

2. Literature Review

2.1 Overview of Math Education Technology

2.1.1 Evolution of Digital Tools in Mathematics Learning

Mathematics education has consistently been a focus of innovation in educational technology. Researchers have explored various tools and methodologies to enhance learning experiences and address challenges in mathematical understanding. Several AI-powered mathematical tools have been developed to solve equations and provide solutions. Heffernans [4] examined the potential of AI to automate learning tasks and improve student outcomes. However, these tools are often standalone applications that do not combine note-taking functionalities with AI-driven problem-solving.

The evolution of digital tools in mathematics education reflects a paradigm shift in how mathematical concepts are taught, learned, and applied. Early tools such as calculators and computer algebra systems offered computational support but limited interactivity. [21] Over recent decades, dynamic software platforms like GeoGebra and Desmos have transformed learning by enabling students to visualize and manipulate mathematical objects interactively, fostering deeper conceptual understanding beyond rote calculation. [22] The increased availability of web-based and mobile technologies has broadened access, making learning more flexible and personalized.

Furthermore, digital tools have evolved from static calculators to intelligent systems embedded with adaptive learning algorithms that tailor content and feedback to individual learner needs. This shift aligns with a growing recognition that mathematics learning is not merely procedural but highly conceptual and exploratory. [23] Drijvers emphasizes that the integration of technology in math education demands a dialectic approach, balancing human reasoning with computational power to create meaningful learning experiences. [24] These tools support collaborative learning, enable formative assessment, and enhance teacher facilitation in classrooms.

2.1.2 Impact of Mobile Applications on Self-Directed Learning

Mobile applications have catalyzed self-directed learning by providing anytime-access to mathematical content and problem-solving support. [25] Their portability allows learners to engage with math in informal settings, encouraging frequent practice and spontaneous inquiry. [26] Mobile math apps often incorporate gamification and immediate feedback mechanisms, which increase motivation and promote sustained engagement, especially among younger learners.

However, the effectiveness of mobile learning depends heavily on the app's design to accommodate screen size constraints and input modalities. [27] Clear, concise presentation of complex symbols and the ability to input handwritten equations are critical features. Moreover, mobile apps contribute to personalized pacing and adaptive learning paths, allowing students to revisit difficult topics and progress at their own rhythm, an essential factor for mastery in mathematics.

2.1.3 Role of Visual Learning and Note-Taking in Math Understanding

Visual learning strategies and note-taking practices are indispensable in mathematics education because they externalize abstract thinking and enable reflection. [18] Visual tools such as dynamic graphs, color-coded annotations, and symbolic manipulations help students organize and interpret complex mathematical relationships, thereby enhancing comprehension and retention. [17] The advent of digital note-taking platforms that support multimedia inputs—including handwritten notes, sketches, and embedded calculations—allows learners to document the dynamic process of mathematical reasoning more naturally.

Research shows that students who engage in active note-taking and utilize visual aids demonstrate better problem-solving abilities and conceptual understanding. [19] Digital note-taking applications designed for math learning can scaffold student thinking by integrating stepwise solution pathways and linking visual representations with symbolic expressions. This multimodal approach aligns with cognitive theories that advocate for reducing working memory load and promoting meaningful learning through external representations.

User experience (UX) and interface design are critical factors influencing the success of educational apps. Cognitive load theory, introduced by Sweller, provides a foundational framework for managing the complexity of mathematical content within learning applications. [17] Minimizing extraneous cognitive load enables students to allocate more cognitive resources to understanding core mathematical concepts. Incorporating this theory into app design can help create an intuitive learning environment that scaffolds student comprehension effectively.

2.2 Intelligent Tutoring Systems (ITS) and Math Solvers

2.2.1 Overview of ITS in STEM Education

Intelligent Tutoring Systems (ITS) have become a cornerstone in STEM education by offering highly personalized and adaptive learning experiences. These systems leverage models of student knowledge and behavior to detect errors and misconceptions in real time, tailoring instructional feedback accordingly. [19] The adaptive nature of ITS allows for differentiated learning paths that respond dynamically to the learner's pace and proficiency, which has been

demonstrated to produce learning outcomes comparable to those achieved through human tutoring. ITS not only provide stepwise problem-solving guidance but also foster metacognitive skills by encouraging students to reflect on their thinking processes. As a scalable educational tool, ITS help bridge gaps in resource-limited environments by providing quality tutoring to diverse populations.

