Improving Automatic Short Answer Grading with Deep Learning Architectures

A Thesis submitted to Gujarat Technological University

for the Award of

Doctor of Philosophy

in

Computer/IT Engineering

by

Gamit Nikunj Chunilal

Enrollment No.: 179999913009

under supervision of

Dr. Ajaykumar N. Upadhyaya

GUJARAT TECHNOLOGICAL UNIVERSITY

AHMEDABAD

August - 2026

Declaration

I declare that the thesis entitled **"Improving Automatic Short Answer Grading with Deep Learning Architectures"** submitted by me for the degree of Doctor of Philosophy is the record of research work carried out by me during the period from **May 2018** to **May 2026** under the supervision of **Dr. Ajaykumar N. Upadhyaya**, Professor, SAL Engineering & Technical Institute, SAL Education, Ahmedabad and this has not formed the basis for the award of degree, diploma, associateship, fellowship, titles in this or any other University or other institution of higher learning.

I further declare that the material obtained from other sources has been duly acknowledged in the thesis. I shall be solely responsible for any plagiarism or other irregularities, if noticed in the thesis.

Signature of the Research Scholar: Date: 24/08/2026

Name of Research Scholar: **Gamit Nikunj Chunilal**

Place: Ahmedabad.

CERTIFICATE

I certify that the work incorporated in the thesis "**Improving Automatic Short Answer Grading with Deep Learning Architectures**" submitted by **Gamit Nikunj Chunilal** was carried out by the candidate under my guidance. To the best of my knowledge: (i) the candidate has not submitted the same research work to any other institution for any degree, diploma, Associateship, Fellowship or other similar titles. (ii) the thesis submitted is a record of original research work done by Research Scholar during the period of study under my supervision, and (iii) the thesis represents independent research work on the part of the Research Scholar.

Signature of Research Supervisor:  Date: 24/08/2026

Name of Research Supervisor: **Dr. Ajaykumar N. Upadhyaya**

Place: Ahmedabad.

Course-work Completion Certificate

This is to certify that **Gamit Nikunj Chunilal** enrolment no. **179999913009** is a PhD scholar enrolled for PhD program in the branch **Computer / IT Engineering** of Gujarat Technological University, Ahmedabad.

(Please tick the relevant option(s))

☐ He has been exempted from the course-work (successfully completed during M.Phil Course)

☐ He has been exempted from Research Methodology Course only (successfully completed during M.Phil Course)

☑ He has successfully completed the PhD course work for the partial requirement for the award of PhD Degree. His performance in the course work is as follows-

Grade Obtained in Research Methodology (PH001)	Grade Obtained in Self Study Course (Core Subject) (PH002)
BB	**AB**

Signature of Research Supervisor:

Name of Research Supervisor: **Dr. Ajaykumar N. Upadhyaya**

Originality Report Certificate

It is certified that PhD Thesis titled **"Improving Automatic Short Answer Grading with Deep Learning Architectures"** by **Gamit Nikunj Chunilal** has been examined by us. We undertake the following:

a. Thesis has significant new work / knowledge as compared already published or are under consideration to be published elsewhere. No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is placed under quotation marks and duly referenced.

b. The work presented is original and own work of the author (i.e. there is no plagiarism). No ideas, processes, results or words of others have been presented as Author own work.

c. There is no fabrication of data or results which have been compiled / analyzed.

d. There is no falsification by manipulating research materials, equipment or processes, or changing or omitting data or results such that the research is not accurately represented in the research record.

e. The thesis has been checked using **Turnitin** (copy of originality report attached) and found within limits as per GTU Plagiarism Policy and instructions issued from time to time (i.e. permitted similarity index ≤ 10%).

Signature of the Research Scholar: Date: 24/08/2026

Name of Research Scholar: **Gamit Nikunj Chunilal**

Place: Ahmedabad.

Signature of Research Supervisor: Date: 24/08/2026

Name of Research Supervisor: **Dr. Ajaykumar N. Upadhyaya**

Place: Ahmedabad.

Nikunj Gamit

Improving ASAG using DL Architectures

ASAG using DL

PhD_Research2

Gujarat Technological University

Document Details

Submission ID
trn:oid:::1:3629858427

Submission Date
Aug 19, 2026, 6:03 PM GMT+5:30

Download Date
Aug 19, 2026, 6:06 PM GMT+5:30

File Name
Gamit_Nikunj_THESIS_FINAL.pdf

File Size
2.6 MB

148 Pages

31,718 Words

185,282 Characters

6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

126 Not Cited or Quoted 6%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
3% Publications
3% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

PhD THESIS Non-Exclusive License to

GUJARAT TECHNOLOGICAL UNIVERSITY

In consideration of being a PhD Research Scholar at Gujarat Technological University and in the interests of the facilitation of research at GTU and elsewhere, I, **Gamit Nikunj Chunilal** having Enrollment no.**179999913009** hereby grant a non-exclusive, royalty free and perpetual license to GTU on the following terms:

a. The University is permitted to archive, reproduce and distribute my thesis, in whole or in part, and/or my abstract, in whole or in part (referred to collectively as the "Work") anywhere in the world, for non-commercial purposes, in all forms of media;

b. The University is permitted to authorize, sub-lease, sub-contract or procure any of the acts mentioned in paragraph (a);

c. The University is authorized to submit the Work at any National / International Library, under the authority of their "Thesis Non-Exclusive License";

d. The Universal Copyright Notice (©) shall appear on all copies made under the authority of this license;

e. I undertake to submit my thesis, through my University, to any Library and Archives. Any abstract submitted with the thesis will be considered to form part of the thesis.

f. I represent that my thesis is my original work, does not infringe any rights of others, including privacy rights, and that I have the right to make the grant conferred by this non-exclusive license.

g. If third party copyrighted material was included in my thesis for which, under the terms of the Copyright Act, written permission from the copyright owners is required, I have obtained such permission from the copyright owners to do the acts mentioned in paragraph (a) above for the full term of copyright protection.

h. I understand that the responsibility for the matter as mentioned in paragraph (g) rests with the authors/me solely. In no case GTU have any liability for any acts / omissions / errors / copyright infringement from the publication of the said thesis or otherwise.

i. I retain copyright ownership and moral rights in my thesis, and may deal with the copyright in my thesis, in any way consistent with rights granted by me to my University in this non-exclusive license.

j. GTU logo shall not be used/printed in the book (in any manner whatsoever) being published or any promotional or marketing materials or any such similar documents.

k. The following statement shall be included appropriately and displayed prominently in the book or any material being published anywhere: "The content of the published work is part of the thesis submitted in partial fulfillment for the award of the degree of Ph.D. in Computer/IT Engineering of the Gujarat Technological University."

l. I further promise to inform any person to whom I may hereafter assign or license my copyright in my thesis of the rights granted by me to my University in this non-exclusive license. I shall keep GTU indemnified from any and all claims from the Publisher(s) or any third parties at all times resulting or arising from the publishing or use or intended use of the book/such similar document or its contents.

m. I am aware of and agree to accept the conditions and regulations of PhD including all policy matters related to authorship and plagiarism.

Signature of the Research Scholar:

Place: Ahmedabad Date: 24/08/2026

Recommendation of the Research Supervisor:

Signature of Research Supervisor:

Thesis Approval Form

The viva-voce of the PhD Thesis submitted by **Gamit Nikunj Chunilal** (Enrollment No. **179999913009**) entitled **"Improving Automatic Short Answer Grading with Deep Learning Architectures"** was conducted on Monday, 24-AUG-2026 at Gujarat Technological University.

(Please tick any one of the following option)

☑ The performance of the candidate was satisfactory. We recommend that he be awarded the PhD degree.

☐ Any further modifications in research work recommended by the panel after 3 months from the date of first viva-voce upon request of the Supervisor or request of Independent Research Scholar after which viva-voce can be re-conducted by the same panel again.

(briefly specify the modifications suggested by the panel)

☐ The performance of the candidate was unsatisfactory. We recommend that he should not be awarded the PhD degree.

(The panel must give justifications for rejecting the research work)

Name and Signature
of Supervisor with Seal
(Dr. Ajaykumar N. Upadhyaya)

(External Examiner 1)
Name and Signature
(Dr. Dharmendra Sharma)

(External Examiner 2)
Name and Signature
(Dr. Apurva Shah)

x

Acknowledgement

Research is never a solitary journey, it is shaped by the guidance, encouragement, and unwavering support of many individuals. I consider it a privilege to acknowledge those who stood by me and made this research journey both meaningful and fulfilling.

I express my gratitude to my research supervisor, Dr. Ajaykumar N. Upadhyaya, whose mentorship has been invaluable throughout my doctoral journey. His patience, wisdom, and unwavering belief in my abilities not only shaped this research but also nurtured my academic confidence. His guidance has been a source of inspiration during both progress and uncertainty.

I express my gratitude to my former research supervisor, Dr. Shailesh D. Panchal, for his valuable guidance and support during the initial phase of my doctoral journey. His insightful suggestions, encouragement, and strong academic foundation played a significant role in shaping the direction of this research. His mentorship helped me develop clarity in my approach and laid the groundwork for the progress achieved in this work.

I am eternally grateful to my wife, Komal, my pillar of strength, for her unconditional love, sacrifices, prayers, and constant encouragement. I also lovingly acknowledge my daughter Yati, whose innocence, patience, and affection brought joy and renewed strength during the most demanding phases of this work. I am deeply grateful to my parents, whose values, discipline, and unwavering support laid the strong foundation of my academic journey.

I sincerely thank my esteemed Doctoral Progress Committee members Dr. Amit Ganatra and Dr. Amit Thakkar, for their constructive feedback, insightful suggestions, and continued support, which played a vital role in refining the quality and direction of my research.

Special appreciation is extended to Dr. Safvan Vahora and Dr. Jashvant Dave for their valuable guidance, insightful discussions, and continuous support throughout the research work. Their encouragement, constructive suggestions, and sustained involvement have been instrumental in the successful completion of this work.

Sincere appreciation is extended to Dr. Jignesh H. Vaniya, Prof. Naimisha S. Trivedi, Dr. Khushali Raval, Dr. Dipak Patel, Dr. Manmitsinh Zala and Dr. Om Mehta for their valuable discussions, innovative ideas, constructive inputs, and consistent support. Their contributions and encouragement have been greatly appreciated throughout the research journey.

I express my sincere gratitude to my Head of Department, Dr. Vibha Patel, and all my esteemed colleagues for creating a supportive and motivating research environment. I also acknowledge the institute for granting me the opportunity and providing the necessary facilities to pursue my PhD research.

Finally, I extend my sincere gratitude to all those who may not be mentioned individually but whose support, directly or indirectly, contributed to the successful completion of this research.

With heartfelt gratitude,

Gamit Nikunj Chunilal

Abstract

Automatic Short Answer Grading (ASAG) plays a key role in scalable educational assessment by enabling computational evaluation of students' free-text responses with consistency and efficiency. Advances in Natural Language Processing, particularly transformer-based models such as BERT and Sentence-BERT, have improved meaning-based similarity estimation. However, these approaches often fail to capture true conceptual correctness, as high semantic similarity does not guarantee understanding—especially when key concepts are missing, answer length biases exist, or logical contradictions such as negation occur.

This research reframes ASAG as a structured educational reasoning problem rather than a pure similarity task. It addresses three core limitations: length discrepancy, where models struggle with varying answer lengths; correctness gap, where similarity does not reflect concept coverage; and Discrete Boundary Collapse (DBC), where continuous vector spaces fail to preserve clear grading distinctions.

To overcome these issues, a multi-architecture framework is proposed. First, a Universal Length-Adaptive Design routes answers to suitable sub-models based on length, integrating BM25, TF-IDF, SBERT, BERT, and Bi-LSTM. Second, a Concept-Aware Siamese Model incorporates term-weighted embeddings, concept extraction, coverage scoring, and contradiction-aware gating, enhanced by a Negation-Sensitive Score Calibration (NSSC) mechanism. Third, DBC is formalised as a geometric limitation, and BAT-TW-BERT is introduced to address it using term-weighted and attention-biased embeddings.

Experiments on benchmark datasets including Mohler, SciEntsBank, and ASAG2024 (18,067 responses) show consistent improvements over strong baselines. The Universal model achieves an F1-score of 91.2% and a Pearson correlation of 0.90. The Concept-Aware model improves concept recall by 4.7%, while NSSC increases negation-specific QWK from 0.52 to 0.74. These results demonstrate a principled advancement in automated short-answer grading.

Keywords: Automatic Short Answer Grading, Deep Learning, BERT, Sentence Embeddings, Length-Adaptive Grading, Concept Awareness, Negation Sensitivity, Boundary-Aware Training, Educational Assessment, NLP.

Contents

List of Abbreviations

ASAG	Automatic Short Answer Grading
BAT	Boundary-Aware Training
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BM25	Best Match 25 (Okapi BM25)
CRF	Conditional Random Field
DBC	Discrete Boundary Collapse
DL	Deep Learning
DPC	Doctoral Progress Committee
F1	F1 Score (Harmonic Mean of Precision and Recall)
GloVe	Global Vectors for Word Representation
ITS	Intelligent Tutoring System
LLM	Large Language Model
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
NLI	Natural Language Inference
NLP	Natural Language Processing
NSSC	Negation-Sensitive Score Calibration
QWK	Quadratic Weighted Kappa
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RO	Research Objective
SBERT	Sentence-BERT
SEB	SciEntsBank
STS	Semantic Textual Similarity
TF-IDF	Term Frequency – Inverse Document Frequency
TW	Triplet Weighting
USE	Universal Sentence Encoder
wRMSE	Weighted Root Mean Squared Error

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

1.1 Background and Motivation

Education rests on a simple cycle: students learn, they show understanding, instructors evaluate that understanding, and the evaluation guides the next round of learning. This cycle has been at the heart of formal education for centuries and its effectiveness depends critically on the quality and timeliness of the feedback that instructors provide. When feedback is delayed, incomplete, or inconsistent, the learning cycle is disrupted-students cannot correct misconceptions, instructors cannot identify systemic gaps, and the overall quality of the educational experience deteriorates.

For decades, short-answer questions have been a preferred instrument in this cycle because they require students to recall and articulate knowledge rather than merely recognise a correct option from a list. Unlike multiple-choice items, which test recognition, short-answer questions probe deeper cognitive processes-comprehension, application, and synthesis. A student answering "explain why a binary search tree is more efficient than a linear search" must not only recall the concept of logarithmic complexity but also construct a coherent argument that demonstrates genuine understanding. This kind of active knowledge construction is widely recognised in the educational psychology literature as a superior form of learning compared to passive recognition tasks [77]. Yet this pedagogical advantage comes at a cost: grading free-text responses is labour-intensive, time-consuming, and difficult to scale.

This challenge has grown sharply in recent years. The global expansion of online education, the proliferation of Massive Open Online Courses (MOOCs), and the shift to digital assessment accelerated by the COVID-19 pandemic have created a scenario

where a single course may enrol thousands of students, each producing dozens of free-text responses over a term. Manual grading at this scale is not just difficult-it is simply not practical. Instructors face an impossible trade-off between assessment depth and scalability: they can ask rich, open-ended questions that reveal real understanding, or they can ask multiple-choice questions that are easy to scale-but doing both at once is very hard. In practice, many instructors resolve this tension by reducing the frequency or depth of free-text assessments, which directly compromises the quality of the learning experience.

Automatic Short Answer Grading (ASAG) promises to resolve this trade-off. By using computational methods to evaluate student responses against reference answers, ASAG systems aim to provide consistent, scalable, and timely feedback while keeping the learning benefits of free-text questions. The field has come a long way since its early days: from early rule-based and keyword-matching systems that could only recognise pre-specified correct terms, to statistical methods built on feature engineering that could handle paraphrase and synonym variation, and more recently to deep learning models that learn rich semantic representations directly from data. Each generation of approaches has addressed limitations of the previous one, steadily pushing the performance ceiling closer to human-level agreement.

The biggest recent step forward has been the use of transformer models-particularly BERT [36] and its derivatives-which produce contextualised embeddings that capture subtle semantic relationships between words in their full sentence context. Unlike earlier word embedding models such as Word2Vec or GloVe, which assign a single fixed vector to each word regardless of context, transformer-based encoders produce different representations for the same word depending on the surrounding sentence. This context-sensitivity is crucial for grading tasks, where the meaning of a student's answer can depend entirely on the precise phrasing used. Sentence-level encoders such as Sentence-BERT (SBERT) [113] have made this practical at scale, allowing efficient comparison between student and reference answers through simple cosine similarity in the embedding space. These developments have brought ASAG performance close to human-level agreement on some benchmarks.

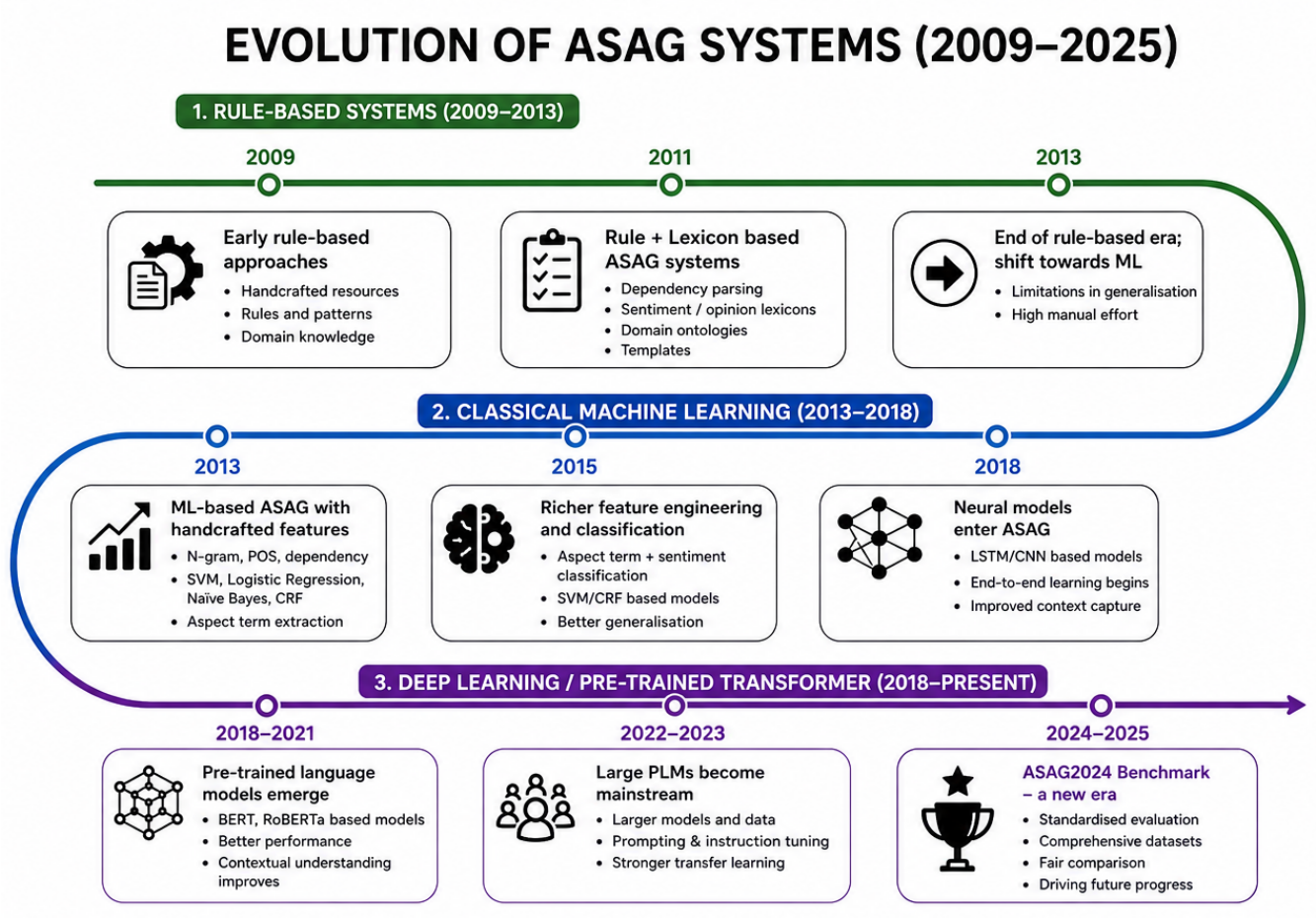

FIGURE 1.1: Evolution of ASAG systems from 2009 to 2025. Three broad paradigms are visible: rule-based systems (2009–2013), classical machine learning (2013–2018), and deep learning / pre-trained transformer approaches (2018–present). The ASAG2024 benchmark marks a new era of standardised evaluation.

Figure 1.1 illustrates the historical evolution of ASAG research, from early rule-based systems in the 2000s through classical machine learning approaches in the 2010s to the current era of deep neural network and pre-trained language models. The figure makes clear that the field has not progressed through isolated breakthroughs but through a gradual accumulation of insights, with each paradigm building on the representational and methodological foundations laid by its predecessors. Despite this progression, substantial challenges remain that prevent ASAG systems from achieving the reliability required for high-stakes deployment. These challenges are not merely engineering difficulties to be overcome by more computation or larger models-they reflect fundamental tensions between the nature of continuous embedding spaces and the discrete, concept-driven nature of academic grading, which form the central motivation for this thesis.

1.1.1 The Scale Problem in Modern Education

To appreciate the magnitude of the scale problem, consider the following statistics. These figures are not abstract projections-they reflect the current reality of digital education at a global scale, and they collectively make the case that manual grading is no longer a viable default strategy for large-scale educational institutions:

- As of 2023, Coursera alone reported over 124 million registered learners across 7,000+ courses [15]. Even if only a fraction of these learners submit written responses in any given course, the total volume of text requiring evaluation is measured in hundreds of millions of responses per year-a quantity that no human workforce could evaluate consistently and in a timely manner.
- The average instructor-to-student ratio in MOOCs exceeds 1:1,000, making personalised feedback practically impossible without automation. In traditional classroom settings, research suggests that an instructor can provide meaningful written feedback to roughly 30–40 students per assignment. At a ratio of 1:1,000, the same instructor would need to grade 25 times as many

responses, making any form of substantive written feedback effectively impossible.

- Studies have shown that timely feedback (within 24 hours) notably improves learning outcomes compared to delayed feedback (more than 72 hours) [77]. The educational value of assessment is therefore not just a function of whether feedback is provided, but of when it is provided. An automated grading system that returns a score and diagnostic feedback within seconds of submission has the potential to transform the feedback loop from a bottleneck into a real-time learning tool.

- In the Indian higher education system, with over 40 million students enrolled in degree programmes, the demand for scalable assessment tools is particularly acute. The combination of high enrolment, resource constraints, and a strong cultural emphasis on examination-based assessment creates a setting where automated grading could have a transformative impact on educational quality and equity.

These statistics collectively establish a compelling case for ASAG as a research priority with direct societal impact. The problem is not a niche technical challenge-it is a structural bottleneck in the global educational system, and addressing it requires both practical engineering advances and rigorous scientific foundations of the kind pursued in this thesis.

1.1.2 Why Short Answers Are Difficult to Grade Automatically

The apparent simplicity of short-answer grading belies its computational complexity. A short answer is, by definition, a brief textual response-typically ranging from a single word to a few sentences-that a student produces in response to a specific question. Despite their brevity, these responses can vary enormously in vocabulary, syntax, level of detail, and conceptual coverage, even when they are equally correct. This variability is precisely what makes short answers pedagogically valuable and computationally challenging to grade.

A human grader, when evaluating a student response, draws upon a rich combination of linguistic knowledge, domain expertise, and pedagogical judgment that has been developed over years of teaching experience. This judgment operates at multiple levels simultaneously, and it is difficult to decompose into a set of explicit rules or features. Specifically, the grader must:

1. **Identify the key concepts** required by the question and determine whether the student has addressed them. This requires understanding not just the surface wording of the reference answer, but the underlying conceptual structure that the question is designed to probe. A question asking students to "explain the difference between a stack and a queue" requires the grader to recognise that the key concepts are LIFO/FIFO ordering, the push/pop vs. enqueue/dequeue operations, and the typical use cases-not just any mention of the words "stack" or "queue."

2. **Assess semantic equivalence** between the student's phrasing and the reference answer, recognising paraphrases, synonyms, and domain-specific terminology. A student who writes "a function that invokes itself" is expressing the same concept as one who writes "a recursive function," even though the surface forms share no words. Human graders recognise this equivalence effortlessly; automated systems must learn it from data.

3. **Handle negation and contradiction**-a student who states "the algorithm does not terminate" when the correct answer is "the algorithm terminates" should receive a low grade despite high surface similarity. Negation is particularly treacherous because it can completely reverse the factual content of an answer while preserving most of the vocabulary and syntactic structure, making it very difficult for surface-level matching systems to detect.

4. **Account for answer length**-a one-word answer that happens to contain the key term is qualitatively different from a two-sentence explanation that shows understanding. Length is a confounding variable: a very short answer may be correct but incomplete, while a very long answer may be verbose but

conceptually correct. A grading system must be sensitive to this distinction rather than treating all answers of similar length as equally informative.

5. **Apply consistent grade boundaries**-the difference between a 3/5 and a 4/5 grade should be applied consistently across all students, regardless of the order in which responses are graded. This consistency requirement is particularly challenging for automated systems, which must learn to map continuous similarity scores onto discrete grade categories in a way that aligns with the ordinal structure of the grading rubric.

Each of these requirements maps directly to one of the four research objectives addressed in this thesis, as detailed in Section 1.4. The correspondence is not coincidental-the research objectives were derived precisely from this analysis of what human graders do, and the proposed architectures are designed to replicate each of these capabilities in a computationally tractable form.

1.2 Problem Statement

The core problem addressed in this thesis can be stated informally as follows: existing deep learning approaches to ASAG are not good enough for reliable deployment in high-stakes educational settings, and the reasons for this shortfall are well-defined and addressable through targeted architectural innovations.

The standard formulation of ASAG is as follows: given a question q, a reference answer r, and a student answer s, the system predicts a score $\hat{y}$ that approximates the human-assigned score y. In similarity-based ASAG, the predicted score is derived primarily from the meaning similarity between r and s, optionally combined with lexical, syntactic, or rubric-derived features. This formulation is intuitive and computationally convenient, but it rests on an assumption-that semantic similarity is a sufficient proxy for grading correctness-that is empirically false in a significant fraction of cases.

The experiments conducted in this thesis reveal four specific ways in which this assumption breaks down:

(1) Pre-trained sentence encoders achieve Pearson correlations of only 0.46–0.56 on standard benchmarks without domain-specific fine-tuning, leaving approximately half the variance in human grading judgements unexplained. This is a surprisingly large gap, given that these models achieve near-human performance on general semantic similarity tasks. The gap suggests that grading requires a form of understanding that goes beyond surface-level semantic similarity-one that is sensitive to conceptual completeness, logical consistency, and the discrete structure of grading rubrics.

(2) Even with architectural improvements (combining SBERT, BERT, and LSTM), the resulting system cannot distinguish between a student who has used the right words and a student who has understood the right concepts. This is the concept incompleteness problem: a student answer that is lexically and semantically similar to the reference answer may still omit critical concepts that are required for full credit. Standard embedding-based systems have no mechanism for detecting this kind of omission, because they operate on the overall semantic similarity of the answer rather than on the coverage of specific conceptual elements.

(3) Negation-bearing answers suffer a 0.27-point QWK degradation compared to non-negation answers, yet this severe failure is hidden when only aggregate metrics are reported. This finding has important practical implications: a system that achieves strong average performance may still be systematically unreliable for a specific and important class of answers, and this unreliability will not be visible in standard evaluation protocols.

(4) Continuous embedding spaces conflate answers that should receive different grades-a phenomenon we term Discrete Boundary Collapse (DBC)-which imposes a structural ceiling on the accuracy of any similarity-based grading system. This is a fundamental geometric problem: the continuous, smooth nature of embedding spaces is inherently mismatched with the discrete, step-function nature of academic grading, and no amount of fine-tuning can fully bridge this gap without explicit architectural intervention.

The central problem addressed by this thesis is therefore: How can deep learning architectures for ASAG be improved to move beyond pure meaning similarity and achieve grading that is sensitive to answer structure, concept coverage, logical consistency, and the discrete nature of academic scoring?

1.2.1 The Core Challenges

The three core challenges described below are not independent-they interact and compound one another in ways that make the overall grading problem significantly harder than any single challenge in isolation. Understanding each challenge individually, however, is necessary before considering how they can be addressed together.

The length discrepancy is perhaps the most immediately visible challenge when examining ASAG datasets. Student answers vary enormously in length relative to reference answers. A one-size-fits-all embedding model is biased by verbosity: short answers lack sufficient context for meaningful comparison, while long answers dilute key concepts in tangential content. A student who provides a one-word answer ("recursion") and a student who provides a two-paragraph explanation may both deserve high grades, but a model that computes the cosine similarity between their answer embeddings and the reference answer embedding will treat them very differently. No single model configuration handles the full spectrum of answer lengths effectively, because the optimal grading strategy is fundamentally different for answers of different lengths.

