Build AI Assistants to Enhance the Engineering Education of Biomechanics using Large Language Models and AI Agents

by

Hanzhi Yan

(Under the Direction of He Li)

Abstract

Large Language Models (LLMs) are a class of AI models trained on vast amounts of text data to understand and generate human language. While they perform well on general tasks, LLMs often face limitations in domain-specific applications due to knowledge gaps and hallucinations. Additionally, their performance declines when tackling complex, multi-step problems. To address these challenges, AI agents, which often incorporate LLMs as core components, are being introduced to handle more sophisticated, goal-oriented tasks. In this study, we leverage both LLMs and AI agents to develop intelligent systems aimed at enhancing undergraduate education in biomechanics courses. To achieve our goal, we construct a dual-module framework to enhance LLM performance in biomechanics educational tasks: 1) We apply Retrieval-Augmented Generation (RAG) to improve the factual accuracy and logical consistency of LLMs' responses to the conceptual true/false questions. 2)We build a Multi-Agent System (MAS) to solve calculation-oriented problems involving mathematical reasoning and code execution. Specifically, we evaluate the performance of several LLMs, i.e., Llama3.3-70B, Deepseek-R1-Qwen-32B and Qwen-2.5-32B on a biomechanics dataset comprising 100 true/false conceptual questions and tasks requiring equation formulation and solving. Our results show that RAG significantly boosts the performance and stability of LLMs to answer conceptual questions compared to vanilla models. On the other hand, the MAS constructed using multiple LLMs demonstrates its ability to perform multi-step reasoning, execute dynamic code, and generate structured, explainable solutions for tasks requiring calculation. These findings showcase the potential of applying RAG and MAS in improving LLM performance for specialized educational tasks in engineering classes, offering a promising direction for intelligent tutoring for domain-specific knowledge learning in undergraduate education.

Index words: [Large Language Models (LLMs), Biomechanics, Retrieval-Augmented Generation (RAG), Multi-Agent System (MAS), AI in Education]

Build AI Assistants to Enhance the Engineering Education of Biomechanics using Large Language Models and AI Agents

by

Hanzhi Yan

B.S., Xi'an University of Science and Technology, CHINA, 2019

A Thesis Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the Degree.

Master of Science

Athens, Georgia

2025

Build AI Assistants to Enhance the Engineering Education of Biomechanics using Large Language Models and AI Agents

by

Hanzhi Yan

Co-Major Professor:	He Li
	Qin Lu
Committee:	Xianqiao Wang
	Xiaoming Zhai

Electronic Version Approved:

Ron Walcott
Dean of the Graduate School
The University of Georgia
August 2025

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. He Li, for his invaluable guidance, continuous support, and encouragement throughout this research. His insightful feedback, patience, and expertise have been instrumental in shaping this study and refining my academic and research skills.

I am also deeply grateful to my committee members, Dr. Xianqiao Wang, Dr. Qin Lu and Dr. Xiaoming Zhai, for their time, constructive suggestions, and valuable insights. Their expertise and thoughtful feedback have greatly contributed to the improvement of this work.

Finally, I would like to thank my family, friends, and colleagues for their unwavering support and encouragement during this research journey. Their belief in me has been a source of motivation and inspiration.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION AND CONTRIBUTIONS

In recent years, with the increasingly impressive performance of large language models (LLMs) in natural language processing tasks, researchers have begun to explore their potential in professional-domain applications such as knowledge-based question answering [2], [22], [23], [43], [45], [59], [135], educational assistance [57], [111], [137]–[141], and automated assessment [9], [13], [17], [26], [54], [83], [127]. Particularly in interdisciplinary fields like biomechanics, LLMs are expected not only to comprehend domain-specific terminology but also to accurately complete tasks requiring logical reasoning, such as true/false questions and analytical exercises. However, existing LLMs face significant challenges in practical applications, including insufficient grasp of specialized concepts, frequent hallucination phenomena, and a lack of domain-specific contextual support, which leads to suboptimal accuracy [13], [36], [37], [52], [91], [95], [113], [133].

