Towards Shorter and More Effective Feedback Cycles in Educational Settings

By

Shyam Agarwal

Signature Work, submitted for
completion of the University Honors Program

University Honors Program
University of California, Davis

APPROVED

Ali Moghimi, Ph.D.
Biological and Agricultural Engineering,
College of Agricultural and Environmental Sciences

Kate Andrup Stephensen, M.Ed.
Undergraduate Education,
Director of the University of California at Davis Honors Program

The thesis was updated to include recent work, after the original submission was approved, with due permission.

Shyam Agarwal. Towards Shorter and More Effective Feedback Cycles in Educational Settings. Thesis, University Honors Program, UC Davis.

ACKNOWLEDGEMENTS

This thesis is not only possible because of my commitment and passion towards making quality education accessible and feasible for all, but also due to the guidance, support and encouragement of many individuals to whom I am very grateful.

First of all, I would like to thank Dr. Ali Moghimi, my advisor, for providing me with the opportunity to work under his guidance. His invaluable advice, patient support, and always supporting remarks were crucial in my journey. Due to his mentorship, I have grown significantly, both in my technical abilities as well as my personal life.

I am also highly thankful to the University Honors Program committee at UC Davis for providing me with resources, connecting me with people, and providing an intellectually simulating environment that was crucial for my academic growth. Special thanks to Kate Andrup Stephensen, Taylor Cameron, and Pamela Pretell for helping me out with all the program logistics and for their kind guidance whenever requested.

My heartfelt appreciation goes to my mother, Tulika Agarwal, and my father, Ashish Agarwal. They have shaped who I am as a person, and their belief in my abilities always act as a constant source of encouragement. Their sacrifices and hard work are the reason for anything that I have been able to do and I will always be in debt to them.

To my sister, Aashika Agarwal, you have always been someone I look upto and I am thankful for your advice and moral upliftment whenever needed, even in times when I doubted myself. Thanks for always helping me identify opportunities and get clarity on what I am trying to do in life.

Lastly, I have met many exceptional people during the past three years, be it during my internships, research, club work, classes or elsewhere. My conversations with them have directly or indirectly impacted my work and me as a person. To all my professors, mentors, industry colleagues, lab colleagues, club members, friends, and everyone who has been part of my journey. Your support, whether big or small, has been invaluable.

Huge thanks to God!

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Chapter 1

AUTOMATED SHORT ANSWER GRADING

Agarwal, Shyam and Ali Moghimi (2024). "Grading with Near-Domain Data: Towards Human-level Accuracy for Automated Feedback on Short Response Questions". In: Under Review.

Grading with Near-Domain Data: Towards Human-level Accuracy for Automated Feedback on Short Response Questions

Shyam Agarwal[1], Ali Moghimi[2]

[1]Department of Computer Science, University of California, Davis, One Shields Ave, Davis, 95616, California, USA

[2]Department of Biological and Agricultural Engineering, University of California, Davis, One Shields Ave, Davis, 95616, California, USA

1.1 Abstract

Free-response questions are a crucial way to encourage generative processing and test a learner's understanding of core concepts. However, limited availability of instructor time, large class sizes, and other resource constraints pose significant challenges in providing timely and detailed assessment which is crucial for a holistic educational experience. Providing this assessment is a quite difficult problem since manual solutions are labor intensive, and automated solutions are difficult to generalize to scenarios where questions are open-ended. This paper proposes a novel and practical approach to grade short-answer responses to open-ended questions. We discuss the reasons why this problem is challenging, define the nature of questions on which our method works, and finally propose a framework that instructors can use to evaluate their classes. Our method outperforms the SOTA machine learning models as well as large language models like GPT 3.5, GPT 4 and GPT 4o by a huge margin of over 10-20% in some case, even after providing the LLMs with reference/model answer accross all three categories. Our framework does not require the use of pre-written grading rubrics, and is specifically designed keeping in mind the practical classroom settings. Our results also reveal exciting insights about learning from near-domain data and we believe this is the first work that formalizes the problem of automated short answer grading (eventually, towards automated feedback generation) based on the near-domain data.

1.2 Introduction and Related Works

Assessments play a very significant role in a student's learning outcomes and behaviors. On one hand, they provide valuable information to students about where they are lacking and what

improvements they can make. On the other hand, they can inform teachers about student progress and understanding of the various concepts that are being taught. Well-designed assessments can also promote the development of logical aptitude, problem-solving, critical thinking, and other high-order cognitive skills in students. Feedback from these assessments has a strong influence on their learning and achievement (Hattie and Timperley, 2007). In fact, many historical works have identified the importance of feedback in either improving knowledge and acquiring skills (Azevedo and Bernard, 1995; Bangert-Drowns et al., 1991; Corbett and Anderson, 1989; M. L. Epstein et al., 2002; Moreno, 2004; Pridemore and Klein, 1995) or an essential factor in motivating the learning process (Lepper and Chabay, 1985; Narciss and Huth, 2004).

The feedback from assessments can be categorized into two primary categories: formative feedback and summative feedback. Formative feedback can be understood as an ongoing process that provides valuable insights to the student and the instructor during the learning process. The aim of this feedback is to improve learning by modifying learner's thinking or behavior. Such feedback allows an opportunity to students to understand their limits and guides them towards directions that can better help them. In general, such formative feedback assessments are low stake or ungraded opportunities that show students where they currently are and how they can improve their knowledge. They also enable the instructors to understand how to modify their teaching methodology to specifically focus on student's learning gaps. In contrast, a summative feedback is an evaluation of the final product of student's learning. It rarely implies the need for further engagement with the assignment. Instead, it usually serves as an overall reflection of student's performance. Although feedback plays a very important role in one's learning, it takes a lot of time, effort, expertise, and resources to provide, especially when the formative feedback cycles need to be shorter and quicker.

Recently, there has been a lot of focus on making high-quality education as accessible to people as possible. The rate of increase in education costs per student is already higher than the rate of increase in economy-wide costs (Bowen, 2012). With an increasing demand for quality education, scaling feedback has been a widely accepted challenge, often referred as the "scaling feedback" problem. A major problem in providing the feedback is to essentially evaluate the student responses in the first place. Once we know which category the response falls in (correct or incorrect, for example), then it becomes easy to provide a template-based feedback or a more personalized feedback with some additional overhead work. This work, in particular, focuses on the former challenge of grading responses first.

It is important to note that the challenge in grading also depends on the type of question that is being considered. On one hand, objective questions like multiple-choices are easier to grade since the possible response choices are limited, but at the same time, these questions do not provide much insight into a student's learning journey. On the other hand, constructed-response questions (CRQ) are a great way to understand student thinking more deeply as they provide a powerful means of evaluating a learner's understanding of concepts as it requires an ability

to organize thoughts, provide explanations, and demonstrate higher-level thinking skills which further requires an ability to reason through elaborate answers (Zhao et al., 2021), but they are hard to grade.

We look at short answers, which are a type of CRQ that asks the learners to write their answers in a few sentences. The grading of these questions focuses on the content, instead of the writing quality and the possible correct answers are limited by the closed-form of the question in most cases (Burrows, Gurevych, and Stein, 2015) but sometimes there might be inconsistencies because of the subjective nature on both learner and evaluator's end. Although such free-response short answers are a powerful way of understanding a learner's thought processes, there is a huge barrier in providing quality and accurate evaluation/grading because of the wide range of possibilities in a shorter response length that is usually very straightforward and clear, with less direct information about the student's thought process or understanding. Providing accurate evaluation has emerged to be a hard problem and the main reasons for the difficulty of this problem and many other similar human-centered AI tasks are (M. Wu et al., 2018; Malik et al., 2021):

1. Student work shows a great amount of diversity, with many responses being unique (long tail / zipf distribution). Thus, statistical supervised grading techniques would find it hard to produce good results.
2. The manual labeling is a laborious task and only works when there is enough available data, which is not true in most practical settings.
3. Grading is a domain where precision is paramount due to the high stakes involved in misgrading.
4. Predictions must be transparent and justifiable to instructors and students alike.

Automated Short Answer Grading (ASAG) is a widely studied problem, and thus, has a strong literature background with people using many approaches, from traditional statistical ones to modern day neural networks. The choice of what approach to use mostly boils down to the availability of enough manually annotated dataset.

If enough data is available to train the models, then high performance can be obtained from just using supervised learning techniques or statistical models with n-gram features. For example, (Heilman and Nitin Madnani, 2013) took a domain adaptation approach with using word and character n-gram features to produce promising results. (Sultan, Salazar, and Sumner, 2016) employs supervised learning methods like linear regression and achieves high accuracy scores. (Hou and Tsao, 2011) used support vector machines and incorporated POS tags and term frequency to get good results. (N. Madnani et al., 2013) built a logistic regression classifier using simple features like count of commonly used words and the length of responses. These

methods are great when there is a large amount of available data, but in most practical settings, that is not the case. Instead, most instructors have access to near-domain data (data related to a similar question, but not the exact one) which is what we leverage in our work.

Many researchers have also used embedding-based neural networks. (Riordan et al., 2017) showed that standard neural architectures using conventional word embeddings, attention, and LSTM layers can produce good results on a number of varied datasets. The list goes on with (Y. Zhang, Shah, and Chi, 2016) using combination of deep belief networks and feature engineering, (X. Yang et al., 2018) using deep autoencoder models, (Liu et al., 2019) using multi-way attention networks, (Tan et al., 2020) encoding student responses using Graph Convolutional Networks and (Qi et al., 2019) using CNN and Bi-LSTM models to create hierarchical word-sentence model. Transformer architectures have also been very widely used by a number of researchers. In (Sung et al., 2019), human-comparable performance was reached by just fine-tuning BERT model on domain-specific task. (Gaddipati, Nair, and Plöger, 2020) evaluated different response embeddings for their ASAG task, namely ELMo, GPT, BERT, and GPT-2. The size of transformers and their ability to generalize to different languages has also been studied (Camus and Filighera, 2020). Unfortunately, availability of large amount of data is a concern here too which motivated us to take the approach that we will discuss in this paper.

Since ASAG and automated feedback generation are very close problems, many people have taken approaches beyond predicting just holistic scores. This line of researchers try to predict analytical scores and justification keys which are helpful for feedback generation too. (Mizumoto et al., 2019) manually annotated justification keys and trained a bidirectional LSTM to reduce the difference between these keys and system attention span. This is commendable because in resource-scarce domains, the joint learning can help improve the holistic score prediction results. In a similar spirit, (T. Wang et al., 2021) utilized a self-learning approach where the automatically detected justification keys were replaced with manual annotations too. The addition of annotations had a positive correlation with the results, but the impact reduced on adding more manually-scored data.

Another line of research tackles this problem in three steps. The first step involves the creation of reference answers based on a rubric or model answer. Then, conventional information retrieval methods are used to extract answer spans or justification keys from the student's answer. The last step utilizes any of the various semantic similarity measures to calculate similarity scores between student answer and reference answers (Haller et al., 2022). Most recently, people have started using large language models for these grading tasks too (Henkel et al., 2023; Schneider, Schenk, and Niklaus, 2024). Although LLMs are not trained for classification tasks, many works have shown promising results on these tasks by using few-shot prompting techniques by giving few examples in the system prompt(Cohn et al., 2024; Kortemeyer, 2023; Ouyang et al., 2022). Golchin and others try to replace peer grading in massive open online courses using OpenAI's GPT 3.5 and GPT 4 models (Golchin et al., 2024). Yoon uses one-shot prompting

and text similarity scoring (aligned with the three-step approach) (Yoon, 2023). Kortemeyer utilized the standard benchmark 2-way and 3-way datasets SciEntsBank and Beetle, and studied the performance of OpenAI's GPT-4 on them with and without a reference answer (Kortemeyer, 2023).

Keeping the above approaches in mind and the recent progresses in large language models like BERT and GPT, we show that the necessary algorithms to achieve the state-of-the-art performance already exist if utilized in a proper manner in the field of education. The rest of this paper discusses this novel approach and shares our insights gathered throughout the process. Our main contributions can be summarized as following:

1. We provide a novel framework to automate the grading of short constructed-response questions using near-domain data.
2. We compare the performance of high-compute (large number of parameters) models like GPT 3.5 Turbo, GPT 4 Turbo, and GPT 4o with the performance of low-compute (smaller number of parameters) models like BERT (and its fine-tuned versions).
3. We identify and analyze the patterns of mistakes that the models make.
4. We provide a discussion on model size and its impact on the environment towards the end.

1.3 Methodology

The Grading with near-domain data approach tries to learn information about the current set of question/problem involved by using data from different but related questions. We use this near-domain data and train basic BERT models to automatically classify the student responses as correct, incorrect, or incomplete. We also compare and contrast the performance of these models with non-fine-tuned and fine-tuned large language models from OpenAI's GPT series. The following sections explain our approach in detail.

1.3.1 Dataset

Our dataset consists of three related questions about the biological information flow (Central Dogma of Molecular Biology). In this section, we share some important insights about the dataset.

1.3.1.1 Significance of the underlying topic

The topic of information flow has been acknowledged as a fundamental biological concept by many researchers (American Association for the Advancement of Science, 2011; NGSS Lead States, 2013). However, both secondary as well as postsecondary level students have shown difficulty in understanding it (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016). As

Table 1.1: This table shares the question asked to the students, some examples of their responses, and the summative feedback output of our algorithm.

Question: Effect on Replication **Response:** Because this change would give the UAG codon this would signal to stop the replication, meaning the three codons toward the 3' side would not be read. This could significantly change the amino acids build or amino acid sequence. **Classification:** Incorrect
Question: Effect on Replication **Response:** DNA replication will not really be affected. DNA replication is not activated or deactivated by codons, so the DNA replication will continue as it normally would. **Classification:** Correct
Question: Effect on Transcription **Response:** Transcription of the entire strand will not take place. **Classification:** Incorrect
Question: Effect on Transcription **Response:** This alteration will not influence transcription because the DNA will still be transcribed into mRNA. The mRNA will have an A instead of a G at that location. **Classification:** Correct
Question: Effect on Translation **Response:** The stop codons are UAA, UAG, and UGA. None of these are in this strand of DNA even with the base change, so nothing will happen with the stop codon. Translation will decode the mRNA that is made by replicating the template strand, which will then produce polypeptides. **Classification:** Incorrect
Question: Effect on Translation **Response:** The alteration will be slightly detrimental in translation because the polypeptide chain will be very short and may not consist of all the amino acids needed to produce the right proteins needed by the body. In this sense, the body becomes deprived of some proteins, and different diseases associated with this deprivation occur as a result. **Classification:** Correct

highlighted by Lewis and others (Lewis, Leach, and Wood-Robinson, 2000) and Mills and others (Mills Shaw et al., 2008), the difficulty for many students lies in understanding how the genetic information is stored and exchanged. In contrast, many students find it difficult to understand how the genetic information (DNA) is encoded into proteins through a RNA intermediate. For example, Fisher highlights that many students falsely think that the amino acids used in translation are manufactured in the translation process (Fisher, 1985). The relationship between DNA and protein or between genes and proteins is also a hard concept to grasp for many (Mills Shaw et al., 2008; Wood-Robinson, 2000; Marbach-Ad, 2001). (M K Smith and J K Knight, 2012; M K Smith, Wood, and J K Knight, 2008) also reveal further misunderstandings about these relationships through students' ideas about impact of mutation on transcription and translation.

Table 1.2: Distribution of the Collected Data for Replication, Transcription, and Translation

Category	**Label**	**Training**	**Validation**	**Testing**
Replication	Correct	785	219	554
	Incomplete	240	64	169
	Incorrect	405	146	279
Transcription	Correct	759	217	554
	Incomplete	220	79	146
	Incorrect	451	133	302
Translation	Correct	940	272	665
	Incomplete	310	96	203
	Incorrect	180	61	134

Many researchers have tried to understand why this barrier in understanding this concept exists. Sometimes it is attributed to the usage of uninsightful diagrams to summarize the concept. For example, (Wright, Fisk, and Newman, 2014) highlights how the usage of the visual diagram (DNA -> RNA -> protein) might cause difficulties for those who do not understand what the arrows mean. For those who understand the processes represented by the arrows, the multiple steps involved in each process and how the processes relate to one another might be difficult to understand. (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016)

1.3.1.2 Origin of the Dataset

Genetics Concept Assessment is a concept inventory about genetics information flow and impacts of different mutations on the same (M K Smith, Wood, and J K Knight, 2008). It reveals understanding of nine learning goals through twenty-five multiple choice questions. Our work is focused on one of these goals: comparing different mutation types and describing their effect on genes, corresponding mRNAs and proteins, which is assessed by two questions that ask students to identify which mutation would result in a shortened RNA. For our work,

we use a version of the GCA multiple-choice question that was modified by Prevost and others in (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016). They modified the questions to create three open-ended prompts that asked students to explain the effect of a mutation that resulted in a premature stop codon on replication, transcription, and translation. These questions were administered to undergraduate biology majors in introductory cell and molecular biology-focused courses at two large public research universities as a part of their regular online homework assignments after the relevant content on central dogma was taught. The completion of the assignment was rewarded with one point and students were encouraged to just try their best. Initially, 1043 responses were collected where the student answered all three questions. After collecting some more responses (overall, data coming from multiple universities, class populations, and time periods), there were a total of 2861 responses which were marked as either correct, incomplete or incorrect by human experts with inter-grader agreement of 80%. Majority vote was accounted for when any disagreements happened. Some responses were removed from the data because they were nonsensical (like "IDK"). The data distribution is shown in table 2.1.

