Integrating LLMs into Database Systems Education

Kishore Prakash, Shashwat Rao, Rayan Hamza, Jack Lukich, Vatsal Chaudhari, Arnab Nandi
Computer Science and Engineering
The Ohio State University
USA
{prakashsailaja.1,rao.462,hamza.23,lukich.9,chaudhari.55,nandi.9}@osu.edu

ABSTRACT

Large Language Models (LLMs) have sparked a drastic improvement in the ways computers can understand, process, and generate language. As LLM-based offerings become mainstream, we explore the incorporation of such LLMs into introductory or undergraduate database systems education. Students and instructors are both faced with the calculator dilemma: while the use of LLM-based tools may "solve" tasks such as assignments and exams, do they impede or accelerate the learning itself? We review deficiencies of using existing off-the-shelf tools for learning, and further articulate the differentiated needs of database systems *students* as opposed to trained data practitioners. Building on our exploration, we outline a vision that integrates LLMs into database education in a principled manner, keeping pedagogical best practices in mind. If implemented correctly, we posit that LLMs can drastically amplify the impact of existing instruction, minimizing costs and barriers towards learning database systems fundamentals.

KEYWORDS

database systems education, undergrad databases, intro to db, llm, large language models, foundation models, ChatGPT

ACM Reference Format:

Kishore Prakash, Shashwat Rao, Rayan Hamza, Jack Lukich, Vatsal Chaudhari, Arnab Nandi. 2024. Integrating LLMs into Database Systems Education. In *3rd International Workshop on Data Systems Education: Bridging education practice with education research (DataEd '24), June 9, 2024, Santiago, AA, Chile.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3663649.3664371

1 INTRODUCTION

Modern education environments are rife with digital assistance tools that augment curricular materials, such as web search engines, interactive IDEs, and more recently, LLM-powered conversational agents such as ChatGPT, Gemini, or Claude. However, this method of learning needs to impart a comprehensive understanding. Our focus falls specifically on database education, a cross-disciplinary field of study that has wide-ranging effects on various disciplines. It is crucial to have a learning environment in database education that provides a comprehensive and pedagogically effective system that allows students to interact deeply with the content.

DataEd '24, June 9, 2024, Santiago, AA, Chile

ACM ISBN 979-8-4007-0678-3/24/06…$15.00
https://doi.org/10.1145/3663649.3664371

The future of education is rapidly changing, and large language models (LLMs) are at the forefront of this transformation. Such tools are capable of ingesting massive amounts of educational material, providing personalized learning support and recommendations for educational resources, and even offering real-time problem-solving and academic guidance through dialogue and interaction with students [16]. Such tools are now available [42] that connect into modern Learning Management Systems provide fine-tuning and RAG-based conversational access to the existing course materials. This allows students to ask natural language questions against course materials and get synthesized natural language answers about the material, without relying on an instructor or teaching assistant. Beyond acting as a conversational reference, such tools can provide students easy access to a wealth of knowledge and practical skills, such as techniques for handling dirty data [5]. Integrating AI (and particularly generative AI) into teaching is a rapidly developing topic at multiple universities [12, 24]. Current educational tools for data science, including interactive SQL editors and auto-graders, are primarily aimed at assessing technical accuracy. However, this approach overlooks the nuanced educational needs of students, which extend beyond mere correctness to encompass a deeper comprehension and mastery of the subject matter. LLMs can shift this focus towards achieving learning outcomes in a personalized and conversational way, allowing students to reason back and forth with their educational material. In the following sections, we outline our vision of a virtual tutor that blends LLMs and database systems to accelerate database systems education.

Instructors, TAs, and Tutors: In the current undergraduate educational settings, an instructor is aided by teaching assistants or graders to provide instruction, conduct labs, grade exams, and guide the student through a comprehensive learning experience. While several activities for these roles can be automated (e.g., autograders, recorded lectures), each plays a critical part in the educational experience, not just providing information to the students but also continually monitoring their performance and guiding them through the learning journey. In an ideal setting with infinite resources, this teaching experience can be paired with a dedicated personal *tutor*. Unlike the instructor, TAs, and graders who focus on the entire class, a tutor can take the lens of the student. The tutor can personalize the course materials based on the student's prior experience and performance, and looks out for the student's learning experience and day-to-day performance in the class. Furthermore, an ideal tutor is available to the student at any time to assist study sessions, even outside of regular hours. Such a tutor would allow each student to unlock their fullest potential, especially for students who may struggle with a particular aspect of the class and hence fall back on the overall class performance. Providing such a dedicated human tutor to all students is extremely human-intensive and in most settings, untenable. As an alternative, can we develop AI-powered *virtual tutors* – that are infinitely scalable and always available – to all our students?