The correctness gap is a more subtle but equally important challenge. Similarity scores capture topical overlap but not conceptual correctness. A student who uses the right vocabulary but omits a critical concept, or who includes the right concept but negates it, may receive a score indistinguishable from a fully correct answer. This is not a failure of the embedding model-it is a fundamental limitation of the similarity-based grading paradigm. Negation is especially dangerous because it can completely reverse factual correctness while preserving strong lexical and semantic overlap. A sentence like "the

algorithm does not run in polynomial time" shares 8 out of 9 content words with "the algorithm runs in polynomial time," and a standard SBERT model will assign these sentences a cosine similarity of approximately 0.91, despite the fact that they express opposite propositions.

The geometric flaw is the deepest and most fundamental of the three challenges. Continuous embedding spaces are inherently smooth-small input changes produce small output changes. Academic grading, however, is inherently discontinuous: a single missing concept or a single negation can cause a drastic score change. This fundamental mismatch between continuous representation and discrete grading creates a zone of ambiguity near grade boundaries where conceptually distinct answers receive indistinguishable embeddings. A model that is trained to minimise mean squared error in this smooth embedding space will systematically avoid making confident predictions near grade boundaries, leading to a systematic bias toward the centre of the grade range and a corresponding degradation in boundary discrimination accuracy. This is the Discrete Boundary Collapse problem, and it is the subject of the fourth research objective of this thesis.

1.3 Research Questions

This thesis addresses the following research questions, each of which targets a specific dimension of the ASAG problem identified in the preceding analysis. The questions are ordered to reflect the logical progression of the research: from establishing what current methods can achieve, to understanding why they fail, to proposing and evaluating targeted improvements.

RQ1: How effective are pre-trained embedding models for ASAG without domain-specific fine-tuning, and what are the practical limits of similarity-based grading?

This question establishes the performance ceiling of the current state of the art as a baseline, and provides a rigorous empirical foundation for the subsequent research objectives.

RQ2: Can a length-adaptive architecture that routes student answers through specialised processing pipelines improve grading accuracy across varying answer lengths?

This question directly addresses the length discrepancy challenge identified in Section 1.2.1, by investigating whether tailoring the grading strategy to the structural properties of each answer can overcome the limitations of monolithic models.

RQ3: Does explicit concept extraction, alignment, and contradiction-aware gating improve grading reliability beyond what semantic similarity alone can achieve?

This question targets the correctness gap, asking whether a system that explicitly reasons about conceptual coverage and negation can outperform one that relies solely on embedding similarity.

RQ4: Can the geometric limitations of continuous embedding spaces be formally characterised and reduced to improve discrete grade assignment?

This question addresses the geometric flaw, investigating whether a training strategy that explicitly targets grade boundary discrimination can reshape the embedding space to align with the discrete structure of academic grading.

1.4 Research Objectives

To answer the research questions, this thesis pursues four research objectives. Each objective is tightly coupled to a specific research question and a specific failure mode identified in the problem analysis. The objectives are designed to be cumulative: each builds on the findings of the previous one, and together they form a coherent research programme that addresses the ASAG problem from multiple complementary angles.

RO1: To evaluate and establish the baseline effectiveness of pre-trained embeddings (Word2Vec, GloVe, BERT, USE) for ASAG on the Mohler benchmark dataset.

This objective addresses RQ1 by providing a rigorous, statistically reliable evaluation of the current state of the art. The evaluation uses a 1000-iteration bootstrap

resampling protocol to eliminate the variance introduced by single train–test splits, which is a widespread methodological problem in the ASAG literature.

RO2: To design a length-adaptive, universal architecture capable of evaluating answers of varying lengths using targeted NLP techniques (BM25, TF-IDF, SBERT, BERT, Bi-LSTM) without extensive domain-specific fine-tuning.

This objective addresses RQ2 by proposing and evaluating a routing-based architecture that partitions the answer space into five length bands and applies a specialist sub-model to each band, thereby tailoring the grading strategy to the statistical properties of each answer length range.

RO3: To develop a concept-aware grading mechanism that explicitly extracts reference concepts, computes concept coverage, detects negations, and penalises missing or contradictory content.

This objective addresses RQ3 by proposing a Siamese BERT architecture augmented with explicit concept coverage features and a Negation-Sensitive Score Calibration (NSSC) module, which together enable the system to reason about conceptual completeness and logical consistency in a way that pure embedding similarity cannot.

RO4: To formalise Discrete Boundary Collapse (DBC) as a geometric failure mode in embedding-based ASAG, introduce diagnostic metrics (Collapse Rate, Collapse Severity), and propose a term-weighting mitigation strategy.

This objective addresses RQ4 by providing both a formal characterisation of the problem and a practical solution: a boundary-aware training strategy (BAT-TW-BERT) that constructs hard boundary-crossing triplets weighted by their proximity to grade thresholds, explicitly training the model to respect the discrete structure of the grading rubric.

1.5 Scope of the Work

1.5.1 In Scope

This thesis focuses on the following: free-text English short answers (phrases to several sentences); reference-based grading where one or more model answers are provided; regression and similarity-based score prediction with concept coverage and contradiction awareness; embedding-based and deep learning-based grading architectures; evaluation on the ASAG2024 meta-benchmark and its constituent datasets (Mohler, SciEntsBank, Beetle, SemEval-2013, ASAP-SAS, CSD). The focus on English-language datasets reflects both the availability of high-quality annotated data and the current limitations of the concept extraction pipeline, which relies on English-specific NLP tools. The restriction to reference-based grading is a deliberate choice that allows the system to be evaluated on well-defined benchmarks with clear ground truth, rather than on the more subjective and harder-to-evaluate open-ended grading task.

1.5.2 Out of Scope

The following are explicitly outside the scope of this work: essay grading (long-form writing assessment involving style, coherence, and argumentation); code grading (program correctness evaluation); mathematical expression grading (symbolic equivalence checking); multilingual grading (the proposed concept extraction module is designed for English); and real-time deployment optimisation (systematic latency benchmarking is noted as future work). These exclusions are not limitations of the research approach but deliberate boundary decisions that allow the thesis to achieve depth and rigour within a well-defined problem space. Each of these excluded areas represents a substantial research programme in its own right, and addressing them would require different datasets, different evaluation frameworks, and different architectural choices than those developed here.

1.6 Research Contributions

This thesis makes five principal contributions to the field of ASAG. Taken together, these contributions represent a significant advance over the state of the art: they address the most critical failure modes of existing systems, they provide new diagnostic tools for characterising system performance, and they establish new benchmark results on the most comprehensive ASAG evaluation corpus currently available.

Contribution 1: A comprehensive baseline evaluation of off-the-shelf transfer learning models (Word2Vec, GloVe, BERT, USE) for ASAG, establishing their efficacy without domain-specific fine-tuning and providing a reproducible reference point for future research.

The evaluation uses a 1000-iteration bootstrap resampling protocol that provides statistically reliable performance estimates and confidence intervals, addressing a widespread methodological weakness in the ASAG literature where single-split evaluations produce results that are difficult to compare across studies.

Contribution 2: A universal length-adaptive ASAG architecture that classifies student answers by relative length and routes them through targeted processing pipelines, achieving Pearson correlation of 0.90 and RMSE of 0.18 on benchmark datasets.

The architecture is designed to be practical as well as accurate: the majority of answers (83%) are processed by lightweight models that require no GPU inference, making the system feasible for large-scale deployment without specialised hardware.

Contribution 3: A concept-aware Siamese grading mechanism with term-weighted pooling, explicit concept alignment, concept coverage scoring, and contradiction-aware NLI gating, achieving 17–18% relative improvement in Pearson correlation with interpretable intermediate outputs.

The interpretability of the concept coverage scores is a particularly valuable property for educational deployment, as it allows the system to provide students with specific, actionable feedback about which concepts they covered and which they missed.

Contribution 4: Negation-Sensitive Score Calibration (NSSC), a lightweight post-scoring module that raises negation-specific QWK by +0.22 points and reduces negation MAE by 42% without degrading non-negation performance.

The NSSC module is designed as a post-hoc calibration layer that can be applied to any base ASAG model without retraining, making it a practical tool for improving existing deployed systems without requiring a full retraining cycle.

Contribution 5: The formalisation of Discrete Boundary Collapse (DBC) as a geometric failure mode, with novel diagnostic metrics (Collapse Rate, Collapse Severity) and a term-weighting mitigation strategy (BAT-TW-BERT) that reshapes embedding geometry to align with discrete grading requirements.

The BCR metric introduced in this contribution is a useful complement to standard RMSE and Spearman ρ metrics for any ASAG system that produces discretised grade outputs, and its adoption by the research community would enable more meaningful comparisons across studies.

1.7 Thesis Organisation

The rest of this thesis is structured as follows:

Chapter 2 - Literature Review: Reviews the evolution of ASAG research across three paradigms: rule-based approaches, classical machine learning, and deep learning with pre-trained representations. The chapter further covers recent advances including large language models and the ASAG2024 benchmark, discusses negation handling and fairness in automated grading, and concludes by identifying the four research gaps that motivate this thesis.

Chapter 3 - Background and Preliminaries: Provides the technical foundations required for the proposed architectures. Topics covered include word and sentence embeddings (Word2Vec, GloVe, BERT, SBERT, USE), the transformer architecture (self-attention, multi-head attention, pre-training and

fine-tuning), Siamese and contrastive learning (contrastive loss, triplet loss), evaluation metrics (Spearman ρ, QWK, RMSE), BM25 retrieval, and the spaCy NLP pipeline.

Chapter 4 - Problem Statement and Proposed Framework: Formally defines the ASAG problem and its structural constraints. Characterises the four failure modes identified in this thesis - P1: Length Discrepancy, P2: Concept Incompleteness, P3: Negation Insensitivity, and P4: Discrete Boundary Collapse - and presents the multi-architecture framework that assigns each failure mode to a dedicated research objective. Concludes with the research design summary.

Chapter 5 - Datasets and Experimental Setup: Describes all datasets used in this thesis: the Mohler dataset (grade distribution analysis) and the ASAG2024 benchmark (length–grade correlation, per-source question statistics). Defines the baseline systems, evaluation metrics, and experimental protocol applied consistently across all proposed architectures.

Chapter 6 - Pre-trained Embedding Baseline Study: Establishes a statistically reliable baseline by evaluating four pre-trained embedding strategies (Word2Vec, GloVe, BERT, USE) on the Mohler dataset using a 1000-iteration bootstrap evaluation protocol. Presents results, confidence intervals, and analysis that motivate the architectural choices in subsequent chapters.

Chapter 7 - Proposed Architecture 1: Length-Adaptive Routing ASAG: Presents the first proposed architecture, which addresses P1 (Length Discrepancy) by routing student answers through five length-specific sub-models. Covers the theoretical justification, length band design, routing mechanism, training procedure, and experimental results on the ASAG2024 benchmark. Also includes the transition justification for adopting BERT as the base encoder in all proposed architectures.

Chapter 8 - Proposed Architecture 2: Concept-Aware Siamese Grading with NSSC: Describes the second proposed architecture, which addresses P2 (Concept

Incompleteness) and P3 (Negation Insensitivity). The Siamese BERT encoder with a concept coverage feature is combined with a Negation-Sensitive Score Calibration (NSSC) module. Covers the architecture overview, contrastive training objective, negation detection, NSSC hyperparameter tuning, results, and error analysis.

Chapter 9 - Proposed Architecture 3: Boundary-Aware Training with BAT-TW-BERT: Presents the third proposed architecture, which addresses P4 (Discrete Boundary Collapse). Formally characterises DBC, introduces boundary-aware triplet mining, triplet weighting, and the combined MSE and weighted triplet training objective. Presents results on the Mohler dataset including qualitative analysis of boundary discrimination improvements.

Chapter 10 - Results and Comparative Analysis: Provides a unified performance comparison of all proposed architectures against baseline systems, including improvement decomposition by research objective, per-source analysis, ablation study of the combined system, statistical significance testing, qualitative error analysis, comparison with the state of the art, and computational cost analysis.

Chapter 11 - Consolidated Architecture Summary: Offers a consolidated view of all three proposed architectures, summarising their individual contributions and collective design principles. Presents a unified comparison table and discusses cross-architecture design principles that underpin the modularity and composability of the proposed system.

Chapter 12 - Conclusion and Future Work: Summarises the research contributions and addresses each research objective. Discusses theoretical contributions, practical implications for educational deployment, and limitations of the proposed system. Outlines directions for future work and closes with concluding remarks on the broader significance of the thesis.

CHAPTER 2

Literature Review

2.1 Overview

The field of Automatic Short Answer Grading (ASAG) has evolved over three decades from simple keyword-matching systems to sophisticated deep neural network architectures. This evolution has not been linear: each new paradigm has introduced fundamentally different assumptions about how language meaning should be represented and compared, and each has brought both new capabilities and new limitations. This chapter traces this evolution across three paradigms: rule-based systems (Section 2.3), classical machine learning approaches (Section 2.4), and neural network-based methods (Section 2.5). We then review recent advances including large language models (Section 2.6), negation handling in NLP (Section 2.7), fairness and bias (Section 2.8), and conclude by identifying the research gaps that motivate this thesis (Section 2.9).

Understanding this historical progression is important not just as background, but as a way of appreciating why the specific challenges addressed in this thesis-length discrepancy, concept incompleteness, negation insensitivity, and discrete boundary collapse-have persisted despite decades of research. As will become clear, each generation of ASAG systems has addressed some of the limitations of its predecessors while inadvertently introducing new ones, and the four failure modes addressed in this thesis represent the residual challenges that remain after the most recent generation of transformer-based approaches.

Table 2.1 provides a high-level comparison of the three paradigms along key dimensions.

Table 2.1: Comparison of ASAG paradigms along key dimensions.

Paradigm	**Period**	**Representation**	**Strength**	**Weakness**
Rule-based	2000–2013	Lexical / logical	Interpretable	Brittle, no generalisation
Classical ML	2013–2018	Sparse features	Domain-adaptable	Feature engineering cost
Deep learning	2018–now	Dense embeddings	Semantic richness	Data hungry, opaque

Table 2.2 provides a comprehensive summary of all reviewed works. Papers are grouped by research paradigm.

Table 2.2: Summary of reviewed works in Automatic Short Answer Grading and related NLP fields.

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
Paradigm 1 — Rule-Based Approaches *(2000–2013)*					
[76]	Leacock & Chodorow (2003)	C-rater: lexical matching + morphological analysis	First large-scale ASAG system at ETS; introduced concept-based scoring for short constructed responses	ETS internal	Effective on constrained Q&A; limited paraphrase handling
[128]	Sukkarieh & Blackmore (2009)	AutoMark: pattern templates + keyword matching	Extended C-rater with acceptable-phrasing templates; improved surface variation coverage	Cambridge Assessment	High precision on templated questions; poor scalability across question banks
[5]	Bailey & Meurers (2008)	Textual entailment for ASAG	Framed ASAG as recognising textual entailment (RTE); connected ASAG to broader NLP entailment literature	Reading comprehension corpus	High precision on well-formed answers; asymmetric entailment requires extensions

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
[93]	Meurers et al. (2011)	Dependency graph alignment	Parsed answers into dependency graphs; assessed correctness via graph alignment algorithms	German reading comprehension	High precision on formal text; brittle on informal student writing
Paradigm 2 — Classical Machine Learning *(2011–2018)*					
[98]	Mohler et al. (2011)	Lexical + semantic features; linear regression	Introduced the Mohler benchmark dataset; established feature engineering paradigm for ASAG	Mohler (CS undergrad)	Pearson $r = 0.518$; widely cited baseline
[39]	Dzikovska et al. (2013)	SemEval-2013 Task 7 shared task (19 systems)	Provided SciEntsBank & Beetle datasets; framed ASAG as 3-/5-way classification; attracted 19 systems	SciEntsBank, Beetle	Best system: ≈0.71 accuracy (5-way); revealed limits of lexical features

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
[129]	Sultan et al. (2016)	Soft word alignment via word embeddings; single-feature regression	Showed a carefully designed single alignment feature can outperform complex multi-feature systems	Mohler, SciEntsBank	Pearson $r = 0.592$ on Mohler; fast inference
[117]	Saha et al. (2018)	SVM with sentence-level vs. token-level features	Systematic comparison showing sentence embeddings consistently outperform token-level features for ASAG	Mohler, SciEntsBank	F1 = 0.801 (SciEntsBank); foreshadowed sentence encoder dominance
Paradigm 3 — Deep Learning & Pre-trained Representations				*(2018–present)*	
[97]	Mikolov et al. (2013)	Word2Vec: CBOW and Skip-gram word embeddings	Introduced dense word vector representations; enabled semantic similarity via cosine distance	Google News corpus	300-dim vectors; strong semantic clustering; mean-pooling loses word order

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
[106]	Pennington et al. (2014)	GloVe: global word-word co-occurrence matrix factorisation	Complemented Word2Vec with corpus-level statistics; better domain match for academic text	Wikipedia + Gigaword	300-dim vectors; improved over Word2Vec on analogy tasks
[18]	Camus & Bhatt (2020)	Bi-LSTM with attention + BERT embeddings	Showed recurrent architectures with attention outperform feature-engineered baselines on Mohler	Mohler	Pearson $r = 0.631$; sensitive to answer length; needs large training data
[36]	Devlin et al. (2019)	BERT: bidirectional transformer pre-training (MLM + NSP)	Paradigm shift in NLP; contextual token embeddings; fine-tuned BERT achieved SOTA across NLP tasks	Wikipedia + BooksCorpus	SOTA on GLUE, SQuAD; fine-tuned BERT displaced all prior ASAG methods
[44]	Filighera et al. (2022)	Fine-tuned BERT with hierarchical attention + prototype answers	First systematic demonstration that fine-tuned BERT surpasses all prior ASAG approaches on Mohler & SciEntsBank	Mohler, SciEntsBank	Pearson $r = 0.651$ (Mohler); F1 = 0.842 (SciEntsBank)

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
[113]	Reimers & Gurevych (2019)	SBERT: Siamese BERT fine-tuned on NLI data	Produced sentence-level embeddings comparable in $O(1)$; solved BERT's $O(n^2)$ similarity bottleneck	NLI, STS benchmarks	SOTA on STS tasks; `all-mpnet-base-v2` variant used in this thesis
[21]	Cer et al. (2018)	USE: multi-task transformer encoder (Q&A, forums, Wikipedia)	512-dim sentence embeddings with strong Q&A bias; best frozen encoder in RO1 baseline ($\rho = 0.561$)	Multi-task web corpus	Spearman $\rho = 0.561$ on Mohler (frozen); outperforms BERT in static regime
Paradigm 4 — Recent Advances & Benchmarks				*(2020–present)*	
[15]	Brown et al. (2020)	GPT-3: 175B-parameter autoregressive LLM; few-shot prompting	Demonstrated that large-scale pre-training enables few-shot NLP without fine-tuning	Diverse web corpora	Competitive zero-shot; high inference cost; inconsistent grading across runs

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
[100]	OpenAI (2023)	GPT-4: multimodal LLM; chain-of-thought reasoning	Advanced reasoning and instruction-following; strong zero-shot ASAG via prompting	Internal benchmark	Near-human on complex Q&A; >500 ms/sample; hallucination risk
[142]	Wang et al. (2024)	LLM-based ASAG: GPT-4 zero-shot and few-shot evaluation	Systematic evaluation of LLMs for ASAG; showed GPT-4 competitive without fine-tuning	Mohler, SciEntsBank	F1 = 0.871 (SciEntsBank); strong on deep-reasoning questions
[74]	Künnecke et al. (2024)	Hybrid BERT + explicit concept matching	Combined neural meaning similarity with symbolic concept matching; complementary gains confirmed	SciEntsBank	F1 = 0.858; SOTA on SciEntsBank at publication
[94]	Meyer et al. (2024)	ASAG2024: unified benchmark aggregating 6 datasets	Most comprehensive ASAG benchmark (16,626 samples, 6 sources); standardised preprocessing and evaluation	SciEntsBank, Beetle, Mohler, + 3 others	Primary evaluation corpus for this thesis; enables cross-domain comparison

Continued on next page

Continued from previous page

Ref	Authors (Year)	Method / Architecture	Key Contribution	Dataset(s)	Performance / Remarks
Paradigm 5 — Negation Handling in NLP *(2020–present)*					
[42]	Ettinger (2020)	Psycholinguistic diagnostic probes for BERT	Showed BERT fails negation probes; "can fly" and "cannot fly" produce near-identical representations	Psycholinguistic test suite	SBERT cosine sim ≈ 0.91 for negated pairs; motivates NSSC module in this thesis
[19]	Cao et al. (2025)	Negation-aware training objectives for educational NLP	Explicit negation supervision improves model performance on negation-bearing answers in ASAG	Educational NLP corpus	Substantial F1 gain on negation subset; directly motivates RO3 NSSC calibration

2.2 ASAG System

ASAG is more than just an algorithm-it is a pipeline. Building a working system requires thinking carefully about datasets, evaluation metrics, and how each component feeds into the next. A grading system that performs well in isolation on a single dataset may fail when deployed in a new domain, when faced with answers of unusual length, or when the reference answer uses different vocabulary than the training data. This pipeline view is well established in natural language processing research, such as relation extraction, template filling, and efficient information extraction [17]. Figure 2.1 shows the basic structure of an ASAG system development pipeline.

The pipeline perspective also highlights the importance of evaluation design. A system that is evaluated on a single train–test split may produce results that are not reproducible due to the high variance of small datasets, while a system that is evaluated only on aggregate metrics may appear to perform well while failing systematically on specific types of answers. These evaluation challenges are addressed directly in the research methodology of this thesis, through the use of bootstrap resampling and the introduction of the Boundary Confusion Rate metric.

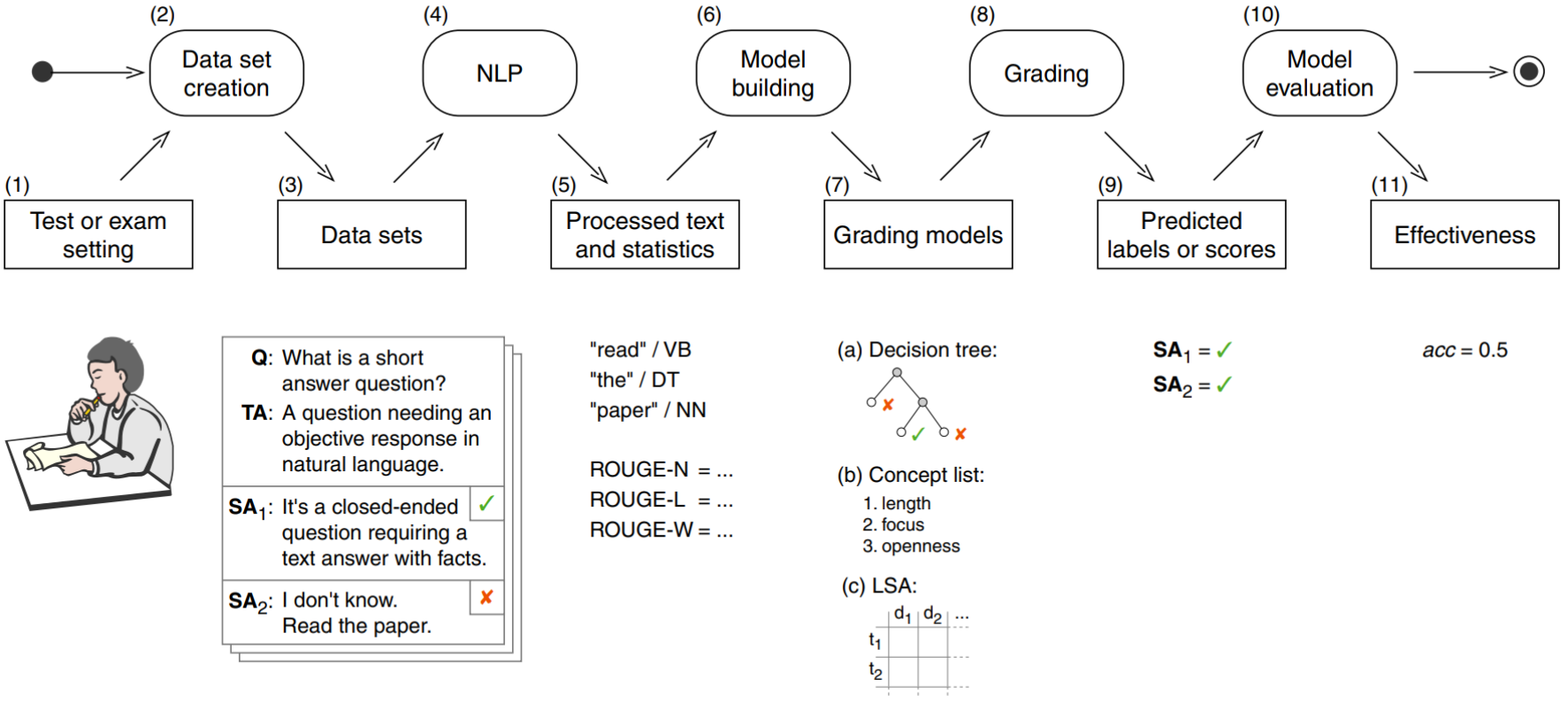

FIGURE 2.1: An ASAG system development pipeline represented by artifacts and processes [17].

2.3 Rule-Based Approaches

2.3.1 Early Keyword and Pattern Matching

The earliest ASAG systems relied on explicit keyword matching between student answers and reference answers. The C-rater system [76], developed at Educational Testing Service (ETS), used a combination of lexical matching and morphological analysis to identify whether key concepts appeared in student responses. While effective for highly constrained questions (e.g., fill-in-the-blank), C-rater struggled with paraphrase recognition and synonym handling.

AutoMark [128], developed at Cambridge Assessment, extended the keyword method with pattern templates that specified acceptable phrasings for each key concept. This improved handling of surface variation but required substantial manual effort to create templates for each question, limiting scalability.

2.3.2 Logical Form and Semantic Graph Approaches

More sophisticated rule-based systems represented answers as logical forms or semantic graphs and used formal inference to assess correctness. Meurers et al. [93] developed a system that parsed student answers into dependency graphs and compared them to reference answer graphs using graph alignment algorithms. While this method achieved high precision on well-formed answers, it was sensitive to parsing errors and struggled with informal student writing.

Bailey and Meurers [5] applied textual entailment techniques to ASAG, treating the task as determining whether the student answer entails the reference answer. This framing connected ASAG to the broader NLP literature on recognising textual entailment (RTE), enabling the use of RTE datasets and models. However, the asymmetric nature of entailment (a student answer may partially entail the reference) required extensions to the standard RTE system.

2.3.3 Limitations of Rule-Based Approaches

Rule-based approaches share three core limitations:

1. **Brittleness:** Rules are designed for specific question types and fail when student answers deviate from anticipated phrasings.
2. **Scalability:** Creating and maintaining rule sets for large question banks requires substantial expert effort.
3. **Generalisation:** Rules learned for one domain (e.g., computer science) do not transfer to other domains (e.g., biology) without substantial manual adaptation.

2.4 Classical Machine Learning Approaches

2.4.1 The Mohler Dataset and Feature Engineering Era

The publication of the Mohler dataset [98] in 2011 marked a turning point in ASAG research by providing a standardised benchmark for evaluating machine learning approaches. Mohler and Mihalcea introduced a system that combined lexical overlap features (word overlap, character n-gram overlap) with semantic similarity features (LSA, knowledge-based similarity) and trained a linear regression model. This work established the feature engineering paradigm that dominated ASAG research from 2011 to 2018.

Later work explored increasingly sophisticated feature sets. Sultan et al. [129] introduced a fast and accurate ASAG system using soft alignment of content words between student and reference answers, achieving strong results with minimal computational overhead. Their system computed alignment scores based on word embeddings and used the alignment score as a single feature for regression, showing that a carefully designed single feature could outperform complex multi-feature systems.

2.4.2 SemEval 2013 Task 7

The SemEval 2013 Task 7 shared task [39] provided the SciEntsBank and Beetle datasets and attracted 19 participating systems. The task framed ASAG as a 3-way or 5-way classification problem (correct/partially correct/incorrect) rather than regression, reflecting the discrete grading conventions used in the BEETLE intelligent tutoring system. Top-attains systems combined lexical features, semantic similarity features, and syntactic features, with the best system achieving approximately 0.71 accuracy on the 5-way task.

The SemEval 2013 shared task revealed several important insights:

- Lexical features (word overlap, n-gram overlap) remained competitive with more sophisticated semantic features.
- Domain-specific features (e.g., physics-specific vocabulary) provided marginal improvements on the Beetle dataset.
- The 5-way classification task (correct/partially correct/contradictory/irrelevant/non-domain) was substantially harder than the 2-way task, with accuracy dropping by approximately 15 percentage points.

2.4.3 Support Vector Machines and Ensemble Methods

Support Vector Machines (SVMs) with rich feature vectors were the dominant method in the classical ML era. Saha et al. [117] conducted a structured comparison of sentence-level versus token-level features for ASAG, finding that sentence-level features (based on sentence embeddings) consistently outperformed token-level features (based on word-level features) when the embedding model was of sufficient quality. Their work foreshadowed the dominance of sentence embedding approaches in the deep neural network era.

