to improve the prompt similarity measure's performance in AES.

The proposed features are concatenated with existing lexical and grammatical features with the fine-tuned transformer model's output, then trained upon a stacking regressor machine learning model. The results demonstrated that the proposed method outperforms the existing state-of-the-art solutions in the long essays segment. Additionally, an improvement of 0.005 Quadratic Cohen Kappa (QWK) score is observed in the narrative essays even when the QWK score is already achieved above 0.8, which represents almost perfect agreement with the prediction result.

The proposed features are tested to be robust and perform better compared to the base model. The topic modelling feature performed better on the response genre essays, while the sentence-prompt similarity features performed better on the argumentative and narrative essays. Furthermore, a study is made to estimate the optimum topic number to reduce the computation cost of training LDA topic modelling.

Declaration

This thesis is an original work of my research and contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Signature: _____

Print Name: Tan Jih Soong

Date: June 2023

Publications during enrolment

1) J. S. Tan, I. K. Tan, L. K. Soon, and H. F. Ong. Improved automated essay scoring using gaussian multi-class smote for dataset sampling. In Proceedings of the 15th International Conference on Educational Data Mining, page 647, 2022.

Acknowledgements

First and foremost, I would like to express my deep and sincere gratitude to my supervisor, Dr Ong Huey Fang, co-supervisor, Dr Soon Lay Ki, and also external supervisor, Dr. Ian Tan for the continuous support of my Master's study and research. Their guidance helped me in all the time of research and writing of the reports and this thesis. Their advice and comments have allowed me to overcome the challenges I faced.

Besides my supervisors, I would like to thank Dr Lim Wern Han and Dr Jacky Rong for their insightful comments which further helped me to improve my work. Much appreciate for the guidance provided by Dr Lim Wern Han and Dr Jacky Rong for producing better quality of thesis.

Contents

Copyright notice i

Abstract ii

Declaration iii

Publications during enrolment iv

Acknowledgements v

List of Figures ix

List of Tables x

Abbreviations xi

1 Introduction 1
- 1.1 Research Overview . . . 1
  - 1.1.1 Problem Statement . . . 2
  - 1.1.2 Research Questions . . . 3
  - 1.1.3 Research Objectives . . . 3
- 1.2 Scope of Research . . . 4
- 1.3 Potential Research Contributions . . . 4
- 1.4 Structure of Thesis . . . 5

2 Literature Review 6
- 2.1 Review on the Development of AES . . . 6
  - 2.1.1 Feature-based Approaches . . . 7
  - 2.1.2 Neural Network Approaches . . . 11
  - 2.1.3 Hybrid Approaches . . . 14
- 2.2 Review on the topic modeling techniques . . . 16
  - 2.2.1 Background . . . 17
  - 2.2.2 Existing Topic Model Techniques . . . 19
    - 2.2.2.1 Latent Dirichlet Allocation . . . 19
    - 2.2.2.2 Correlated Topic Model . . . 21
    - 2.2.2.3 Dynamic Topic Model . . . 22
    - 2.2.2.4 Non-negative Matrix Factorization . . . 23
    - 2.2.2.5 Notable topic models . . . 24
- 2.3 Review on the similarity algorithm . . . 24

2.3.0.1 Lexical Similarity . . . 25
2.3.0.2 Grammatical Similarity . . . 26
2.3.0.3 Semantic Similarity . . . 26
2.3.1 Similarity approaches for AES . . . 27
2.3.1.1 N-gram . . . 28
2.3.1.2 LSA . . . 28
2.3.1.3 Pretrained embeddings . . . 29
2.4 Summary . . . 29

3 Research Framework **34**
3.1 Topic Model - Latent Dirichlet Allocation . . . 36
3.1.1 Data Preprocessing for Topic Model . . . 36
3.1.2 Fine-tuning of LDA . . . 36
3.2 Sentence similarity toward prompt . . . 37
3.2.1 Transformer model to sentence similarity toward prompt . . . 39
3.2.2 The feature engineering of sentence similarity . . . 39
3.2.3 Existing Feature Engineering . . . 40
3.3 Hybrid modeling . . . 40
3.3.1 RoBERTa fine-tuning . . . 41
3.4 Summary . . . 41

4 Experiment Setup **42**
4.1 Benchmark Dataset . . . 42
4.2 Feature engineering . . . 43
4.3 Learning algorithms . . . 44
4.4 Performance Metric . . . 44
4.4.1 Quadratic Weighted Kappa . . . 44
4.4.2 Paired t-test . . . 45
4.5 Summary . . . 46

5 Content Detection with Topic Modeling **47**
5.1 Hyperparameter fine-tuning of LDA topic model . . . 47
5.2 Evaluations of LDA Topic Model features . . . 48
5.2.1 QWK Evaluations for the LDA Topic Model features . . . 48
5.2.2 Significant test of LDA topic model features . . . 50
5.3 Discussions . . . 50
5.3.1 Effect of number of topics on LDA topic models . . . 52
5.3.2 LDA topic modelling in AES model . . . 53

6 Sentence Similarity Toward Prompt **54**
6.1 Evaluations of Sentence similarity towards prompt features . . . 54
6.1.1 QWK Evaluations for the sentence similarity towards prompt features . . . 55
6.1.2 Significant test of sentence similarity features . . . 56
6.2 Discussions . . . 56
6.2.1 The effect of sentence similarity towards prompt on different essay sets. . . . 57
6.2.2 Sentence similarity towards prompt in AES model . . . 57

7 Hybrid modeling with Content Detection and Sentence Similarity towards Prompt **60**
7.1 Topic vectors with sentence similarity features 61
7.2 Comparison of hybrid models 62
7.3 Discussion 63

8 Conclusion **65**
8.1 Research Objective Review 65
8.2 Limitations and further research 66

A Other experiment results **68**

Bibliography **69**

List of Figures

2.1 General Hybrid Approaches' architecture diagram 17
2.2 The LDA model's graphical representation diagram. The largest box, M, indicates documents, and the box, N, indicates topics and words within a document. The smallest box, K, represents sampling for each topic . . . 20
2.3 The CTM model's graphical representation diagram. The $\sum$ indicates the covariance, μ indicates mean. 22
2.4 The DTM model's graphical representation diagram. 23
2.5 Publication Count over the Years 30
2.6 The filters to select the most suitable topic model for AES. 32

3.1 Whole Level Architecture Diagram 35
3.2 The flow of the implementation of Topic Model in AES 36
3.3 The flow of the implementation of Sentence Similarity toward Prompt . . 38

4.1 The flow of the experiments in the next three chapters. 46

5.1 Effect of a different number of topics for LDA topic model on the essay sets. 51
5.2 Further study on the effect of the number of topics on essay set 7. 53

6.1 Average sentence similarity score against essay score on different essay sets. 58

8.1 Example evaluation feedback report for students 67

List of Tables

2.1 Table summary of reviewed AES papers . . . 30
2.2 Summary of reviewed similarity measures in AES. . . . 33

3.1 Sentence similarity features descriptions. . . . 39
3.2 Features based on the existing works. . . . 40

4.1 ASAP dataset details. . . . 43

5.1 The optimum LDA hyperparameters from the grid search. . . . 48
5.2 QWK comparison between the LDA Topic Model and the base model. . . . 49
5.3 Variance of QWK comparison between LDA Topic Model and the base model. . . . 49
5.4 The LDA topic model features paired t-test test results. . . . 50

6.1 Result of sentence similarity towards prompt features with existing features on each essay sets in QWK. . . . 55
6.2 Variance of QWK comparison between sentence similarity against prompt and the base model. . . . 56
6.3 The Sentence similarity paired t-test test results . . . 57

7.1 Result for the combination of topic vectors and sentence similarity features. 61
7.2 Variance of QWK comparison between LDA + Sim models and the base models. . . . 62
7.3 Result of the proposed model against the fine-tuned RoBERTa and the state-of-the-art models of recent years. . . . 63

A.1 Result comparison based on BERTopic Topic Model against the base model. 68
A.2 Result of hybrid modeling compared to base, Roberta and BERT. . . . 68

Abbreviations

AES	**A**utomated **E**ssay **S**coring
LDA	**L**atent **D**irichlet **A**llocation
LSA	**L**atent **S**emantic **A**nalysis
CTM	**C**orrelated **T**opic **M**odel
QWK	**Q**uadratic **W**eighted **K**appa
ASAP	**A**utomated **S**tudent **A**ssessment **P**rize
SVM	**S**upport **V**ector **M**achine
RASP	**R**obust **A**ccurate **S**tatistical **P**arsing
NLTK	**N**atural **L**anguage **T**ool **K**it
EASE	**E**nhanced **A**rtificial Intelligence **S**coring **E**ngine
BLRR	**B**ayesian **L**inear **R**idge **R**egression
LSTM	**L**ong **S**hort-**T**erm **M**emory
CNN	**C**onvolution **N**eural **N**etwork
RNN	**R**ecurrent **N**eural **N**etwork
BoW	**B**ag-**o**f-**W**ords
GloVe	**Glo**bal **Ve**ctor for Word Representation
XGBoost	e**X**treme **G**radient **Boost**ing
BERT	**B**idirectional **E**ncoder **R**epresen**T**ations
RoBERTa	**R**obustly **o**ptimized **B**idirectional **E**ncoder **R**epresen**T**ations **a**pproach
tf	*t*erm *f*requency
idf	*i*nverse *d*ocument *f*requency
POS	**P**art-**O**f-**S**peech

Chapter 1

Introduction

Every language proficiency test, such as the Test of English as a Foreign Language (TOEFL) and International English Language Testing System (IELTS), has essay writing as one of the most important elements to assess one's proficiency in specific language writing abilities. However, the human evaluation of essays is time-consuming and labour-intensive. Every human has a different bias that is affected by subjectivity factors which raise the issue of unfairness in the evaluation [88]. Hence, Automated Essay Scoring (AES) was invented for evaluating essays, reducing the human resources needed and bringing fairness to the evaluation by having the same evaluator in the process.

1.1 Research Overview

In recent years, most AES systems are generally based on extracting three linguistic feature groups, lexical, grammatical, and semantic features [2, 13, 31, 42]. However, the performance of such systems is bound by the quality of the feature input.

For human essay evaluation, the human grader will typically follow the parameters provided, such as the development of ideas, coherence, cohesion, grammar, vocabulary, and others. However, the vast majority of the AES system aims to improve the performance of AES models without considering the parameters for essay scoring tasks. From the ablation study on AES features, [92], semantic features are identified as the current weakness in the AES system. Hence, a revision to the semantic feature group is proposed based on a generic feature engineering technique in AES. The semantic feature

group in the generic AES system has three main problems. First is the inability to determine if the essay is written around a specific prompt semantically. It is crucial for essay evaluation that the essay is written around a prompt or essay topic semantically [68]. Second, the non-consideration of the development of ideas in semantic features and sentence-level semantic meaning. Typically, a paragraph should have one topic sentence that contains the main idea, and the other sentences as the reasoning around the main idea [20]. Third, the non-consideration of coherence and cohesion attributes in semantic features. The coherence between sentences is crucial for an essay to have relevant semantic meaning [62]. Hence, the proposed method enriches the semantic features by tackling the first two previously stated problems.

Klebanov et al. [46] have experimented on the relationship of semantic features on essay scoring and determined that the supporting ideas or content are the most significant element in the AES model. Hence, one of the goals of this research is to build a topic model that detects topic sentences in the essays, which also tackles the second problem stated previously. With the topic sentences detected, it is also significant to identify if the topic sentences are relevant to the prompt. This led to the second goal of this research which is to build an algorithm that measures the sentence similarity score toward the prompt.

This chapter provides an overview of the research by presenting the problem statement, research questions, objectives, and proposed framework's contributions.

1.1.1 Problem Statement

The current Automated Essay Modeling (AES) model trend is moving towards deep learning and pre-trained models [91, 95, 108]. However, the needs of feature engineering are still necessary to improve the model, as proven in [86]. Also, the current stage of the AES model with deep learning is still lacking in performance due to the small dataset size and the high cost of natural language processing tasks. With the help of feature engineering, it can help machine learning algorithms to understand data better and provide more reasonable results. The majority of the AES systems' feature engineering is based on three main feature groups, namely lexical, grammatical and semantic feature groups. An ablation study on the existing AES model discovered that the semantic features are lacking in the field of AES [92]. In the field of AES semantic attributes'

feature engineering, recent studies have focused on coherence element [53, 58] in the semantic part of essay scoring. Hence, there is a need to create a feature to identify supporting ideas and arguments in the essay scoring to close the gap in the field. The n-gram matching semantic features are proven weak for the AES model's performance [92]. Hence, a higher level of semantic features needs to be discovered, developed and evaluated.

1.1.2 Research Questions

The research questions to be addressed are as follows:

RQ1 How can a topic modeling algorithm be used and evaluated effectively in the AES field?

Topic modeling algorithms such as Latent Semantic Analysis, Latent Dirichlet Allocation (LDA), and Correlated Topic Model (CTM), and the parameters will be explored and fine-tuned for AES on different types of essays. A new feature has been engineered based on the topic modeling outcomes to evaluate its effectiveness on AES.

RQ2 How to use sentence similarity in relation to a given prompt for effective AES?

An exploration using the transformer-based model and prompt on sentence similarity has been investigated.

RQ3 How can topic modeling be integrated with sentence similarity to become a new feature for AES?

An ablation study for the new features is being done here, and the results have concluded my research.

1.1.3 Research Objectives

A prompt in essays is a statement that provides a potential idea for students to write surrounding it. Content in this project refers to the sentences that contain the main idea or support points. Topic modeling and similarity scores will be applied at the sentence level. Hypothetically, as a human grader, for semantic evaluation, we will evaluate

whether the essay has sufficient content in relation to the topic and whether the content is similar to the prompt provided. In other words, an effective AES will determine how well the essay fits into the topic as understood by all the essays submitted and how related the essay is to the prompt.

To address the research questions stated in the previous section, the research objectives are structured as follows:

RO1 To identify a suitable topic modeling technique and features that determine how well the content of a particular essay is in relation to all the other essays.

RO2 To design a sentence similarity score in relation to a given prompt based on a transformer-based model for effective AES.

RO3 To integrate the topic modeling and sentence similarity as new features to existing AES.

1.2 Scope of Research

This research evaluates the newly proposed method by comparing its performance against the existing state-of-the-art methods' performance. The main performance evaluation metric is the Quadratic Weighted Kappa (QWK) score. The Automated Student Assessment Prize (ASAP) competition's dataset is being used in this research to evaluate the proposed method.

1.3 Potential Research Contributions

In the field of AES, there's no research directed toward identifying supporting ideas or main content in essays. Hence, the first contribution of the proposed work is the newly proposed method of topic modeling to determine how well the content of a particular essay is in relation to all the other essays. The research provides guidelines and a framework for AES researchers, demonstrating how existing algorithms can be applied effectively. While it may not introduce new algorithms, its methodology and

experimental results have the potential to benefit future investigations and advance the field of AES. Secondly, another contribution of the proposed work is the newly proposed method to measure the sentence similarity score in relation to a given prompt based on a transformer-based model. This will be the main contribution of the proposed work because of the newly proposed method to measure the essay to prompt similarity score in sentence level.

1.4 Structure of Thesis

The rest of this report is structured as follows:

Chapter 2 reviews the approaches on the AES, topic modeling and text similarity.

Chapter 3 presents the proposed methodology framework.

Chapter 4 presents the experimental setup of the work.

Chapter 5 illustrates the experiment results of the content detection with topic modeling. The effect of the topic modeling on AES will be discussed in this chapter.

Chapter 6 illustrates the experiment results of the sentence similarity toward the prompt. The effect of the sentence similarity towards the prompt on AES will be discussed in this chapter.

Chapter 7 presents the experiment results of the hybrid model that concatenates content detection with topic modeling and sentence similarity toward the prompt.

Chapter 8 reviews the research objective, limitations and further direction of the research.

Chapter 2

Literature Review

The essay scoring should follow the parameters such as the development of ideas, similarity of content in relation to the prompt, cohesion, coherence, grammar, syntax, etc. However, in the field of AES, not all proposed approaches follow the parameters. Hence, Session 2.1 will review the AES approaches to illuminate the evolution within the field and use the knowledge to strengthen this research.

Regarding the previous study [92], **RO1** is proposed to identify the development of ideas in the essays. Hence, an investigation into topic modeling techniques is required to proceed. Section 2.2 will present a review of the topic modeling techniques to investigate suitable topic modeling techniques for AES purposes.

Additionally, the evaluation of essay scoring should identify if the development of ideas is in relation to the prompt. A proper essay writing should be written surrounding the given prompt. Thus, a study of text similarity approaches will be made and written in Section 2.3.

Finally, Session 2.4 summarizes the AES approach over the years, the most suitable topic modeling method for AES and the similarity measures in AES.

2.1 Review on the Development of AES

After Kaggle's Automated Student Assessment Prize (ASAP) competition, different types of feature engineering approaches were proposed by researchers to enhance the

performance of the AES system. With the deep learning algorithms growing mature, the recent works on AES are starting to involve deep learning elements. The proposed approaches for the AES system are reviewed and synthesized into three approaches, namely feature-based approaches, deep learning approaches, and hybrid approaches. The feature-based ones are more explainable and also has the potential to provide significant result. The deep learning approaches often provide better results in AES than feature-based approaches. Lastly, the hybrid approaches that merge the implementation of the feature engineering and the deep learning algorithms enable it to provide a better performance model with explainable results. A few notable pieces of research will be selected to be further discussed in each approach.

2.1.1 Feature-based Approaches

The feature-based approaches usually involve feature engineering to extract features from the essays and then inputting the features to train the machine learning algorithms. Feature engineering mainly extracts three features: lexical, grammatical, and semantic. These approaches are typically proposed with new feature engineering methods to improve lexical, grammatical, or semantic performance. In these approaches, the whole process will convert the essays into features in numbers such as length of words, number of word usage errors, and number of vocabulary used. Hence, there will be a high chance of ignoring the semantic features in the feature engineering.

In 1999, Foltz et al. [31] proposed a new feature based on Latent Semantic Analysis (LSA) to obtain the domain-representation text from the course materials and then use it to detect the contents in the essays. In LSA, term co-occurrences in a corpus are measured by means of a dimensionality reduction based on singular value decomposition (SVD) on the term-by-document matrix. However, this method has several limitations. First is the lacking of course materials provided in most datasets. Second, an immense amount of domain in course material could dilute content detection. Third, lacking other types of features to support the AES task.

In 2006, Attali and Burstein [2] proposed an AES regression model that is based on 10 features. Besides the common grammatical and lexical features (grammar error, word usage error, vocabulary), the author has few notable semantic features such as organization & development and prompt-specific word usage. The author implemented

an online tool called Criterion to obtain the grammatical and lexical features. The organization & development features assume the essays are written in the following structure to determine the organization score and main points. However, this proposed method is not practical as the essays may have different structures. The prompt-specific word usage features are a content measure feature by calculating the relative frequency of words used in high-scored essays against low-scored ones. However, the proposed method is limited to n-gram matching methods, which can only capture the meaning up to the word level.