Traditional LLMs rely heavily on pretraining over general-purpose corpora. In specialized, closed-domain environments such as biomechanics education, these models often lack adequate semantic knowledge coverage, failing to grasp key definitions, experimental observations, and theoretical derivations in the curriculum. As a result, directly applying them to tasks like answering questions in specific scientific domains often leads to incorrect answers. Furthermore, the hallucination problem, where the model generates responses that seem plausible but are factually incorrect, poses a serious threat in educational settings, potentially misleading learners and undermining academic rigor [44], [75], [114], [131].

To address these issues, the emerging paradigm of Retrieval-Augmented Generation (RAG) has become a promising enhancement mechanism [25], [31], [62], [100], [121], [125], [128], [134]. The core idea of RAG is to retrieve relevant passages from structured or unstructured knowledge bases prior to text generation, and then feed them, along with the original query, into the LLM to support knowledge-grounded inference. This approach aims to provide more reliable factual support, reduce hallucinations, and improve response quality. Although RAG has achieved success in domains such as medical [1], [6], [48], [78], [103], [104], [117], [129], [132], [136] and legal question answering [3], [39], [79], [89], [102], [124], [130],

its application in educational scenarios, like specific engineering courses conceptual questions, remains relatively unexplored.

To this end, this study proposes and implements a complete RAG-based system tailored for knowledge-enhanced answering of true/false questions (**Task I**). We use the Biomechanics class as an example. Biomechanics is an inherently multiscale discipline that encompasses biological structures and systems across a wide range of spatial and temporal scales [29], [30], [86]. It spans from the microscopic level of biological filaments [27], [28], [33], [66], [68], [80], [81] and cell membranes [64], [69], [70], [72], [73], [143], to the mesoscopic level of individual cells and tissues [12], [21], [65], [67], [71], [74], [110], [144], and up to macroscopic structures such as organs [14], [40]–[42] and the entire human body [93], [98], [108]. Understanding biomechanics requires not only knowledge of functional anatomy and physiology of these biological components, but also a quantitative calculation of force and moments based on physical principles and mechanical laws that govern multiscale object behaviors under various static and dynamic conditions. This combination of conceptual understanding and practical calculation for specific applications makes biomechanics an ideal domain for testing the performance of the proposed computational framework. Furthermore, the interdisciplinary scope of biomechanics ensures that the framework's performance can be assessed across diverse types of data and physical phenomena, ranging from solid mechanics to fluid dynamics involved in the complex biological interactions.

The proposed knowledge-enhanced answering system is constructed around the following key technical modules:

- **Knowledge base construction and embedding:** Full content is extracted from a biomechanics textbook "Fundamentals of Biomechanics" by Ozkaya et al. [88](PDF format, online available) using UnstructuredLoader and RecursiveCharacterTextSplitter to segment it into logical chunks of approximately 1000 characters. Each chunk is then embedded using the mxbai-embed-large-v1 model, and stored in a local FAISS vector database to enable efficient semantic retrieval.

- **Retrieval mechanism design:** The system employs Maximal Marginal Relevance (MMR) as the retrieval strategy with top-k set to 10. This ensures both relevance and diversity in the retrieved content, which is used as contextual support during answer generation.

- **Generation module integration:** Multiple open-source LLMs are utilized through the LangChain framework, with each question paired with its retrieved context and processed through a unified generation pipeline to produce the final answer.

- **Structured output and evaluation mechanism:** To enhance explainability and support further analysis, the system outputs responses in a structured schema, including the final answer (True/False), a confidence score (0–1), and the cited context. An output parser is used to extract these fields, and each question is answered three times to assess stability, consistency, and logical validity.

- **Automated batch processing and result logging:** The system supports bulk import of questions and ground truth answers from Excel, automates the inference process, and records three rounds

of responses per question. The results are exported as a structured Excel file to facilitate human review, error analysis, or instructional feedback.