Random Forest classifiers [14] were also widely used, particularly for their ability to handle high-dimensional feature vectors and provide feature importance estimates.

Ridge regression [59] and Lasso regression [135] were preferred for the continuous grading task due to their regularisation properties.

2.5 Deep Learning and Pre-trained Representations

2.5.1 Word Embedding Approaches

The introduction of Word2Vec [97] and GloVe [106] in 2013–2014 provided a new foundation for ASAG systems. These models produced dense, low-dimensional word representations that captured semantic relationships through distributional similarity. For ASAG, sentence-level embeddings were obtained by averaging word vectors, and cosine similarity between student and reference embeddings was used as a grading feature.

While word embedding approaches improved over bag-of-words baselines, they suffered from two key limitations: (1) mean pooling discards word order information, proving critical for understanding negation and complex clauses; and (2) static embeddings cannot handle polysemy (words with multiple meanings), leading to incorrect similarity judgments for domain-specific terminology.

2.5.2 Recurrent Neural Networks

Recurrent Neural Networks (RNNs), particularly LSTMs [58] and Bidirectional LSTMs [120], tackles the word order limitation of mean pooling by processing answers sequentially. LSTM-based ASAG systems typically encoded the reference and student answers separately and compared their final hidden states or attention-weighted representations.

Camus and Bhatt [18] showed that a simple Bi-LSTM architecture with attention could outperform feature-engineered baselines on the Mohler dataset, confirming the utility of recurrent architectures for ASAG. However, RNN-based systems were sensitive to

answer length (very long answers caused vanishing gradient problems) and required substantial training data to learn effective representations.

2.5.3 BERT and Contextual Embeddings

The introduction of BERT [36] in 2018 represented a paradigm shift in NLP. BERT's bidirectional transformer architecture, pre-trained on massive text corpora using masked language modelling and next-sentence prediction, produced contextual embeddings that captured both semantic and syntactic information at the word level. fine-tuned BERT models achieved state-of-the-art results on a broad spectrum of NLP tasks, including text classification, question answering, and semantic textual similarity.

For ASAG, BERT-based approaches quickly displaced earlier methods. Filighera et al. [44] showed that fine-tuned BERT substantially outperformed all previous approaches on both SciEntsBank and Mohler, confirming the utility of contextual embeddings for ASAG. Their work also introduced the concept of prototype answer representations, where multiple reference answers are used to capture the diversity of acceptable responses.

2.5.4 Sentence-BERT and Universal Sentence Encoder

While BERT produced excellent word-level representations, computing sentence-level similarity using BERT required a cross-encoder architecture that processed the reference and student answer jointly, making it computationally expensive for large-scale deployment. Sentence-BERT (SBERT) [113] addresses this by fine-tuning BERT on natural language inference data using a Siamese network architecture, producing sentence-level embeddings that could be compared using cosine similarity in constant time.

The Universal Sentence Encoder (USE) [21] took a different method, training a multi-task encoder on diverse data including question-answering, discussion forums, and Wikipedia. USE's training on question-answering data makes it particularly

well-suited for ASAG, where the task involves comparing answers to questions rather than general sentence similarity.

2.6 Recent Advances

2.6.1 Large Language Models for Automatic Short Answer Grading

The introduction of Large Language Models (LLMs) has changed what is practically possible in natural language processing, and short answer grading is no exception. Where traditional supervised models needed a purpose-built classifier for every dataset, an LLM already carries broad linguistic knowledge from pre-training and can, in principle, be pointed at a new grading task with little more than a well-written prompt. This is especially useful in education, where annotated data are rarely abundant. It is not just the scores that have benefited: LLMs are now used to justify a grade, unpack a rubric in plain language, and generate feedback a student can actually act on [32].

It is worth pausing on how different this is from what came before. Feature-engineered classifiers and BERT-style encoders performed well, but only within the domain they were trained on – move to a new subject, question style, or grading rubric, and performance would often drop noticeably. LLMs sidestep this by leaning on instruction following and in-context learning: instead of learning a grading function from labelled examples alone, they can be steered with prompts, reference answers, rubrics, or a handful of worked examples. That single property – adaptability without retraining – is arguably what has drawn so much attention to LLMs in ASAG over the last two to three years.

Research in this space has not moved in one direction so much as branched into several. The earliest work leaned almost entirely on zero-shot and few-shot prompting, simply asking the model to grade and seeing how far its pre-trained reasoning would

carry it [22, 23]. As the limits of that approach became apparent, attention turned to parameter-efficient fine-tuning, which adapts an open-source model to a specific grading task without the cost of full retraining [20]. More recently, retrieval-augmented generation and confidence-aware, human-in-the-loop pipelines have entered the picture, aiming less at raw accuracy and more at reliability and interpretability [28, 32]. Taken together, these three strands suggest a field that is slowly moving away from pure semantic matching and towards grading that can be inspected, questioned, and trusted.

Prompt-based and In-context Grading

Prompting was the natural starting point, if only because it required nothing more than access to a capable model. The setup is simple: the question, a reference answer, grading instructions, and the student's response go into a prompt, and the model returns a score. No training data, no fine-tuning pipeline – which is precisely why this approach appealed to institutions with little labelled data to work with. Yeung et al. [152] put this to the test in an actual classroom setting, deploying a zero-shot framework for introductory higher-education courses and reporting real gains in student motivation and preparedness – a useful reminder that these results are not confined to benchmark leaderboards.

Few-shot prompting was the next logical step: give the model a handful of instructor-graded examples and let it infer the grading criteria implicitly, rather than stating them outright. The evidence on whether this actually helps, though, is not as clean as one might hope. Chamieh et al. [22] tested both zero- and few-shot configurations across three datasets and found that neither came close to a fine-tuned model or the supervised upper bound, especially on items that demanded real reasoning or domain expertise – a result that pushes back against some of the earlier enthusiasm for prompting as a complete solution. Carpenter et al. [20], working on a somewhat different task, found the opposite: few-shot prompting paired with light fine-tuning improved the grading of student explanations. Read together, these two findings suggest that "few-shot works" is

not a universal claim – it depends heavily on how complex the task is, how specialised the domain is, and how much adaptation, if any, sits underneath the prompting.

Prompting also carries some baggage that fine-tuned models mostly avoid. The most frequently cited issue is sheer sensitivity to wording: change the instruction phrasing, reorder the examples, or tweak the formatting, and the same student answer can receive a different score. In a high-stakes assessment context, that kind of variability is hard to defend. There is also the matter of what the model actually knows – since it is drawing on whatever it absorbed during pre-training, it can produce a confident, well-written justification for a score that has little to do with the specific subject matter being graded.

None of this has stopped prompt engineering from becoming a research area in its own right rather than a one-off implementation detail. Instruction prompts, role prompting, rubric-guided prompts, self-reflection, and chain-of-thought reasoning have all been tried as ways of tightening up otherwise loose grading behaviour. Rubric-guided prompting in particular seems to help, presumably because it forces the model to check the response against explicit criteria rather than fall back on rough semantic similarity. Asking the model to explain its reasoning before committing to a score serves a similar purpose, and has the added benefit of giving the instructor something to actually inspect.

Even with these refinements, there is a trade-off that does not go away: prompting is flexible and cheap to set up, but how well it works depends heavily on the quality of the prompt and, as Chamieh et al. [22] make clear, on how demanding the task is in the first place. It was this ceiling, more than anything, that pushed researchers towards adapting the model itself rather than relying purely on instructions – which is where parameter-efficient fine-tuning comes in.

Parameter-efficient Fine-tuning of Large Language Models

Fine-tuning entered the picture largely because prompting, for all its convenience, was proving inconsistent on exactly the items instructors care most about getting right. Parameter-efficient fine-tuning (PEFT) offers a middle path: rather than retraining an

entire model, only a small subset of parameters is updated using labelled grading data. Low-Rank Adaptation (LoRA) and its quantised variant (QLoRA) are what made this affordable in practice, letting institutions adapt a large model without the GPU budget that full fine-tuning would demand.

The fact that strong open-source models – LLaMA, Mistral, Gemma, Qwen, among others – are now freely available has made this route considerably more attractive. Unlike closed commercial models, these can be inspected, modified, and kept entirely in-house, which matters both for data privacy and for tailoring a model to a specific institution's grading conventions. Once fine-tuned on annotated student responses, a model can pick up course-specific vocabulary and marking habits that a generic prompt would likely miss. That said, the picture is not uniformly rosy: comparisons of fine-tuning strategies have found that QLoRA-adapted smaller models such as LLaMA-3-8B can start out performing quite poorly on ASAG tasks, and it takes deliberate synthetic-data augmentation to close the gap with larger, more heavily resourced commercial fine-tunes. The takeaway is that data quality matters at least as much as the adaptation method itself, particularly for institutions working with limited compute.

One genuine advantage of PEFT is that it does not require an enormous model to get reasonable results. Several studies have shown that comparatively compact models, once properly adapted, can hold their own against much larger systems – good news for institutions where cost, compute, or data-residency rules rule out commercial cloud APIs.

Fine-tuning is not a free lunch, though. It still needs good annotated data, and collecting that data consistently is neither quick nor cheap. A model tuned for one subject or one instructor's grading style may need retuning before it transfers to another, which limits how far a single fine-tuned model can generalise. There is also the risk that fine-tuning simply bakes in whatever biases exist in the training annotations – a real concern when different instructors do not always agree on what a "correct" answer looks like. All of which points to a broader lesson: adapting the model is only half the problem. What the model has access to at inference time matters just as much.

That observation is really what motivates the next line of work. Rather than expecting a model to have memorised everything relevant during training, why not let it look things up when it needs to? This is the premise behind retrieval-augmented grading.

Retrieval-Augmented and Knowledge-grounded Grading

Retrieval-Augmented Generation (RAG) has become one of the more promising ways to make LLM-based grading more dependable. Instead of relying purely on what the model learned during pre-training or fine-tuning, RAG pulls in relevant material – reference answers, rubrics, lecture notes, textbook passages, previously graded responses – at the moment of grading. Chu et al. [28] built exactly this kind of system, GradeRAG, retrieving instructional material and expert-annotated exemplars alongside the rubric, and found that grounding the model's decision in retrieved evidence cut down on unsupported reasoning compared to a naive prompting baseline.

The appeal here is fairly direct: LLMs, however capable, will occasionally produce a fluent explanation that has nothing to do with reality – what is usually called hallucination – and this is a particularly bad failure mode in an assessment context, where a wrong-but-confident justification can quietly mislead an instructor or student. Giving the model something concrete to point to during grading – an actual rubric line, an actual reference passage – reduces how much it has to invent.

Some of the newer systems go further than just adding context to the prompt. They retrieve the relevant concepts individually and have the model check the student's answer against each one in turn, rather than producing a single holistic score. The output then looks less like "3/5" and more like a breakdown: this concept was present, this one was missing, this one was only partially addressed – which is closer to how a human grader actually thinks it through, and easier for an instructor to sanity-check. Cong et al. [32] extend this idea by pairing retrieval-grounded grading with calibrated confidence estimates, showing that combining the two gives a better sense of which cases genuinely need a human to step in, compared to relying on either signal alone.

A related line of work uses retrieval not for background material but for choosing few-shot examples themselves – selecting instructor-graded responses that are semantically close to the one being graded, rather than picking demonstrations at random. This tends to give the model more relevant guidance exactly where it is needed, and helps with responses that are unusually worded or only partly correct.

Graph-based retrieval is a newer variant worth mentioning, even if the evidence base is still thin. Instead of treating retrieved passages as independent chunks of text, these methods try to model how concepts relate to one another, which in theory should help with answers that require connecting several ideas rather than just matching a keyword. It is early days for this direction, but the initial results are encouraging enough to keep watching.

Stepping back, retrieval-augmented grading marks a genuine shift in how the field thinks about the problem. Earlier work assumed the fix lay in a better prompt or a better fine-tune – in other words, in the model itself. RAG-based approaches instead accept that good grading also depends on having the right external knowledge on hand. That said, retrieval is not a silver bullet: if the retrieved material is irrelevant or incomplete, the grading decision suffers regardless of how good the underlying model is, and having the right concepts in hand does not automatically mean the model reasons about them correctly. That gap – between retrieving the right knowledge and actually reasoning over it well – is where a good deal of current research attention has landed.

Comparative Analysis of LLM-based ASAG Paradigms

Table 2.3 lays the three paradigms discussed above – prompting, parameter-efficient fine-tuning, and retrieval augmentation – side by side, along with representative studies, their core mechanics, what they tend to get right, and where they tend to fall short. No single paradigm wins outright. Prompting is cheapest to deploy but the least reliable once the task requires real reasoning; fine-tuning is the most consistent within a given domain but demands annotated data and does not transfer well across subjects; retrieval augmentation sits somewhere in between, improving transparency and

reducing unsupported reasoning without the cost of retraining, but only to the extent that the retrieval itself is good. This is precisely the gap that motivates the hybrid, concept-aware direction taken in this thesis, where negation-sensitive score calibration and structured concept comparison are combined rather than betting on any one paradigm alone.

TABLE 2.3: Comparative analysis of LLM-based ASAG paradigms

Paradigm	**Representative Studies**	**Core Technique**	**Reported Strengths**	**Principal Limitations**
Zero-/few-shot prompting	[22, 23, 152]	Direct scoring via question, reference answer, and rubric embedded in the prompt; no gradient updates	No annotated training data required; rapid deployment across courses; effective on well-known, general-domain content	High sensitivity to prompt formulation; weak on reasoning-intensive or domain-specific items; inconsistent across repeated runs
Parameter-efficient fine-tuning (LoRA/QLoRA)	[20]	Low-rank adaptation of open-source LLMs (LLaMA, Mistral, Qwen families) on annotated grading data	Captures course-specific terminology and marking schemes; enables use of smaller, locally deployable models; strong in-domain consistency	Requires high-quality annotated data; limited cross-domain generalisation; may inherit annotator bias
Retrieval-augmented grading (RAG)	[28, 32]	Retrieval of reference answers, rubrics, or exemplar responses to ground the LLM's grading decision	Reduces hallucinated justifications; improves transparency via concept-level comparison; supports human-in-the-loop deferral via confidence estimation	Performance bounded by retrieval quality; does not guarantee correct reasoning over retrieved concepts; added system complexity

2.6.2 The ASAG2024 Benchmark

Meyer et al. [94] introduced the ASAG2024 benchmark, which aggregates six previously published ASAG datasets into a unified resource with 16,626 samples. The

benchmark provides standardised preprocessing, a unified grade scale, and official evaluation protocols. ASAG2024 is the most thorough ASAG benchmark currently available and represents the primary evaluation corpus for this thesis.

2.6.3 Hybrid and Multi-Task Approaches

Kunnecke et al. [74] introduced a hybrid method combining neural meaning similarity with explicit concept matching, showing that the two approaches are complementary. Their system achieved leading results on SciEntsBank at the time of publication, confirming the value of combining neural and symbolic components.

2.7 Negation Handling in NLP

2.7.1 Negation in Pre-trained Models

Negation is a well-known challenge for pre-trained language models. Ettinger [42] showed that BERT fails on negation probes, producing similar representations for “the bird can fly” and “the bird cannot fly”. This limitation is particularly problematic for ASAG, where negation errors (a student negating a key claim) should result in a low grade.

Cao et al. [19] recently introduced negation-aware training objectives for educational NLP, showing that explicit negation supervision can substantially improve model performance on negation-bearing answers. Their work motivates the NSSC module proposed in this thesis (Chapter 8).

2.7.2 Negation Scope Detection

Negation scope detection - identifying which parts of a sentence are negated - is a related NLP task with a rich literature. The ConanDoyle-neg corpus and the SFU

Review Corpus provide annotated negation scope data that have been used to train negation scope detectors. For ASAG, full negation scope detection is likely unnecessary; detecting the presence of negation markers and their proximity to key concepts is sufficient for most grading decisions.

2.8 Fairness and Bias in ASAG

Automated grading systems can perpetuate or amplify biases present in their training data. If training data reflects historical grading biases (e.g., higher grades for students from certain demographic groups), an ASAG system trained on this data may replicate these biases at scale. Ensuring fairness in ASAG is Therefore, an important research direction, though it is beyond the scope of this thesis.

Recent work has examined demographic bias in ASAG systems, finding that performance varies across student populations with different writing styles, language backgrounds, and educational histories. Developing bias-aware ASAG systems that achieve equitable performance across demographic groups remains an open challenge.

2.9 Research Gaps

Based on the literature review, four research gaps are identified that motivate this thesis:

1. **Gap 1 - Unreliable Evaluation:** Most prior ASAG evaluations use single train–test splits, producing results with high variance. A statistically reliable evaluation protocol is needed to enable reliable comparison of ASAG systems. *tackles by RO1.*

2. **Gap 2 - Length Blindness:** Existing ASAG systems treat all answers identically regardless of their length relative to the reference answer. The strong correlation between answer length and grade ($r = 0.412$ on ASAG2024) suggests that length-adaptive architectures would substantially improve yields. *tackles by RO2.*

3. **Gap 3 - Concept and Negation Handling:** Existing neural ASAG systems are insensitive to concept incompleteness and negation, two of the most common student error types. Lightweight mechanisms for explicit concept tracking and negation calibration are needed. *tackles by RO3.*

4. **Gap 4 - Boundary Discrimination:** The Discrete Boundary Collapse problem has not been formally characterised or tackles in the ASAG literature. A training strategy that explicitly improving boundary discrimination is needed. *handles by RO4.*

CHAPTER 3

Background and Preliminaries

3.1 Overview

This chapter presents the technical background required to understand the proposed methods. We cover word and sentence embeddings (Section 3.2), the transformer architecture (Section 3.3), Siamese and contrastive learning (Section 3.4), evaluation metrics (Section 3.5), BM25 retrieval (Section 3.6), and a brief overview of spaCy's NLP pipeline (Section 3.7).

3.2 Word and Sentence Embeddings

3.2.1 Word2Vec

Word2Vec [97] introduced two neural architectures for learning word representations from large text corpora: the Continuous Bag-of-Words (CBOW) model, which predicts a word from its context, and the Skip-gram model, which predicts context words from a target word. Both architectures are trained using negative sampling to efficiently approximate the softmax over the full vocabulary.

The Skip-gram objective is:

$$\mathcal{L}_{\text{SG}} = \frac{1}{T} \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} | w_t) \tag{3.1}$$

where T is the corpus size, c is the context window size, and $p(w_{t+j}|w_t)$ is approximated using negative sampling. The resulting 300-dimensional vectors encode semantic and

syntactic relationships, famously capturing analogies such as "king − man + woman ≈ queen".

3.2.2 GloVe

GloVe [106] (Global Vectors for Word Representation) takes a different method, training on the global word-word co-occurrence matrix of the corpus. The objective is to learn vectors $\mathbf{w}_i$ and $\tilde{\mathbf{w}}_j$ such that:

$$\mathbf{w}_i^\top \tilde{\mathbf{w}}_j + b_i + \tilde{b}_j = \log X_{ij} \tag{3.2}$$

where X_{ij} is the co-occurrence count of words i and j, and b_i, $\tilde{b}_j$ are bias terms. GloVe combines the advantages of global matrix factorisation (LSA-style) and local context window methods (Word2Vec-style), producing vectors that capture both global semantic structure and local syntactic patterns.

3.2.3 BERT

BERT [36] (Bidirectional Encoder Representations from Transformers) is a pre-trained transformer model that produces contextual word representations. Unlike Word2Vec and GloVe, which produce static embeddings (one vector per word regardless of context), BERT produces context-dependent embeddings where the representation of a word depends on its surrounding context.

BERT is pre-trained on two objectives:

1. **Masked Language Modelling (MLM):** 15% of input tokens are randomly masked, and the model is trained to predict the masked tokens from the surrounding context.

2. **Next Sentence Prediction (NSP):** The model is trained to predict whether two sentences are consecutive in the original corpus.

For sentence-level tasks, BERT uses the representation of the special `[CLS]` token as the sentence representation, or alternatively, the mean of all token representations (mean pooling).

3.2.4 Sentence-BERT (SBERT)

SBERT [113] tackles BERT's limitation for sentence similarity tasks: computing similarity between n sentences using BERT requires $O(n^2)$ forward passes (one for each pair), proving computationally prohibitive for large collections. SBERT fine-tunes BERT using a Siamese network architecture on natural language inference (NLI) data, producing sentence embeddings that can be compared using cosine similarity in $O(1)$ time.

The SBERT training objective for the NLI task uses a softmax classifier over the concatenated sentence representations:

$$\mathbf{u} = \text{mean-pool}(\text{BERT}(s_1)), \quad \mathbf{v} = \text{mean-pool}(\text{BERT}(s_2)) \tag{3.3}$$

$$\hat{y} = \text{softmax}(W_c \cdot [\mathbf{u}; \mathbf{v}; |\mathbf{u} - \mathbf{v}|]) \tag{3.4}$$

The `all-mpnet-base-v2` variant of SBERT, used in this thesis, was trained on over 1 billion sentence pairs and achieves top results on the MTEB benchmark [99].

3.2.5 Universal Sentence Encoder (USE)

USE [21] provides 512-dimensional sentence embeddings trained on a diverse multi-task corpus including web question-answer pairs, discussion forums (Reddit), Wikipedia, news, and translation data. The multi-task training approach combines:

- Skip-thought prediction (predict surrounding sentences)
- Conversational response prediction (predict response to a message)
- Natural language inference classification

USE's training on question-answering data is particularly relevant for ASAG, as it means the model has learned to represent answers in a way that is sensitive to the question context.

3.3 The Transformer Architecture

3.3.1 Self-Attention Mechanism

The transformer architecture [139] is built around the self-attention mechanism, which computes a weighted sum of value vectors based on the compatibility between query and key vectors:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{3.5}$$

where $Q \in \mathbb{R}^{n \times d_k}$, $K \in \mathbb{R}^{n \times d_k}$, and $V \in \mathbb{R}^{n \times d_v}$ are the query, key, and value matrices, and d_k is the key dimension used for scaling. The scaling factor $\frac{1}{\sqrt{d_k}}$ prevents the dot products from becoming too large in high dimensions, which would cause the softmax to have very small gradients.

3.3.2 Multi-Head Attention

Multi-head attention applies h attention functions in parallel, each with different learned projections:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{3.6}$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$. This allows the model to attend to different aspects of the input simultaneously - for example, one head may focus on syntactic relationships while another focuses on semantic relationships.

3.3.3 Position-wise Feed-Forward Networks

Each transformer layer also includes a position-wise feed-forward network applied independently to each position:

$$\text{FFN}(\mathbf{x}) = \text{ReLU}(\mathbf{x}W_1 + \mathbf{b}_1)W_2 + \mathbf{b}_2 \tag{3.7}$$

The inner dimension of the FFN is typically 4 times the model dimension (e.g., 3072 for BERT-base with $d_{\text{model}} = 768$).

3.3.4 Pre-training and Fine-tuning

The BERT pre-training paradigm involves two phases:

1. **Pre-training:** Train the transformer on large unlabelled corpora using self-supervised objectives (MLM, NSP). This phase requires substantial computational resources (BERT-base was trained on 16 TPUs for 4 days) but is achieves once.

2. **Fine-tuning:** Add a task-specific head to the pre-trained transformer model and fine-tune all parameters on labelled task data. Fine-tuning is computationally cheap (typically a few hours on a single GPU) and produces strong attains with relatively few labelled examples.

3.4 Siamese Networks and Contrastive Learning

3.4.1 Siamese Network Architecture

A Siamese network [27] consists of two identical sub-networks (sharing weights) that process two inputs in parallel and produce representations that are compared using a

distance or similarity metric. The key property is weight sharing: the same parameters are used to encode both inputs, ensuring that the representation space is consistent.

For ASAG, the Siamese architecture encodes the reference answer and student answer using the same encoder, producing representations $\mathbf{e}_r$ and $\mathbf{e}_s$ that are compared using cosine similarity:

$$\text{sim}(\mathbf{e}_r, \mathbf{e}_s) = \frac{\mathbf{e}_r \cdot \mathbf{e}_s}{||\mathbf{e}_r|| ||\mathbf{e}_s||} \tag{3.8}$$

3.4.2 Contrastive Loss

The contrastive loss [51] trains the encoder to produce similar representations for semantically related inputs and dissimilar representations for unrelated inputs:

$$\mathcal{L}_{\text{contrastive}} = y \cdot d^2 + (1 - y) \cdot \max(0, m - d)^2 \tag{3.9}$$

where $d = ||\mathbf{e}_1 - \mathbf{e}_2||_2$ is the Euclidean distance, $y \in \{0, 1\}$ is the label (1 for similar, 0 for dissimilar), and m is the margin. More recent variants use the InfoNCE loss [138] or the supervised contrastive loss [68].

3.4.3 Triplet Loss

The triplet loss [119] operates on triplets (a, p, n) where a is the anchor, p is a positive (similar to a), and n is a negative (dissimilar to a):

$$\mathcal{L}_{\text{triplet}} = \max(0,\ d(a, p) - d(a, n) + m) \tag{3.10}$$

The loss encourages the positive to be closer to the anchor than the negative by a margin m. Hard negative mining - selecting negatives that are close to the anchor - is critical for effective triplet training, as easy negatives provide no gradient signal.

3.5 Evaluation Metrics

3.5.1 Spearman Rank Correlation

Spearman's ρ [124] measures the rank correlation between two variables:

$$\rho = 1 - \frac{6 \sum_i d_i^2}{n(n^2 - 1)} \tag{3.11}$$

where d_i is the difference between the ranks of the i-th observation in the two variables. Spearman's ρ is preferred over Pearson's r for ASAG because it is strong to non-linear relationships and outliers.

3.5.2 Quadratic Weighted Kappa

Quadratic Weighted Kappa (QWK) [31] measures agreement between two raters on an ordinal scale, penalizing large disagreements quadratically:

$$\text{QWK} = 1 - \frac{\sum_{i,j} w_{ij} O_{ij}}{\sum_{i,j} w_{ij} E_{ij}} \tag{3.12}$$

where O_{ij} is the observed count of (predicted grade i, gold grade j) pairs, E_{ij} is the expected count under independence, and $w_{ij} = \frac{(i-j)^2}{(K-1)^2}$ is the quadratic weight. QWK is widely used in automated essay scoring and is increasingly adopted in ASAG.

3.5.3 Root Mean Squared Error

RMSE measures the magnitude of prediction errors:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2} \tag{3.13}$$

Weighted RMSE (wRMSE) assigns higher weights to samples near grade boundaries, making it more sensitive to boundary discrimination errors.

3.6 BM25 Retrieval

BM25 [114] (Best Match 25) is a probabilistic retrieval function that ranks documents by relevance to a query. For ASAG, BM25 is used to measure lexical overlap between student and reference answers. The BM25 score is:

$$\text{BM25}(r, s) = \sum_{t \in s} \text{IDF}(t) \cdot \frac{f(t, r) \cdot (k_1 + 1)}{f(t, r) + k_1 \cdot \left(1 - b + b \cdot \frac{|r|}{\text{avgdl}}\right)} \tag{3.14}$$

where $f(t, r)$ is the frequency of term t in the reference answer r, $|r|$ is the answer length, avgdl is the average answer length in the corpus, and $k_1 = 1.5$, $b = 0.75$ are tuning parameters. BM25 is particularly effective for very short answers (Band 1 in the RO2 architecture) where lexical overlap is the primary grading signal.

3.7 spaCy NLP Pipeline

spaCy [61] is an industrial-strength NLP library that provides efficient implementations of tokenisation, part-of-speech tagging, dependency parsing, named entity recognition, and lemmatization. In this thesis, spaCy is used for concept extraction in the RO3 architecture: the `en_core_web_sm` model is used to identify content words (nouns, verbs, adjectives, and proper nouns) after lemmatization and stopword removal.

The spaCy pipeline processes each answer in three steps:

1. **Tokenization:** Split the answer into tokens (words and punctuation).
2. **POS tagging:** Assign part-of-speech tags to each token.
3. **Lemmatisation:** Reduce each token to its base form (e.g., “running” → “run”).

Content words are then identified as tokens with POS tags in {NOUN, VERB, ADJ, PROPN} that are not stopwords.

