Issues in Information Systems
Volume 25, Issue 4, pp. 507-519, 2024

DOI: https://doi.org/10.48009/4_iis_2024_139

Using generative artificial intelligence to create structured query language statements: examining student work in an introductory database course

Nicholas L Ball, *Utah Valley University*, *nicholas.ball@uvu.edu*

Abstract

Generative Artificial Intelligence (AI) technologies can be used to create human-quality content, including technical content like programming code. This study describes a major revision of an introduction to database course at Utah Valley University. As part of the revision, students are encouraged to create SQL statements using generative AI. Student performance is compared between those taking the AI version of the course to those taking a more traditional version. Students in the AI sections of the course scored higher on homework assignments and exams. Students in the AI sections also exhibited less variability in performance. The implications of these findings as well as suggestions for future research are also presented.

Keywords: Generative AI, Structured Query Language, Introduction to Database

Introduction

Structured Query Language (SQL) is a fundamental tool for data analysis and manipulation. It is the basis for working with relational databases. As decision-making in business becomes more data driven, those who understand SQL are increasingly valuable. Because of this many university students, not just those in traditionally technical fields (such as computer science or information systems), are taking introduction to database classes to learn SQL. Learning SQL can be difficult for many students because it not only requires students to learn the logic of interacting with a database, but also the syntax of the SQL language.

This research paper explores the potential of generative Artificial Intelligence (AI) to revolutionize the way we teach Structured Query Language. Generative AI models, capable of learning from vast amounts of data and producing human-quality text, offer exciting possibilities for personalized and interactive learning experiences. This study investigates how generative AI can be utilized to develop innovative approaches for teaching SQL, aiming to improve learning outcomes, accessibility, and overall student performance on exercises that require creating SQL statements.

AI, particularly generative AI, has received a recent surge of interest and enthusiasm. Much of this enthusiasm comes from the remarkable capabilities of AI. Generative AI can analyze massive datasets of text and code, allowing them to learn complex patterns and relationships. This empowers these tools to generate human-quality content, translate languages, write different kinds of creative text formats, and even create realistic images and videos. In the field of education, generative AI holds immense promise for personalizing learning experiences, automating tasks, and providing students with immediate and engaging feedback. In fact, the writing of elements of this paper was aided by generative AI tools.

This paper reports on the results of revising a college introduction to database class to take advantage of generative AI. In the revised class, students were encouraged to use generative AI to create the syntax of the SQL statements used to solve homework and exam exercises. The results showed that students in the revised sections demonstrated higher performance on homework exercises and exams than those in the former sections. The revised sections also exhibited less variation in student performance.

Literature Review

Not surprisingly, with the increased attention to generative AI has come an increase in attention from academics. Early research in this area focused on using AI tools to customize and improve the learning process. More recent research has highlight the benefits and potential pitfalls of using generative AI tools in an educational setting.

Much of this more recent research focusses on the generic aspects of the tools. For example, a review by Dasgupta et al. (2023) emphasized the benefits and challenges of generative AI technologies. They highlight the complexities involved in leveraging these tools effectively. No doubt as AI tools mature and become more prevalent in education, more targeted research will be forthcoming.

Many view the field of education among those potentially impacted the most by generative AI. The work of Xu & Ouyang (2022), for example, includes a systematic review of AI technologies in STEM education. Their work offers insights into the implications and benefits of incorporating AI techniques in educational settings. Baidoo-anu and Owusu Ansah (2023) describe potential uses of generative artificial intelligence in education. Specifically, their work highlights the potential benefits and drawbacks of using ChatGPT to create personalized learning experiences, translate languages, and provide feedback on essays. They also acknowledge that ChatGPT can generate false information and may not be a reliable source, so educators must be aware of its limitations.

Research has also examined teaching SQL using artificial intelligence tools (Matek, et al, 2017). This research demonstrated the effectiveness of a teaching AI tool of their own creation that provide students with customized hints for improving their query writing. They find that the tool is effective for student learning. While their tool could not be classified as generative AI, it does demonstrate how aritificial intelligence can be used to improve student learning.

Additional work has examined AI in education. For example, studies by Boscardin (2023) and Su & Yang (2023) provide a definition of generative AI, and propose theoretical frameworks for applying generative AI in education. Both stress the importance of addressing the ethical considerations inherent in incorporating generative AI tools in educational settings. They also emphasize the need for a structured approach to effectively leverage generative AI to help achieve desired learning outcomes.