In 2009, Mohler and Mihalcea [63] have proposed a text-to-text semantic similarity feature. The features are based on two types of text similarity metrics: knowledge-based and corpus-based metrics. There are eight knowledge-based metrics and two corpus-based metrics being tested by the authors. The eight knowledge-based metrics are the shortest path, Leacock & Chodorow, Lesk, Wu & Palmer, Resnik, Lin, Jiang & Conrath, and Hirst & St. Onge. The shortest path similarity metric determines the distance of the shortest path between two contexts using node counting. The Leacock & Chodorow similarity metric is similar to the shortest path metric but involves taxonomy depth. The Lesk similarity determines the overlap between the contexts based on a dictionary. The Wu & Palmer similarity metric combines the depth of two given contexts in the WordNet and the depth of the least common subsumer to determine the similarity score. The Resnik similarity metric determines the information content of the least common subsumer of the two contexts. The Lin similarity metric is based on the Resnik metric with added normalization factor. The Jiang & Conrath similarity metric is similar to the Lin metric but includes probability factors. The Hirst & St. Onge similarity metric determines the similarity by using the WordNet hierarchy to detect the lexical chains between two contexts. The two corpus-based metrics are LSA and Explicit semantic analysis (ESA), both trained on Wikipedia data. ESA is a variation of a standard vectorial model where the dimensions of the vector are directly equal to abstract contexts. The LSA can achieve the highest accuracy score among all the similarity metrics. However, the result might not apply to all essays as the author only trained the model on short essays.

Yannakoudakis et al. [109] have implemented robust, accurate statistical parsing (RASP) system [11] to perform feature engineering in the essays. Then, they input the features into the Support Vector Machine (SVM) for the AES classification task. Similar to

Yannakoudakis et al. [2] proposed approach, the RASP system extracts lexical features, including essay length, word n-gram, and grammatical features, including grammatical error, structures error. However, the RASP system does not provide semantic features, which could be the system's weakness. Also, the author has mentioned that a coherence feature should be significant as their study is based on n-grams.

In 2013, Persing and Ng [71] used a support vector machine (SVM) model to classify the score of the essay. Similar to Yannakoudakis et al. [109], the authors have grammatical and lexical features like a spelling error, the word n-gram. The author proposed new features such as keywords and semantic roles for semantic features. The keywords are determined by the manual human efforts that defeat AES's purpose to automate the whole process. The semantic roles are defined based on the frame-semantic parsing method proposed by Das et al. [22]. Again, it defeats AES's purpose as the output of the semantic roles required human effort to identify each frame semantics role.

In 2014, Adamson et al. [1] proposed an AES regression model based on support vector regression (SVR). They have featured the essays and obtained four lexical features, including word n-grams, number of characters, number of words, number of sentences, two grammatical features including part of speech n-grams, number of misspellings, and one semantic feature, essay similarity score. The word n-grams determine the vocabulary used in the essays as it identifies how many times a word appeared. The part-of-speech n-grams are based on Natural Language Toolkit (NLTK) to identify the grammar error in the different combinations of part-of-speech. The proposed essay similarity score is based on LSA to obtain the reduced term matrix and calculate the cosine similarity between essays. The author has identified that the proposed similarity score feature is overfitting the model, which shows that the LSA is not a suitable similarity measure for the AES task. The author obtained a QWK score between 0.629 and 0.819 based on the ASAP dataset.

In 2015, Phandi et al. [72] implemented the Enhanced AI Scoring Engine (EASE) engine as the feature engineering technique to extract the lexical, grammatical, and semantic features from the essays and use it to train a Bayesian Linear Ridge Regression (BLRR). The lexical and grammatical features are similar to Adamson et al.'s proposed features. The semantic features of EASE compare the words used in the essays against the prompt to identify if the essays are semantically similar to the prompt. One of the top 3 winners

of the ASAP competition invented the EASE engine. Using EASE on BLRR, they obtained an average 0.7045 Quadratic Weighted Kappa (QWK) score based on the ASAP dataset. However, the previous study [92] shows that the semantic attributes are the current weakness of the AES model.

In 2016, Cummins et al. [19] proposed a Perceptron Ranking model that is based on a margin-based linear classifier, Timed Aggregate Perceptron (TAP) vector. They use the tap to provide a ranking to all the essays, then transform the ranking to classify the essay's score. Their proposed features are somewhat identical to the RASP system implemented by Yannakoudakis et al. [109], with added two more lexical features (max word length, sentence length). However, similar features cause the model to have similar weaknesses in the semantic features. They managed to obtain an average 0.747 QWK score on the ASAP dataset.

In 2018, Contreras et al. [17] have proposed to use an ontology tool, OntoGen, to find the main contents from the course material provided by teachers in the essays and NLTK library to extract lexical and grammatical features that are similar to the RASP system implemented by Yannakoudakis et al. [109]. Besides content detection, the author also implemented LSA to identify the similarity score of the essays against the sample representative essay of each prompt. However, this might not be applicable in most cases because a different individual might have a different writing style. Using the features generated, they input the features into a linear regression model to predict the score of the essays.

Additionally, in 2019, Darwish and Mohamed [21] proposed a new feature engineering approach on lexical, grammatical, and semantic features. In the lexical feature, they proposed Lexical Analysis to measure lexical richness. In the grammatical feature, they proposed constructing parse trees to detect broken trees as syntax errors in the essays. They combine the Lexical Analysis and parse tree results to form the syntax score. The semantic features are similarity analysis, spatial data analysis, and spatial autocorrelation. The similarity analysis is similar to the method proposed by Foltz et al., but the essays are tokenized into sentences to find average sentence similarity in the essay itself. The spatial data analysis measures the Euclid distance between the centre and each point. The spatial autocorrelation determines how the sentences incline to be grouped. The three semantic features proposed by the author are mainly targeted to detect the

coherence in the essays. Again, the author has no features related to content detection in the essays.

Moreover, in 2019, Salim et al. [82] proposed an XGBoost Machine Learning classifier model to classify the essays. The features used are somewhat a combination of features proposed over the years, such as words count, sentences count, average sentence length, commas count, grammatical errors, parse tree depth, part-of-speech, unique vocabulary, and sentence similarity [1, 21, 42, 72, 109]. In contrast, they also proposed some new semantic features such as noun ratio in each sentence and transition word ratio to detect coherence and cohesion in the essays. This proposed method does not provide any feature that detects content in the essays.

For human essay evaluation, the human grader will typically follow the parameters, such as developing ideas, coherence, cohesion, grammar, and lexical. However, the vast majority of the AES system aims to improve the performance of AES models without considering the parameters for essay scoring tasks. However, most feature-based approaches focus on lexical and grammatical features to train the AES models instead of semantic features. Among semantic features, the most focused direction is coherence and cohesion. Only a few works perform content detection, and the works are not achieving good results in content detection. In the current stage of AES, to identify the relevance of the essays' content to the prompt, there is no further development nor discussion after the n-gram matching technique. N-gram matching techniques are limited to only capturing the meaning based on the words. Different combinations of words or phrases might contain different meanings.

2.1.2 Neural Network Approaches

Deep learning approaches in AES typically involve modifying the design of the neural networks or transformer-based models.

Taghipour and Ng [91] proposed the first approach that implements deep learning algorithms in AES. They proposed to implement a convolution neural network (CNN) and recurrent neural networks (RNN) based on long short-term memory (LSTM) networks to perform the AES task. The network implements a lookup table with the one-hot

representation of the word vector of essays. They obtained a 0.708 average QWK score with CNN+LSTM on the ASAP dataset.

Dong and Zhang [28] have proposed a Hierarchical Convolutional Neural Network (CNN) with two convolution layers. The first convolution layer is the word-level convolution layer used to extract sentence representations from word embedding. The second convolution layer is the sentence-level convolution layer used to extract the essays' main content and representatives. A fully connected dense layer predicts scores for essays. The proposed model obtained a 0.734 QWK score on the ASAP dataset.

Dong et al. [29] have proposed an attention-based Recurrent Convolutional Neural Network for the AES task. This proposed method is based on the architecture proposed by Dong and Zhang [28] with further enhancement. Unlike Dong et al.'s proposed architecture, this method inputs character embedding and word embedding into CNN with added attention pooling layers. The character embedding and word embedding are obtained using the NLTK library. The output of CNN will provide a sentence vector with sentence weight. With that, the LSTM layer with attention-pooling layers will go through a sigmoid layer for the final score prediction of essays. It managed to obtain an average 0.764 QWK score on the ASAP dataset.

Zhao et al. [113] have proposed a new neural network concept, Memory-Augmented Neural Network. The proposed model has four layers: the input representation layer, memory addressing layer, memory reading layer, and output layer. The input representation layer converts essays into a vector representation. The memory addressing layer generates the weight of each term from the sample representative essay from every score domain. The memory reading layer takes the sum of the weights from the previous layer to produce the weighted sum. The output layer takes the weighted sum to measure the final score of the essay.

In 2018, Wang et al. [102] proposed a bidirectional LSTM RNN model with reinforced learning. The reinforced learning is based on the average QWK to measure each epoch's loss. The essays are converted into dense vector representations via word2vec algorithms and input into the model. Four-layer Bi-LSTM is implemented. The proposed model achieved an average 0.724 QWK score on the ASAP dataset. Mayfield and Black[59] has proposed to implement a transformer-based model of the Bidirectional Encoder Representations from Transformers (BERT) in AES. The proposed method uses 20 percent

of the dataset as the validation set for loss measuring. The authors implement three cyclical learning rate curricula for fine-tuning until they reach a threshold. The lower bound learning rate is $(0.04 \times upperboundlearningrate)$, and the upper bound learning rate 0.00001. A halting criterion is used whenever the validation set has a 0.01 decrease in QWK. The halting criterion will return to the previous epoch and start training with one order of smaller magnitude learning rate. The untuned BERT model achieves 0.75 average QWK on the ASAP dataset, and the fine-tuned BERT model achieves 0.754 average QWK on the ASAP dataset. According to the author, they did not perform enough fine-tuning, which caused poorer performance than the other baseline AES models. Hence, Mayfield & Black have assumed that the BERT model could have better results if fine-tuned in a better manner. However, AES's fine-tuning task would be costly as the authors stated the time spent on training increases linearly with the number and document length of epochs.

Later on, Yang et al. [108] proposed a finely-tuned BERT model in AES with ranking and regression loss. The proposed method implements BERT to obtain all essays' vector representations into a score mapping function. The score mapping function is constrained by regression loss and ranking loss to obtain the final score. They obtained a 0.794 average QWK score using the ASAP dataset, which is better than most state-of-the-art AES methods. This proves the assumption made by Mayfield and Black [59] is correct, as the BERT can achieve a better result if fine-tuned.

Liao et al. [53] have proposed a hierarchical coherence model to measure the coherence between sentences in the essays. They proposed using a bi-linear tensor layer to obtain the coherence vector representation and semantic vector representations. The coherence vector representation measures the coherence using max-coherence pooling. The semantic vector representations aggregate the output of sentence layers. Then, the output of the bi-linear tensor layer will pass into the output layers for AES prediction. They managed to obtain a 0.786 average QWK score using the ASAP dataset. Also, it is proven that semantic attributes such as text coherence are crucial in AES.

Wang et al. [103] have introduced the collaboration of multiple essay representations for BERT that can be learned together. They implement one BERT model to obtain the document-level and token-level essay representation. Then, they concatenate both essay representations to input into a dense regression layer that predicts the score related to

the document and token levels. In another aspect, they implement the BERT model to get the last hidden state output and apply it to the LSTM model with an attention layer to get the segment-level representation. The segment-level representation is input into another dense regression layer to get the score related to the segment level. The final score is calculated based on document-level, token-level, and segment-level scores.

Ramesh and Sanampudi [76] have proposed a coherence-based AES model using sentence-based embedding and recurrent neural network. They implemented sentence-based embedding to encapsulate the essays' coherence and cohesion into vectors. LSTM and Bi-LSTM are implemented to determine the coherence in the connection of sentences and the essays' relevance to the prompt. The model achieved a 0.76 QWK score on average.

Overall, the neural network approaches are more accurate than feature-based approaches but have dataset size issues. In AES, there is no dataset of more than a size of 10k. Also, most neural network approaches are trained on word embedding at the word, sentence, or character level. This might ignore some critical factors influencing essay-scoring tasks, such as spelling errors and content detection. In addition, the word embedding method, such as word2vec, typically captures the word representative in a uni-direction. This will result in missing the actual semantic context in the essays, as a word might have different meanings in different contexts. Other than word embedding input, some neural network approaches implement bag-of-words (BoW) features to train the neural network. The BoW features ignore the relationship between words and the semantic meaning in a different context. Furthermore, the neural network approaches usually target extracting semantic attributes for the AES tasks.

2.1.3 Hybrid Approaches

The hybrid approaches are the approaches that use both machine learning algorithms and deep learning algorithms. These approaches are more explainable and better in performance compared to the others.

In 2016, Shehab et al. [87] proposed a hybrid approach of AES system based on neural network and feature engineering modules. The neural network performs the AES task to classify the score of the essays, and feature engineering modules mainly provide feedback

to the students. Before the neural network training, the essays will go through preprocessing steps such as spelling check, tokenization, stopword removal, and stemming. The neural network implemented by the authors is the Learning Vector Quantization that integrates competitive learning with supervision. The feature engineering modules use similar features compared to the RASP system [109] such as grammar errors, n-gram, and structure errors. However, their feature engineering modules do not have any semantic-specific features to provide feedback for students.

In 2018, Dasgupta et al. [23] proposed an enhanced recurrent convolution neural network architecture for the AES task. The model uses word-level and sentence-level representations inputs and a linguistic feature input vector based on handcrafted features. The word-level and sentence-level representations are captured from the GloVe word vector [70]. The linguistic feature input vector is based on existing feature engineering techniques similar to [72, 109]. Some examples of linguistic features generated are lexical diversity, part-of-speech, and sentence cohesion. One notable feature being proposed is the psychological feature derived from the Linguistic Information, and Word Count (LIWC) tool [94]. The feature counts words containing psychological meanings, such as emotionality and social relationship. The model achieved an average 0.786 QWK score on the ASAP dataset.

In 2019, Liu et al. [55] proposed a hybrid approach to the AES system based on two-stage learning. In the first stage, the model calculates the semantic, coherence, and prompt-relevant scores based on deep neural networks. Then, in the second stage, the work concatenates these scores with handcrafted lexical and grammatical features and inputs into a machine learning algorithm, XGBOOST, to perform the classification. The handcrafted features such as grammar error, essay length, sentence count, word count, and unique vocabulary are very similar to RASP [109] and EASE systems [72]. Their work obtained an average 0.773 QWK score on the ASAP dataset.

Later in 2021, Litman et al. [54] proposed a hybrid AES model with a combination of handcrafted features based on the rubrics of the essay evaluation and hierarchical self-attention model. They compared the hybrid model against the feature-based model using their proposed handcrafted features and deep learning neural network based on a co-attention-based neural network to evaluate fairness in AES. Their result detected that all AES systems have a slight bias, but the hybrid method has been evaluated as

the fairest AES model among the three. Also, a notable mention about the work is that one of the features proposed by the work captures the number of supporting points based on manually provided topics, which is similar to the proposed approach in this paper.

Also, in 2021, Sharma et al. [86] proposed a hybrid method of AES by combining Capsule Neural Networks, BERT-based text representation, and handcrafted features. The Capsule Neural Network and BERT-based text representation can capture the semantic attributes in the essays. However, the deep learning part cannot tell which semantic attributes are captured (e.g., coherence, organizational structure, content). The handcrafted features capture the fundamental grammatical and lexical features of the essays. The hybrid model obtained an average 0.81 QWK score, which is overall the best AES model in 2021. The result proves that the feature engineering method is still required to improve the current state of AES works.

Figure 2.1 summarizes the overall hybrid approaches over the years. Most of the proposed work will perform feature engineering to extract hand-crafted features from the essays. Then, using the features to obtain output through model training. Simultaneously, the essays will be converted into embedding, such as word embedding and character embedding to be input into a neural network for essay scoring. The proposed works will ensemble the output obtained from feature engineering and neural network through a dense layer for the hybrid element in the majority of the proposed hybrid approaches.

2.2 Review on the topic modeling techniques

One of the most critical essay scoring criteria is determining the concepts of the written essays. A human scoring the essays can understand this information clearly, but a machine cannot do that directly. To achieve this task in a machine, topic modeling is utilized. Topic modeling is a well-known statistical tool to extract latent features from large datasets, especially for text data [6]. A few examples of recent topic modeling applications in research are human-computer interaction trend studies [34], Artificial Intelligence (AI) investigation in marketing [65], and patient message studies for medical purpose [24].

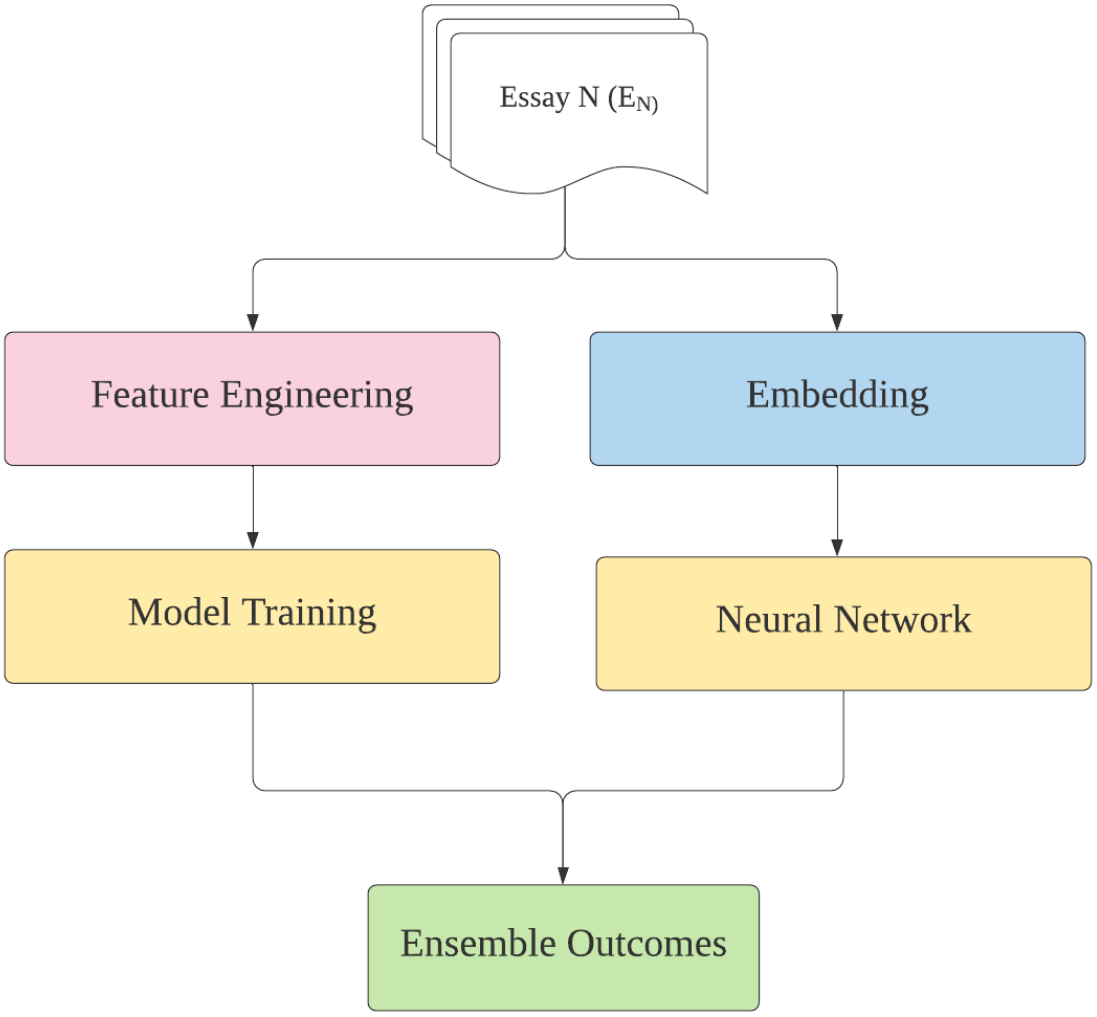


Figure 2.1: General Hybrid Approaches' architecture diagram

2.2.1 Background

The first topic modeling technique developed was proposed in 1983 for information retrieval by Salton [83]. This technique is based on the vocabulary of each document in the corpus. The count of occurrences of each unique term of each document is counted to form term frequency counts (tf). The count of a unique term over the corpus is counted to form inverse document frequency count (idf). Next, the two values of tf and idf will be normalized and compared to form a term-by-document matrix that has the tf-idf values for entire documents [83]. This enables the corpus to be reduced into a V-by-D (number of words in the vocabulary against the entire document) dimensional matrix and documents into a fixed-length vector containing real positive numbers. Despite this technique effectively identifying the set of terms that distinguish documents in a document collection, the reduction in the dimensional matrix was fairly negligible,

and the technique could not provide enough significant information for statistical correlation within or between documents [8]. Hence, Deerwester et al. have decided to develop another dimensionality reduction method for topic modeling based on Salton's work called latent semantic indexing or analysis (LSI/LSA) in 1990 [25]. LSI factorized the *tf-idf* dimensional matrix by using singular value decomposition (SVD). SVD will form three different matrices, including two unitary matrices of V-by-V and D-by-D and a V-by-D diagonal matrix containing non-negative real numbers. This operation results in matrices that reflect the document's text breakdown into linearly-independent vectors. Thus, the concepts in the corpus can be found using the matrices to identify the linear subspace [25]. Although LSA can provide significant data compression on a large corpus, it lacks a method to interpret or analyze the outcome [8]. Also, it required a huge size of documents and terms to obtain a better result. Hence, in 1999, Hofman [37] came up with an improved version of LSA called the probabilistic LSA model (PLSA), which removes the dimensionality methods and instead focuses on probabilistic modeling. Particularly, PLSA is based on the fundamentals of topic models, where each document contains a mixture of topics and a collection of terms.

$$P(D, W) = P(D) \sum_{Z} P(Z|D) P(W|Z) \tag{2.1}$$

The Equation 2.1 shows the joint probability of a document and word. The $P(Z|D)$ is the probability of topic Z in document D, which tells the possibility it appeared in documents. The $P(W|Z)$ is the probability of word W in the topic Z, which tells the possibility of finding W within that document D. From this, the concepts of a document can be converted into a probability distribution over a preset array of topics [37]. However, this technique lacks a method to determine the mixture ratio of documents. This resulted in the PLSA being incapable of identifying topics outside of the training set [8]. Nevertheless, these techniques marked the first probabilistic approach for topic modeling. The above techniques are the background of current topic modeling techniques such as Latent Dirichlet Allocation (LDA), Non-negative Matrix Factorization (NMF).