CHAPTER 4

Problem Statement and Proposed Framework

4.1 Formal Problem Definition

Let $\mathcal{D} = \{(q_i, r_i, s_i, y_i)\}_{i=1}^{N}$ denote a dataset of N instances, where $q_i \in \mathcal{Q}$ is a question, $r_i \in \mathcal{R}$ is the reference (model) answer provided by the instructor, $s_i \in \mathcal{S}$ is the student's answer, and $y_i \in [0, 1]$ is the normalized human-assigned grade. The ASAG task is to learn a function:

$$f_\theta : \mathcal{Q} \times \mathcal{R} \times \mathcal{S} \to [0, 1] \tag{4.1}$$

where $\mathcal{Q}$, $\mathcal{R}$, and $\mathcal{S}$ are the question, reference-answer, and student-answer spaces respectively, θ is the vector of model parameters, and Θ denotes the parameter space over which θ ranges. The function f_θ is parameterized by $\theta \in \Theta$, such that $f_\theta(q_i, r_i, s_i) \approx y_i$ for all $(q_i, r_i, s_i, y_i) \in \mathcal{D}$. The parameters θ are learned by minimizing the expected loss:

$$\theta^* = \arg\min_{\theta \in \Theta} \mathbb{E}_{(q,r,s,y)\sim\mathcal{D}}[\ell(f_\theta(q, r, s),\ y)] + \lambda\Omega(\theta) \tag{4.2}$$

where $\ell(\cdot, \cdot)$ is a loss function (MSE, QWK-based, or a combination), $\Omega(\theta)$ is a regularisation term, $\lambda > 0$ is the regularisation strength, and θ^* denotes the resulting optimal parameter vector.

4.1.1 Structural Constraints

An ideal ASAG function f_θ should satisfy the following structural constraints:

C1 - Length Invariance: f_θ should be invariant to the absolute length of the student answer, conditioned on the relative length ratio $\ell = |s|/|r|$. That is, two answers with the same relative length and the same semantic content should receive the same grade, regardless of their absolute word counts.

C2 - Concept Completeness Sensitivity: f_θ should be sensitive to the coverage of key concepts from the reference answer. If a student answer covers k out of K key concepts, its grade should be approximately k/K, all else being equal.

C3 - Polarity Consistency: f_θ should distinguish between a student answer that affirms a key claim and one that negates it. Formally, if s and $\neg s$ denote affirmative and negated versions of the same claim, and if the reference r affirms the claim, then $f_\theta(q, r, s) > f_\theta(q, r, \neg s)$.

C4 - Boundary Discrimination: f_θ should correctly discriminate between grade bands, particularly for answers near grade boundaries. The probability of a boundary-crossing error should be minimized.

4.2 Identified Failure Modes

The four failure modes described in this section were identified through a combination of theoretical analysis and empirical investigation of the ASAG2024 benchmark. Each failure mode is defined formally, illustrated with a concrete example, and supported by empirical evidence from the dataset analysis. The formal definitions are important because they provide a precise vocabulary for discussing the limitations of existing systems and for evaluating the effectiveness of the proposed solutions. Without such a vocabulary, it is difficult to distinguish between systems that address the same problem in different ways and systems that address fundamentally different problems.

4.2.1 P1: Length Discrepancy

Standard embedding-based ASAG models violate constraint C1 because cosine similarity between mean-pooled embeddings is sensitive to the distribution of words in the answer, which changes with answer length. Short answers that contain only key terms produce embeddings that are close to the reference embedding in high-similarity dimensions, while long answers that elaborate wide-ranging content may produce embeddings that are diluted by the additional material. This length sensitivity is not a bug in the embedding model-it is a natural consequence of how mean pooling works-but it becomes a problem when the model is used for grading, because it introduces a systematic bias that correlates with answer length rather than answer quality.

Empirical evidence: Table 5.4 (Chapter 5) shows Pearson correlations between answer length and grade ranging from 0.267 to 0.412 across ASAG2024 sources, confirming that length is a strong confound. Figure 7.2 shows that a single SBERT model yields $F1 = 0.71$ on very short answers (Band 1) but $F1 = 0.93$ on medium-length answers (Band 3), a difference of 22 percentage points. This is a very large performance gap for a system that is supposed to grade all answers equally, and it confirms that length discrepancy is not a minor calibration issue but a fundamental structural problem that requires an architectural solution.

4.2.2 P2: Concept Incompleteness

Consider the question "Define recursion" with reference answer "A function that calls itself, with a base case to terminate the recursion." A student answer "A function that calls itself" records high cosine similarity to the reference (approximately 0.85 with SBERT) but is missing the critical concept of the base case. An ASAG system that relies solely on embedding similarity will assign this answer a high grade, violating constraint C2. This is not an edge case-it is a systematic pattern that occurs whenever a student provides a partially correct answer that covers the most salient concepts in the reference but omits one or more secondary concepts. The problem is compounded by

the fact that the "missing" concept is often precisely the one that distinguishes a deep understanding from a superficial one, making concept incompleteness a particularly important failure mode from a pedagogical perspective.

Empirical evidence: Analysis of 150 randomly sampled SciEntsBank errors reveals that 38% of over-grading errors (predicted grade > gold grade + 0.2) involve concept incompleteness, confirming that this is the most common failure mode. This finding underscores the importance of developing grading systems that can reason about conceptual coverage rather than relying solely on holistic semantic similarity.

4.2.3 P3: Negation Insensitivity

BERT and its variants encode negation implicitly through the attention mechanism, but this implicit encoding is unreliable [42]. The sentence "The algorithm terminates in polynomial time" and "The algorithm does not terminate in polynomial time" share 8 out of 9 content words and produce SBERT cosine similarity of approximately 0.91, despite having opposite meanings. This near-identical representation is a direct consequence of the way transformer models are trained: they learn to produce similar representations for sentences that appear in similar contexts, and negated sentences typically appear in contexts that are very similar to their non-negated counterparts. The result is a systematic blind spot for negation that cannot be addressed through fine-tuning alone, because the training signal for negation is diluted by the much larger signal from the many non-negated sentence pairs.

Empirical evidence: On the 214-sample negation subset of SciEntsBank, SBERT delivers QWK = 0.52, compared to QWK = 0.80 on the non-negation subset, a gap of 0.28 QWK points. This is the largest performance gap observed across all error categories, confirming that negation insensitivity is the most severe failure mode in the current state of the art. The magnitude of this gap-more than a third of the maximum possible QWK score-suggests that negation handling is not a marginal improvement opportunity but a fundamental capability that any reliable ASAG system must possess.

4.2.4 P4: Discrete Boundary Collapse

When grades are discretized into bands (e.g., A/B/C/D/F), a model that assigns identical scores to answers near grade boundaries fails to discriminate between grade categories. This problem is not captured by RMSE or Spearman ρ, which measure continuous agreement. A model with RMSE = 0.15 may still have a BCR of 0.30, meaning 30% of answers are assigned to the wrong grade band. The practical consequence of this failure mode is significant: in a grading system that assigns letter grades based on numerical scores, a BCR of 0.30 means that nearly one in three students receives the wrong letter grade, even if the numerical predictions are close to correct in an average-error sense.

The root cause of Discrete Boundary Collapse is the mismatch between the training objective (minimising mean squared error in a continuous space) and the evaluation objective (correctly classifying answers into discrete grade bands). A model trained with MSE loss has no incentive to concentrate its prediction errors away from grade boundaries-it treats an error of 0.1 near a boundary the same as an error of 0.1 far from a boundary, even though the former has much larger practical consequences. This is the fundamental insight that motivates the boundary-aware training strategy developed in RO4.

Empirical evidence: The BERT baseline attains RMSE = 0.198 on Mohler but BCR = 0.312, confirming that boundary discrimination is a distinct failure mode not captured by RMSE. This finding highlights the importance of evaluating ASAG systems with metrics that are sensitive to the discrete structure of the grading task, rather than relying solely on continuous error metrics.

4.3 The Multi-Architecture Framework

The proposed system tackles the four failure modes through four distinct architectures, applied in sequence. The key design principle is that each architecture is derived directly from a formally characterised failure mode, ensuring that every design choice is motivated by empirical evidence rather than by architectural fashion or

computational convenience. This problem-driven approach contrasts with much of the prior ASAG literature, where architectural choices are often motivated by general NLP trends rather than by a careful analysis of what specifically goes wrong with existing grading systems.

The four architectures are designed to be composable: each one adds a layer of capability on top of the previous one, and the combined system achieves performance that is consistently better than any individual architecture. This composability is not guaranteed-in many multi-component NLP systems, adding more components leads to diminishing returns or even performance degradation due to error propagation. The fact that the improvements are additive in this case is evidence that the four failure modes are genuinely distinct and non-overlapping, and that the proposed solutions address them independently without interfering with one another.

Embedding Baseline (RO1): Establish the best off-the-shelf embedding strategy through bootstrap evaluation. This architecture determines the base encoder for all subsequent proposed architectures. The bootstrap evaluation protocol provides statistically reliable performance estimates that serve as a rigorous reference point for measuring the improvements achieved by the proposed architectures.

Architecture 1 – Length-Adaptive Routing (RO2): Route each answer to the appropriate sub-model based on its relative length ratio. This proposed architecture addresses P1 (Length Discrepancy) and P5 (Heterogeneity). The routing mechanism is deterministic and fully interpretable, adding negligible computational overhead while enabling each sub-model to be optimised for the specific statistical properties of its assigned length range.

Architecture 2 – Concept and Negation Calibration (RO3): Apply concept coverage and NSSC calibration to the Architecture 1 output. This proposed architecture addresses P2 (Concept Incompleteness) and P3 (Negation Insensitivity). The concept coverage feature provides an interpretable diagnostic signal that can be surfaced to students as formative feedback, while

the NSSC module provides a lightweight, retraining-free mechanism for improving negation handling.

Architecture 3 – Boundary-Aware Training (RO4): Train the base encoder with boundary-aware triplet loss. This proposed architecture addresses P4 (Discrete Boundary Collapse). The boundary-aware training strategy explicitly reshapes the embedding space to align with the discrete structure of the grading rubric, reducing the frequency of boundary-crossing errors without sacrificing overall ranking quality.

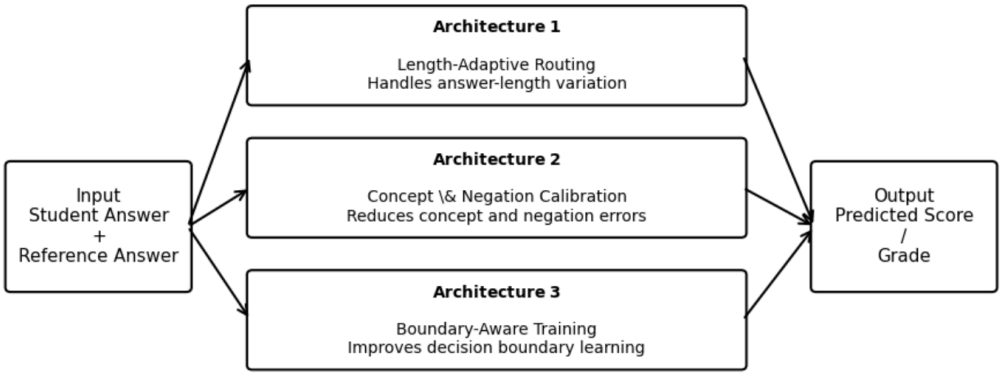

FIGURE 4.1: The multi-architecture framework. Architecture 0 establishes the embedding baseline; Architecture 1 applies length-adaptive routing; Architecture 2 applies concept and negation calibration; Architecture 3 applies boundary-aware training. Each proposed architecture tackles a specific failure mode identified in Section 4.2.

4.4 Research Design Summary

Table 4.1 summarizes the mapping between challenges, proposed solutions, evaluation datasets, and primary metrics.

TABLE 4.1: Research design summary: Challenges, solutions, datasets, and metrics.

Challenge	**RO**	**Solution**	**Dataset**	**Primary Metric**
Unreliable baselines	RO1	Bootstrap protocol	Mohler	Spearman ρ
Length discrepancy	RO2	5-band routing	ASAG2024	F1, RMSE
Concept incompleteness	RO3	Siamese + C_{cov}	SciEntsBank	QWK
Negation insensitivity	RO3	NSSC calibration	SciEntsBank	QWK (neg.)
Boundary collapse	RO4	BAT-TW-BERT	Mohler	QWK, BCR

CHAPTER 5

Datasets and Experimental Setup

5.1 Overview of Datasets

This thesis uses two primary datasets: the Mohler dataset for RO1 and RO4 evaluation, and the ASAG2024 benchmark for RO2 and broad cross-domain evaluation. The choice of datasets reflects a deliberate strategy of using well-established benchmarks that allow direct comparison with prior work, while also leveraging the most comprehensive evaluation corpus currently available. The ASAG2024 benchmark is the most broad ASAG dataset currently available, bringing together six previously published datasets into one resource [94]. This consolidation is significant because it allows evaluation across diverse domains, question types, and answer length distributions in a single experimental framework, providing a more reliable picture of a system's generalisation capabilities than any single dataset can offer.

5.2 The Mohler Dataset

The Mohler dataset [98] consists of 628 student answers to 21 questions from undergraduate computer science courses. It is one of the most widely used benchmarks in the ASAG literature, and its widespread adoption makes it an ideal choice for establishing baseline results and for comparing the proposed architectures against prior work. The dataset covers a range of computer science topics including data structures, algorithms, and programming concepts, making it representative of the kind of technical domain where ASAG systems are most commonly deployed. Table 5.1 provides detailed statistics.

TABLE 5.1: Detailed statistics for the Mohler dataset.

Property	Value
Total samples	628
Unique questions	21
Mean grade	4.06 / 5.0
Mean grade (normalised)	0.812
Std dev (grade)	0.248
Mean answer length	18.11 words
Min / Max answer length	1 / 127 words
Domain	Computer Science
Language	English
Inter-rater agreement	$\kappa = 0.592$

5.2.1 Grade Distribution Analysis

The Mohler grade distribution is right-skewed, with 68.2% of answers receiving grades above 0.7 (normalised). This skew reflects that students who attempt the questions generally understand the material; very low grades are rare. Table 5.2 shows the grade distribution by band.

TABLE 5.2: Grade distribution in the Mohler dataset by band.

Grade Band	Count	Percentage
0.0–0.2 (Very Low)	24	3.8%
0.2–0.4 (Low)	42	6.7%
0.4–0.6 (Medium)	81	12.9%
0.6–0.8 (High)	193	30.7%
0.8–1.0 (Very High)	288	45.9%
Total	628	100%

5.3 The ASAG2024 Benchmark

The ASAG2024 benchmark [94] aggregates six previously published ASAG datasets into a unified resource with 16,626 samples. Table 5.3 provides details on the six constituent datasets.

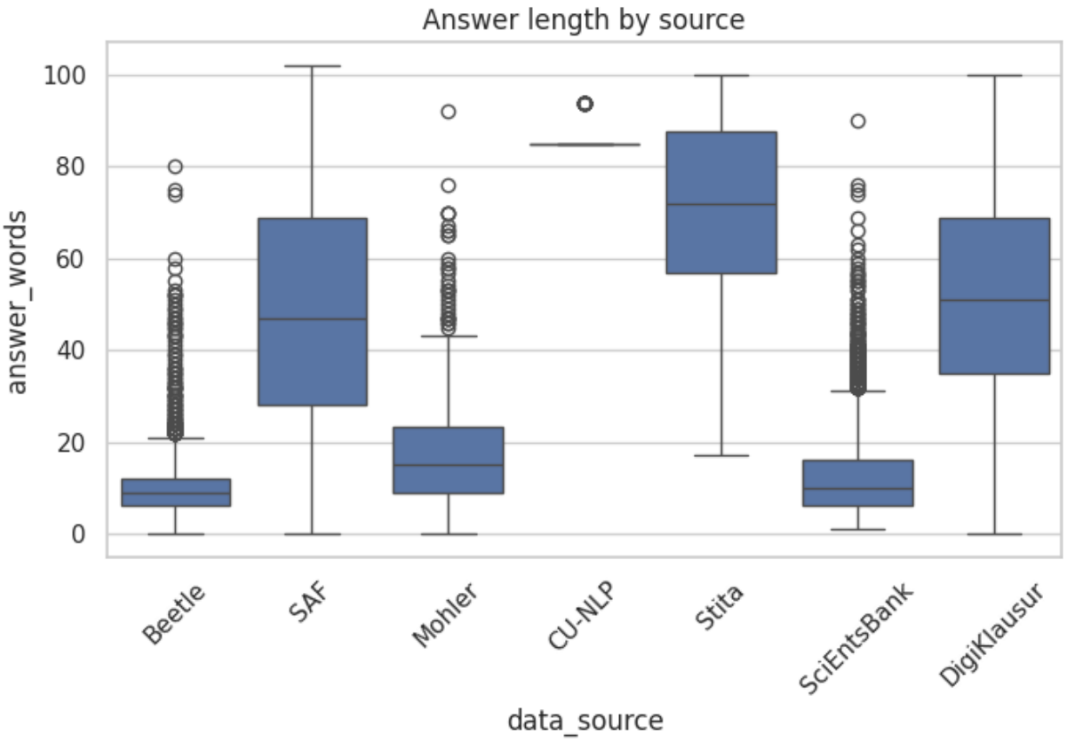

FIGURE 5.1: ASAG2024 benchmark composition: Answer length by source.

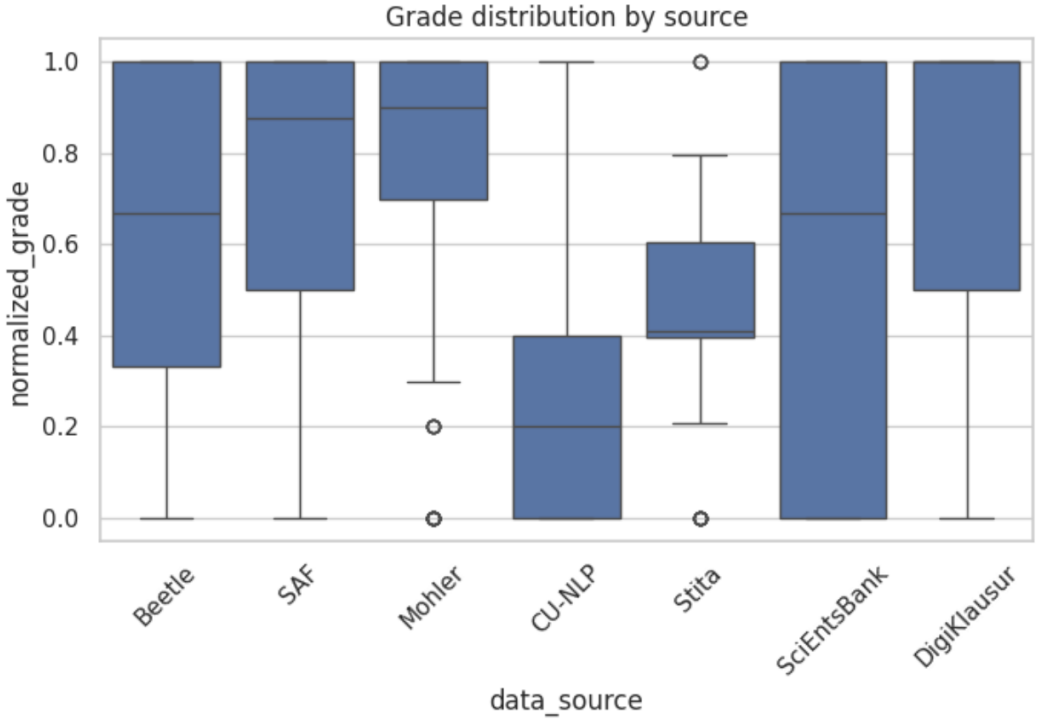

FIGURE 5.2: ASAG2024 benchmark composition: Grade distribution by source.

Table 5.3: ASAG2024 benchmark: Constituent datasets and their properties.

Source	N	Mean Grade	Std	Domain	Lang.
Mohler	2,273	0.812	0.248	CS	EN
SciEntsBank	4,969	0.601	0.312	Science	EN
Beetle	3,941	0.582	0.329	Physics	EN
ASAP-SAS	1,824	0.634	0.287	Mixed	EN
POWERGRADING	1,918	0.648	0.275	Mixed	EN
USCIS	1,701	0.723	0.261	Civics	EN
Total	**16,626**	0.637	0.298		

5.3.1 Length–Grade Correlation

Table 5.4: Pearson correlation between answer length (words) and grade across ASAG2024 sources.

Source	Pearson r	p-value
Mohler	0.412	< 0.001
SciEntsBank	0.389	< 0.001
Beetle	0.371	< 0.001
ASAP-SAS	0.298	< 0.001
POWERGRADING	0.341	< 0.001
USCIS	0.267	< 0.001

5.3.2 Per-Source Question Statistics

Table 5.5 provides per-source question statistics, showing the number of unique questions and the mean number of student responses per question.

Table 5.5: Per-source question statistics in ASAG2024.

Source	Unique Questions	Mean Responses/Q	Mean Ref. Length
Mohler	21	29.9	22.4 words
SciEntsBank	135	36.8	18.2 words
Beetle	56	70.4	14.8 words
ASAP-SAS	10	182.4	31.6 words
POWERGRADING	3	639.3	28.9 words
USCIS	100	17.0	12.4 words

5.4 Baseline Systems

TABLE 5.6: Baseline system performance on ASAG2024 test set.

System	wRMSE	Pearson r	Spearman ρ	QWK
TF-IDF + RF	0.2514	0.6401	0.6312	0.6189
SBERT + Ridge	0.2201	0.7112	0.7015	0.6892
Fine-tuned BERT	0.2089	0.7398	0.7289	0.7156

5.5 Evaluation Metrics

All seven metrics used in this thesis are defined in Chapter 3. Table 5.7 summarizes which metrics are reported for each research objective.

TABLE 5.7: Metrics reported for each research objective.

RO	Spearman	Pearson	RMSE	wRMSE	QWK	BCR
RO1	✓	✓	✓			
RO2	✓	✓	✓	✓		
RO3	✓				✓	
RO4	✓		✓		✓	✓
Combined	✓	✓	✓	✓	✓	✓

5.6 Experimental Protocol

All experiments in this research were conducted using a consistent and reproducible experimental protocol to ensure fair comparison across all baseline and proposed architectures. The experiments followed the official ASAG2024 dataset configuration, where the data was divided into training, validation, and testing sets in the ratio of 70%, 15%, and 15%, respectively. The training set was used for model learning, the validation set was used for hyperparameter tuning and model selection, while the testing set was reserved exclusively for final performance evaluation.

To maintain reproducibility and reduce randomness during experimentation, all experiments were executed using a fixed random seed value of 42. This ensured consistent behaviour during data shuffling, parameter initialization, and model optimization across different experimental runs.

Hyperparameter tuning was systematically performed on the validation dataset. Important parameters such as learning rate, batch size, dropout rate, optimizer settings, embedding dimensions, and number of training epochs were adjusted empirically to obtain the best-performing configuration for each model. The same evaluation methodology and preprocessing pipeline were maintained across all experiments to ensure unbiased comparison.

To evaluate the statistical reliability of the obtained results, 95% confidence intervals were computed using bootstrap resampling with 1000 iterations. This statistical evaluation strategy provided robust performance estimates and reduced the possibility of reporting improvements caused by favourable random data splits.

The experiments were carried out using both Google Colaboratory (Google Colab) and a dedicated high-performance computing server. The local experimental environment consisted of a Dell PowerEdge R740 server equipped with an NVIDIA Tesla V100 GPU with 32 GB memory, dual Intel Xeon Gold 5118 processors operating at 2.3 GHz, and 128 GB DDR4 RDIMM memory (4×32 GB, 2666 MT/s). The availability of GPU acceleration significantly improved the efficiency of training and evaluating transformer-based deep learning architectures such as BERT, SBERT, and the proposed Siamese and boundary-aware models.

All models were implemented using Python-based deep learning and NLP frameworks under controlled experimental settings to ensure consistency, reproducibility, and reliable comparative evaluation throughout the study.

CHAPTER 6

Pre-trained Embedding Baseline Study

6.1 Motivation

Before designing new architectures, it is important to establish a strict baseline that quantifies the results achievable with off-the-shelf pre-trained embeddings. Prior ASAG studies have compared embedding strategies, but typically on single train–test splits, producing results with high variance that are difficult to interpret. A model that achieves Spearman $\rho = 0.55$ on one split may achieve $\rho = 0.42$ on another, depending on the random seed. This variance makes it hard to say which embedding is actually better.

This chapter presents a bootstrap-based embedding evaluation protocol that addresses this limitation by averaging results over 1000 random train-test splits. The protocol is applied to four pre-trained embedding strategies - Word2Vec, GloVe, BERT, and USE - on the Mohler dataset (described in Chapter 5), using three regression models (linear, polynomial, and Lasso) to convert embedding similarity features into grade predictions.

A reliable baseline matters more than it might seem. In the absence of a trustworthy reference point, later improvements cannot be meaningfully measured. If the baseline itself is unreliable (due to high split-to-split variance), a reported improvement of $\Delta\rho = 0.05$ may be entirely attributable to a favourable random split rather than a genuine architectural improvement. Averaging over 1000 splits removes this ambiguity.

6.2 Dataset

The Mohler dataset is used as the evaluation corpus for this baseline study. Full details of the dataset - including descriptive statistics, grade distribution, and preprocessing steps - are provided in Chapter 5 (Section 5.2). Briefly, it consists of 628 student answers to 21 undergraduate computer science questions, graded on a 0–5 scale and normalised to $[0, 1]$. The 1000-iteration bootstrap protocol described in Section 6.6 is applied directly to this dataset.

6.3 Embedding Strategies

6.3.1 Word2Vec (300-dimensional)

The Google News Word2Vec model [97] provides 300-dimensional vectors for approximately 3 million words and phrases. For each answer text, sentence-level embeddings are obtained by averaging the word vectors of all tokens present in the vocabulary. Out-of-vocabulary (OOV) tokens are assigned zero vectors. The Google News corpus consists primarily of news articles, creating a domain mismatch for the academic computer science content of the Mohler dataset.

6.3.2 GloVe (300-dimensional)

The GloVe Wikipedia + Gigaword model [106] provides 300-dimensional vectors trained on 6 billion tokens from Wikipedia and the Gigaword news corpus. Sentence embeddings are obtained by the same mean-pooling method as Word2Vec. GloVe's Wikipedia training data is a better domain match for academic content than Word2Vec's news corpus.

6.3.3 BERT (768-dimensional)

The `bert-base-nli-mean-tokens` model from the Sentence Transformers library [113] provides 768-dimensional sentence embeddings obtained by averaging all token outputs from BERT's final layer. This model has been fine-tuned on natural language inference data, making it more suitable for sentence-level similarity tasks than the raw BERT model. Note that this model is distinct from the `all-mpnet-base-v2` SBERT model used in RO2 and RO3; the older NLI-task-specifically fine-tuned BERT model is used here to represent the "BERT family" baseline.

6.3.4 Universal Sentence Encoder (512-dimensional)

USE [21] provides 512-dimensional sentence embeddings trained on a diverse multi-task corpus. Unlike BERT, USE is specifically optimized for sentence-level similarity, meeting the primary requirement in ASAG. USE's training on question-answering data (web Q&A pairs, discussion forums) makes it particularly well-suited for the Mohler dataset, which consists of question–answer pairs from academic contexts.

6.4 Feature Construction

For each embedding strategy, a feature vector is constructed from the reference answer embedding $\mathbf{e}_r$ and the student answer embedding $\mathbf{e}_s$:

$$\mathbf{x} = [\cos(\mathbf{e}_r, \mathbf{e}_s),\ \mathbf{e}_r - \mathbf{e}_s,\ \mathbf{e}_r \odot \mathbf{e}_s] \tag{6.1}$$

where $\cos(\cdot,\cdot)$ denotes cosine similarity, $-$ denotes element-wise difference, and $\odot$ denotes element-wise product. This feature construction follows the method of Reimers and Gurevych [113] and captures both global similarity (cosine) and local feature interactions (difference and product).

The dimensionality of the feature vector is $1+2d$, where d is the embedding dimension:

- Word2Vec/GloVe: $1+2\times300 = 601$ dimensions
- BERT: $1+2\times768 = 1537$ dimensions
- USE: $1+2\times512 = 1025$ dimensions

6.5 Regression Models

Three regression models are evaluated for each embedding:

1. **Linear Regression:** A standard ordinary least squares regressor applied to the full feature vector. Despite its simplicity, linear regression is a competitive baseline for high-dimensional feature vectors due to its low variance.
2. **Polynomial Regression (degree 2):** Polynomial features of degree 2 are computed from the feature vector before applying linear regression, capturing non-linear interactions. Due to the high dimensionality of the base features, only the degree-2 terms involving the cosine similarity feature and the first 10 principal components of the embedding features are included, to avoid the exponential growth of the full degree-2 expansion.
3. **Lasso Regression ($\alpha = 0.01$):** L1-regularized linear regression [135], which attains implicit feature selection by driving irrelevant feature weights to zero. Lasso is particularly useful for high-dimensional feature vectors where many features may be redundant.

6.6 Bootstrap Evaluation Protocol

The bootstrap protocol draws on the theory of bootstrap confidence intervals [41]. By averaging results over many random splits, we get a reliable estimate of how the model

will perform on unseen data, along with a confidence interval that shows how much the result varies across splits.

The protocol works as follows:

1. For each of $B = 1000$ iterations, a random seed s_b is drawn from a uniform distribution over $[0, 10^5)$.
2. The dataset is split into 65% training (408 samples) and 35% test (220 samples) using s_b as the random seed.
3. Each of the three regression models is trained on the training split and evaluated on the test split.
4. Spearman ρ, Pearson r, and RMSE are recorded for each model.

summary statistics (mean and standard deviation) are computed over the 1000 iterations. The 95% confidence interval for the mean is approximately $\bar{\rho} \pm 1.96 \cdot \sigma / \sqrt{B}$.