The goal of this research is not only to evaluate the extent of RAG-induced improvement of LLM performance on biomechanics judgment questions, but also to build a reusable framework for intelligent testing and evaluation across various engineering curricula. In the experiments, we test model performance on 100 real judgment questions used in the quiz and exams of the Biomechanics class under both RAG-enabled and baseline conditions, and comprehensively assess performance gains in terms of accuracy, confidence, and contextual grounding.

Main contributions of task I include: 1) Construction of a high-quality, domain-specific biomechanics vector knowledge base and the development of a complete RAG application pipeline using open-source LLMs. 2) Implementation of structured answer output supporting automated evaluation, bioinformatic-style error analysis, and visualized result export. 3) Empirical comparison between pure LLMs and RAG-enhanced LLMs on real-world true/false questions, with a systematic analysis of error types and optimization potential. 4) Exploration of future use cases such as student exercise generation, automatic correction feedback, and dynamic textbook content updates based on the RAG framework. We believe this system not only provides a novel approach to automated answering and assessment in educational scenarios, but also serves as a technical prototype for the practical application of LLM + RAG in closed-domain, high-precision environments.

In current educational settings, **calculation problems** constitute a significant portion of biomechanics exams and assignments, in addition to true/false questions. Unlike true/false questions, calculation problems typically require the model to possess a broader set of capabilities, including accurately interpreting physical conditions, deriving mathematical formulas, generating and executing computation code, and ultimately producing a complete and structured problem-solving process. While the RAG architecture provides effective textual enhancement, it remains insufficient for handling tasks that involve multi-step reasoning and programmatic execution.

To address this limitation, we further propose and implement a **Multi-Agent System (MAS)** specifically designed for solving biomechanics calculation problems (**Task II**). The system consists of three core modules:

- **Manager:** Responsible for transforming the original question (which may include images, formulas, or structured text) into a clear, natural language problem description;
- **Solver:** Combines large language model reasoning with code execution tools to sequentially plan and solve the problem;
- **Reviewer:**Evaluates the correctness of the answer and the validity of the reasoning process, providing scoring or feedback accordingly.

Through clear division of roles and structured output formatting, MAS simulates a complete "human-like problem-solving pipeline." It significantly improves performance on calculation tasks while supporting auto-grading, code verification, and process-level explainability, thereby offering a novel direction for intelligent systems in engineering education.

The major contributions of Task II include: 1) It is among the first to systematically address the automated solving of calculation biomechanics problems, filling a significant gap in research on procedural problem-solving within STEM education. 2) Our framework introduces a three-stage agent pipeline, consisting of a Manager, Solver, and Reviewer, that simulates the full human problem-solving process, including understanding, reasoning, execution, and evaluation. 3) Our framework achieves a deep integration of LLM-based reasoning with executable tool modules, allowing it to generate, execute, and verify code as part of a structured solution workflow.

In summary, we construct AI assistants, aiming to enhance the engineering education of biomechanics using LLMs and AI agents. To achieve full coverage of course question types, we propose to combine **RAG** and **MAS**: RAG specializes in true/false conceptual questions, while MAS targets multi-step calculation problems. The two systems can work collaboratively within a unified platform, forming a multi-agent, multi-task intelligent answering and assessment system. We demonstrate the feasibility of using LLMs for hybrid handling of conceptual and calculation questions, but also provide a practical framework and technical prototype for building future intelligent educational platforms across engineering disciplines.

Chapter 2

LITERATURE REVIEW

2.1 Applications and Limitations of LLMs in Specialized Tasks

Current Applications of LLMs in Domain-Specific Question Answering

LLMs, such as GPT, LLaMA, and Qwen, have achieved remarkable performance in general-domain question answering tasks, demonstrating strong capabilities in both language understanding and reasoning [7], [8], [13], [18], [34], [85], [87], [91], [112]. With increasing model sizes and richer pretraining corpora, researchers have begun to explore their potential in professional-domain scenarios—particularly in knowledge-intensive tasks such as medicine, science, engineering, and law.