Studies like those of Escalante (2023) and Lane (2024) focus on the ethical and effective use of generative AI tools, particularly in providing feedback on student writing and improving teaching methodologies in the context of nursing education. These studies discuss the potential of generative AI in enhancing teaching practices and preparing students for future professional roles. They also discuss the challenges related to academic integrity and technology verification.

Additionally, the work of Tenakwah (2023) and Johri et al. (2023) underscores the humanlike creative capacity of generative AI. They also describe the implications of this for academic integrity and assessment.

Most of these studies highlight how AI can be used to enhance the learning process. This study examines the use of AI in the classroom from a different prospective. It describes the potential of teaching students how to effectively employ generative AI to writing structured query language programming code. It aims to determine if teaching student to use generative AI to write SQL queries will improve their ability to solve problems that require an SQL solution.

Context of the Study

This study was conducted in the context of four consecutive semesters of an introduction to database course at Utah Valley University. All four courses where taught online. Most students complete this course during their sophomore year. This course provides the first opportunity many students have to interact with a database.

Approximately 75% of the course engages students in creating structured query language (SQL) statements. The focus of the course is on writing SELECT statements to retrieve data from a database. Students learn about all six clauses (SELECT, FROM, WHERE, GROUP BY, HAVING, and ORDER BY) of the SELECT statement and are expected to eventually create fairly sophisticated query statements including those that require inner and outer joins involving more than two tables, aggregated results, and subqueries. Because most students are unfamiliar with SQL statements, they find learning both the logic and syntax of SQL to be difficult and intimidating.

The Traditional Course Model

Two of the sections of the course were delivered using what we will term the "Traditional" model. The traditional model includes recorded lectures and hybrid help sessions. The help sessions are conducted live and recorded so that students who are not able to connect with the meeting can view them at their convenience.

Student mastery of the SQL content in the course is evaluated using five homework assignments and two exams. The assessments ranged in difficulty from simple to relatively complex. Simple SQL statements typically involve only the SELECT and FROM clauses of the select statement. The most complex queries involve a combination of advanced joins, compound where conditions, subqueries, or aggregate calculations.

The course materials include an electronic textbook, the course assignments, and a platform for executing queries against a database using a web browser. Student queries are graded using a real-time grading system that provides students will detailed feedback on their work. Student are allowed to learn from this feedback, adjust their work, and resubmit each assignment and exam. The larger of the first two submission for any assessment is recorded in the course grade book.

The Generative Artificial Intelligence (AI) Course Model

The other two sections of the course used the "AI" model. This modality for these courses was also online. Students were presented with prerecorded lectures and hybrid help sessions as with the traditional model. The AI model involved a significant shift in the presentation of content for the course. In this model, students where taught to use generative artificial intelligence to create the SQL statements used to interact with the databases.

Since the students were not expected to create their own SQL statements, there was concern that they would not be able to understand how they worked. To address this problem, students were provided with a framework for retrieving data from a database. This framework involves a four-step process:

1. ***Plan*** to retrieve the data
2. ***Generate*** a SQL statement using generative AI

3. ***Evaluate*** the proposed SQL statement
4. ***Try*** the statement

Plan

First the student creates a plan to retrieve the data that they need. The data need is determined by the business task the data must support. Specifically, the plan must address these four questions:

1. Which columns should be shown?
2. Which tables hold the data?
3. Which rows should be included?
4. What order should the rows be in?

The answers to these questions will be used to help the student to effectively interact with the generative AI tool and also evaluate the proposed SQL statement generated by the tool.

Generate

Second, the students then use a generative AI tool, such as ChatGPT, Google Gemini, or Microsoft CoPilot to create the SQL statement. These tools are extremely effective at writing SQL code if two conditions are met. First, the tool must know the structure of the data in the database. Students are provided with the SQL statement necessary for building the tables. These statements are provided to the generative AI tool to train it on the structure of the data.

Students must also formulate a precise request of the tool that will elicit a SQL statement that will satisfy the data need. The formulation of the requests is commonly termed prompt engineering. As students progress through the course, they become more proficient at formulating requests that precisely describe the data requirements. The generative AI tool provides both a valid SQL statement and a brief explanation documenting the elements of the statement.

Evaluate

Third, once the SQL statement is generated, students are expected to evaluate the proposed code. This is done by referring to the query plan created in the first step of the process. While students are not expected to create their own SQL code, they must know enough about SQL statements to determine if the proposed statement will meet the requirements of the query plan.