The question is as follows: *The following DNA sequence occurs near the middle of the coding region of a gene. DNA 5' A A T G A A T G G* G A G C C T G A A G G A 3' There is a G to A base change at the position marked with an asterisk. Consequently, a codon normally encoding an amino acid becomes a stop codon. How will this alteration influence DNA replication? How will this alteration influence DNA transcription? How will this alteration influence DNA translation?*

1.3.2 Terminology and Approach

We will use the following symbols to denote different aspects in our process. As described above, our dataset is composed of student responses to three different questions. We denote each student as S_1, S_2, ..., S_n. Q_1 refers to the first question which deals with the effect on replication, Q_2 refers to the second question which deals with the effect on transcription, and Q_3 refers to the third question which deals with the effect on translation. The responses from a given student S_k are written as $S_{k|1}$, $S_{k|2}$, $S_{k|3}$. From the data that we have, we reserve responses from 50% of the students for training (T_1', T_2', T_3' for Q_1, Q_2, and Q_3 respectively) and 15% of the students for validation purposes (V_1, V_2, V_3 for Q_1, Q_2, and Q_3 respectively) to ensure that a student's response to one question is not fed into our model to predict any characteristics about the same student's response to other questions. This helps maintain generality throughout the process. The responses from remaining 35% of students for Q_1, Q_2, and Q_3 are reserved for testing, hereby referred as T_1, T_2, T_3 respectively. This is to say that without loss of generality, given that we have S_1, S_2, ..., S_n students in some random order and let $p = 0.5n$ and $q = 0.15n$, then for o = 1, 2, 3 for some Q_o, we have

$$T'_o = \{S_{1|o}, S_{2|o}, \ldots, S_{p|o}\} \tag{1.1}$$
$$V_o = \{S_{p+1|o}, S_{p+2|o}, \ldots, S_{p+q|o}\} \tag{1.2}$$
$$T_o = \{S_{p+q+1|o}, S_{p+q+2|o}, \ldots, S_{n|o}\} \tag{1.3}$$

which gives us,

$$T'_1 = \{S_{1|1}, S_{2|1}, \ldots, S_{p|1}\} \tag{1.4}$$
$$T'_2 = \{S_{1|2}, S_{2|2}, \ldots, S_{p|2}\} \tag{1.5}$$
$$T'_3 = \{S_{1|3}, S_{2|3}, \ldots, S_{p|3}\} \tag{1.6}$$
$$V_1 = \{S_{p+1|1}, S_{p+2|1}, \ldots, S_{p+q|1}\} \tag{1.7}$$
$$V_2 = \{S_{p+1|2}, S_{p+2|2}, \ldots, S_{p+q|2}\} \tag{1.8}$$
$$V_3 = \{S_{p+1|3}, S_{p+2|3}, \ldots, S_{p+q|3}\} \tag{1.9}$$
$$T_1 = \{S_{p+q+1|1}, S_{p+q+2|1}, \ldots, S_{n|1}\} \tag{1.10}$$
$$T_2 = \{S_{p+q+1|2}, S_{p+q+2|2}, \ldots, S_{n|2}\} \tag{1.11}$$
$$T_3 = \{S_{p+q+1|3}, S_{p+q+2|3}, \ldots, S_{n|3}\} \tag{1.12}$$

We start with a basic BERT model (hereby, referred as base model BMQ_0), and fine tune it on different subsets of T_1 to get the best model BMQ_1, fine tune it on different subsets of T_2 to get the best model BMQ_2 , fine tune it on different subsets of T_3 to get the best model BMQ_3. We also fine tune BMQ_1 on different subsets of T_2 to obtain the best model PT_1Q_2, and fine tune BMQ_2 on different subsets of T_3 to obtain the best model, PT_2Q_3. These subsets were chosen by using a fixed random seed, and stratified sampling the particular training dataset in increments of 2.5% (from 0% to 100%). The definition of the "best" metric is described in 1.3.4 below. Further details about the process are explained in 1.3.3.1. For reference, refer to Figure 1.1 for easier understanding of the terminology.

Apart from BERT experiments, we also prompt different models in the GPT series to find how well they perform. We also fine-tune GPT 3.5 Turbo (the only model allowed to be fine-tuned at the time of the experiments), but for this fine-tuning, we use 100% of the training data. All these experiments are done using different temperatures (0, 0.5, and 1). Further details about the process are explained in 1.3.3.2.

1.3.3 Model Architecture and Training

Since the advent of BERT (Devlin et al., 2018) and GPT (Radford, Narasimhan, et al., 2018), the dominant adaptation technique has been model tuning or fine-tuning (Lester, Al-Rfou, and Constant, 2021) even for variants of GPT that came afterwards like (Radford, J. Wu, et al.,

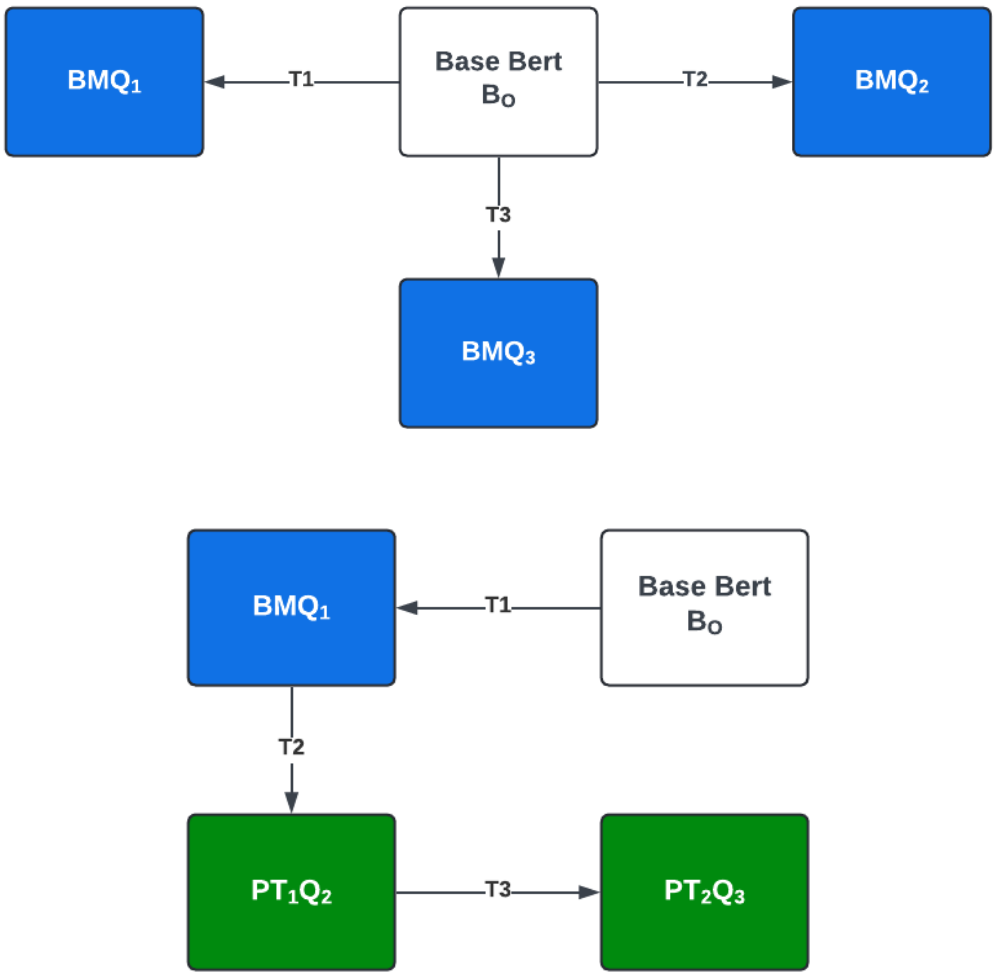

Figure 1.1: Model Name Terminology.

2019). The two most common ways of fine-tuning include freezing the weights in the earlier layers and only changing the ones in the later layers or retraining all the parameters in the model as proposed by (Howard and Ruder, 2018). We tried our experiments with both the approaches and found out that the former produced results that were suboptimal by a huge margin of 10-20% across all questions. Thus, the results shared in the rest of the paper are based on training all the parameters. Since all of our experiments are based on either fine-tuning BERT model or OpenAI's GPT series of models, we share the specific details around training these models below:

1.3.3.1 BERT

BERT (Bidirectional Encoder Representations from Transformers) is a language representation model based on the Transformer model (Vaswani et al., 2017) that is trained on a large corpus of unlabeled text. One particularly distinguishing characteristic of BERT is the Transformer

model's self-attention mechanism that compares the significance of every word in a piece of sentence to every other word in that sentence enabling it to collect context in both the forward and the backward direction. This bidirectional nature helps BERT capture and represent deeper and richer language representations than traditional unidirectional models.

. The B_0 model consists of input ids and attention masks mapping to the main layer, which comprises 12 layers and 12 attention heads with the pooler output having 768 nodes (totalling, 108310272 parameters). For BMQ_1, BMQ_2, and BMQ_3, we attach this main layer to an intermediate layer (IL_1) with 512 nodes (393728 parameters) which is attached to a classification head (CH_1) of 3 nodes (1539 parameters). This model consists of a total of 108705539 parameters, all of which are trainable. For PT_1Q_2, we remove the classification head from BMQ_1, attach the intermediate layer IL_1 to another intermediate layer (IL_2) with 256 nodes (131328 parameters) and then attach that to a new classification head (CH_2) with 3 nodes (771 parameters). In total, PT_1Q_2 has 108836099 parameters, all of which are trainable. In a similar manner, for PT_2Q_3, we remove the classification head CH_2 from PT_1Q_2, attach the intermediate layer IL_2 to another intermediate layer (IL_3) with 256 nodes (65792 parameters) and then attach that to another new classification head (CH_3) with 3 nodes (771 parameters). In total, PT_2Q_3 has 108901891 parameters, all of which are trainable.

For training these models, we employ the Adam Optimizer (Kingma and Ba, 2014) with a learning rate of $1e-5$, and use categorical cross entropy as the loss function. The number of epochs are guided by an early stopping callback function that monitors the validation accuracy for 10 successive epochs without an improvement before it restores the weights of the model performing the best on validation data.

1.3.3.2 GPT Series

For GPT fine-tuning, we follow the standard procedure as suggested by OpenAI. For the prompt that is used to train the GPT models, we take inspiration from (Golchin et al., 2024) to decide our root prompt which is as follows:

You are a fair and knowledgeable instructor whose task is to evaluate the student's assignments in accordance with the correct answers to each of the questions that are presented in the section that follows.
The student does not need to share every information from the model answer category as long as they convey the right idea.

Then, we provide the question, the model answer, the student response, and ask the model to classify it as either correct, incorrect, or incomplete as a multiple-choice question as suggested and done in (Santurkar et al., 2023). The rest of the prompt is given below, where <substring> represents the answer provided by the student and <sample_correct>, <sample_incomplete>, and <sample_incorrect> refer to the sample correct, sample incomplete, and sample incorrect

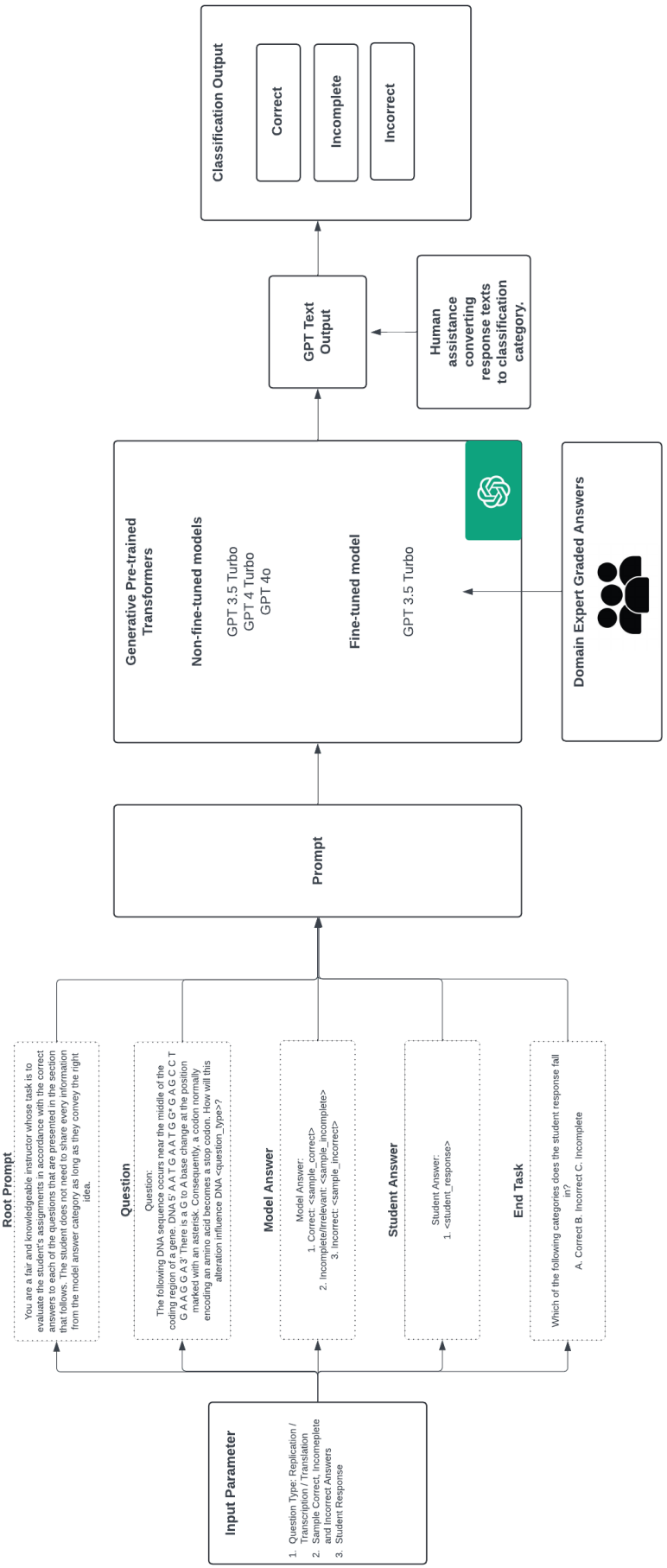

Figure 1.2: GPT Model Prompting Cycle

responses for each <question_type> (replication, transcription, translation). The table 1.3 shares the exact samples, which is inspired from (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016)

Question:
The following DNA sequence occurs near the middle of the coding region of a gene.
DNA 5' A A T G A A T G G G A G C C T G A A G G A 3'*
There is a G to A base change at the position marked with an asterisk. Consequently, a codon normally encoding an amino acid becomes a stop codon.
How will this alteration influence DNA <question_type>?
Model Answer:
1. Correct: <sample_correct>
2. Incomplete/Irrelevant: <sample_incomplete>
3. Incorrect: <sample_incorrect>.
Student Answer:
1. <substring>
Which of the following categories does the student response fall in?
A. Correct
B. Incorrect
C. Incomplete

These experiments were performed with three different temperatures (0, 0.5, 1) which is an OpenAI parameter that determines the creativity of the model. The usual range is from 0 to 2 with higher values making the output more random, while lower values making it more focused and deterministic. When the temperature is set to 0, the model uses log probability to automatically increase the temperature until certain thresholds are hit. We also performed experiments with higher temperatures like 1.5 and 2.0, but the output from the model was gibberish in most cases so we do not report any results from these experiments.

1.3.4 Best Metric

We need to weigh two factors while picking the best model - the metric of evaluation (chosen as accuracy here), and the amount of data fed into the model training. We train the models using increments of 2.5% of the training data up until 100% of the training data. Then, we find the top 5 most accurate models, calculate their average accuracy, and choose the most accurate model with the least amount of data that is within 1 standard deviation of this average accuracy. Note that BMQ_1, BMQ_2, BMQ_3, PT_1Q_2, PT_2Q_3 are the models chosen using the best metric.