Incorporating mastery-driven education: Bloom's Framework emphasizes *mastery* [4] as the goal of education, advocating for personalized learning that adapts to each student's unique pace and needs. This approach, grounded in the principle of individualized instruction, is posited to significantly improve educational outcomes by aligning teaching methods with the diverse learning styles and rates of students. Such personalization is not just ideal but necessary, considering the diverse backgrounds and learning styles of students. Piaget and Vygotsky[15, 46] further enrich this perspective by emphasizing the role of active engagement and the social context in learning, suggesting that knowledge is constructed most effectively through experiences tailored to the learner's current understanding and social interactions .

Integrating technology into this personalized learning paradigm offers unprecedented opportunities to achieve mastery at scale. The SAMR [20] and TPACK [30] frameworks guide us in evolving from the mere substitution of traditional tools with digital equivalents to redefining educational tasks in new ways. For example, technology can enable adaptive learning paths that are responsive to individual progress, thereby strengthening mastery. Furthermore, the ICAP framework's emphasis on interactive and constructive activities aligns with the use of technology to foster deeper cognitive engagement and collaborative learning [8]. The essence of practical education in the digital age lies in its ability to personalize and interactively engage with the learner, principles that are foundational to both traditional educational theories and modern technological frameworks. This sets the stage for exploring how current LLM technologies, despite their advancements, still face challenges in fully realizing this ideal of personalized, mastery-based education. The next section will delve into the limitations of LLMs in their current state, highlighting the necessity of our vision.

2 CHALLENGES OF LLMS IN EDUCATION

While LLMs present a powerful generic tool for assisting both students and instructors in different tasks, their direct integration into the classroom experience raises several issues.

Bias in responses: When developing models, especially for education, it is essential to address the problem of biases inherent in the model due to the training data [28, 37]. Providing students with biased responses or examples are especially problematic and can have long-term amplified effects. Approaches towards the removal of biases in LLMs [31, 36] have considered manipulating the training data or applying corrections to the model after training has been done. Aside from the *demographic* biases which are the typical consideration in this space, LLMs have the potential to be biased against newly proposed problem solutions, approaches, and curricula, due to the vast amount of data it can find on older and more prominent solutions and curriculum structures. Finally, for LLMs trained on data from unscreened corpora, some sources may contain incorrect or not fully accurate information, which can 'taint' student learning.

Data privacy and security: In addition to bias concerns, the privacy and security of the data [28, 37] is a concern. Specifically, the initial training data can be accessed and misused, and the input provided to the model can be potentially exploited. In the other direction, students trust that their interactions with the models to be secure and private (in many jurisdictions, this may be a legal requirement), and such interactions can not be included in future fine tuning, training, or other model development activities.

Student's over-reliance on the model, critical thinking: While ChatGPT has made several positive impacts in education [1, 48, 53], there is growing concern about students' increasing dependency on it [25, 28]. While similar shifts may have occurred with the launch of other disruptive information tools such as web search, the added convenience of ChatGPT has the potential to hamper a deeper understanding of the material and critical thinking skills [6, 40].

Cheating and misuse: General-purpose LLM-based tools have made it extremely easy to generate verbose, human-like content. With this in mind, students may leverage it to generate answers to a particular assignment [10]. In addition, it is often challenging to distinguish between a student response and an LLM-generated response [10], especially as LLMs improve in quality. While LLMs can be useful in gathering data and developing ideas, many universities currently prohibit the submission of AI-generated text and content without citation, analogous to plagiarism policies [41].

Sensitivity to prompting: Prior work has highlighted that LLMs are sensitive to the wording of prompts and the sequence of demonstrations provided during fine-tuning [45]. This sensitivity can lead to inconsistent results, making it challenging to assess their performance fairly. This increases the onus on the students to rely on specially-worded prompts to effectively extricate full insights from the LLMs. Here, instead of focusing on learning database systems concepts, students may end up spending a large amount of their focus on prompt engineering.

3 RELATED WORK

Virtual Tutor Systems, Database Tools: While LLMs have been crucial in many advancements, there has been much ingenuity in the field of virtual tutoring systems in database education. Currently, virtual tutoring systems like AutoTutor [2, 17] and RAG-based LLMs, amongst others, are paving the way to a new classroom experience for students in the future. Simultaneously, there are several practitioner tools that can aid with various aspects of learning database systems, including but not limited to systems like DB-BERT [51], DukeDB [39], data tweening [29], and I-Rex [38].

Language Models in Database Systems: Various projects have attempted to utilize language models for data management tasks, which can translate easily into database education tasks. DB-BERT is a tool that leverages natural language analysis to read through manuals and understand databases [51]. By understanding the manual, DB-BERT can suggest changes to database settings to optimize performance. Google Gemini, formerly known as Bard, has integrated implicit code execution to boost the accuracy of the model's response in the domain of logic and reasoning by 30% [32]. Google does this by leveraging the model itself, in that the model generates code whenever it feels necessary to answer a specific question and integrates the output into the response. ChatGPT also employs the implicit code execution technique. While it has been used in math and reasoning tasks, other areas, such as visualization, have also benefited. With this capability, both technical and non-technical professionals can analyze and interpret information that they have within a short time frame [43]. These advances in database query tuning and implicit code execution could provide ample assistance to students who will likely experiment with mock databases and SQL code during their studies.

