# Analysis of Plagiarism via ChatGPT on Domain-Specific Exams

Jinyoung Jo*, Sean Choi†
*University of California, Los Angeles
jinyoungjo@ucla.edu
†Santa Clara University
sean.choi@scu.edu

***Abstract*—This work presents a case study, linguistic analysis and potential prevention methods on the use of large language models (LLM) for generating solutions for exams on cloud computing course that require domain-specific knowledge. The study involves analyzing the responses of three groups of students: a group who used ChatGPT to plagiarize solutions, another group who referred to external non-LLM resources (e.g., web search) to plagiarize solutions, a control group who generated solutions without any external assistance. Results show that solutions from groups that participated in plagiarism tend to be lengthy, use uncommon words, and are similar to each other compared to human-generated solutions. This study not only shows that it is possible to generate legitimate solutions for exams that require extensive domain-specific knowledge using ChatGPT, but also shows some potential signals one can use to detect plagiarism, thus providing potential of promoting academic integrity by curbing unethical use of AI in academic settings.**

***Keywords*—Large Language Model, Academic Integrity, Computer Science Education, Plagiarism Detection**

## I. Introduction

Large-Language Model (LLM) is a term used for deep neural network designed to represent the human language, often consisting of billions of parameters and are trained on a huge amount of textual data. Performances of various LLMs have improved vastly over the last decade, now excelling in tasks such as question answering, semantic search and summarizing text. In particular, a company called OpenAI recently released an application called ChatGPT [1], [2] that is powered by a large language model, and it has gained huge popularity due to its ease of use and excellent performance in text/code generation, summarization and question answering. In fact, ChatGPT now holds the record for fastest-growing user base of any application in history with 100 million users in just two month [3].

However, ChatGPT's sudden gain in popularity came with a side effect that the academic community is struggling to react to: violation of academic integrity. Since ChatGPT can easily generate a seemingly unique text based on user query, many academic institutions are reporting violations of academic integrity by using ChatGPT to generate solutions for homework, essays and even timed exams. Therefore, to understand the effectiveness of ChatGPT and to recommend signals that can be used to detect the use of ChatGPT, this study presents a set of findings from analyzing students' exam responses in a real-world class setting.

In a particular computer science class that require deep background knowledge, the teaching staff discovered that some students resorted to using ChatGPT to generate answers to multiple exam questions. This study compares the responses of a group of students from that class who used ChatGPT to generate answers to those of a control group who wrote their answers without using LLM with respect to various linguistic properties. Responses of the students who were found to have relied on contents from a search engine were also included for comparison. This work is a first step towards understanding the linguistic characteristics of advanced large language model as compared to human language and providing suggestions for potential signals to discern text generated by a human versus text generated by large language models. Using such suggestions, the goal of this work is to encourage academic integrity and to minimize the impact of large language models in reducing the effectiveness of the computer science education.

In summary, the main contributions of this paper are as follows:

- Feasibility of ChatGPT in generating solutions for questions that require deep background knowledge.
- Understanding the linguistic characteristics of text generated by large language models.
- Providing potential signals for detecting violation of academic integrity.

The paper is structured as follows. First, the paper presents the background on large language models, cloud computing concepts and metrics used for language characterization (Section II), followed by an overview of the course, examination given to the students and types of plagiarisms observed (Section III). Then the paper presents the outcome of the evaluations (Section IV) and related work (Section V). It concludes by discussing planned future directions (Section VI).

## II. Background

This section provides relevant background information and the context of this work, which are large language models, cloud computing and language characteristic metrics.

### A. Large Language Model & ChatGPT

Language models have become an essential tool for a wide range of natural language processing (NLP) tasks, such as machine translation, text summarization, and question answering. Recent advancements in deep learning techniques have led to the development of large-scale language models that

979-8-3503-2759-5/23/$31.00 ©2023 IEEE
DOI 10.1109/CSCE60160.2023.00171

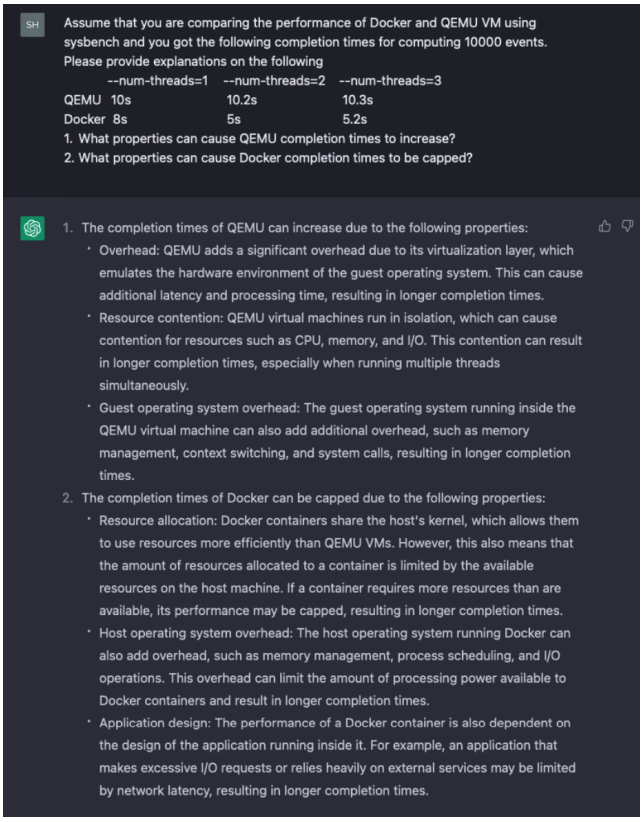


Figure 1: ChatGPT interface solving an exam question.