2.2.2 Existing Topic Model Techniques

Many topic modeling techniques are being developed over the years. This section describes some of the notable existing topic model techniques.

2.2.2.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a commonly implemented and the first few topic modeling methods proposed by Blei, Ng, and Jordan in 2003 [8]. This topic modeling method converts the data into three structure levels - word, topic, and document. LDA considers documents as random mixtures of hidden topics, treated as probability distributions of words. In LDA, it generates documents by sampling a K-vector θ, which indicates the mixture ratio of k topics from the Dirichlet prior distribution $p(\theta|\alpha)$. The k variable represents the dimension of the distribution, which also defines the topic variable z. With this, the number of the total topic will be calculated and input into the model. In LDA, it presumed the variables k and z to be static and known. Moreover, the matrix β of dimension $k \times V$ represent parameter of word probabilities such that $\beta_{ij} = p(w^j = 1|z^i = 1)$ where i equals to 0,1,...,K and j equals to 0,1,...,V. If $\theta_i >= 0$ and $\sum_{i=1}^{k} \theta_i = 1$, a k-dimensional of Dirichlet variable θ can hold values in the (k-1)-simplex, which contain a probability density on the simplex defined by the Equation 2.2.

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} ... \theta_k^{\alpha_k - 1} \tag{2.2}$$

The general graphical representation for the process of LDA topic modeling is depicted in Figure 2.2. The boxes represent repeated processes within the model, which the parameters M,N, and K indicating the number of repetitions. The white circle represents latent or hidden variables, and the grey-shaded circles represent visible variables. Lastly, the arrows represent the hierarchical flow of the process and the effect of one variable on another.

The α in Equation 2.2 and Figure 2.2 is one of the hyperparameters in LDA. It represents the document-topic density, which can be considered a prior number of occurrences of a topic seen in a document. The α parameter will affect the word combinations of topics,

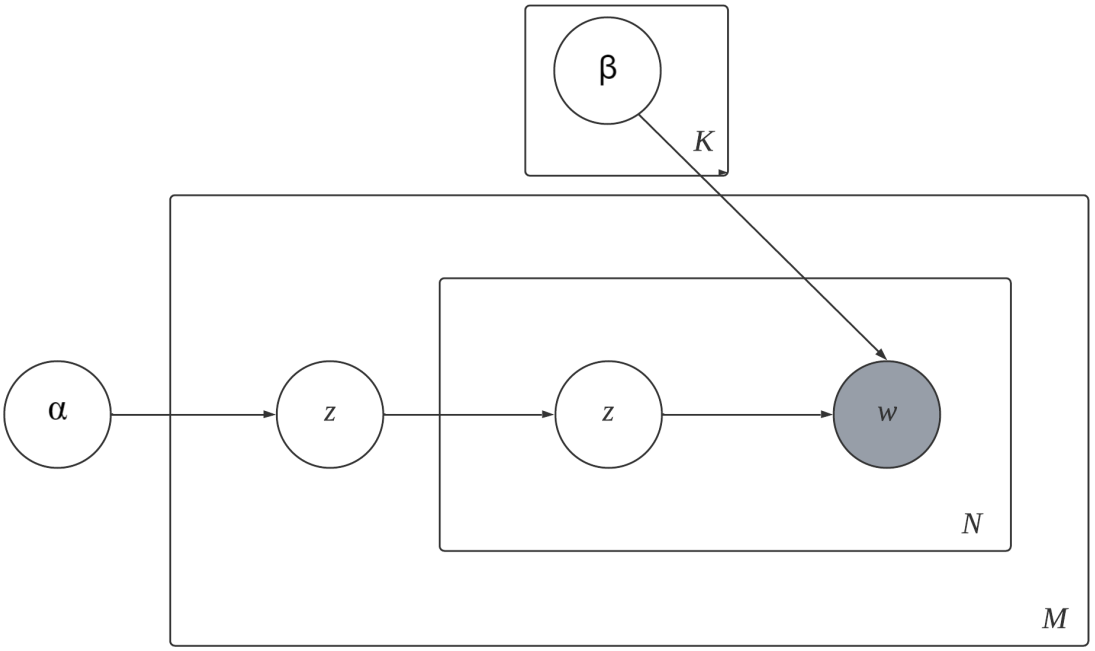


Figure 2.2: The LDA model's graphical representation diagram. The largest box, M, indicates documents, and the box, N, indicates topics and words within a document. The smallest box, K, represents sampling for each topic

which enables the "smoothing" process. The "smoothing" process is when the α value defined can cause topic modeling to avoid the extremities of a simplex [8]. Additionally, the hyperparameter β represents the topic-word density, which can be considered as the prior number of occurrences of words generated from a topic within the given corpus [89]. The value of β will affect the number of words composed within a topic [8].

The LDA topic modeling method has notably improved over the previous work, including Salton's method [83], LSA, PLSA [37]. However, it still has multiple limitations. First, the hyperparameters k, α, and β require additional steps of parameter fine-tuning to maximize their performance which could be costly and ineffective. Second, LDA has difficulty encountering a corpus with a vast vocabulary size [8]. Third, LDA treats all topics independently, ignoring the correlation between topics. However, in the case of topic detection for AES, the correlation between topics is not required as the current goal is to detect the concept or topics within the essay instead of the coherence. Four, LDA does not account for the order of words within a document. Similarly, the current objective of implementing topic modeling is to detect the concept or topics within the essay instead of the coherency score. Five, LDA ignores the event of word use change over time. This would not be an issue for AES, as every new set of essays will require a

new LDA model that ignores the change of words used over time.

To tackle the limitations, multiple extensions based on LDA have proposed [7, 9, 12, 36, 45, 51, 52, 77, 93, 100, 101, 104, 106, 112]. Some of them will be discussed in later sections.

2.2.2.2 Correlated Topic Model

To tackle the limitation of independent topics in LDA, the Correlated Topic Model (CTM) was proposed by the same author of LDA in 2006 [9]. Unlike topic modelling, the CTM focuses on sampling from a normal logistic distribution. This distribution allows for a layout of variation between the elements by converting a multivariate normal random variable to a simplex. The logistic normal was initially implemented to analyze compositional data, which inspired the author to implement it in a hierarchical model to describe the interrelationship between the latent or hidden topics [9].

The following equation defines the natural parameterization of the K-dimensional multinomial distribution:

$$p(z|\eta) = \exp\left(\eta^T z - \alpha(\eta)\right) \tag{2.3}$$

As shown in Equation 2.3, the random variable z can take on the K values. Also, the K-dimensional vector represents Z. The mean parameterization and the natural parameterization have the mapping function as the following equation:

$$\eta_i = \log \beta_i / \beta_k \tag{2.4}$$

The normal logistic distribution treats η as normally distributed and then maps to the mean parameterization through the inverse of Equation 2.4. Figure 2.3 shows the overall process of topic distribution generation, which is somewhat similar to the process of LDA, excluding the topics generated from the logistic normal.

Overall, the CTM can tackle one of the limitations of LDA by introducing the covariance matrix of logistic normal in the model, which enables it to consider the factor of correlations. It can better analyze a large corpus of documents by using covariance as

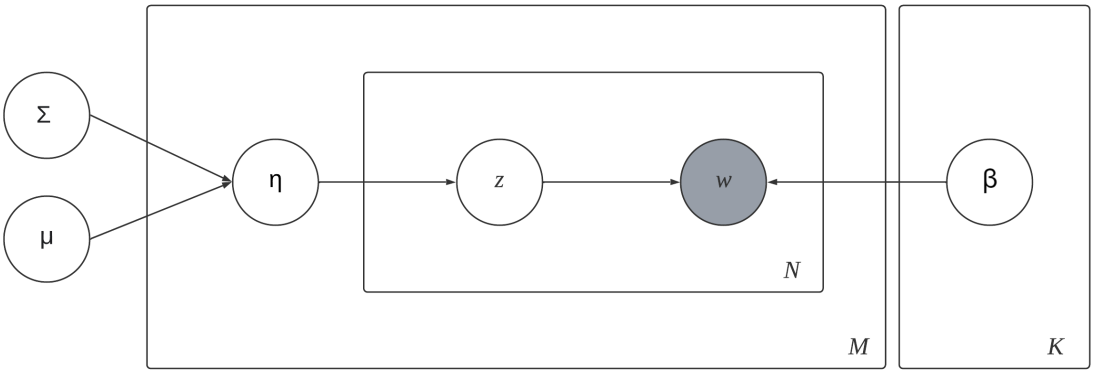


Figure 2.3: The CTM model's graphical representation diagram. The $\sum$ indicates the covariance, μ indicates mean.

an exploratory device. The CTM is better in performance for larger topic numbers k compared to LDA, where CTM has the best performance below 90 topic numbers and LDA has the best performance below 30 topic numbers. However, it has the main issue of determining the posterior inference. The author of CTM suggested implementing mean-field variational methods to construct an approximate distribution by minimizing the Kullback-Leibler (KL) divergence between the approximate and true posterior [9]. The LDA has a similar issue but has more options of approximate inference algorithms. Additionally, the CTM is limited to only detecting correlations between two topics, which might be ineffective for documents with complex topic correlations.

The CTM is unsuitable for AES as it commonly works best if the topic is large. The essays are written surrounding the same prompt, resulting in a relatively small topic size.

2.2.2.3 Dynamic Topic Model

The Dynamic Topic Model (DTM) is another proposed model based on LDA [7]. It primarily deals with the evolution of words over time, which is one of the LDA limitations. It trained upon a sequence of models with a defined time unit, t, which could range from days to years. To do that, the model chain the natural parameters of each topic in a space state model that extends from Gaussian noise, which the following function shows the simplest version:

$$\beta_{t,k}|\beta_{t-1,k} \sim N(\beta_{t-1,k}, \sigma^2 I) \tag{2.5}$$

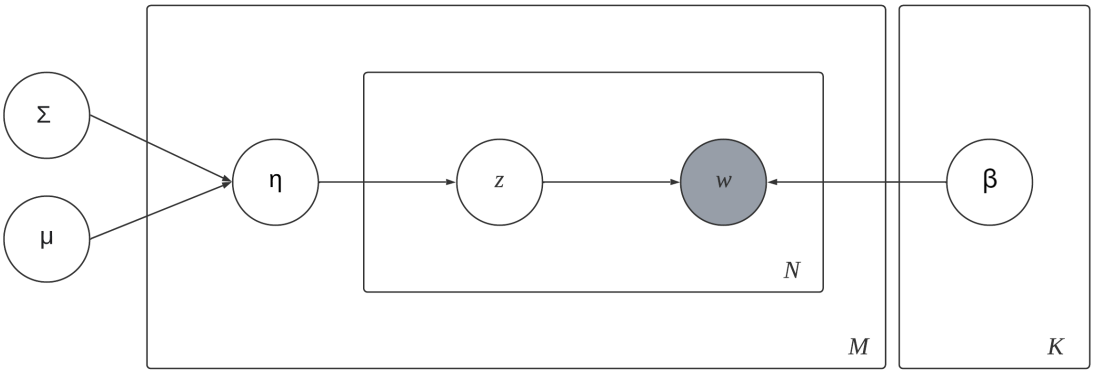


Figure 2.4: The DTM model's graphical representation diagram.

Unlike LDA, the DTM implements a logistic normal with a mean to convey uncertainty over topic proportions. A simple dynamic model is represented in the following equation:

$$\alpha_t|\alpha_{t-1} \sim N(\alpha_{t-1}, \delta^2 I) \tag{2.6}$$

DTM can chain a sequential topics models collection by constructing a connector between topic distributions and proportions. The general process of DTM is depicted in Figure 2.4.

The parameters in the diagram are similar to LDA. However, the distributions are Gaussian instead of Dirichlet. Additionally, the horizontal arrows represent the evolution of parameters between times, while the vertical arrows represent the process of each topic model in the sequence. Regarding limitations, the DTM has a similar issue of unmanageable posterior inference and hence needed calculation for approximate inference.

For AES, the DTM is not a suitable topic modeling as a set of essays are written simultaneously, which ignores the evolution of words over time.

2.2.2.4 Non-negative Matrix Factorization

Non-negative Matrix Factorization (NMF) is a different algorithm from LDA developed in 1999 [50]. This method has proven to work well in text corpus even though it has existed before LDA [99]. It takes advantage of the reduction in dimension for non-negative matrices [50]. It creates a corpus as a document-by-term matrix of the V-by-D dimension. NMF will try to find the weights of each component in the two k-dimensional non-negative matrices, W and H [99]. The rows of H represent the weights of every

unique word in the corpus, while the column of W represents each document's weights in relation to the topics.

This method randomly assigns the weights to the components of W and H over iterations to achieve the best performance. However, the random assignment of weights results in high inaccuracy and a lack of the ability to reproduce the same result.

2.2.2.5 Notable topic models

Tan et al. [93] proposed a topic model based on LDA called Foreground and Background LDA, which purifies topics of the foreground. Their method is designed to determine special topics and different components. [112] have proposed an LDA model based on a customized hashtag recommendation method that discovers hidden topics in blogs, named Hashtag-LDA. [38] have proposed an associated Bayesian model that can carry out two tasks (event segmentation and topic modeling) in a single framework. With that, they can obtain the event's topics and analyze the behavior of tweets on social media Twitter. [110] have proposed another topic model based on LDA, called as Conceptual Dynamic Latent Dirichlet Allocation (CDLDA) model, to track the topic detection for verbal communication. This model is designed to extract the dependencies between topics and verbal interactions. The CDLDA used hypernym information and verbal interactions to detect and track conversation topics. BERTopic [33] is a new topic modeling technique that produces the topic representations. BERTopic clusters the pre-trained language models' embedding, then extract the topic representations using the class-based *tf-idf*.

2.3 Review on the similarity algorithm

The similarity between the essay and prompt is a crucial element in essay scoring to determine if the essay is written surrounding the given prompt. In most cases, it is difficult to identify the similarities between texts because words and sentences may come up with different meanings based on the context. Most of the time, the similarity algorithm is calculated using a mathematical algorithm that assigns a real number between 0 and 1. A value zero in the similarity algorithm represents that the two texts differ. A value

one represents that the two texts are the same. Text similarity can be measured in three ways lexical, grammatical, and semantic.

2.3.0.1 Lexical Similarity

In lexical similarity, it considers two texts are similar if they have a similar character or word sequence. Hence, lexical similarities are measured by considering the given sequence of words and the composition of characters.

For character-based similarity algorithms, there are different types of variations. First, the n-gram matching similarity algorithm [60] is measured by comparing the n-grams of each character in two texts. The similarity score is calculated by dividing the count of similar n-grams by the total number of n-grams. The N-gram matching similarity algorithm does not work well in most cases as it does not consider the semantic attributes of the texts [3]. However, n-gram matching can be implemented at word and phrase-level by using different sizes of n-grams such as unigram, bigram, and trigram. Jaro distance similarity algorithm calculates the number or sequence of the characters between two strings that are the same [43]. Hence, this method is mostly implemented for database records linkage. Jaro-Winkler is an extension based on Jaro that allows to adjust of similarity weights [105]. The Needleman-Wunsch similarity algorithm is one of the first few dynamic programming algorithms to find the similarity between biological sequences. It splits an entire sequence of characters into a series of smaller character sequences to find the best alignment over the entire two sequences, which is only suitable when the two comparison texts are similar in length [67].

For term-based similarity algorithms, there are also different types of variations available. One of the most implemented term-based similarity algorithms is the cosine similarity, which is measured by calculating the cosine angle between the two texts in the vector space [26]. Besides that, Manhattan distance, also known as Block distance, is measured by the sum of the absolute differences between the word vectors of two texts. To obtain a similarity score through Manhattan distance, it is required to implement one-hot coding [26]. Jaccard similarity [41] is measured by the number of similar terms divided by the total unique terms in both texts. Dice's coefficient [27] is measured by double the similar terms in both texts and divided by the number of the total terms in both strings. The overlap coefficient is similar to the Dice coefficient, which measures the

overlap terms between two texts. The matching coefficient is a simple method that counts the occurrence of similar terms by dividing them by total terms.

N-gram Matching coefficient is one of the most implemented similarity algorithms in the AES domain [1, 42, 72, 82, 86]. However, lexical similarity ignores the semantic attributes of the strings, which lead to lower performance if the comparison texts are semantically similar but different in words.

2.3.0.2 Grammatical Similarity

The grammatical similarity is measured based on the organization or the structure of words/sentences. In most cases, the grammatical similarity is combined with either a lexical or semantic similarity algorithm to measure the similarity between two texts.

Part-of-speech tags represent the word functions in a sentence. Vuk & Dragan [4] proposed to implement part-of-speech as a weighting scheme on a bag-of-words approach, allowing it to compare structure information between the comparison texts. Also, a parse tree can be implemented to find similarities by searching similar sub-trees in terms of POS tags, and terms used [32, 63]. However, the parse tree is significantly slower in the calculation time compared to the other similarity algorithm [32]. Similar to Vuk & Dragon's algorithm, Yang et al. [107] measure the similarity by combining a parse tree and bag-of-word approach to retrieve the semantic representation and the grammatical information of the texts.