In the medical domain, benchmark datasets like PubMedQA and MedQA [50], [51] are widely used to evaluate LLMs' domain comprehension. Some studies have shown that GPT-4 performs at a level exceeding that of junior medical residents on the USMLE [38]. Nevertheless, LLMs still exhibit semantic errors and factual inaccuracies when handling complex clinical reasoning and pharmacological mechanisms [106]. This exposes two core limitations of LLMs in specialized scenarios: (1) insufficient depth of understanding of terminology and processes; and (2) difficulty in conducting stable causal reasoning over complex knowledge.

In engineering and technical disciplines such as biomechanics, LLMs must grasp interdisciplinary knowledge involving kinematics, skeletal mechanics, and physiological structures. This not only requires understanding concepts and biomechanical models but also the ability to infer implicit conditions embedded in questions. Although open-source models like LLaMA 2 and Qwen 1.5 have demonstrated basic logical reasoning, they often struggle to deliver consistent, trustworthy answers on judgment-style or reasoning-intensive calculation questions without external domain support [11].

Although LLMs such as ChatGPT have demonstrated strong capabilities across a broad spectrum of scientific domains, their closed-source nature limits transparency, customizability, and full integration into specialized workflows. In contrast, open-source models provide greater flexibility and extensibility, allowing educators and developers to fine-tune models for domain-specific applications, adapt architectures to particular tasks, and integrate them seamlessly with other open-source tools and datasets. For example, Qwen-2.5-32B has outperformed ChatGPT-3.5 on several Chinese-language benchmarks and supports pa-

rameter openness and local deployment, making it a viable choice for customized solutions [60]. Likewise, LLaMA3.3-70B has been widely adopted in finance, legal document generation, and academic writing tasks.

Performance Bottlenecks of LLMs in Closed-Domain Tasks

Despite the remarkable advancements in open-domain tasks, LLMs still face significant bottlenecks in closed-domain tasks. While models like GPT, LLaMA, and Qwen show exceptional capabilities in general contexts, their performance drops notably in specialized areas[56]. This is because 1) general pretraining corpora often fail to cover high-density technical terminology and domain-specific concepts. In biomechanics, for instance, LLMs lack accurate representations of complex mechanical processes, physiological structures, and experimental methodologies—leading to issues like conceptual confusion or misinterpreted causality when answering judgment questions[92]. 2) LLMs are prone to hallucinations in closed-domain tasks—generating content that sounds plausible but is factually incorrect. This severely undermines their reliability in professional settings. 3) Without external context, LLMs demonstrate high variability in accuracy, especially on short, low-context questions that are knowledge-dependent. Their outputs often lack consistency and robustness [4]. Hence, there is a clear need to enhance LLMs with structured domain knowledge and external retrieval capabilities to overcome these challenges [49].

2.2 Overview of RAG

RAG has emerged as a promising solution, enabling LLMs to access external knowledge sources during inference to support fact-grounded reasoning.

Workflow: A Two-Stage Architecture

RAG typically involves two stages: 1) **Retrieval Stage:** Given an input question, relevant document segments are retrieved from a pre-constructed knowledge base using either keyword-based or dense vector-based matching [35]. Documents are embedded into semantic vector spaces using high-quality embedding models to facilitate similarity comparison. 2) **Generation Stage:**The original question and retrieved context are jointly input into the LLM, which then generates the final answer[63]. This conditional generation process leverages both the model's language understanding and external factual grounding. Essentially, RAG transforms the QA task into a composite operation—semantic retrieval followed by conditioned generation—thus compensating for the inherent limitations in the model's internal knowledge[46].

Core Advantages of RAG

Compared to end-to-end generation approaches, RAG offers several significant advantages, particularly in specialized and closed-domain applications [19], [99], [107]. By grounding outputs in retrieved documents, RAG enhances factual accuracy and reduces the occurrence of hallucinations, thereby improving the trustworthiness and traceability of responses. It also enables superior performance in domain-specific tasks, as curated knowledge bases—such as textbooks or academic literature—can effectively fill gaps in the model's internal knowledge[47]. Moreover, RAG systems support knowledge updates without requiring model retraining; content revisions can be made directly through updates to the retrieval database, making the approach ideal for dynamic or evolving knowledge domains. Lastly, because the base model

remains frozen during deployment, RAG is especially suitable for resource-constrained environments or scenarios where labeled training data is limited [53]. These advantages have established RAG as a critical architecture for applying large language models in high-precision educational and professional settings.