Try

Finally, the course materials include an electronic textbook that allows students to execute their queries against the database. The materials allow students to use Oracle or SQLite as the database management system. By executing the queries, students can see if the SQL statement produces the data needed to address the original data task.

Course Assignments

Because the students are not generating their own SQL statements, they can complete more assignments during the course. This version of the course has eight homework assignments, two exams, and a comprehensive final exam. As with the traditional model, student work is assessed using a real-time automatic grading system. This system provides detailed feedback and the opportunity for students to complete multiple submissions. As with the traditional model, student scores were calculated using the highest score from the first two submissions.

Assessing Learning

Students were allowed to use AI on both assignments and exams. This leads to a practical concern that it can be difficult to determine what the students are learning in the course, particularly if student performance on writing SQL statements is the only measurement of learning. To address this, students were asked to create a query plan (described earlier in this section) for a number of the queries they wrote during the semester. This provides some assurance that students are learning how to logically approach creating SQL statements.

Additionally, students were asked to evaluate SQL statements. Students were presented with statements that were either correct or contained errors. They were asked to identify when a statement contained errors and how the errors might be corrected. This assessed the degree to which students could spot and correct errors in SQL statements generated by AI.

Students and Delivery Models

As mentioned at the beginning of this section, the study was conducted over the course of four semesters. The AI model represented a significant revision to the course. Because of this, the first two courses used the traditional model. These courses were conducted during the Fall semester of 2022 and Spring 2023. The AI model was used starting in the Fall of 2023 and used again in the Spring of 2024. In total 56 students completed the course with the traditional model and 57 students completed the AI version of the course.

There are not any formal hypotheses for this study. It is an exploratory examination of the effectiveness of the AI model for delivering the introduction to database course. While the author intuitively felt that the AI model has some advantages for students, it was unclear when the course was revised if these advantages would translate to improvements in students' ability to use SQL to address business needs. Using data from student performance in these courses, we examine the effectiveness of the AI model as it compares to the traditional model.

Data Analysis

In this study student performance is operationalized in terms of student scores on the homework assignments and exams. Student performance in the traditional sections of the course is first presented. Scores from the AI sections are then outlined. Finally, performance is compared across the two groups.

Student Performance in the Traditional Course

Students completed five homework assignments and two exams in the sections using the traditional model. Figure 1 shows the average student scores on each of these assessments. Average scores on these assessment range from a low of 71.4 on the first exam to a high of 94.65 on the first assignment.