2.3.0.3 Semantic Similarity

Semantic similarity has the most variations as it focuses on capturing and comparing contextual meanings between texts. There are three main directions to calculate semantic similarity: knowledge base, corpus-based, and hybrid.

Knowledge base similarity is measured based on predefined information from multiple resources such as encyclopedias, thesauri, and dictionaries. These resources are provided with structured information about the words. WordNet is one of the most widely implemented knowledge-based resources for measuring the similarity between texts. It can provide the synonyms of words together with part-of-speech tagging. Using the

synonym information, lexical similarity algorithms can be implemented to measure the shortest distance between the texts [72, 86].

Corpus-based similarity computes the similarity of texts based on information from corpora. A corpus is an extensive collection of text that stores computer knowledge to understand the texts. Latent Semantic Analysis (LSA) is a popular bag-of-words technique to create a corpus to perform corpus-based similarity. In LSA, words with similar meanings are assumed to occur in a similar group. In addition, embedding is one type of corpus that captures texts' lexical, grammatical, and semantic attributes, represented in a high-dimensional vector. Embedding is flexible and powerful as it can be trained and re-implemented across different models. There are two main types of embedding: word embeddings and sentence embeddings. The word embedding contains grammatical and semantic information defined from the unlabeled dataset as a vector. Word2vec is a word embedding method to compute the similarity between texts. It has two pre-trained models: the continuous bag of words (CBOW) and Skip-gram [79]. The CBOW determines the weights of the current word depending on the context, and the skip-gram determines the weights of related words given the current words. Some other examples of word embedding algorithms include Global Vectors for Word Representation (GloVe) [70]and Embeddings for Language Models (ELMo). The sentence embedding converts a sentence input into a fixed-length vector representing lexical, grammatical, and semantic information [114]. The current state-of-the-art sentence embedding is mainly from pretrained language models such as BERT.

2.3.1 Similarity approaches for AES

There have been multiple similar approaches implemented in AES. The similarity approaches in AES are implemented in three ways:

- To measure the similarity score between the student essay and sample essay or model answer
- To measure the similarity score between two points (e.g. sentences, paragraphs) in the essays
- To measure the similarity score between the student essay and the given prompt text

In the first method, the essays are directly compared to sample essays. The second method is to measure the coherence of the essays. The third method determines if the essay is written surrounding the prompt. The details of similarity approaches in AES will be discussed below.

2.3.1.1 N-gram

N-gram at character, word, and phrase level is one of the most popular similarity approaches for AES purposes. The n-gram is a contiguous sequence of n components from a given sequence. N-gram models generate the similarity score by comparing n-grams between essays, sample essays, or assigned prompt texts. It can be as simple as n-gram matching to measure the probability of a given word that appeared in another set of words. Otherwise, it can be converted into a vector space and input into similarity algorithms such as Jaro, Manhattan distance, and Cosine Similarity to obtain the similarity score. These studies by [1, 17, 30, 40, 42, 69, 72, 82, 86] have implemented n-gram to determine the similarity score between essays and sample essays or given prompt texts. However, the n-gram-based similarity models do not consider the semantic attributes of the strings due to the context, which often leads to a miscalculation in similarity. For example, the word "jam" might have different meanings in a different context, such as "traffic jam" or "strawberry jam".

2.3.1.2 LSA

LSA in AES is commonly implemented to score essays by measuring the similarity between two sequences of texts. The LSA will convert both sequences of texts into vectors and calculate the cosine angle to generate the similarity score. The studies implemented LSA to determine the similarity score and use it as a feature for AES, including [1, 21, 31, 35, 40, 81, 85]. However, LSA ignores the semantic attributes similar to n-gram as it is based on the bag-of-words approach, which does not take account of the order of the words in a sentence. Additionally, it lacks a method to interpret its result [8].

2.3.1.3 Pretrained embeddings

The pretrained embeddings are encoded words or sentences in real number dense vectors. Similarity algorithms such as cosine similarity are commonly implemented on top of pretrained embeddings to obtain the similarity score. However, many variations of pretrained embeddings are being implemented in the AES studies. Some of them will be discussed below.

Saha & Rao implemented a pretrained sentence embedding, InferSent, to measure the similarity score between the sentences in the sample essay and sentences in the students' essay [81]. InferSent is a sentence embedding approach that generates a semantic sentence representation from sentence text [15]. It is based on bi-directional LSTM to construct a fixed-length vector using mean-max pooling.

Huang et al. [39] proposed measuring the similarity score between the essay and the given prompt using noun phrase vectors based on a hybrid embedding. Hybrid embedding combines distributional semantic embeddings (word2vec and GloVe) and structural embedding (ConceptNet). The authors construct the hybrid embedding by only using synonyms and synonymous noun phrases in the essays. Their approach improved significantly compared to WordNet and Word2Vec in terms of off-topic essay detection.

Other AES studies that implemented pretrained embeddings include [18, 61, 81]. Overall, the pretrained embeddings in AES have better results than the n-gram and LSA approaches to measuring the similarity score. However, the static non-contextual pretrained embeddings such as word2vec and GloVe cannot identify ambiguous words that have multiple meanings [74].

2.4 Summary

To summarise, a bar chart about the publication count on AES over the years is plotted in Figure 2.5. Before 2015, there were only feature-based approaches proposed. After that, neural networking started to be implemented in AES models. The neural network approaches have the most count within 2015-2020. Moving to the year 2021, the trend is starting towards hybrid models, which shows the direction of this research is on the trend.

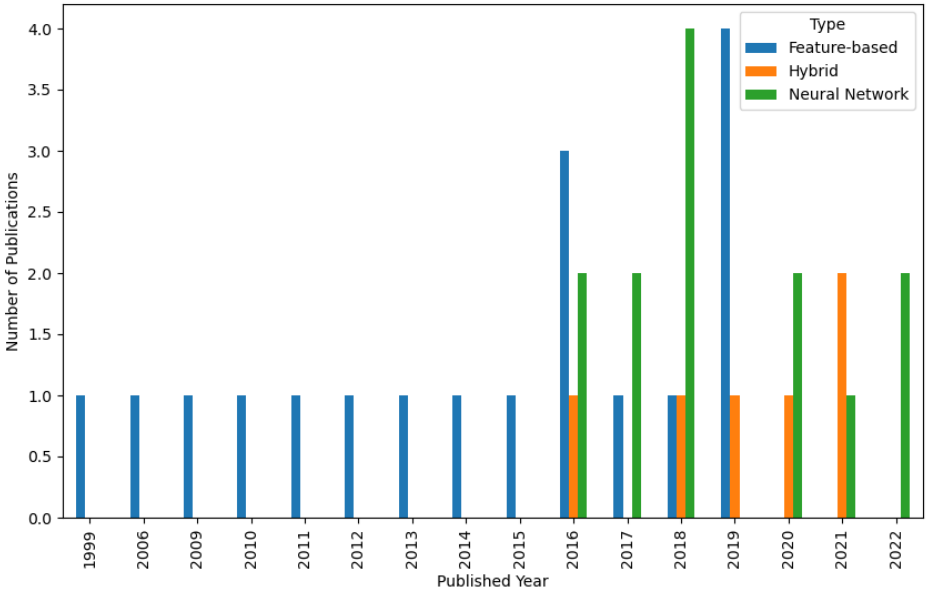


Figure 2.5: Publication Count over the Years

Table 2.1 shows that only 5 out of 37 proposed AES models perform content detection in the works. Content detection will help identify how many main contents in the essays are being told to support the prompt questions. None of the content detection implemented is implemented in the essays, but course materials are provided by examinations. Hence, this would not be practical as English examinations do not have content related to the course materials.

Table 2.1: Table summary of reviewed AES papers

Research	Approach	Content Detection	Prompt Similarity
Foltz et al. 1999 [31]	Feature-based	Yes	No
Attali and Burstein 2006 [2]	Feature-based	No	No
Mohler and Mihalcea 2009 [63]	Feature-based	No	No
Islam and Hoque 2010 [40]	Feature-based	No	No
Yannakoudakis et al. 2011 [109]	Feature-based	No	No
Hastings et al. 2012 [35]	Feature-based	No	No
Persing and Ng 2013 [71]	Feature-based	No	No
Adamson et al. 2014 [1]	Feature-based	No	No
Phandi et al. 2015 [72]	Feature-based	No	Yes
Cummins et al. 2016 [19]	Feature-based	No	No
Dong and Zhang 2016 [28]	Neural Network	Yes	No
Shehab et al. 2016 [87]	Hybrid	No	No
Taghipour and Ng 2016 [91]	Neural Network	No	No
Sendra et al. 2016 [85]	Feature-based	No	No
Oduntan et al. 2016 [69]	Feature-based	No	No
Dong et al. 2017 [29]	Neural Network	Yes	No

Zhao et al. 2017 [113]	Neural Network	No	No
Fauzi et al. 2017 [30]	Feature-based	No	No
Contreras et al. 2018 [17]	Feature-based	No	No
Dasgupta et al. 2018 [23]	Hybrid	No	No
Wang et al. 2018 [102]	Neural Network	No	No
Huang et al. 2018 [39]	Neural Network	No	Yes
Mesgar and Strube 2018 [61]	Neural Network	No	No
Cozma et al. 2018 [18]	Neural Network	No	No
Saha and Rao CH 2019 [81]	Feature-based	No	No
Darwish and Mohamed 2019 [21]	Feature-based	No	No
Liu et al. 2019 [55]	Hybrid	No	No
Salim et al. 2019 [82]	Feature-based	No	No
Janda et al. 2019 [42]	Feature-based	Yes	Yes
Beseiso and Alzahrani 2020 [5]	Hybrid	No	No
Mayfield and Black 2020 [59]	Neural Network	No	No
Yang et al. 2020[108]	Neural Network	No	No
Liao et al. 2021 [53]	Neural Network	No	No
Litman et al. 2021 [54]	Hybrid	Yes	No
Sharma et al. 2021 [86]	Hybrid	No	No
Wang et al. 2022 [103]	Neural Network	No	No
Ramesh and Sanampudi 2022 [76]	Neural Network	No	Yes
The number of studies work on:		**5**	**4**
The total number of studies in the table:			**37**

Similarly, Table 2.1 shows that the prompt similarity is only implemented in 4 proposed AES models out of 37. The prompt similarity will provide a score to indicate whether the essay is relevant to the given prompt. The current state of prompt similarity score algorithms is mainly n-gram matching methods. The n-gram matching methods cannot deal with words containing multiple meanings or phrases. In English, a combination of words might have different meanings as well.

Among all the topic model methods, filters are constructed to find the most suitable topic model for AES. Figure 2.6 describes the filters' flow to select the most suitable topic model. First, topic models that only work for short text are filtered out as the essays are at least 100 words. Second, the topic models that consider the correlation between topics are excluded. Implementing LDA in this research aims to detect the concepts or topics in the essays instead of the coherence between words or sentences. Additionally, the topic models that take account of the evolution over time are excluded as the essay sets are written at the same period. Lastly, the designed topic models specified for other tasks are omitted. These filters consider the LDA the most suitable

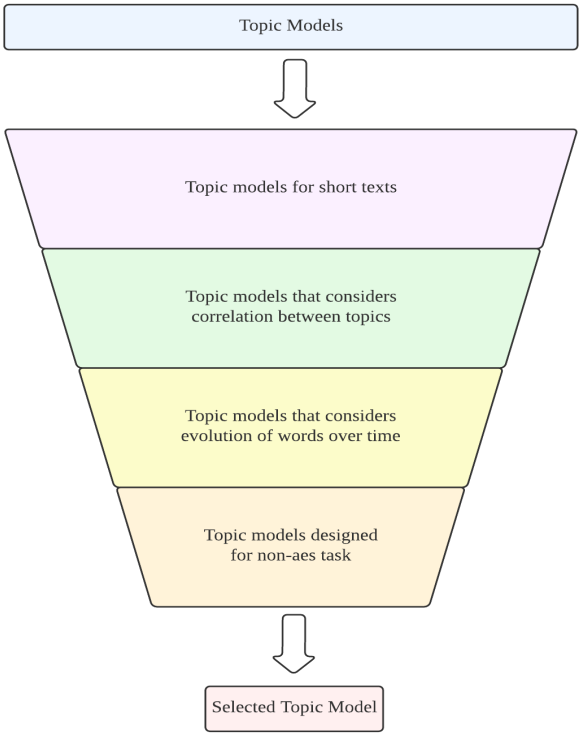


Figure 2.6: The filters to select the most suitable topic model for AES.

topic model for AES. An experiment of BERTopic on AES model was made to compare the performance is tabulated in Table A.1.

Table 2.2 summarizes the similarity methods implemented in AES. The n-gram approach is the most implemented as it is very low demanding in cost and resources and easy to be implemented. However, the n-gram methods do not consider the contextual meaning of words. Similarly, LSA and pretrained embeddings (Word2Vec and GloVe) have the same issue by not considering the sequence of the words in the calculation, which ignores the contextual meaning of words. Additionally, no studies have applied the embeddings from pre-trained models such as BERT and Elmo for similarity measures. The pre-trained models' embeddings are very demanding regarding resources, but they covered the issues in n-gram, LSA, and pretrained embeddings. The pre-trained models' embeddings are contextualized, allowing it to determine the contextual meaning of words across different contexts [56]. Hence, the proposed work will implement the pre-trained model's embeddings to perform the similarity scoring task.

Table 2.2: Summary of reviewed similarity measures in AES.

Approaches implemented	Works	Total number of works
N-gram	[1, 17, 30, 40, 42, 69, 72, 82, 86]	9
LSA	[1, 21, 31, 35, 40, 81, 85]	7
Pretrained embedding	[18, 39, 61, 76, 81]	5
The total number of works in the table:		**21**

Chapter 3

Research Framework

Figure 3.1 shows the architecture framework of our proposed methodology. The work is split into three phases according to RO1, RO2, and RO3.

Section 3.1 describes the framework of the RO1 phase. In essay writing, an essay contains many topics surrounding the prompt. In the RO1 phase, the essays of a prompt (E_N) are input into topic modeling algorithms to find the top N topics (T_N) and their related word tokens. Then, the topic model is implemented to determine the content or topics of each essay. Each essay's topic vectors (V_N) are generated based on the topic model to achieve RO1. An evaluation is performed to determine the effectiveness of the topic vectors.

Section 3.2 describes the framework of the RO2 phase. In the RO2 phase, the similarity score of the sentences is measured against the prompt to identify if the sentence is related to the prompt. Each essay is tokenized into sentence level and converted into the transformer's model embedding. The similarity scores between sentences and prompts are measured by calculating the cosine similarity between sentence embedding and prompt embedding. Feature engineering is performed to obtain features out of the similarity scores obtained. An evaluation is performed to determine the effectiveness of the new sentence similarity features.

Section 3.3 describes the framework of the RO3 phase. Moving to the RO3 phase, the fine-tuning of the transformer model, RoBERTa, is performed to obtain the prediction scores ranging from 0 to 1. Then, the existing grammatical and lexical features are concatenated with the new semantic features generated from RO1 and RO2, and the

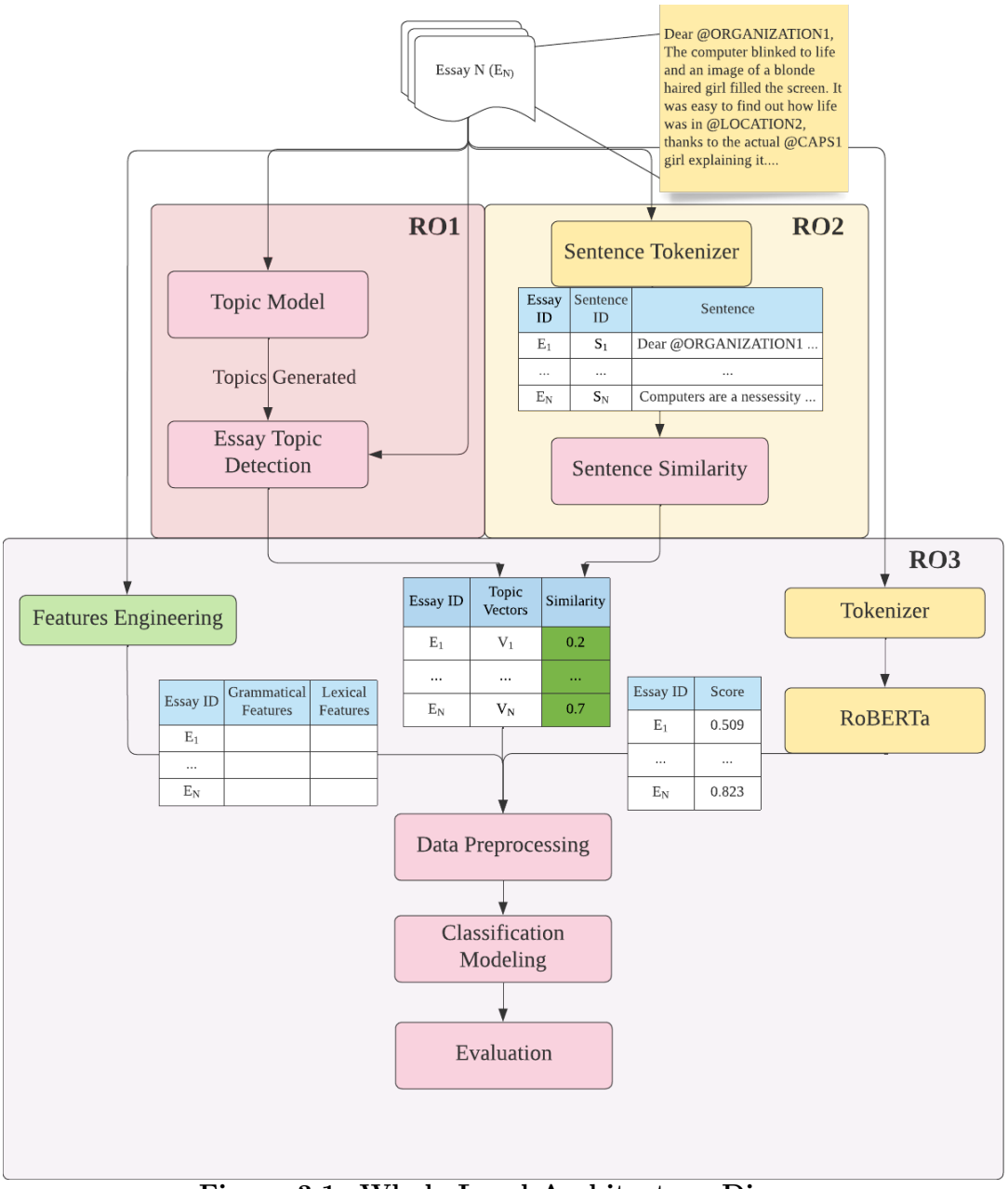


Figure 3.1: Whole Level Architecture Diagram

prediction scores from the transformer model into a final feature set. Subsequently, the final feature set is taken into preprocessing steps. The preprocessed data is input into regression machine learning algorithms to obtain the scores of essays ranging from 0 to 1. Finally, the essay scores are rescaled back into the original score range and evaluated using the Quadratic Cohen's Kappa (QWK) score.

Section 3.4 summarizes the proposed AES model framework.