Implicit Query Execution LLMs and database systems are promoted to have a synergistic relationship, where there are various opportunities for one component to enhance another and vice versa. There has been work promoting the usage of LLMs for data integration and query processing for databases, as well as the usage of databases for data preparation and prompt engineering for LLMs [3]. There have been various suggested areas in which databases and data management can support LLMs to perform better, especially concerning prompting and training [54]. There has already been work done on combining LLMs with vector databases to more effectively utilize high volatility data [26]. The flourishing integration of LLMs and database systems provides both justification and motivation for integrating LLMs for database education. Empowering LLMs to help educate students and tackle sophisticated data management problems leads to improved databases and data management, which could subsequently cascade back to LLMs.

Query Execution via LLM + Database Connection There is a growing body of work exploring how to run SQL queries generated by language models effectively. The BIRD system [35] makes running complex SQL queries more efficient. Dr. Spider [7] tests how well these models perform under different conditions by introducing perturbations to assess their performance. Pedro et. al. [44] show ways to protect against SQL injection attacks, which also proposes mitigation techniques crucial for secure query execution. TREQS [52] is designed for healthcare databases and deals with specific challenges such as medical abbreviations, a capability that is underscored using the MIMICSQL dataset. Finally, the KaggleDBQA dataset [33] provides a testing ground for checking if language models can handle the kind of complex data found in authentic databases. ChatGPT has demonstrated ample success in writing basic SQL queries when given a schema along with the query prompt [14].

Query Understanding and Instruction: One of the defining key features of LLMs is their outstanding ability to understand semantics directly from natural language input. On the topic of database education specifically, LLMs can solve a variety of entity resolution and schema matching problems [14]. This includes mapping similar addresses with different formats or matching columns to their abbreviated counterparts across tables. This advancement enables future database systems students to work on a more varied range of architectures despite having less experience.

Technologies currently integrated with LLMs: While fine-tuning LLMs on datasets tends to meet requirements for many sophisticated tasks, an LLM and a dataset alone may not be ideal for tackling education and tutoring. Retrieval-Augmented-Generation (**RAG**) focuses on cross-referencing text generation with external documents, leading to better scalability and accuracy [34]. There are a variety of approaches available: dialogue response generation [55], machine translation [19], and others [21, 34]. **ReAct** combines ideas of reasoning and action. In this approach, reasoning is used by the model to formulate, monitor, and revise plans for action, while action steps enable the model to collect further information from external resources. This method of prompt engineering is designed to reduce the occurrence of hallucinations by encouraging the model to consult external knowledge bases for verification [56]. **Meta-prompting** causes the LLM to reflect on its performance and amend or adjust its responses accordingly. In our case, a fully functional system would ideally be a chain of conversation between the backend system and the student [23]. Focusing on enhancing pedagogical capabilities, *LearnLM* [27] is a recent initiative that advances the Gemini LLM by integrating key learning science principles such as active learning, cognitive load management, learner adaptation, curiosity stimulation, and metacognitive development.

EMT conversations: Expectation-Misconception Tailored (EMT) conversations have been employed in previous models like AutoTutor [17], to facilitate deep learning for students when discussing concepts related to the class. The idea is to have problems that require 3-7 sentences of response. Students will then interact with the system in a conversational style such that the system tries to lead the student to the answer. If misconceptions are provided to the system, the system will converse with the student and correct them so that the student can be redirected to the track of the expected response or expectations required to answer the question [2, 17, 18]. ChatGPT can be used with various hallucination mitigating techniques to generate these EMT scripts for students to interact with [2].

Visual Representations: While text-based explanations are useful, students may benefit from visual representations of the information given by query explanation tools. Data tweening [29], for example, provides a visual and incremental view of the transformations between query result sets, facilitating a deeper understanding of data manipulation. MOCHA [50], which is focused on SQL, illustrates the impact of different physical operators on query execution plans, connecting theoretical knowledge with practical applications. The I-Rex tool [38] traces query evaluations and offers counterexamples, enhancing educational content with useful insights. Aside from visuals, the GIFT [2] framework has been used alongside ChatGPT to utilize data from images, presentations, and videos for content generation.