Issues in Information Systems
Volume 25, Issue 4, pp. 507-519, 2024

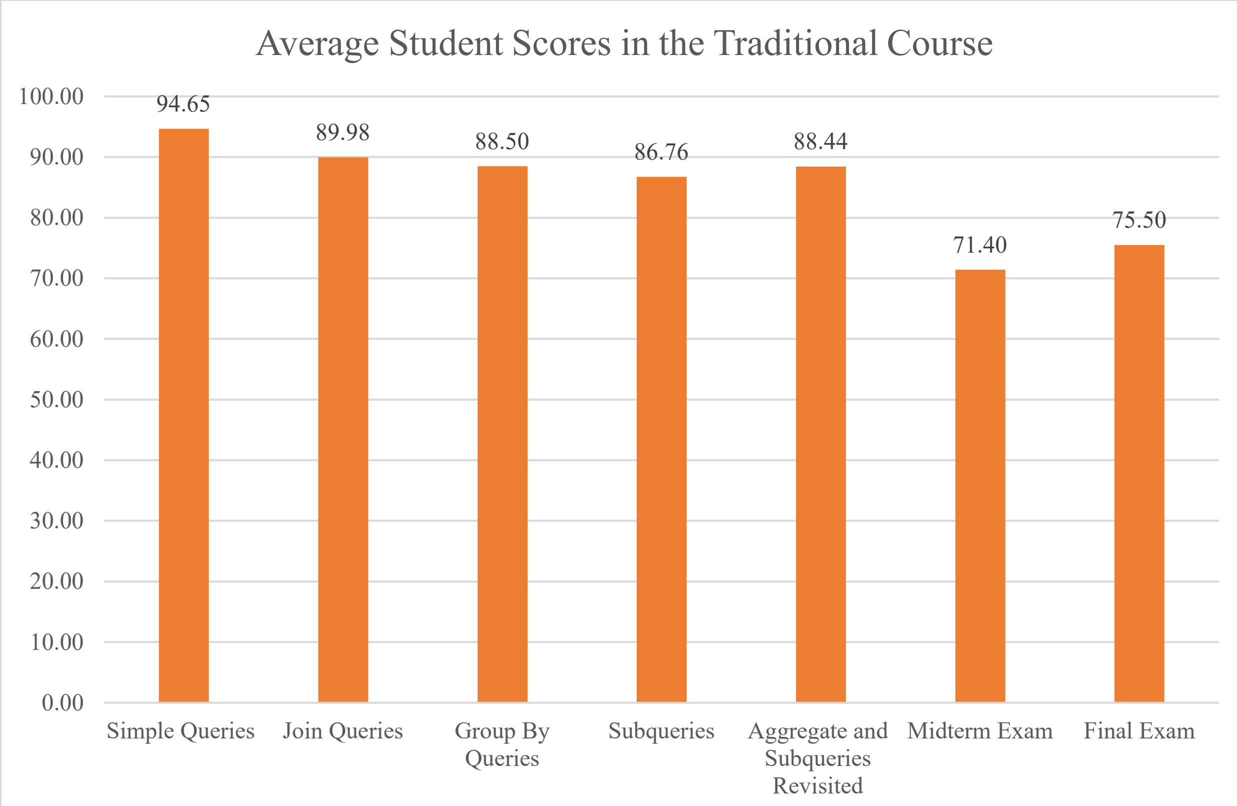

Figure 1 – Average Student Scores in the Traditional Sections of the Class

Table 1 shows provides additional details on student performance in sections using the traditional model. It highlights the average score, highest score, lowest score, and standard deviation of student performance on each of the assessments. Note that the highest score for each assessment is 100 percent. Multiple students scored 100 percent on one or more assessments though none of the students scored 100 percent on all assessments. Despite this, there is strong evidence that a number of students performed well in the class.

There is also strong evidence of variation in student scores. On each of the assessments, multiple students scored poorly. The lowest score on each assessment is at most 59/100. The standard deviation for each assessment is also presented. It will be discussed further once the performance for the students in the sections using the AI model is presented.

Table 1 – Detailed Student Scores in the Traditional Sections

Assessment	Average	High	Low	Standard Deviation
Simple Queries	94.65	100	55	8.23
Join Queries	89.98	100	59	13.44
Group By Queries	88.50	100	10	19.01
Subqueries	86.76	100	33	21.34
Aggregate and Subqueries Revisited	88.44	100	50	15.54
Midterm Exam	71.40	100	24	19.53
Final Exam	75.50	100	22	20.33

Student Performance in the AI Course

Students completed eight homework assignments and three exams in the course using the AI model. Figure 2 shows the average scores for students on each of these assessments. These scores ranges from a high of 98.87 on the grouping assessment to a low of 87.23 on the subqueries assessment.