can generate human-like text and perform a wide range of language-related tasks. One such model is ChatGPT [1], [2], a large-scale generative language model trained by OpenAI. ChatGPT is based on the transformer architecture, which was introduced as an alternative to traditional recurrent neural networks (RNNs) for sequence modeling tasks. The transformer architecture consists of an encoder and a decoder, each composed of multiple layers of self-attention and feed-forward neural networks. The encoder and decoder work together to process an input sequence and generate an output sequence. The encoder first processes the input sequence and produces a set of context vectors, which are then used by the decoder to generate the output sequence. The self-attention mechanism allows the model to capture long-range dependencies between words in the input sequence, making it particularly effective for language modeling tasks.

ChatGPT has been trained on a massive corpus of text data, consisting of over 8 million unique documents, or 45 terabytes in size, from a diverse range of sources, including web pages, books, and online forums. In addition to using massive amount of training data, ChatGPT is one of the largest language models; ChatGPT version 3.5 consists of 175 billion parameters and ChatGPT version 4 consists of 100 trillion parameters, which translates to about 800GB and 500TB of model size, respectively. ChatGPT has been shown to be effective for a wide range of NLP tasks, attributed to the large size, including language modeling, which involves the model predicting the next word in a sequence given the previous words, question answering, as well as generation of text, code and dialog. In particular, ChatGPT is widely renowned as a highly sophisticated chat bot that engages in human-like conversations with users. Figure 1 shows an example of ChatGPT user interface that solves an exam question, which involves domain-specific knowledge.

### *B. Cloud Computing and Virtualization*

The main topic of the course analyzed in this study is cloud computing, thus this section is to provide high-level context into cloud computing to help understand the exam questions and solutions. Cloud computing refers to the practice of using remote servers accessed over the internet to store, manage, and process data, instead of using local servers or personal computers. Cloud computing enables businesses and individuals to leverage the power of large-scale computing resources, which can be quickly scaled up or down based on demand, to meet their computing needs. It also allows for greater flexibility and reliability compared to traditional computing models.

Main driving technology behind cloud computing is virtualization. Virtualization is a technology to create virtual representations of servers, storage, networks, and operating systems using software [4]. Virtualization has revolutionized how software applications are managed and deployed, making it easier and more efficient to provision, scale, and manage applications at a large scale. Two pivotal virtualization technologies are virtual machines (VMs) and containers.

A virtual machine is an emulation of a physical machine that runs on top of a hypervisor, a piece of software that allows multiple VMs to share the same physical resources. Each VM has its own operating system, software stack, and resources, including CPU, memory, and disk. VMs are typically isolated from each other and from the underlying host system, providing a high degree of security and flexibility. Some examples of VMs are QEMU [5] and VirtualBox [6]. VMs offer several advantages, including the ability to run multiple operating systems and applications on a single physical machine, the ability to easily migrate VMs between different hosts, and the ability to provision and scale resources dynamically.

A container is a lightweight and portable package that includes an application and its dependencies, but shares the same kernel and resources as the host system. Containers are isolated from each other using namespaces and cgroups [7], which provide a similar level of isolation as VMs, but with lower overhead and faster startup times. Some examples of containers are Docker [8] and Kata [9]. Containers offer several advantages, including improved performance and scalability, faster deployment and startup times, and easier management and orchestration. Containers are also portable and can be easily moved between different hosts and environments.

In terms of architecture and performance, VMs are more complex and heavyweight than containers, requiring a full operating system and virtual hardware emulation. Containers, on the other hand, are lightweight and share the same kernel and resources as the host system. Thus, containers have lower overhead and faster startup times than VMs, making them more efficient for deploying and scaling applications.

### C. Metrics for Language Characterization

This study analyzes and compares the linguistic properties of three types of responses to the exam questions, ChatGPT-aided solutions, search engine-aided solutions and "valid" answers that were generated without any aid from external references, with respect to the following metrics: proportion of stop words, length of responses as measured by the number of characters, words, and sentences, sentence length, type-token ratio as a proxy for lexical diversity, word frequency, use of *I*, Automated Readability Index, Jaccard index [10] and cosine similarity of SBERT [11] encodings. The two text similarity measures, namely the Jaccard index and cosine similarity, are calculated for each pair of responses that serve as answers to the same question. All other measures are calculated for each response.

