

SteLLA: A Structured Grading System Using LLMs with RAG

Hefei Qiu*, Brian White[†], Ashley Ding[‡], Reinaldo Costa[†], Ali Hachem[†], Wei Ding[†] and Ping Chen[†]

*Department of Computer Science

Fitchburg State University, 160 Pearl Street, Fitchburg, MA 01420-2697

[†]Department of Computer Science

University of Massachusetts Boston, 100 Morrissey Blvd, Boston, MA 02125

Email: {brian.white, reinaldo.costa001, ali.hachem002, wei.ding, ping.chen}@umb.edu

[‡]Chantilly High School

4201 Stringfellow Rd, Chantilly, VA 20151

Email: ashley.ding6@gmail.com

Abstract—Large Language Models (LLMs) have shown strong general capabilities in many applications. However, how to make them reliable tools for some specific tasks such as automated short answer grading (ASAG) remains a challenge. We present SteLLA (Structured Grading System Using LLMs with RAG) in which a) Retrieval Augmented Generation (RAG) approach is used to empower LLMs specifically on the ASAG task by extracting structured information from the highly relevant and reliable external knowledge based on the instructor-provided reference answer and rubric, b) an LLM performs a structured and question-answering-based evaluation of student answers to provide analytical grades and feedback. A real-world dataset which contains students' answers in an exam was collected from a college-level Biology course. Experiments show that our proposed system can achieve substantial agreement with the human grader while providing break-down grades and feedback on all the knowledge points examined in the problem. A qualitative and error analysis of the feedback generated by GPT4 shows that GPT4 is good at capturing facts while may prone to inferring too much implication from the given text in the grading task which provides insights into the usage of LLMs in the ASAG system.

Index Terms—LLM-based ASAG system, RAG, QA-based evaluation, structured evaluation

I. INTRODUCTION

Assessment plays an important role in the teaching and learning process. It usually includes closed-ended questions such as multiple choices and open-ended questions such as short-answer questions. Although open-ended questions are more powerful in evaluating students' learning, grading on such questions are more time-consuming. In some scenarios such as introductory-level courses in college with hundreds of students, or online courses with an even larger scale of learners, the potentially heavy workload due to the manual grading on open-ended short-answer questions hinders their usage in practice. An automated grading system can provide prompt feedback to a learner, can support large scale learning environment, and further facilitate active and life-long learning. The recent development of Large Language Models (LLMs) has shown their strong general capabilities in many tasks. However, how to use them to automatically provide

reliable grading and feedback remains a challenge. We propose SteLLA (Structured Grading System Using LLMs with RAG), an automatic grading system that performs a structured grading based on Question Answering (QA) techniques, which is empowered by highly relevant augmented information retrieved from the instructor-provided reference answer and rubric.

The field of automatic grading and feedback systems has been explored through various domains such as programming [1], [2] and mathematics [3], [4], as well as on different types of answers such as essays [5], [6] and short answers [7]–[9]. Compared with an essay which is usually long and with multiple paragraphs, a short answer is much shorter and with just a couple of sentences. Grading on short answers is more focused on correctness and does not consider text coherence or writing style as in essay grading. SteLLA is a system designed for automatic short-answer grading (ASAG).

There have been many attempts to build automatic grading and feedback systems. Many of them utilize the recent development in Natural Language Processing (NLP). Motivated by the huge progress of LLMs and the needs of instructors and learners, our design uses LLMs as a key component. To ground a general LLM on the specific task of grading, we propose a reference answer and rubric based retrieval augmented generation (R-RAG) approach. Given an instructor-provided reference answer and a rubric, R-RAG extracts highly relevant and structured information from them. It applies question-generation and question-answering techniques to generate a set of evaluation questions and corresponding answers. An LLM performs a structured grading by checking how well a student's response answers these evaluation questions. Eventually, an overall grade, the breakdown grades and feedback are generated to the user.

The contributions of this work are as follows:

- We propose an LLM-based ASAG system, SteLLA, that shows substantial agreement with human graders.
- We present R-RAG which is specifically designed for the grading task. It treats an instructor-provided reference answer and rubric as a knowledge base to extract highly

relevant augmented information to ground a general-trained LLM on the grading task.

- Our system is the first attempt to apply a QA-based structured grading. Compared with the text-similarity-based grading approach, i.e., directly comparing the similarity between the student answer and the reference answer, the QA-based approach provides a tool to induce a deeper semantic understanding of the text in grading. Moreover, It provides not only an overall grade but also decomposed grades and feedback on the knowledge points examined in the problem.
- We systematically analyze the responses generated by an LLM and show both of its capability and the errors it is prone to make, which provides some insights on how to properly use an LLM in the grading task.

The rest of the paper is organized as follows: Section II introduces the background and related work; method and system architecture are explained in Section III; how we collected the data is described in Section IV; Section V presents the experiments and results; the last section, Section VI, gives the conclusion and future work.

II. BACKGROUND AND RELATED WORK

A. QA-Based Evaluation

While Question Answering itself is one of the major tasks in NLP, the QA-based approach is novel in applying QA techniques to perform text evaluation for other NLP tasks. This approach has been applied to evaluate the quality of texts in summarization or text compression tasks. Some of the earlier work used the QA evaluation diagram to examine to what extent documents could be summarized while not affecting comprehension on them [10], to perform human evaluations of summaries [11]. Along with the progress of question generation techniques, multiple researches have been done on automatically generate questions from the reference summary [12], [13], from the source document [14], and from the evaluated summary [15], [16] to check fact-based consistency or faithfulness. Extended from previous work, QuestEval combines both recall and precision approaches and shows an improved QA-based metric on evaluating summarization [17]. QuestEval is also applied on evaluating text simplification [18] and text converted from semi-structured data such as table [19]. To the best of our knowledge, our work is the first attempt to apply QA-based evaluation to the grading and feedback task.

B. Large Language Models (LLMs)

Language Modeling (LM) has been one of the central tasks in NLP. In general, LM is to learn a probability distribution over sequences of tokens by predicting the probabilities of the next or missing token(s). Pre-trained language models such as BERT [20] have shown surprising capability in learning context-aware word representations and achieved high performance in a series of NLP tasks. Since the launching of GPT-3 [21], LLMs have attracted a lot of attention. Compared with pre-trained language models, LLMs are scaled with a much

larger size of model parameters and training data. They show emerging abilities to solve more complex tasks. ChatGPT (OpenAI 2022), developed upon the GPT-3 (OpenAI 2021) and above series, provides a highly accessible and effective way to use LLMs in a conversational manner and without fine-tuning. This intimates a large number of research and applications. The most recent versions, GPT-4 (OpenAI 2023) and GPT-4o (OpenAI 2024), are multimodal models that accept both text and images as inputs.