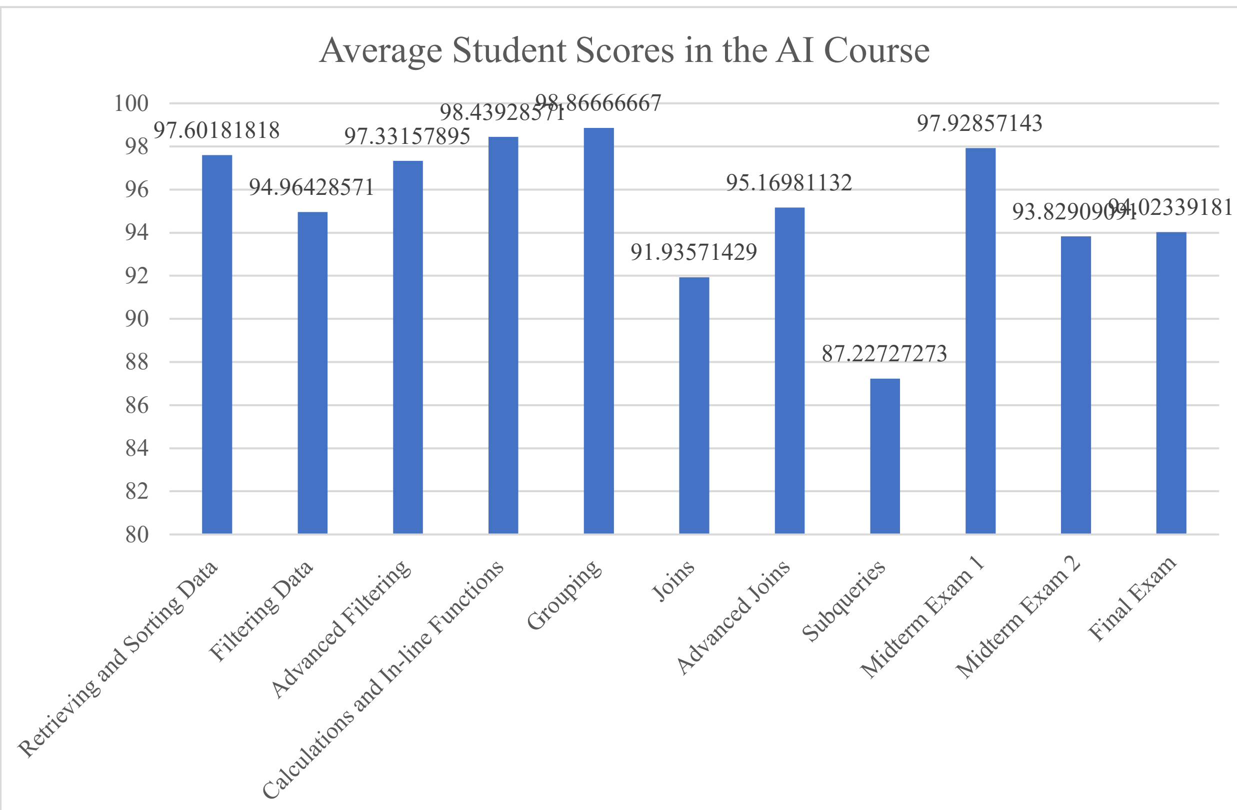

Figure 2 – Average Student Scores in the AI Sections of the Class

Table 2 shows more detailed statistics on student performance in the sections using the AI model. As in the traditional model, there is strong evidence that a number of students performed well in these sections of the course. The highest score for each of the assessments is 100 percent. Multiple students scored 100 percent on each assessment and one student scored 100 percent on all assessments.

There is also evidence of variability in student scores. The lowest student scores ranged from a low of 46.9 on the first midterm exam to a 84 on the retrieving and sorting data assessment. The standard deviations for each assessment is also presented.

Table 2 – Detailed Student Scores in the Traditional Sections

Assessment	Average	High	Low	Standard Deviation
Retrieving and Sorting Data	97.60	100.00	84.00	4.80
Filtering Data	94.96	100.00	56.00	10.65
Advanced Filtering	97.33	100.00	70.50	5.10
Calculations and In-line Functions	98.44	100.00	83.60	3.75
Grouping	98.87	100.00	83.80	3.14
Joins	91.94	100.00	59.50	10.86
Advanced Joins	95.17	100.00	82.80	5.42
Subqueries	87.23	100.00	57.40	15.78
Midterm Exam 1	97.93	100.00	46.90	8.21
Midterm Exam 2	93.83	100.00	63.30	6.69
Final Exam	94.02	100.00	76.33	6.40

Comparing Student Scores Across Sections

Direct comparisons of student performance across sections are somewhat difficult to make for two reasons. The first of these is that the assessment used in the two models are not the same. Additionally, the grading technology used in different models is also different. Because of this, only aggregate comparisons of student performance are made and care should be taken in interpreting the results of these comparisons.

Both models had both homework assignments and exams. Homework in both sections is used to assess student learning, though the primary purpose is to allow students to experience writing SQL statements. Because of this, students are encouraged to collaborate on homework assignments. Additionally, homework assignments are the subject of each of the student help sessions. Students are allowed to ask questions of the professor about each of the assignments. Often each homework assignment question is discussed during the homework help session.

The primary purpose of exams in both models is to gauge student mastery of the course topics. There are not any help sessions for exams. Professor help on exams is limited to clarifying exam questions and instructions. Students are expected to complete the exams without the help of other students. Exams are also available during a specific time window.

Because of the similarity in how homework assignments and exams are treated across all sections, average scores on all homework assignments and exams are compared for students in the traditional and AI sections. Figure 3 shows the average homework and exam score for all assessments in the traditional and AI sections of the course. The average score on all homework assignment in the traditional model is 89.45 and the average exam score is 73.38. The averages in the AI sections are 95.19 for homework assignments and 95.93 for exams.