- Proportion of stop words: Stop words are a set of words that carry little meaning; for example, in English, *the*, *is* and *and* can be classified as stop words. The evaluation uses stop words corpus from the Natural Language Toolkit package [12] of Python. The proportion of stop words is calculated as the number of stop words divided by the total number of words of a response. Stop words are excluded when calculating the number of characters and words, as well as sentence length, type-token ratio, and mean word frequency.
- Length of answers: As measures of answer length, the following metrics are calculated: (i) the number of characters, (ii) the number of words and (iii) the number of sentences contained in an answer. As will be discussed in Section III-B, we observed that increased length of answers was one of the signals of plagiarism.
- Length of sentences: This metric is the average number of words contained in a sentence.
- Lexical diversity: Type-token ratio (TTR) was used as a proxy for lexical diversity. TTR is calculated as type frequency of a response (the number of unique words) divided by token frequency (the total number of words). A higher TTR indicates that the text has more diverse vocabulary.
- Word frequency: This metric represents the mean frequency of words present in each solution. The frequency of each word is obtained from the SUBTLEX-us [13], frequency data based on a corpus of American English film subtitles. Higher word frequency means that the word is more commonly used. As will be noted in Section III-B, use of sophisticated vocabulary may be a signal of plagiarism.
- Proportion of *I*: One of an informal observation was that valid answers more frequently use phrases that include the first person singular pronoun *I*, e.g. *I think*, *I would*. Thus, another metric used is the number of times *I* appears in each answer, normalized to the total number of words.
- Automated Readability Index (ARI): ARI [14] is an index of readability or understandability of a text. It is calculated as $ARI = 0.5 * ASL + 4.71 * AWL - 21.43$, where $ASL$ stands for average sentence length (average number of words in a sentence) and $AWL$ stands for average word length (average number of characters in a word).
- Text similarity: It is plausible to think that solutions that violated academic integrity, either aided by ChatGPT or a search engine, should share many words in common. To test this hypothesis, two measures of text similarity were collected: the Jaccard index and cosine similarity of an embedding generated from a language model. The Jaccard index is calculated as the number of unique words common to two texts divided by the total number of unique words in both texts. The cosine similarity is a measure of similarity between two vectors, as it holds a unique property where the cosine similarity only considers the angle between the vectors, not their magnitude. Given this definition, two orthogonal vectors have a similarity of 0 and two proportional vectors have a similarity of 1. The cosine similarity between two vectors $A, B$ is computed as:

$$\cos(\theta) = \frac{A \cdot B}{\|A\|_2 \|B\|_2}$$

  In order to compute the cosine similarity, SBERT model is used to generate an embedding, which is a technique where individual words are represented as real-valued vectors. Then, the similarity values reported in Section IV are computed between two embedding vectors generated from two distinct sets of words. Within each group of students, we obtained text similarity measures for every pair of responses of the same question, as calculating similarity between responses of different questions would trivially result in low similarity and holds little scientific value.

## III. Overview

This section provides a comprehensive analysis of the course and examination structure, specifically focusing on the various types of questions and responses that are incorporated within the exam. In addition, this section delves into the intricacies of plagiarism as it pertains to the aforementioned responses.

### A. Course & Exam Format

The course in question is entitled "Introduction to Cloud Computing", offered at Santa Clara University under the course code COEN 241 [15]. COEN 241 is a graduate-level course that presupposes completion of an undergraduate curriculum with prerequisites in computer networks and operating systems. The course typically enrolls between 35 to 40 students, with the average final grade of the class being a B+. The course primarily emphasizes project-based learning, with the final project constituting 50% of the total grade, whereas a single exam accounts for 25% of the total grade.

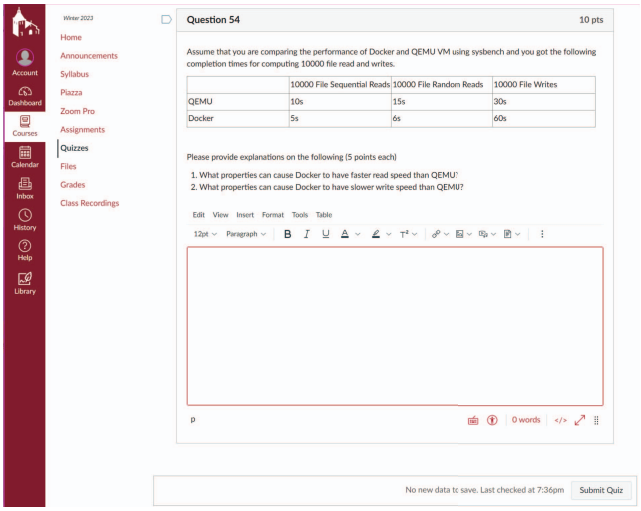


Figure 2: Student's view of the exam on Canvas.

The exam, which is the main topic of the study, is given during the sixth week of the ten-week curriculum. The exam is administered via an online platform known as Canvas [16] and it is expected that the students taking the exam are also present on a Zoom [17] meeting throughout the exam. An example view of the exam in Canvas from the student's point of view is shown in Fig. 2. The exam is composed of 50 to 60 questions, comprising both multiple-choice and two open-ended essay questions. The topics covered within the exam are predominantly concerned with various forms of virtualization, such as system, operating system, and application virtualization as provided in Section II. The exam has a duration of two hours, and once submitted, no further modifications to the solutions are accepted.