Recent LLMs use Transformer [22] as the backbone architecture of the models. Originally introduced for the machine translation task, the vanilla Transformer is built on an encoder-decoder structure. The encoder and decoder are both a stack of transformer blocks. Through the multi-head attention mechanism, the encoder encodes the input sentence in one language into a latent space of representation; the decoder decodes this representation to autoregressively generate the translated sentence. Different from the vanilla Transformer, the GPT series uses the decoder only.

C. Retrieval-Augmented Generation

Although LLMs have shown strong general capabilities, there are some key challenges these models are still suffering from, e.g., factual hallucination [23]–[25]. Retrieval-Augmented Generation (RAG) [26], [27] has been proposed and established to be a technique to alleviate such challenges. It references reliable external knowledge by retrieving relevant information and further enhances the performance of LLMs. Some of the works use the retrieved data as augmented inputs to guide the generation of LLMs [26], [28]. Others apply this approach in the middle of generation [29], [30] or after the generation [31], [32]. We apply RAG by using it to retrieve augmented information as inputs. We treat an instructor-provided reference answer as an external knowledge base, extract information that contains the target answer to an evaluation question, and send it together with the student response and the evaluation question to an LLM to perform the assessment.

D. Automatic Short-Answer Grading

The research on the ASAG has a long history. In the earlier days of ASAG research, many traditional methods used rule-based models [33]. For example, the idea of Concept Mapping is more rule-based, which breaks the student answers into several concepts and detects if each concept is present or not [34]–[36]. The approach that uses information retrieval techniques is also more rule-based. It usually checks student answers more by relying on pattern matching through, e.g., regular expressions or parsing trees [37], [38].

Along with the development of machine learning in NLP, it also has become popular in ASAG systems. Some of them apply clustering methods such as grouping together student responses using LDA clustering to lessen the workload for a human grader [7] or using k-means algorithm based on common word similarity [8]. Others treat it as a classification

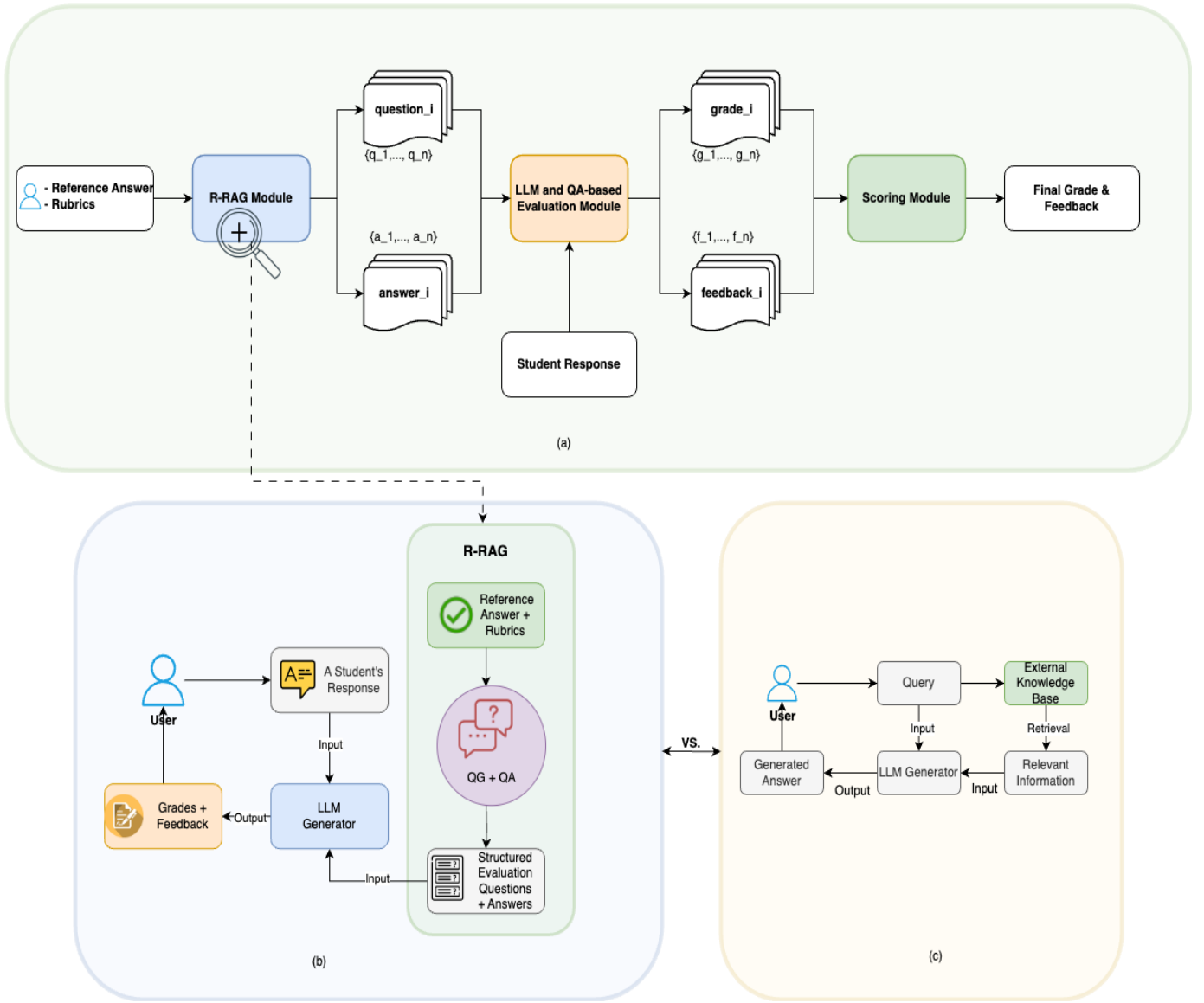


Fig. 1. (a) System architecture of SteLLA consisting of i) R-RAG Module which takes the instructor-provided reference answer and rubrics as inputs, generates and extracts a list of evaluation questions with gold answers, and sends it to the LLM; ii) LLM and QA-based Evaluation Module in which an LLM is prompted to perform grading using QA-based evaluation approach; iii) Scoring Module which generates a final grade and feedback. (b) R-RAG approach (c) Typical RAG approach

problem using, for example, a k-nearest neighbor classifier to detect and diagnose semantic errors in student answers [39].