1.4 Experiments and Results

We share the main experiments and their key results in this section. For details of all other results, please refer to Appendix A.

Table 1.3: Examples of student responses and rubric coding, with each response shown given by a different student.

Question	<sample_correct>	<sample_incomplete>	<sample_incorrect>
Replication	It will not have any effect on the replication process.	This would be an example of a nonsense mutation.	The DNA will stop replicating when it reaches the stop codon.
Transcription	It will not have any effect on transcription.	This will cause a mutation in the transcription process.	In the process of transcribing DNA into RNA, the newly added stop codon will inhibit the rest of the chain from being transcribed into RNA.
Translation	This will influence translation because the stop codon will cause the amino acid sequence to end before it should. This will create a different polypeptide or protein that will either not function or function differently than it should have.	The code will be translated with a different base and will be read differently. This will result in a different protein being built.	This will have no influence on translation.

1.4.1 Base BERT on Q_1, Q_2, Q_3

Starting from Base BERT B_0, and fine-tuning on Q_1, Q_2, and Q_3, we get BMQ_1, BMQ_2, and BMQ_3 which have an accuracy of 92.81%, 91.52%, and 83.83% respectively. The amount of training data used to achieve these accuracy is 30.0%, 12.5% and 27.5% respectively. The rest of the metrics can be found in table **??**.

1.4.2 Pretrained model results on Q_1, Q_2, Q_3

As a reminder, models BMQ_i and $PT_{i-1}Q_i$ differ from each other in the method used to train them. Model BMQ_i is trained directly on Base BERT using the question i whereas model PT_1Q_2 is trained on BMQ_1 and model PT_2Q_3 is trained on BMQ_2. For Question 2, PT_1Q_2 achieves an accuracy of 91.32% after training on 62.5% of the data. We see that the model accuracy for PT_1Q_2 started at a higher percentage than the model accuracy for BMQ_2, but they both stabilize around accuracy with negligible point differences later on. This suggests that near-domain data from other questions is helpful in improving accuracy on a particular question. For Question 3, PT_2Q_3 achieves an accuracy of 86.93% after training on 82.5% of the data. This time, both PT_2Q_3 and BMQ_3 start around same accuracy percentages but PT_2Q_3 has a smaller data threshold than BMQ_3 which again suggests that PT_2Q_3 was able to learn insights

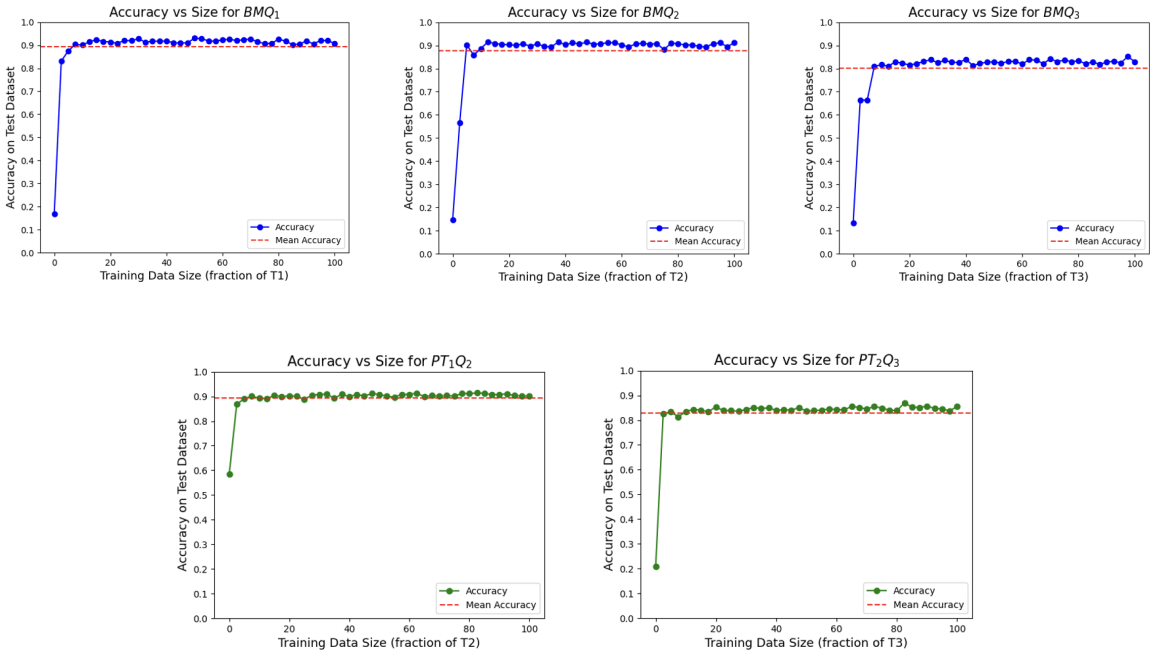

Figure 1.3: Comparison of training data percentage and model accuracy for different quantization methods: (a) BMQ_i and (b) $PT_{i-1}Q_i$

from Q_1 and Q_2 that helped grade Q_3. The rest of the metrics can be found in table **??**.

1.4.3 Prompting GPT Results on Q_1, Q_2, Q_3

We use the same prompt as described in section 1.3.3.2 to prompt GPT 3.5 Turbo, GPT 4 Turbo (gpt-4-0125-preview endpoint), and GPT 4o. We perform these experiments with different temperatures as described in section **??**. On Q_1 with temperature 0, GPT 3.5 Turbo achieves an accuracy of 69.46%, GPT 4 Turbo achieves an accuracy of 74.75%, and GPT 4o achieves an accuracy of 77.15%. On Q_1 with temperature 0.5, GPT 3.5 Turbo achieves an accuracy of 68.76%, GPT 4 Turbo achieves an accuracy of 75.25%, and GPT 4o achieves an accuracy of 76.95%. On Q_1 with temperature 1, GPT 3.5 Turbo achieves an accuracy of 66.37%, GPT 4 Turbo achieves an accuracy of 75.05%, and GPT 4o achieves an accuracy of 74.65%. On Q_2 with temperature 0, GPT 3.5 Turbo achieves an accuracy of 64.77%, GPT 4 Turbo achieves an accuracy of 73.05%, and GPT 4o achieves an accuracy of 67.56%. On Q_2 with temperature 0.5, GPT 3.5 Turbo achieves an accuracy of 63.37%, GPT 4 Turbo achieves an accuracy of 73.25%, and GPT 4o achieves an accuracy of 65.77%. On Q_2 with temperature 1, GPT 3.5 Turbo achieves an accuracy of 63.97%, GPT 4 Turbo achieves an accuracy of 73.05%, and GPT 4o achieves an accuracy of 64.17%. On Q_3 with temperature 0, GPT 3.5 Turbo achieves an accuracy of 68.56%, GPT 4 Turbo achieves an accuracy of 77.84%, and GPT 4o achieves an accuracy of 67.56%. On Q_3 with temperature 0.5, GPT 3.5 Turbo achieves an accuracy of 68.46%, GPT 4 Turbo achieves an accuracy of 77.45%, and GPT 4o achieves an accuracy of 67.37%. On Q_3 with temperature 1, GPT 3.5 Turbo achieves an accuracy of 66.97%, GPT 4

Turbo achieves an accuracy of 78.04%, and GPT 4o achieves an accuracy of 68.26%. The rest of the metrics can be found in table **??**.

1.4.4 Fine-tuned GPT Results on Q_1, Q_2, Q_3

Using the same prompt and multiple temperatures as described in section 1.3.3.2, we fine-tune GPT 3.5 Turbo (the only model that OpenAI allowed to fine-tune at the time these experiments were performed). On Q_1, we get an accuracy of 93.71%, 93.81%, 93.41% with temperatures 0, 0.5, and 1 respectively. On Q_2, we get an accuracy of 93.91% with all temperatures 0, 0.5, and 1. On Q_3, we get an accuracy of 88.02%, 87.62%, 87.92% with temperatures 0, 0.5, and 1 respectively. The rest of the metrics can be found in table **??**.

Table 1.4: This table shares some metrics that are useful in evaluating the performance of the different models on their respective questions. We also share the results from LLMs for a more comprehensive comparison. All the metrics used are macro versions. N/A means not-applicable, indicating that no training happened in the experimental setting.

Model Name	**Training Data %age**	**Accuracy**	**Precision**	**Recall**	**F1**
BMQ_1	30.0	92.81	89.73	91.33	90.43
gpt-3.5-turbo (Q_1, fine-tuned, temp=0)	100	93.71	91.51	91.00	91.20
gpt-3.5-turbo (Q_1, fine-tuned, temp=0.5)	100	93.81	91.69	91.06	91.32
gpt-3.5-turbo (Q_1, fine-tuned, temp=1)	100	93.41	91.10	90.54	90.76
gpt-3.5-turbo (Q_1, non-fine-tuned, temp=0)	N/A	69.46	65.77	68.14	64.36
gpt-3.5-turbo (Q_1, non-fine-tuned, temp=0.5)	N/A	68.76	65.34	67.47	63.69
gpt-3.5-turbo (Q_1, non-fine-tuned, temp=1)	N/A	66.37	63.67	65.39	61.73
gpt-4-0125-preview (Q_1, non-fine-tuned, temp=0)	N/A	74.75	71.24	65.45	61.71
gpt-4-0125-preview (Q_1, non-fine-tuned, temp=0.5)	N/A	75.25	70.94	66.03	62.47
gpt-4-0125-preview (Q_1, non-fine-tuned, temp=1)	N/A	75.05	69.56	65.49	61.65

(continued)

Model Name	Training Data %age	Accuracy	Precision	Recall	F1
gpt-4o (Q_1, non-fine-tuned, temp=0)	N/A	77.15	70.99	73.96	71.03
gpt-4o (Q_1, non-fine-tuned, temp=0.5)	N/A	76.95	71.23	74.41	71.34
gpt-4o (Q_1, non-fine-tuned, temp=1)	N/A	74.65	69.17	72.20	68.87
BMQ_2	12.5	91.52	87.81	86.85	87.26
PT_1Q_2	62.5	91.32	88.14	86.09	86.87
gpt-3.5-turbo (Q_2, fine-tuned, temp=0)	100	93.91	90.43	92.95	91.53
gpt-3.5-turbo (Q_2, fine-tuned, temp=0.5)	100	93.91	90.43	92.95	91.53
gpt-3.5-turbo (Q_2, fine-tuned, temp=1)	100	93.91	90.43	92.95	91.53
gpt-3.5-turbo (Q_2, non-fine-tuned, temp=0)	N/A	64.77	58.07	58.39	57.28
gpt-3.5-turbo (Q_2, non-fine-tuned, temp=0.5)	N/A	63.37	56.86	56.64	55.71
gpt-3.5-turbo (Q_2, non-fine-tuned, temp=1)	N/A	63.97	57.08	57.63	56.44
gpt-4-0125-preview (Q_2, non-fine-tuned, temp=0)	N/A	73.05	58.46	60.38	55.24
gpt-4-0125-preview (Q_2, non-fine-tuned, temp=0.5)	N/A	73.25	59.16	60.62	55.67
gpt-4-0125-preview (Q_2, non-fine-tuned, temp=1)	N/A	73.05	58.77	60.50	55.56
gpt-4o (Q_2, non-fine-tuned, temp=0)	N/A	67.56	64.99	65.71	62.50
gpt-4o (Q_2, non-fine-tuned, temp=0.5)	N/A	65.77	64.17	63.76	60.93
gpt-4o (Q_2, non-fine-tuned, temp=1)	N/A	64.17	62.21	61.12	58.85
BMQ_3	27.5	83.83	74.88	69.23	70.87
PT_2Q_3	82.5	86.93	79.91	74.65	76.70

(continued)

Model Name	Training Data %age	Accuracy	Precision	Recall	F1
gpt-3.5-turbo (Q_3, fine-tuned, temp=0)	100	88.02	82.60	75.80	78.22
gpt-3.5-turbo (Q_3, fine-tuned, temp=0.5)	100	87.62	81.83	74.97	77.39
gpt-3.5-turbo (Q_3, fine-tuned, temp=1)	100	87.92	82.37	75.55	77.94
gpt-3.5-turbo (Q_3, non-fine-tuned, temp=0)	N/A	68.56	61.82	56.89	56.99
gpt-3.5-turbo (Q_3, non-fine-tuned, temp=0.5)	N/A	68.46	61.94	57.04	57.15
gpt-3.5-turbo (Q_3, non-fine-tuned, temp=1)	N/A	66.97	55.61	55.32	54.80
gpt-4-0125-preview (Q_3, non-fine-tuned, temp=0)	N/A	77.84	66.09	66.12	66.10
gpt-4-0125-preview (Q_3, non-fine-tuned, temp=0.5)	N/A	77.45	65.57	65.12	65.33
gpt-4-0125-preview (Q_3, non-fine-tuned, temp=1)	N/A	78.04	66.99	66.87	66.92
gpt-4o (Q_3, non-fine-tuned, temp=0)	N/A	67.56	68.58	62.71	60.66
gpt-4o (Q_3, non-fine-tuned, temp=0.5)	N/A	67.37	67.20	62.30	60.05
gpt-4o (Q_3, non-fine-tuned, temp=1)	N/A	68.26	66.32	61.49	59.87

1.5 Discussion

1.5.1 Sources of Error in Model Results

In total, BMQ_1 had 70, BMQ_1 had 85, BMQ_3 had 162, PT_1Q_2 had 87, and PT_2Q_3 had 131 human-machine disagreements. With help from domain experts, we studied the different errors in the model predictions and classified them into three categories that are useful for further research in this field.

1. ***Response Construction and Structure***: These responses were not classified correctly by any of the BMQ_i or $PT_{i-1}Q_i$ models. In total, there are 62 such Q_2 responses and 102 such Q_3 responses (number of samples for each combination is shown in 1.5). If we include

the results from GPT models too, there are 35 responses that are not classified correctly by any model. We attribute this behavior to the general zipf distribution of responses which has been noted by many works (Nguyen et al., 2014; Piech, 2016). A distribution is said to follow zipf's law if the probability of the n-th most probable outcome follows an inverse power-law with respect to n. This is to say that the probability of an outcome o is proportional to:

$$p(o) \propto \frac{1}{\text{rank}(o)^{\beta}} \tag{1.13}$$

where rank(o) is calculated after ordering all the outcomes based on their probabilities, and β is the exponent parameter guiding the shape of the distribution. Many useful inferences can be made from identifying the zipf nature of distributions, but the most relevant one is identifying that zipf distributions have a long tail which implies that a huge fraction of the responses will only be observed exactly once and hence, it is difficult to generalize algorithms/models to them.

Table 1.5: Response Construction Errors: Number of Samples Where Both Models Predicted Wrong Output (for Q_2 and Q_3)

Question	BMQ_i	$PT_{i-1}Q_i$	**Correct**	**Total**
Transcription	0	0	1	12
	0	0	2	3
	0	1	2	0
	0	2	1	3
	1	0	2	2
	1	1	0	14
	1	1	2	5
	1	2	0	0
	2	0	1	1
	2	1	0	1
	2	2	0	4
	2	2	1	17
Translation	0	0	1	17
	0	0	2	26
	0	1	2	2
	0	2	1	5
	1	0	2	5
	1	1	0	7
	1	1	2	27
	1	2	0	1
	2	0	1	2
	2	1	0	1
	2	2	0	2
	2	2	1	7

2. ***Model-agnostic errors***: These are the responses that were classified incorrectly by one model, but were classified correctly by the other (for Q_2 and Q_3 obviously). Table 1.6 lists number of samples in this category. Based on domain expert evaluation, there were some themes that were noticed across all model

Table 1.6: Model-agnostic Errors: Number of Samples Where One Model Predicted Wrong Output Whereas the Other Model Predicted Correct Output (for Q_2 and Q_3)

Question	BMQ_i	$PT_{i-1}Q_i$	**Correct**	**Total**
Transcription	0	1	0	3
	0	1	1	5
	0	2	0	5
	0	2	2	2
	1	0	1	8
	1	0	0	5
	1	2	1	7
	1	2	2	4
	2	0	2	1
	2	0	0	2
	2	1	2	1
	2	1	1	5
Translation	0	1	0	3
	0	1	1	14
	0	2	0	2
	0	2	2	6
	1	0	1	5
	1	0	0	7
	1	2	1	12
	1	2	2	17
	2	0	2	4
	2	0	0	7
	2	1	2	3
	2	1	1	9

3. ***Human Expert Errors***: While evaluating the results given by the models, we ran them via domain experts who realized that many of the test responses were marked incorrectly by the human experts in the first place. This is quite exciting because this suggests that the model is able to learn well despite the presence of potential noise in the data. Overall , BMQ_1 had 10, BMQ_1 had 8, BMQ_3 had 17, PT_1Q_2 had 12, and PT_2Q_3 had 11 possible human misscores. This is summarized in table 1.7.