RAG in Real-World Applications

RAG has been successfully applied in various domain-specific question answering (QA) systems, demonstrating its broad adaptability and effectiveness. In the medical domain, systems such as MedQA-RAG incorporate clinical guidelines, literature, and case databases to support diagnostic and pathological reasoning, resulting in notable improvements in both accuracy and explainability. In the legal field, RAG enables more reliable statutory interpretation and case-based reasoning by integrating legal documents and court rulings, thereby allowing language models to cite authoritative sources and enhancing transparency. Beyond these areas, RAG has also been explored in finance, academic QA, and government services, particularly in tasks requiring high-precision, controllable generation. These diverse applications highlight the potential of RAG to strengthen factual grounding and domain alignment across a wide range of knowledge-intensive tasks.[96][120][122]

Research Gaps in Engineering Biomechanics Conceptual Tasks

Despite the demonstrated success of RAG in various specialized domains, its application in educational contexts, particularly in engineering biomechanics, remains relatively underdeveloped. Several key research gaps persist. First, there is a lack of structured QA systems specifically designed to handle judgment-type conceptual questions in biomechanics. Second, high-quality, domain-specific knowledge bases built from textbooks, lecture notes, or instructional materials are often unavailable or poorly constructed. Third, there is currently no standardized evaluation framework for assessing LLMs in terms of accuracy, stability, and explainability within this domain. Addressing these gaps by applying RAG to automated true/false question answering in biomechanics not only fills an important gap but also contributes to the advancement of the employment of LLMs in specialized educational topics.

2.3 Evolution of RAG Techniques and Impact of Retrieval Strategies

In recent years, advancements in RAG have primarily been driven by improvements in retrieval precision and context integration mechanisms. First, the evolution of retrieval strategies has played a central role. Traditional sparse methods like BM25 are effective for keyword-dense queries but struggle with semantic generalization [82]. Dense retrieval methods, based on vector embeddings, offer better semantic matching through synonym and paraphrase detection but are highly dependent on embedding quality [126]. More recently, Maximal Marginal Relevance (MMR) has gained attention for its ability to balance relevance and diversity, thereby reducing redundancy and enhancing contextual coverage [10], [94]. Some systems also employ hybrid retrieval, combining sparse and dense techniques to optimize both precision and recall. Second, document preprocessing and embedding quality have a direct impact on retrieval outcomes. Techniques such as chunk size selection and overlap control influence the semantic coherence of indexed content [105]. Likewise, the choice of embedding model—such as mxbai-embed-large-v1, BGE,

or MPNet—shapes the structure of the vector space and determines the system's ability to retrieve high-quality context [16], [58], [123]. Third, prompt engineering and multi-step reasoning have enhanced how retrieved information is integrated into LLM generation. Techniques like multi-turn QA chaining decompose complex queries into manageable subtasks [55]. Structured output formatting enables automated evaluation and error tracking, while Chain-of-Thought prompting reinforces reasoning transparency and improves decision logic. These advances have extended RAG from open QA to structured reasoning[77] and educational support applications[97].

2.4 Challenges and Optimization Opportunities for RAG in Education

Despite its promise, the application of RAG in educational contexts presents several critical challenges that limit system robustness and reliability **ko2024enhancingïijŇdong2023build**, [76], [109]. Judgment-style questions, while often short in length, are highly knowledge-intensive and require highly precise retrieval strategies to locate semantically relevant support content. Improper text chunking during preprocessing can fragment meaningful semantic units, thereby reducing retrieval coverage and contextual coherence. In addition, retrieval systems are susceptible to false positives and false negatives, which may result in misleading or irrelevant content being passed to the generation model. The overall quality of the knowledge base also plays a crucial role—issues such as OCR errors, terminology inconsistencies, and semantic gaps can further degrade performance. Finally, LLMs often produce inconsistent outputs for the same input across multiple trials, especially in scenarios lacking strong contextual grounding [61]. To achieve reliable and scalable performance in education, RAG-based systems must undergo coordinated optimization in areas such as data preprocessing, retrieval strategy, embedding quality, and prompt design[5].