Most recently, interest in Pre-trained Language Models (PLMs) and LLMs has increased significantly. In accordance, there has been many research on the possible applications of LLMs to the educational field [40]. PLMs such as BERT can be pre-trained on domain resources to improve ASAG. [9] uses LLM-based one-shot prompting and a text similarity scoring model based on Sentence-BERT [41] to grade short answers. [42] evaluated using ChatGPT to perform auto-grading on short text answers, in which they use ChatGPT to directly assess answers by both the educator and the students. They concluded that LLMs currently can be used as a complementary viewpoint but are not ready as an independent tool yet.

Our approach is different from the above in the way that we use the instructor-provided reference answer and rubrics as highly relevant external knowledge base, extract structured information in the form of evaluation question-answer pairs, and then ask LLMs to assess to what extent a student's response answers all these evaluation questions.

III. METHOD AND SYSTEM ARCHITECTURE

In this section, we present our approach and the system design. The overall method is to apply the RAG approach to generate structured evaluation questions and corresponding answers from the instructor-provided reference answer and rubrics to a problem. These augmented evaluation question-answer pairs are used to ground an LLM's grading. Together

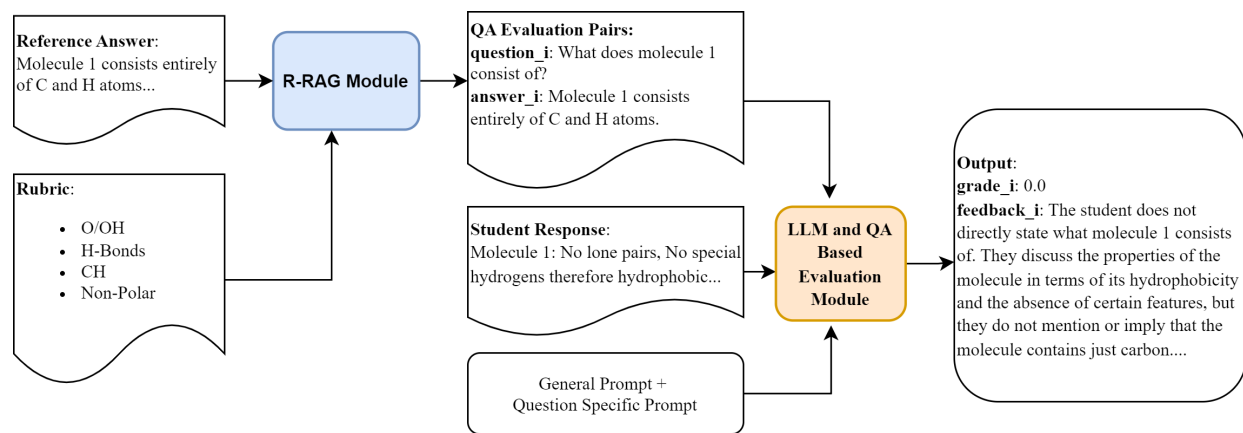


Fig. 2. An example to show the flow of grading.

with a student's response and prompts, they are sent to an LLM as inputs. The LLM performs the question-answering task to assess to what extent a student's response answers all these evaluation questions and gives the grades and feedback. The grades of all these questions are eventually consolidated into a final grade. Figure 1 part(a) shows the design of the entire system. A concrete example in Figure 2 illustrates the flow of grading. The proposed system is composed of three key modules: a) the R-RAG module based on the reference answer and rubric, b) the Evaluation module based on LLM and QA, and c) the Scoring module. All the modules are explained in detail in the following.

A. R-RAG Module

The R-RAG module applies RAG approach based on the reference answer and rubrics. It is specifically designed for the grading task. A typical RAG approach is shown in Figure 1 part (c). Given a query from the user, a retriever, usually using information retrieval techniques, retrieves relevant information from an external knowledge base such as Wikipedia or other reliable datasets. This highly relevant information serves as part of the prompts and guides the LLM to generate specific results for the given query.

As shown in Figure 1 part (b), the R-RAG module takes the instructor-provided reference answer and rubrics as inputs, generates and extracts a list of evaluation questions with gold answers, and sends it to the LLM. More specifically, given a full-credit reference response r and a rubric b , each rubric point is marked as a conditioned target answer. A question-generation model will generate a corresponding question for each target answer based on the reference answer. For example, For the rubric point "C and H", the corresponding question could be "What does molecule 1 consist of?" Eventually, this module will generate a set of evaluation questions $Q = \{q_1, \dots, q_n\}$ and their gold answers $L = \{l_1, \dots, l_n\}$, where n is the length of the rubric points. Each evaluation question reflects a rubric point. Each gold answer is supported by both the reference response and the rubric. The R-RAG

module has some unique designs specifically for the grading task.

Highly Relevant Knowledge Base. R-RAG treats the instructor-provided reference answers and rubrics as an external knowledge base, which is highly relevant to the grading task that the LLM is going to perform. Normally, the external knowledge that the RAG approach relies on is very large and requires sophisticated techniques to retrieve query-relevant information. Inspired by the traditional learning assessment process in which an instructor usually provides a reference answer and rubrics to facilitate graders in grading, we directly use such available data as external knowledge. They are small and highly relevant to the student's responses that are needed to be graded. This gives the potential to simplify the system and further enhance its usage.

Structured Information. Due to the nature of external knowledge typically used in RAG which is large, some information retrieval techniques such as ranking are usually used to get the most relevant information. In R-RAG, instead of retrieving ranked relevant information, we aim to extract structured information. This is chosen to perform a structured assessment. To a learner, while it's important to get a correct grade on the answer, it's even more important to understand the knowledge points tested in the problem and how he/she does on each of them. A structured assessment provides more valuable feedback to improve both learning and teaching. Under this consideration, the outputs from the R-RAG are structured following the rubrics, each of which reflects a rubric point.

QA-Based Evaluation. When humans grade a student's response to a problem, we do not just compare how similar it is with the reference answer. Instead, for each knowledge point, we ask if the student's response answers it correctly. Inspired by this human grading process, question-answering becomes a natural approach in our automatic grading system. Each bullet point in a rubric is marked as a conditioned answer, a question generation model is applied to generate a question to it based on the reference answer. Meanwhile, a subset of the reference answer which contains the conditioned answer

phrase is also extracted for the generated question. They form a question-answer pair. A list of such pairs will be sent as part of inputs to the LLM.