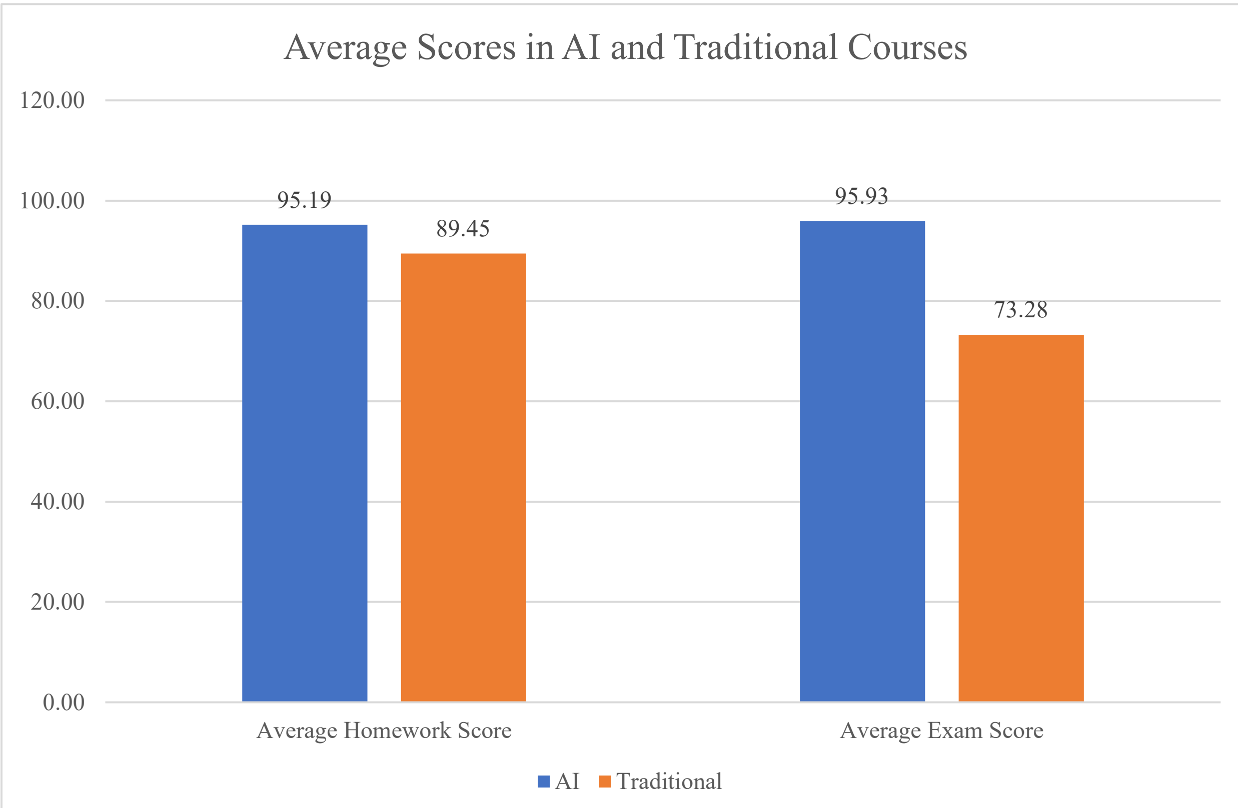

Figure 3 – Comparing Average Student Scores in the Traditional and AI Sections

On average, students in the AI sections of the class scored 5.74 points higher in the AI sections than the traditional sections. This represents a statistically significant different (see table 3). Likewise the difference in exam scores between the traditional and AI sections is also statistically significant. On average, students in the AI sections scored 22.65 points higher than students in the traditional sections.

Table 3 – Detailed Student Scores across Sections

Course Element	AI Average	Traditional Average	Difference	Standard Deviation AI	Standard Deviation Traditional	P-Value
Homework	95.19	89.45	5.74	5.47	12.03	0.0017
Exam	95.93	73.28	22.65	5.58	16.63	0.0000

Statistical significance is determined using paired t-tests. The t-tests assumed unequal variances. Because of the exploratory nature of the student, there was not an assumption that one group would outperform another group. Therefore the tests were two-tailed. The p-value for the t-test of differences in average homework scores between students in the AI and traditional sections was 0.0017. The p-value for the differences in exam scores was 0.000. Both are significant at the 0.05 level.

These differences in average scores are interesting. Additionally, there is evidence that there is less variability in student performance in the AI sections. The standard deviation for the homework and exam scores in the AI sections are 5.47 and 5.58, respectively. The standard deviation for student scores in the traditional sections are 12.03 for homework and 16.63 for exams. The standard deviations for homework and exams are much higher in the transitional sections than the AI sections. This is clear evidence that there is less variability in student performance in the AI sections.