1.5.2 Model BMQ_i vs Model $PT_{i-1}Q_i$

As can be seen in 1.3, for Q_2 and Q_3, the model fine-tuned on Base BERT and the model fine-tuned on the particular pre-trained model achieve similar accuracy (with negligible point

Table 1.7: Human Expert Errors: Number of Samples Counted as Misclassified by the Models but Misunderstood Due to Original Human Miscodes

Question	Model	Human-Machine Disagreements	Possible Human Miscodes
Replication	BMQ_1	70	10
Transcription	BMQ_2	85	8
	PT_1Q_2	87	12
Translation	BMQ_3	162	17
	PT_2Q_3	131	11

differences) when trained on large amounts of data. However, there are two important insights to note here.

For Q_2, BMQ_2 achieves an accuracy of merely 14.77% and 56.59% with 2.5% and 5.0% of T_2 respectively, but for PT_1Q_2, the accuracy is 58.38% and 86.83% with 2.5% and 5.0% of T_2 instead. This suggests that instructors in academic settings can achieve greater automated grading accuracy with same small amount of data if they have access to near-domain data. We define this as **accuracy advantage**.

For Q_3, the amount of data needed to achieve mean accuracy (defined previously as data threshold) is way smaller for PT_2Q_3 model than for BMQ_3 model. Precisely, BMQ_3 achieves the mean accuracy or more when trained on 10% or more of T_3 whereas PT_1Q_2 achieves the mean accuracy or more when trained on 2.5% or more of T_3. This suggests that instructors in academic settings can achieve same automated grading accuracy with smaller amount of data if they have access to near-domain data. We define this as **data advantage**.

This is useful because in practical classroom settings, instructors might not have the resources to manually grade responses to a question to help with the automation of the rest of the grading. In such a case, they can use the data from previously asked questions about the same topic to compensate for the unavailability of data for the current question set. All they would need to do is to grade a handful of questions from the current question set.

Notice that the data advantage is observed in Q_2 as well, but to a very small extent. The accuracy advantage is also observed in Q_3, but only after we feed 5% of T_3. According to us, this is because unlike Q_1 and Q_2, where the correct answer is that there will be no effect on replication and transcription respectively, the correct answer for Q_3 is that there would be an effect on translation. Since the PT_2Q_3 model does not have any context about the change of question or change of correct answer, it took some more data for it to adapt itself and for us to see the accuracy advantage.

Table 1.8: Model Performance Comparison Across Three Questions Calculated from Accuracy in Table 1.4

Question	Model Name	Mean Accuracy	Sample Standard Deviation	1 SD Range
Replication	GPT 3.5 Turbo (fine-tuned)	93.643	0.208	93.435 - 93.851
	GPT 3.5 Turbo (non-fine-tuned)	68.197	1.620	66.577 - 69.817
	GPT 4 Turbo (non-fine-tuned)	75.017	0.252	74.765 - 75.269
	GPT 4o (non-fine-tuned)	76.250	1.389	74.861 - 77.639
Transcription	GPT 3.5 Turbo (fine-tuned)	93.910	0.000	93.910 - 93.910
	GPT 3.5 Turbo (non-fine-tuned)	64.037	0.702	63.335 - 64.739
	GPT 4 Turbo (non-fine-tuned)	73.117	0.115	73.002 - 73.232
	GPT 4o (non-fine-tuned)	65.833	1.695	64.138 - 67.528
Translation	GPT 3.5 Turbo (fine-tuned)	87.853	0.208	87.645 - 88.061
	GPT 3.5 Turbo (non-fine-tuned)	67.997	0.891	67.106 - 68.888
	GPT 4 Turbo (non-fine-tuned)	77.777	0.300	77.477 - 78.077
	GPT 4o (non-fine-tuned)	67.730	0.469	67.261 - 68.199

1.5.3 Variability across non-fine-tuned GPT results

For each of the GPT model experiments, the results obtained from using different temperatures are very close to each other (some metrics presented in **??**). Thus, we can consider the different accuracy from changing temperatures to be insignificant and thus, we feel comfortable comparing results for each model using the mean accuracy. Across all the three questions, GPT 3.5 Turbo does not perform very well with a gap of 5% - 10% from the best performing model in each category. This gap is maximum for Q_3 (10.047 %), lesser for Q_2 (9.08 %), and the least for Q_1 (8.053 %). Comparing GPT 4 Turbo and GPT 4o, the former performs way better than the latter for Q_2 and Q_3, surpassing the accuracy by 7.284 % and 10.047 % on those questions respectively. However, GPT 4o beats GPT 4 Turbo on Q_1 by a small margin of 1.233%. It is interesting to note that for Q_3, GPT 3.5 Turbo also surpasses the performance of GPT 4o, although by merely 0.267 points.

We do not have a good reasoning on why some architectures perform better on some questions than others. Just like Shazeer mentions in (Shazeer, 2020), we attribute their success, as all else, to divine benevolence.

1.5.4 Fine-tuned vs Non-Fine-Tuned GPT 3.5 Turbo

From table 1.8, we can clearly see that fine-tuning GPT 3.5 Turbo performs way better than its non-fine-tuned counterpart with a margin of 25.446%, 29.873%, and 19.856% on Q_1, Q_2, and Q_3 respectively. This suggests that although GPT models are trained on vast amounts of data, feeding information particular to the question under consideration helps them better understand the patterns.

1.5.5 Model Size and Environment

From our results, it is clear that the best performing models are the fine-tuned versions of BERT and GPT. Fine-tuned GPT 3.5 Turbo surpasses the fine-tuned BERT model (the poorer performing one of the BMQ_i or $PT_{i-1}Q_i$ where applicable) by 0.83%, 2.59%, and 4.02% on Q_1, Q_2, and Q_3 respectively. However, the GPT models are quite huge, consist of a lot more parameters (billions) and require a lot of time to train. Fine-tuning and training these large models also requires a lot of time, energy, and computation that releases a lot of carbon dioxide gas which is harmful for the environment (Agarwal and Chakraborti, 2024). In light of the growing climate-related concerns, this tradeoff in accuracy seems quite reasonable and suggests that training of small models for automated grading tasks is better than utilizing foundational and large language models like GPT.

1.6 Conclusion

We conclude that providing automated feedback at scale is a hard problem for a variety of reasons that we discussed in this paper. However, fine-tuning BERT model using near-domain data achieves great results, especially compared to non-fine-tuned GPT models. Although fine-

tuning GPT offers the best solution, it is not feasible in the light of the negative impact that it has on the environment. We want to end this paper by acknowledging that this paper suffers from a few limitations. Firstly, the inter-grader agreement on the dataset used is only 80%. Secondly, the results in this paper are based on a particular question about a very particular concept (Central Dogma) from molecular biology. Thus, it is hard to generalize our results to other questions in general. Last but not the least, we did not have access to fine-tune other GPT models, which could have added lot more insights to the work. We strongly believe that our work would provide significant direction for other future work and that the summative feedback produced by our algorithm can be integrated into other mechanisms to be translated into formative feedback.

1.7 Acknowledgement

We want to thank Professor Kevin C Haudek from Michigan State University for providing us with the data, domain expertise and feedback whenever needed. This work would have not been possible without his generous support.

Chapter 2

AUTOMATED FEEDBACK GENERATION

Agarwal, Shyam and Ali Moghimi (2024). ""Oh, so that's what went wrong!": Targeted Feedback through Concept and Category Aware Prompting in Educational Settings". In: Planned Submission.

Oh, so that's what went wrong!": Targeted Feedback through Concept and Category Aware Prompting in Educational Settings

Shyam Agarwal[1], Ali Moghimi[2]

[1]Department of Computer Science, University of California, Davis, One Shields Ave, Davis, 95616, California, USA

[2]Department of Biological and Agricultural Engineering, University of California, Davis, One Shields Ave, Davis, 95616, California, USA

2.1 Abstract

Providing timely and detailed feedback to student assessments is crucial for a holistic educational experience. However, reviewing assessments and providing feedback to assessment takers is a challenging problem because of the limited availability of instructor time, large class sizes, and other resource constraints. This problem is difficult since manual solutions are labor intensive, and automated solutions are difficult to generalize to scenarios where questions are open-ended. Unlike objective questions like multiple-choice questions or fill-in-the-blanks, free-response short-answer questions help better understand a learner's capability as they encourage generative processing and test a learner's understanding of core concepts. Thus, it is important to develop systems that can provide automated feedback on such open-ended questions. This paper proposes a novel and practical approach to providing feedback on short-answer responses to open-ended questions by utilizing our concurrent work to classify student responses as either correct, incorrect, or incomplete using near-domain data and then uses semantic clustering with human assistance to analyze the key concept of student responses. These pieces of information are then fed into a generative pre-trained model to generate targeted and personalized feedback for the students. Our automated grading method outperforms the SOTA machine learning models as well as some large language models in providing summative feedback by a huge margin of over 10-20%. Our feedback generation framework extends on this leverage, does not require the use of any pre-written grading rubrics, and is specifically designed keeping in mind the practical classroom settings. We believe that this is the first work that undertakes a semantic clustering approach along with utilizing a large language model and grade scores

based on near-domain data to provide automated feedback.

2.2 Introduction and Related Works

Feedback plays a crucial role in a student's learning journey and overall growth. As defined by Ambrose and others, feedback is "information given to students about their performance that guides future behavior" (Nyquist and Jubran, 2012). As per Boud and Molly, feedback is "the process whereby learners obtain information about their work in order to appreciate the similarities and differences between the appropriate standards for any given work, and the qualities of the work itself, in order to generate improved work". In simple words, feedback is any information that can inform an individual about their relative performance and can guide them towards strategies to improve themselves.

Feedback is said to be formative when it aims to modify the thinking or behavior of the learner with the aim of improving their learning. These feedbacks are usually given as part of short learning cycles where a student is supposed to understand the feedback and either redo the assignment or build on top of it. Constructive feedback can not only have a positive impact on student's comprehension of concepts, but it also encourages the development of critical thinking abilities and fosters metacognitive skills (Hattie and Timperley, 2007). Timely, specific, non-evaluative, supportive and meaningful feedback is very strongly related to increased learner motivation and improved academic performances (Shute, 2008). Many researches support these impacts on improving knowledge and acquiring skills (Azevedo and Bernard, 1995; Bangert-Drowns et al., 1991; Corbett and Anderson, 1989; M. L. Epstein et al., 2002; Moreno, 2004; Pridemore and Klein, 1995) or on acting as a motivating factor in the learning process (Lepper and Chabay, 1985; Narciss and Huth, 2004).

Given that feedback plays such a crucial role in a student's learning process, the ability to provide quality feedback is quite important. A good feedback must provide specific information to improve the gap between what is understood by a student and what is the ground truth (Sadler, 1989). In recent times, the focus has shifted on providing more personalized feedback that is tailored to the specific needs, styles, pace, and knowledge gap of the student. Generic feedback might be helpful in some cases, but it often fails to guide students appropriately because in most cases, it is no better than providing student with a score or the source text from a book. In contrast, personalization increases the applicability and relevance of feedback, making it more impactful for learner growth. For example, feedback targeted towards student's misconception reinforces self-directed learning (Nicol and Macfarlane-Dick, 2006) that can encourage self-regulation and self-confidence. Similarly, when feedback is combined with correctional review, the feedback and instruction are intertwined such that "the process itself takes on the forms of new instruction, rather than informing the student solely about correctness" (Kulhavy, 1977).

The scope of this personalization depends on the type of question. Questions like multiple-choice, one word or fill in the blank provide little to no room to understand student's thought

process and learning. In contrast, essay-style questions or constructed-response questions (CRQ) provide more room for both, the learners and the instructors to engage in deeper cognitive processes. They require students to articulate their understandings into their own words which offers valuable insights into their thought processes and misconceptions, especially because there is more room for creativity and critical thinking (Zhao et al., 2021). They also allow instructors more opportunity to understand how to change their teaching style, and make it more targeted towards their audience. Out of all CRQs, short-response ones are quite interesting. They do not constraint the student thinking with pre-defined answers, but at the same time, owing to their shorter length, they do not provide much opportunity to students to explain in detail, their understanding. At the same time, the possible correct answers are usually limited because of the closed-form nature of questions (Burrows, Gurevych, and Stein, 2015) as opposed to opinion-based long-form essays.

In practical scenarios, achieving the above mentioned quality feedback is quite difficult because it is a laborious and time-consuming task, especially when the instructor to student ratio is small and when the formative feedback cycles need to be shorter in length. This problem of being able to scale quality feedback to a wide audience is often referred as the "scaling feedback" problem. It is challenging because (M. Wu et al., 2018; Malik et al., 2021):

1. Student work displays significant diversity in most cases and is usually characterized by a distribution that has a long tail aka the majority of solutions are infrequent.
2. Supervised techniques are not very feasible because the process of labeling student work with detailed feedback is both arduous and costly.
3. Generating wrong feedback is not acceptable because it could mean wrong learning outcomes for the student.
4. The feedback produced must be easily understood, and also explainable to the student and the instructor.

Since the response evaluation is highly subjective, mainly depending on the evaluator's interpretation of student response, the end result can be inconsistent feedback. Moreover, students might also misinterpret the question on their end and provide response to something that was not asked in the question. The problem worsens in short-response questions because of shorter response length.

Different people have approached the problem of feedback generation from different perspectives. Some have used supervised learning methods like regression analysis and have achieved high accuracies (Sultan, Salazar, and Sumner, 2016), but this approach requires availability of huge amounts of data depending on the specific application and hence, cannot be used widely.

Others have used techniques where humans can intervene and design rubrics which can be used for automated grading by a rubric-focused learning algorithm (Basu, Jacobs, and Vanderwende,

2013; Süzen et al., 2020; Lan et al., 2015). There have been numerous works that rely on these grading rubrics in the form of a model answer, and measure the syntactic and semantic similarities between the two of them (Leacock and Chodorow, 2003). These works differ in the way they define similarity with some simply using word matching (Selvi and Bnerjee, 2010), some using knowledge and corpus based methods (Mohler and Mihalcea, 2009), some using semantic vector embeddings (Saha et al., 2018), and some using semantic similarity and dependency graph alignments (Mohler, Bunescu, and Mihalcea, 2011). Some recent works have also tried targeted feedback generation at key point level. Zhao and others extract key points from a response and match them to a model answer to identify the missing key points that are used to generate feedback based on a template (Zhao et al., 2021). However, depending on the domain, it might not be possible to come up with a model answer or there might be a lot of possible model answers.

Another set of researchers employ the use of probabilistic programs to write generative descriptions of student cognition and synthesize millions of labeled examples of student responses which are then used to infer feedback on real student solutions (Malik et al., 2021). However, it is neither easy nor generalizable to model student cognition using probabilistic programs for all domains of questions. Wu and colleagues also propose a human-in-the-loop rubric sampling approach that is zero-shot in nature (M. Wu et al., 2018). However, the results would depend widely on the complexity of the task and their method also suffers from the need to involve instructors to mimic student thinking which is not very feasible.

The personalization aspect has also been studied well, especially in many works by Linn and her students. They investigate how automated guidance can support short answer responses from middle school students. They have compared automated feedback alone and in combination with information about personalized nature of feedback (Tansomboon et al., 2017). They have also compared feedback alone and in combination with students providing feedback on sample essay (Gerard, Linn, and Madhok, 2016) and in combination with a revision process modeling interface (Gerard and Linn, 2022). In all these cases, they used C-rater-ML tool from ETS (Heilman and Nitin Madnani, 2013) which depends on model/reference answer or grading rubrics and provides mainly templated feedback.

More recently, people have employed large language models (LLMs) to generate these feedbacks. Aggarwal and others generate Engineering Short Answer Feedback (EngSAF) dataset where they generate synthetic feedback using LLMs like Gemini and ChatGPT (Aggarwal, Bhattacharyya, and Raman, 2024). Scarlatos and others use GPT-4 to annotate human and LLM-generated feedback along with proposing a reinforcement-learning based framework to optimize the correctness and alignment of the feedback (Scarlatos et al., 2024). Meyer and others compared the effectiveness of LLM-generated feedback to no feedback and highlighted that LLM-generated feedback has a positive correlation to students' cognitive and affective-motivational outcomes (Meyer et al., 2024).