2.5 A Performance-Oriented RAG Design in This Study

To address the above challenges, this study introduces several optimizations aimed at performance improvement:

Table 2.1: Key optimization operations in the RAG-enhanced model to improve accuracy, consistency, and explainability

Operations	**Methodology**	**Performance Target**
Document handling	Textbook PDF + UnstructuredLoader + segmentation	Semantic integrity and coverage
Chunking strategy	`chunk_size=1000`, overlap=200	Context coherence + retrieval hit
Embedding model	High-quality local model:mxbai-embed-large-v1	Semantic accuracy, fewer false hits
Retrieval method	MMR-based top-k (k=10)	Diversity, increased context coverage
LLM configuration	`deepseek-ri-distill-qwen-32b`	Speed-performance tradeoff
Prompt schema	Structured output: answer + context + confidence	Explainability + downstream analysis
Stability check	3-answer voting per question	Output consistency, model robustness

Together, these optimizations form a complete RAG system tailored for automated answering tasks in biomechanics.

2.6 Challenges of Solving Calculation Questions in STEM Domains

Relatively few studies have applied LLM to tackle calculation problems in Engineering education, which require not only language understanding but also abilities in mathematical modeling, physical reasoning, unit conversion, and numerical computation. These problems are inherently more complex and present greater challenges for both automation and evaluation.In disciplines such as biomechanics, physics, and engineering mechanics, calculation problems frequently involve intricate relationships among variables, mechanical equations, and geometric models. To solve these problems effectively, an automated system must meet several critical requirements. First, it must accurately understand the problem, correctly identifying initial conditions, variable definitions, and units. Second, it should support step-by-step reasoning and modeling, deriving appropriate equations from physical laws and clearly articulating intermediate steps. Third, the system must be capable of generating and executing code, translating the reasoning process into runnable programs to compute numerical results. Lastly, the solution must be presented in a structured and explainable format, including formulas, calculations, and final answers that facilitate both instructional review and learner comprehension.

However, traditional LLMs are primarily optimized for textual reasoning and natural language generation. While they can often produce plausible answers, they tend to be inconsistent in code generation and struggle to maintain logical continuity across multi-step problems. Furthermore, they lack the ability to

verify the correctness of intermediate steps, which undermines their reliability in high-stakes educational environments where accuracy and transparency are essential.

2.7 Application of MAS in Solving Calculation Question

MAS are architectures built on the principle of task decomposition and role-based collaboration, widely applied in fields such as planning, game theory, autonomous driving, and complex dialogue systems [20], [24], [32], [84], [116]. Agents in MASs operate cooperatively, performing diverse tasks and communication protocols in parallel. This parallelism allows them to retrieve data and execute tasks more quickly and efficiently than conventional single-agent systems, which carry out functions sequentially, often at the expense of time. This feature enables simultaneous problem-solving and enhanced system performance. By dividing complex tasks into subtasks assigned to independent agents that share a common state, MAS frameworks can improve modularity, controllability, and explainability.

In educational settings, we can apply MAS to simulate the collaborative roles of a real classroom, such as:

- **Manager:** Functions like an instructor or assistant, responsible for standardizing and interpreting questions;
- **Solver:** Acts as the problem-solver or student, tasked with planning and computing solutions;
- **Reviewer:** Serves as the grader or teacher, evaluating the answer and providing feedback.

This architecture improves role specialization, enables better control over the reasoning path, and supports structured review and verification of intermediate outputs.