2.2.1 AI-Powered Math Problem Solving

The emergence of AI-powered math solvers has revolutionized the way students interact with complex mathematical problems. [10] These solvers, often based on neural network program synthesis and other deep learning techniques, are capable of generating novel problems as well as providing detailed, stepwise solutions across a wide array of mathematical domains including calculus, differential equations, and linear algebra. Unlike traditional calculators, these AI systems provide explanations that facilitate conceptual understanding and encourage learners to engage deeply with the underlying mathematics. By enabling learners to explore alternative solution paths and receive immediate, contextualized feedback, AI-powered math solvers support the development of both procedural fluency and higher-order problem-solving skills.

Recent advancements in AI-driven mathematics problem-solving have shown promising results, particularly in the domain of neural networks. Neural networks can solve and generate mathematics problems using program synthesis. Their research covers a broad range of topics, including calculus, differential equations, and linear algebra. [10] By applying program synthesis, a method that leverages machine learning to automate the generation of programs that can solve mathematical problems, the study demonstrates how neural networks can generate solutions and even propose new mathematical formulations.

Beyond problem-solving accuracy, educational technology research underscores the importance of digital tools that facilitate effective note-taking and problem-solving in mathematics. [16] Du and Wagner examined how digital environments can enhance students' ability to organize and comprehend mathematical concepts, thereby improving learning outcomes. These findings suggest that a math notes app must not only support accurate computation but also foster meaningful cognitive engagement through well-designed interfaces and functionalities.

The role of external representations, such as notes, diagrams, and symbolic manipulations, in supporting mathematical reasoning has been well-documented. [18] The work highlights how well-structured representations aid in problem-solving by offloading cognitive demands onto the environment, which is directly relevant when designing features for math notes apps.

2.2.2 Natural Language and Handwriting Recognition in Math Tools

The integration of natural language processing (NLP) and handwriting recognition technologies significantly enhances the usability and accessibility of math learning tools. [1] NLP allows learners to interact with math solvers using conversational queries, which lowers barriers for students who might find formal mathematical syntax intimidating. Concurrently, handwriting recognition enables the natural input of mathematical notation on digital devices, such as tablets and smartphones, making the learning process more intuitive and aligned with traditional note-taking habits. Together, these technologies create an interface that closely mirrors human tutoring interactions, improving engagement and providing precise, context-aware feedback that adapts to the user's input style. This multimodal interaction is essential for modern math learning applications aiming to support diverse learners and complex mathematical content.

2.3 Comparison of Existing Math Applications

2.3.1 Photomath

Photomath is a popular mobile application that uses the phone's camera to capture handwritten or printed math problems and provides step-by-step solutions. It is designed primarily for solving arithmetic and algebraic problems, including word problems, fractions, and basic calculus. The app's strength lies in its ability to recognize various math problems from images, offering solutions with clear explanations. However, it is limited to solving predefined types of math problems and lacks a robust note-taking feature. Photomath is ideal for immediate problem-solving but does not allow users to store notes or engage in complex mathematical reasoning beyond basic to intermediate levels.

2.3.2 Microsoft Math Solver

Microsoft Math Solver offers a robust platform combining AI and OCR technologies to solve a wide variety of math problems. It extends its functionality by providing interactive graphs, related educational resources such as video tutorials, and quizzes, which help contextualize the problem-solving process. [14], [9] The app's seamless integration across devices enhances accessibility for learners. Nevertheless, it occasionally struggles to provide detailed conceptual feedback for complex problems, and its performance can be hindered by ambiguous or imprecise input.

2.1.4 Mathway

Mathway is a versatile math solver that handles a wide range of problems, from basic math to advanced calculus. The app allows users to input problems via typing or snapping a photo of handwritten notes. Mathway's core feature is its ability to solve equations and provide answers instantaneously. However, its user interface does not integrate well with note-taking or personal note storage. The app focuses more on solving problems efficiently rather than providing a complete mathematical education tool. Additionally, Mathway has some limitations in handling non-standard notation and complex word problems, where human intervention may still be necessary.

2.3.3 Desmos and Wolfram Alpha

Both Desmos and Wolfram Alpha are widely used in academic settings, offering strong computational and graphing capabilities. Desmos specializes in graphing and visualizing mathematical functions and is widely used for teaching and learning purposes. It enables users to input equations and visualize their graphs in real-time. Wolfram Alpha, on the other hand, is a computational knowledge engine that covers a vast range of topics, including math, science, and data analysis. Wolfram Alpha excels in providing high-quality, detailed step-by-step solutions, but its free version offers limited features. The premium version is more expensive than other alternatives, and despite its power, it lacks the simplicity and ease of use found in other apps.