B. LLM-based Evaluation Module

The LLM-based evaluation module takes the outputs from the R-RAG Module, a student's response, and other prompts as inputs. The outputs from this module are a set of numeric grades and detailed feedback to justify its grading.

We apply zero-shot and few-shot learning when prompting the LLM. To better select shots, which are a few task-specific samples provided to an LLM, we use clustering techniques to select learning samples. All students' responses are sent to a sentence encoder such as SBERT [41] to get their embeddings. Then a clustering algorithm such as KMeans is applied to group them into k clusters. The centroids of all clusters are identified and selected as the few-shots. If a centroid is not a student's response, then find the student's response that is the closest to the centroid.

C. Scoring Module

The Scoring module takes the set of grades and feedback from the Evaluation module as inputs. Based on the weights of each evaluation question, this module performs the calculation such as weighted sum to generate a final grade of a student's response and a unified feedback. Since the final grade & feedback and the breakdown grades & feedback are all valuable, they are all presented to the user as the outputs from the system.

IV. DATA

In this section, we report the data collected for this study. We first describe the data source, then explain how we redact the data to protect students' privacy, and lastly present statistics of the data.

A. Data Source

The data used in this study are collected from an undergraduate-level introductory Biology course in the semester of Fall 2018 at a public university in the United States. The data are student's answers to a problem from an exam. We will make the dataset public after publication. As shown in Figure 3, in part (a) of this problem, students are provided with 3 images of different molecules and asked to rank them in the order from the most hydrophobic to the most hydrophilic. In part (b) of the problem, students are asked to briefly explain their choices in part a. Their short answers in part (b) are the data collected for this study.

B. Privacy Protection

We take our responsibility to protect students' privacy seriously. The data used in this study are all under the approval of the Institutional Review Board (IRB) at the school where the data are collected. We redact the data to make them de-identified through the following pre-processing: a) Removing student names and using file names as index instead; b) Removing any information in the answers that can be linked to any specific individual.

C. Labeling Process

Two undergraduate Research Assistants, who had taken the same Biology course before and understood the course materials well, did the labeling as human graders. The entire labeling is an iterated process, two graders working first respectively to give only a final grade to each student's answer, then adding grades to all rubric points, and in the end consolidating two graders' labels into an agreed version. For the selected few-shot samples, the human graders also give the text feedback to justify their grading. This process lasted about two semesters.

The two human graders are first trained by the instructor on how to do grading specifically for assignments or exams for this course. Then they label the data in two steps. In step one, they do the labeling respectively. For each evaluation question on a problem, they check to what extent a student's response answers the question. If it answers the question completely correctly, then label it with 1; otherwise, label it with 0. We do not consider partial credits since the evaluation has been decomposed into a set of questions, each of which is focused on one knowledge point. We original start with only the one final grade for each answer. Along with the development of the approach, the graders are instructed to add labels to all the rubric points for a problem. Then in step two, under the guidance of the instructor, the two human graders identify all the labels that they do not agree with each other, have a discussion, and come across the labels they all agree. Eventually, this process gives us the ground-truth labels for evaluation.

D. Characteristics and Statistics

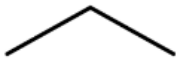
The collected data contain a total of 176 samples. Due to one empty entry, the number of valid samples is 175. The average length of a student's answer is around 39 words. Each answer, which is a paragraph, contains around 2 sentences on average. This is consistent with the normal description of short answers such as the length is "phrases to three to four sentences" or "a few words to approximately 100 words" [43].

To facilitate the human graders, the instructor provides one reference answer and a grading rubric which contains 4 key rubric points such as O/OH and H-Bonds. The score of each rubric point is 1. This leads to 4 points total as the full score for the problem. Originally part b in the exam is 5 points. The instructor adjusted it to be 4 points based on the rubric. Accordingly, the score on each rubric point is binary (0/1) and the score of each student is an integer value in the range of 0-4 inclusive. The following are the reference answer and a sample student answer:

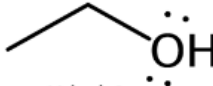
Reference answer: *Molecule 1 consists entirely of C and H atoms. This makes molecule 1 entirely non-polar and therefore very hydrophobic. Molecule 3 has an O atom which can form hydrogen bonds, making it polar and hydrophilic.*

Sample student answer: *Molecule 1: No lone pairs, No special hydrogens therefore hydrophobic.*

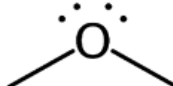
Question 2: Hydrophobic & Hydrophilic (14 points)
Consider the following three molecules:



Molecule 1



Molecule 2



Molecule 3

a) Rank the three molecules in order from the most hydrophobic to the most hydrophilic. Note that they are not amino acids. Fill in the circle that corresponds to the right answer. (9 pts.)

| Most Hydrophobic | | Most Hydrophilic |
|-------------------------------------|------------|------------------|
| a) <input type="radio"/> Molecule 1 | Molecule 2 | Molecule 3 |
| b) <input type="radio"/> Molecule 1 | Molecule 3 | Molecule 2 |
| c) <input type="radio"/> Molecule 2 | Molecule 3 | Molecule 1 |
| d) <input type="radio"/> Molecule 2 | Molecule 1 | Molecule 3 |
| e) <input type="radio"/> Molecule 3 | Molecule 2 | Molecule 1 |
| f) <input type="radio"/> Molecule 3 | Molecule 1 | Molecule 2 |

b) In the space below, briefly explain why the molecule you indicated as most hydrophobic is more hydrophobic than the molecule in the middle of the three. (5 pts)

Reference Answer:
Molecule 1 consists entirely of C and H atoms. This makes molecule 1 entirely non-polar and therefore very hydrophobic. Molecule 3 has an O atom which can form hydrogen bonds, making it polar and hydrophilic.

Rubric:

- O/OH (1 point)
- H-Bonds (1 point)
- CH (1 point)
- Non-Polar (1 point)

Fig. 3. The problem, the reference answer, and the rubric in the dataset.

Molecule 2: Two lone pairs, has special hydrogen therefore more hydrophilic than molecule 1

V. EXPERIMENTS AND RESULTS

In this section, we describe our experiment settings and report the experiment results.