In this work, we try to utilize large language models (LLMs) along with the traditional non-supervised technique of semantic clustering in an attempt to produce targeted feedback on a dataset of short-response questions about the biological concept of central dogma. We build on top of the approach of grading with near-domain data that Agarwal and Moghimi present using the same dataset(Agarwal and Moghimi, 2024). Our key contributions are as follows:

1. We present a novel framework to produce automated feedback for short-response questions.
2. We compare the performance of the results from this technique with standard baseline prompts.
3. We present a new technique of prompting, Concept and Category Aware Prompting (C2AP).

2.3 Methods

In this section, we will explain the dataset that we are using, the terminology involved as well as the different steps involved in our Targeted Feedback through Concept and Category Aware Prompting (TF-C2AP) framework. Overall, we identify the category (correct, incomplete, or incorrect), the concept that each response relates to, and feed that into a large language model like GPT-4 using Concept and Category Aware Prompting (C2AP).

Table 2.1: Distribution of the Collected Data for Replication, Transcription, and Translation

Category	**Label**	**Training**	**Validation**	**Testing**
Replication	Correct	785	219	554
	Incomplete	240	64	169
	Incorrect	405	146	279
Transcription	Correct	759	217	554
	Incomplete	220	79	146
	Incorrect	451	133	302
Translation	Correct	940	272	665
	Incomplete	310	96	203
	Incorrect	180	61	134

2.3.1 Dataset

Information flow is considered as a fundamental concept in biology (American Association for the Advancement of Science, 2011; NGSS Lead States, 2013), but research suggests that it is usually tough for students to grasp (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016). Some potential reasons that have been identified are: difficulty in understanding the storage and exchange of information (Lewis, Leach, and Wood-Robinson, 2000; Mills Shaw et al., 2008), challenges with understanding how DNA is encoded into protein through RNA intermediate

like misinterpreting that the amino acids used in the translation process are created during translation (Fisher, 1985), and confusion between DNA, gene, and protein (Mills Shaw et al., 2008; Wood-Robinson, 2000; Marbach-Ad, 2001). Smith and others find out further issues by studying students' interpretation on the impact on transcription and translation (M K Smith and J K Knight, 2012; M K Smith, Wood, and J K Knight, 2008). Several researchers have also tried to understand why this is such a difficult topic to understand, and many owe this to lack of usage of insightful visual diagrams in classes (Wright, Fisk, and Newman, 2014) or to multiple complex steps involved in each step. (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016)

The dataset used for this work comes from GCA which is a concept inventory about information flow (M K Smith, Wood, and J K Knight, 2008) involving nine learning goals through 25 multiple choice questions. Two of these questions evaluate the understanding about comparing different mutation types and describing the effects on on genes, corresponding mRNAs and proteins. Prevost and others modified this into a constructed short-answer question (Prevost, Michelle K. Smith, and Jennifer K. Knight, 2016) that was later used by Agarwal and Moghimi (Agarwal and Moghimi, 2024). The dataset used consisted of three open-ended prompts asking students to explain how a mutation that resulted in a premature stop codon affected replication, transcription, and translation. A total of 1043 responses were collected from undergraduate biology majors from two large public research universities. These students were taught the relevant central dogma concepts in their introductory cell and molecular biology-focused courses. The questions were admininstered as a part of their online homework assignment, and students were rewarded one point for completion so they were asked to give their best shot. Later, more responses were collected, bringing the total to 2861. The collected data represented multiple institutions, class populations and time periods. This was then filtered to remove some nonsensical responses (like "IDK") and the filtered data was marked as either correct, incomplete, or incorrect by human experts who had an inter-grader agreement of 80%. In cases of disagreement, the majority vote was taken into consideration. The data distribution is shown in 2.1.

This is the question: *The following DNA sequence occurs near the middle of the coding region of a gene. DNA 5' A A T G A A T G G* G A G C C T G A A G G A 3' There is a G to A base change at the position marked with an asterisk. Consequently, a codon normally encoding an amino acid becomes a stop codon. How will this alteration influence DNA replication? How will this alteration influence DNA transcription? How will this alteration influence DNA translation?*

2.3.2 Terminology and Approach

Our dataset consists of responses from students to three different questions. Let's denote each student as S_1, S_2, ..., S_n. Q_1 refers to the effect on replication, Q_2 refers to the effect on transcription, and Q_3 refers to the effect on translation. For a student S_k, the responses are referred as $S_{k|1}$, $S_{k|2}$, $S_{k|3}$. We reserve responses from 50% students for training (T'_1, T'_2, T'_3 for

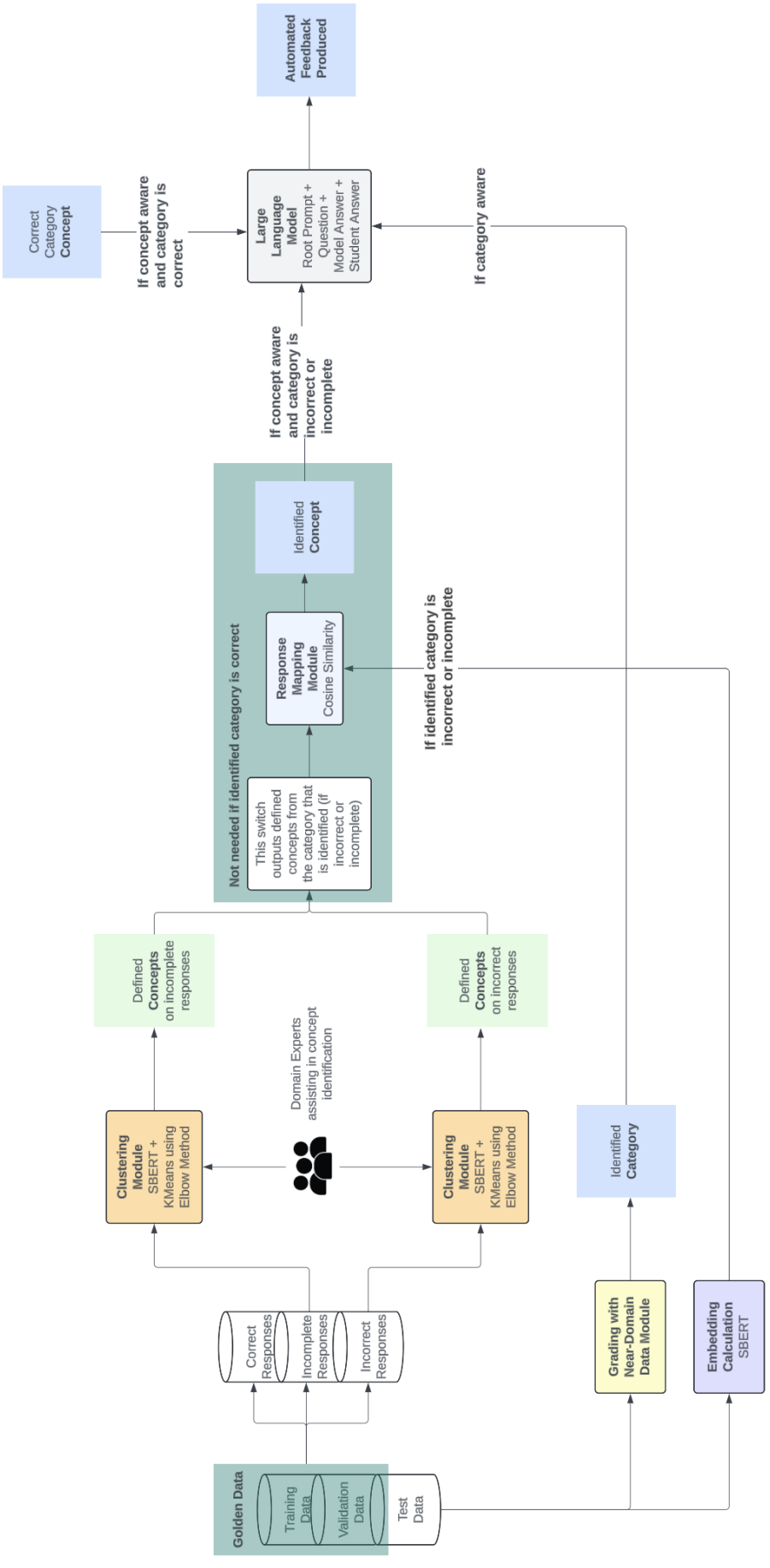

Figure 2.1: Approach for Targeted Feedback Through Concept and Category-Aware Prompting and its Variantions

Table 2.2: Examples of student responses and rubric coding, with each response shown given by a different student.

Question	<sample_correct>	<sample_incomplete>	<sample_incorrect>
Replication	It will not have any effect on the replication process.	This would be an example of a nonsense mutation.	The DNA will stop replicating when it reaches the stop codon.
Transcription	It will not have any effect on transcription.	This will cause a mutation in the transcription process.	In the process of transcribing DNA into RNA, the newly added stop codon will inhibit the rest of the chain from being transcribed into RNA.
Translation	This will influence translation because the stop codon will cause the amino acid sequence to end before it should. This will create a different polypeptide or protein that will either not function or function differently than it should have.	The code will be translated with a different base and will be read differently. This will result in a different protein being built.	This will have no influence on translation.

Q_1, Q_2, and Q_3 respectively), responses from 15% students for validation (V_1, V_2, V_3 for Q_1, Q_2, and Q_3 respectively), and the responses from the remaining 35% students for testing (T_1, T_2, T_3 for Q_1, Q_2, and Q_3 respectively).

This is done for the category identification using grading with near-domain data, and it is in alignment with what Agarwal and Moghimi (Agarwal and Moghimi, 2024) do, using the exact same partitions from their work. For the concept identification and other parts, we do not need a separate validation and training dataset, so we combine them into a golden dataset which consists of the remaining 50% responses that were not used for testing (hereby, referred as G_1, G_2, G_3 for Q_1, Q_2, and Q_3 respectively). We further create three subsets of G_i: $G_{i|correct}$, $G_{i|incomplete}$, and $G_{i|incorrect}$ which consists of the responses in G_i that were marked correct, incomplete and incorrect respectively in the golden dataset (i = 0, 1, 2 for Q_1, Q_2, and Q_3 respectively). In more mathematical terms, we get that without loss of generality, given that we have S_1, S_2, . . . , S_n students in some random order and let $p = 0.5n$ and $q = 0.15n$, then for o = 1, 2, 3 for some Q_o, we have

$$T'_o = \{S_{1|o}, S_{2|o}, \ldots, S_{p|o}\} \tag{2.1}$$
$$V_o = \{S_{p+1|o}, S_{p+2|o}, \ldots, S_{p+q|o}\} \tag{2.2}$$
$$T_o = \{S_{p+q+1|o}, S_{p+q+2|o}, \ldots, S_{n|o}\} \tag{2.3}$$

which gives us,

$$T'_1 = \{S_{1|1}, S_{2|1}, \ldots, S_{p|1}\} \tag{2.4}$$
$$T'_2 = \{S_{1|2}, S_{2|2}, \ldots, S_{p|2}\} \tag{2.5}$$
$$T'_3 = \{S_{1|3}, S_{2|3}, \ldots, S_{p|3}\} \tag{2.6}$$
$$V_1 = \{S_{p+1|1}, S_{p+2|1}, \ldots, S_{p+q|1}\} \tag{2.7}$$
$$V_2 = \{S_{p+1|2}, S_{p+2|2}, \ldots, S_{p+q|2}\} \tag{2.8}$$
$$V_3 = \{S_{p+1|3}, S_{p+2|3}, \ldots, S_{p+q|3}\} \tag{2.9}$$
$$T_1 = \{S_{p+q+1|1}, S_{p+q+2|1}, \ldots, S_{n|1}\} \tag{2.10}$$
$$T_2 = \{S_{p+q+1|2}, S_{p+q+2|2}, \ldots, S_{n|2}\} \tag{2.11}$$
$$T_3 = \{S_{p+q+1|3}, S_{p+q+2|3}, \ldots, S_{n|3}\} \tag{2.12}$$
$$\tag{2.13}$$

For o as above, we also have

$$G_o = T'_o \cup V_o \tag{2.14}$$

which gives us,

$$G_1 = \{S_{1|1}, S_{p+q+2|1}, \ldots, S_{p+q|1}\} \tag{2.15}$$
$$G_2 = \{S_{1|2}, S_{p+q+2|2}, \ldots, S_{p+q|2}\} \tag{2.16}$$
$$G_3 = \{S_{1|3}, S_{p+q+2|3}, \ldots, S_{p+q|3}\} \tag{2.17}$$

Please note that,

$$G_{o|correct} = \{x \in G_o \mid \text{x's category is labeled correct}\} \tag{2.18}$$
$$G_{o|incomplete} = \{x \in G_o \mid \text{x's category is labeled incomplete}\} \tag{2.19}$$
$$G_{o|incorrect} = \{x \in G_o \mid \text{x's category is labeled incorrect}\} \tag{2.20}$$

It is important to mention that technically, we do not need to reserve all responses (Q_1, Q_2, Q_3) from a student in either T_i, T'_i, V_i or G_i (i as above) for rest of the framework. This is to say that each of the questions could have been considered separately (for instance, for a student S_m, $S_{m|1}$

could have been in T_1, $S_{m|2}$ could have been in V_2, and $S_{m|3}$ could have been in T'_3, which means G_1 would have $S_{m|1}$, G_2 would have $S_{m|2}$, and G_3 would not have $S_{m|3}$) since the framework is separate for each of the three processes (replication, transcription, and translation). However, given that we use the grading with near-domain data approach in this framework which utilizes data from similar questions to grade a particular question, the authors decided to keep the sets separately to ensure that a student's response to one question is not used to calculate their response to another question as practical academic settings would abide by it. Thus, we decide to keep all responses from a student in either of the sets for the rest of the framework too, and instead of T_p and V_p, we use G_p (for p = 1, 2, 3 for Q_1, Q_2, and Q_3 respectively). In a more generalized setting where grades are calculated without depending on near-domain data, the rest of the framework can still be used with any response being in either of T_p or G_p (p as above).

Overall, there are 5 steps in this process.

1. We first use semantic clustering to group the incorrect and incomplete training and validation data separately. This is to say that we cluster $G_1|incorrect$, $G_1|incomplete$, $G_2|incorrect$, $G_2|incomplete$, $G_3|incorrect$, and $G_3|incomplete$ separately.
2. We ask human experts to look at some responses in each cluster to define common themes/concepts.
3. Then, using Agarwal and Moghimi's grading with near-domain data approach (Agarwal and Moghimi, 2024), we produce category-labels (correct, incomplete, or incorrect) for test responses.
4. Based on the category, we use cosine similarity to map the test responses to the human-expert defined concept from clustering.
5. The extracted concept and category are then fed into GPT-4o along with other pieces of information to create the finalized feedback.

Also, we find the need for clustering correct responses as unnecessary because to generate feedback, the reference concept from the model answer can be used instead. The approach is also graphically represented in figure 2.1.

2.3.3 Semantic Clustering

In this section, we share some common challenges in clustering short-response texts as well as how we cluster the responses as a part of the framework.

2.3.3.1 Challenges with clustering short-response texts

Clustering is the process of grouping together entities into multiple groups (one group is trivial) such that entities in one group are similar to each other in some aspect, and farther from entities in

the other groups in that aspect. In our case, the entities are the student responses (vectorized into embeddings as explained later) and this aspect is defined as the semantic similarity between the responses because we want responses with similar themes/misunderstandings/gap to be clustered together.

Owing to the short length of short-answer text responses, clustering them is a more challenging task, especially because traditional similarity measures (like cosine similarity) rely heavily on co-occurrence of terms between the responses. This means that even if responses are topically related, but they do not have similar words, they are more likely not going to be clustered together. This problem is more relevant in scenarios where vocabulary size is huge and text length is short, like in tweets (Pinto, Benedí, and Rosso, 2007). We realize that this is not an issue with our dataset, because owing to the narrow focus of the question topic (Central dogma), the vocabulary size is limited and thus, many keywords are repeatedly used by students as shown in SOME FIGURE. Many other works have also shown promising results with clustering short-texts using similarity measures like cosine similarity (Kang, Lerman, and Plangprasopchok, 2010; Ni et al., 2011; H. Yang, Mysore, and Wallace, 2009)

Some other challenges with clustering short texts include lack of context (Metzler, Dumais, and Meek, 2007) since it is hard to interpret the meaning of certain words/phrases without proper context. In that case, responses might have similar words but different meanings. We realize that this is not an issue because the responses given by learners share the same context as the question asked to them. Also, although students come from different backgrounds, class populations, universities and time periods, since they all were exposed to similar class content, would have solved similar problems, and would have read similar material, they actually share a lot more context than we would expect, which is also highlighted in another work on clustering student responses (Hämäläinen et al., 2018). Thus, we do not think this to be a major problem with our data.