4 OUR VISION: AN LLM-POWERED DB TUTOR

Given the context, challenges, and related work, our solution (Fig. 1) combines some state-of-the-art ideas under the focus of mastery into an LLM-based framework. The virtual tutor will be made available as a chatbot in the University's chat tool (e.g., Teams, Slack, Discord) that allows students to interact with the system about database concepts, starting with SQL. Students will be able to learn how to read and write SQL (structured query language) while visualizing the step-throughs of schema, result diagrams, and

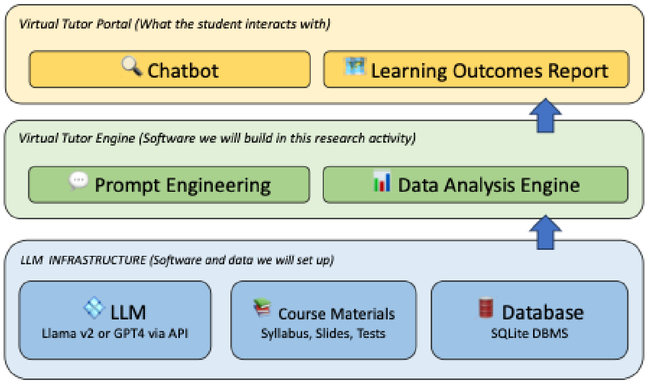

Figure 1: An example architecture for enhancing LLMs to benefit education at a high level as a virtual tutor.

query trees. Students will have round-the-clock access to the tutor, allowing them to receive immediate responses for questions, clarifications, and follow-ups at any time they are studying. Such a DB Tutor will also take the initiative to engage them through pop quizzes, notifications, and suggestions. Unlike off-the-shelf products like ChatGPT, the tutor will prioritize learning outcomes, focus on avoiding inaccurate results (e.g., hallucinations), and provide verified results through SQL runtimes, as described below.

4.1 Architecture

The proposed architecture for a virtual tutor leveraging LLMs comprises three main components:

Virtual Tutor Portal:

- The user interface layer for student interaction with the system.
- Features a chatbot as the conversational interface for student queries and provides a Learning Outcomes Reports to offer feedback on student performance.

Virtual Tutor Engine:

- Houses the Prompt Engineering mechanism to translate natural language into SQL queries.
- Includes a Data Analysis Engine to execute queries and analyze results, while facilitating interaction between the user's input and the backend LLM and database.

LLM Infrastructure:

- The backend system utilizing an LLM such as Llama 2 or GPT via API.
- Accesses the course materials for contextually informed responses and interacts with a Database like SQLite DBMS to execute and produce query results.

In use, a student's query is processed through the Chatbot at the Virtual Tutor Portal, then through the Virtual Tutor Engine's prompt engineering module to the SQLite database in the LLM Infrastructure. The LLM interprets the database execution results, cross-references course materials, and provides detailed explanations or corrections as necessary, thus supporting a personalized and contextualized learning experience.

Feature	I-REX [38]	ChatGPT [43]	MOCHA [50]	ReAct [56]	EQG[1] [13]
Debugging	✓		✓	✓	
Natural Language Explanations		✓		✓	✓
Step-hroughs	✓		✓		✓
EMT		✓		✓	✓
Mastery			✓	✓	✓

Table 1: An example component breakdown of the proposed system.

Subsequent sections will delve deeper into the Learning Outcomes Report and other specific features this architecture facilitates. Table 1 contains an overview of the proposed system outlining its components and their contributions. Each of these features is designed to complement the system's educational capabilities. Through this fusion, our system hopes to provide a comprehensive, mastery-focused learning experience.

4.2 Implicit Query Execution

Our vision involves using an LLM tailored to incrementally teach students, with class-specific schemas that evolve throughout the semester. However, in the context of natural language to SQL (NL2SQL) tasks, the LLM may hallucinate, or provide inaccurate answers. Here, we draw from ideas in *implicit code execution used in Gemini and ChatGPT* [32], where the LLM can generate a query, which is then run in an isolated database. The SQL result (or SQL query error) is then checked and returned to the user. By integrating a backend to host these schemas for implicit code execution, we enhance the precision of the LLM's responses, ensuring they align with the course content. This system facilitates a continuous exchange with the database, yielding accurate and relevant answers. Additionally, by abstracting away the need for students to set up a database instance on their own computers, we streamline the learning process, allowing students to focus directly on the educational material.

4.3 Visual Step-Throughs

As discussed in related works, our project will incorporate elements of visual representations. We believe that the integration of visual step-throughs with query explanations generated by LLMs can significantly enhance students' comprehension, catering to diverse learning preferences—some may favor textual explanations while others might benefit more from visual aids. We envision a scenario where students interact with a chatbot to discuss database schemas or query operations. Concurrently, the output from an implicit executor guides a visual display that illustrates the step-by-step execution of each operation and its impact on the database schema, such as row deletion or query extraction. This visual component, displayed alongside the chatbot interface, offers direct insight into the underlying processes, facilitating a deeper understanding of runtime behavior and theoretical principles.