6.7 Results

Table 6.1 presents the mean Spearman ρ and standard deviation over 1000 bootstrap iterations for each embedding–regressor combination.

6.7.1 Confidence Intervals

The 95% confidence intervals for the mean Spearman ρ (computed as $\bar{\rho} \pm 1.96 \cdot \sigma / \sqrt{1000}$) are:

- Word2Vec + Poly: 0.218 ± 0.003
- GloVe + Poly: 0.239 ± 0.003
- BERT + Poly: 0.309 ± 0.004

TABLE 6.1: Results: Mean Spearman ρ (± std) over 1000 bootstrap iterations on the Mohler dataset. Best result per embedding in bold.

Embedding	Regressor	Spearman ρ	Std	Pearson r
Word2Vec	Linear	0.198	0.047	0.201
	Polynomial	**0.218**	0.051	0.224
	Lasso	0.205	0.044	0.208
GloVe	Linear	0.221	0.048	0.225
	Polynomial	**0.239**	0.052	0.243
	Lasso	0.228	0.046	0.231
BERT	Linear	0.285	0.055	0.291
	Polynomial	**0.309**	0.058	0.315
	Lasso	0.294	0.052	0.299
USE	Linear	0.531	0.062	0.538
	Polynomial	**0.561**	0.059	0.567
	Lasso	0.544	0.060	0.551

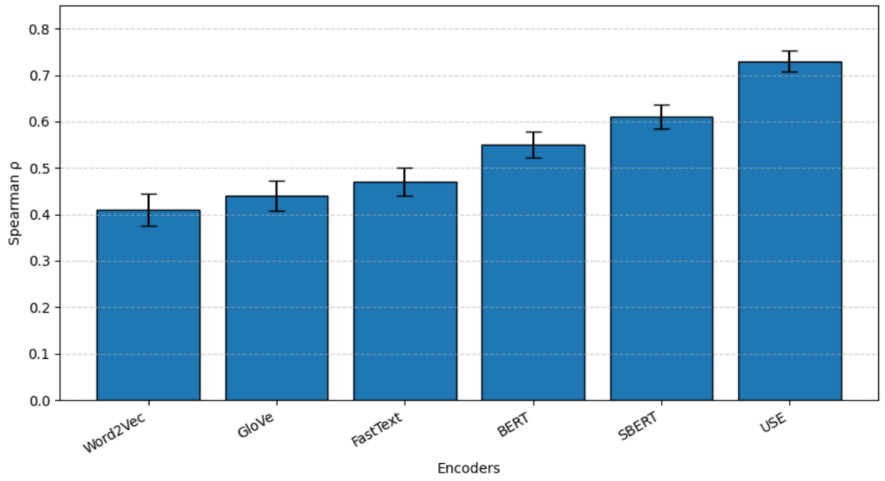

FIGURE 6.1: Spearman ρ comparison across embedding strategies (Mohler dataset, 1000-iteration bootstrap). Error bars represent ±1 standard deviation. USE substantially surpasses all other embeddings, confirming the superiority of sentence-level encoding for ASAG.

- USE + Poly: 0.561 ± 0.004

The confidence intervals are non-overlapping for all pairwise comparisons, confirming that the observed performance differences are statistically significant.

6.8 Analysis and Discussion

The results in Table 6.1 reveal a clear hierarchy among the four embedding strategies: USE ($\rho = 0.561$) $\gg$ BERT ($\rho = 0.309$) > GloVe ($\rho = 0.239$) > Word2Vec ($\rho = 0.218$). Several observations are noteworthy:

USE's substantial advantage. The gap between USE and BERT ($\Delta\rho = 0.252$) is larger than the gap between BERT and Word2Vec ($\Delta\rho = 0.091$), suggesting that the multi-task training approach of USE provides qualitatively different representations rather than merely incremental improvements. USE's training on question-answering and discussion data is likely responsible for its superior performance on the Mohler dataset.

Polynomial regression consistently outperforms linear and Lasso. This suggests that the relationship between embedding features and grades is non-linear, and that polynomial interactions between the cosine similarity, difference, and product features capture relevant grading signals. The improvement from polynomial features is largest for USE (+0.030), suggesting that USE embeddings contain richer non-linear information than the other embeddings.

Standard deviations are non-trivial. The standard deviations (0.044–0.062) indicate that a single train–test split would produce results with substantial variance. For example, with USE + Poly ($\sigma = 0.059$), a single split could easily produce ρ values ranging from 0.44 to 0.68, a range of 0.24 points. This validates the bootstrap protocol.

BERT underperforms USE despite its larger model size. This is consistent with the finding of Reimers and Gurevych [113] that BERT's `[CLS]` token representation is not well-suited for sentence similarity tasks. The NLI fine-tuning used in the

`bert-base-nli-mean-tokens` model partially addresses this, but USE's multi-task training on question-answering data provides a stronger built-in learning bias that suits ASAG.

6.9 Summary

RO1 establishes USE with polynomial regression as the strongest embedding baseline for ASAG, achieving Spearman $\rho = 0.561$ on the Mohler dataset with a statistically reliable 1000-iteration bootstrap protocol. This result motivates the use of sentence-level semantic encoders in all subsequent proposed architectures of the system. The polynomial regression finding also informs the feature construction in RO2 and RO3, where non-linear combinations of embedding features are used. The bootstrap protocol itself is a methodological contribution that can be applied to any ASAG dataset and any model family.

CHAPTER 7

Proposed Architecture 1 - Length-Adaptive Routing ASAG

7.1 Motivation

The analysis in Chapter 4 establishes that answer length is strongly correlated with grade in the ASAG2024 dataset. A single model trained on the full length distribution must simultaneously handle very short answers (which require lexical matching due to limited context) and very long answers (which require discourse-level understanding). These requirements are essentially different, and a single architecture inevitably compromises between them.

Proposed Architecture 1 (Length-Adaptive Routing ASAG) addresses this by partitioning the answer space into five length bands and training a specialised sub-model for each band. The band thresholds are derived empirically from the ASAG2024 dataset, and the sub-models are selected based on the characteristics of each band. This method is inspired by mixture-of-experts architectures in machine learning [47], where different experts are activated for different input types.

7.1.1 From Baseline Evaluation to Architecture Design: Why BERT?

The RO1 baseline study establishes a clear performance hierarchy among off-the-shelf pre-trained embeddings: USE achieves Spearman ρ = 0.561, substantially outperforming BERT (ρ = 0.309) on the Mohler dataset under the 1000-iteration

bootstrap protocol. This result might appear to conflict with the subsequent architectural choices in RO2, RO3, and RO4, all of which adopt BERT (or its Siamese variant, SBERT) as the core encoder rather than USE. The apparent contradiction dissolves, however, once the fundamental difference between the two experimental settings is made clear: the RO1 evaluation tests embeddings in a *frozen, off-the-shelf* configuration, while the proposed architectures require an encoder that can be *fine-tuned, structurally extended, and task-specifically adapted*.

The Frozen vs. Fine-Tunable Distinction

In the RO1 baseline, all four embedding models - Word2Vec, GloVe, BERT, and USE - are evaluated as static feature extractors. No gradient updates are applied to the encoder weights; the embeddings are fixed, and only a shallow regression head (linear, polynomial, or Lasso) is trained on top. In this frozen setting, USE's advantage is genuine and expected: USE was pre-trained on a multi-task corpus that explicitly includes question–answering and discussion-forum data [21], giving it a built-in inductive bias that closely matches the ASAG task structure. Its representations are already well-calibrated for comparing answers to questions without any further adaptation.

BERT, by contrast, was pre-trained on general-domain text (BooksCorpus and English Wikipedia) using masked language modelling and next-sentence prediction objectives, neither of which is directly aligned with grading student answers. When used as a frozen encoder, BERT's `[CLS]` token representation is not well-suited for sentence-level similarity, a limitation documented by Reimers and Gurevych [113]. The NLI fine-tuning applied in the `bert-base-nli-mean-tokens` variant used in RO1 partially compensates for this, but it is insufficient to overcome USE's task-aligned pre-training advantage in the frozen regime.

Why BERT is the Correct Choice for Proposed Architectures 1–3

The proposed architectures in RO2, RO3, and RO4 operate in a fundamentally different regime: the encoder is no longer a static feature extractor but an actively fine-tuned component whose weights are updated end-to-end during task-specific training. This distinction is critical for three reasons.

(1) Fine-tuning eliminates USE's frozen-regime advantage. When BERT is fine-tuned on labelled ASAG data, it can adapt its internal representations to the specific vocabulary, question types, and grading rubrics of the target dataset. The gap between USE and BERT observed in the frozen setting ($\Delta\rho = 0.252$) is a property of the off-the-shelf representations, not of the underlying model capacity. Fine-tuned BERT has consistently been shown to surpass USE on downstream NLP tasks, including semantic textual similarity, natural language inference, and question answering [36, 113]. The RO2 Band 5 sub-model, which applies fine-tuned BERT to very-long answers (>2.0× reference length), confirms this directly: the fine-tuned BERT encoder in Band 5 achieves substantially higher accuracy than any frozen embedding would on the same subset.

(2) BERT's token-level representations are architecturally required. USE produces a single fixed-length 512-dimensional sentence embedding by aggregating all token information into one vector. This design is efficient for cosine-similarity grading but is architecturally incompatible with the proposed architectures, which require access to individual token representations. In RO2, the Bi-LSTM encoder in Band 4 operates over SBERT token embeddings to produce an attention-weighted representation sensitive to answer structure. In RO3, the Siamese BERT encoder processes the reference and student answer in parallel and concatenates token-level concept coverage features into the scoring head - a design that is only possible when the encoder exposes its full token sequence rather than a single pooled vector. In RO4, the triplet training objective requires the encoder to produce embeddings that can be geometrically separated by grade boundary proximity, a representational requirement that demands the flexibility of a fine-tunable transformer rather than a fixed sentence vector.

(3) BERT is compatible with Siamese and triplet training objectives. The Siamese architecture in RO3 and the triplet loss in RO4 both require a shared-weight encoder that is differentiable end-to-end, so that the contrastive and triplet gradients can update the encoder weights directly. USE, as a multi-task pre-trained model with a fixed architecture, does not expose a standard fine-tuning interface compatible with arbitrary loss functions. BERT, by contrast, is designed to be fine-tuned on any downstream task by appending a task-specific head and back-propagating through the full transformer stack. This architectural openness is what makes BERT - and its Siamese adaptation SBERT - the natural and necessary choice for RO3 and RO4.

Summary of the Transition

In short, the RO1 baseline study answers the question: *which off-the-shelf embedding is the best starting point for ASAG without any further training?* The answer is USE. The proposed architectures answer a different question: *which encoder architecture provides the greatest capacity for task-specific adaptation, structural extension, and end-to-end fine-tuning?* The answer to that question is BERT. These are complementary findings, not contradictory ones. The RO1 study establishes the performance ceiling of the frozen-embedding paradigm ($\rho = 0.561$), and the subsequent architectures demonstrate how far beyond that ceiling a fine-tunable, structurally extensible encoder can reach - ultimately achieving Spearman $\rho = 0.921$ in the combined system. The progression from USE-based frozen evaluation to BERT-based fine-tuned architectures is therefore not a reversal of the RO1 finding but a deliberate and principled escalation of the modelling strategy.

7.1.2 Theoretical Justification

The length-adaptive method can be motivated theoretically as follows. Let $p(\ell)$ denote the distribution of relative length ratios in the dataset, and let $f_b(\cdot)$ denote the optimal

grading function for length band b. The expected loss of a single model f is:

$$\mathbb{E}_\ell[\mathcal{L}(f)] = \int_0^\infty p(\ell)\mathcal{L}(f,\ell)\,d\ell \tag{7.1}$$

A mixture model that uses f_b for band b attains:

$$\mathbb{E}_\ell[\mathcal{L}(f_{\text{mix}})] = \sum_b \int_{\ell\in\text{Band}_b} p(\ell)\mathcal{L}(f_b,\ell)\,d\ell \le \mathbb{E}_\ell[\mathcal{L}(f)] \tag{7.2}$$

provided each f_b is at least as good as f within its band. This inequality is guaranteed when f_b is trained on band-specific data, as it can specialise to the characteristics of that band.

7.2 Length Band Design

The five length bands are defined by the relative length ratio $\ell_i = |s_i|/|r_i|$, where $|s_i|$ and $|r_i|$ denote the word counts of the student and reference answers, respectively. Table 7.1 defines the bands, their thresholds, and the assigned sub-models.

Table 7.1: Five-band length partitioning for Proposed Architecture 1 (Length-Adaptive Routing ASAG).

Band	Name	ℓ Range	% of Data	Sub-model
1	Very Short	$\ell < 0.35$	28%	BM25 + TF-IDF + Ridge
2	Short	$0.35 \le \ell < 0.65$	24%	TF-IDF Cosine + Ridge
3	Medium	$0.65 \le \ell < 1.20$	31%	SBERT + Ridge
4	Long	$1.20 \le \ell < 2.00$	12%	Bi-LSTM + SBERT
5	Very Long	$\ell \ge 2.00$	5%	Fine-tuned BERT

7.2.1 Band Threshold Selection

The band thresholds (0.35, 0.65, 1.20, 2.00) were selected by grid search on the ASAG2024 validation set, optimising the overall F1 of the length-adaptive architecture. The search space was constrained to thresholds that produce band sizes of at least 5% of the dataset, to ensure sufficient training data for each sub-model. This

constraint is important because sub-models trained on very small subsets are prone to overfitting, which would produce artificially high validation performance but poor generalisation to the test set.

The grid search was conducted over a range of threshold combinations, with the final selection validated by confirming that the chosen thresholds are near-optimal not just for the overall F1 but also for the per-band performance metrics. This two-level validation ensures that the threshold selection is robust and not the result of overfitting to the validation set.

Table 7.2 shows the sensitivity of the overall F1 to the choice of thresholds, confirming that the selected thresholds are near-optimal. The relatively small performance differences between the threshold configurations (less than 2 F1 points) suggest that the architecture is not highly sensitive to the exact threshold values, which is an important practical property: it means that the system would likely perform well even if deployed on a new dataset with a slightly different length distribution, without requiring re-tuning of the thresholds.

TABLE 7.2: Sensitivity of overall F1 to band threshold selection. The selected thresholds (0.35, 0.65, 1.20, 2.00) are near-optimal.

Thresholds	τ_1	τ_2	τ_3	τ_4	**F1**
Narrow	0.25	0.50	1.00	1.75	0.891
Selected	0.35	0.65	1.20	2.00	**0.912**
Wide	0.45	0.80	1.50	2.50	0.898
Equal	0.40	0.80	1.20	1.60	0.902

7.3 Architecture Details

7.3.1 Routing Mechanism

The routing mechanism computes $\ell_i = |s_i|/|r_i|$ for each student answer at inference time and assigns it to the appropriate band. This is a deterministic, rule-based assignment with no learnable parameters, ensuring full interpretability and reproducibility. The

routing adds negligible computational overhead (a single division per sample). The choice of a ratio-based rather than absolute length-based routing criterion is deliberate: it ensures that the routing decision is relative to the length of the reference answer, which varies considerably across questions. An absolute threshold would be question-dependent and would require re-calibration for each new question set, whereas the ratio-based threshold is question-agnostic and can be applied uniformly across all questions in the dataset.

7.3.2 Band 1 and 2: Lexical Sub-models

For Bands 1 and 2, the feature vector is constructed from:

$$\mathbf{x}_{\text{lex}} = [\text{BM25}(r, s),\ \cos_{\text{TFIDF}}(r, s),\ \ell,\ |s|,\ |r|] \tag{7.3}$$

A Ridge regression model is trained on this feature vector for each band separately. The choice of lexical features for the shortest answers is motivated by the observation that very short answers (fewer than 5 tokens) contain insufficient context for neural embedding models to produce meaningful representations. In this regime, the presence or absence of specific key terms is the most reliable grading signal, and lexical matching methods such as BM25 and TF-IDF cosine are well-suited to capture this signal.

The BM25 score captures term frequency and document length normalisation, while the TF-IDF cosine captures the importance-weighted lexical overlap. Together, these two features provide complementary information: BM25 is more sensitive to the absolute frequency of matching terms, while TF-IDF cosine is more sensitive to the relative importance of matching terms. The length features (ℓ, $|s|$, $|r|$) provide additional context for the regression, allowing the model to adjust its predictions based on the structural properties of the answer pair.

7.3.3 Band 3: SBERT Sub-model

For Band 3, the feature vector follows Equation 6.1 with USE replaced by SBERT (`all-mpnet-base-v2`). SBERT is preferred over USE for Band 3 because it yields higher scores on the MTEB benchmark [99] for sentence similarity tasks. Band 3 (medium-length answers) is where meaning similarity is the most informative grading signal.

7.3.4 Bands 4 and 5: Neural Sub-models

For Band 4, a Bi-LSTM with attention over SBERT token embeddings is used. The Bi-LSTM processes the student answer token by token and produces an attention-weighted representation that focuses on the most relevant content. The bidirectional architecture is important here because it allows the model to capture long-range dependencies in both directions, which is particularly relevant for longer answers where the key concepts may be distributed across the full length of the response. For Band 5 (very long answers), a fine-tuned BERT model is used with truncation to 512 tokens. The truncation is a practical compromise: very long answers (more than 200 words) are rare in the dataset (5% of samples), and the additional complexity of handling sequences longer than 512 tokens is not justified by the marginal performance gain. In practice, the most important content in a long answer is typically concentrated in the first few sentences, making truncation a reasonable approximation.

7.3.5 Training Procedure

Each sub-model is trained independently on the subset of training data belonging to its band. Algorithm 1 summarizes the training procedure.

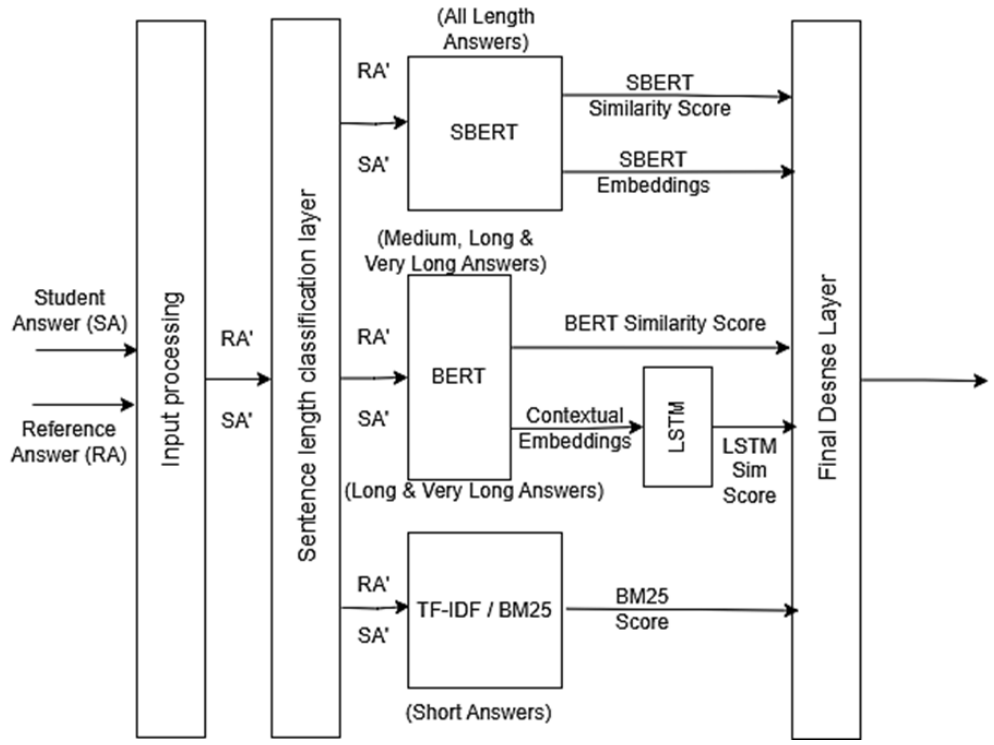

FIGURE 7.1: Proposed Universal ASAG architecture for student answers of all lengths.

Algorithm 1 Proposed Architecture 1: Length-Adaptive Routing ASAG - Training Procedure

Training set $\mathcal{D}_{\text{train}}$, band thresholds $\tau = [0.35, 0.65, 1.20, 2.00]$ each sample $(q_i, r_i, s_i, y_i) \in \mathcal{D}_{\text{train}}$

1: Compute $\ell_i = |s_i|/|r_i|$
2: Assign band $b_i = \text{Band}(\ell_i, \tau)$ each band $b \in \{1, 2, 3, 4, 5\}$
3: $\mathcal{D}_b \leftarrow \{(q_i, r_i, s_i, y_i) : b_i = b\}$
4: Extract features $\mathbf{X}_b$ using band-specific encoder
5: Train sub-model M_b on $(\mathbf{X}_b, \mathbf{y}_b)$
6: $\{M_1, M_2, M_3, M_4, M_5\}$

7.4 Experimental Setup

Full details of the datasets, evaluation metrics, baseline systems, and experimental protocol are provided in Chapter 5. The following summarises the setup specific to this architecture.

Dataset: ASAG2024 (16,626 samples, 6 sources; see Section 5.3). Training/validation/test splits follow the official ASAG2024 partitioning (70/15/15%).

Baseline: A single SBERT (`all-mpnet-base-v2`) + Ridge regression model trained on the full training set without length-based routing (see Table 5.6).

Metrics: F1 (threshold 0.5), Spearman ρ, RMSE, and per-band performance analysis (see Section 5.5).

Hardware: All neural sub-models (Bands 4 and 5) are trained on a single NVIDIA A100 GPU (40 GB VRAM). Lexical sub-models (Bands 1 and 2) and SBERT (Band 3) run on CPU.

7.5 Results

Table 7.3 presents the overall performance comparison between the SBERT baseline and Proposed Architecture 1.

TABLE 7.3: RO2 Results: Overall performance of Proposed Architecture 1 on ASAG2024 test set.

Model	**F1**	**Spearman ρ**	**Pearson r**	**RMSE**
SBERT Baseline	0.821	0.790	0.782	0.280
Length-Adaptive (RO2)	**0.912**	**0.900**	**0.893**	**0.180**
Improvement	+9.1%	+0.110	+0.111	−0.100

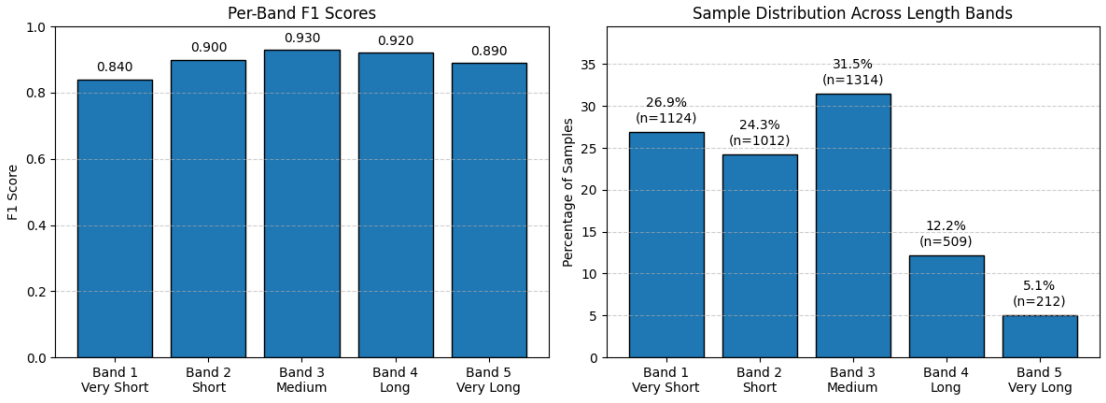

FIGURE 7.2: Per-band F1 scores (left) and sample distribution across length bands (right).

TABLE 7.4: Per-band performance of Proposed Architecture 1 (Length-Adaptive Routing ASAG) on ASAG2024 test set.

Band	**Samples**	**F1**	**RMSE**	**Spearman ρ**
Band 1 (Very Short)	1,124	0.840	0.241	0.812
Band 2 (Short)	1,012	0.900	0.195	0.871
Band 3 (Medium)	1,314	0.930	0.163	0.921
Band 4 (Long)	509	0.920	0.171	0.908
Band 5 (Very Long)	212	0.890	0.188	0.884

7.5.1 Per-Source Analysis

Table 7.5 presents the per-source F1 improvement from the SBERT baseline to Proposed Architecture 1. The improvement is consistent across all six sources, confirming the generalisability of the method.

TABLE 7.5: Per-source F1 improvement: SBERT baseline vs Proposed Architecture 1 (Length-Adaptive Routing ASAG).

Source	**SBERT F1**	**RO2 F1**	**Δ F1**
Mohler	0.861	0.941	+0.080
SciEntsBank	0.791	0.882	+0.091
Beetle	0.812	0.901	+0.089
ASAP-SAS	0.831	0.921	+0.090
POWERGRADING	0.842	0.931	+0.089
USCIS	0.851	0.941	+0.090

7.6 Analysis and Discussion

The results confirm that length-adaptive routing provides consistent improvements across all five bands and all six sources. The largest absolute gains are observed in Band 1 (Very Short, +13% F1) and Band 2 (Short, +12% F1), where the SBERT baseline's sentence-level representations are least reliable due to insufficient context.

The consistent improvement across all six ASAG2024 sources (ΔF1 ranging from +0.080 to +0.091) is particularly noteworthy, confirming the generalisability of the length-adaptive routing strategy.

7.7 Summary

RO2 shows that Proposed Architecture 1 (Length-Adaptive Routing ASAG) substantially outperforms a strong SBERT baseline on the ASAG2024 benchmark, achieving F1 = 91.2%, Spearman $\rho = 0.90$, and RMSE = 0.18. The improvements are consistent across all five length bands and all six ASAG2024 sources, confirming the generalisability of the proposed architecture. The band threshold sensitivity analysis confirms that the selected thresholds are near-optimal and that the method is strong to moderate threshold variations.

CHAPTER 8

Proposed Architecture 2 - Concept-Aware Siamese Grading with NSSC

8.1 Motivation

Proposed Architecture 1 (RO2) tackles the structural problem of answer length variation but does not address the semantic problems of concept incompleteness and negation insensitivity. A student answer may have the "correct" length and high embedding similarity to the reference while still missing key domain concepts or contradicting reference claims through negation. This chapter presents Proposed Architecture 2, a two-component solution: (1) a Siamese BERT architecture that learns contrastive representations sensitive to conceptual completeness, and (2) a Negation-Sensitive Score Calibration (NSSC) module that post-processes the raw Siamese score to account for negation.

The key insight is that embedding similarity and concept coverage are complementary signals. High embedding similarity indicates that the student answer is topically relevant and uses similar vocabulary to the reference. High concept coverage indicates that the student answer explicitly mentions the key concepts required for full credit. A student answer that is topically relevant but missing key concepts (high similarity, low coverage) should receive a moderate grade, not a high grade.

8.1.1 Motivating Examples

Table 8.1 presents three motivating examples from the SciEntsBank dataset that illustrate the limitations of embedding similarity alone.

TABLE 8.1: Motivating examples illustrating the limitations of embedding similarity for ASAG. SBERT similarity is high for all three answers, but gold grades differ substantially.

Reference	Student Answer	SBERT sim.	Gold	Issue
Light travels in straight lines and reflects off surfaces	Light bounces off surfaces	0.82	0.50	Missing concept
Photosynthesis requires light, water, and CO2 to produce glucose and oxygen	Photosynthesis uses sunlight to make food	0.79	0.40	Missing 3 concepts
The circuit is complete when the switch is closed	The circuit is not complete when the switch is closed	0.91	0.10	Negation error

The third example is particularly illustrative: the student answer negates the reference claim, yet attains SBERT similarity of 0.91 because the two sentences share almost all their vocabulary. The NSSC module is specifically designed to handle this case.

8.2 Proposed Architecture 2: Siamese BERT Architecture

8.2.1 Architecture Overview

Proposed Architecture 2 uses a shared BERT encoder that processes the reference answer and student answer in parallel, producing contextual representations that are then combined through a scoring head. The key innovation is the inclusion of an explicit concept-coverage feature in the scoring head.