2.3.3.2 Clustering the G_i datasets

We first convert the train text responses into numerical vector embeddings using Sentence-BERT (SBERT) model (Reimers and Gurevych, 2019). We choose SBERT over other conventional embeddings (like Word2Vec (Mikolov et al., 2013), GloVe (Pennington, Socher, and Manning, 2014), BERT (Devlin et al., 2018) etc.) because the latter focus on generating embeddings based on individual words. They produce token-level embeddings which are then aggregated using sum, average or [CLS] token embedding. In contrast, SBERT generates fixed-size sentence embeddings by fine-tuning BERT on sentence-pair tasks. Thus, it is able to capture the rich semantic relationships at a sentence-level which is quite significant for short text responses. In particular, we used the *paraphrase-MiniLM-L6-v2* model from Hugging Face's Sentence Transformer library because it is a standard model that is pre-trained for sentence similarity tasks. This model is inspired from Microsoft's MiniLM work (W. Wang et al., 2020). It

is trained using a masked language modeling objective on large text corpra, followed by a siamese network setup with cosine similarity as the loss function to fine-tune it on paraphrase identification task (identifying whether two texts convey the same meaning even if they have different words). It produces dense 384-dimensional vector representations.

Then, we use KMeans clustering algorithm to partition the embeddings into a predefined number of clusters (k). KMeans works by iteratively assigning data points to the nearest cluster centroid followed by recalculating the centroids based on each cluster at the given step. In particular, we used Lloyd's algorithm which uses euclidean distance to find the nearest cluster and calculates the centroids for each cluster using mean. We used Scikit-learn tool to perform this clustering. The assignment and update steps are shown below.

Assignment Step:

$$c_i = \arg \min_{j \in \{1,2,\ldots,k\}} \|\mathbf{x}_i - \mu_j\|^2$$

Update Step:

$$\mu_j = \frac{1}{|C_j|} \sum_{\mathbf{x}_i \in C_j} \mathbf{x}_i$$

The value of k was determined using the Elbow method which is used to find the ideal number of clusters such that the compactness of clusters as well as the separation between different clusters is balaced. It involves plotting the Within-Cluster Sum of Squares (WCSS) for different values of k and selecting the point where the WCSS curve flattens (like an "elbow"). The WCSS curves and the chosen number of clusters are present in figure 2.2.

We cluster the following sets separately: $G_1|incorrect, G_1|incomplete, G_2|incorrect, G_2|incomplete$, $G_3|incorrect$, and $G_3|incomplete$. There are two reasons for this. Firstly, clustering different questions separately ensures the generality and makes sure that responses from a particular question do not impact the clustering of other responses. Secondly, our understanding is that since incomplete responses were hard for human experts to identify as either correct or incorrect, there presence during the clustering process could bias the clustering of responses that were clearly identified as incorrect.

We do not cluster the correct responses because it is an unnecessary step. When the result is correct, we can always use the reference concept (from model answer) to produce the feedback instead, utilizing the power of the grading technique. This is especially true because the grading depends on content instead of the writing quality, and in most cases, the possible correct answers are limited by the closed-form of the question (Burrows, Gurevych, and Stein, 2015) as shared earlier, although there can be several reasons for it being incorrect.

2.3.4 Human Defined Concepts

The clustered responses were then studied by a human expert manually who identified overarching themes in the responses. The identified themes were ran through two more subject experts

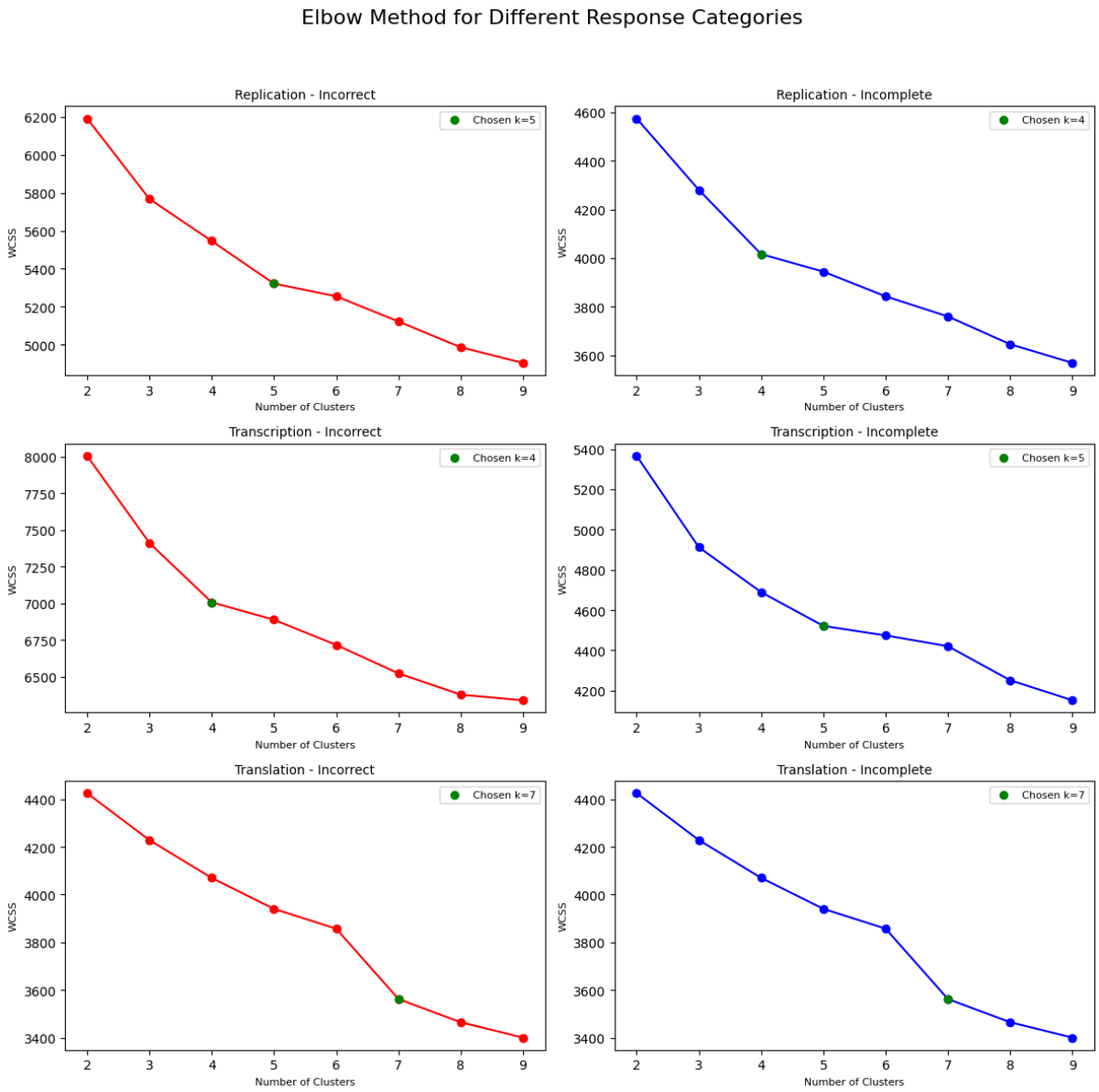

Figure 2.2: The Within-Cluster Sum of Squares (WCSS) is plotted against the number of clusters to visualize the diminishing gains in clustering quality as k increases. Green markers indicate the selected k values for each category: 5 and 4 for replication, 4 and 5 for transcription, and 7 and 7 for translation.

later. All the experts were undergraduates majors in biology and related fields (namely, Genetics, Biotechnology, and Biomedical Engineering) and were in their senior year at University of California, Davis. The identified concepts and their detailed description can be found in tables 2.3 for incorrect replication, 2.4 for incorrect transcription, 2.5 and 2.6 for incorrect translation, 2.7 for incomplete replication, 2.8 for incomplete transcription and 2.9 and 2.10 for incomplete translation.

Cluster	Explanation
0	The answer is incorrect because a premature stop codon doesn't halt or generate a partial strand of DNA. It remains unaffected by the stop codon and replicates the complementary strand irrespective of the mutation. However, if the mutation is related to a single or double-stranded break in the DNA, then it might generate an incomplete replicate.
1	The answers are incorrect because the DNA strand is unaffected by the stop codon. The process of replication won't pause due to the mutation because the stop codon doesn't have a function in replication that affects the DNA. Thus, no pause or shortening of DNA strands will occur. The stop codon will be replicated as usual as in other replications.
2	The answer is incorrect because the process of DNA replication won't stop and hence won't be affected due to stop codon introduction, it will still carry out the replication process normally. Thus, the process of replication will still occur irrespective of the premature stop codon.
3	The answers are incorrect because DNA replication is just focused on producing a complementary strand irrespective of the presence of the STOP codon. The role of the stop codon comes into play during the process of translation, where the process of translation will stop upon encountering a premature stop codon, and generate a shorter stretch of amino acids. Thus, the process of translation terminates early due to the stop codon and doesn't impact DNA replication.
4	The answers are incorrect because the addition of a premature stop codon does not affect the DNA strand or the RNA transcript. A premature stop codon only affects an amino acid during translation. A shorter amino acid will be produced due to the premature stop codon.

Table 2.3: Cluster Explanations for Incorrect Replication Responses

2.3.5 Category identification via Grading with Near-Domain Data

In Grading with near-domain data (Agarwal and Moghimi, 2024), Agarwal and Moghimi show that since in most practical academic situations, the availability of large amount of manually-labeled data is less, one can grade short-response questions by fine-tuning the model on data coming from similar domain. For Q_2 and Q_3, they used very small number of responses from that particular question and still achieved similar or better accuracies by utilizing data from all previous questions. They also evaluate the performance of base BERT and large language models (fine-tuned and non-fine-tuned) on this task and share some other insights as well, but we mainly use the categories identified from their base model on Q_1, and previous question

Cluster	Explanation
0	The answer is incorrect because, despite a mutation that adds a stop codon, the process of transcription of DNA to RNA will still occur without any premature stop in the process. The stop codon doesn't affect the process of transcription. It just stops prematurely only while translating and produces a shorter amino-acid sequence.
1	The case that a stop codon prevents proper transcription of DNA to RNA is wrong because, like replication, the process of transcription is also unaffected by the presence of a stop codon. An RNA will be transcribed from DNA without any pauses or stops and isn't affected by its presence. It only affects the level of translation because the codons are read by the tRNA from the mRNA at this point.
2	The answer that shorter mRNA strand is incorrect because the stop codon, or any other codon mutation, doesn't affect the mRNA sequence that is produced as a result of transcription. Transcription is simply converting the DNA sequence into an RNA format that can be processed into an amino acid. Thus, it's unaffected by the stop codon, and no reduction in the transcript length occurs.
3	The answer is incorrect because the stop codon doesn't have an effect on either of the processes of replication or transcription and hence won't affect the DNA or the RNA strand. Both the strands will be replicated and transcribed completely irrespective of the stop codon. Thus, the DNA will be fully replicated and will produce a fully transcribed mRNA irrespective of the stop codon.

Table 2.4: Cluster Explanations for Incorrect Transcription Responses

pre-trained model for Q_2 and Q_3 to identify the category for our responses in test datasets (T_1, T_2, and T_3).This is done to simulate the academic settings where neither much data would be available to train from scratch, nor much computational resources to fine-tune large language models. Anyway, the performances are all very similar across all three settings with data, time, and computational trade offs. For Q_1 though, there is no previous question pre-trained model so we have to use the fine-tuned base BERT model.

2.3.6 Concept identification via Response mapping

Cosine similarity is a popular metric used to measure similarity between two vectors in multidimensional space. It does so by calculating the angles between the vectors to capture the orientation instead of the magnitude. For a test response, once we know the category, we calculate the vector embeddings using the same procedure as explained in 2.3.3.2 and find the closest cluster by calculating the cosine similarity with the centroid (mean of embeddings of all responses in a cluster). The human-identified concept for that particular cluster is then utilized.

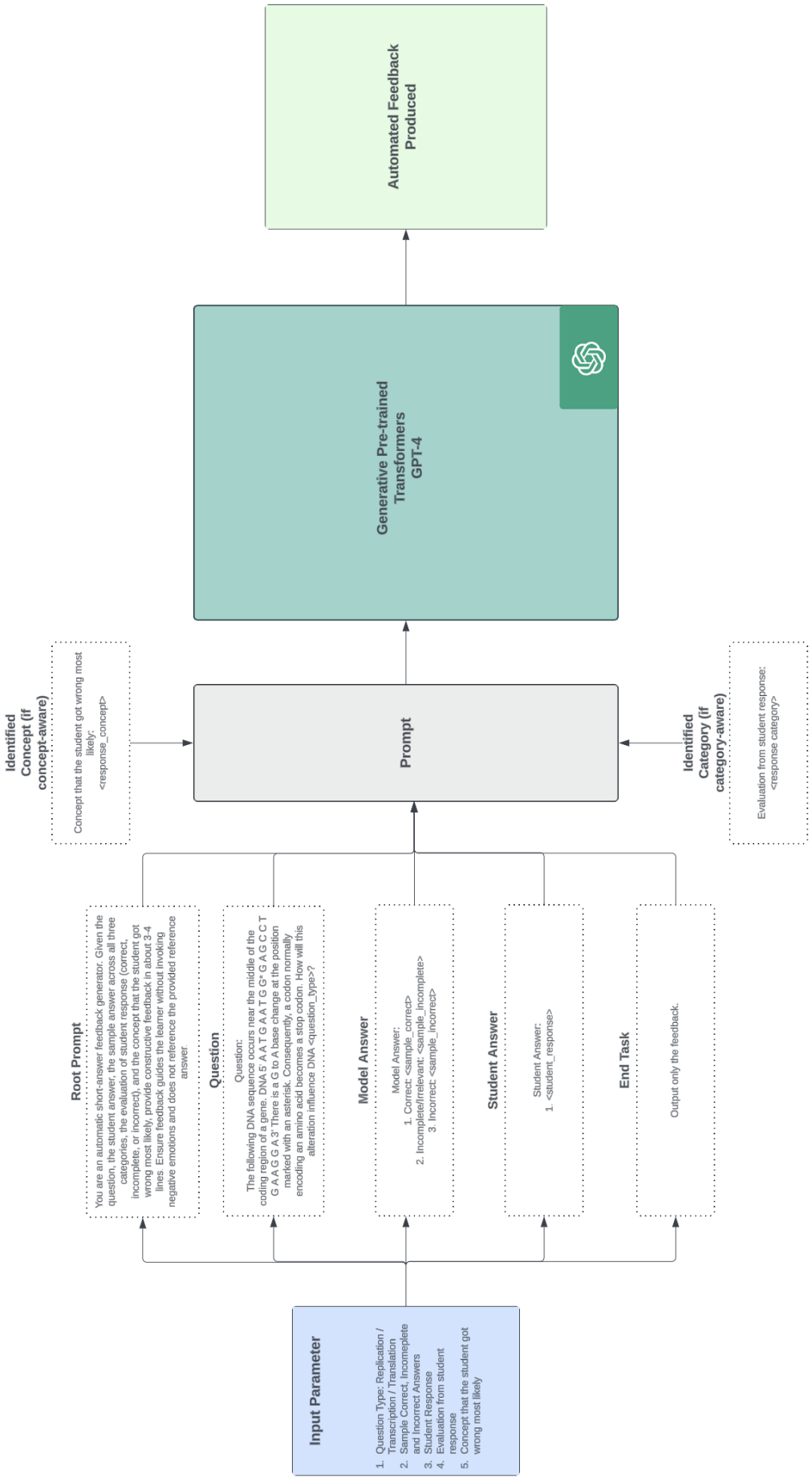

Figure 2.3: GPT 4 Model Prompting

Cluster	Explanation
0	The answer is wrong here because a premature stop codon serves the purpose of introducing an early stop while translating. Thus, a full amino acid sequence is not translated, and a premature stop codon causes a shorter peptide chain to be produced. Furthermore, the mutation is not a missense but a nonsense mutation. Had it been a missense mutation, a change in the amino acid would still have produced the same length of peptide and translated for a different amino acid. But a nonsense mutation is the one that introduces a premature stop. So, it's a nonsense mutation.
1	These answers are incorrect because overall the function of a stop codon is to bring a stop or end the process of translation when it's encountered by the ribosomes while translating. A premature stop ends the process of translation earlier than a normal translation process for that particular sequence of codons, and it's not supposed to affect the DNA or the RNA sequence during the process of replication or transcription respectively. Also, each stop codon has the same function, just like other amino acids that have multiple sets of codons coding for a particular amino acid, a stop codon codes for a stop and occurs in 3 sets of codons.
2	The process of translation will occur but it will just come to a stop earlier than usual for a particular sequence of codons or genes for which we are encoding a protein because of the introduction of a premature stop codon. Thus, it's incorrect to say that the process of translation won't occur. *Note: any mutation associated with the start codon in translation may prevent translation (depending on the mutation), but a premature stop doesn't prevent translation.
3	The answers stated by the students are incorrect because there is no frameshift mutation occurring here. The only mutation occurring is a nonsense mutation, in which a functional amino acid is converted to a stop codon. No shifting of codons is seen due to any indels (or any other mutation) that may contribute to a frameshift mutation. Furthermore, there is no missense mutation happening as well that changes the amino acid being coded for to another amino acid. The claim that a shorter mRNA transcript will be produced is false, the RNA sequence will just have a variation in the arrangement of the codons, where post the premature stop codon, there might be sequences that continue to code for the amino acid in a non-mutated situation. Thus, it's at the level of translation that the ribosome detects a stop codon and introduces a stop in translation, producing a smaller peptide chain, and leaving behind the rest of the transcript unread.