Discussion

There are a number of interesting findings in this study. One interesting element of the transition to the AI sections is the fact that students in these sections were able to complete more assignments. Additionally, the differences in student performance on both the homework assignments and exams is intriguing. Both of these findings will be examined in this section. As with many exploratory studies, these results also uncover new questions that will require subsequent research to answer. In order to understand the implications of the findings it is important to understand the limitations of the study. With these in mind, suggestions for additional research are discussed.

Increased Student Efficiency

Probably the most incontrovertible finding of this study is that when students are encouraged to use generative AI to write SQL statements, they can complete more exercises for the class. In this case, students both completed more homework assignments, but in many cases more exercises in the assignments.

This is likely due to the fact that less class time is devoted to learning the syntax of structured query language. This does not mean that students are not taught about SQL syntax. The focus instead is on evaluating existing SQL code rather than creating code from scratch. This is particularly notable in the more advanced SQL topics (such as joins, subqueries, and aggregate queries).

The time saved by teaching students how to evaluate rather than write SQL statements in this study was reinvested in providing students more practice solving problems. This time might also have been reinvested in covering additional database topics or teaching students to solve more advanced problems. It is unclear which approach would be more beneficial to students. It is likely that providing students with additional practice lead to at least some of the performance differences found, particularly on exams.

Increased Student Performance

Differences in student performance were found between the AI and traditional sections of the course. In particular, students performed 5.75 points (out of 100) on average higher on homework assignments in the AI sections than the traditional sections. Additionally, they scored 22.65 points high on average on exams. Both differences are statistically significant. Interestingly, the AI sections also exhibited less variability in student scores.

This is strong evidence that students performed better on the SQL exercises in the AI sections than the traditional sections. These differences might be attributed to a few factors.

The AI Approach

The first of these may well be that using generative AI to develop SQL statements is a better approach to writing SQL statements from scratch, at least for those learning SQL. One plausible reason for this is that a novice programmer must learn both the logic and syntax of the programming language. Using generative AI to create the syntax frees the learner to learn the logic involved in solving a problem using SQL, thereby lessoning the cognitive load placed on the learner. This allows the student to more quickly and completely understand SQL logic making them better at solving problems with SQL.

The Effectiveness of the Tools

Another reason why student performance is higher in the AI sections may relate to the overall effectiveness of the generative AI tools. There are two important hallmarks of these tools. The first of these is that the tools are adept at "understanding" the questions posed to the tools even if the questioner has a limited technical capability. While it is true that some user requests are formatted better than others, generative AI tools are typically good a deciphering less that precise requests. Second, generative AI tools are good at creating SQL code, particularly if the structure of the data in the database is available to the tool.

The student performance increases in this study may be attributed to the fact that the solutions proposed by generative AI are generally very effective. Many required only modest modifications, if any, to be correct for the assessments.

This is definitely true of the least complex exercises in the class. More complex exercises were likely to require students to make more adjustments and more significant adjustments to the queries proposed by the AI tool for the proposed solutions to be correct.

It is important to note that as the complexity of the exercises increased in the class, student performance remained consistently higher for those in the AI sections than those in the traditional sections. This suggests that the increased performance may not be completely attributed to the effectiveness of the tool alone. Of course. If using the tool makes students better at solving SQL exercises, ultimately, they should be encouraged to use the tool.

Other Course Differences

The use of generative AI tools for creating the SQL statements was not the only difference between the sections. These other differences are discussed in the limitations section of the paper. As the study was designed it is impossible to rule out other factors that might have important contributions to student performance.

Another interesting finding is the discrepancy in the average differences between the homework assignments and exams. Why is the performance difference so much higher for exams than it is for homework assignments. There may be a few reasons for this.

First, the professor of all sections has created a culture of collaboration on assignments, but not exams. Students were encouraged to help each other and seek the help of the professor on homework assignments. In all sections, the professor held an online help session where students could ask questions about the homework assignments. It was not uncommon for each exercise on an assignment to be discussed during the help sessions. These sessions were recorded and available to all students. This was not the case for exams. With exams, students were expected to work on their own. Any help provided by the professor would not provide any guidance on specific exercises. The culture of collaboration might account for higher scores on homework assignments in both sections, creating a smaller difference for exams than homework assignments.

Another explanation for a larger difference on exams than homework assignments might be that students in the AI sections had more practice with SQL exercises than those in the traditional sections. With additional practice came increased proficiency. The larger gap in exam performance may well be attributed to students being more adept at solving the SQL exercises.