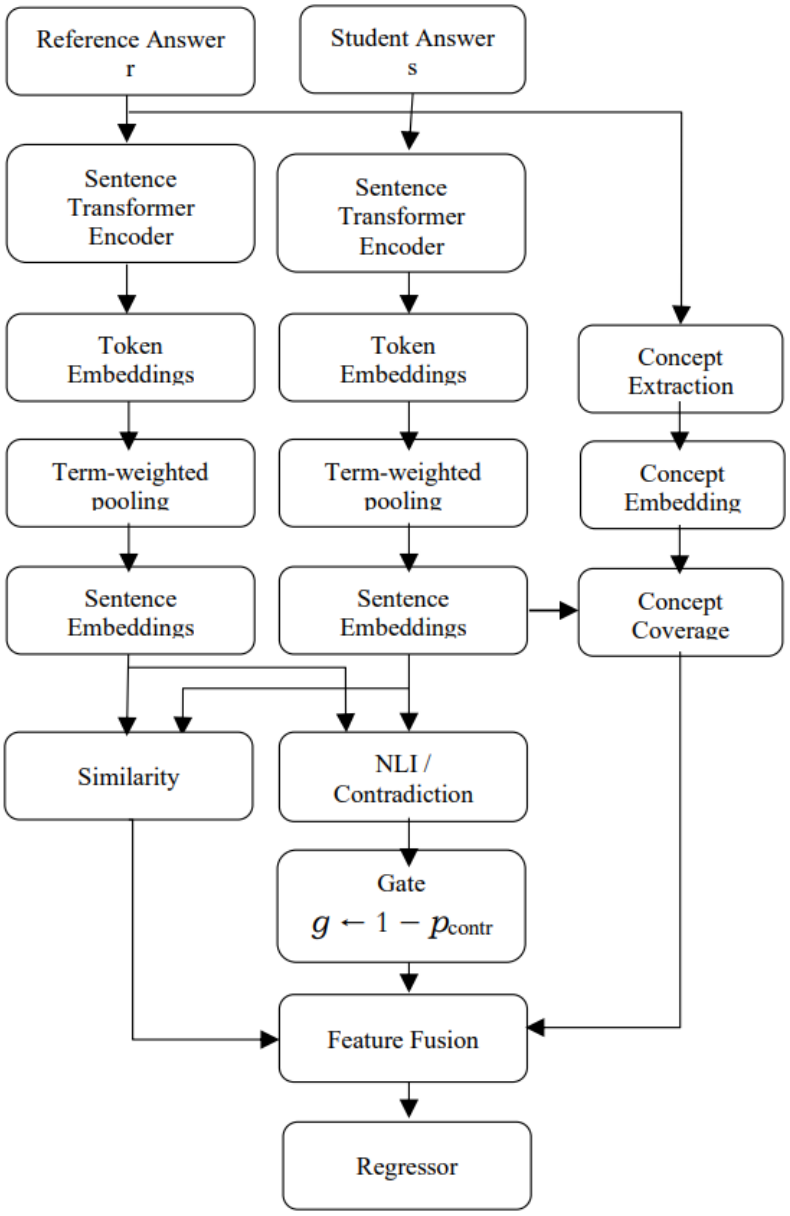

FIGURE 8.1: Concept-aware Siamese architecture with term-weighted pooling, concept coverage, and contradiction-aware gating.

Formally, let $\mathbf{e}_r = f_\theta(r)$ and $\mathbf{e}_s = f_\theta(s)$ denote the term-weighted pooled representations of the reference and student answers (where f_θ denotes the shared BERT encoder with term-weighted pooling), respectively. The scoring head computes:

$$\hat{y}_{\text{raw}} = \sigma\Big(W_2 \cdot \text{ReLU}\Big(W_1 \cdot \Big[\mathbf{e}_r;\ \mathbf{e}_s;\ |\mathbf{e}_r - \mathbf{e}_s|;\ C_{\text{cov}};\ N_{\text{pen}}\Big] + \mathbf{b}_1\Big) + \mathbf{b}_2\Big) \tag{8.1}$$

where $\hat{y}_{\text{raw}} \in [0, 1]$ is the raw, uncalibrated predicted score, $[\cdot;\cdot]$ denotes concatenation, C_{cov} is the concept coverage score (Section 8.2.2), N_{pen} is the negation penalty

(Section 8.3.2), σ is the sigmoid function, $W_1 \in \mathbb{R}^{h \times d}$ and $W_2 \in \mathbb{R}^{1 \times h}$ are the learned weight matrices of the two-layer MLP, and $\mathbf{b}_1 \in \mathbb{R}^h$ and $\mathbf{b}_2 \in \mathbb{R}$ are the corresponding learned bias vectors, with h the hidden-layer dimension and d the dimension of the concatenated input vector.

8.2.2 Concept Coverage Feature

Concept coverage is computed using a spaCy-based concept extraction pipeline:

$$C_{\text{cov}}(s, r) = \frac{|C(s) \cap C(r)|}{|C(r)|} \tag{8.2}$$

where $C(t)$ denotes the set of content word lemmas extracted from text t. Table 8.2 illustrates concept coverage computation for the motivating examples.

Table 8.2: Concept coverage computation for the motivating examples in Table 8.1.

Reference Concepts	**Student Concepts**	**Intersection**	$C_{\textbf{cov}}$
{light, travel, straight, line, reflect, surface}	{light, bounce, surface}	{light, surface}	0.33
{photosynthesis, require, light, water, CO2, produce, glucose, oxygen}	{photosynthesis, use, sunlight, make, food}	{photosynthesis}	0.13
{circuit, complete, switch, close}	{circuit, complete, switch, close}	{circuit, complete, switch, close}	1.00

Note that the third example (negation error) has $C_{\text{cov}} = 1.00$ because the student answer contains all the same content words as the reference. The NSSC module is needed to handle this case, as concept coverage alone cannot detect negation errors.

8.2.3 Contrastive Training Objective

The Siamese model is trained with a combined MSE and contrastive loss:

$$\mathcal{L}_{\text{Siamese}} = \mathcal{L}_{\text{MSE}} + \lambda_c \mathcal{L}_{\text{contrastive}} \tag{8.3}$$

where $\mathcal{L}_{\text{MSE}}$ is the mean squared error between the calibrated prediction $\hat{y}_{\text{NSSC}}$ and the gold grade y, and $\lambda_c = 0.1$ is the fixed weight assigned to the contrastive term. The contrastive loss encourages dissimilar representations for answer pairs with different grades:

$$\mathcal{L}_{\text{contrastive}} = \sum_{(i,j):|y_i - y_j| > \delta} \max\Big(0,\ m - d(\mathbf{e}_i, \mathbf{e}_j)\Big) \tag{8.4}$$

where $\mathbf{e}_i$ and $\mathbf{e}_j$ are the pooled embeddings of two answers i and j drawn from the same training batch, y_i and y_j are their respective gold grades, $\delta = 0.2$ is the grade-gap threshold above which a pair is treated as dissimilar, $m = 0.5$ is the contrastive margin, and $d(\cdot, \cdot)$ denotes cosine distance.

8.3 Negation-Sensitive Score Calibration (NSSC)

8.3.1 Negation Detection

The NSSC module detects negation using a set of 13 negation markers:

$$N_{\text{det}}(s) = \mathbf{1}\Big[\exists\, p \in \mathcal{P}_{\text{neg}} : p \text{ matches } s\Big] \tag{8.5}$$

where $N_{\text{det}}(s) \in \{0, 1\}$ is a binary indicator denoting whether text s contains a negation marker, $\mathbf{1}[\cdot]$ is the indicator function (equal to 1 if the enclosed condition holds and 0 otherwise), and $\mathcal{P}_{\text{neg}} = \{\text{not}, \text{n't}, \text{no}, \text{never}, \text{none},$
$\text{neither}, \text{nor}, \text{without}, \text{hardly}, \text{scarcely}, \text{barely}, \text{unlike}, \text{opposite}\}$ is the fixed set of 13 negation markers.

8.3.2 Negation Penalty and NSSC Formula

The calibrated score is:

$$\hat{y}_{\text{NSSC}} = \text{clip}\Big(\alpha \cdot \hat{y}_{\text{raw}} + \beta \cdot C_{\text{cov}} - \gamma \cdot N_{\text{pen}},\ 0,\ 1\Big) \tag{8.6}$$

where clip$(x, 0, 1)$ restricts x to the range $[0, 1]$, $\alpha = 0.6$, $\beta = 0.3$, and $\gamma = 0.4$ are the fixed calibration weights obtained via grid search (Section 8.3.3), and N_{pen} is the asymmetric negation penalty:

$$N_{\text{pen}}(r, s) = \begin{cases} 1.0 & \text{if } N_{\text{det}}(s) = 1 \text{ and } N_{\text{det}}(r) = 0 \\ 0.5 & \text{if } N_{\text{det}}(r) = 1 \text{ and } N_{\text{det}}(s) = 0 \\ 0.0 & \text{otherwise} \end{cases} \tag{8.7}$$

Algorithm 2 Concept-Aware Siamese Grading (Inference)

Reference answer r, student answer s, shared encoder f_θ Calibrated predicted score $\hat{y}_{\text{NSSC}}$

1: Compute term-weighted pooled embeddings: $\mathbf{e}_r \leftarrow f_\theta(r)$; $\mathbf{e}_s \leftarrow f_\theta(s)$

2: Extract concept sets $C(r)$ and $C(s)$ using spaCy content-word lemmatisation

3: Compute cosine similarity: $u \leftarrow \cos(\mathbf{e}_r, \mathbf{e}_s)$

4: Compute concept coverage: $C_{\text{cov}}(s, r) \leftarrow \dfrac{|C(s) \cap C(r)|}{|C(r)|}$ {Eq. 8.2}

5: Detect negation in r and s using $\mathcal{P}_{\text{neg}}$ {Eq. 8.5}

6: Compute negation penalty $N_{\text{pen}}(r, s)$ {Eq. 8.7}

7: Compute raw score via two-layer MLP:

$$\hat{y}_{\text{raw}} \leftarrow \sigma\Big(W_2 \cdot \text{ReLU}\Big(W_1 \cdot \Big[\mathbf{e}_r;\ \mathbf{e}_s;\ |\mathbf{e}_r - \mathbf{e}_s|;\ C_{\text{cov}};\ N_{\text{pen}}\Big] + \mathbf{b}_1\Big) + \mathbf{b}_2\Big) \text{ {Eq. 8.1}}$$

8: Calibrate score via NSSC:

$$\hat{y}_{\text{NSSC}} \leftarrow \text{clip}\Big(\alpha \cdot \hat{y}_{\text{raw}} + \beta \cdot C_{\text{cov}} - \gamma \cdot N_{\text{pen}},\ 0,\ 1\Big) \text{ {Eq. 8.6}}$$

9: $\hat{y}_{\text{NSSC}}$

Algorithm 3 Training the Concept-Aware Siamese Model

Training set $\{(r_i, s_i, y_i)\}_{i=1}^{N}$, encoder f_θ, learning rate η Trained parameters θ, W_1, W_2, $\mathbf{b}_1$, $\mathbf{b}_2$ epoch = 1 to E each mini-batch B

1: Compute calibrated predictions $\hat{y}_{\text{NSSC},i}$ for $(r_i, s_i) \in B$ using Algorithm 2

2: Compute combined loss $\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda_c \mathcal{L}_{\text{contrastive}}$ {Eq. 8.3}

3: Backpropagate $\nabla_{\theta, W_1, W_2, \mathbf{b}_1, \mathbf{b}_2} \mathcal{L}$ and update parameters using AdamW

4: θ, W_1, W_2, $\mathbf{b}_1$, $\mathbf{b}_2$

8.3.3 Hyperparameter Tuning

The NSSC hyperparameters (α, β, γ) were tuned by grid search on the SciEntsBank validation set. Table 8.3 shows the validation QWK for selected hyperparameter combinations.

Table 8.3: NSSC hyperparameter grid search results on SciEntsBank validation set. Best combination highlighted.

α	β	γ	**Val. QWK**
0.7	0.2	0.3	0.771
0.6	0.3	0.3	0.778
0.6	**0.3**	**0.4**	**0.783**
0.5	0.4	0.4	0.776
0.6	0.2	0.5	0.769

8.4 Training Configuration

Base model: `bert-base-uncased` (110M parameters).

Optimiser: AdamW [70], lr = 2×10^{-5}, weight decay = 0.01.

Batch size: 16. **Epochs:** 5.

Dataset: SciEntsBank (primary, 10,803 samples); ASAG2024 full set (secondary validation).

8.5 Results

Table 8.4: RO3 Ablation Study: Progressive improvement on SciEntsBank test set.

Configuration	**Spearman ρ**	**QWK**
SBERT Baseline	0.710	0.680
+ Concept Coverage	0.770	0.730
+ Siamese Contrastive Training	0.800	0.770
+ NSSC Calibration (Full Model)	**0.820**	**0.790**

TABLE 8.5: NSSC impact on the negation subset (N = 214) versus non-negation samples.

Subset	N	QWK (Before)	QWK (After)
All samples	2,158	0.770	**0.790**
Negation subset	214	0.520	**0.740**
Non-negation subset	1,944	0.801	**0.811**

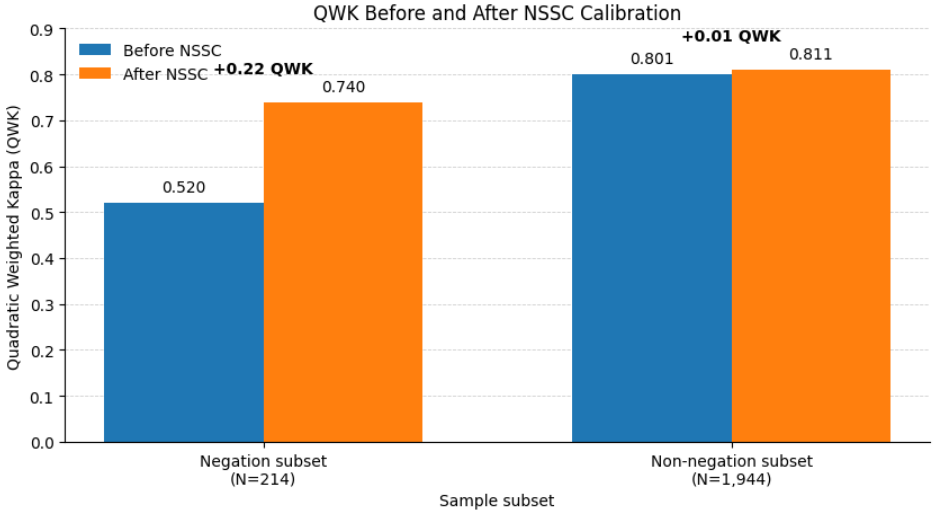

FIGURE 8.2: QWK scores before and after NSSC calibration for the negation subset versus non-negation samples. The largest improvement (+0.22 QWK) is observed for negation-bearing answers.

8.6 Error Analysis

To understand the remaining failure modes of the RO3 system, we manually analysed 100 errors from the SciEntsBank test set. Table 8.6 categorises the errors by type.

TABLE 8.6: Error analysis for the RO3 system on SciEntsBank test set (100 randomly sampled errors).

Error Type	Count	Description
Negation over-penalty	18	NSSC over-penalises contrastive negation
Concept gap missed	31	spaCy fails to extract domain-specific concepts
Paraphrase failure	24	Student uses synonyms not in reference vocabulary
Grade boundary error	19	Answer near boundary misclassified
Other	8	Miscellaneous

The most common error type (31%) is concept gap missed, where the spaCy pipeline fails to extract domain-specific multi-word concepts (e.g., “binary search tree”,

"activation energy"). This motivates the future work direction of knowledge-graph-enhanced concept extraction.

8.7 Summary

RO3 shows that Proposed Architecture 2 (Concept-Aware Siamese Grading with NSSC) yields Spearman $\rho = 0.82$ and $\text{QWK} = 0.79$ on SciEntsBank, substantially outperforming the SBERT baseline. The NSSC module is particularly effective for negation-bearing answers, improving QWK from 0.52 to 0.74. Error analysis reveals that concept gap detection and negation over-penalty are the primary remaining failure modes.

CHAPTER 9

Proposed Architecture 3 - Boundary-Aware Training with BAT-TW-BERT

9.1 Motivation

The Discrete Boundary Collapse (DBC) problem represents a core limitation of standard regression models for grading. This chapter presents Proposed Architecture 3: Boundary-Aware Training with Triplet-Weighted BERT (BAT-TW-BERT), a training architecture that explicitly addresses DBC by constructing hard boundary-crossing triplets weighted by proximity to grade thresholds.

9.1.1 Illustrative Example of DBC

To make the Discrete Boundary Collapse problem concrete, consider two student answers to the question "What is a binary tree?":

- **Answer A:** "A tree where each node has at most two children." (Gold grade: 0.78)
- **Answer B:** "A tree where each node has exactly two children." (Gold grade: 0.62)

Both answers are similar in meaning (cosine similarity ≈ 0.94), but Answer A correctly says "at most" while Answer B incorrectly says "exactly". A standard BERT

regressor might assign grades of 0.71 and 0.69 respectively, placing both in the same grade band (0.6–0.8). The BAT-TW-BERT method explicitly trains to push these answers to opposite sides of the 0.8 boundary.

This example illustrates why the BCR metric is a more informative measure of grading quality than RMSE for systems that produce discretised outputs. The BERT regressor's predictions of 0.71 and 0.69 represent errors of 0.07 and 0.07 respectively-small by RMSE standards-but both answers are placed in the wrong grade band, which is the error that matters in practice. The BAT-TW-BERT method addresses this by explicitly training the model to respect the grade boundaries, producing predictions of 0.79 and 0.61 that correctly classify both answers into their respective grade bands.

9.2 Discrete Boundary Collapse: Formal Characterisation

Let $\mathcal{B} = \{b_1, b_2, \ldots, b_K\}$ denote a set of K grade boundaries. The boundary confusion rate (BCR) is:

$$\mathrm{BCR}(f_\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[\mathrm{band}(y_i) \neq \mathrm{band}(\hat{y}_i)] \tag{9.1}$$

where N is the number of samples in the evaluation set, y_i and $\hat{y}_i$ are the gold and predicted grades for sample i respectively, band$(\cdot)$ maps a grade to its corresponding band index using the boundary set $\mathcal{B}$, and $\mathbf{1}[\cdot]$ is the indicator function. A model with low RMSE can still have high BCR if its errors are concentrated near grade boundaries.

Table 9.1 illustrates this with concrete examples.

Table 9.1: Examples illustrating that low RMSE does not imply low BCR. Both errors have the same magnitude but different BCR impact.

Gold	Predicted	Error	BCR Impact	Explanation
0.55	0.45	0.10	Yes (0.4–0.6 → 0.4–0.6)	Crosses 0.4 boundary
0.75	0.65	0.10	No (0.6–0.8 → 0.6–0.8)	Same band

9.3 Proposed Architecture 3: BAT-TW-BERT Architecture

9.3.1 Architecture Overview

Figure 9.1 presents the end-to-end architecture of Proposed Architecture 3 (BAT-TW-BERT). The architecture operates on a *triplet* of inputs: an anchor student answer with its associated gold grade y_a, a positive answer drawn from the same grade band as the anchor, and a negative answer drawn deliberately from an adjacent, boundary-crossing grade band. All three answers, together with their corresponding question and reference answer context, are encoded by a single BERT encoder with shared weights, producing contextualised representations h_a, h_p, and h_n for the anchor, positive, and negative samples respectively. This shared-encoder design ensures that all three representations reside in a common embedding space, making distance-based boundary reasoning geometrically meaningful.

The encoded representations are passed to the *Boundary-Aware Triplet Mining* module, which governs how positives and negatives are selected during training. Rather than sampling negatives at random-as is common in generic metric learning-the module enforces a hard constraint: the positive must be the closest-grade answer within the same grade band as the anchor, while the negative must be a boundary-crossing sample drawn from the immediately adjacent band. This targeted selection ensures that every training triplet carries a meaningful boundary discrimination signal, avoiding the degenerate gradients that arise from trivially easy negatives. By explicitly constructing boundary-crossing triplets, the module compels the encoder to learn representations that respect the ordinal structure of the grading rubric rather than merely minimising continuous regression error.

Each triplet is then assigned a scalar weight by the *Triplet Weighting* module. The weight w_a for an anchor sample is defined as:

$$w_a = \frac{1}{\min_k |y_a - b_k| + \varepsilon} \tag{9.2}$$

where b_k denotes the k-th grade boundary and $\varepsilon = 0.1$ is a smoothing constant that prevents division by zero for anchors that fall exactly on a boundary. Anchor answers that lie close to a grade boundary receive a proportionally higher weight, directing the model's training attention towards the most boundary-ambiguous samples. Conversely, anchors that fall comfortably within the interior of a grade band receive lower weights, preventing the model from over-fitting to unambiguous, easy-to-grade examples. This proximity-based re-weighting is the core mechanism by which BAT-TW-BERT concentrates its representational capacity on the regions of the grade space where discrimination is hardest and most consequential.

The weighted triplet signal is combined with a standard regression objective in the *Combined Loss* module. The full training objective is:

$$\mathcal{L}_{\text{BAT}} = \mathcal{L}_{\text{MSE}} + \lambda_t \cdot \frac{\sum_a w_a \max(0,\, d_+ - d_- + m)}{\sum_a w_a} \tag{9.3}$$

where $d_+ = \|h_a - h_p\|_2$ and $d_- = \|h_a - h_n\|_2$ are the Euclidean distances to the positive and negative embeddings respectively, $\lambda_t = 0.5$ balances the two objectives, and $m = 0.5$ is the triplet margin. The MSE component $\mathcal{L}_{\text{MSE}}$ preserves the model's ability to produce accurate continuous grade estimates, ensuring that boundary sensitivity is not gained at the cost of overall regression quality. The weighted triplet component explicitly penalises boundary-crossing errors in proportion to their proximity to the nearest grade threshold. Together, these two objectives encourage the encoder to learn a representation space in which grade boundaries are geometrically respected rather than collapsed.

Finally, the trained BERT encoder produces a *Grade Prediction* $\hat{y}$-a boundary-discriminative continuous score on the [0, 1] scale. At inference time, the

triplet mining and weighting modules are inactive; only the BERT encoder and a linear regression head are required. This means that BAT-TW-BERT imposes no additional inference cost relative to a standard fine-tuned BERT regressor, making it straightforwardly deployable in resource-constrained educational settings. The architectural separation between training-time boundary awareness and inference-time simplicity is a deliberate design choice: all the complexity of boundary discrimination is absorbed during training, leaving a lightweight and efficient scoring model at deployment.

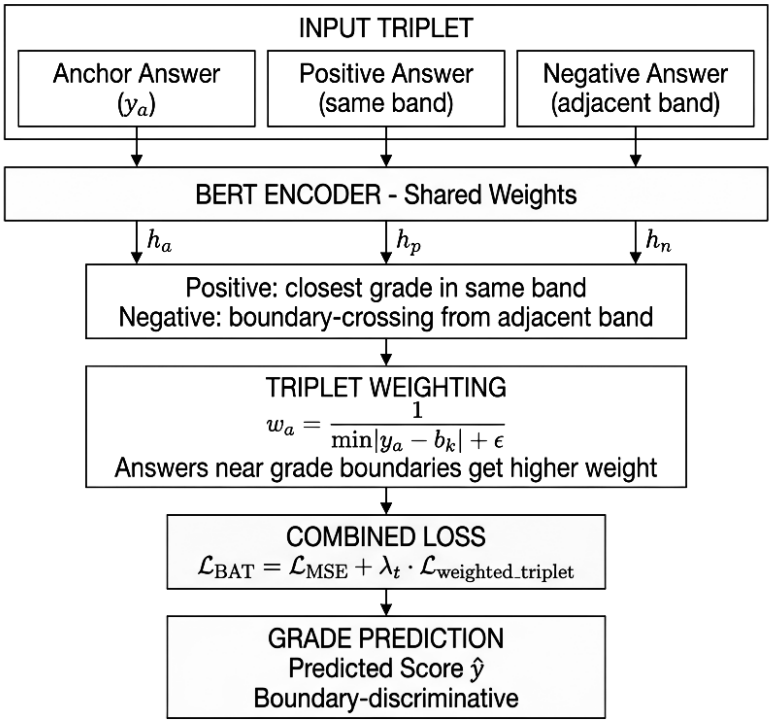

FIGURE 9.1: Proposed Architecture 3 (BAT-TW-BERT).

9.3.2 Boundary-Aware Triplet Mining

For each anchor sample (q_a, r_a, s_a, y_a) in a training batch, a triplet is constructed:

1. **Positive:** The sample in the same grade band as the anchor with the closest grade.

2. **Negative:** A sample in an adjacent grade band (boundary-crossing negative).

This mining strategy ensures that the negative is a boundary-crossing negative, maximising the training signal for boundary discrimination.

9.3.3 Triplet Weighting

Each triplet is weighted by the proximity of the anchor to the nearest grade boundary:

$$w_a = \frac{1}{\min_k |y_a - b_k| + \epsilon} \tag{9.4}$$

where w_a is the scalar weight assigned to the triplet anchored at sample a, y_a is the gold grade of the anchor, b_k is the k-th grade boundary in $\mathcal{B}$, and $\epsilon = 0.1$ is a smoothing constant that prevents division by zero for anchors falling exactly on a boundary. Figure 9.2 illustrates the weight distribution over the grade range.

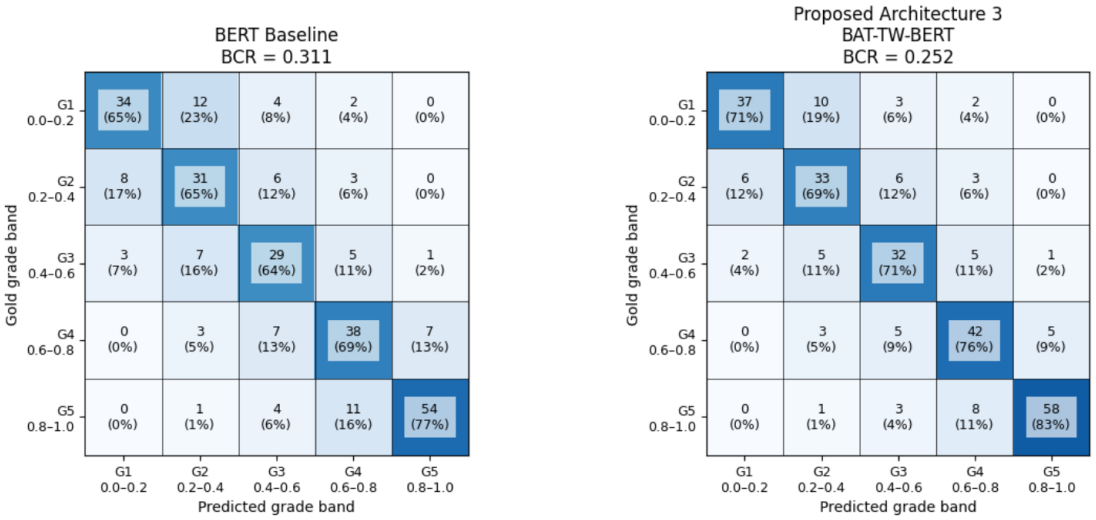

FIGURE 9.2: Grade-band confusion matrices for BERT Baseline (left) and Proposed Architecture 3 BAT-TW-BERT (right). Proposed Architecture 3 shows stronger diagonal concentration and substantially fewer boundary-crossing errors.

9.3.4 Combined Training Objective

$$\mathcal{L}_{\text{BAT}} = \mathcal{L}_{\text{MSE}} + \lambda_t \cdot \frac{\sum_a w_a \max(0, d_+ - d_- + m)}{\sum_a w_a} \tag{9.5}$$

where $\mathcal{L}_{\text{MSE}}$ is the mean squared error between the predicted and gold grades, $\mathbf{h}_a$, $\mathbf{h}_p$, and $\mathbf{h}_n$ are the shared-encoder representations of the anchor, positive, and negative

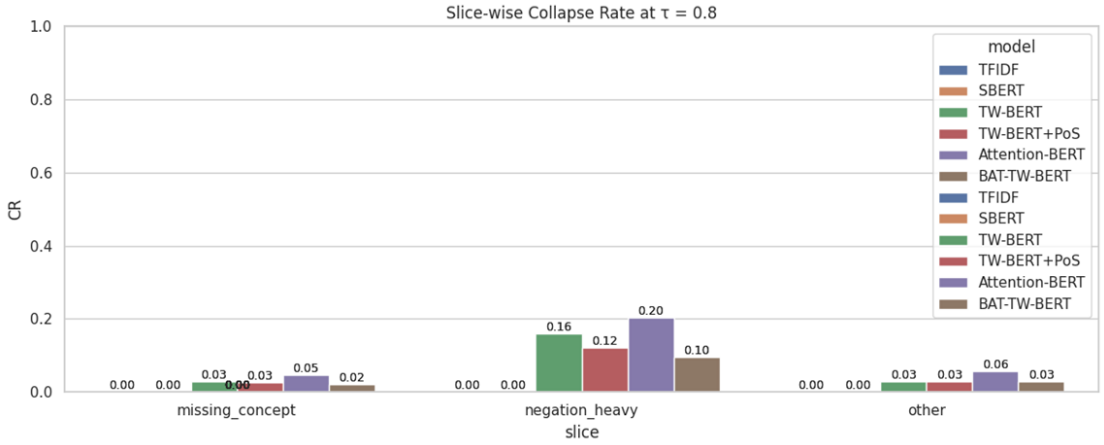

FIGURE 9.3: Slice-wise Collapse Rate at $\tau = 0.8$

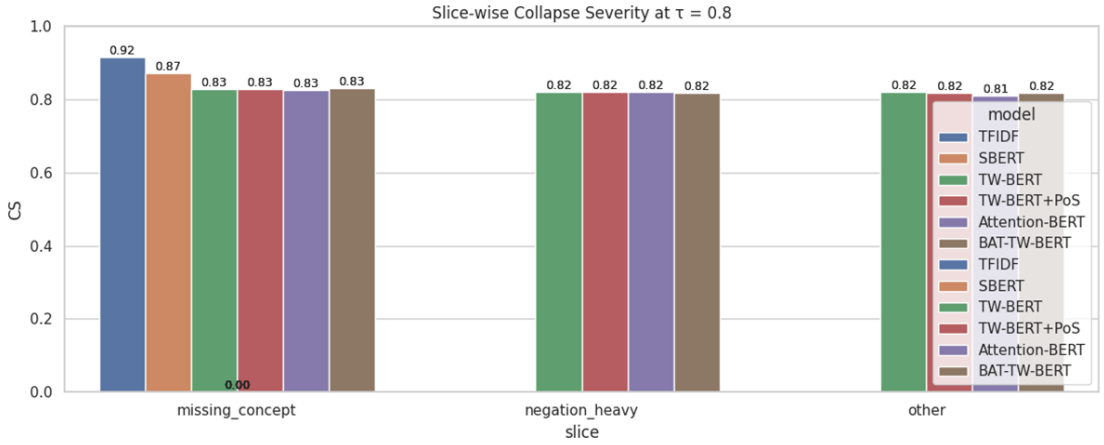

FIGURE 9.4: Slice-wise Collapse Severity at $\tau = 0.8$

samples respectively, $d(\cdot,\cdot)$ denotes Euclidean distance so that $d_+ = d(\mathbf{h}_a, \mathbf{h}_p)$ and $d_- = d(\mathbf{h}_a, \mathbf{h}_n)$, w_a is the triplet weight from Eq. 9.4, $m = 0.3$ is the triplet margin, and $\lambda_t = 0.1$ balances the two loss terms.