Table 2.5: Cluster Explanations for Incorrect Translation Responses - Part 1/2

2.3.7 Feedback Generation using C2AP and variations

We utilize the generative capabilities of the state-of-the-art large language models to produce feedback for a given response. The prompt consists of a root prompt (inspired from work by Aggarwal and others (Aggarwal, Bhattacharyya, and Raman, 2024)) which is as follows:

Cluster	Explanation
4	The answers by the students are incorrect because the size of the RNA remains unaffected by the introduction of a premature stop codon. It will still copy the complementary sequences from the DNA until it receives a signal to stop transcription (which is different from the stop codon). It's also wrong that an individual will lack RNA post transcription due to the stop codon because it's independent of the stop codon sequence for translation. Thus, the individual won't lack an mRNA for a premature stop codon mutation. *Note: until the mutation of a stop codon is highly deleterious and during the process of translation, there might be an occurrence of a nonsense-mediated decay and the RNA is destroyed, but this occurs while translation where the ribosome encounters a premature stop codon and if it's highly deleterious the mRNA will be destroyed later. But overall RNA is produced post-transcription. However, yes, an RNA copy can be said to be incorrect because the transcript produced for the translation of amino acid is not normal and has a premature stop that affects the peptide synthesis but the RNA copy is still incorrect.
5	The answers are incorrect because the protein, if translated, will have an effect due to the introduction of a premature stop codon and the ribosome thus might miss out on reading important sequences essential for the functioning of the organism. Thus, there will be an effect on the protein, but it cannot be said that there is no effect of the mutation on the protein. Furthermore, the mRNA is fully transcribed from the DNA irrespective of the stop codon, which only functions during translation. Thus, it's wrong to say that it'll be short. Thus, the mRNA will retain its size and it will be processed due to transcription. *Note: The effect may vary depending on the location, and the importance of the codons that failed to be translated. Thus, it can be the case that there was little effect on the protein.
6	The answers stated by the students are incorrect because the process of translation will be affected by a premature stop codon. Translation will fail to produce a whole length of peptide by stopping earlier due to the PSC (premature stop codon). Therefore, translation is impacted.

Table 2.6: Cluster Explanations for Incorrect Translation Responses - Part 2/2

You are an automatic short-answer feedback generator. Given the question, the student answer, the sample answer across all three categories, the evaluation of student response (correct, incomplete, or incorrect), and the concept that the student got wrong most likely, provide constructive feedback in about 3-4 lines. Ensure feedback guides the learner without invoking negative emotions and does not reference the provided reference answer.

Then, we provide the question, the student answer, the sample answer, the evaluation, as well as the human-identified concept in the rest of the prompt, where <substring>represents the answer provided by the student and <sample_correct>, <sample_incomplete>, and<sample_incorrect>refer to the sample correct, sample incomplete, and sample incorrect responses for each <ques-

Cluster	Explanation
0	The answers provided by the student are incomplete because, although there will be a variation in the DNA sequence due to the mutation that eventually codes for a stop codon, there is no effect of that mutation on the process of replication. The process of replication will continue as is.
1	These answers stated by the students are incomplete because there is no reference to the effect of premature stop codon on the process of DNA replication. The process of replication is supposed to have no effect on DNA synthesis and will only affect the process of translation, where it produces a shorter protein or a non-functional gene product.
2	The answers here are incomplete because these answers are more focused on the fact of the effect of the introduction of a stop codon on the DNA sequence rather than on the process of replication. Thus, the answers are missing the fact that the process of replication will still occur normally, irrespective of a nonsense mutation that affects the sequence at the translational or protein level.
3	The claim that a mutation occurs and the DNA will be error-prone is correct but the students fail to suggest the effects of these changes on the process of replication and hence it's incomplete. The process of replication despite the mutation will still occur and produce another complementary strand of DNA.

Table 2.7: Cluster Explanations for Incomplete Replication Responses

tion_type>(replication, transcription, translation). <response_category>is the identified categoy (correct, incomplete, or incorrect) and <response_concept>is the human-identified concept that the response was previously mapped to. The rest of the prompt is as follows:

Question:

The following DNA sequence occurs near the middle of the coding region of a gene.

DNA 5' A A T G A A T G G G A G C C T G A A G G A 3'*

There is a G to A base change at the position marked with an asterisk. Consequently, a codon normally encoding an amino acid becomes a stop codon.

How will this alteration influence DNA <question_type>?

Model Answer:

1. Correct: <sample_correct>

2. Incomplete/Irrelevant: <sample_incomplete>

3. Incorrect: <sample_incorrect>.

Student Answer:

1. <substring>

Evaluation from student response:

<response_category>

Concept that the student got wrong most likely:

<response_concept>

Cluster	Explanation
0	The answer stated by the student that the process of transcription will be incorrect is correct but it doesn't suggest the effect of the mutation on transcription. Therefore it's incomplete. Thus, the process of transcription will remain unaffected and transcribe or copy the incorrect DNA sequence.
1	These answers are incomplete because, although the transcript has a variation introduced in the sequence due to the premature stop codon mutation, the answers don't state that the process of transcription will still occur producing an mRNA. Furthermore, some answers also suggest the process of early stoppage of the process of translation due to the stop codon, which is correct but fails to mention the effects of the mutation on the process of transcription, and hence the answers are incomplete.
2	These answers stated by the student are incomplete because the answers don't focus on the effect of mutation on the process of transcription. The answers provided by the students are rather focused on the variations produced in the DNA or the transcripts as a result of the mutation. Furthermore, students also focus a lot on the effects of the mutation on the end product of translation, the peptide chain which will be shorter and will be unable to function properly. Thus, all these claims made by the students are correct (majorly) but they are missing the effect of the mutation on the process of transcription.
3	The answer suggesting a different mRNA sequence being produced is true, however, the answers are incomplete because they lack the explanation that the process of transcription is unaffected by the mutation of the introduction of a premature stop codon, and hence they can transcribe the DNA sequence like the non-mutated sequence.
4	The answers stated by the students are incomplete because they tend to show the variation in the mRNA sequence due to the mutation and hence the introduction of the stop codon. However, they fail to explain the effects of the mutation on the process, which is the process of transcription remains unaffected and encodes for an mRNA which is processed for the translation of the amino acid.

Table 2.8: Cluster Explanations for Incomplete Transcription Responses

Output only the feedback.

We also experiment different settings, with and without the category, with and without the concept and their possible combinations to analyze if this has any impact on the results. In that case, we change the root prompt to only suggest the information that is actually present. The GPT model prompting cycle has been visually shown in 2.3. The sample responses are shown in 2.2.

Cluster	Explanation
0	The answer is incomplete because it suggests the final result of the translation process. Initially, the ribosomes will be able to translate the mRNA to a peptide chain until it reaches a stop codon. And, in this case, the process of translation ends early since the ribosome encounters a premature stop codon. Thus, the polypeptide chain expressed is incorrect due to the mutation which is a result of the change in the sequence of nucleotides.
1	The answer that the stop codon prevents the process of translation from occurring is true but incomplete, because the stop codon prevents the process of translation from occurring beyond the point of a stop codon, and here since it's a premature stop codon, translation fails to occur beyond that. Furthermore, the answers concerning a mutation changing codon to different amino acid are incomplete because, the mutation that led to the substitution of a G to A (hence forming a stop codon), led to a change in the production of amino acid, changing from Trp to a stop codon. Thus, the answer is incorrect in the sense that the mutation does bring about a change in the coding of an amino acid, which is eventually a stop codon (premature), that stops the process of translation earlier.
2	The answer that translation won't occur, or it will be unable to translate is incomplete and is true only after the introduction of a stop codon, in this case, a premature stop codon. Thus, the process of translation won't occur after that point of the stop codon. *Note: Translation will occur but if the protein produced is deleterious, it might lead to degradation of the transcript and the peptide chain.
3	The answers are incomplete because they tend to focus on the replication process and the changes in the DNA that will produce a different final product, however, the missing part is that the process of translation will occur irrespective of a premature or a normal stop codon. Thus, a different or mutated genetic material will be translated due to the premature stop codon.

Table 2.9: Cluster Explanations for Incomplete Translation Responses - Part 1/2

2.3.8 Evaluation of Generated Feedback

To evaluate how credible and reliable the produced feedback is, we randomly sample 30% of the responses in each question type (replication, transcription, and translation) and distribute them equally across the three categories to create a representative sample of our feedback results. We ask three human experts to evaluate the produced feedback on a scale of 1-5 across the following metrics, as done by Aggarwal and others while creating EngSAF dataset (Aggarwal, Bhattacharyya, and Raman, 2024):

1. Grammatical Correctness: This aspect tests whether the feedback is grammatically correct and fluent in English. This ensures that the model generated responses sound like human responses to the learners. A higher score represents more fluent response.

Cluster	Explanation
4	The answers are incomplete because the answers focus on the RNA sequence (mutated or incorrect) that in consequence produces a shorter amino acid upon translation. The answers lack the fact that a fully transcribed RNA(irrespective of the location of the stop codon, but here it's a premature stop codon), which is incorrect due to the presence of a premature stop codon sequence, will lead to earlier stoppage of translation since the ribosome will encounter a UGA stop earlier than a normal sequence. Thus, it's incomplete in the fact that the answers don't state that these incomplete or incorrect RNA will carry a premature stop codon that will in turn affect the early stoppage of translation, producing an incomplete peptide sequence.
5	The answers are incomplete because the students are seen to draw a connection between the RNA or the DNA with the final product, the protein. This isn't incorrect but the answers (most of them) fail to mention the effect of the stop codon that will influence the process of translation, stopping it early, and producing a reduced peptide chain or protein. This may or may not be non-functional depending upon the amino acid sequences present in the chain.
6	The answer is incomplete because the students fail to state that there won't be any influence in the process of translation and that the process of translation will occur normally until it encounters a premature stop codon. Upon encountering a premature stop codon, the process of translation stops earlier.

Table 2.10: Cluster Explanations for Incomplete Translation Responses - Part 2/2

2. Feedback Accuracy: This aspect tests the overall quality of the provided feedback across relevancy to the student's response, content produced and justification of the explanation provided. A higher score represents higher quality.

3. Emotional Impact: This aspect tests the impact of the feedback on student's emotional state. It ensures that the feedback produced does not trigger any negative emotions that might distress them or discourage them (like when the word "fail", "fool", or "dumb" is used). A higher score represents more emotionally aware feedback that encourages students to learn from their mistakes.

Researchers have used metrics like Rouge-2 (Post, 2018), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and BERTSCORE (T. Zhang et al., 2019) as metrics to evaluate feedback in the past. Rouge-2 measures the overlap of bigrams, BLEU evaluates precision of n-grams (usually, upto 4), METEOR assesses the alignment using synonymy, stemming, and word order, and BERTScore uses BERT embeddings and calculates cosine similarity between generated feedback and a reference feedback. Unfortunately, due to lack of human feedback responses, none of these metrics could be calculated.

Table 2.11: Metrics for feedback evaluation across all experimental settings

	Human evaluation		
Experimental Setting	GC	FA	EI
Human Baseline	3.6	4.3	4.4
SP	3.5	2.2	4.5
CAP	3.7	3.2	4.4
C2AP	3.5	4.7	4.6

2.4 Experiments, Results and Discussion

We use the following experimental settings so we can compare the feedbacks produced and understand what are some important factors in generating quality feedback.

- Standard Prompting (SP) - We neither include the identified concept nor the identified category in our prompt to the large language model.
- Category Aware Prompting (CAP) - We only include the identified category, but not the concept in our prompt to the large language model.
- Concept and Category Aware Prompting (C2AP) - We include both, the identified concept as well as the identified category in our prompt to the large-language model.

It is important to note that we also compare the performance with human baseline, which refers to providing the concepts identified by human graders to the students as the feedback. In the baseline experiment, we got a score of 3.6, 4.3, and 4.4 for grammatical correctness, feedback accuracy, and emotional impact respectively. With the SP setting, we got a score of 3.5, 2.2, and 4.5 for grammatical correctness, feedback accuracy, and emotional impact respectively. With the CAP setting, we got a score of 3.7, 3.2, and 4.4 for grammatical correctness, feedback accuracy, and emotional impact respectively. Finally, with the C2AP setting, we got a score of 3.5, 4.7, and 4.6 for grammatical correctness, feedback accuracy, and emotional impact respectively. The metrics are also shown in table 2.11.

This suggests that overall, grammatical correctness and emotional impact were similar across all the feedback categories. This suggests that the GPT-4o model was able to keep up with the instructions and ensure that feedback is encouraging for students. However, the main difference across all the four settings came in the feedback accuracy. The standard prompting performs the worst with a score of 2.2, which is reasonable because based on the grading results from Agarwal and Moghimi's work (Agarwal and Moghimi, 2024) on the same dataset, the non-fine tuned models performed far worse on category identification task itself. In such a case, it is reasonable for us to assume that the feedback calculated won't be accurate either. However, given the category of the response, the model seems to perform better with a score of 3.2 as

expected. The human baseline from the concepts is still better with a score of 4.3. However, providing the category and the concept improved the results considerably, even better than the human baseline with a score of 4.7/5. Our understanding is that this is reasonable because the human baseline has the exact same feedback for all the responses in a cluster, despite clusters having some outlying responses. It seems from the results that the large language model was able to understand the overarching idea and refine the feedback for even these outlying responses to provide a higher accuracy, which was the intent behind the approach.

2.5 Conclusion

In this paper, we presented a novel framework for automated feedback generation that is based on Grading with Near Domain data to identify categories, semantic clustering to identify concepts (with human assistance), and prompting large language models to generate the feedback. Based on our results, we conclude that providing both categories and concepts are useful for the generation of good quality feedback using LLMs.

2.6 Acknowledgment

We owe special thanks to our human-experts who spent their valuable time reading through the different responses, and evaluating the generated feedback. Special thanks to Maanvi Kaushal Shah, a Genetics and Genomics Senior at University of California, Davis for helping analyze the student responses and identifying overarching themes and concepts. This work wouldn't have been possible without their support and kind assistance.

BIBLIOGRAPHY

Agarwal, Shyam and Mahasweta Chakraborti (2024). "The Hidden AI Race: Tracking Environmental Costs of Innovation". In: *arXiv preprint*. Preprint.

Agarwal, Shyam and Ali Moghimi (2024). "Grading with Near-Domain Data: Towards Human-level Accuracy for Automated Feedback on Short Response Questions". In: *arXiv preprint*. Preprint.

Aggarwal, Dishank, Pushpak Bhattacharyya, and Bhaskaran Raman (2024). ""I understand why I got this grade": Automatic Short Answer Grading with Feedback". In: *arXiv preprint* arXiv:2407.12818. URL: https://arxiv.org/pdf/2407.12818.

American Association for the Advancement of Science (2011). *Vision and Change in Undergraduate Biology Education: A Call to Action*. Washington, DC: American Association for the Advancement of Science.

Azevedo, Roger and Robert M Bernard (1995). "A meta-analysis of the effects of feedback in computer-based instruction". In: *Journal of Educational Computing Research* 13.2, pp. 111–127.

Banerjee, Satanjeev and Alon Lavie (2005). "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments". In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72.

Bangert-Drowns, Robert L et al. (1991). "The instructional effect of feedback in test-like events". In: *Review of Educational Research* 61.2, pp. 213–238.

Basu, Sreyasi, Claire Jacobs, and Lucy Vanderwende (2013). "Powergrading: a clustering approach to amplify human effort for short answer grading". In: *Transactions of the Association for Computational Linguistics* 1, pp. 391–402.

Bowen, William G. (2012). "The 'cost disease' in higher education: Is technology the answer?" In: *The Tanner Lectures, Stanford University*.