As shown in Fig. 2, the exam format is quite generic and does not enforce special rules or usage of tools that put restrictions on students' computers while they are taking the exam. For example, there are tools such as LockDown Browser [18] that prohibit students from navigating away from the page while they are taking the exam. The rationale behind such an approach is to promote an open-book assessment environment that empowers students to refer to any class materials they may require. Also, the students are expected to comply with the academic integrity pledge outlined in their institution's honor code. This unbound nature, however, creates an opportunity for students to easily and freely seek solutions from online resources and language models, which in turn significantly increases instances of plagiarism, as we discuss in the subsequent section.

### *B. Types of Responses & Plagiarism*

Detecting instances of plagiarism in an academic setting remains a critical concern, particularly with regard to the free-form essay section of exams as the multiple-choice format may not present clear indications of duplication. Nevertheless, there exist certain informal indicators that may assist in the identification of plagiarism. Such signals, while seemingly trivial, can prove instrumental in discerning instances of academic dishonesty. A few examples of these signals include:

- Similarities in the contents between solutions
- Solutions discussing topics, often incorrectly, that were not taught in the class
- Hard-to-replicate errors, such as spelling, that are shared in multiple solutions

While these signals are widely used to detect plagiarism in prior classes, more recent iteration of the course presented a new set of interesting signals of plagiarism that were found in some subset of the solutions, which are:

- Increased sophistication of the language
- Increased length of the solutions
- Increased correctness of the solutions that discuss topics not covered in the class
- Similarities in solution formats, such as bullet points and paragraphs/line breaks

To facilitate a comparative analysis, one may contrast the solutions that received no help (1) versus solutions that been plagiarized via ChatGPT (2) with an illustrative instance of a sample question: ***"What properties can cause Docker to have slower write speed than QEMU?"*** Also, note that both solutions received full credit on the question.

(1) *One potential reason is that Docker has write policies of AUFS, which means that Docker need to copy up those files into its container layer for the first write to some files. In this case, there is an overheating for Docker writing.*

(2) *QEMU employs a full system virtualization approach, where a complete virtual machine is created with its own operating system. This approach provides better isolation and security but also incurs additional overhead, particularly when handling I/O operations. To improve I/O performance, QEMU can leverage hardware acceleration techniques such as using the VirtIO interface to directly access the hardware. This can lead to faster write speeds when compared to Docker, which employs a copy-on-write mechanism and may require more I/O operations for write operations.*

Even at a glance, we can easily see that solution (1) is far more simple and error prone than solution (2) in terms of both types of words used and language structure. Such *discovery* (or observation) of rather trivial signals from distinct groups, which sparked this intriguing study, incited the teaching staff to delve deeper into how the students came about these groups of solutions. Through an investigation, the teaching staff was able to identify a case of academic plagiarism and determined that the perpetrator received assistance from one of two sources: (1) utilization of a search engine, such as Google, or (2) aid of a language model such as ChatGPT. The solution sets from each respective group were subsequently collated and subjected

to a detailed analysis of their linguistic characteristics, as expounded in Section II-C. The findings from evaluating the data are presented in the ensuing section (Section IV).

### C. Statistical analysis

In order to investigate whether the numerical differences among the three groups (i.e. solutions that consulted ChatGPT, those aided by online contents obtained from a search engine, and "valid" answers that consulted no outside sources) in each language measure are statistically significant, we established mixed effects linear regression models using *lmerTest* [19] package in R [20]. The response variables were each language measure presented in Section II-C and the fixed effect was GROUP with three levels (ChatGPT, Online and Valid). In all models except the ones for text similarity measures, random intercepts for STUDENT and QUESTION were also included. For these models, we used the *anova* function in R to test whether the factor GROUP significantly increases the model fit to the data by comparing two models that are in a subset relationship, i.e. a model with GROUP and one without. The results of this likelihood ratio tests are reported as chi-squares. Post-hoc pairwise comparisons of all levels of GROUP were conducted using the *emmeans* function of the *emmeans* package [21], with *p*-values adjusted for multiple comparisons using the Tukey method.

## IV. Evaluations

To analyze the differences in responses that are generated by language model versus humans, we collected the set of data described in Section IV-A and obtained the results presented in the following sections.

### A. Data Overview

There were 10 different questions that students provided solutions for and in total, the dataset consists of about 150 samples of the student responses that are split into three classes: (1) Valid: human-generated without any external references, (2) Online: human-generated from contents retrieved via a search engine, (3) ChatGPT: LLM-generated and copied over. Valid class consists of solutions that are not subject to plagiarism, whereas the other two classes are determined to be plagiarized. The solutions from class (1) consist of solutions with scores ranging from 60% to 100% to reflect a true distribution of student grades, where as class (2) and (3)'s responses mostly received 90+%. Most of the deductions from responses in class (3) were from topics that were correct, but out of the scope of the class, which may or may not be legitimate deduction depending on the classroom settings.

### B. Per Class Characteristics

Results of the comparison between the three classes are presented in Table I. First, the effect of CLASS did not significantly increase the model fit to the data of the proportion of stop words ($\chi^2(2)$=5.3, $p$=0.07), indicating that the three classes had a comparable proportion of stop words.