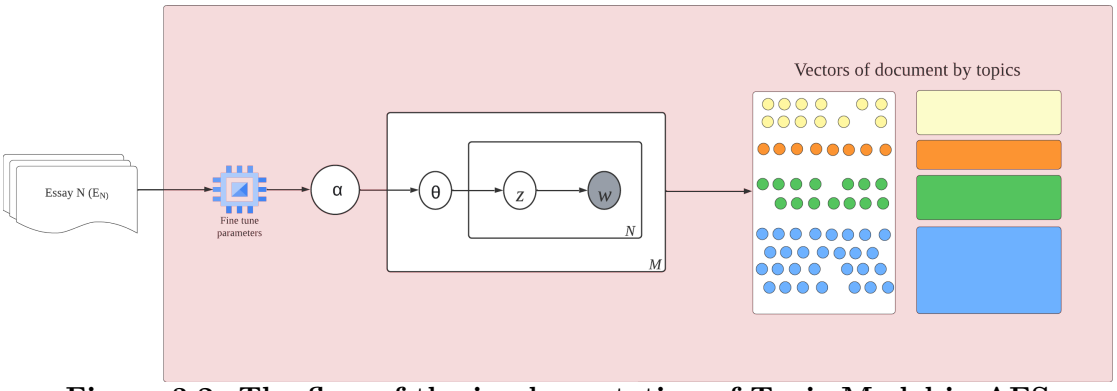


Figure 3.2: The flow of the implementation of Topic Model in AES

3.1 Topic Model - Latent Dirichlet Allocation

Figure 3.2 describes the overall flow of the implementation of the topic model in AES. First, the work has fine-tuned the crucial hyperparameters of LDA topic models, topic number and alpha. Then, the LDA model is trained with the essays using the parameters found. The proposed work presumes that the topics detected by the topic model will be the right supporting ideas for the essay prompt. Hence, the work implements the trained topic models to obtain each essay's topic vectors (topic representations of supporting ideas).

3.1.1 Data Preprocessing for Topic Model

Multiple data preprocessing steps are performed before the topic model training. First, words containing '@' are removed to avoid hidden entities in the data. Second, single quotes and empty spaces are removed to prevent empty n-grams. Third, bi-gram and tri-gram are constructed to train the topic models to include phrases in training. Fourth, a lemmatization filter is added to keep nouns, adjectives, verbs, and adverbs using the python library spaCy.

3.1.2 Fine-tuning of LDA

The research selected the most suitable topic model algorithms, Latent Dirichlet Allocation (LDA) [8] for AES fields based on the study made across different methods. Three crucial parameters to be fine-tuned in LDA are topic number (k), alpha α and beta β. The topic number is the number of topics determined by the LDA Topic Model. α is a

parameter limiting the prior distribution over topic weights in each document. β is the parameter for the prior distribution over word weights in each topic.

The usual rule to fine-tune the hyperparameter α is to implement $\alpha < 1$, which will help the model to determine the modes of the Dirichlet distribution and having a bias toward sparsity [8]. Typically, a paragraph should have one topic sentence that contains the main idea and the other sentences as the sub-topics around the main idea [20]. Using that logic, an essay with 500 words will contain an average of 3 to 5 paragraphs, which indirectly leads to 3 to 5 main topics, and each topic will contain an average of 3 subtopics. Hence, an essay below 500 words will likely have at most 15 topics.

To determine the optimum topic number (k) and α, a grid search will be executed using topic numbers of 5 to 50 and alpha of 0.01, 0.05, 0.1, 0.2, 0.5, and 1. The topic number is set up to 50 to evaluate whether the hypothesis is valid. For beta, we follow the suggestion by the author of LDA to keep parameter β of 1 over the topic number, $\frac{1}{k}$. The evaluation of the grid search will be based on the C_v coherence score. C_v coherence score constructs vectors of words based on the word occurrences [98]. The vectors calculate the score by implementing Normalized Pointwise Mutual Information (NPMI) and cosine similarity. The C_v coherence score is selected to be the evaluation measure of LDA in grid search because it is known and proven to measure the most significant correlation to human interpretability [78, 90]. Additionally, it is often being implemented to measure the performance of LDA [44, 64, 66, 73, 98, 111].

3.2 Sentence similarity toward prompt

The research proposes a new feature, sentence similarity, in relation to a given prompt using a transformer model's embedding. Unlike the n-gram, LSA, and pretrained embedding methods, the transformer model's embedding considers the contextual meaning of sentences in a different context, which is hypothesized to provide better performance. Additionally, the calculation of prompt similarity at the sentence level is proposed to deal with the unfair comparison between essay and prompt with different lengths. Figure 3.3 shows the flow of the implementation of sentence similarity toward prompt using the transformer model. First, essays are tokenized to the sentence level. Then, each sentence and the prompt will be input into the transformer model to obtain the

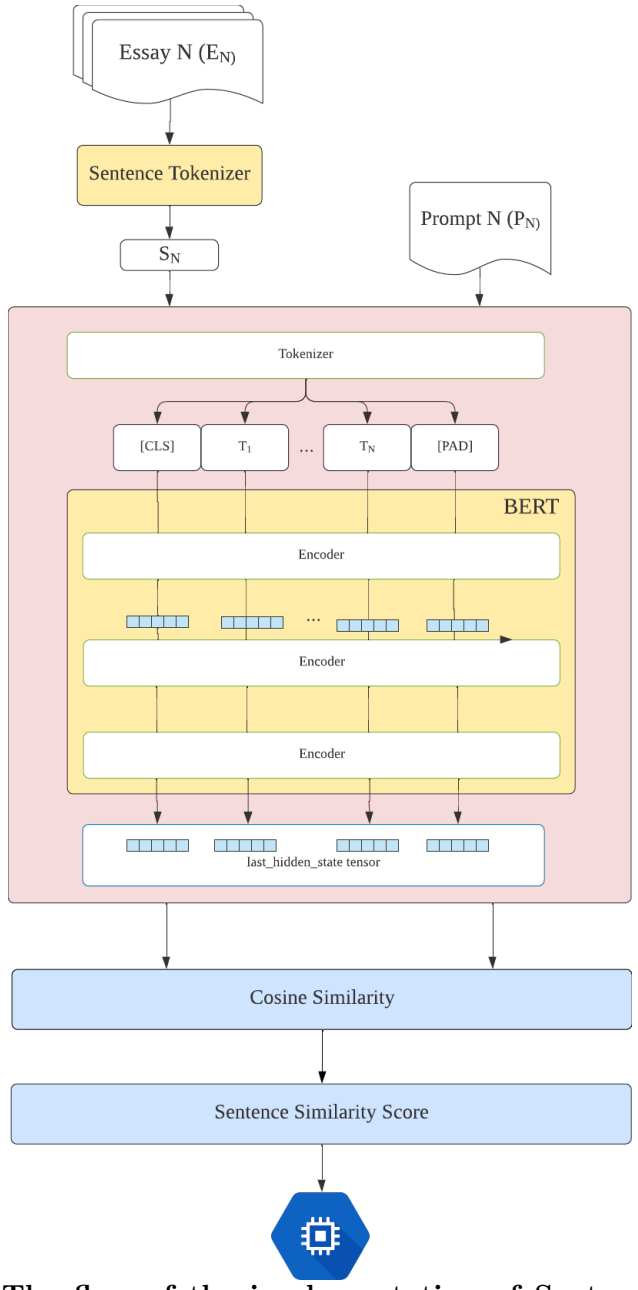


Figure 3.3: The flow of the implementation of Sentence Similarity toward Prompt

last_hidden_tensor. The tensors will be implemented to calculate the similarity between sentence and prompt. Finally, feature engineering is performed to convert the sentence similarity scores into the same dimension as the other existing features to allow the training of the AES model. The details of each step are further explained in the subsequent subsections.

Table 3.1: Sentence similarity features descriptions.

Feature names	Type	Description
number of good similarity sentences	Count	the count of sentences that have similarity score **more** than mean similarity scores of whole essay sets' sentences
number of bad similarity sentences	Count	the count of sentences that have similarity score **less** than mean similarity scores of whole essay sets' sentences
good similarity score	Score	the average of sentence similarity scores that have score **more** than mean similarity scores of whole essay sets' sentences
bad similarity score	Score	the average of sentence similarity scores that have score **less** than mean similarity scores of whole essay sets' sentences
mean sentence similarity	Average	the average sentence similarity scores of each essay

3.2.1 Transformer model to sentence similarity toward prompt

The research implements the bert-base-uncased transformer model to measure the similarity by using its ability to embed the semantic representations of the sentences or essays into dense vectors. Using the vectors, the respective similarity is calculated using cosine similarity. For each essay, the essay will be tokenized into sentences, then further tokenized into tokens to input into BERT. Each encoder layer in BERT will result in a set of dense vectors with 512×768 size from the inputs. The research will be using the last_hidden_state tensors of the BERT to measure the sentence similarity by converting the tensor into vector size 768 using the mean pooling method. The mean pooling method will measure the mean of the token embedding and convert them into a 768-size vector. Then, it can be used to measure sentence similarity through cosine similarity between the vectors of the sentences and prompt.

3.2.2 The feature engineering of sentence similarity

The research performs feature engineering to convert sentence similarity scores of each essay into the same dimension of the other existing features to allow the training of machine learning algorithms. The features engineering performed includes the number of good similarity sentences, number of bad similarity sentences, good similarity score, bad similarity score, and mean sentence similarity. Table 3.1 describes the calculation of the sentence similarity features. These features represent how relevant each essay's sentences are to the prompt.

Table 3.2: Features based on the existing works.

Feature Type	Feature Description
Lexical	Count of characters Count of words Count of commas Count of apostrophes Count of sentences Count of punctuation Average word length Count of unstemmed n-grams (unigrams and bigrams) Count of effective stemmed n-grams (unigrams and bigrams)
Grammatical	Count of correct POS n-grams Ratio of correct POS n-grams over total words count Count of spelling errors Count of grammatical errors
Semantic	Prompt words count Ratio of prompt words count over total words count Synonym of prompt words count Ratio of synonym of prompt words count over total words count

3.2.3 Existing Feature Engineering

Feature engineering still plays a vital role in the AES model, disregarding the booming of the transformer model and deep learning in the natural language processing field [55, 86]. There are three important elements in essay writing which are grammatical, lexical, and semantic attributes [88, 92]. The grammatical features extract information related to the structural rules of the components of paragraphs, phrases, and words. The lexical features extract information related to the vocabulary, words used, or morphemes. The semantic features extract information related to the ideas, context, and meaning of words or phrases. The features are also beneficial in the AES field as they can provide feedback to the students regarding the evaluation. Table 3.2 describes the grammatical, lexical, and semantic features based on the existing works [72, 86]. The features will be implemented with the new semantic features proposed in RO1 and RO2 to achieve RO3.

3.3 Hybrid modeling

The proposed hybrid AES model concatenates the topic vectors feature from RO1, the sentence similarity features from RO2, prediction scores of fine-tuned RoBERTa model from RO3 and existing grammatical and lexical features to form the final feature set. The feature set will be taken to preprocessing step.

For data preprocessing, the research scales the new features (topic vectors and sentence similarity features), the output of RoBERTa, and existing features (grammatical and lexical features) [72, 86] to 0 to 1. To perform the regression prediction, each essay's

score will be scaled to a range of 0 to 1. Then, it will re-scale back to the essay set score range after the predictions made by regression models.

3.3.1 RoBERTa fine-tuning

Currently, there are several variants of the transformer model have been released. For the experiments, the research implements the RoBERTa base model [57]. The optimal hyperparameter values for fine-tuning are task-specific; hence, different learning rates are being tested. The model is trained on 4 for batch size, 1e-5 of the learning rate, and fine-tuned over five epochs.

3.4 Summary

The research proposes a hybrid approach to the AES model by combining the feature-based model with a neural network-based model. The features of essay topic detection, sentence similarity, and output of the transformer model will be concatenated with the existing features from the state-of-the-art AES model as the input features of a feature-based AES model. Then, the SVM regressor and BLRR are implemented on top of the features to calculate the final prediction score.

Chapter 4

Experiment Setup

This chapter describes the experimental setup being implemented across the research experiments. The experiments were conducted on a machine with 32GB memory and NVIDIA 2070 GPU.

Section 4.1 describes the benchmark dataset implemented in the research experiments.

Section 4.2 presents the feature engineering methods implemented in the research experiments.

Section 4.3 presents the learning algorithms that are implemented to predict the final outcome of the experiments.

Section 4.4 presents the performance metric implemented to evaluate the results from the research experiments.

Section 4.5 summarizes the experiments flow that is presented in the following few chapters.

4.1 Benchmark Dataset

The research will be using the dataset from Automated Student Assessment Prize (ASAP) Competition [1]. This dataset includes eight essay sets with different prompts of different genres (narrative, argumentative, or response). The dataset was the essays

[1] ASAP dataset http://www.kaggle.com/c/asap-aes/data

Table 4.1: ASAP dataset details.

Essay Set	Set size	Genre	Average Length	Score range
1	1783	Argumentative	350	2 to 12
2	1800	Argumentative	350	1 to 6
3	1726	Response	150	0 to 3
4	1772	Response	150	0 to 3
5	1805	Response	150	0 to 3
6	1800	Response	150	0 to 3
7	1569	Narrative	250	0 to 30
8	723	Narrative	650	0 to 60

written by students ranging from grade 7 to grade 10, which were evaluated and scored by at least two human graders. Each essay set has different characteristics refer to Table 4.1.

The essay sets 1 and 2 are argumentative essays with a similar set size and average length. The essay sets 3, 4, 5, and 6 are response essays with a similar set size, a relatively shorter average length of 150, and the smallest score range from 0 to 3. Unlike other essay sets, essays sets 7 and 8 do not have similar characteristics of set size, average length nor score range. The vast range of scores in essay sets 7 and 8 will be an issue since not every score in the range has enough dataset to support the learning of the AES models. Essay set 8 will be the essay set with the worst data quality as it has the smallest set size and the most extensive range, resulting in nulls for multiple scores in the dataset. A research work has been proposed and written in a paper regarding data quality in the ASAP dataset, which has already been published at a conference.

The research implemented 5-fold cross-validation on the ASAP training dataset for evaluation. Each of the essay sets is split randomly into five folds. Since the transformer model required a validation set, three folds will be the training set, one fold will be the validation set, and the last fold will be the test dataset. Hence, 20 runs of evaluation will be executed for each set.

4.2 Feature engineering

For experiments comparing with the base model [72], the proposed new features will replace the existing semantic features and be trained in an AES model with the existing grammatical and lexical features.

4.3 Learning algorithms

For experiments comparing with the base model [72], SVM regressor and BLRR are implemented. For final hybrid experiments, *StackingRegressor* from *sklearn* python library is implemented to stack the two models together to form the final outcome. Stacking regression is an ensemble learning technique to combine multiple regression models via a meta-regressor.

4.4 Performance Metric

Two performance metrics are proposed in the following subsections to evaluate the proposed models' performance.

4.4.1 Quadratic Weighted Kappa

Quadratic Weighted Kappa (QWK) [96] is proposed to measure the performance of the proposed framework on the AES dataset. It is an extension of Cohen's Kappa [14], a well-known statistical measure used to evaluate agreement in categorical or nominal data. Cohen's Kappa considers observed and expected agreement, providing a more comprehensive assessment beyond simple agreement percentages. The measure is particularly valuable when the data involves multiple categories, or the chance agreement is likely.

Cohen's Kappa is the ratio of the observed agreement beyond chance to the maximum possible agreement beyond chance. The equation for Cohen's Kappa is as follows:

$$k = \frac{P_o - P_e}{1 - P_e} \tag{4.1}$$

In the equation, P_o represents the proportion of observed agreement, catching the actual level of agreement between the raters, while P_e represents the proportion of agreement expected to occur by chance alone, providing a baseline measure of agreement. The resulting value of Cohen's Kappa ranges from -1 to 1, with 1 indicating perfect agreement, 0 representing agreement that is no better than chance, and -1 indicating complete disagreement.

However, despite its usefulness, Cohen's Kappa has certain limitations when capturing the severity of disagreement in categorical data. In certain scenarios, not all misclassifications carry the same weight, and the magnitude of disagreement becomes crucial. To address this limitation, Quadratic Weighted Cohen's Kappa (QWK) was developed as an extension of Cohen's Kappa.

QWK addresses the shortcomings of Cohen's Kappa by incorporating weighted values that account for varying degrees of disagreement. These weights are often derived from the hierarchical or ordinal nature of the assessed categories.Considering the severity of misclassifications, QWK provides a more reflective and informative measure of agreement. The equation of QWK is as follows:

$$k_{qw} = 1 - \frac{\sum_{i,j}(w_{ij} \cdot o_{ij})}{\sum_{i,j}(w_{ij} \cdot e_{ij})} \tag{4.2}$$

Here, the k_{qw} represents the QWK. It uses a weighting scheme, w_{ij}, which assigns weights to the agreement or disagreement between raters i and j. The observed agreement frequency, o_{ij}, is obtained by measuring the count of each rater's agreements. The expected agreements, e_{ij}, are calculated based on chance agreement. The formula calculates the agreement between the human grader and the AES model, where 0 represents no agreement appearing by chance, and 1 represents perfect agreement.

Unlike direct performance measurements such as accuracy, F1-score, precision, and recall, QWK considers the probability of the agreement happening by chance [97], which makes it a robust measure for the performance of AES. Additionally, the QWK metric is the official evaluation metric of the ASAP competition. Other works such as [72], [108], [42], [49], [86], [28], [102], [59], [53], [55] that applied the ASAP dataset also implement QWK as their evaluation metric, which also allows comparing the performance of proposed work against the other works.

4.4.2 Paired t-test

To assess the statistical significance of the improvement achieved by adding the new features, a paired t-test is proposed. This test compares the prediction scores of the base model and the model with the new features. The p-value for the paired t-test

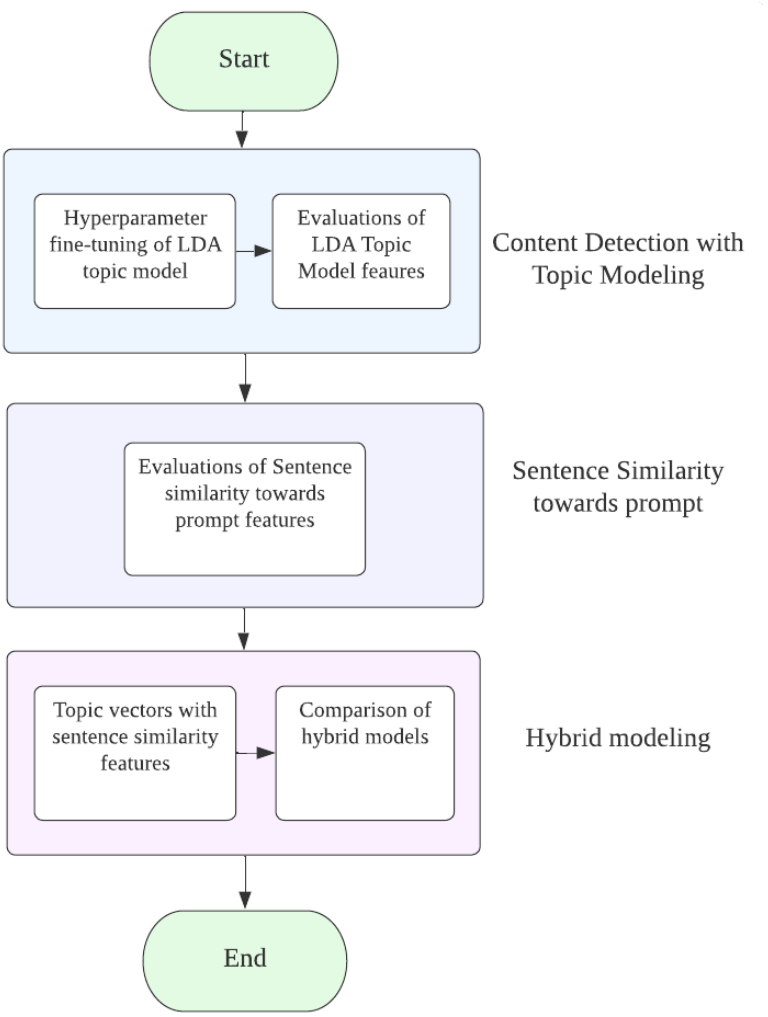


Figure 4.1: The flow of the experiments in the next three chapters.

is predetermined to be 0.05, indicating that a change is considered significant if the calculated p-value is less than or equal to 0.05.