While both Desmos and Wolfram Alpha excel in their respective areas, they do not provide direct support for note-taking or freehand drawing of mathematical equations, which could limit their use for students who prefer a more interactive experience.

2.3.4 Apple Notes (Math Notes)

Apple Notes is a well-known note-taking app with features such as text, image, and handwriting support. Recently, Apple has integrated advanced math-related features within the Notes app, such as the ability to scan handwritten math equations and perform basic calculations directly within the notes. While this integration is helpful for simple note-taking and equation solving, it does not offer the full range of functionality available in dedicated math-solving apps. For example, Apple Notes lacks step-by-step problem-solving, AI integration for advanced equation analysis, and the ability to store or retrieve solved math equations for later use.

The Math Notes feature in Apple Notes offers basic support for solving equations, but it remains limited in scope compared to specialized applications like Photomath and Microsoft Math Solver.

2.3.5 Summary of Limitations in Existing Tools

Despite the many advantages offered by existing math applications, several key limitations persist across most platforms. Firstly, note-taking functionality is often lacking; while apps like Photomath and Microsoft Math Solver excel in solving problems, they do not provide robust features for storing, organizing, or managing notes. As a result, users who wish to take comprehensive notes alongside solving problems must rely on separate applications, limiting the convenience of a unified platform. Additionally, many of these applications do not fully integrate advanced AI for mathematical reasoning, which restricts their ability to analyze complex equations or provide step-by-step solutions for more intricate problems. Tools like Wolfram Alpha, although powerful, often lack the capability to process handwritten inputs in a meaningful way, especially for non-standard notation.

Moreover, offline functionality remains a challenge for many applications, as most rely on cloud-based services to power their computational engines, making them unusable without an internet connection. This is a significant limitation for users in regions with unreliable internet access. Another issue is the complexity of user interfaces; while applications such as Mathway and Wolfram Alpha offer detailed computational features, their user interfaces can often be overwhelming for less experienced users, particularly beginners, which reduces their accessibility. Finally, although many apps, including Photomath and Mathway, provide handwritten equation recognition, their ability to accurately process freeform handwriting, particularly for more complex or unconventional equations, remains limited. This makes them less effective for users who prefer to input problems in their natural handwriting. These limitations highlight the gap in the market for an application that combines both advanced problem-solving capabilities and comprehensive note-taking, a gap that the Math Notes application seeks to address.

2.4 Generative AI

AI has been widely implemented in educational tools to improve mathematics learning. Tools such as intelligent tutoring systems (ITS) have demonstrated the ability to provide personalized assistance, adapt to individual learning needs, and offer targeted feedback. Shah et al. [3] explored the use of AI for generating complex mathematical questions, showcasing its potential to challenge students and develop critical thinking skills. However, most AI-powered tools

focus on either assessment or tutoring, with limited emphasis on integrating these features into user-friendly note-taking applications.

The integration of Artificial Intelligence (AI) in mathematics education has gained considerable attention in recent years, particularly with the advent of advanced large language models (LLMs). Moon's comprehensive benchmarking of various LLMs, including Chat GPT 4o, Claude Pro, and others, highlights their capabilities and limitations in solving calculus problems. [15] This work emphasizes the potential of AI to support procedural learning while also pointing out challenges in conceptual understanding and algebraic manipulation, crucial considerations for designing AI-enhanced math learning tools.

The performance differences among the five evaluated large language models (LLMs) reveal significant variation in their mathematical reasoning capabilities. Chat GPT 4o stood out with a remarkable 94.71% overall success rate, demonstrating a strong and generalizable understanding of mathematical principles. Its superior ability to handle complex problems and translate natural language into accurate mathematical expressions—evidenced by a 98% success rate on word problems compared to Meta AI's 4%—highlights its deeper conceptual grasp rather than reliance on pattern recognition. In contrast, Claude Pro and Gemini Advanced performed well but fell short of Chat GPT 4o, with success rates of 85.74% and 84.42% respectively. Each showed distinct strengths in specific areas, such as Claude Pro's excellence in the product rule and implicit differentiation, and Gemini Advanced's high accuracy with the power rule and conceptual problems related to function behavior.