Next, solution length as measured by the number of characters, words and sentences generally suggest that solutions aided by ChatGPT are longest, and Valid solutions are shortest. As for the number of characters, CLASS significantly improved the model fit to the data ($\chi^2(2)$=8.5, $p<0.05$). As shown in Fig. 3, which shows the distribution of the number of characters contained in a response, a post-hoc analysis revealed that the difference in the number of characters between ChatGPT and Valid was significant ($\beta$=176.4, SE=59.6, t=2.96, $p<0.05$), while the difference between ChatGPT and Online ($\beta$=60.5, SE=42.8, t=1.41, $p$=0.34), and that of Online and Valid ($\beta$=115.9, SE=61.1, t=1.90, $p$=0.16) were non-significant. At the word level, there was a numerical trend in which ChatGPT had a greater number of words than Online, which in turn had a greater number of words than Valid. However, none of the pairwise comparisons were significant: the difference between ChatGPT and Valid missed significance ($\beta$=21.4, SE=8.65, t=2.47, $p$=0.056), and the difference between ChatGPT and Online was not significant ($\beta$=10.6, SE=6.34, t=1.67, $p$=0.22), nor was the difference between Online and Valid ($\beta$=10.8, SE=8.88, t=1.21, $p$=0.46). In terms of the number of sentences, CLASS significantly improved the model fit to the data ($\chi^2(2)$=14.3, $p<0.001$). ChatGPT had a significantly greater number of sentences than both Online ($\beta$=1.58, SE=0.48, t=3.27, $p<0.01$) and Valid ($\beta$=1.88, SE=0.66, t=2.84, $p<0.05$), while Online and Valid did not significantly differ from each other ($\beta$=0.30, SE=0.68, t=0.45, $p$=0.90).

An informal observation shows that sentences in the Valid class are shorter than those of the ChatGPT and the Online class. However, a statistical analysis of the number of words contained in a sentence shows that while this is numerically true, adding CLASS to the model did not significantly improve the fit to the data ($\chi^2(2)$=2.8, $p$=0.24).

We also examined word-level characteristics of the responses using type-token ratio (TTR), frequency of words, and the proportion of the first person pronoun *I*. The model fit to the TTR data was significantly improved by adding CLASS ($\chi^2(2)$=6.1, $p<0.05$). However, none of the pairwise comparisons were significant. ChatGPT had a lower TTR than Online, but the difference was not statistically significant ($\beta$=-0.04, SE=0.03, t=-1.51, $p$=0.29). Online had a lower TTR than Valid, which was not significant, either ($\beta$=-0.03, SE=0.03, t=-0.94, $p$=0.62). The difference between ChatGPT and Valid missed significance ($\beta$=-0.07, SE=0.03, t=-2.42, $p$=0.06). With respect to word frequency based on SUBTLEX-us, CLASS significantly improved the model fit ($\chi^2(2)$=8.1, $p<0.05$). As can be seen in Fig. 4, a post-hoc analysis showed that Valid had a significantly higher mean word frequency than both ChatGPT ($\beta$=5088, SE=1840, t=2.77, $p<0.05$) and Online ($\beta$=4926, SE=1921, t=2.57, $p<0.05$). ChatGPT and Online did not significantly differ from each other ($\beta$=-162, SE=1740, t=-0.09, $p$=0.995). However, CLASS did not significantly improve the model fit to the data for the proportion of *I* ($\chi^2(2)$=1.9, $p$=0.40).

As for readability of texts, there was a numerical trend in which

| | ChatGPT | Online | Valid |
|---|---|---|---|
| **Proportion of Stop Words** | M=0.41, SD=0.1 | M=0.42, SD=0.1 | M=0.44, SD=0.1 |
| **Number of Characters** | M=388.1, SD=177.9 | M=358.4, SD=101.8 | M=225.5, SD=186.1 |
| **Number of Words** | M=55.1, SD=25.1 | M=51.2, SD=14.7 | M=36.7, SD=29.0 |
| **Number of Sentences** | M=4.1, SD=1.9 | M=3.6, SD=1.2 | M=2.7, SD=2.0 |
| **Number of Words in a Sentence** | M=14.5, SD=4.5 | M=14.9, SD=3.9 | M=14.0, SD=6.8 |
| **Type-token Ratio** | M=0.77, SD=0.1 | M=0.81, SD=0.1 | M=0.84, SD=0.1 |
| **Word Frequency** | M=7698.3, SD=4514.7 | M=8126.5, SD=4317.9 | M=12905.9, SD=8740.3 |
| **Proportion of *I*** | M=0.0017, SD=0.003 | M=0.0004, SD=0.002 | M=0.0022, SD=0.005 |
| **Automated Readability Index** | M=16.08, SD=4.8 | M=16.13, SD=3.4 | M=13.19, SD=6.3 |
| **Jaccard Index** | M=0.22, SD=0.1 | M=0.39, SD=0.2 | M=0.17, SD=0.1 |
| **Cosine Similarity of SBERT Encoded Solutions** | M=0.73, SD=0.1 | M=0.84, SD=0.1 | M=0.67, SD=0.2 |

Table I: Mean and standard deviation of each language measure for the three classes