4.5 Summary

Figure 4.1 summarizes the experiments flow in the following three chapters. First, Chapter 5, content detection with topic modelling, will start with the experiment on the hyperparameter fine-tuning of the LDA topic model, followed by the evaluations of LDA topic model features. Next, Chapter 6 will describe the evaluations for sentence similarity towards the prompt. Lastly, Chapter 7, hybrid modelling, will describe the experiments combining the topic vectors with sentence similarity features, then compare the proposed hybrid model against the recent state-of-the-art AES models.

Chapter 5

Content Detection with Topic Modeling

This chapter presents and discusses the experimental results of the proposed Topic Modeling approach in AES. The experiments are held for a different purpose, each will be discussed in a subsection:

1. Hyperparameter fine-tuning of LDA topic model
    - Grid search across a range of parameters suggested by the authors Blei et al. [8]
2. Evaluations of LDA topic model features
    - Compared against the base model by Phandi et al. [72] in term of QWK score
    - Compared against the base model by Phandi et al. [72] in term of the variance of QWK score
    - Paired t-test between new model's output and base model's output

5.1 Hyperparameter fine-tuning of LDA topic model

First, the research has implemented a grid search on the hyperparameters of the LDA topic model to determine the best LDA topic model. To this end, the grid search has

Table 5.1: The optimum LDA hyperparameters from the grid search.

Essay Set	Number of Topic k	Alpha α	Beta β
1	10	1	0.1
2	11	1	0.09
3	5	0.01	0.2
4	7	0.01	0.14
5	6	1	0.17
6	6	0.2	0.17
7	10	0.5	0.1
8	8	0.2	0.13

been performed on the topic numbers k from range 5 to 50, α of (0.01, 0.05, 0.1, 0.2, 0.5, and 1), and β of $\frac{1}{k}$ on each essay set. The result is summarized in Table 5.1.

The optimum number of topics for LDA topics lies between 5 to 11, proving that essays with below 500 words will have at most 15 topics is valid. Additionally, the result demonstrated that the essay sets (1, 2, 7, and 8) with higher average length appeared to have more topics than the shorter essays (3, 4, 5, and 6). The essay sets 1 and 2 with the same genre of argumentative share the same α. However, more argumentative essay sets would be required to validate this situation.

5.2 Evaluations of LDA Topic Model features

This section describes the evaluation results for the LDA topic model features in AES.

5.2.1 QWK Evaluations for the LDA Topic Model features

Table 5.2 shows the result of implementing the topic models in the model training compared to the base algorithm. In argumentative essays, the LDA topic models are slightly underperforming in SVM and have a minor improvement in BLRR. It can be observed that essay set 1 is slightly underperforming in both algorithms. However, essay set 2, with the same genre as essay set 1, has achieved quite a significant improvement. In response & short essays, the average QWK of the LDA models has significant improvement in both SVM and BLRR by nearly 0.04 QWK score. Essay set 3 falls short in the BLRR algorithm for response genre essays. The rest of the response essay, including essay, sets 4, 5, and 6, are significantly improving, especially on set 6 LDA model in BLRR, where it can improve its performance by a 0.087 QWK score. In narrative essays, the average QWK of the LDA models has only a minor improvement of 0.002 QWK in SVM and 0.015 QWK in BLRR. The narrative essay set 7 has achieved a major improvement,

Table 5.2: QWK comparison between the LDA Topic Model and the base model.

Essay set	SVM		BLRR	
	Base	LDA	Base	LDA
1	**0.827**	0.821	**0.813**	0.812
2	0.665	**0.688**	0.645	**0.658**
3	0.670	**0.679**	**0.649**	0.643
4	0.678	**0.725**	0.666	**0.719**
5	0.790	**0.792**	0.777	**0.787**
6	0.661	**0.751**	0.659	**0.747**
7	0.712	**0.722**	0.702	**0.731**
8	**0.675**	0.670	0.681	**0.682**
Avg 1-8	0.710	**0.731**	0.699	**0.722**
Avg 1-2 (Argumentative)	**0.746**	0.745	0.729	**0.735**
Avg 3-6 (Response & short essays)	0.700	**0.737**	0.688	**0.724**
Avg 7-8 (Narrative)	0.694	**0.696**	0.692	**0.707**
Avg 1-2 & 7-8 (Long essays)	0.720	**0.725**	0.710	**0.721**

Table 5.3: Variance of QWK comparison between LDA Topic Model and the base model.

Essay set	SVM		BLRR	
	Base	LDA	Base	LDA
1	0.120E-03	**0.101E-03**	**0.050E-03**	0.096E-03
2	1.628E-03	**0.812E-03**	1.869E-03	**1.411E-03**
3	1.787E-03	**1.271E-03**	1.870E-03	**1.736E-03**
4	0.854E-03	**0.563E-03**	**0.243E-03**	0.411E-03
5	0.356E-03	**0.208E-03**	**0.041E-03**	0.305E-03
6	1.580E-03	**0.147E-03**	1.746E-03	**0.406E-03**
7	0.397E-03	**0.357E-03**	**0.200E-03**	0.233E-03
8	**2.069E-03**	2.737E-03	2.341E-03	**2.036E-03**
1-8	4.549E-03	**3.263E-03**	4.387E-03	**3.881E-03**
1-2 (Argumentative)	8.109E-03	**5.300E-03**	8.716E-03	**7.236E-03**
3-6 (Response & short essays)	3.866E-03	**2.221E-03**	3.639E-03	**3.515E-03**
7-8 (Narrative)	**1.472E-03**	2.108E-03	**1.234E-03**	1.680E-03
1-2 & 7-8 (Long essays)	5.267E-03	**4.409E-03**	5.097E-03	**4.446E-03**

but the narrative essay set 8 is falling short by a little. The results show that the LDA topic model has a more significant impact on short essays than on long ones. Overall, on average, of all eight essay sets, the QWK scores of the LDA models are improved in both SVM and BLRR.

Table 5.3 describes the variance of QWK score across five-fold cross-validation to determine the robustness of the proposed topic features against the base model. Overall, the LDA and base models have relatively low variance. Still, the proposed LDA models managed to have lower variance compared to the base model, which proves that the proposed features are more consistent and robust. Comparing against different genres of essays, the LDA models in both SVM and BLRR have more consistent results with lower variance in argumentative and response essays. Also, the LDA models slightly fall short on the narrative essays with slightly higher variance compared to the base model.

Table 5.4: The LDA topic model features paired t-test test results.

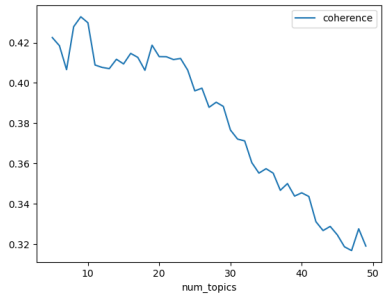
Model	Essay Set	H0 counts	H1 counts
SVM	1	2	3
	2	0	5
	3	2	3
	4	0	5
	5	1	4
	6	0	5
	7	0	5
	8	2	3
	Total	7	33
BLRR	1	0	5
	2	0	5
	3	1	4
	4	0	5
	5	0	5
	6	0	5
	7	0	5
	8	0	5
	Total	1	39

5.2.2 Significant test of LDA topic model features

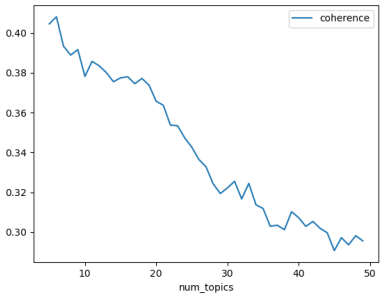
Table 5.4 describes the result of the paired t-test of the new proposed topic vector features against the existing features from EASE. H0 represents the two pairs that are the same. Vice versa, H1 represents that the two pairs are not the same. In total, there will be 40 t-tests performed as 5-fold cross-validation is implemented on the eight essay sets. 90% (72/80 H1) of topic vectors' results significantly differ from existing EASE features in SVM and BLRR machine learning algorithms. However, on SVM essay sets 1,3 and 8, the H0 counts are relatively higher than the rest of the sets. The 3 sets that have higher H0 counts are the sets that have minor improvement or no improvement on QWK in Table 5.2. Hence, the new proposed topic vectors feature justified to improve the AES model's performance in a different aspect of the existing features with the two pieces of evidence. First, the improving performance in the Table 5.2. Second, the result of paired t-test 5.4.

5.3 Discussions

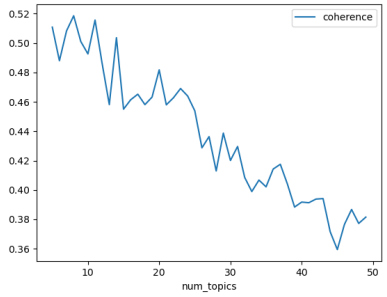
This sections will further discuss on the effect of number of topics on LDA topic models and the LDA topic modeling in AES model.



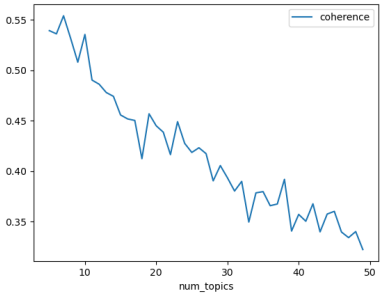
(a) Essay set 1



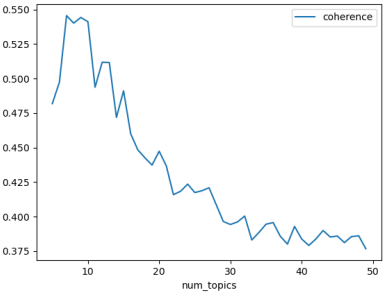
(b) Essay set 2



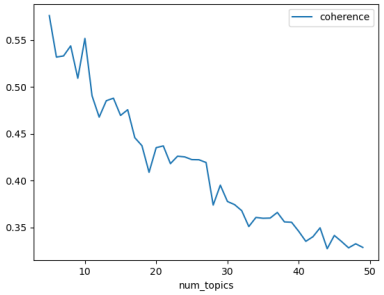
(c) Essay set 3



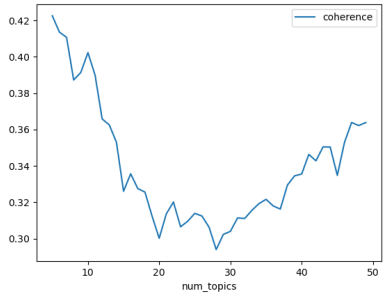
(d) Essay set 4



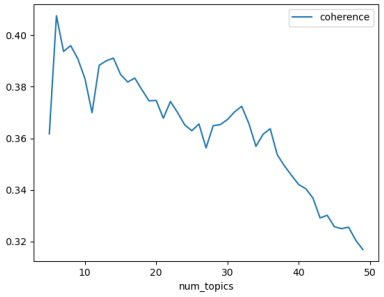
(e) Essay set 5



(f) Essay set 6



(g) Essay set 7



(h) Essay set 8

Figure 5.1: Effect of a different number of topics for LDA topic model on the essay sets.

5.3.1 Effect of number of topics on LDA topic models

Figure 5.1 shows the effect of different topic numbers for the LDA topic model on the essay sets in terms of coherence score. Figure 5.1(a) and 5.1(b) represent the argumentative essays with a similar downward trend of the number of topics on the coherence score. Similarly, Figure 5.1(c), 5.1(d), 5.1(e) and 5.1(f) represents the response essays have a similar downward trend of the number of topics on the coherence score. However, the downward trend of Figure 5.1(e) stops at 35 number of topics. Compared with argumentative essays, the response essays have a larger fluctuation on the downward trend of the number of topics on the coherence score. Unlike the other essay genres, Figure 5.1(g) and 5.1(h) have a different trend of the number of topics on the coherence score. The essay set 7 (Figure 5.1(g)) has a dramatic downward trend from 0 to 30 topic numbers, then continues with a gradual upward trend. Essay set 8 gradually declined to 35 number of topics, then continued with a steep decline.

The essays in the essay sets have an average length of 150 to 650 words. The average sentence length in the 2000s is 15 words [80], but it may vary depending on the genre. A topic is normally described within an average of 3 sentences [20]. Formula 5.1 is proposed based on the theories to estimate the optimum topic number. AW is the average word of the essay set; W is the average word length per sentence, which is 15; S is the average sentence count per topic, which is 3.

$$EstimatedTopicNumber = \frac{AW}{W * S} \tag{5.1}$$

The formula reduces the computation cost of implementing a huge range of topic numbers in fine-tuning the LDA model. Using the formula, the essay in the essay sets should have 3 to 15 topics in English essays. The theories are further proven by the LDA models, as it tends to have optimum performance at 5 to 15 topic numbers in the essay sets.

A further study has been made to determine if increasing topic numbers in essay set 7 will improve performance. Figure 5.2 shows the effect of increasing topic numbers up to 150 in essay set 7. At approximate topic number 30, the coherence score creates an upward trend. However, the upward trend stopped at around topic number 100, where the fluctuation of the coherence score started to become very minimal. The peak coherence score of the topic number is 10, which is still within the 5 to 15 topic number.

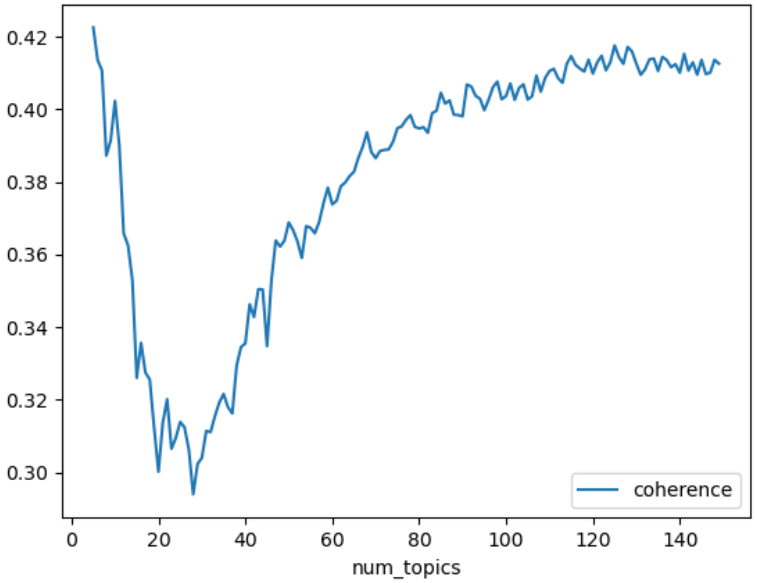


Figure 5.2: Further study on the effect of the number of topics on essay set 7.

From the study, it can reduce the computation cost needed to train the LDA model by limit down the topic numbers to train the LDA model.

5.3.2 LDA topic modelling in AES model

The proposed feature based on LDA topic modelling has successfully captured significant inputs to train the AES model. The proposed features can be applied to any essay regarding length as it captures the topics from the training set. It does not require labelled topics to provide the score of each topic due to the unsupervised characteristics of LDA. Recent works such as [86, 103] that achieved the best performance compared to works before the year 2022 have implemented RoBERTa's embedding to capture the semantic representation of the essays. However, the RoBERTa's embedding is weak in only providing a maximum of 512 tokens representation. For essays longer than 512 words, the RoBERTa's embedding required truncating words that exceed 512 words or performing a sliding window method. Furthermore, LDA requires fewer resources to perform an average of 23 seconds to obtain the topic modelling features.

Chapter 6

Sentence Similarity Toward Prompt

This chapter presents and discusses the experimental results of the proposed sentence similarity toward prompt features AES. The experiments are held for a different purpose; each will be discussed in a subsection:

Evaluations of Sentence similarity towards prompt features:

- Compared against the base model by Phandi et al. [72] in term of QWK score
- Compared against the base model by Phandi et al. [72] in term of the variance of QWK score
- Paired t-test between new model's output and base model's output

6.1 Evaluations of Sentence similarity towards prompt features

This section describes the evaluation results for the sentence similarity towards prompt features in AES.

Table 6.1: Result of sentence similarity towards prompt features with existing features on each essay sets in QWK.

Essay set	SVM		BLRR	
	Base	Similarity	Base	Similarity
1	0.827	**0.832**	0.813	**0.818**
2	0.665	**0.690**	0.645	**0.679**
3	**0.670**	0.663	**0.649**	0.647
4	0.678	**0.707**	0.666	**0.676**
5	**0.790**	0.788	**0.777**	0.778
6	0.661	**0.668**	0.656	**0.667**
7	0.712	**0.739**	0.702	**0.725**
8	0.675	**0.682**	0.681	**0.697**
Avg	0.710	**0.721**	0.699	**0.711**
Avg 1-2 (Argumentative)	0.746	**0.761**	0.729	**0.749**
Avg 3-6 (Response & short essays)	0.700	**0.707**	0.688	**0.692**
Avg 7-8 (Narrative)	0.694	**0.711**	0.692	**0.711**
Avg 1-2 & 7-8 (Long essays)	0.720	**0.736**	0.710	**0.730**

6.1.1 QWK Evaluations for the sentence similarity towards prompt features

Table 6.1 describes the effect of new sentence similarity on each essay set's QWK outcome. For each essay set, the better result is bold-faced. Overall, the sentence similarity has a minor improvement in the performance of the base AES model. Specifically, sentence similarity tends to perform better on more extended essay sets; namely, essay sets 1, 2, 7, and 8. The essay sets 1, 2, 7, and 8 have argumentative and narrative genre prompts, which require the students to write ideas surrounding the prompt with their own words. The sentence similarity of the prompt is highly crucial in these two genres, as the essays written with content not related to the prompt are often given low scores. However, as referred to in the prompts, essay sets 3, 4, 5, and 6 are response essays where the essays might have relatively high similarity to the prompt. The response essays are required to write with examples from the essays. The response essays will have similar high sentence similarity scores among the distribution, which would dilute the similarity features proposed in the work.

Table 6.2 describes the variance of the QWK score across five-fold cross-validation to determine the robustness of the proposed sentence similarity towards prompt features against the base model. Overall, the proposed features managed to have a lower variance than the base model, proving that the proposed features are more consistent and robust. Specifically, the sentence similarity towards prompt features in both SVM and BLRR have more consistent results with lower variance in long essays (essay sets 1, 2, 7 & 8). However, it falls short in response & short essays in terms of variance.

Table 6.2: Variance of QWK comparison between sentence similarity against prompt and the base model.