A. Experiment Settings

In the R-RAG module, the instructor-provided reference answer and the rubric are both supplied. Answer-conditioned question generation is applied to the reference answer, in which one rubric point is set as a conditioned answer to generate one question. To make sure the generated questions are of high quality then we can have a more consistent and solid evaluation of LLM’s performance, we manually generate three questions for each rubric point based on the reference answer. The course instructor reviews these questions and selects the best one out of the three. In the Evaluation Module, the system calls GPT4 API (the version of GPT4-Turbo-Preview). When prompting GPT4, we design general instruction and question-specific instruction. The general instruction is to specify the role, task, detailed instruction, and constraints on how to grade such as the grade scale, criteria of each grade, etc. The question-specific instruction is to address a grader’s personal criteria. For example, in the evaluation question “What does molecule 1 consist of?”, although the reference answer expects a student’s answer to contain the information that molecule 1 consists of C and H atoms, the course instructor thinks if a student’s answer only mentions C (carbon) atom, it’s also considered as being correct. This personal criteria is addressed in the question-specific instruction. We apply few-shot learning in which the 4-shot gives us the best performance. To select samples, we perform random selection. Selected samples are excluded from evaluation. The following shows an example of instructions in the prompt:

General instruction: You are the instructor of a college-level Introductory Biology course. You are going to grade the exam for this course. Your grading should be based on the question asked, the full-credit answer, the student’s answer, and nothing else. Give the binary score 1 or 0, in which 1 means the student’s answer is correct and 0 means the student’s answer is incorrect or does not answer the question, and justify your grading.

Question-specific instruction: As long as the answer mentions or implies that the molecule contains just carbon, it should be considered as being correct and graded as 1.

B. Evaluation Results

Stella essentially takes the role of a grader. Thus We evaluate the results by calculating the agreement with the human grader’s grading, which is commonly used in grading evaluation. Because this work is pioneering in applying QA-based evaluation on ASAG task, on a newly collected real-world dataset, and this field is relatively new, we weren’t able to find highly related models or systems to compare with. As explained in the labeling process section, under the instructor’s supervision, the two human graders discussed the difference in the grades they assigned to the same questions, reached an agreement, and reassigned the agreed grades to those questions as the ground-truth labels. We compare the agreement between the results from our system and the ground-truth labels and report both Cohen’s Kappa coefficient (κ) [44] and Raw Agreement (Accuracy).

Agreement Results. As shown in Table I, Cohen’s Kappa coefficient value between the human grader and the ground-truth labels reaches 0.8315 which is normally accepted as a

near-perfect agreement. Although our system still does not reach human performance, it achieves a substantial agreement with the ground-truth labels by $\kappa = 0.6720$. As for the raw agreement, it's about 8% lower than the human grader. These results show that our system is promising in automatic grading while maintaining high accuracy.

TABLE I
AGREEMENT RESULTS BETWEEN THE SYSTEM AND LABELS

| | Cohen's Kappa | Raw Agreement |
|------------|---------------|---------------|
| Human | 0.8315 | 0.9157 |
| Our System | 0.6720 | 0.8358 |

Human Evaluation on Feedback. In order to further understand the generated text from LLM which is to justify the grading, we did a human evaluation of all the justifications generated by GPT4. The two human graders are instructed to do the evaluation. In human evaluation, the question we ask is how relevant the justification generated by GPT4 is to support its grading. In other words, if the grade assigned by GPT4 is correct or incorrect, does the justification support this grading? The data we use for human evaluation is from a 6-shot learning experiment setting which leaves a total of 169 samples for evaluation. Since 4 evaluation questions are generated for the problem, there is a total of 676 GPT4 responses to be evaluated. Very surprisingly, only 1 response is evaluated to be irrelevant to the numeric grade. Even when the grading of GPT4 is incorrect, it's usually still based on the relevant facts but with too much or not enough inference which will be shown in the sample results analysis in the following. This shows that GPT4 does do the grading based on the relevant facts which increases the confidence in using an application based on it.

C. Sample Grading and Feedback Analysis

In Table II, we list three sample students' responses and GPT4 grading results to the evaluation questions. We have several findings about using GPT4 to do grading as:

- GPT4 is good at identifying relevant facts or statements. For example, in Q1 and Q2 to the student response 9328795, GPT4 is able to identify that molecule 3 has an Oxygen atom and can form hydrogen bonds even though the two phrases are a bit far from each other in the original text answer. In student response 9328809, GPT4 identifies question-related information that molecule 3 has an O atom and it cannot form H-bonds and then grades the student response on Q1 is correct and on Q3 is incorrect.
- GPT4 can be tolerant of some typos in the input. For example, in student response 9328795, there are typos or errors such as *tho* and *then*. But they do not affect GPT4's understanding of the response text.
- GPT4 sometimes can infer the meaning of the text properly, while sometimes infers too much implication from

the given text. For example, in Q3 to the student response 9328790, based on "Molecule 1 is most hydrophobic because it is all carbons and it can't make hydrogen bonds.", GPT4 properly infers that the student implies molecule 1 consists of carbon atoms and does not contain elements like oxygen or nitrogen which can form hydrogen bonds, and further grades it as being correct on this evaluation question. While in Q4 to the student response 9328809, GPT4 interprets the student's statement "Molecule 1 does not have donor or acceptor" as suggesting that molecule 1 is non-polar and grades it as being correct which is actually incorrect. In this example, GPT4's interpretation might be true in general. However, it infers too much from the student's response in this specific problem, in which the instructor tries to test the concept of non-polar. We notice this is a type of error that GPT4 is prone to make in this grading task. This error type shows that, since LLM such as GPT is trained on massive data which is expected to have learned a large amount of general knowledge, how to ground it to some specific task and some specific domain is a big challenge. Our methods of R-RAG and structured evaluation provide an approach to address this issue. We also experimented with prompting engineering to set some constraints, such as defining the role to be a college-level Biology instructor and explicitly asking GPT4 to do the grading based only on the student's response, the evaluation question, the reference answer to the question, and nothing else. However, we find it's still hard to eliminate such error types by refining the prompts only.

- Error cases of Q1 and Q2 in the student response 9328790 show the complexity of the grading task. Due to the student not giving any statements about molecule 3, GPT4 grades the response to be incorrect on these two questions which are both about molecule 3. However, the human grader is more focused on the concept that the most hydrophilic molecule has an OH which makes it able to form H-Bonds. Based on this, although the student discusses molecule 2 instead of molecule 3, the response shows he/she indeed understands the concept correctly. Accordingly, human graders give the student full credit on these two questions. During the human evaluation process, the course instructor and two human graders all agree that, in such cases, GPT4 does the job properly based on the instructions it's given. The challenge lies not only in how to make an LLM understand the abstract concept behind the text, but also in how to formulate what is examined in a problem in the learning process itself.