Copilot Pro achieved a moderate overall success rate of 76.30%, showing consistent but unremarkable performance across most categories. It demonstrated a foundational understanding of mathematical reasoning without excelling in any specific area, suggesting limited advanced problem-solving ability. Meta AI, with a significantly lower success rate of 56.75%, exhibited pronounced weaknesses in mathematical reasoning. Despite perfect scores on some procedural tasks, such as limit processes and the chain rule, it struggled greatly with conceptual and complex problems, particularly word problems. This disparity points to a lack of integrative reasoning skills, emphasizing the gap between procedural competence and the conceptual understanding required for high-level mathematical reasoning in current AI systems.

To facilitate understanding of the accuracy of each AI in solving mathematical formulas, the table "Accuracy Performance on math problems of AI Model" explains the overall accuracy of each AI model in solving math problems by identifying each performance and weakness of

each AI model. By dissecting each AI model, it is hoped that it can provide a broader view of how each AI model can work well to solve a mathematical problem.

Table 1 Accuracy Performance on Math Problems of AI Models

AI Model	Overall Accuracy (%)	Performance Highlights	Notable Weaknesses
Chat GPT 4o	94.71	Highest overall accuracy; perfect scores on basic differentiation by limit process and chain rule; excels in word problems (98%) and tangent line equations (98%).	Still struggles with complex conceptual problems and multi-step reasoning; occasional algebraic manipulation errors.
Claude Pro	85.74	Strong in product rule (94%) and implicit differentiation (96%); generates most challenging problems.	Difficulty with increasing/decreasing intervals (only 23% success); struggles with conceptual connections.
Gemini Advanced	84.42	Best power rule performance (99%); relatively good at conceptual problems like increasing/decreasing intervals (70%).	Moderate conceptual limitations; fewer strengths in complex multi-step problems compared to top models.
Copilot Pro	76.30	Balanced, moderate success rates (70-90% range); performs adequately on most problem types.	Struggles in advanced problem generation, especially with complex problem types 11-13; algebraic manipulation errors frequent.

Meta AI	56.75	Perfect on basic differentiation problems (limit process and chain rule); shows competence in implicit differentiation (88%).	Performs poorly on complex problems like word problems (4%) and product rule (35%); frequent algebraic and conceptual errors.
---------	-------	---	---

One notable development is the use of generative AI models, such as ChatGPT, in aiding mathematical problem-solving. [8] The paper explores the potential of AI in enhancing mathematical capabilities, highlighting how tools like ChatGPT can assist in solving complex mathematical problems by providing step-by-step explanations, suggesting alternative solutions, and offering real-time feedback to students. [7] Other studies have demonstrated the broader potential of AI in educational tools, focusing on multilanguage frameworks for note-taking and content analysis. These frameworks highlight how AI can personalize learning, and integrating such systems with math-specific tools like ChatGPT could further enhance problem-solving and learning experiences for students.

Moreover, Koedinger and Aleven discuss the “assistance dilemma” in intelligent tutoring systems, balancing the need for AI guidance against promoting independent problem-solving skills. [19] Their insights are valuable for determining the level and nature of AI assistance integrated into math notes applications to optimize learning outcomes.

Lastly, the ability of AI tools to provide clear explanations and guided solutions is fundamental to their educational effectiveness. [20] A research into explanation generation in technology-supported math learning emphasizes that explanation quality can significantly influence students’ conceptual grasp and engagement. For a math notes app aiming to support diverse learners, incorporating such AI-driven explanatory features can greatly enhance its pedagogical value.

2.5 Use of ViewModel and LiveData in Android MVVM Architecture

The Model-View-ViewModel (MVVM) architectural pattern has gained significant traction in Android development due to its ability to separate concerns, making the app more modular, maintainable, and testable. The ViewModel and LiveData components are key components in this architecture, facilitating efficient data management and interaction between the UI and business logic.

2.5.1 The MVVM Architecture in Android

MVVM is a software architectural pattern that divides the application into three main components:

- **Model:** Represents the data layer, typically consisting of business logic and data management (such as APIs or databases).
- **View:** The UI components that display the data to the user, which in Android consists of Activities, Fragments, and Views.
- **ViewModel:** The bridge between the Model and the View. It holds and manages UI-related data in a lifecycle-conscious way. The ViewModel is responsible for preparing data for the UI and handling business logic.

In the MVVM pattern, the View interacts with the ViewModel, and the ViewModel interacts with the Model. This separation ensures that the View is only responsible for displaying data, and the ViewModel takes care of processing and managing data in response to the user's actions.

2.5.2 ViewModel in Android

The ViewModel is a key component in Android's MVVM architecture, designed to store and manage UI-related data in a way that survives configuration changes such as device rotations. The primary responsibility of the ViewModel is to provide the data to the View and to handle business logic without directly interacting with the UI elements. This helps decouple the UI from the data processing logic, improving maintainability and testability.