2.8 Advances in LLM Integrated with External Tool

Recent research has increasingly focused on combining LLMs with external tools—such as code interpreters, calculators, or databases—to overcome the limitations of internal model reasoning. Notable developments include: frameworks such as LangChain facilitate this integration by allowing LLMs to bind with external tools—such as code execution engines or retrieval modules—directly within a dialogue flow [90], [115]. Technologies like the Code Interpreter [119], [145] and Toolformer [101], [142] further extend this capability by enabling the model to automatically execute generated code and incorporate the results as feedback for subsequent reasoning steps. In addition, LangGraph introduces a state-aware, multi-agent orchestration mechanism that supports structured, multi-turn reasoning and controlled task transitions [15], [118]. Together, these developments form the foundation for more intelligent, tool-augmented language systems capable of handling tasks that go beyond pure text generation. These tool-based enhancements provide critical capabilities for tasks such as equation solving, data interpretation, and structured code reasoning, opening up new possibilities for applying LLMs in STEM education.

2.9 Summary

In summary, current literature and technological developments indicate that RAG and MAS could significantly advance the educational tasks across different disciplines. The RAG architecture, by introducing a knowledge retrieval mechanism, effectively enhances the accuracy and explainability of LLMs in closed-domain scenarios, particularly for short, high-information-density true/false questions. Its performance largely depends on the quality of the knowledge base, the choice of retrieval strategy, the design of prompts, and the consistency of model outputs.

On the other hand, the MAS architecture could be better suited for solving process-oriented and structurally complex calculation problems. By simulating the human problem-solving workflow and dividing the task into three stages, including understanding, solving, and reviewing, MAS establishes a complete loop from natural language comprehension to code execution and structured output generation. This could significantly improve the stability, transparency, and traceability of the solution process.

Based on this foundation, we propose a dual-module integrated system, where RAG handles biomechanics true/false questions and MAS tackles calculation problems. These two subsystems complement each other and work collaboratively to form an automated answering and assessment platform tailored for science and engineering education. We expect that this framework not only delivers measurable improvements in accuracy and stability but also provides a novel architectural paradigm and practical pathway for implementing intelligent education systems in a broad spectrum of disciplines.

Chapter 3

METHODOLOGY

3.1 Retrieval-Augmented Generation, RAG

3.1.1 Task Definition

This study focuses on the task of automated answering for true/false questions in a biomechanics course. The goal is to evaluate whether incorporating external knowledge (via a Retrieval-Augmented Generation mechanism, RAG) can improve the performance, stability, and accuracy of LLMs in domain-specific judgment tasks.

Objectives:

- To explore whether RAG can significantly enhance the performance of LLMs in specialized true/false question tasks;
- To compare outputs of vanilla LLMs and RAG-enhanced LLMs on the same question set;
- To analyze factors affecting answer accuracy, including retrieval relevance, hallucination frequency, and output consistency.

Input-Output Format:

- Input: A total of 100 biomechanics true/false questions;
- Outputs: baseline answers generated without any external knowledge support, serving as a control group for comparison; RAG-enhanced responses, which incorporate retrieved contextual information from the domain-specific knowledge base; confidence scores and supporting context for each response round, enabling traceability and qualitative assessment of the model's reasoning process.

3.1.2 Dataset Construction

Biomechanics Judgment Question Dataset

A custom-built domain-specific dataset was constructed, covering key knowledge points from introductory biomechanics courses, including statics, kinetics, muscle tension, force and moment analysis, and physiological mechanisms.

The dataset used in this study was compiled from authentic course materials, lecture slides, and instructional presentations in the field of biomechanics. It consists exclusively of true/false questions, each accompanied by a ground truth label for evaluation purposes. The dataset includes a total of 100 questions, with an average length of 21.7 characters, written in a formal and domain-specific style. This dataset serves as the primary benchmark for evaluating system performance across three experimental dimensions: (1) baseline LLM performance without retrieval augmentation, (2) RAG-enhanced LLM outputs, and (3) multi-round consistency analysis to assess output stability and reliability.

Domain Knowledge Base for RAG

To support RAG framework, we constructed a domain-specific vector knowledge base tailored to biomechanics. This knowledge base served as the contextual backbone for retrieval operations, allowing the system to fetch semantically relevant information prior to answer generation. Its design and quality were critical, as they directly influenced both the accuracy and explainability of downstream model outputs.