D. Ablation Study

We did the following ablation studies to show the effect of some parameters and settings. Due to the time and cost constraints, the following experiments were done using GPT-4.

Effect of Clustering. As shown in Figure 4, applying a clustering algorithm to select samples for few-shot learning

TABLE II
EXAMPLE GRADINGS AND FEEDBACK

| Question | GPT4's Grading | Label |
|---|---|-------|
| Q1: Does molecule 3 have Oxygen (O) atom or OH Group? Q2: Can molecule 3 form H-Bonds or Hydrogen Bonds? Q3: What does molecule 1 consist of? Q4: Is molecule 1 entirely non-polar? | | |
| Student response 9328795: Molecule 1 is the most hydrophobic because it is a carbon chain tho does not have the ability to create hydrogen or ionic bonds. Whereas Molecule 3 is more hydrophilic then molecule 1 due to its ability to form hydrogen bonds because of the lone pairs on the oxygen atom. | | |
| Q1 | The student's score is 1. The student answers the question correctly. The student states, "Molecule 3 is more hydrophilic then molecule 1 due to its ability to form hydrogen bonds because of the lone pairs on the oxygen atom." <i>This indicates that molecule 3 has an Oxygen atom</i> , which is consistent with the full-credit answer. Accordingly, the student answers the question correctly. | 1 |
| Q2 | The student's score is 1. The student answers the question correctly. The student's answer states, "Molecule 3 is more hydrophilic then molecule 1 due to its ability to form hydrogen bonds because of the lone pairs on the oxygen atom," <i>indicating that molecule 3 can form hydrogen bonds</i> . This aligns with the full-credit answer which states "Molecule 3 has an O atom which can form hydrogen bonds". Accordingly, the student answers the question correctly. | 1 |
| Q3 | The student's score is 1. The student answers the question correctly. The student states, "Molecule 1 is the most hydrophobic because it is a carbon chain," <i>implying it consists of Carbon</i> . Moreover, the student says that molecule 1 "does not have the ability to create hydrogen or ionic bonds," which <i>indirectly suggests the presence of Hydrogen</i> . Both of these components are consistent with the full-credit answer, "Molecule 1 consists entirely of C and H atoms." Accordingly, the student answers the question correctly. | 1 |
| Q4 | The student's score is 0. The student's answer is not relevant to the question. Even though the student discusses the hydrophobic nature of molecule 1, they <i>do not answer directly if molecule 1 is entirely non-polar</i> . Accordingly, the student's answer is not relevant to this question. | 0 |
| Student response 9328790: Molecule 1 is most hydrophobic because it is all carbons and it can't make hydrogen bonds. But Molecule #2 has an OH at the end, allowing it to create h-bonds. | | |
| Q1 | The student's score is 0. The student's answer is not relevant to the question. <i>The student's answer does not include any information about molecule 3. Instead, the answer discusses molecules 1 and 2</i> . Accordingly, the student answer is not relevant to the question. | 1 |
| Q2 | The student's score is 0. The student does not answer the question. <i>The student provides information about molecules 1 and 2, but does not mention whether or not molecule 3 can form hydrogen bonds</i> . Accordingly, the student does not answer the question. | 1 |
| Q3 | The student's score is 1. The student answers the question correctly. The student states, "Molecule 1 is most hydrophobic because it is all carbons and it can't make hydrogen bonds." <i>This implies that molecule 1 consists of carbon atoms and does not contain elements like oxygen or nitrogen which can form hydrogen bonds</i> . This is consistent with the full-credit answer "Molecule 1 consists entirely of C and H atoms." Accordingly, the student answers the question correctly. | 1 |
| Q4 | The student's score is 0. The student's answer does not directly address the question asked. <i>While the student correctly identifies molecule 1 as being hydrophobic and incapable of forming hydrogen bonds, they do not explicitly state that this makes molecule 1 entirely non-polar</i> . The student also brings in comparison with other molecules (Molecule #2), which is not relevant to the specific question asked. Therefore, the answer is incorrect. | 0 |
| Student response 9328809: Molecule 1 does not have donor or acceptor. It is probably a covalent bond, which is the strongest bond. Molecule 2 has a -OH bond, which makes it partly hydrophilic, but more hydrophobic. Molecule 3 only has an O atom, which is an acceptor, but cannot form H-bonds. | | |
| Q1 | The student's score is 1. The student answers the question correctly. The student states, "Molecule 3 only has an O atom, which is an acceptor, but cannot form H-bonds." <i>This indicates that molecule 3 has an O atom</i> , which aligns with the full-credit answer, "Molecule 3 has an O atom". Accordingly, the student answers the question correctly. | 1 |
| Q2 | The student's score is 0. The student answers the question incorrectly. The student's answer "Molecule 3 only has an O atom, which is an acceptor, but cannot form H-bonds" contradicts the full-credit answer "Molecule 3 has an O atom which can form hydrogen bonds". <i>An oxygen atom in a molecule can be a hydrogen bond acceptor, meaning that it can form a hydrogen bond. Therefore, the student's understanding of the concept is incorrect</i> . | 0 |
| Q3 | The student's score is 0. The student's answer is not relevant to the question. <i>The student mentions various properties of molecule 1, but does not state what molecule 1 consists of in terms of atoms</i> . Accordingly, the student's answer is not relevant to the question. | 0 |
| Q4 | The student's score is 1. The student answers the question correctly. <i>The student's response includes "Molecule 1 does not have donor or acceptor", which can be interpreted as suggesting that molecule 1 is non-polar</i> , aligning with the full-credit answer. Thus, the student correctly answers the question. | 0 |

consistently improves Cohen's Kappa coefficient compared with that without using clustering, e.g., about 0.2 increments in the κ value under one-shot. This supports the effectiveness of the clustering approach in selecting learning samples that are expected to better represent the distribution of the data, and further empower the capability of the LLM such as GPT4 on this specific dataset and task.

Number of Shots. We experimented with different shot numbers. The Cohen's Kappa coefficient values in Figure 4 show that a few learning samples can significantly improve

the performance of a general LLM on a specific task such as grading. Under the setting with clustering, the 4-shot gives the best result which is significantly higher than the 3-shot while slightly higher than the 5-shot. Under the setting without clustering, the performance under the 6-shot is significantly better than the 1-shot, while the 10-shot does not show much further improvement compared with the 6-shot. This is consistent with the common understanding that the few-shot in-text learning can guide a general LLM toward a specific task such as grading in this experiment. Meanwhile, the effect

declines when reaching a reasonable shot number.