Burrows, Steven, Iryna Gurevych, and Benno Stein (2015). "The eras and trends of automatic short answer grading". In: *International Journal of Artificial Intelligence in Education* 25.1, pp. 60–117. DOI: 10.1007/s40593-014-0026-8.

Camus, L. and A. Filighera (2020). "Investigating Transformers for Automatic Short Answer Grading". In: *International Conference on Artificial Intelligence in Education*. Springer, Cham, pp. 43–48.

Cohn, Chloe et al. (2024). "A Chain-of-Thought Prompting Approach with LLMs for Evaluating Students' Formative Assessment Responses in Science". In: *arXiv preprint*. arXiv: 2403.14565 [cs.CL]. URL: http://arxiv.org/abs/2403.14565.

Corbett, Albert T and John R Anderson (1989). "Feedback timing and student control in the LISP intelligent tutoring system". In: *Proceedings of the fourth international conference on Artificial Intelligence and Education*. Ed. by D Bierman, J Brueker, and J Sandberg. Springfield, VA: IOS, pp. 64–72.

Devlin, Jacob et al. (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *arXiv preprint arXiv:1810.04805*.

Epstein, Michael L et al. (2002). "Immediate feedback assessment technique promotes learning and corrects inaccurate first responses". In: *The Psychological Record* 52, pp. 187–201.

Fisher, K M (1985). "A misconception in biology: amino acids and translation". In: *J Res Sci Teach* 22, pp. 53–62.

Gaddipati, S. K., D. Nair, and P. G. Plöger (2020). "Comparative Evaluation of Pretrained Transfer Learning Models on Automatic Short Answer Grading". In: *arXiv preprint*. URL: https://arxiv.org/abs/.

Gerard, Lisa and Marcia C. Linn (2022). "Computer-based guidance to support students' revision of their science explanations". In: *Computers & Education* 176, p. 104351. DOI: 10.1016/j.compedu.2021.104351.

Gerard, Lisa, Marcia C. Linn, and Jyotsana Madhok (2016). "Examining the impacts of annotation and automated guidance on essay revision and science learning". In: *Transforming Learning, Empowering Learners: The International Conference of the Learning Sciences (ICLS)*. Ed. by Chee-Kit Looi et al.

Golchin, Shahriar et al. (2024). "Grading Massive Open Online Courses Using Large Language Models". In: *arXiv preprint arXiv:2406.11102*. https://arxiv.org/abs/2406.11102.

Haller, Stefan et al. (2022). "Survey on Automated Short Answer Grading with Deep Learning: From Word Embeddings to Transformers". In: *arXiv preprint arXiv:2204.03503*.

Hämäläinen, Wilhelmiina et al. (2018). "Clustering students' open-ended questionnaire answers". In: *arXiv preprint arXiv:1809.07306*. Accessed: 2024-11-29. URL: https://arxiv.org/pdf/1809.07306.

Hattie, John and Helen Timperley (2007). "The power of feedback". In: *Review of Educational Research* 77.1, pp. 81–112.

Heilman, Michael and Nitin Madnani (2013). "ETS: Domain Adaptation and Stacking for Short Answer Scoring". In: *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 275–279.

Henkel, Owen et al. (2023). "Can LLMs Grade Short-Answer Reading Comprehension Questions? An Empirical Study with a Novel Dataset". In: Submitted: October 2023.

Hou, W. J. and J. H. Tsao (2011). "Automatic Assessment of Students' Free-Text Answers with Different Levels". In: *International Journal on Artificial Intelligence Tools* 20.2, pp. 327–347.

Howard, Jeremy and Sebastian Ruder (July 2018). "Universal Language Model Fine-tuning for Text Classification". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 328–339. DOI: 10.18653/v1/P18-1031. URL: https://aclanthology.org/P18-1031.

Kang, J. H., K. Lerman, and A. Plangprasopchok (2010). "Analyzing microblogs with affinity propagation". In: *Proceedings of the First Workshop on Social Media Analytics*. New York, NY, USA: ACM, pp. 67–70.

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

Kortemeyer, Gerd (2023). "Performance of the Pre-Trained Large Language Model GPT-4 on Automated Short Answer Grading". In: *Educational Development and Technology, ETH Zurich*. Dated: September 19, 2023.

Kulhavy, Raymond W. (1977). "Feedback in Written Instruction". In: *Review of Educational Research* 47.2, pp. 211–232. DOI: 10.3102/00346543047002211.

Lan, Amy S et al. (2015). "Mathematical language processing: Automatic grading and feedback for open response mathematical questions". In: *Proceedings of the Second (2015) ACM Conference on Learning@Scale*, pp. 167–176.

Leacock, Claudia and Martin Chodorow (2003). "C-rater: Automated scoring of short-answer questions". In: *Computers and the Humanities* 37.4, pp. 389–405.

Lepper, Mark R and Ruth W Chabay (1985). "Intrinsic motivation and instruction: Conflicting views on the role of motivational processes in computer-based education". In: *Educational Psychologist* 20.4, pp. 217–230.

Lester, Brian, Rami Al-Rfou, and Noah Constant (Nov. 2021). "The Power of Scale for Parameter-Efficient Prompt Tuning". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 3045–3059. DOI: 10.18653/v1/2021.emnlp-main.243. URL: https://aclanthology.org/2021.emnlp-main.243.

Lewis, J, J Leach, and C Wood-Robinson (2000). "All in the genes? Young people's understanding of the nature of genes". In: *J Biol Educ* 34, pp. 74–79.

Liu, T. et al. (2019). "Automatic Short Answer Grading via Multiway Attention Networks". In: *International Conference on Artificial Intelligence in Education*. Springer, Cham, pp. 169–173.

Madnani, N. et al. (2013). "Automated Scoring of a Summary Writing Task Designed to Measure Reading Comprehension". In: *Proceedings of the 8th Workshop on Innovative Use of NLP for Building Educational Applications*. Ed. by J. B. Tetreault and C. Leacock. Atlanta: Association for Computational Linguistics, pp. 163–168.

Malik, Aisha et al. (Mar. 2021). *Generative grading: Near human-level accuracy for automated feedback on richly structured problems*. URL: https://arxiv.org/abs/1905.09916v5.

Marbach-Ad, G (2001). "Attempting to break the code in student comprehension of genetic concepts". In: *J Biol Educ* 35, pp. 183–189.

Metzler, D., S. Dumais, and C. Meek (2007). "Similarity measures for short segments of text". In: *Proceedings of the 29th European Conference on IR Research*. Berlin, Heidelberg: Springer-Verlag, pp. 16–27.

Meyer, Jennifer et al. (2024). "Using LLMs to bring evidence-based feedback into the classroom: AI-generated feedback increases secondary students' text revision, motivation, and positive emotions". In: *Computers and Education: Artificial Intelligence* 6, p. 100199. ISSN: 2666-920X. DOI: 10.1016/j.caeai.2023.100199. URL: https://www.sciencedirect.com/science/article/pii/S2666920X23000784.

Mikolov, Tomas et al. (2013). *Efficient Estimation of Word Representations in Vector Space*. arXiv: 1301.3781 [cs.CL]. URL: https://arxiv.org/abs/1301.3781.

Mills Shaw, K R et al. (2008). "Essay contest reveals misconceptions of high school students in genetics content". In: *Genetics* 178, pp. 1157–1168.

Mizumoto, Tomoya et al. (2019). "Analytic Score Prediction and Justification Identification in Automated Short Answer Scoring". In: *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 316–325.

Mohler, Michael, Razvan Bunescu, and Rada Mihalcea (2011). "Learning to grade short answer questions using semantic similarity measures and dependency graph alignments". In: *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 752–762.

Mohler, Michael and Rada Mihalcea (2009). "Text-to-text semantic similarity for automatic short answer grading". In: *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pp. 567–575.

Moreno, Roxana (2004). "Decreasing cognitive load for novice students: Effects of explanatory versus corrective feedback in discovery-based multimedia". In: *Instructional Science* 32, pp. 99–113.

Narciss, Susanne and Katrin Huth (2004). "How to design informative tutoring feedback for multimedia learning". In: *Instructional design for multimedia learning*. Ed. by Hans M Niegemann, Detlev Leutner, and Roland Brunken. Munster, New York: Waxmann, pp. 181–195.

NGSS Lead States (2013). *Next Generation Science Standards: For States, by States*. Washington, DC: National Academies Press.

Nguyen, Andy et al. (2014). "Codewebs: scalable homework search for massive open online programming courses". In: *Proceedings of the 23rd International Conference on World Wide Web*, pp. 491–502.

Ni, X. et al. (June 2011). "Short text clustering by finding core terms". In: *Knowledge and Information Systems* 27.3, pp. 345–365.

Nicol, D. and D. Macfarlane-Dick (2006). "Formative Assessment and Self-Regulated Learning: A Model and Seven Principles of Good Feedback Practice". In: *Studies in Higher Education* 31, pp. 199–218. DOI: 10.1080/03075070600572090.

Nyquist, Julie G and Rima Jubran (Fall 2012). "How Learning Works: Seven Research-Based Principles for Smart Teaching". eng. In: *The Journal of Chiropractic Education* 26.2, pp. 192–193. ISSN: 1042-5055. DOI: 10.7899/JCE-12-022.

Ouyang, Long et al. (2022). "Training language models to follow instructions with human feedback". In: *arXiv preprint*. DOI: 10.48550/arXiv.2203.02155. arXiv: 2203.02155 [cs.CL]. URL: https://doi.org/10.48550/arXiv.2203.02155.

Papineni, Kishore et al. (2002). "Bleu: a Method for Automatic Evaluation of Machine Translation". In: *Annual Meeting of the Association for Computational Linguistics*. URL: https://api.semanticscholar.org/CorpusID:11080756.

Pennington, Jeffrey, Richard Socher, and Christopher Manning (Oct. 2014). "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Alessandro Moschitti, Bo Pang, and Walter Daelemans. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

Piech, Christopher James (2016). "Uncovering patterns in student work: Machine learning to understand human learning". PhD thesis. Stanford University.

Pinto, D., J.-M. Benedí, and P. Rosso (2007). "Clustering narrow-domain short texts by using the Kullback-Leibler distance". In: *Proceedings of the 8th International Conference on Computational Linguistics and Intelligent Text Processing*. Berlin, Heidelberg: Springer-Verlag, pp. 611–622.

Post, Matt (2018). "A call for clarity in reporting BLEU scores". In: *arXiv preprint arXiv:1804.08771*.

Prevost, Luanna B., Michelle K. Smith, and Jennifer K. Knight (2016). "Using Student Writing and Lexical Analysis to Reveal Student Thinking about the Role of Stop Codons in the Central Dogma". In: *CBE Life Sciences Education* 15.4, ar65. DOI: 10.1187/cbe.15-12-0267.

Pridemore, Duane R and James D Klein (1995). "Control of practice and level of feedback in computer-based instruction". In: *Contemporary Educational Psychology* 20, pp. 444–450.

Qi, H. et al. (2019). "Attention-based Hybrid Model for Automatic Short Answer Scoring". In: *SIMUtools 2019*. Vol. 295. LNICST. Springer, Cham, pp. 385–394.

Radford, Alec, Karthik Narasimhan, et al. (2018). "Improving language understanding by generative pre-training". In.

Radford, Alec, Jeffrey Wu, et al. (2019). "Language models are unsupervised multitask learners". In: *OpenAI Blog* 1.8, p. 9.

Reimers, Nils and Iryna Gurevych (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. arXiv: 1908.10084 [cs.CL].

Riordan, Brian et al. (2017). "Investigating Neural Architectures for Short Answer Scoring". In: *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 159–168.

Sadler, D. R. (1989). "Formative assessment and the design of instructional systems". In: *Instructional Science* 18, pp. 119–144. DOI: 10.1007/BF00117714.

Saha, Suman et al. (2018). "Sentence level or token level features for automatic short answer grading?: Use both". In: *International conference on artificial intelligence in education*. Springer, pp. 503–517.

Santurkar, Shibani et al. (2023). *Whose Opinions Do Language Models Reflect?* Available at https://arxiv.org/pdf/2303.17548v1.pdf. arXiv: 2303.17548 [cs.CL].

Scarlatos, Alexander et al. (2024). "Improving the Validity of Automatically Generated Feedback via Reinforcement Learning". In: *arXiv preprint* arXiv:2403.01304. License: CC BY 4.0. URL: https://arxiv.org/abs/2403.01304.

Schneider, Johannes, Bernd Schenk, and Christina Niklaus (2024). "Towards LLM-based Autograding for Short Textual Answers". In: *arXiv preprint*. Submitted on 9 Sep 2023 (v1), last revised 8 Jul 2024 (this version, v2).

Selvi, P. and A. K. Bnerjee (2010). *Automatic Short -Answer Grading System (ASAGS)*. arXiv: 1011.1742 [cs.OH]. URL: https://arxiv.org/abs/1011.1742.

Shazeer, Noam (2020). "GLU Variants Improve Transformer". In: *arXiv preprint arXiv:2002.05202*. DOI: 10.48550/arXiv.2002.05202. arXiv: 2002.05202 [cs.LG].

Shute, V. J. (2008). "Focus on Formative Feedback". In: *Review of Educational Research* 78.1, pp. 153–189. DOI: 10.3102/0034654307313795.

Smith, M K and J K Knight (2012). "Using the Genetics Concept Assessment to document persistent conceptual difficulties in undergraduate genetics courses". In: *Genetics* 191, pp. 21–32.

Smith, M K, W B Wood, and J K Knight (2008). "The Genetics Concept Assessment: a new concept inventory for gauging student understanding of genetics". In: *CBE Life Sci Educ* 7, pp. 422–430.

Sultan, Muhammad Abdul-Mageed, Cristian Salazar, and Tamara Sumner (2016). "Fast and easy short answer grading with high accuracy". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1070–1075.

Sung, Chul et al. (2019). "Pretraining BERT on Domain Resources for Short Answer Grading". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 6071–6075.

Süzen, Nevzat et al. (2020). "Automatic short answer grading and feedback using text mining methods". In: *Procedia Computer Science* 169, pp. 726–743.

Tan, H. et al. (2020). "Automatic Short Answer Grading by Encoding Student Responses via a Graph Convolutional Network". In: *Interactive Learning Environments*, pp. 1–15.

Tansomboon, Chanchai et al. (2017). "Designing automated guidance to promote productive revision of science explanations". In: *International Journal of Artificial Intelligence in Education* 17, pp. 729–757. DOI: 10.1007/s40593-017-0145-0.

Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *arXiv preprint arXiv:1706.03762*.

Wang, Tianqi et al. (2021). "Data Augmentation by Rubrics for Short Answer Grading". In: *Journal of Natural Language Processing* 28.1, pp. 183–205.

Wang, Wenhui et al. (2020). *MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers*. arXiv: 2002.10957 [cs.CL]. URL: https://arxiv.org/abs/2002.10957.

Wood-Robinson, C (2000). "Young people's understanding of the nature of genetic information in the cells of an organism". In: *J Biol Educ* 35, pp. 29–36.

Wright, L K, J N Fisk, and D L Newman (2014). "DNA → RNA: what do students think the arrow means?" In: *CBE Life Sci Educ* 13, pp. 338–348.

Wu, Mike et al. (2018). "Zero Shot Learning for Code Education: Rubric Sampling with Deep Learning Inference". In: *ArXiv* abs/1809.01357. URL: https://api.semanticscholar.org/CorpusID:52166291.

Yang, H., A. Mysore, and S. Wallace (2009). "A semi-supervised topic-driven approach for clustering textual answers to survey questions". In: *Advanced Data Mining and Applications*. Vol. 5678. Lecture Notes in Computer Science (LNCS). Springer Berlin Heidelberg, pp. 374–385.

Yang, X. et al. (2018). "Automatic Chinese Short Answer Grading with Deep Autoencoder". In: *International Conference on Artificial Intelligence in Education*. Springer, Cham, pp. 399–404.

Yoon, Su-Youn (2023). "Short Answer Grading Using One-shot Prompting and Text Similarity Scoring Model". In: *arXiv preprint arXiv:2305.18638*. https://arxiv.org/abs/2305.18638.

Zhang, Tianyi et al. (2019). "BERTScore: Evaluating Text Generation with BERT". In: *arXiv preprint arXiv:1904.09675*.

Zhang, Y., R. Shah, and M. Chi (2016). "Deep Learning + Student Modeling + Clustering: A Recipe for Effective Automatic Short Answer Grading". In: *Proceedings of the International Educational Data Mining Society*.

Zhao, Jinjin et al. (2021). "Targeted Feedback Generation for Constructed-Response Questions". In: URL: https://api.semanticscholar.org/CorpusID:233295226.