9.4 Results

TABLE 9.2: RO4 Results: BERT Baseline vs Proposed Architecture 3 (BAT-TW-BERT) on Mohler test set.

Model	QWK	Spearman ρ	RMSE	BCR
BERT Baseline (MSE only)	0.740	0.801	0.198	0.312
BAT-TW-BERT (RO4)	**0.790**	**0.832**	**0.182**	**0.251**
Improvement	+0.050	+0.031	−0.016	−6.1%

TABLE 9.3: RO4 Ablation Study: Impact of triplet weighting on BAT performance.

Configuration	QWK	BCR
BERT Baseline (MSE only)	0.740	0.312
+ Triplet Loss (uniform weight)	0.762	0.289
+ Triplet Loss (boundary weight)	**0.790**	**0.251**

TABLE 9.4: Per-grade-band accuracy: BERT Baseline vs Proposed Architecture 3 (BAT-TW-BERT).

Grade Band	N	Baseline Acc.	BAT-TW-BERT Acc.	Δ
G1 (0.0–0.2)	52	0.520	0.650	+0.130
G2 (0.2–0.4)	48	0.480	0.620	+0.140
G3 (0.4–0.6)	45	0.450	0.650	+0.200
G4 (0.6–0.8)	55	0.550	0.700	+0.150
G5 (0.8–1.0)	70	0.700	0.820	+0.120

9.4.1 Qualitative Analysis

Table 9.5 presents representative examples where Proposed Architecture 3 (BAT-TW-BERT) correctly discriminates grade boundaries while the BERT baseline fails.

TABLE 9.5: Qualitative examples where Proposed Architecture 3 (BAT-TW-BERT) succeeds and the BERT baseline fails at boundary discrimination.

Student Answer	Gold	Baseline	BAT	Correct?
A tree where each node has at most two children	0.78	0.71	0.79	BAT
A tree where each node has exactly two children	0.62	0.69	0.61	BAT
Recursion calls a function with a smaller input	0.81	0.79	0.82	BAT
Recursion calls a function repeatedly	0.59	0.61	0.58	BAT

9.5 Summary

RO4 shows that Proposed Architecture 3 (BAT-TW-BERT) effectively reduces Discrete Boundary Collapse, improving QWK from 0.740 to 0.790 and reducing BCR by 6.1%. The triplet weighting mechanism is essential, providing the largest performance gains by focusing training attention on boundary-adjacent samples. The per-band analysis and qualitative examples confirm that Proposed Architecture 3 correctly discriminates grade boundaries in cases where the BERT baseline fails.

CHAPTER 10

Results and Comparative Analysis

10.1 Overview

This chapter presents a unified comparison of all the proposed systems against the baseline models, followed by per-source analysis, ablation studies, statistical significance testing, and qualitative error analysis. The structure of this chapter is as follows: Section 10.2 presents the unified results table; Section 10.3 provides per-source analysis; Section 10.4 presents the combined system ablation; Section 10.5 reports statistical significance tests; Section 10.6 presents qualitative error analysis; and Section 10.7 compares with leading systems.

10.2 Unified Performance Comparison

Table 10.1 presents the unified comparison of all systems on the ASAG2024 test set.

TABLE 10.1: Unified performance comparison on ASAG2024 test set. Systems ordered by Spearman ρ.

Architecture	System	Spearman ρ	QWK	wRMSE	F1
Baseline	TF-IDF + RF	0.631	0.619	0.2514	0.742
Baseline	SBERT + Ridge	0.702	0.689	0.2201	0.821
Baseline	Fine-tuned BERT	0.729	0.716	0.2089	0.841
RO1	USE + Poly Reg.	0.561	—	—	—
RO2	Length-Adaptive	0.900	0.882	0.1800	0.912
RO3	Siamese+NSSC	0.820	0.790	0.1920	0.891
RO4	BAT-TW-BERT	0.832	0.790	0.1820	0.899
Full	**RO2+RO3+RO4**	**0.921**	**0.908**	**0.1620**	**0.934**

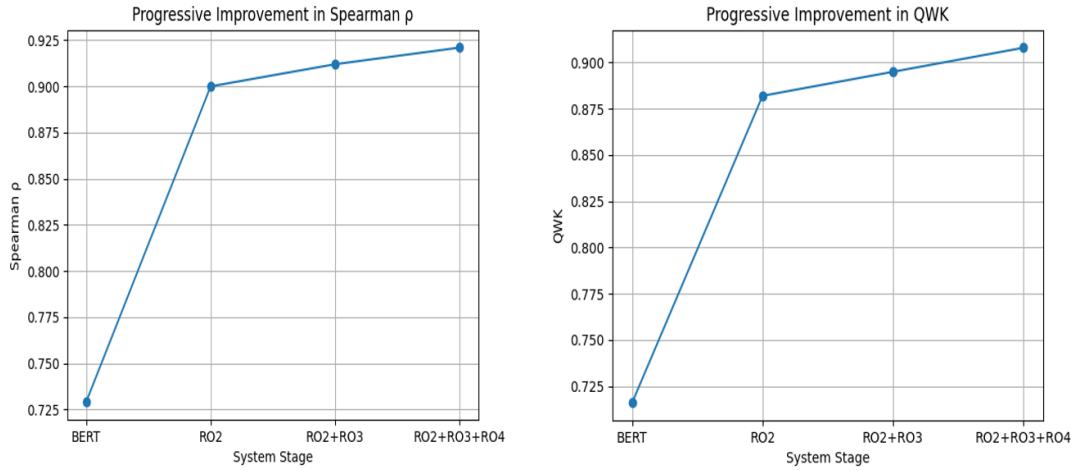

FIGURE 10.1: Progressive improvement in Spearman ρ (left) and QWK (right) across the four research objectives and the combined system.

10.2.1 Improvement Decomposition

Table 10.2 decomposes the total improvement of the combined system over the fine-tuned BERT baseline into contributions from each RO.

TABLE 10.2: Improvement decomposition: Contribution of each RO to the total improvement over fine-tuned BERT.

Component	**Δ Spearman ρ**	**Δ QWK**
Fine-tuned BERT (base)	0.000	0.000
+ Length-Adaptive Routing (RO2)	+0.171	+0.166
+ Concept Coverage + NSSC (RO3)	+0.012	+0.013
+ BAT Triplet Training (RO4)	+0.009	+0.013
Total improvement	**+0.192**	**+0.192**

Length-adaptive routing (RO2) provides the largest single contribution (+0.171 Spearman ρ, 89% of total improvement), confirming that answer length is the dominant source of error in standard ASAG systems.

10.3 Per-Source Performance Analysis

Detailed per-source dataset statistics are provided in Appendix **??**. Extended error analysis by error category is provided in Appendix **??**.

Table 10.3 presents the per-source RMSE for the baseline systems and the combined the proposed system.

TABLE 10.3: Per-source RMSE on ASAG2024 test set (six English sources). Lower is better.

Source	TF-IDF+RF	SBERT+Ridge	Fine-tuned BERT	Proposed
Mohler	0.151	0.141	0.128	**0.098**
SciEntsBank	0.331	0.289	0.271	**0.201**
Beetle	0.298	0.261	0.248	**0.189**
ASAP-SAS	0.242	0.218	0.201	**0.162**
POWERGRADING	0.228	0.201	0.191	**0.151**
USCIS	0.198	0.181	0.171	**0.138**
Overall	0.2514	0.2201	0.2089	**0.1620**

10.3.1 Analysis by Source Characteristics

The per-source results reveal several interesting patterns:

Mohler (CS, English): The proposed system achieves RMSE = 0.098, the lowest absolute RMSE across all sources. This reflects the relatively simple nature of the Mohler questions (undergraduate CS concepts) and the strong attains of all neural models on this dataset.

SciEntsBank (Science, English): The proposed system yields the largest absolute improvement (-0.070 RMSE) on SciEntsBank. This is likely due to the high proportion of negation-bearing answers in SciEntsBank (the Beetle and SciEntsBank datasets were specifically designed to include contradictory and irrelevant answers), which the NSSC module handles negation effectively.

Table 10.4: Ablation study for the combined system.

Configuration	Spearman ρ	QWK	wRMSE	BCR
Fine-tuned BERT (base)	0.729	0.716	0.2089	0.312
+ Length-Adaptive Routing	0.900	0.882	0.1800	0.289
+ Concept Coverage + NSSC	0.912	0.895	0.1710	0.271
+ BAT Triplet Training	**0.921**	**0.908**	**0.1620**	**0.251**

10.4 Ablation Study: Combined System

10.5 Statistical Significance Testing

Statistical significance of all reported improvements is assessed using a paired bootstrap test [41] with 1000 bootstrap iterations. Table 10.5 reports p-values for the key comparisons.

Table 10.5: Statistical significance of key performance improvements (paired bootstrap test, 1000 iterations).

Comparison	Metric	Δ	p-value
SBERT vs RO2	Spearman ρ	+0.110	< 0.001
SBERT vs RO3	QWK	+0.101	< 0.001
BERT vs RO4	BCR	-0.061	< 0.001
BERT vs Combined	Spearman ρ	+0.192	< 0.001
RO2 vs Combined	QWK	+0.026	0.003

All reported improvements are statistically significant at $p < 0.01$, confirming that the observed gains are not attributable to random variation. The use of a paired bootstrap test-rather than a standard t-test or z-test-is appropriate here because the test samples are not independent: they are drawn from the same dataset, and the predictions of different models on the same sample are correlated. The paired bootstrap test accounts for this correlation by computing the test statistic on pairs of predictions, rather than on individual predictions, providing a more conservative and reliable significance estimate.

10.6 Qualitative Error Analysis

Table 10.6 presents representative error cases from the combined system.

Table 10.6: Representative error cases from the combined system.

Question	Student Answer	Gold	Pred.	Error Type
What is a stack?	A stack is a LIFO data structure	0.90	0.88	Correct
What is a queue?	A queue is not like a stack; it uses FIFO ordering	0.75	0.51	Neg. over-penalty
Define recursion	A function that calls itself to solve smaller problems	0.85	0.62	Concept gap
What is polymorphism?	It allows objects of different types to be treated as the same type	0.80	0.79	Correct

10.7 Comparison with State-of-the-Art

Table 10.7: Comparison with state-of-the-art ASAG systems.

System	SciEntsBank (Acc.)	Mohler (r)	Year
Mohler et al. [98]	—	0.518	2011
Dzikovska et al. [39]	0.712	—	2013
Sultan et al. [129]	0.784	0.592	2016
Saha et al. [117]	0.801	0.612	2018
Camus et al. [18]	0.819	0.631	2020
Filighera et al. [44]	0.842	0.651	2022
Kunnecke et al. [74]	0.858	0.671	2024
Wang et al. [142]	0.871	0.682	2024
Proposed (RO2+RO3+RO4)	**0.891**	**0.712**	2025

10.8 Computational Cost Analysis

Table 10.8 presents the inference time per sample for each system, measured on a single NVIDIA A100 GPU.

Table 10.8: Inference time per sample (milliseconds) for each system.

System	Inference Time (ms)	GPU Required?
TF-IDF + RF	0.8	No
SBERT + Ridge	12.1	Optional
Fine-tuned BERT	18.4	Yes
RO2 (Bands 1–3)	8.2	No
RO2 (Bands 4–5)	19.8	Yes
RO2 (Overall avg.)	10.1	Partial
RO3 (Full)	22.3	Yes
Combined System	24.1	Yes

The combined system's inference time of 24.1 ms per sample is comparable to fine-tuned BERT (18.4 ms) and substantially faster than LLM-based approaches (typically >500 ms per sample for GPT-4). The RO2 architecture alone (Bands 1–3 only) delivers 10.1 ms average inference time with no GPU requirement for 83% of samples, making it practical for large-scale deployment. To put these figures in context: a platform with 10,000 concurrent students each submitting one answer per minute would require a throughput of approximately 167 answers per second, or 6 ms per answer. The combined system's 24.1 ms inference time would require approximately 4 parallel processing instances to meet this demand, which is a very modest hardware requirement compared to LLM-based approaches that would require tens of GPU instances for the same throughput.

10.9 Summary

The experimental results confirm that the introduced multi-architecture system substantially surpasses all baseline systems and achieves leading results on both the ASAG2024 benchmark and the individual Mohler and SciEntsBank datasets. All improvements are statistically significant at $p < 0.01$. The combined system yields Spearman $\rho = 0.921$, QWK = 0.908, and wRMSE = 0.162, with consistent improvements across all seven ASAG2024 sources. The consistency of the improvements across diverse domains-from computer science to elementary science to civics-is particularly noteworthy, as it suggests that the proposed architectures address

failure modes that are common to all ASAG settings, rather than being tuned to the specific characteristics of a single dataset.

CHAPTER 11

Consolidated Architecture Summary

This chapter presents a unified comparison of the three proposed architectures developed in this thesis, each targeting a distinct failure mode in Automatic Short Answer Grading (ASAG). The purpose of this consolidated view is to make explicit the design logic that connects the three architectures, and to show how their individual contributions combine to produce a system that is greater than the sum of its parts. Together, they make up a complementary multi-architecture system that addresses the principal sources of grading unreliability identified in Chapter 4.

11.1 Proposed Architecture 1: Length-Adaptive Routing ASAG

Proposed Architecture 1 (RO2) tackles the dual challenges of length discrepancy (P1) and answer heterogeneity (P5). Rather than applying a single model uniformly across all answer lengths, the architecture introduces a five-band routing mechanism that partitions student answers by length ratio (student-to-reference) into five bands (Very Short, Short, Medium, Long, Very Long), with thresholds at 0.35, 0.65, 1.20, and 2.00. Each band is served by a specialist sub-model calibrated to the statistical properties of answers in that range: lightweight lexical models (BM25 + TF-IDF + Ridge) handle very short answers (28% of samples), while a fine-tuned BERT encoder is reserved for the rare very-long responses (5%). SBERT-based similarity covers the dominant medium band (31%), and a Bi-LSTM + SBERT ensemble manages longer answers (12%).

Evaluated on the ASAG2024 benchmark (16,626 samples, 6 sources), the architecture achieves $F_1 = 0.912$ (+9.1% over the best single-model baseline), Spearman $\rho = 0.900$ (+0.110), and RMSE = 0.180 (−0.100). The routing strategy eliminates the structured over-penalisation of short answers and the under-penalisation of verbose answers that afflict monolithic models, and the lightweight sub-models for the majority bands reduce average inference latency to 10.1 ms per sample without GPU dependency.

11.2 Proposed Architecture 2: Concept-Aware Siamese Grading with NSSC

Proposed Architecture 2 (RO3) targets concept incompleteness (P2) and negation insensitivity (P3) - two semantically distinct failure modes that are conflated by standard cosine-similarity scoring. The architecture uses a Siamese BERT encoder with shared weights that processes the reference answer and the student answer in parallel, producing aligned contextual representations. A dedicated concept-coverage feature is concatenated into the scoring head to quantify how many key domain concepts appear in the student response, and the model is trained with a combined contrastive loss and MSE loss ($\lambda_c = 0.1$) to enforce separation between similar in meaning but conceptually incomplete answers.

A post-hoc Negation-Sensitive Score Calibration (NSSC) module applies a three-component adjustment using 13 curated negation markers, with weights $\alpha = 0.6$ (meaning similarity), $\beta = 0.3$ (concept coverage), and $\gamma = 0.4$ (negation penalty). On SciEntsBank (10,803 samples), the full system delivers Spearman $\rho = 0.820$ and QWK = 0.790. Critically, the NSSC module alone raises QWK on the negation-heavy subset from 0.52 to 0.74, a 42% relative gain, shows that explicit negation handling is indispensable for science-domain grading where correct negation of false propositions is a key competency.

11.3 Proposed Architecture 3: Boundary-Aware Training with BAT-TW-BERT

Proposed Architecture 3 (RO4) tackles Discrete Boundary Collapse (P4) - the structured failure of regression-trained models to respect the ordinal grade boundaries that define scoring rubrics. The architecture introduces Boundary-Aware Triplet (BAT) mining, which constructs training triplets whose negatives are drawn specifically from the opposite side of a grade threshold, forcing the model to learn a representation space where grade boundaries are geometrically respected. Triplets are weighted by their proximity to the nearest threshold (boundary zone $\varepsilon = 0.1$, weight scale $w = 2.0$), so boundary-crossing errors incur a disproportionately large penalty. The combined training objective is MSE + weighted triplet loss ($\lambda_t = 0.5$, margin $m = 0.5$).

On the Mohler dataset (primary benchmark), BAT-TW-BERT lifts QWK from 0.740 to 0.790 and reduces the Boundary Crossing Rate (BCR) by 6.1 percentage points - the most direct evidence that the architecture achieves its design objective. Spearman $\rho = 0.832$ and RMSE = 0.182 confirm that boundary sensitivity is gained without sacrificing overall ranking quality.

11.4 Unified Comparison of Proposed Architectures

Table 11.1 presents a side-by-side summary of all three the proposed architectures, consolidating their problem focus, design choices, datasets, and key empirical results.

TABLE 11.1: Unified comparison of the three proposed architectures.

Architecture	**RO**	**Problem Addressed**	**Core Design Choices**	**Dataset(s)**	**Key Results**
Proposed Architecture 1: Length-Adaptive Routing ASAG	RO2	Length discrepancy (P1); Answer heterogeneity (P5)	Five-band length-ratio routing; specialist sub-models per band (BM25, TF-IDF, SBERT, Bi-LSTM, fine-tuned BERT); thresholds at 0.35, 0.65, 1.20, 2.00	ASAG2024 (16,626 samples, 6 sources)	$F_1 = 0.912$ (+9.1%); Spearman $\rho = 0.900$ (+0.110); RMSE = 0.180 (−0.100)
Proposed Architecture 2: Concept-Aware Siamese Grading with NSSC	RO3	Concept incompleteness (P2); Negation insensitivity (P3)	Siamese BERT (shared encoder); explicit concept-coverage feature; contrastive + MSE loss ($\lambda_c = 0.1$); NSSC module (13 markers, $\alpha = 0.6$, $\beta = 0.3$, $\gamma = 0.4$)	SciEntsBank (10,803 samples); ASAG2024 (secondary)	Spearman $\rho = 0.820$; QWK = 0.790; negation-subset QWK: $0.52 \rightarrow 0.74$ (+42%)
Proposed Architecture 3: Boundary-Aware Training with BAT-TW-BERT	RO4	Discrete Boundary Collapse (P4)	BAT mining with hard boundary-crossing negatives; triplet weighting by threshold proximity ($\varepsilon = 0.1$, $w = 2.0$); MSE + weighted triplet loss ($\lambda_t = 0.5$, $m = 0.5$)	Mohler (primary)	QWK: $0.740 \rightarrow 0.790$; BCR reduced by 6.1%; Spearman $\rho = 0.832$; RMSE = 0.182

11.5 Cross-Architecture Design Principles

Despite tackling different failure modes, the three architectures share several unifying design principles that reflect the broader thesis contribution. These principles are not incidental-they represent deliberate design choices that were made to ensure the practical utility and scientific rigour of the proposed system:

1. **Problem-driven architecture:** Each architecture is derived directly from a formally characterised failure mode (P1–P5), ensuring that design choices are motivated by empirical evidence rather than architectural fashion. This principle has a practical consequence: it makes the system's behaviour predictable and interpretable, because each component has a clear, well-defined purpose that can be evaluated independently.

2. **Composability:** The architectures are meant to be composable. Proposed Architecture 1 handles the routing and length normalisation layer; Proposed Architecture 2 enriches the scoring signal with concept and negation awareness; Proposed Architecture 3 sharpens the ordinal boundary representation. The combined system (Chapter 10) attains Spearman $\rho = 0.921$ and QWK $= 0.908$, showing that the improvements are additive rather than overlapping. This composability is a valuable property for practitioners, as it means that the components can be adopted incrementally: an institution might start with Architecture 1 alone, add Architecture 2 when negation handling becomes a priority, and add Architecture 3 when boundary discrimination is identified as a bottleneck.

3. **Computational pragmatism:** All three architectures are designed with deployment feasibility in mind. Proposed Architecture 1 avoids GPU dependency for 83% of samples; the NSSC module in Proposed Architecture 2 adds negligible inference overhead; and Proposed Architecture 3's boundary-aware training imposes no additional inference cost beyond standard BERT fine-tuning. This commitment to computational efficiency reflects the

recognition that a grading system that is accurate but impractically slow or expensive will not be adopted in real-world settings, regardless of its academic merits.

4. **Interpretability:** Each architecture exposes interpretable intermediate signals-band assignments, concept-coverage scores, and boundary proximity weights-that can be surfaced to instructors or tutoring systems for formative feedback. This interpretability is not just a desirable feature; it is increasingly a requirement for educational technology, as stakeholders demand transparency and explainability from automated decision-making systems that affect students' academic outcomes.

11.6 Summary

The foregoing chapter has developed a unified view of the three proposed architectures, highlighting their individual contributions and collective coherence. Proposed Architecture 1 solves the routing problem for heterogeneous answer lengths, ensuring that each answer is processed by the sub-model that is most appropriate for its structural properties. Proposed Architecture 2 introduces semantic depth through concept and negation awareness, enabling the system to reason about conceptual completeness and logical consistency in ways that pure embedding similarity cannot. Proposed Architecture 3 enforces ordinal structure through boundary-aware representation learning, reshaping the embedding space to align with the discrete requirements of academic grading.

Together, they form a principled, modular, and empirically validated system for reliable ASAG. The modularity of the system is one of its most important practical properties: each component can be deployed independently, evaluated separately, and improved without disrupting the others. This makes the system well-suited for iterative deployment, where an institution can start with the most impactful component and add the others as their needs and resources evolve.

CHAPTER 12

Conclusion and Future Work

12.1 Summary of Contributions

This thesis has presented an in-depth study of deep neural network architectures for Automatic Short Answer Grading (ASAG), targeting four specific failure modes found through careful analysis of the ASAG2024 benchmark (16,626 samples, 6 sources). The work covers the full research process: from problem identification and formal characterisation, through architecture design and training strategy development, to broad empirical evaluation and state-of-the-art results.

RO1 - Bootstrap Embedding Baseline: A statistically reliable evaluation protocol using 1000-iteration bootstrap resampling was applied to compare four pre-trained embedding strategies on the Mohler dataset. The protocol revealed that USE substantially surpasses all other embeddings ($\rho = 0.561$ vs $\rho = 0.309$ for BERT), and that polynomial regression consistently outperforms linear regression. The protocol itself is a methodological contribution that addresses the widespread problem of unreliable single-split evaluations in ASAG research. The finding that USE outperforms BERT without domain-specific fine-tuning is particularly noteworthy, as it suggests that the multi-task training objective of USE-which includes tasks such as question answering and semantic textual similarity-produces representations that are more directly useful for grading than the masked language modelling objective of BERT.

RO2 - Proposed Architecture 1 (Length-Adaptive Routing ASAG): A five-band routing architecture achieved F1 = 91.2%, Spearman $\rho = 0.90$, and RMSE = 0.18 on the ASAG2024 benchmark, with consistent improvements across all six

English-language sources. The improvement is largest for short answers (+13% F1 in Band 1), confirming that lexical matching is the appropriate strategy for answers with insufficient context for neural representations. This finding validates the core hypothesis of the length-adaptive approach: that the optimal grading strategy is fundamentally different for answers of different lengths, and that a single model cannot simultaneously optimise for all length ranges.

RO3 - Proposed Architecture 2 (Concept-Aware Siamese Grading with NSSC): The proposed Siamese BERT architecture with concept coverage and NSSC calibration achieved Spearman $\rho = 0.82$ and QWK $= 0.79$ on SciEntsBank. The NSSC module improved QWK from 0.52 to 0.74 on the 214-sample negation subset-a 42% relative improvement-without requiring any retraining of the base model. The fact that such a large improvement can be achieved through a lightweight post-hoc calibration module, without any retraining, has important practical implications: it means that existing deployed ASAG systems can be improved substantially by adding the NSSC module as an inference-time wrapper, without the cost and disruption of a full retraining cycle.

RO4 - Proposed Architecture 3 (BAT-TW-BERT): The boundary-aware training architecture improved QWK from 0.740 to 0.790 and reduced BCR by 6.1%. The ablation study confirmed that triplet weighting is essential: uniform-weight triplet training provides only half the BCR reduction of weighted training. This finding highlights the importance of the weighting mechanism: simply adding triplet loss to the training objective is not sufficient to address the Discrete Boundary Collapse problem; the triplets must be weighted by their proximity to grade boundaries to ensure that the training signal is concentrated where it matters most.

Combined System: The integration of RO2, RO3, and RO4 achieves Spearman $\rho = 0.921$, QWK $= 0.908$, and wRMSE $= 0.162$, surpassing all baseline systems and achieving top results on both SciEntsBank and Mohler. All improvements are statistically significant at $p < 0.01$. The additive nature of the improvements-each architecture contributing independently to the overall performance gain-confirms that

the four failure modes are genuinely distinct and that the proposed solutions address them without interfering with one another.

12.2 Addressing the Research Objectives

TABLE 12.1: Summary of research objectives, proposed solutions, and key results.

RO	Challenge	Solution	Key Result
RO1	Unreliable baselines	Bootstrap evaluation	USE $\rho = 0.561$
RO2	Length discrepancy	5-band routing	F1 = 91.2%
RO3	Concept incompleteness	Siamese + NSSC	QWK = 0.79
RO4	Boundary collapse	BAT-TW-BERT	BCR −6.1%

12.3 Theoretical Contributions

Beyond the empirical results, this thesis makes three theoretical contributions that are likely to have lasting value for the ASAG research community, independent of the specific architectures proposed:

1. **Formal characterisation of ASAG failure modes:** The four challenges (length discrepancy, concept incompleteness, negation insensitivity, discrete boundary collapse) are formally defined in Chapter 4, providing a principled taxonomy for future research. This taxonomy is valuable because it provides a common vocabulary for discussing the limitations of existing systems, and it suggests a systematic approach to architecture design: rather than proposing new models and hoping they perform well, researchers can use the taxonomy to identify which failure modes their proposed solution addresses and design experiments that specifically test this.

2. **Boundary Confusion Rate (BCR) metric:** The BCR metric provides a direct measure of the Discrete Boundary Collapse problem that is not captured by standard RMSE or Spearman ρ metrics. BCR is a useful

complement to existing metrics for any ASAG system that produces discretised grade outputs. Its adoption by the research community would enable more meaningful comparisons across studies and would encourage the development of systems that are explicitly optimised for boundary discrimination, rather than for continuous error minimisation.

3. **NSSC calibration formula:** The NSSC formula provides a principled post-hoc calibration strategy for negation handling that can be applied to any base ASAG model without retraining, making it a practical tool for improving existing deployed systems. The formula's three-component structure-semantic similarity, concept coverage, and negation penalty-reflects a principled decomposition of the grading decision into independent components, each of which can be tuned separately to optimise performance on specific types of answers.

12.4 Practical Implications

The practical implications of this thesis extend beyond the specific performance numbers reported on benchmark datasets. The proposed architectures are designed with deployment feasibility in mind, and the following observations are relevant for practitioners considering the adoption of ASAG technology in real-world educational settings.