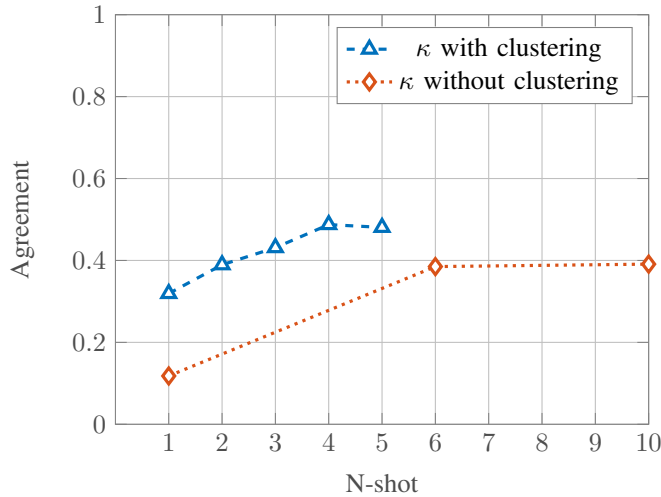


Fig. 4. Effect of shot number.

VI. CONCLUSION AND FUTURE WORK

We propose SteLLA, an automatic short-answer grading system that uses RAG techniques based on the instructor-provided reference answer and rubric to facilitate an LLM performing structured question-answering-based assessment of student responses. Experiments on a real-world dataset show that our system is able to achieve substantial agreement with the human graders. It can also provide analytical grades and feedback on knowledge points examined in the problem. In the future, one direction of the work could be on generating structured evaluation question-answer pairs in the context of missing rubrics, i.e., only the reference answer available. Another direction could be to add human-interactive components to increase the system's adaptability in personalization.

ACKNOWLEDGMENT

This paragraph is omitted due to the blind review policy.

REFERENCES

- [1] T. Phung, J. Cambronero, S. Gulwani, T. Kohn, R. Majumdar, A. Singla, and G. Soares, "Generating high-precision feedback for programming syntax errors using large language models," in *Proceedings of the 16th International Conference on Educational Data Mining*, M. Feng, T. Käser, and P. Talukdar, Eds. Bengaluru, India: International Educational Data Mining Society, July 2023, pp. 370–377.
- [2] X. Liu, S. Wang, P. Wang, and D. Wu, "Automatic grading of programming assignments: An approach based on formal semantics," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, 2019, pp. 126–137.
- [3] S. Baral, A. Botelho, A. Santhanam, A. Gurung, L. Cheng, and N. Heffernan, "Auto-scoring student responses with images in mathematics," in *Proceedings of the 16th International Conference on Educational Data Mining*, M. Feng, T. Käser, and P. Talukdar, Eds. Bengaluru, India: International Educational Data Mining Society, July 2023, pp. 362–369.
- [4] A. Botelho, S. Baral, J. A. Erickson, P. Benachamardi, and N. T. Heffernan, "Leveraging natural language processing to support automated assessment and feedback for student open responses in mathematics," *J. Comput. Assist. Learn.*, vol. 39, no. 3, pp. 823–840, 2023. [Online]. Available: <https://doi.org/10.1111/jcal.12793>
- [5] W. A. Mansour, S. Albatarni, S. Eltanbouly, and T. Elsayed, "Can large language models automatically score proficiency of written essays?" in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, Eds. Torino, Italia: ELRA and ICCL, May 2024, pp. 2777–2786. [Online]. Available: <https://aclanthology.org/2024.lrec-main.247>
- [6] Y. Wang, C. Wang, R. Li, and H. Lin, "On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, M. Carpuat, M.-C. de Marneffe, and I. V. Meza Ruiz, Eds. Seattle, United States: Association for Computational Linguistics, Jul. 2022, pp. 3416–3425. [Online]. Available: <https://aclanthology.org/2022.naacl-main.249>
- [7] S. Basu, C. Jacobs, and L. Vanderwende, "Powergrading: a Clustering Approach to Amplify Human Effort for Short Answer Grading," *Transactions of the Association for Computational Linguistics*, vol. 1, pp. 391–402, 10 2013. [Online]. Available: https://doi.org/10.1162/tac1_a_00236
- [8] N. Süzen, A. N. Gorban, J. Levesley, and E. M. Mirkes, "Automatic short answer grading and feedback using text mining methods," *Procedia Computer Science*, vol. 169, pp. 726–743, 2020, postproceedings of the 10th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2019 (Tenth Annual Meeting of the BICA Society), held August 15-19, 2019 in Seattle, Washington, USA. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1877050920302945>
- [9] S.-Y. Yoon, "Short answer grading using one-shot prompting and text similarity scoring model," 2023.
- [10] A. H. Morris, G. M. Kasper, and D. A. Adams, "The effects and limitations of automated text condensing on reading comprehension performance," *Info. Sys. Research*, vol. 3, no. 1, p. 17–35, mar 1992. [Online]. Available: <https://doi.org/10.1287/isre.3.1.17>
- [11] J. Clarke and M. Lapata, "Discourse constraints for document compression," *Computational Linguistics*, vol. 36, no. 3, pp. 411–441, Sep. 2010. [Online]. Available: <https://aclanthology.org/J10-3005>
- [12] P. Chen, F. Wu, T. Wang, and W. Ding, "A semantic qa-based approach for text summarization evaluation," 2018.
- [13] M. Eyal, T. Baumel, and M. Elhadad, "Question answering as an automatic evaluation metric for news article summarization," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 3938–3948. [Online]. Available: <https://aclanthology.org/N19-1395>
- [14] T. Scialom, S. Lamprier, B. Piwowarski, and J. Staiano, "Answers unite! unsupervised metrics for reinforced summarization models," 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)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3246–3256. [Online]. Available: <https://aclanthology.org/D19-1320>
- [15] A. Wang, K. Cho, and M. Lewis, "Asking and answering questions to evaluate the factual consistency of summaries," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 5008–5020. [Online]. Available: <https://aclanthology.org/2020.acl-main.450>
- [16] E. Durmus, H. He, and M. Diab, "FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 5055–5070. [Online]. Available: <https://aclanthology.org/2020.acl-main.454>
- [17] T. Scialom, P.-A. Dray, S. Lamprier, B. Piwowarski, J. Staiano, A. Wang, and P. Gallinari, "QuestEval: Summarization asks for fact-based evaluation," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican