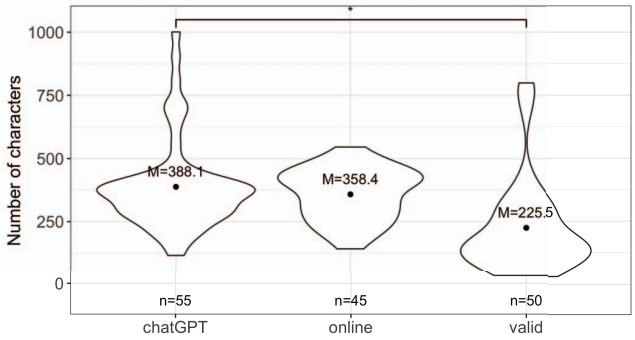


Figure 3: The distribution of the number of characters per response. M represents the mean value, and n is the number of samples included in each class. Stop words and white spaces are excluded in the count. (* indicates $p < 0.05$, ** $p<0.01$, *** $p<0.001$)

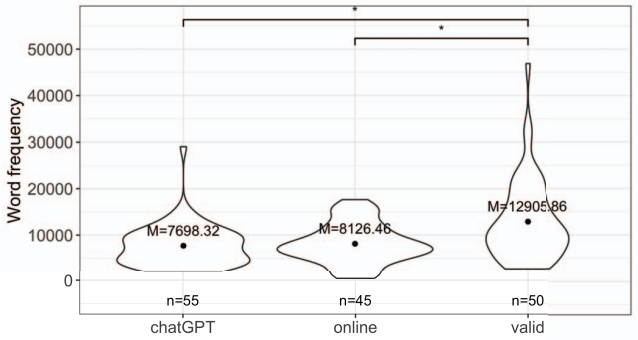


Figure 4: The distribution of the mean SUBTLEX-us frequency of words contained in a response. M represents the mean value, and n is the number of samples included in each class. (* indicates $p < 0.05$, ** $p<0.01$, *** $p<0.001$)

Valid had a lower ARI than both ChatGPT and Online, and adding CLASS to the model did significantly improve the fit to the data ($\chi^2(2)$=6.0, $p<0.05$). However, the difference between ChatGPT and Valid was not significant ($\beta$=2.05, SE=1.61, t=1.27, $p$=0.42), nor did the difference between Online and Valid reach significance ($\beta$=3.98, SE=1.67, t=2.38, $p$=0.06). The difference between ChatGPT and Online was also non-significant ($\beta$=-1.93, SE=1.44, t=-1.34, $p$=0.38).

### C. Intra-class Similarities

We examined text similarity among responses within each class, using the Jaccard index and cosine similarity of encodings using a distilled MiniLM [22] model called *all-MiniLM-L6-v2* [23]. As noted in Table I, both measures show that the similarity values of Online is the highest, followed by ChatGPT, with Valid having the lowest similarity.

For both measures, CLASS had a significant effect on text similarity. It was found that Online had a higher Jaccard similarity than ChatGPT ($\beta$=0.16, SE=0.01, t=11.90, $p<0.001$), which in turn had a higher Jaccard similarity than Valid ($\beta$=0.06, SE=0.01, t=4.35, $p<0.001$; Fig. 5). Similarly, Online had a higher cosine similarity than ChatGPT ($\beta$=0.11, SE=0.02, t=5.88, $p<0.001$), which in turn had a higher cosine similarity than Valid ($\beta$=0.09, SE=0.02, t=5.11, $p<0.001$; Fig. 6).

The result is as expected, since Online class consists of solutions simply copied from similar online sources, whereas ChatGPT tends to generate varying solutions of similar context. Finally, it is expected that Valid class responses show lowest similarity, since students come up with their own sentences without relying on external sources.

## V. Related Works

At a high level, this work combines the background knowledge of three topics of study: 1) Detection of language model usage, 2) detection and prevention of plagiarism, and 3) study of linguistic properties of language models. This section provides a short summary of prior works in each of these fields and shows how each work is related to the present study.

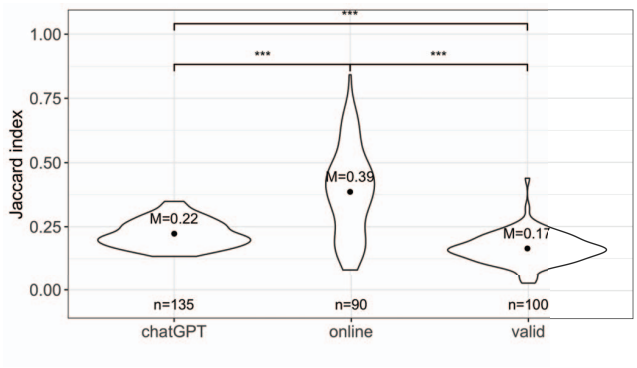


Figure 5: The distribution of Jaccard index. M represents the mean value, and n is the number of samples included in each class. (* indicates $p < 0.05$, ** $p<0.01$, *** $p<0.001$)