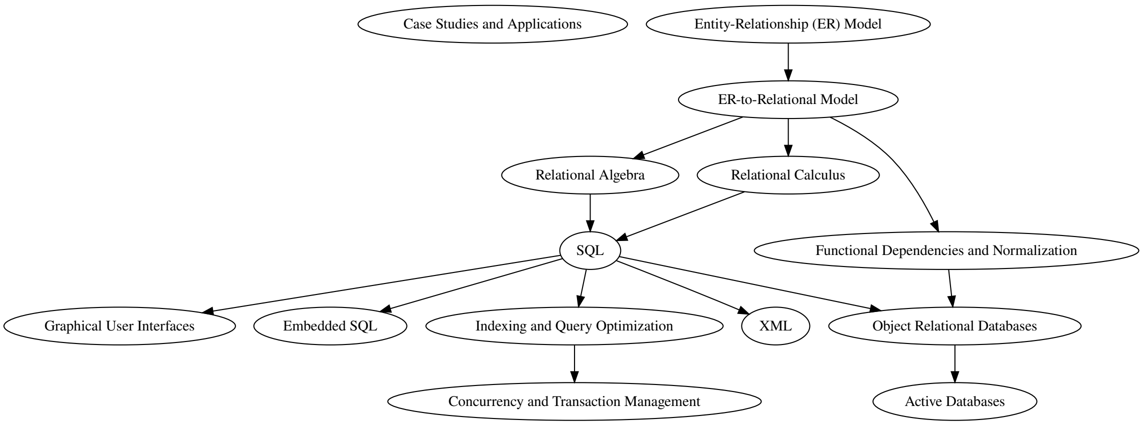

Figure 2: An example concept dependency diagram derived from an *Introduction to Database Systems* course syllabus.

4.4 Data Personalization

Educational challenges, such as facilitating contextual learning, and providing hands-on practice are crucial [49]. Addressing these challenges, we integrate personalized data, drawing on the power of tailored queries and contextually relevant analogies.

In addition to general semantic understanding, LLMs can generate cross-domain analogies that are helpful in reformulating problems [11]. Furthermore, LLMs can be used to synthesize training data for a specific task, which can be useful when real training data is scarce [47]. For instance, LLMs are already able to generate mock data regarding a certain topic, as shown in the prompt *[Generate a sample data table of the NBA roster]* to Microsoft Copilot:

```
|Player|Position|Team|Games|Starts|...|
|Lebron James|Forward|Lakers|60|...|
|Luka Doncic|Guard|Mavericks|59|...|
|Giannis Antetokounmpo|Center|Bucks|...|
```

Building on LLMs' generative AI abilities, our system personalizes the learning experience by generating database schemas with synthetic data that reflect individual interests. For example, a student interested in pop culture might engage with database exercises featuring music data about *Taylor Swift* and *discographies*, while a sports enthusiast might prefer football score examples related to *Travis Kelce* and *Superbowl track record*. This personalization extends to real-world examples too, meta-prompts that consider the student's interests for added context, and even case studies and motivations that align with what the student finds most engaging. Our aim is to discern the user's interests and leverage when providing responses, making learning more relevant and engaging.

4.5 Pop quizzes

Research indicates that integrating quizzes within the learning and study process promotes better engagement and information retention [9, 22]. We propose that the tutor provides quizzes when a student provides an incorrect query or response, the tutor could provide targeted questions utilizing strategies such as EMT to evaluate and reinforce the student's understanding. Separately, when explicitly requested for study questions pertaining to a quiz or exam, the virtual tutor should yield relevant questions reflecting the context or current progress of the course. This approach is designed to bolster material retention and clarify misconceptions.

4.6 Learning Outcomes Report

The Learning Outcomes Report offers an evaluation of the student's progress and comprehension in the database course, grounded in a knowledge graph. This graph functions as the foundation for generating a thorough report, as depicted in Figure 2. It enables the system to compare student responses to the curriculum, identifying areas where understanding may be lacking based on interconnected concepts. By analyzing conversations and assignments, the system assesses the student's position within the knowledge graph, highlighting knowledge deficiencies. The report visualizes these interactions to pinpoint gaps in learning and charts the student's conceptual development over time.

5 CONCLUSION

Database concepts such as SQL and relational algebra are often a stumbling block for students, requiring significantly more attention and interaction than just lectures. While students can look up answers and information from LLM-powered tools, these sources focus on information lookups and problem solving rather than a student's learning and mastery of the topics. Other state-of-the-art educational tools such as autograders and interactive SQL editors focus on technical correctness rather than comprehensive understanding of the essential underlying principles. Addressing the deficiencies of these offerings, an LLM-based Database Tutor can provide a variety of features for students, including learning outcomes reports, implicit code execution, data personalization, and pop quizzes. Here, LLMs' current focus on immediate solutions and technical correctness can be shifted to a more pedagogical learning-based approach. While we envision the virtual tutor as a tool focused on database education, most of the concepts discussed can be easily adapted to other topics and disciplines as well.

ACKNOWLEDGMENTS

This material is based upon work supported by the NSF Grant #1910356 and an OSU Undergraduate Research Access Seed Grant.