The ViewModel is lifecycle-aware, meaning that it is designed to survive configuration changes such as device rotations, ensuring that the UI remains consistent without needing to re-fetch data from sources such as databases or remote APIs. [5] ViewModels are typically used in conjunction with LiveData to handle data updates in a lifecycle-safe manner.

For example, in the Math Notes application, the DrawingViewModel and NoteEditorViewModel are ViewModels responsible for managing the data related to mathematical drawings and notes. These ViewModels handle tasks such as saving or updating notes and fetching equations or solutions from the backend API. The ViewModel ensures that data is preserved during configuration changes and that business logic is cleanly separated from the UI.

2.5.3 LiveData in Android

LiveData is an observable data holder class designed to be lifecycle-aware. Unlike traditional data holders, LiveData only notifies its observers (UI components like Activities or Fragments) when the data changes. Importantly, it ensures that the UI is updated only when it is in an active lifecycle state (when the activity or fragment is in the foreground). This eliminates common issues with data synchronization and lifecycle management in Android.

LiveData has several advantages, including:

- **Lifecycle Awareness:** It is automatically managed with respect to the lifecycle of the View (Activity/Fragment). The ViewModel uses LiveData to store data, and the UI can observe changes in this data.
- **Decoupling UI from Data:** Since LiveData allows UI components to observe changes without directly referencing the ViewModel, the app's UI becomes more modular and easier to maintain.
- **Automatic UI Updates:** When LiveData is updated, it notifies all active observers to update the UI accordingly, without needing to manually manage the UI update process.

In the Math Notes application, the DrawingViewModel uses LiveData to expose the drawing solutions and errors that occur when interacting with the Gemini API. When the drawing solution is ready, the LiveData in the ViewModel is updated, which triggers the UI to display the solution.

2.5.4 Integration of ViewModel and LiveData in Math Notes Application

In the Math Notes app, the ViewModel and LiveData components are used extensively to ensure efficient data management and smooth user interaction. The DrawingViewModel, for example, holds the data for mathematical equations that are being drawn on the canvas. As the user interacts with the app, the ViewModel manages all necessary changes to the data, such as updating the equation or solving it with the Gemini API.

The LiveData in this case is used to notify the View (UI components) whenever there are changes in the data, such as the availability of a solution or an error message. The UI then observes these changes and updates itself accordingly, ensuring a smooth and responsive experience for the user. This separation allows the UI to focus solely on displaying the data, while the ViewModel handles the logic of solving equations and managing the state of the app.

This pattern improves both the performance and maintainability of the app. For instance, when a user rotates the device, the ViewModel preserves the state of the current note or equation, while LiveData ensures that the UI is updated appropriately. This architecture also makes the application more testable by allowing the business logic in the ViewModel to be unit tested without needing to depend on the UI components.

2.6 Summary of Literature Review

The literature review explores the evolution of digital tools in mathematics education, emphasizing the shift from basic calculators to interactive software like GeoGebra and Desmos, which support deeper conceptual understanding through dynamic visualizations. With the advent of mobile applications, learning has become more flexible and personalized, allowing for self-directed learning and offering features such as gamification and immediate feedback to enhance engagement.

The integration of visual learning strategies and digital note-taking platforms has been found to significantly improve students' mathematical understanding. Tools that support handwritten notes and symbolic manipulations help students organize and reflect on their reasoning, leading to better problem-solving abilities and retention of concepts. The importance of minimizing cognitive load and promoting efficient learning through well-designed interfaces is also highlighted.

Intelligent Tutoring Systems (ITS) and AI-powered math solvers have transformed math learning by providing personalized, adaptive experiences. These systems offer step-by-step guidance and foster metacognitive skills by encouraging reflection on problem-solving processes. AI tools like ChatGPT, combined with natural language processing and handwriting recognition, enhance the learning experience by making problem-solving more intuitive and accessible.

A comparison of existing math applications reveals that while tools like Photomath, Microsoft Math Solver, and Mathway excel in problem-solving, they lack robust note-taking capabilities. Additionally, their ability to process complex problems and freehand input is limited. These shortcomings highlight the gap that the Math Notes application aims to fill by combining AI-driven problem-solving with note-taking functionalities, offering a comprehensive learning tool for modern math students.

In summary, the literature underscores the need for integrated tools that combine problem-solving, note-taking, and AI-powered analysis. The Math Notes application aims to address