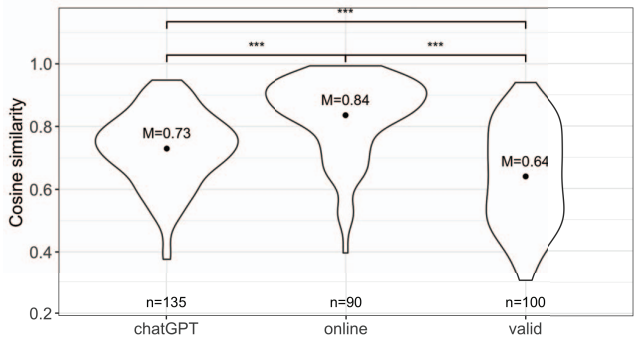


Figure 6: The distribution of cosine similarity values. M represents the mean value, and n is the number of samples included in each class. (* indicates $p < 0.05$, ** $p<0.01$, *** $p<0.001$)

***Detection of Language Model Usage:*** The ubiquitous use of large language models, particularly in academic environments, is a recent phenomenon that has stimulated considerable interest in detecting their usage [24]. Thus, detecting the usage of language model is a very active field of study as of writing [25]. The development of tools for identifying language model usage has been an active area of research, with noteworthy contributions from the GPTZero framework [26], [27], which is gaining traction in both academia and industry [28]. Furthermore, OpenAI, the creators of ChatGPT, has released a tool to identify the presence of AI-generated text [29]. However, these tools are still in their early stages of development, and OpenAI acknowledges that its model has a true positive rate of only 26% and a false positive rate of 9%, with performance deteriorating as text length decreases. These findings underscore the significant challenges associated with detecting language model usage, and highlight the ongoing efforts in this area.

***Detection and Prevention of Plagiarism:*** Detection and prevention of plagiarism has been a topic of study for multiple decades. Some notable works specific to computer science education include Donalson et al. [30], which is one of the earliest attempts at detecting plagiarism in computer science programming assignments, and MOSS, discussed in [31], which is widely used in various institutions to detect plagiarism. Furthermore, there are tools such as Turnitin [32] to detect plagiarism, but these tools rely on similarity to existing documents of databases, which can easily be avoided by a language model if it is optimized to generate new sequence of text that has not been submitted to the Turnitin database. There are also recent works on fooling MOSS via language model [33]. These findings show that while the study of plagiarism detection has a long history of research, detecting plagiarism via a large language model is a relatively recent challenge and research on detecting plagiarism will continue on for a long period of time. This paper attempts to provide new insights to this field of study.

***Study of Linguistic Properties of Language Models:*** This study also involves analyzing linguistic properties of text generated by language models and compares it against text generated by humans. Studying linguistic features of AI-generated text is still at a beginning stage, e.g. [34], and the present work is one of the first efforts to understand whether and how AI-aided text differs from human-generated text in an academic and educational setting.

## VI. Discussion & Future Works

This section summarizes findings of this study and discusses potential use cases and future directions.

***Summary of the results:*** The present study investigates which metrics of linguistic characteristics, if any, distinguish three types of solutions to essay questions in an exam of a cloud computing class, i.e. solutions aided by ChatGPT, those aided by a search engine and honest solutions. We found that solutions that consulted ChatGPT are longer than honest solutions as measured by the number of characters. Similarly, an analysis of the number of sentences shows that ChatGPT-aided responses are longer than both search engine-aided solutions and honest ones. Together, the findings suggest that students write in a concise manner when they come up with solutions on their own. We also found that words used in honest solutions have a higher frequency on average than those used in ChatGPT- and search engine-aided solutions, suggesting that the former group contains more common and familiar words than the latter groups. Finally, based on the Jaccard index and cosine similarity, we found that solutions that consulted search engine had the highest text similarity, and honest ones had the lowest text similarity.

***Additional Signals via Language Models:*** A planned future work is to use LLMs to generate additional signals for violation of academic integrity. For example, ChatGPT is known to excel at summarizing text. Therefore, it is possible to feature engineer additional signals using the summaries of each solution, further providing additional context in detecting academic integrity violations.

***Data Error, Data Bias and Sample Size:*** Another planned future work is to increase the number of samples to reduce bias in data. A potential issue that can be seen in this work is that certain metrics can easily be disturbed by one or two data points with different characteristics. One way to resolve this issue is to add more samples to diminish the effects of such bias. Another way is to add more logic into processing the data. Both are planned future tasks for this work.

***Complementing Existing Plagiarism Detectors:*** Since even the most advanced detector from OpenAI [29] is showing a low true positive rate in detecting use of LLMs, this paper can aid in providing features to use in improving such detectors. Furthermore, this work shows that success in detecting use of LLMs can increase greatly if there is a context to refer to. For example, by having the exam question and the class content as the context, one can easily generate another solution with similar linguistic characteristics to compare against other potential violations. This means that instructors can prepare their own LLM-generated solution to use as a basis for comparison, which is shown to increase the probability of detecting the use of LLM. While instructors should not rely solely or even heavily on these tools to automatically detect plagiarism, they can be used as a signal for further investigation.

## VII. Conclusion

The present study investigates the potential of AI tools such as ChatGPT to generate solutions to complex essay questions that require deep domain knowledge. While ChatGPT was able to generate coherent and relevant solutions, it provided solutions with certain linguistic patterns that were statistically different from the solutions from the control group. Thus, this work contributes to encouraging academic integrity and to minimizing the impact of AIs in reducing the effectiveness of the computer science education.