REFERENCES

[1] Rania Abdelghani, Yen-Hsiang Wang, Xingdi Yuan, Tong Wang, Pauline Lucas, Hélène Sauzéon, and Pierre-Yves Oudeyer. 2023. GPT-3-Driven Pedagogical Agents to Train Children's Curious Question-Asking Skills. *International Journal of Artificial Intelligence in Education* (June 2023). https://doi.org/10.1007/s40593-023-00340-7

[2] Faruk Ahmed, Keith Shubeck, and Xiangen Hu. 2023. Chatgpt in the generalized intelligent framework for tutoring. In *Proceedings of the 11th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym11)*. US Army Combat Capabilities Development Command–Soldier Center, U.S. Army Combat Capability Development Command - Solider Center (DEVCOM-SC), Orlando, FL, 109–113.

[3] Sihem Amer-Yahia, Angela Bonifati, Lei Chen, Guoliang Li, Kyuseok Shim, Jianliang Xu, and Xiaochun Yang. 2023. From Large Language Models to Databases and Back: A Discussion on Research and Education. *SIGMOD Rec.* 52, 3 (nov 2023), 49–56. https://doi.org/10.1145/3631504.3631518

[4] Benjamin S Bloom. 1984. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational researcher* 13, 6 (1984), 4–16.

[5] Angela Bonifati, Sihem Amer-Yahia, Chen Lei, Li Guoliang, Shim Kyuseok, Xu Jianliang, and Yang Xiaochun. 2023. From Large Language Models to Databases and Back A discussion on research and education. *SIGMOD record* (2023).

[6] Dominik Borovskỳ, Jozef Hanč, and Martina Hančová. 2023. Innovative approaches to high school physics competitions: Harnessing the power of AI and open science. *arXiv preprint arXiv:2309.02986* (2023).

[7] Shuaichen Chang, Jun Wang, Mingwen Dong, Lin Pan, Henghui Zhu, Alexander Hanbo Li, Wuwei Lan, Sheng Zhang, Jiarong Jiang, Joseph Lilien, et al. 2023. Dr.Spider: A Diagnostic Evaluation Benchmark Towards Text-to-SQL Robustness. In *International Conference on Learning Representations*.

[8] Michelene T.H. Chi and Ruth Wylie. 2014. The ICAP framework: Linking cognitive engagement to active learning outcomes. *Educational Psychologist* 49, 4 (2014), 219–243.

[9] Brian Robert Cook and Andrea Babon. 2017. Active learning through online quizzes: Better learning and less (busy) work. *Journal of Geography in Higher Education* 41, 1 (2017), 24–38.

[10] Debby RE Cotton, Peter A Cotton, and J Reuben Shipway. 2023. Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. *Innovations in Education and Teaching International* (2023), 1–12.

[11] Zijian Ding, Arvind Srinivasan, Stephen Macneil, and Joel Chan. 2023. Fluid Transformers and Creative Analogies: Exploring Large Language Models' Capacity for Augmenting Cross-Domain Analogical Creativity. In *Proceedings of the 15th Conference on Creativity and Cognition* (, Virtual Event, USA,) *(C and C '23)*. Association for Computing Machinery, New York, NY, USA, 489–505. https://doi.org/10.1145/3591196.3593516

[12] Karthik Duraisamy, Varun Agrawal, Brian D. Athey, Tazin Daniels, Kapotaksha Das, Nick Gaspar, Anne Ruggles Gere, Cait Hayward, Libby Hemphill, Robert Jones, Nigel P. Melville, David Mendez, Rada Mihalcea, John Rodriguez, Kayte Spector-Bagdady, Theresa Tinkle, Colleen van Lent, Paul Watta, Mallory Kwiatkowski, and Thomas Burns. 2023. Generative Artificial Intelligence Advisory Committee Report. *UM Generative AI* (2023).

[13] Sabina Elkins, Ekaterina Kochmar, Jackie C.K. Cheung, and Iulian Serban. 2024. How Teachers Can Use Large Language Models and Bloom's Taxonomy to Create Educational Quizzes. *ArXiv* abs/2401.05914 (2024). https://api.semanticscholar.org/CorpusID:266933483

[14] Raul Castro Fernandez, Aaron J Elmore, Michael J Franklin, Sanjay Krishnan, and Chenhao Tan. 2023. How large language models will disrupt data management. *Proceedings of the VLDB Endowment* 16, 11 (2023), 3302–3309.

[15] Lev Semenovič forme avant 2007 Vygotskij and Vera John-Steiner. 1979. *Mind in society: The development of higher psychological processes*. Harvard University Press.

[16] Wensheng Gan, Zhenlian Qi, Jiayang Wu, and Jerry Chun-Wei Lin. 2023. Large language models in education: Vision and opportunities. In *2023 IEEE International Conference on Big Data (BigData)*. IEEE, 4776–4785.