Essay set	SVM		BLRR	
	Base	Similarity	Base	Similarity
1	0.120E-03	**0.061E-03**	**0.050E-03**	0.168E-03
2	**1.628E-03**	2.412E-03	1.869E-03	**1.716E-03**
3	**1.787E-03**	2.191E-03	1.870E-03	**1.507E-03**
4	0.854E-03	**0.358E-03**	0.243E-03	**0.202E-03**
5	**0.356E-03**	0.384E-03	**0.041E-03**	0.255E-03
6	1.580E-03	**0.490E-03**	1.746E-03	**0.786E-03**
7	0.397E-03	**0.075E-03**	0.200E-03	**0.180E-03**
8	2.069E-03	**1.247E-03**	2.341E-03	**1.455E-03**
1-8	4.549E-03	**4.366E-03**	4.387E-03	**4.013E-03**
1-2 (Argumentative)	8.109E-03	**6.776E-03**	8.716E-03	**6.397E-03**
3-6 (Response & short essays)	**3.866E-03**	3.914E-03	3.639E-03	**3.590E-03**
7-8 (Narrative)	1.472E-03	**1.138E-03**	1.234E-03	**0.864E-03**
1-2 & 7-8 (Long essays)	5.267E-03	**4.493E-03**	5.097E-03	**3.800E-03**

6.1.2 Significant test of sentence similarity features

Table 6.3 shows the paired t-test result of the new proposed sentence similarity features against the existing features from EASE. H0 represents the two pairs that are the same. Vice versa, H1 represents that the two pairs are not the same. In total, there will be 40 t-tests performed as 5-fold cross-validation is implemented on the eight essay sets. The majority of sentence similarity' results are significantly different from existing EASE features in SVM and BLRR machine learning algorithms. On SVM, it has a higher chance of being the same as the result of the base model, which is something to be concerned about. The higher H0 count is most likely to occur in the sets with minor improvement or no improvement shown in Table 6.1. However, the rest of the results can justify that the proposed sentence similarity features are improving the AES model's performance in a different aspect of the existing features with the two pieces of evidence. First, the improving performance in the Table 6.1. Second, the paired t-test results are in Table 6.3.

6.2 Discussions

This sections will further discuss on the effect of sentence similarity towards prompt on different essay genres and the sentence similarity towards prompt in AES.

Table 6.3: The Sentence similarity paired t-test test results

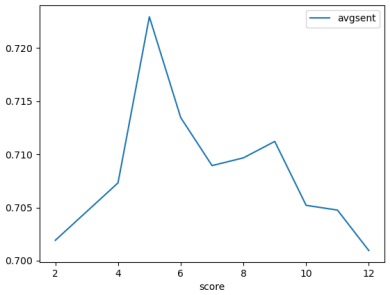
Model	Essay Set	H0 counts	H1 counts
SVM	1	0	5
	2	1	4
	3	2	3
	4	0	5
	5	3	2
	6	0	5
	7	0	5
	8	2	3
	Total	8	32
BLRR	1	2	3
	2	2	3
	3	0	5
	4	2	3
	5	1	4
	6	1	4
	7	0	5
	8	0	5
	Total	8	32

6.2.1 The effect of sentence similarity towards prompt on different essay sets.

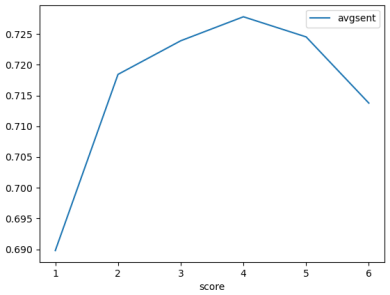
Figure 6.1 shows the average sentence similarity score against essay score on different essay sets. For argumentative essays (essay sets 1 & 2) in Figure 6.1(a) and 6.1(b), the average sentence similarity score is at its highest around the median of the essay score. However, the higher and lower scores of essays tend to have lower average sentence similarity scores. The lower scores of essays can be written content unrelated to the prompt. The higher scores of essays can be written with content related to the prompt but in a unique way to describe their arguments. For response essays (essay sets 3, 4, 5 & 6) in Figure 6.1(c), 6.1(d), 6.1(e) and 6.1(f), the average sentence similarity scores have a similar trend of higher sentence similarity scores for higher essay scores. This again proves our assumption that response essays will have a higher sentence similarity score for higher essay scores. The reason is that the response essays required students to write based on the sentences given in the prompt. For narrative essays (essay sets 7 & 8) in Figure 6.1(g) and 6.1(h), the average sentence similarity scores have a similar pattern to the argumentative essays, except they have a larger scale of scores.

6.2.2 Sentence similarity towards prompt in AES model

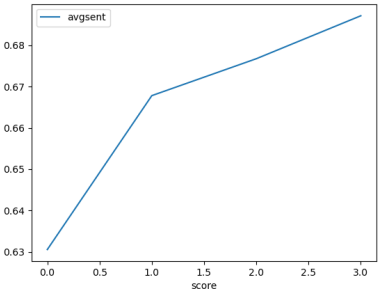
The second proposed feature, sentence similarity towards prompt, has further improved the results of the topic modelling feature. It avoided the limitation of RoBERTa's embedding, which only allows a maximum of 512 words of embedding by comparing



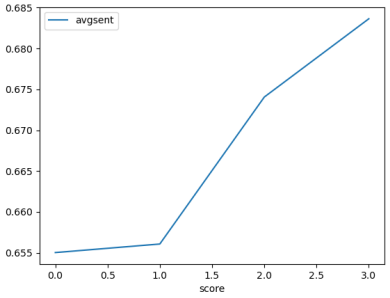
(a) Essay set 1



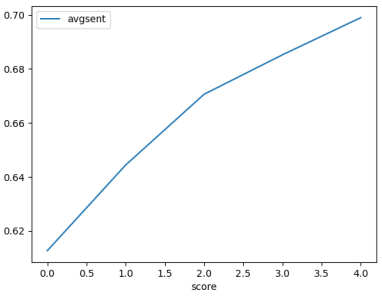
(b) Essay set 2



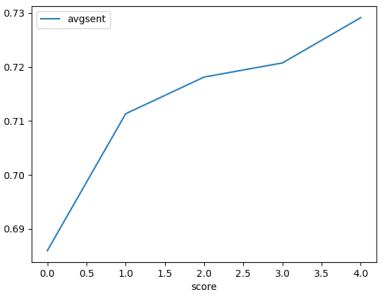
(c) Essay set 3



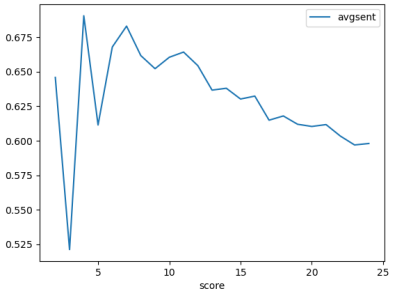
(d) Essay set 4



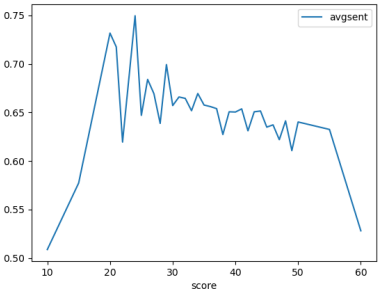
(e) Essay set 5



(f) Essay set 6



(g) Essay set 7



(h) Essay set 8

Figure 6.1: Average sentence similarity score against essay score on different essay sets.

each essay sentence towards the prompt. However, there is still one possible limitation to the proposed sentence similarity. If the prompt exceeds 512 words, the embedding of the prompt will truncate the rest of the prompt after 512 words.

Chapter 7

Hybrid modeling with Content Detection and Sentence Similarity towards Prompt

This chapter presents and discusses the experimental results of the proposed sentence similarity toward prompt features AES. The experiments are held for a different purpose, each will be discussed in a subsection:

1. Evaluation of the integration of LDA topic model features and sentence similarity features
    - Compared against the base model by Phandi et al. [72] in term of QWK score
    - Compared against the base model by Phandi et al. [72] in term of the variance of QWK score
2. Hybrid model
    - Compared against recent state-of-the-art models by Yang et al.[108], Liao et al.[53], and Wang et al.[103] in term of QWK score

Table 7.1: Result for the combination of topic vectors and sentence similarity features.

Essay Set	SVM		BLRR	
	Base	LDA + Sim	Base	LDA + Sim
1	0.827	**0.827**	0.813	**0.818**
2	0.665	**0.685**	0.645	**0.679**
3	**0.670**	0.668	**0.649**	0.636
4	0.678	**0.739**	0.666	**0.726**
5	0.790	**0.799**	0.777	**0.790**
6	0.661	**0.754**	0.656	**0.754**
7	0.712	**0.751**	0.702	**0.741**
8	0.675	**0.694**	0.681	**0.703**
Avg	0.710	**0.740**	0.699	**0.731**
Avg 1-2 (Argumentative)	0.746	**0.756**	0.729	**0.749**
Avg 3-6 (Response & short essays)	0.700	**0.740**	0.688	**0.727**
Avg 7-8 (Narrative)	0.694	**0.723**	0.692	**0.722**
Avg 1-2 & 7-8 (Long essays)	0.720	**0.739**	0.710	**0.735**

7.1 Topic vectors with sentence similarity features

Table 7.1 shows the result of new proposed features against the base. The best result is bold-faced. LDA + Sim represents the result of combining new features (topic vectors & sentence similarity features) and base features in a machine learning algorithm. Overall, the LDA + Sim models perform better than Base models except on essay set 3 due to the underperforming topic vector features and similarity features on essay set 3. However, by combining topic vector features and similarity features, the result of most essay sets can achieve the highest performance. On average, the LDA + Sim model can achieve a new highest performance of 0.74 QWK for SVM and 0.731 for BLRR. Specifically, the LDA + Sim models work well in every essay genre. The LDA topic modelling features work well in response essays. In contrast, sentence similarity features work well in argumentative and narrative essays.

Table 7.2 describes the variance of the QWK score across five-fold cross-validation to determine the robustness of the proposed topic features against the base model. Overall, the LDA + Sim and base models have relatively low variance with minor differences. Still, the proposed models managed to be more robust by having lower variance than the base model. Specifically, on the different essay genres, the LDA + Sim models have lower variance in argumentative and response essays. However, it falls short in the narrative essay in terms of variance. The narrative essay sets have the most extensive score range and smallest sample size, which could have a higher chance of inconsistent training results.

Table 7.2: Variance of QWK comparison between LDA + Sim models and the base models.

Essay set	SVM		BLRR	
	Base	LDA + Sim	Base	LDA + Sim
1	**0.120E-03**	0.138E-03	**0.050E-03**	0.122E-03
2	**1.628E-03**	1.633E-03	1.869E-03	**1.022E-03**
3	1.787E-03	**1.226E-03**	1.870E-03	**1.648E-03**
4	**0.854E-03**	0.941E-03	**0.243E-03**	0.457E-03
5	0.356E-03	**0.140E-03**	**0.041E-03**	0.101E-03
6	1.580E-03	**0.097E-03**	1.746E-03	**0.154E-03**
7	0.397E-03	**0.295E-03**	**0.200E-03**	0.295E-03
8	**2.069E-03**	2.445E-03	2.341E-03	**1.749E-03**
1-8	4.549E-03	**3.114E-03**	4.387E-03	**3.376E-03**
1-2 (Argumentative)	8.109E-03	**5.364E-03**	8.716E-03	**5.281E-03**
3-6 (Response & short essays)	3.866E-0	**2.304**E-03	3.639E-03	**3.365E-03**
7-8 (Narrative)	**1.472E-03**	2.258E-03	**1.234E-03**	1.621E-03
1-2 & 7-8 (Long essays)	5.267E-03	**4.085E-03**	5.097E-03	**3.511E-03**

7.2 Comparison of hybrid models

The research has worked on a hybrid approach to the AES model by combining the feature-based machine learning model with the output of the neural network-based transformer model. The features of topic vectors and sentence similarity are implemented together with the existing features to train the machine learning models. The selected transformer model, RoBERTa, is fine-tuned with the essay text. To integrate the result of the two models, the hybrid method uses the prediction score provided by the transformers model as the input of the machine learning algorithms to train the model. Table A.2 presents the result of hybrid modeling trained in SVM regressor and BLRR compared with the base model, the RoBERTa model and the BERT model. Since the result of the SVM regressor hybrid model and BLRR hybrid model is performing well in a different genre, the ensemble learning technique, stack regressor model, is applied to ensemble the multiple regression models.

In this section, the experimental results of the proposed hybrid model are compared with the fine-tuned RoBERTa and the state-of-the-art models from recent years. Table 7.3 tabulates the stack regressor hybrid model's experimental results compared with the state-of-the-art models in recent years. Table 7.3 tabulates the experimental results of the proposed model against the reported results from Yang et al., 2020 [108], Liao et al., 2021 [53], and Wang et al., 2022 [103]. All the essay sets can outperform RoBERTa except essay 3, where both the topic model and sentence features are lacking, referring to Table 7.1. Two of the eight datasets achieved the highest QWK score, and five achieved more than 0.8 QWK, which is considered an almost perfect agreement with the original scores [14]. The overall average QWK score is 0.789, just 0.005 QWK scores

Table 7.3: Result of the proposed model against the fine-tuned RoBERTa and the state-of-the-art models of recent years.

Essay Set	Proposed Model	Fine-tuned RoBERTa	Wang et al., 2022[103]	Liao et al., 2021[53]	Yang et al., 2020[108]
1	0.836	0.817	0.834	**0.839**	0.817
2	0.711	0.694	0.716	0.702	**0.719**
3	0.683	0.683	**0.714**	0.711	0.698
4	0.820	0.816	0.812	0.809	**0.845**
5	0.818	0.814	0.813	0.801	**0.841**
6	0.827	0.821	0.836	0.827	**0.847**
7	**0.846**	0.841	0.839	0.820	0.839
8	**0.770**	0.755	0.766	0.631	0.744
Avg 1-8	0.789	0.780	0.791	0.763	**0.794**
Avg 1-2 (Argumentative)	0.774	0.756	**0.775**	0.771	0.768
Avg 3-6 (Response & Short essays)	0.787	0.784	0.794	0.787	**0.808**
Avg 7-8 (Narrative)	**0.808**	0.798	0.803	0.726	0.792
Avg 1-2 & 7-8 (Long essays)	**0.791**	0.777	0.789	0.748	0.780

lower than the best overall QWK score by Yang et al. [108]. However, when looking into the performance in different genres, the proposed model achieved the best performance in narrative essays and the top two in the argumentative essays by just 0.001 lower in QWK score. When looking into the performance of essay length, the proposed model also achieved the best performance in QWK in long essays (more than 250 words). These results prove that our proposed model is robust enough to deal with most essay types, as the model achieves the best performance in long essays while also achieving more than 0.8 QWK among 3 of the response and short essays.

In summary, most results are rather close except for essay set 3, where both newly proposed features are lacking. These results prove that regardless of which pre-trained language model, there is still a need for features engineering to improve further on the AES performance. Although RoBERTa is selected as one of the hybrid elements in the proposed model, the proposed framework can be easily switched to other more advanced pre-trained models in the future.

7.3 Discussion

Based on the experimental results discussed in the previous section, it is proven that the proposed framework is robust and can outperform the existing state-of-the-art solutions in the long essays. The idea of topic detection using topic modelling and sentence similarity toward prompts aims to overcome the missing element in scoring an essay.

As presented in Table 7.3, the proposed feature further improves the result of RoBERTa and outperforms the existing state-of-the-art solutions in the long essays. The transformer model's embedding common has a limitation of a maximum of 512 words which requires truncating the essay to more than 512 words. Performing truncation will make the transformer miss out on some information from the essays. Introducing topic model features and sentence similarity features fills up the gap.

The proposed model framework implements a regression model instead of a classification model. The proposed model framework will predict the final score into a range of 0 to 1, then rescale it back to the intended score range of the train set. Hence, the framework can be implemented easily without considering the number of scores available in the essay sets.

Chapter 8

Conclusion

8.1 Research Objective Review

Three research objectives have been proposed in this research work. For **RO1**, the topic modelling features (topic vectors) were produced successfully and improved from the base model, specifically on the short essays. The sentence similarity features were produced successfully to achieve **RO2**. The new features improved further from the base model, specifically on the long essays. The two features were concatenated with the output from the fine-tuned RoBERTa and existing grammatical and lexical features from the base model. It was then used to train the new model using a stacked regressor that combined the two machine-learning algorithms from the base model. Finally, **RO3** was achieved from the hybrid model that concatenates the outputs from feature engineering and neural network. The proposed hybrid model outperformed the state-of-the-art models in recent years in the narrative genre and long essays.

To summarize the achievements completed in the research, the achievements are shown as follows:

RO1 was achieved with the produced topic modelling features from LDA. The features further improved the base model significantly on short essays.

RO2 was achieved with the produced sentence similarity towards prompt features based on the transformer model. The features further improved the base model significantly on long essays.

RO3 was achieved with the integration of handcrafted features and RoBERTa's output to form a hybrid model. The hybrid model outperforms the recent year's state-of-the-art models in narrative and long essays.

In this research work, a further study on the optimum topic numbers to train the LDA topic model for essay scoring purposes. From the study, the formula 5.1 has been proposed to estimate the optimum topic number to reduce the computation cost for LDA topic model fine-tuning. For sentence similarity, it is found that the response essays tend to have higher similarity for higher essay scores. However, other genres react in a totally different behaviour. Essays from other genres have the highest similarity score near the median of the essay score.

The newly proposed features are proven to overcome the missing semantic elements in the essay scoring models through the evaluations. The features can fill the gaps of the transformer model by only allowing a maximum of 512 words. Also, it provides semantic feedback for students to identify the element lacking in their writing.

8.2 Limitations and further research

One limitation of this research is that all experiments were conducted using the ASAP dataset. While the ASAP dataset provides valuable insights and serves as a basis for analysis, it is important to acknowledge that the ASAP dataset represents only a specific domain in AES. Future research could focus on exploring how to build a more generalizable model that can be implemented across different datasets and domains in AES. This would involve investigating the transfer learning of the proposed approaches to different datasets and domains.

In the research, the pre-trained transformer language models implemented focused are only BERT base and RoBERTa base models. However, there are many other variants of language models with similar architecture, such as GPT-2 [75], DistilBERT [84], ALBERT[48], XLM-RoBERTa [16] and many more, that are worth exploring for better performance on similarity and essay scoring.

The dataset implemented in the experiments is relatively small in size, which makes it unable to perform well in deep learning models [10, 47]. Further research can be more

deep learning focused if essay-scoring datasets can be enlarged. The current research concatenates the features from different models and then trains upon machine learning algorithms. Instead of performing feature engineering or feature selection on topic vectors and sentence similarity, further research can feed all inputs into a deep learning model and let the model provide weights on each input.

The current research concatenates the output of fine-tuned BERT models with the existing and new proposed features. Further research can take the last layer of fine-tuned BERT models with max or mean pooling technique and concatenates them with the existing and proposed features to train on a deep learning model. The last layer of fine-tuned BERT models has the best representation of the essay, which can be further trained with new inputs.

In the academic field, it is important to provide evaluation feedback for students to help determine what works well and what can be improved in an essay. Hence, the two proposed features can provide semantic insights to students to identify what is lacking in their writing. Figure 8.1 shows an example of a student evaluation feedback report based on the proposed semantic features and existing lexical and grammatical features. Evaluation feedback is one of the crucial elements for human labour to be required in essay scoring, which can be solved by providing feedback using the features.