Limitations

Before discussing the findings of this study, it is important to understand its limitations. Perhaps the most significant limitation of the study is that there are at least two significant pedagogical differences between the traditional and the AI sections of the course. The first of these is the encouraged use of generative AI to create the students' SQL statements. The second is the prescribed approach (plan, generate, evaluate, and try) to solving each exercise in the class. While the reported differences in student performance are likely due, at least in part, to the use of generative AI; the approach to solving the exercise likely also had some positive effect on student performance. Given the nature of this study it is impossible separate the effect of AI use from that of the problem solving approach.

Additionally, students did not complete the same exercises in the two versions of the class. Although the problem sets the students used ultimately assessed the same class objectives, different problem sets were administered to the different groups. Differences in student performance may be partially due to the different assessments. Anecdotally, the course instructor believes that the exercises students were asked to complete were more difficult in the AI sections than in the traditional sections which may lower the performance in the AI sections.

In addition to differences in the exercises, students in the AI sections of the class were required to complete more SQL exercises than those in the traditional sections. This may suggest that the AI sections are more difficult than the traditional sections. However, the extra practice provided to the students in the AI sections may help explain their superior performance on the exams.

Given these limitations to the study, care should be taken in interpreting the results. In particular, these findings provide some initial evidence that students who use generative AI to aid in the creation of SQL statements will outperform those who do not. Additional research is needed to confirm these findings.

Suggestions for Future Research

Additional research should carefully separate the major treatments in this study, namely the use of generative AI and the general approach to problem solving, and there independent effects on student performance. Likewise, future research could assign students to complete the same exercises or those that are empirically shown to be equivalent. Finally, it is also be interesting to examine different measures of student effectiveness. One that might be particularly interesting is student confidence. Are students more confident in the SQL statement they write themselves or those that were created with the help of a generative AI tool? Understanding this may provide a more holistic understanding of how the use of generative AI affects how well students perform coding tasks such as writing SQL statements.

Conclusion

This paper reports on the results of a significant revision to a college introduction to database class. In the revised class, students were encouraged to use generative AI to create the syntax of the SQL statements used to solve homework and exam exercises. Students in the revised class were also encouraged to adopt the plan, generate, evaluate, and try approach to solving problems with SQL statements. The results demonstrate that students in the revised sections had higher performance on homework exercises and exams than those in the traditional sections. The revised sections also exhibited less variation in student performance.

References

Baidoo-anu, D., & Owusu Ansah, L. (2023). Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning. Journal of AI, 7(1), 52-62. https://doi.org/10.61969/jai.1337500

Boscardin, C. (2023). Chatgpt and generative artificial intelligence for medical education: potential impact and opportunity. Academic Medicine, 99(1), 22-27. https://doi.org/10.1097/acm.0000000000005439

Dasgupta, D., Venugopal, D., & Gupta, K. (2023). A review of generative ai from historical perspectives.. https://doi.org/10.36227/techrxiv.22097942

Escalante, J. (2023). Ai-generated feedback on writing: insights into efficacy and enl student preference. International Journal of Educational Technology in Higher Education, 20(1). https://doi.org/10.1186/s41239-023-00425-2

Johri, A., Katz, A., Qadir, J., & Hingle, A. (2023). Generative artificial intelligence and engineering education. Journal of Engineering Education, 112(3), 572-577. https://doi.org/10.1002/jee.20537

Lane, S. (2024). Tool or tyrant: guiding and guarding generative artificial intelligence use in nursing education. Creative Nursing, 30(2), 125-132. https://doi.org/10.1177/10784535241247094

Matek, Tadej & Zrnec, Aljaz & Lavbič, Dejan. (2017). Learning SQL with Artificial Intelligent Aided Approach. International Journal of Information and Education Technology. 7. 803-808. 10.18178/ijiet.2017.7.11.976.

Su, J. and Yang, W. (2023). Unlocking the power of chatgpt: a framework for applying generative ai in education. Ecnu Review of Education, 6(3), 355-366. https://doi.org/10.1177/20965311231168423

Tenakwah, E. (2023). Generative ai and higher education assessments: a competency-based analysis.. https://doi.org/10.21203/rs.3.rs-2968456/v1

Xu, W. and Ouyang, F. (2022). The application of ai technologies in stem education: a systematic review from 2011 to 2021. International Journal of Stem Education, 9(1). https://doi.org/10.1186/s40594-022-00377-5