[17] A.C. Graesser, P. Chipman, B.C. Haynes, and A. Olney. 2005. AutoTutor: an intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education* 48, 4 (2005), 612–618. https://doi.org/10.1109/TE.2005.856149

[18] Arthur C Graesser, Kurt VanLehn, Carolyn P Rosé, Pamela W Jordan, and Derek Harter. 2001. Intelligent tutoring systems with conversational dialogue. *AI magazine* 22, 4 (2001), 39–39.

[19] Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor Li. 2017. Search Engine Guided Non-Parametric Neural Machine Translation. *Proceedings of the AAAI Conference on Artificial Intelligence* 32 (05 2017). https://doi.org/10.1609/aaai.v32i1.12013

[20] Erica R Hamilton, Joshua M Rosenberg, and Mete Akcaoglu. 2016. The substitution augmentation modification redefinition (SAMR) model: A critical review and suggestions for its use. *TechTrends* 60 (2016), 433–441.

[21] Tatsunori B. Hashimoto, Kelvin Guu, Yonatan Oren, and Percy Liang. 2018. A retrieve-and-edit framework for predicting structured outputs *(NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 10073–10083.

[22] Alice F Healy, Matt Jones, Lakshmi A Lalchandani, and Lindsay Anderson Tack. 2017. Timing of quizzes during learning: Effects on motivation and retention. *Journal of Experimental Psychology: Applied* 23, 2 (2017), 128.

[23] Yutai Hou, Hongyuan Dong, Xinghao Wang, Bohan Li, and Wanxiang Che. 2022. MetaPrompting: Learning to Learn Better Prompts. In *Proceedings of the 29th International Conference on Computational Linguistics*, Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (Eds.). International Committee on Computational Linguistics, Gyeongju, Republic of Korea.

[24] Michael V. Drake Institute. 2023. Artificial Intelligence and Instruction. https://drakeinstitute.osu.edu/resources/artificial-intelligence-and-instruction

[25] Nayab Iqbal, Hassaan Ahmed, and Kaukab Abid Azhar. 2022. Exploring teachers' attitudes towards using chatgpt. *Glob. J. Manag. Adm. Sci* 3 (2022), 97–111.

[26] Zhi Jing, Yongye Su, Yikun Han, Bo Yuan, Haiyun Xu, Chunjiang Liu, Kehai Chen, and Min Zhang. 2024. When Large Language Models Meet Vector Databases: A Survey. arXiv:2402.01763 [cs.DB]

[27] Irina Jurenka et al. 2024. Towards Responsible Development of Generative AI for Education: An Evaluation-Driven Approach. *Google Technical Report* (2024).

[28] Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tilman Michaeli, Claudia Nerdel, Jürgen Pfeffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, Matthias Stadler, Jochen Weller, Jochen Kuhn, and Gjergji Kasneci. 2023. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences* 103 (2023), 102274. https://doi.org/10.1016/j.lindif.2023.102274

[29] Meraj Ahmed Khan, Larry Xu, Arnab Nandi, and Joseph Hellerstein. 2017. Data tweening: incremental visualization of data transforms. *Proceedings of the VLDB Endowment* 10 (02 2017), 661–672. https://doi.org/10.14778/3055330.3055333

[30] Matthew J. Koehler and Punya Mishra. 2009. What is Technological Pedagogical Content Knowledge (TPACK)? *Contemporary Issues in Technology and Teacher Education* 9, 1 (2009), 60–70.

[31] Hadas Kotek, Rikker Dockum, and David Sun. 2023. Gender bias and stereotypes in Large Language Models. In *Proceedings of The ACM Collective Intelligence Conference* (, Delft, Netherlands,) *(CI '23)*. Association for Computing Machinery, New York, NY, USA, 12–24. https://doi.org/10.1145/3582269.3615599

[32] Jack Krawczyk and Amarnag Subramanya. 2023. Bard is getting better at logic and reasoning. https://blog.google/technology/ai/bard-improved-reasoning-google-sheets-export/

[33] Chia-Hsuan Lee, Oleksandr Polozov, and Matthew Richardson. 2021. KaggleDBQA: Realistic Evaluation of Text-to-SQL Parsers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, Online, 2261–2273. https://doi.org/10.18653/v1/2021.acl-long.176

[34] Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A Survey on Retrieval-Augmented Text Generation. arXiv:2202.01110 [cs.CL]

[35] Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Rongyu Cao, Ruiying Geng, et al. 2023. Can LLM Already Serve as A Database Interface? A Big Bench for Large-Scale Database Grounded Text-to-SQLs. *arXiv preprint arXiv:2305.03111* (2023).

[36] Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and booktitle = "Proceedings of the 38th International Conference on Machine Learning Salakhutdinov, Ruslan". 2021. Towards Understanding and Mitigating Social Biases in Language Models. PMLR, 6560–6570. http://proceedings.mlr.press/v139/liang21a.html

[37] Brady D Lund and Ting Wang. 2023. Chatting about ChatGPT: how may AI and GPT impact academia and libraries? *Library Hi Tech News* 40, 3 (2023), 26–29.