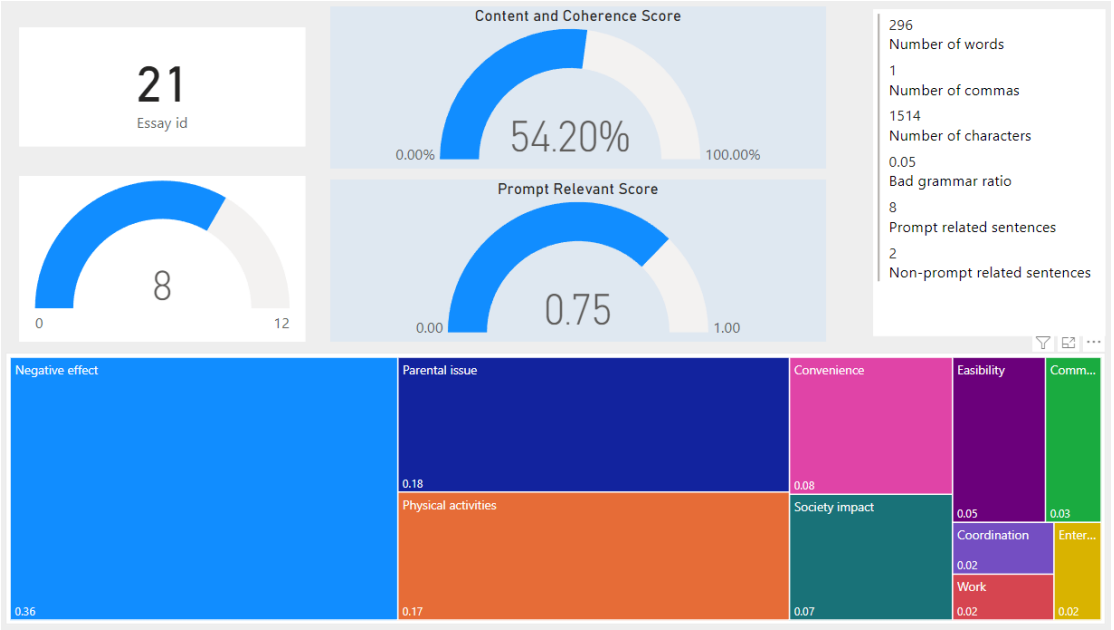


Figure 8.1: Example evaluation feedback report for students

Appendix A

Other experiment results

Table A.1: Result comparison based on BERTopic Topic Model against the base model.

Essay Set	SVM base	SVM+BERTopic	BLRR base	BLRR+BERTopic
1	**0.827**	0.827	**0.813**	0.807
2	**0.665**	0.662	0.645	**0.659**
3	**0.670**	0.667	**0.649**	0.644
4	0.678	**0.702**	0.666	**0.680**
5	**0.790**	0.780	**0.777**	0.776
6	0.661	**0.707**	0.659	**0.709**
7	0.712	**0.727**	0.702	**0.717**
8	**0.675**	0.653	**0.681**	0.679
Average	0.710	**0.716**	0.699	**0.709**

Table A.2: Result of hybrid modeling compared to base, Roberta and BERT.

Essay Set	SVM		BLRR		Roberta	Bert
	Base	Hybrid	Base	Hybrid		
1	0.830	**0.832**	0.813	0.826	0.817	0.812
2	0.668	**0.700**	0.644	0.696	0.694	0.675
3	0.669	0.679	0.647	0.664	**0.683**	0.654
4	0.697	0.816	0.667	0.815	**0.816**	0.796
5	0.789	0.813	0.779	0.811	**0.814**	0.802
6	0.673	**0.821**	0.804	0.821	0.821	0.803
7	0.720	0.839	0.700	**0.842**	0.841	0.828
8	0.672	0.756	0.679	**0.767**	0.755	0.722
Avg	0.715	**0.782**	0.698	0.780	0.780	0.762

Bibliography

[1] A. Adamson, A. Lamb, and R. December. Automated essay grading. 2014.

[2] Y. Attali and J. Burstein. Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, 4(3), 2006.

[3] A. Barrón-Cedeno, P. Rosso, E. Agirre, and G. Labaka. Plagiarism detection across distant language pairs. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 37–45, 2010.

[4] V. Batanović and D. Bojić. Using part-of-speech tags as deep-syntax indicators in determining short-text semantic similarity. *Computer Science and Information Systems*, 12(1):1–31, 2015.

[5] M. Beseiso and S. Alzahrani. An empirical analysis of bert embedding for automated essay scoring. *Int. J. Adv. Comput. Sci. Appl.*, 11(10):204–210, 2020.

[6] D. M. Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.

[7] D. M. Blei and J. D. Lafferty. Topic models. In *Text mining*, pages 101–124. Chapman and Hall/CRC, 2009.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[9] D. M. Blei, J. D. Lafferty, et al. A correlated topic model of science. *The Annals of Applied Statistics*, 1(1):17–35, 2007.

[10] D. Boulanger and V. Kumar. Deep learning in automated essay scoring. In *Intelligent Tutoring Systems: 14th International Conference, ITS 2018, Montreal, QC, Canada, June 11–15, 2018, Proceedings 14*, pages 294–299. Springer, 2018.

[11] T. Briscoe, J. A. Carroll, and R. Watson. The second release of the rasp system. In *Proceedings of the COLING/ACL 2006 interactive presentation sessions*, pages 77–80, 2006.

[12] L. Chen, Y. Wang, Q. Yu, Z. Zheng, and J. Wu. Wt-lda: user tagging augmented lda for web service clustering. In *International conference on service-oriented computing*, pages 162–176. Springer, 2013.

[13] J. R. Christie. Automated essay marking-for both style and content. In *Proceedings of the Third Annual Computer Assisted Assessment Conference, Loughborough University, Loughborough, UK*. Citeseer, 1999.

[14] J. Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[15] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017.

[16] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116, 2019. URL http://arxiv.org/abs/1911.02116.

[17] J. O. Contreras, S. Hilles, and Z. B. Abubakar. Automated essay scoring with ontology based on text mining and nltk tools. In *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pages 1–6. IEEE, 2018.

[18] M. Cozma, A. M. Butnaru, and R. T. Ionescu. Automated essay scoring with string kernels and word embeddings. *arXiv preprint arXiv:1804.07954*, 2018.

[19] R. Cummins, M. Zhang, and E. Briscoe. Constrained multi-task learning for automated essay scoring. Association for Computational Linguistics, 2016.

[20] F. J. D'Angelo. The topic sentence revisited. *College composition and communication*, 37(4):431–441, 1986.

[21] S. M. Darwish and S. K. Mohamed. Automated essay evaluation based on fusion of fuzzy ontology and latent semantic analysis. In *International Conference on Advanced Machine Learning Technologies and Applications*, pages 566–575. Springer, 2019.

[22] D. Das, N. Schneider, D. Chen, and N. A. Smith. Probabilistic frame-semantic parsing. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 948–956, 2010.

[23] T. Dasgupta, A. Naskar, L. Dey, and R. Saha. Augmenting textual qualitative features in deep convolution recurrent neural network for automatic essay scoring. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 93–102, 2018.

[24] A. De, M. Huang, T. Feng, X. Yue, L. Yao, et al. Analyzing patient secure messages using a fast health care interoperability resources (fihr)–based data model: Development and topic modeling study. *Journal of medical Internet research*, 23 (7):e26770, 2021.

[25] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[26] M. M. Deza and E. Deza. Encyclopedia of distances. In *Encyclopedia of distances*, pages 1–583. Springer, 2009.

[27] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[28] F. Dong and Y. Zhang. Automatic features for essay scoring–an empirical study. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1072–1077, 2016.

[29] F. Dong, Y. Zhang, and J. Yang. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st conference on computational natural language learning (CoNLL 2017)*, pages 153–162, 2017.

[30] M. A. Fauzi, D. C. Utomo, B. D. Setiawan, and E. S. Pramukantoro. Automatic essay scoring system using n-gram and cosine similarity for gamification based e-learning. In *Proceedings of the International Conference on Advances in Image Processing*, pages 151–155, 2017.

[31] P. W. Foltz, D. Laham, and T. K. Landauer. The intelligent essay assessor: Applications to educational technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 1(2):939–944, 1999.

[32] B. Galitsky. Machine learning of syntactic parse trees for search and classification of text. *Engineering Applications of Artificial Intelligence*, 26(3):1072–1091, 2013.

[33] M. Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure, 2022. URL https://arxiv.org/abs/2203.05794.

[34] F. Gurcan, N. E. Cagiltay, and K. Cagiltay. Mapping human–computer interaction research themes and trends from its existence to today: A topic modeling-based review of past 60 years. *International Journal of Human–Computer Interaction*, 37(3):267–280, 2021.

[35] P. Hastings, S. Hughes, J. P. Magliano, S. R. Goldman, and K. Lawless. Assessing the use of multiple sources in student essays. *Behavior Research Methods*, 44(3): 622–633, 2012.

[36] I. Heintz, R. Gabbard, M. Srivastava, D. Barner, D. Black, M. Friedman, and R. Weischedel. Automatic extraction of linguistic metaphors with lda topic modeling. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 58–66, 2013.

[37] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, 1999.

[38] Y. Hu, A. John, F. Wang, and S. Kambhampati. Et-lda: Joint topic modeling for aligning events and their twitter feedback. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[39] G. Huang, J. Liu, C. Fan, and T. Pan. Off-topic english essay detection model based on hybrid semantic space for automated english essay scoring system. In *MATEC Web of Conferences*, volume 232, page 01035. EDP Sciences, 2018.

[40] M. M. Islam and A. L. Hoque. Automated essay scoring using generalized latent semantic analysis. In *2010 13th International Conference on Computer and Information Technology (ICCIT)*, pages 358–363. IEEE, 2010.

[41] P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, 37:547–579, 1901.

[42] H. K. Janda, A. Pawar, S. Du, and V. Mago. Syntactic, semantic and sentiment analysis: The joint effect on automated essay evaluation. *IEEE Access*, 7:108486–108503, 2019.

[43] M. A. Jaro. Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. *Journal of the American Statistical Association*, 84(406):414–420, 1989.

[44] A. Karami, M. Lundy, F. Webb, G. Turner-McGrievy, B. W. McKeever, and R. McKeever. Identifying and analyzing health-related themes in disinformation shared by conservative and liberal russian trolls on twitter. *International journal of environmental research and public health*, 18(4):2159, 2021.

[45] P. Kherwa and P. Bansal. Semantic n-gram topic modeling. *EAI Endorsed Transactions on Scalable Information Systems*, 7(26), 2020.

[46] B. B. Klebanov, C. Stab, J. Burstein, Y. Song, B. Gyawali, and I. Gurevych. Argumentation: Content, structure, and relationship with essay quality. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 70–75, 2016.

[47] V. S. Kumar and D. Boulanger. Automated essay scoring and the deep learning black box: How are rubric scores determined? *International Journal of Artificial Intelligence in Education*, 31(3):538–584, 2021.

[48] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942, 2019. URL http://arxiv.org/abs/1909.11942.

[49] S. Latifi and M. Gierl. Automated scoring of junior and senior high essays using coh-metrix features: Implications for large-scale language testing. *Language Testing*, page 0265532220929918, 2020.

[50] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[51] W. Li and A. McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pages 577–584, 2006.

[52] Z. Li, J. Tang, X. Wang, J. Liu, and H. Lu. Multimedia news summarization in search. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(3): 1–20, 2016.

[53] D. Liao, J. Xu, G. Li, and Y. Wang. Hierarchical coherence modeling for document quality assessment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13353–13361, 2021.

[54] D. Litman, H. Zhang, R. Correnti, L. C. Matsumura, and E. Wang. A fairness evaluation of automated methods for scoring text evidence usage in writing. In *International Conference on Artificial Intelligence in Education*, pages 255–267. Springer, 2021.

[55] J. Liu, Y. Xu, and Y. Zhu. Automated essay scoring based on two-stage learning. *arXiv preprint arXiv:1901.07744*, 2019.

[56] Q. Liu, M. J. Kusner, and P. Blunsom. A survey on contextual embeddings. *arXiv preprint arXiv:2003.07278*, 2020.

[57] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[58] J. Ma, X. Li, M. Chen, and W. Yang. Enhanced hierarchical structure features for automated essay scoring. In *China Conference on Information Retrieval*, pages 168–179. Springer, 2021.

[59] E. Mayfield and A. W. Black. Should you fine-tune bert for automated essay scoring? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 151–162, 2020.

[60] P. McNamee and J. Mayfield. Character n-gram tokenization for european language text retrieval. *Information retrieval*, 7(1):73–97, 2004.

[61] M. Mesgar and M. Strube. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 4328–4339, 2018.

[62] E. Miltsakaki and K. Kukich. Automated evaluation of coherence in student essays. In *Proceedings of LREC 2000*, pages 1–8, 2000.

[63] M. Mohler and R. Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575, 2009.

[64] M. Molavi, M. Tavakoli, and G. Kismihók. Extracting topics from open educational resources. In *European Conference on Technology Enhanced Learning*, pages 455–460. Springer, 2020.

[65] M. Mustak, J. Salminen, L. Plé, and J. Wirtz. Artificial intelligence in marketing: Topic modeling, scientometric analysis, and research agenda. *Journal of Business Research*, 124:389–404, 2021.

[66] B. M'sik and B. M. Casablanca. Topic modeling coherence: A comparative study between lda and nmf models using covid'19 corpus. *International Journal*, 9(4), 2020.

[67] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453, 1970.

[68] L. S. Norton. Essay-writing: what really counts? *Higher Education*, 20(4):411–442, 1990.

[69] O. E. Oduntan, S. O. Olabiyisi, I. A. Adeyanju, and E. O. Omidiora. A modified principal component analysis approach to automated essay-type grading. In *2016 Future Technologies Conference (FTC)*, pages 94–98. IEEE, 2016.

[70] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[71] I. Persing and V. Ng. Modeling thesis clarity in student essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 260–269, 2013.

[72] P. Phandi, K. M. A. Chai, and H. T. Ng. Flexible domain adaptation for automated essay scoring using correlated linear regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 431–439, 2015.

[73] L. Pradhan, C. Zhang, and S. Bethard. Towards extracting coherent user concerns and their hierarchical organization from user reviews. In *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*, pages 582–590. IEEE, 2016.

[74] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63 (10):1872–1897, 2020.

[75] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. 2019.

[76] D. Ramesh and S. K. Sanampudi. Coherence based automatic essay scoring using sentence embedding and recurrent neural networks. In S. R. M. Prasanna, A. Karpov, K. Samudravijaya, and S. S. Agrawal, editors, *Speech and Computer*, pages 139–154, Cham, 2022. Springer International Publishing. ISBN 978-3-031-20980-2.

[77] Y. Rao, J. Lei, L. Wenyin, Q. Li, and M. Chen. Building emotional dictionary for sentiment analysis of online news. *World Wide Web*, 17(4):723–742, 2014.

[78] M. Röder, A. Both, and A. Hinneburg. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408, 2015.

[79] X. Rong. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*, 2014.

[80] K. Rudnicka. Variation of sentence length across time and genre. *Diachronic corpora, genre, and language change*, pages 220–240, 2018.

[81] S. K. Saha and D. Rao CH. Development of a practical system for computerized evaluation of descriptive answers of middle school level students. *Interactive Learning Environments*, pages 1–14, 2019.

[82] Y. Salim, V. Stevanus, E. Barlian, A. C. Sari, and D. Suhartono. Automated english digital essay grader using machine learning. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, pages 1–6. IEEE, 2019.

[83] G. Salton. Some research problems in automatic information retrieval. In *ACM SIGIR Forum*, volume 17, pages 252–263. ACM New York, NY, USA, 1983.

[84] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

[85] M. Sendra, R. Sutrisno, J. Harianata, D. Suhartono, and A. B. Asmani. Enhanced latent semantic analysis by considering mistyped words in automated essay scoring. In *2016 International Conference on Informatics and Computing (ICIC)*, pages 304–308. IEEE, 2016.

[86] A. Sharma, A. Kabra, and R. Kapoor. Feature enhanced capsule networks for robust automatic essay scoring. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 365–380. Springer, 2021.

[87] A. Shehab, M. Elhoseny, and A. E. Hassanien. A hybrid scheme for automated essay grading based on lvq and nlp techniques. In *2016 12th International Computer Engineering Conference (ICENCO)*, pages 65–70. IEEE, 2016.

[88] M. D. Shermis and J. C. Burstein. *Automated essay scoring: A cross-disciplinary perspective*. Routledge, 2003.

[89] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

[90] S. Syed and M. Spruit. Full-text or abstract? examining topic coherence scores using latent dirichlet allocation. In *2017 IEEE International conference on data science and advanced analytics (DSAA)*, pages 165–174. IEEE, 2017.

[91] K. Taghipour and H. T. Ng. A neural approach to automated essay scoring. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1882–1891, 2016.

[92] J. S. Tan and I. K. Tan. Feature group importance for automated essay scoring. In *International Conference on Multi-disciplinary Trends in Artificial Intelligence*, pages 58–70. Springer, 2021.

[93] S. Tan, Y. Li, H. Sun, Z. Guan, X. Yan, J. Bu, C. Chen, and X. He. Interpreting the public sentiment variations on twitter. *IEEE transactions on knowledge and data engineering*, 26(5):1158–1170, 2013.

[94] Y. R. Tausczik and J. W. Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54, 2010.

[95] Y. Tay, M. C. Phan, L. A. Tuan, and S. C. Hui. Skipflow: Incorporating neural coherence features for end-to-end automatic text scoring. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[96] S. Vanbelle. A new interpretation of the weighted kappa coefficients. *Psychometrika*, 81(2):399–410, 2016.

[97] S. Vanbelle and A. Albert. A note on the linearly weighted kappa coefficient for ordinal scales. *Statistical Methodology*, 6(2):157–163, 2009.

[98] V. Vargas-Calderón, J. E. Camargo, H. Vinck-Posada, et al. Event detection in colombian security twitter news using fine-grained latent topic analysis. *arXiv preprint arXiv:1911.08370*, 2019.

[99] I. Vayansky and S. A. Kumar. A review of topic modeling methods. *Information Systems*, 94:101582, 2020.

[100] X. Wang, A. McCallum, and X. Wei. Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *Seventh IEEE international conference on data mining (ICDM 2007)*, pages 697–702. IEEE, 2007.

[101] Y. Wang, Y. Rao, X. Zhan, H. Chen, M. Luo, and J. Yin. Sentiment and emotion classification over noisy labels. *Knowledge-Based Systems*, 111:207–216, 2016.

[102] Y. Wang, Z. Wei, Y. Zhou, and X.-J. Huang. Automatic essay scoring incorporating rating schema via reinforcement learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 791–797, 2018.

[103] Y. Wang, C. Wang, R. Li, and H. Lin. On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation. *arXiv preprint arXiv:2205.03835*, 2022.

[104] A. Wilson and P. A. Chew. Term weighting schemes for latent dirichlet allocation. In *human language technologies: The 2010 annual conference of the North American Chapter of the Association for Computational Linguistics*, pages 465–473, 2010.

[105] W. E. Winkler. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. 1990.

[106] P. Xie, D. Yang, and E. P. Xing. Incorporating word correlation knowledge into topic modeling. In *HLT-NAACL*, volume 10, page v1, 2015.

[107] J. Yang, Y. Li, C. Gao, and Y. Zhang. Measuring the short text similarity based on semantic and syntactic information. *Future Generation Computer Systems*, 114: 169–180, 2021.

[108] R. Yang, J. Cao, Z. Wen, Y. Wu, and X. He. Enhancing automated essay scoring performance via cohesion measurement and combination of regression and ranking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1560–1569, 2020.

[109] H. Yannakoudakis, T. Briscoe, and B. Medlock. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 180–189, 2011.

[110] J.-F. Yeh, Y.-S. Tan, and C.-H. Lee. Topic detection and tracking for conversational content by using conceptual dynamic latent dirichlet allocation. *Neurocomputing*, 216:310–318, 2016.

[111] H. Zhang, Y. Cai, B. Zhu, C. Zheng, K. Yang, R. C.-W. Wong, and Q. Li. Incorporating concept information into term weighting schemes for topic models. In

International Conference on Database Systems for Advanced Applications, pages 227–244. Springer, 2020.

[112] F. Zhao, Y. Zhu, H. Jin, and L. T. Yang. A personalized hashtag recommendation approach using lda-based topic model in microblog environment. *Future Generation Computer Systems*, 65:196–206, 2016.

[113] S. Zhao, Y. Zhang, X. Xiong, A. Botelho, and N. Heffernan. A memory-augmented neural model for automated grading. In *Proceedings of the fourth (2017) ACM conference on learning@ scale*, pages 189–192, 2017.

[114] X. Zhu, T. Li, and G. De Melo. Exploring semantic properties of sentence embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 632–637, 2018.