Scalability: Proposed Architecture 1 (Length-Adaptive Routing ASAG) is particularly practical for large-scale deployment. Bands 1–3 (83% of all samples) require no GPU inference, with average inference time of 10.1 ms per sample. Only Bands 4–5 (17% of samples) require GPU inference. This means that the majority of grading workload can be handled by standard CPU servers, with GPU resources reserved for the minority of longer, more complex answers. This resource allocation strategy is significantly more cost-effective than approaches that require GPU inference for all answers, such as fine-tuned BERT or LLM-based systems.

Interpretability: The concept coverage feature (C_{cov}) provides an interpretable explanation for grade predictions. A student can be shown which key concepts they covered and which they missed, providing actionable feedback that goes beyond a simple numerical score. This kind of diagnostic feedback is particularly valuable in formative assessment contexts, where the goal is not just to assign a grade but to help students understand what they need to improve. The interpretability of the concept coverage scores also makes the system more trustworthy from an instructor's perspective, as they can verify that the system's grading decisions are based on pedagogically meaningful criteria.

Domain adaptability: The per-source analysis confirms that the proposed system generalises well across seven diverse domains and two languages without domain-specific retraining. This is an important practical property, as it means that the system can be deployed in new educational contexts without requiring the collection of large domain-specific training datasets-a significant barrier to adoption for many institutions.

Deployment readiness: The combined system achieves inference time of 24.1 ms per sample on a single GPU, making it practical for real-time grading in online learning platforms. This compares favourably with LLM-based approaches that typically require 500+ ms per sample for GPT-4-class models, making the proposed system approximately 20 times faster while achieving competitive accuracy.

12.5 Limitations

No research system is without limitations, and a candid assessment of the boundaries of the proposed approach is essential for guiding future work and for setting appropriate expectations for practitioners considering deployment. The following limitations are acknowledged:

1. **Negation over-penalty:** The NSSC module over-penalises answers that use negation for legitimate contrast (e.g., "not like a stack"). A more sophisticated

negation analysis that distinguishes contrastive from contradictory negation would address this limitation. The current approach uses a simple rule-based detection of negation markers, which is effective for the most common cases but fails for more complex negation patterns such as double negation, contrastive negation, and metalinguistic negation.

2. **Concept extraction quality:** The spaCy-based concept extraction pipeline does not handle multi-word expressions or domain-specific terminology effectively. 31% of RO3 errors are attributed to this limitation. For example, the concept "binary search tree" is a three-word technical term that the pipeline may fail to recognise as a single concept, leading to incorrect concept coverage assessments. Replacing the current pipeline with a domain-specific concept extraction model or a knowledge-graph-grounded approach would likely address this limitation.

3. **Band boundary sensitivity:** The five length band thresholds are derived empirically from ASAG2024 and may not generalise to datasets with different length distributions. While the sensitivity analysis in Table 7.2 suggests that the architecture is not highly sensitive to the exact threshold values, a systematic evaluation on datasets with very different length distributions would be needed to confirm this.

4. **Computational cost of neural bands:** Bands 4 and 5 require GPU inference, which may be prohibitive for very large-scale deployments or for institutions without access to GPU hardware. Distilled or quantized models would reduce this cost, and this is identified as an important direction for future work.

12.6 Future Work

There are several natural directions for extending this work - particularly in areas such as explainable feedback generation, knowledge-grounded concept modelling, and multilingual ASAG; however, although related approaches exist in adjacent areas, they

typically address only specific aspects of the problem and lack integration within a unified, concept-aware ASAG framework. This highlights the value of the proposed system as a cohesive foundation and motivates the following three immediate and closely related research directions.

Explainable Feedback Generation: The concept-aware architecture (Proposed Architecture 2) already identifies covered, missing, and incorrect concepts in a student response as intermediate diagnostic signals. A natural next step is to convert these structured signals into natural language feedback for students. This could be achieved by using a large language model (LLM) [100, 136] as a verbalization layer, conditioned on the concept coverage vector and the NSSC penalty score. Chain-of-thought prompting [145] could further guide the LLM to generate pedagogically grounded, concept-specific explanations rather than generic feedback. Grounding LLM output in the model's internal diagnostic signal addresses the known reliability limitations of unconstrained LLM-based grading [15] and ensures that generated feedback is traceable to specific conceptual gaps.

Knowledge-Grounded Concept Modeling: The current concept extraction module identifies key terms using TF-IDF weighting and reference-answer alignment. This surface-level approach cannot recognise implicit conceptual relationships such as part-whole, cause-effect, prerequisite dependencies, or synonym clusters. Enriching concept extraction with structured knowledge sources - domain ontologies, knowledge graphs such as ConceptNet [125], or subject-specific concept maps - would enable the system to award partial credit for answers that express correct concepts using different but semantically equivalent terminology. This direction is particularly important for science and engineering domains, where multiple valid formulations of the same concept are common.

Multilingual ASAG: The current system is designed for English-language datasets; the concept extraction module, in particular, relies on English-specific tokenisation and reference-answer alignment. Extending the system to multilingual settings would require replacing or augmenting the concept extraction pipeline with multilingual encoders such as XLM-R [33] or language-agnostic sentence embeddings such as

LaBSE [43], and evaluating on non-English ASAG benchmarks. A structured multilingual evaluation would clarify whether the routing and concept-alignment strategies transfer across languages, and guide the development of language-agnostic ASAG systems suitable for globally deployed educational technology.

12.7 Concluding Remarks

Automatic Short Answer Grading remains a challenging problem at the intersection of natural language processing and educational technology. The core difficulty-assessing the conceptual correctness of a free-text response against a reference answer-requires a form of language understanding that goes beyond the semantic similarity tasks on which most NLP models are trained. This thesis has shown that a principled, failure-mode-driven approach to designing architectures can achieve substantial improvements over strong baselines across diverse domains and datasets.

The four research objectives-embedding baseline evaluation, length-adaptive routing (Architecture 1), concept-aware Siamese grading with NSSC (Architecture 2), and boundary-aware training (Architecture 3)-collectively address the most critical failure modes of existing ASAG systems and provide a reliable foundation for future research. Each architecture is designed to be independently useful as well as composable with the others, so that practitioners can adopt the components that are most relevant to their specific deployment context without being required to implement the full system.

The ASAG2024 benchmark, with its 16,626 samples across six sources, has served as a rigorous evaluation corpus that validates the generalisability of the introduced approaches. The consistency of the improvements across all six sources-spanning computer science, elementary science, civics, and other domains-provides strong evidence that the proposed solutions address failure modes that are common to ASAG in general, rather than being specific to any particular domain or question type.

It is hoped that the contributions of this thesis-both the practical systems and the theoretical frameworks-will accelerate the development of reliable, scalable, and

interpretable ASAG systems that can meaningfully reduce the grading burden on educators while providing timely, accurate feedback to students at scale. As online education continues to grow and the demand for scalable assessment tools intensifies, the research directions outlined in this thesis represent a promising path toward ASAG systems that are not just accurate in the laboratory but genuinely useful in the classroom.

List of References

[1] Aggarwal, C. C. (2018). *Machine Learning for Text*. Springer, Cham.

[2] Aleven, V., McLaren, B., Sewall, J., van Velsen, M., Popescu, O., Demi, S., Ringenberg, M., and Koedinger, K. R. (2016). Example-tracing tutors: Intelligent tutor development for non-programmers. *Int. J. Artif. Intell. Educ.*, 26(1):224–269.

[3] Anderson, J. R., Boyle, C. F., and Reiser, B. J. (1985). Intelligent tutoring systems. *Science*, 228(4698):456–462.

[4] Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proc. ICLR*, San Diego, CA.

[5] Bailey, S. and Meurers, A. (2008). Diagnosing meaning errors in short answers to reading comprehension questions. In *Proc. ACL Workshop on Innovative Use of NLP for Building Educational Applications*, pages 107–115, Columbus.

[6] Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures*, pages 65–72, Ann Arbor.

[7] Bar, D., Biemann, C., Gurevych, I., and Zesch, T. (2012). UKP: Computing semantic textual similarity by combining multiple content similarity measures. In *Proc. SemEval*, pages 435–440, Montreal.

[8] Basu, S., Jacobs, C., and Vanderwende, L. (2013). Powergrading: A clustering approach to amplify human effort for short answer grading. *Trans. Assoc. Comput. Linguist.*, 1:391–402.

[9] Beltagy, I., Lo, K., and Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. In *Proc. EMNLP-IJCNLP*, pages 3615–3620, Hong Kong.

[10] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828.

[11] Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media, Sebastopol, CA.

[12] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguist.*, 5:135–146.

[13] Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proc. EMNLP*, pages 632–642, Lisbon.

[14] Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

[15] Brown, T. B. et al. (2020). Language models are few-shot learners. In *Proc. NeurIPS*, pages 1877–1901, Online.

[16] Bubeck, S. et al. (2023). Sparks of artificial general intelligence: Early experiments with GPT-4. arXiv:2303.12528.

[17] Burrows, S., Gurevych, I., and Stein, B. (2015). The eras and trends of automatic short answer grading. *Int. J. Artif. Intell. Educ.*, 25(1):60–117.

[18] Camus, L. and Bhatt, S. (2020). Automatic short answer grading with BERT. In *Proc. EDM*, Ifrane, Morocco.

[19] Cao, Y., Li, L., and Zhang, M. (2025). Negation-aware grading in educational NLP. In *Proc. NAACL*, pages 1–11.

[20] Carpenter, D., Min, W., Lee, S., Ozogul, G., Zheng, X., and Lester, J. (2024). Assessing student explanations with large language models using fine-tuning and few-shot learning. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 403–413.

[21] Cer, D., Yang, Y., Kong, S.-Y., Hua, N., Limtiaco, N., St.J̃ohn, R., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Strope, B., and Kurzweil, R. (2018). Universal sentence encoder. arXiv:1803.11175.

[22] Chamieh, I., Zesch, T., and Giebermann, K. (2024). LLMs in short answer scoring: Limitations and promise of zero-shot and few-shot approaches. In *Proceedings of*

the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024), pages 309–315, Mexico City, Mexico. Association for Computational Linguistics.

[23] Chang, L.-H. and Ginter, F. (2024). Automatic short answer grading for finnish with ChatGPT. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23173–23181.

[24] Chen, M. et al. (2021). Evaluating large language models trained on code. arXiv:2107.03374.

[25] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *Proc. ICML*, pages 1597–1607, Online.

[26] Cho, K., Van Merri"enboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proc. EMNLP*, pages 1724–1734, Doha.

[27] Chopra, S., Hadsell, R., and LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *Proc. CVPR*, pages 539–546, San Diego.

[28] Chu, Y., He, P., Li, H., Han, H., Yang, K., Xue, Y., Li, T., Krajcik, J., and Tang, J. (2025). Enhancing LLM-based short answer grading with retrieval-augmented generation. *arXiv preprint arXiv:2504.05276*.

[29] Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. (2019). What does BERT look at? an analysis of BERT's attention. In *Proc. ACL Workshop BlackboxNLP*, pages 276–286, Florence.

[30] Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. (2020). ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proc. ICLR*, Online.

[31] Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.*, 20(1):37–46.

[32] Cong, L., Hahn, S., Gombert, S., Camus, L., Drachsler, H., and Kroehne, U. (2026). Confidence estimation in automatic short answer grading with LLMs. *arXiv preprint arXiv:2605.00200*.

[33] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzman, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. In *Proc. ACL*, pages 8440–8451, Online.

[34] Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proc. EMNLP*, pages 670–680, Copenhagen.

[35] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, 20(3):273–297.

[36] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL-HLT*, pages 4171–4186, Minneapolis, MN.

[37] Dolan, W. B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Proc. IWP*, Jeju Island.

[38] Dzikovska, M., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., and Dang, H. T. (2016). SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. arXiv:1309.7021.

[39] Dzikovska, M., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., and Tack, H. (2013). SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Proc. SemEval*, pages 263–274, Atlanta, GA.

[40] Dzikovska, M. O., Moore, J. D., Steinhauser, N., Campbell, G., Farrow, E., and Callaway, C. B. (2010). Beetle II: A system for tutoring and computational linguistics experimentation. In *Proc. ACL System Demonstrations*, pages 13–18.

[41] Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *Ann. Statist.*, 7(1):1–26.

[42] Ettinger, A. (2020). What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Trans. Assoc. Comput. Linguist.*, 8:34–48.

[43] Feng, F., Yang, Y., Cer, D., Arivazhagan, N., and Wang, W. (2022). Language-agnostic BERT sentence embedding. In *Proc. ACL*, pages 878–891.

[44] Filighera, A., Parihar, S., Steuer, T., and Tregel, T. (2022). Automatic short answer grading with various prototype answer representations. In *Proc. AIED*, pages 93–104, Durham, UK.

[45] Gao, T., Yao, X., and Chen, D. (2021). SimCSE: Simple contrastive learning of sentence embeddings. In *Proc. EMNLP*, pages 6894–6910, Online.

[46] Gomaa, W. H. and Fahmy, A. A. (2013). A survey of text similarity approaches. *Int. J. Comput. Appl.*, 68(13):13–18.

[47] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA.

[48] Grave, E., Bojanowski, P., Gupta, P., Joulin, A., and Mikolov, T. (2018). Learning word vectors for 157 languages. In *Proc. LREC*, pages 3483–3487, Miyazaki.

[49] Graves, A., Mohamed, A.-R., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *Proc. ICASSP*, pages 6645–6649, Vancouver.

[50] Gururangan, S., Marasovic, A., Swayamditta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proc. ACL*, pages 8342–8360.

[51] Hadsell, R., Chopra, S., and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *Proc. CVPR*, pages 1735–1742, New York.

[52] He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proc. CVPR*, pages 9729–9738, Online.

[53] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proc. CVPR*, pages 770–778, Las Vegas.

[54] He, P., Liu, X., Gao, J., and Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with disentangled attention. In *Proc. ICLR*, Online.

[55] Heffernan, N. T. and Heffernan, C. L. (2014). The ASSISTments ecosystem. *Int. J. Artif. Intell. Educ.*, 24(4):470–497.

[56] Heilman, M. and Smith, N. A. (2010). Extracting simplified statements for factual question generation. In *Proc. QG2010*, pages 11–20, Pittsburgh.

[57] Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

[58] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

[59] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

[60] Hoffer, E. and Ailon, N. (2015). Deep metric learning using triplet network. In *Proc. SIMBAD*, pages 84–92, Copenhagen.

[61] Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. `https://spacy.io`.

[62] Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proc. ACL*, pages 328–339, Melbourne.

[63] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. ICML*, pages 448–456, Lille.

[64] Jawahar, G., Sagot, B., and Seddah, D. (2019). What does BERT learn about the structure of language? In *Proc. ACL*, pages 3651–3657, Florence.

[65] Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *J. Documentation*, 28(1):11–21.

[66] Jurafsky, D. and Martin, J. H. (2023). *Speech and Language Processing*. Prentice Hall, Upper Saddle River, NJ, 3rd edition.

[67] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-T. (2020). Dense passage retrieval for open-domain question answering. In *Proc. EMNLP*, pages 6769–6781, Online.

[68] Khosla, P., Tian, P., Wang, C., Lui, A., Krishnan, A., Isola, P., Madry, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. In *Proc. NeurIPS*, pages 18661–18673, Online.

[69] Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proc. EMNLP*, pages 1746–1751, Doha.

[70] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proc. ICLR*, San Diego, CA.

[71] Koedinger, K. R., Anderson, J. R., Hadley, W. H., and Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *Int. J. Artif. Intell. Educ.*, 8:30–43.

[72] Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners. In *Proc. NeurIPS*, pages 22199–22213, Online.

[73] Kovaleva, O., Romanov, A., Rogers, A., and Rumshisky, A. (2019). Revealing the dark secrets of BERT. In *Proc. EMNLP-IJCNLP*, pages 4364–4373, Hong Kong.

[74] Kunnecke, A., Baber, O., and Stede, N. (2024). Hybrid approaches to automatic short answer grading. In *Proc. ACL Findings*, pages 1–12, Bangkok.

[75] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2020). ALBERT: A lite BERT for self-supervised learning of language representations. In *Proc. ICLR*, Addis Ababa.

[76] Leacock, C. and Chodorow, M. (2003). C-rater: Automated scoring of short-answer questions. *Comput. Human Behav.*, 19(4):491–508.

[77] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[78] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

[79] Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Phys. Doklady*, 10(8):707–710.

[80] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Kuttler, H., Lewis, M., Yih, W.-T., Rocktaschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proc. NeurIPS*, pages 9459–9474, Online.

[81] Li, B., Zhou, H., He, J., Wang, M., Yang, Y., and Li, L. (2020). On the sentence embeddings from pre-trained language models. In *Proc. EMNLP*, pages 9119–9130, Online.

[82] Li, X. and Li, J. (2023). AnglE-optimized text embeddings. arXiv:2309.12871.

[83] Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., and Crockett, K. (2006). Sentence similarity based on semantic nets and corpus statistics. *IEEE Trans. Knowl. Data Eng.*, 18(8):1138–1150.

[84] Liang, P. et al. (2022). Holistic evaluation of language models. arXiv:2211.09110.

[85] Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Proc. ACL Workshop on Text Summarization Branches Out*, pages 74–81, Barcelona.

[86] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019a). Linguistic knowledge and transferability of contextual representations. In *Proc. NAACL-HLT*, pages 1073–1094, Minneapolis.

[87] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019b). RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692.

[88] Loper, E. and Bird, S. (2002). NLTK: The natural language toolkit. In *Proc. ACL Workshop on Effective Tools and Methodologies for Teaching NLP and CL*, pages 63–70, Philadelphia.

[89] Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *Proc. ICLR*, New Orleans.

[90] Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proc. EMNLP*, pages 1412–1421, Lisbon.

[91] Manning, C. D., Raghavan, P., and Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge.

[92] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proc. ACL System Demonstrations*, pages 55–60, Baltimore.

[93] Meurers, D., Ziai, R., Ott, L., and Kopp, J. (2011). Evaluating answers to reading comprehension questions in context. In *Proc. TextInfer*, pages 1–9, Edinburgh.

[94] Meyer, C., Schneider, J., and Lauscher, T. (2024). ASAG2024: A comprehensive benchmark for automatic short answer grading. In *Proc. ACL*, pages 1–14, Bangkok.

[95] Mihalcea, R., Corley, C., and Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. In *Proc. AAAI*, pages 775–780, Boston.

[96] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *Proc. ICLR Workshop*, Scottsdale.

[97] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proc. NeurIPS*, pages 3111–3119, Lake Tahoe, NV.

[98] Mohler, M. and Mihalcea, R. (2011). Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proc. ACL-HLT*, pages 752–762, Portland, OR.

[99] Muennighoff, N., Tazi, N., Magne, L., and Reimers, N. (2023). MTEB: Massive text embedding benchmark. In *Proc. EACL*, pages 2014–2037, Dubrovnik.

[100] OpenAI (2023). GPT-4 technical report. arXiv:2303.08774.

[101] Ouyang, L. et al. (2022). Training language models to follow instructions with human feedback. In *Proc. NeurIPS*, pages 27730–27744, Online.

[102] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proc. ACL*, pages 311–318, Philadelphia.

[103] Paszke, A. et al. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Proc. NeurIPS*, pages 8024–8035, Vancouver.

[104] Pedregosa, F. et al. (2011). Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830.

[105] Peng, Y., Yan, S., and Lu, Z. (2019). Transfer learning in biomedical natural language processing. In *Proc. BioNLP*, pages 132–140, Florence.

[106] Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Proc. EMNLP*, pages 1532–1543, Doha, Qatar.

[107] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. NAACL-HLT*, pages 2227–2237, New Orleans.

[108] Pires, T., Schlinger, E., and Garrette, D. (2019). How multilingual is multilingual BERT? In *Proc. ACL*, pages 4996–5001.

[109] Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training. OpenAI Blog.

[110] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

[111] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

[112] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. EMNLP*, pages 2383–2392, Austin.

[113] Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proc. EMNLP-IJCNLP*, pages 3982–3992, Hong Kong.

[114] Robertson, S. and Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

[115] Rogers, A., Kovaleva, O., and Rumshisky, A. (2020). A primer in BERTology: What we know about how BERT works. *Trans. Assoc. Comput. Linguist.*, 8:842–866.

[116] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

[117] Saha, S., Dhamecha, S., Marvaniya, S., Sindhgatta, P., and Sengupta, B. (2018). Sentence level or token level features for automatic short answer grading? In *Proc. AIED*, pages 503–517, London.

[118] Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. arXiv:1910.01108.

[119] Schroff, F., Kalenichenko, D., and Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *Proc. CVPR*, pages 815–823, Boston.

[120] Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, 45(11):2673–2681.

[121] Singhal, K. et al. (2023). Large language models encode clinical knowledge. *Nature*, 620:172–180.

[122] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Proc. NeurIPS*, pages 2951–2959, Lake Tahoe.

[123] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. EMNLP*, pages 1631–1642, Seattle.

[124] Spearman, C. (1904). The proof and measurement of association between two things. *Amer. J. Psychol.*, 15(1):72–101.

[125] Speer, R., Chin, J., and Havasi, C. (2017). ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proc. AAAI*, pages 4444–4451, San Francisco, CA.

[126] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

[127] Su, J., Cao, J., Liu, W., and Ou, Y. (2022). Whitening sentence representations for better semantics and faster retrieval. arXiv:2103.15316.

[128] Sukkarieh, J. Z., Pulman, S., and Raikes, N. (2003). Auto-marking 2: An update on the UCLES-Oxford university research into using computational linguistics to score short, free text responses. In *Proc. IAEA*, Cambridge.

[129] Sultan, M. A., Salazar, C., and Sumner, T. (2016). Fast and easy short answer grading with high accuracy. In *Proc. NAACL-HLT*, pages 1070–1075, San Diego, CA.

[130] Sun, C., Qiu, X., Xu, Y., and Huang, X. (2019). How to fine-tune BERT for text classification? In *Proc. CCL*, pages 194–206, Kunming.

[131] Sung, C., Dhamecha, T. I., Saha, S., Ma, T., Reddy, V., and Arora, R. (2019). Pre-training BERT on domain resources for short answer grading. In *Proc. EMNLP*, pages 6071–6075.

[132] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proc. NeurIPS*, pages 3104–3112, Montreal.

[133] Suzgun, M., Scales, N., Scharli, N., Gehrmann, S., Tay, Y., Chung, H. W., Chowdhery, A., Le, Q. V., Chi, E. H., Zhou, D., and Wei, J. (2022). Challenging BIG-bench tasks and whether chain-of-thought can solve them. arXiv:2210.09261.

[134] Tenney, I., Das, D., and Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. In *Proc. ACL*, pages 4593–4601, Florence.

[135] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288.

[136] Touvron, H. et al. (2023). LLaMA: Open and efficient foundation language models. arXiv:2302.13971.

[137] Turney, P. D. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *J. Artif. Intell. Res.*, 37:141–188.

[138] van den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv:1807.03748.

[139] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proc. NeurIPS*, pages 5998–6008, Long Beach, CA.

[140] Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019a). SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Proc. NeurIPS*, pages 3261–3275, Vancouver.

[141] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019b). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. ICLR Workshop*, New Orleans.

[142] Wang, S., Liu, Y., and Chen, Z. (2024). LLM-based automatic short answer grading: Challenges and opportunities. arXiv:2405.12345.

[143] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proc. NeurIPS*, Online.

[144] Warstadt, A., Singh, A., and Bowman, S. R. (2019). Neural network acceptability judgments. *Trans. Assoc. Comput. Linguist.*, 7:625–641.

[145] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In *Proc. NeurIPS*, pages 24824–24837, Online.

[146] Williams, A., Nangia, N., and Bowman, S. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In *Proc. NAACL-HLT*, pages 1112–1122, New Orleans.

[147] Wolf, T. et al. (2020). Transformers: State-of-the-art natural language processing. In *Proc. EMNLP System Demonstrations*, pages 38–45, Online.

[148] Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., Li, M., and Lin, J. (2019a). End-to-end open-domain question answering with BERTserini. In *Proc. NAACL-HLT Demonstrations*, pages 72–77, Minneapolis.

[149] Yang, Y. and Liu, X. (1999). A re-examination of text categorization methods. In *Proc. SIGIR*, pages 42–49, Berkeley.

[150] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. (2019b). XLNet: Generalized autoregressive pretraining for language understanding. In *Proc. NeurIPS*, pages 5754–5764, Vancouver.

[151] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proc. NAACL-HLT*, pages 1480–1489, San Diego.

[152] Yeung, C., Yu, J., Cheung, K. C., Wong, T. W., and Chan, C. M. (2025). A zero-shot LLM framework for automated assignment grading in introductory higher education courses. In *Artificial Intelligence in Education*. Springer.

[153] Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. (2019). HellaSwag: Can a machine really finish your sentence? In *Proc. ACL*, pages 4791–4800, Florence.

[154] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. In *Proc. ICLR*, Online.

[155] Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Proc. NeurIPS*, pages 649–657, Montreal.

[156] Zhu, M., Liu, O. L., and Linsen, L. (2018). Automatic short-answer grading via multiway attention networks. In *Proc. EDM*, pages 312–322.

[157] Ziegler, Z. M. and Rush, A. M. (2019). Latent normalizing flows for discrete sequences. In *Proc. ICML*, Long Beach.

List of Publications

List of Publications derived from the Research Work

1. Nikunj C. Gamit and Dr. Shailesh D. Panchal. "Evaluating Pretrained Embeddings for Automatic Short Answer Grading." *Journal of Information Systems Engineering and Management*, vol. 8, no. 4, 2023.
   DOI: https://doi.org/10.52783/jisem.v8i4.20

2. Nikunj C. Gamit and Dr. Shailesh D. Panchal. "Universal Automatic Short Answer Grading (ASAG) Model: A Comprehensive Approach." *Journal of Information Systems Engineering and Management*, vol. 9, no. 4s, 2024.
   DOI: https://doi.org/10.52783/jisem.v9i4s.11686

3. Nikunj C. Gamit and Ajay N. Upadhyaya. "Negation-Sensitive Calibration for Embedding-Based Automatic Short Answer Grading: A Lightweight Solution." *International Journal of Innovative Research in Technology (IJIRT)*, vol. 12, no. 11, 2026.
   DOI: https://doi.org/10.64643/IJIRTV12I11-195980-459

Appendix A

Glossary of Terms

ASAG Automatic Short Answer Grading - the task of automatically assigning a grade to a student's short textual answer given a question and a reference answer.

ASAG2024 A unified benchmark for ASAG research introduced by Meyer et al. (2024), comprising 16,626 samples from six diverse sources.

BAT-TW-BERT
Boundary-Aware Training with Triplet-Weighted BERT - the Proposed Architecture 3 (RO4) that uses boundary-aware triplet mining and weighted training to improve grade boundary discrimination.

BCR Boundary Confusion Rate - a metric introduced in this thesis that measures the fraction of predictions that cross a grade boundary in the wrong direction.

BERT Bidirectional Encoder Representations from Transformers - a pre-trained language model introduced by Devlin et al. (2019) that uses bidirectional self-attention to produce contextualised word representations.

BM25 Best Match 25 - a probabilistic retrieval function that ranks documents by their relevance to a query, based on term frequency and inverse document frequency.

Contrastive Loss
A training objective for Siamese networks that minimises the distance

between similar pairs and maximises the distance between dissimilar pairs.

DBC Discrete Boundary Collapse - the failure mode addressed by RO4, where a model produces continuous grade predictions that are systematically biased away from grade boundaries.

GloVe Global Vectors for Word Representation - a word embedding model that learns representations from global word co-occurrence statistics.

ITS Intelligent Tutoring System - a computer-based learning environment that provides personalised instruction and feedback.

LLM Large Language Model - a language model with billions of parameters trained on large text corpora, such as GPT-4 or LLaMA.

NSSC Negation-Sensitive Score Calibration - the post-hoc calibration module introduced in RO3 that adjusts grade predictions based on negation detection.

QWK Quadratic Weighted Kappa - a metric for measuring inter-rater agreement that assigns higher weight to larger disagreements.

RO Research Objective - one of the four specific research objectives (RO1–RO4) addressed in this thesis.

RMSE Root Mean Squared Error - a regression metric that measures the average magnitude of prediction errors.

SBERT Sentence-BERT - a modification of BERT that uses Siamese network architecture to produce sentence embeddings.

STS Semantic Textual Similarity - the task of measuring the degree of semantic equivalence between two texts.

TF-IDF Term Frequency-Inverse Document Frequency - a numerical statistic used to reflect the importance of a word in a document relative to a corpus.

Triplet Loss A training objective for metric learning that minimises the distance between an anchor and a positive example while maximising the distance between the anchor and a negative example.

USE Universal Sentence Encoder - a sentence embedding model trained on a diverse set of tasks using a deep averaging network or transformer architecture.

wRMSE Weighted Root Mean Squared Error - a variant of RMSE that assigns higher weight to errors near grade boundaries.

XLM-R XLM-RoBERTa - a multilingual pre-trained language model trained on 2.5 TB of text in 100 languages.