## References


[1] T.B. Brown, B. Mann, N. Ryder *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[2] A. Vaswani, N. Shazeer, N. Parmar *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[3] K. Hu, "Chatgpt sets record for fastest-growing user base - analyst note," Feb 2023. https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01

[4] AWS, "What is virtualization," 2023. https://aws.amazon.com/what-is/virtualization/#:~:text=Virtualization%20is%20technology%20that%20you,on%20a%20single%20physical%20machine.

[5] F. Bellard, "QEMU, a fast and portable dynamic translator," in *USENIX Annual Technical Conference, FREENIX Track*, 2005, pp. 41–46.

[6] J. Watson, "Virtualbox: Bits and bytes masquerading as machines," *Linux Journal*, vol. 2008, no. 166, p. 1, 2008.

[7] R. Rosen. (2013) Resource management: Linux kernel namespaces and cgroups. https://sites.cs.ucsb.edu/~rich/class/old.cs290/papers/lxc-namespace.pdf

[8] C. Boettiger, "An introduction to docker for reproducible research," *ACM SIGOPS Operating Systems Review*, vol. 49, no. 1, pp. 71–79, 2015.

[9] A. Randazzo and I. Tinnirello, "Kata containers: An emerging architecture for enabling mec services in fast and secure way," in *Proceedings of the Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*. IEEE, 2019, pp. 209–214.

[10] S. Niwattanakul, J. Singthongchai, E. Naenudorn *et al.*, "Using of Jaccard coefficient for keywords similarity," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2013, pp. 380–384.

[11] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[12] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: Analyzing text with the natural language toolkit*. O'Reilly Media, Inc., 2009.

[13] M. Brysbaert and B. New, "Moving beyond Kučera and Francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English," *Behavior Research Methods*, vol. 41, pp. 977–990, 2009.

[14] R.J. Senter and E.A. Smith, "Automated readability index," Cincinnati University, Tech. Rep., 1967.

[15] Santa Clara University, "Department of computer science and engineering." https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/

[16] Instructure, "Canvas by instructure," https://www.instructure.com/canvas, 2023.

[17] Zoom Video Communications Inc., "Zoom," https://zoom.us, 2022.

[18] G. Cluskey Jr, C.R. Ehlen, and M.H. Raiborn, "Thwarting online exam cheating without proctor supervision," *Journal of Academic and Business Ethics*, vol. 4, no. 1, pp. 1–7, 2011.

[19] A. Kuznetsova, P.B. Brockhoff, and R.H.B. Christensen, "lmerTest package: Tests in linear mixed effects models," *Journal of Statistical Software*, vol. 82, no. 13, pp. 1–26, 2017.

[20] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2021. https://www.R-project.org/

[21] R.V. Lenth, *emmeans: Estimated Marginal Means, aka Least-Squares Means*, 2023, r package version 1.8.4-1. https://CRAN.R-project.org/package=emmeans

[22] W. Wang, F. Wei, L. Dong *et al.*, "MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 5776–5788.

[23] N. Reimers, J. Gante, and O. Espejel, "all-miniLM-L6-v2." https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[24] S. Barnett, "ChatGPT is making universities rethink plagiarism," Jan 2023. https://www.wired.com/story/chatgpt-college-university-plagiarism/

[25] S. Mitrović, D. Andreoletti, and O. Ayoub, "ChatGPT or human? Detect and explain. Explaining decisions of machine learning model for detecting short ChatGPT-generated text," *arXiv preprint arXiv:2301.13852*, 2023.

[26] E. Tian, "GPTZero: The World's No. 1 AI Detector with over 1 Million Users," https://gptzero.me/, 2023.

[27] S.E. Needleman, "ChatGPT creator releases tool to detect AI-generated text, calls it 'unreliable'," Feb 2023. https://www.wsj.com/articles/chatgpt-creator-releases-tool-to-detect-ai-generated-text-calls-it-unreliable-11675204820

[28] T.H. Tran, "A college kid built an app that sniffs out text penned by AI," Jan 2023. https://www.thedailybeast.com/princeton-student-edward-tian-built-gptzero-to-detect-ai-written-essays

[29] J.H. Kirchner, L. Ahmad, S. Aaronson *et al.*, "New AI classifier for indicating AI-written text," Jan 2023. https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text

[30] J.L. Donaldson, A.M. Lancaster, and P.H. Sposato, "A plagiarism detection system," in *Proceedings of the twelfth SIGCSE technical symposium on Computer science education*, 1981, pp. 21–25.

[31] T. Lancaster and F. Culwin, "A comparison of source code plagiarism detection engines," *Computer Science Education*, vol. 14, no. 2, pp. 101–112, 2004.

[32] T. Batane, "Turning to Turnitin to fight plagiarism among university students," *Journal of Educational Technology & Society*, vol. 13, no. 2, pp. 1–12, 2010.

[33] S. Biderman and E. Raff, "Fooling MOSS detection with pretrained language models," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 2933–2943.

[34] D.M. Markowitz, J. Hancock, and J. Bailenson, "Linguistic markers of AI-generated text versus human-generated text: Evidence from hotel reviews and news headlines," Jan 2023. http://psyarxiv.com/mnyz8