[38] Zhengjie Miao, Tiangang Chen, lexander Bendeck, Kevin Day, Sudeepa Roy, and Jun Yang. 2020. I-Rex: an interactive relational query explainer for SQL. *Proceedings of the VLDB Endowment* 13 (08 2020), 2997–3000. https://doi.org/10.14778/3415478.3415528

[39] Zhengjie Miao, Tiangang Chen, and Yihao Hue. 2021. Helping Novices Learn and Debug Relational Queries. https://dukedb-hnrq.github.io/categories

[40] Fatemeh Mosaiyebzadeh, Seyedamin Pouriyeh, Reza Parizi, Nasrin Dehbozorgi, Mohsen Dorodchi, and Daniel Macêdo Batista. 2023. Exploring the Role of ChatGPT in Education: Applications and Challenges. In *Proceedings of the 24th Annual Conference on Information Technology Education*. Association for Computing Machinery, Marietta GA, USA, 84–89.

[41] The Ohio State University Offic of Academic Integrity. 2023. Artificial intelligence and academic integrity. https://oaa.osu.edu/artificial-intelligence-and-academic-integrity.

[42] University of Michigan IT. 2023. U-M Maizey In Depth. *UM Maizey* (2023).

[43] OpenAI. 2023. Introducing ChatGPT Enterprise. https://openai.com/blog/introducing-chatgpt-

[44] Rodrigo Pedro, Daniel Castro, Paulo Carreira, and Nuno Santos. 2023. From Prompt Injections to SQL Injection Attacks: How Protected is Your LLM-Integrated Web Application? arXiv:2308.01990 [cs.CR]

[45] Pouya Pezeshkpour and Estevam Hruschka. 2023. Large Language Models Sensitivity to The Order of Options in Multiple-Choice Questions. arXiv:2308.11483 [cs.CL]

[46] Jean Piaget, Margaret Cook, et al. 1952. *The origins of intelligence in children*. Vol. 8. International Universities Press New York.

[47] Andy Rosenbaum, Saleh Soltan, and Wael Hamza. 2023. Using large language models (llms) to synthesize training data. *Amazon Science* (2023).

[48] Tzu-Yu Tai and Howard Hao-Jan Chen. 2023. The impact of Google Assistant on adolescent EFL learners' willingness to communicate. *Interactive Learning Environments* 31, 3 (2023), 1485–1502. https://doi.org/10.1080/10494820.2020.1841801 arXiv:https://doi.org/10.1080/10494820.2020.1841801

[49] Toni Taipalus. 2023. SQL: A Trojan Horse Hiding a Decathlon of Complexities. In *Proceedings of the 2nd International Workshop on Data Systems Education: Bridging Education Practice with Education Research* (Seattle, WA, USA) *(DataEd '23)*. Association for Computing Machinery, New York, NY, USA, 9–13. https://doi.org/10.1145/3596673.3603142

[50] Jess Tan, Desmond Yeo, Rachael Neoh, Huey-Eng Chua, and Sourav S Bhowmick. 2022. MOCHA: a tool for visualizing impact of operator choices in query execution plans for database education. *Proceedings of the VLDB Endowment* 15, 12 (2022), 3602–3605.

[51] Immanuel Trummer. 2022. DB-BERT: A Database Tuning Tool that "Reads the Manual" *(SIGMOD '22)*. Association for Computing Machinery, New York, NY, USA, 190–203. https://doi.org/10.1145/3514221.3517843

[52] Ping Wang, Tian Shi, and Chandan K. Reddy. 2020. Text-to-SQL Generation for Question Answering on Electronic Medical Records *(WWW '20)*. Association for Computing Machinery, New York, NY, USA, 350–361. https://doi.org/10.1145/3366423.3380120

[53] Zichao Wang, Andrew Lan, and Richard Baraniuk. 2021. Math Word Problem Generation with Mathematical Consistency and Problem Context Constraints. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 5986–5999. https://doi.org/10.18653/v1/2021.emnlp-main.484

[54] Zige Wang, Wanjun Zhong, Yufei Wang, Qi Zhu, Fei Mi, Baojun Wang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023. Data Management For Large Language Models: A Survey. arXiv:2312.01700 [cs.CL]

[55] Jason Weston, Emily Dinan, and Alexander Miller. 2018. Retrieve and Refine: Improved Sequence Generation Models For Dialogue. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI*, Aleksandr Chuklin, Jeff Dalton, Julia Kiseleva, Alexey Borisov, and Mikhail Burtsev (Eds.). Association for Computational Linguistics, Brussels, Belgium, 87–92. https://doi.org/10.18653/v1/W18-5713

[56] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629* (2022).