- Republic: Association for Computational Linguistics, Nov. 2021, pp. 6594–6604. [Online]. Available: <https://aclanthology.org/2021.emnlp-main.529>
- [18] T. Scialom, L. Martin, J. Staiano, Éric Villemonte de la Clergerie, and B. Sagot, “Rethinking automatic evaluation in sentence simplification,” 2021.
- [19] C. Rebuffel, T. Scialom, L. Soulier, B. Piwowarski, S. Lamprier, J. Staiano, G. Scouteeten, and P. Gallinari, “Data-questeval: A referenceless metric for data-to-text semantic evaluation,” 2021.
- [20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2019.
- [21] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,” in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.
- [22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2023.
- [23] M. Cao, Y. Dong, J. Wu, and J. C. K. Cheung, “Factual error correction for abstractive summarization models,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 6251–6258. [Online]. Available: <https://aclanthology.org/2020.emnlp-main.506>
- [24] V. Raunak, A. Menezes, and M. Junczys-Dowmunt, “The curious case of hallucinations in neural machine translation,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds. Online: Association for Computational Linguistics, Jun. 2021, pp. 1172–1183. [Online]. Available: <https://aclanthology.org/2021.naacl-main.92>
- [25] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, “Survey of hallucination in natural language generation,” *ACM Computing Surveys*, vol. 55, no. 12, p. 1–38, Mar. 2023. [Online]. Available: <http://dx.doi.org/10.1145/3571730>
- [26] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS ’20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [27] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, “Dense passage retrieval for open-domain question answering,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 6769–6781. [Online]. Available: <https://aclanthology.org/2020.emnlp-main.550>
- [28] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, “Realm: retrieval-augmented language model pre-training,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML’20. JMLR.org, 2020.
- [29] G. Izacard and E. Grave, “Leveraging passage retrieval with generative models for open domain question answering,” in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, P. Merlo, J. Tiedemann, and R. Tsarfaty, Eds. Online: Association for Computational Linguistics, Apr. 2021, pp. 874–880. [Online]. Available: <https://aclanthology.org/2021.eacl-main.74>
- [30] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. van den Driessche, J.-B. Lespiau, B. Damoc, A. Clark, D. de Las Casas, A. Guy, J. Menick, R. Ring, T. Hennigan, S. Huang, L. Maggiore, C. Jones, A. Cassirer, A. Brock, M. Paganini, G. Irving, O. Vinyals, S. Osindero, K. Simonyan, J. W. Rae, E. Elsen, and L. Sifre, “Improving language models by retrieving from trillions of tokens,” 2022.
- [31] U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, and M. Lewis, “Generalization through memorization: Nearest neighbor language models,” 2020.
- [32] J. He, G. Neubig, and T. Berg-Kirkpatrick, “Efficient nearest neighbor language models,” 2021.
- [33] S. Burrows, I. Gurevych, and B. Stein, “The eras and trends of automatic short answer grading,” *International Journal of Artificial Intelligence in Education*, vol. 25, pp. 60–117, 2015. [Online]. Available: <https://api.semanticscholar.org/CorpusID:5917679>
- [34] J. Burstein, R. Kaplan, S. Wolff, and C. Lu, “Using lexical semantic techniques to classify free-responses,” in *Breadth and Depth of Semantic Lexicons*, 1996. [Online]. Available: <https://aclanthology.org/W96-0304>
- [35] C. Leacock and M. Chodorow, “C-rater: Automated scoring of short-answer questions,” *Computers and the Humanities*, vol. 37, pp. 389–405, 2003. [Online]. Available: <https://api.semanticscholar.org/CorpusID:27443635>
- [36] M. Mohler and R. Mihalcea, “Text-to-text semantic similarity for automatic short answer grading,” in *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, A. Lascarides, C. Gardent, and J. Nivre, Eds. Athens, Greece: Association for Computational Linguistics, Mar. 2009, pp. 567–575. [Online]. Available: <https://aclanthology.org/E09-1065>
- [37] T. Mitchell, T. Russell, P. Broomhead, and N. Aldridge, “Towards robust computerised marking of free-text responses,” 2002. [Online]. Available: <https://api.semanticscholar.org/CorpusID:17936736>
- [38] L. F. Bachman, N. Carr, G. Kamei, M. Kim, M. J. Pan, C. Salvador, and Y. Sawaki, “A reliable approach to automatic assessment of short answer free responses,” in *COLING 2002: The 17th International Conference on Computational Linguistics: Project Notes*, 2002. [Online]. Available: <https://aclanthology.org/C02-2023>
- [39] S. Bailey and D. Meurers, “Diagnosing meaning errors in short answers to reading comprehension questions,” in *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, J. Tetreault, J. Burstein, and R. De Felice, Eds. Columbus, Ohio: Association for Computational Linguistics, Jun. 2008, pp. 107–115. [Online]. Available: <https://aclanthology.org/W08-0913>
- [40] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günemann, E. Hüllermeier, S. Krusche, G. Kutyniok, T. Michaeli, C. Nerdel, J. Pfeffer, O. Poquet, M. Sailer, A. Schmidt, T. Seidel, M. Stadler, J. Weller, J. Kuhn, and G. Kasneci, “Chatgpt for good? on opportunities and challenges of large language models for education,” *Learning and Individual Differences*, vol. 103, p. 102274, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1041608023000195>
- [41] N. Reimers and I. Gurevych, “Sentence-bert: Sentence embeddings using siamese bert-networks,” 2019.
- [42] J. Schneider, B. Schenk, C. Niklaus, and M. Vlachos, “Towards llm-based autograding for short textual answers,” 2023.
- [43] J. Sukkariyah and S. Stoyanchev, “Automating model building in c-rater,” in *Proceedings of the 2009 Workshop on Applied Textual Inference (TextInfer)*, C. Callison-Burch, I. Dagan, C. Manning, M. Pennacchiotti, and F. M. Zanzotto, Eds. Suntec, Singapore: Association for Computational Linguistics, Aug. 2009, pp. 61–69. [Online]. Available: <https://aclanthology.org/W09-2509>
- [44] J. Cohen, “A Coefficient of Agreement for Nominal Scales,” *Educational and Psychological Measurement*, vol. 20, no. 1, p. 37, 1960.