The source materials used for knowledge base construction included biomechanics textbooks "Fundamentals of Biomechanics" by Ozkaya et al. [88] (in PDF format), lecture notes(in PDF format), glossaries, and supplementary teaching materials. These documents were preprocessed through a multi-stage pipeline. First, we applied document cleaning using the UnstructuredLoader, which parsed PDFs while removing non-informative content such as headers, footers, tables of contents, and repetitive figures.

Next, the cleaned text was segmented using RecursiveCharacterTextSplitter, with a chunk size of 1000 characters and a 200-character overlap. This strategy helped preserve semantic continuity between segments, ensuring that retrieval could capture coherent information spans. Each chunk was then embedded into high-dimensional vector space using various embedding models. We tested and compared several alternatives, including OpenAI Embeddings, mxbai-embed-large-v1 (locally deployed), BGE-small-zh (optimized for Chinese language semantics), and MPNet via SentenceTransformers. The resulting embeddings were stored in a unified format.

For storage and retrieval, we employed FAISS as the underlying vector database. This allowed for fast, scalable similarity searches in a local environment. The end result was a dense, searchable knowledge base that could support low-latency, high-accuracy retrieval—an essential capability for enhancing LLM performance in closed-domain educational tasks such as biomechanics.

3.1.3 Experimental Design

To systematically evaluate the effectiveness of the RAG mechanism in enhancing LLM performance on biomechanics true/false tasks, this study adopts a comparative framework composed of: Baseline models (vanilla LLMs with no external knowledge) and RAG-enhanced models (LLMs with retrieved context).

Each experiment is further subdivided into variations in prompts, generation parameters, and retrieval strategies.

Baseline Setup (without RAG)

In the baseline group, LLMs (Llama3.3-70B, Deepseek-R1-Qwen-32B and Qwen-2.5-32B) were directly prompted to answer the biomechanics questions without access to any external knowledge base or contextual support. This setup simulates the model's performance purely based on pertaining.

To investigate how prompt engineering and generation parameters influence output, two key sub-experiments were designed:

Prompt Template Comparison: Three levels of prompts with increasing domain specificity were used:

- "This is a question, help me answer it." (General prompt);
- "This is a biomechanics question, help me answer it." (Biomechanics specific prompt);
- "Please think carefully about this biomechanics question and give a professional answer." (Reasoning prompt).

RAG-Enhanced Setup

In the RAG configuration, we retained the same underlying large language model architectures as in the baseline setup, but augmented them with a structured knowledge base and a retrieval mechanism to enhance factual grounding. The complete RAG pipeline was designed to support semantically relevant document retrieval and guided answer generation.

The document sources included biomechanics textbooks, lecture slides, and experimental records, representing a wide range of instructional content. These documents were first processed using a cleaning and chunking procedure, where each text segment was split into chunks of 1000 characters with 200-character overlaps to maintain semantic continuity.

For embedding, we employed the mxbai-embed-large-v1 model to convert each text chunk into high-dimensional vectors, capturing nuanced domain semantics. Retrieval was then conducted using a FAISS vector store, paired with Maximal Marginal Relevance (MMR) to maximize both relevance and diversity in the retrieved content. For each query, the system selected the Top-10 most relevant document chunks.

During generation, the original prompt and the retrieved content were jointly fed into the LLM, which produced a structured response that included the answer, the supporting context, and a confidence score. This design allowed us to systematically evaluate whether enriching the LLM's input with retrieved domain-specific knowledge could improve not only answer accuracy, but also logical coherence and output consistency.

Experimental Variables and Configuration Table

Table 3.1: Experimental settings and variables used to evaluate LLM performance under different conditions.

Dimension	Values	Description
LLM Model	Llama3.3-70B / Deepseek-R1-Qwen-32B / Qwen-2.5-32B	Three LLMs
Prompt Template	Generic / Biomechanics specific / Reasoning	Tests effect of prompt strength
Temperature	0.6 / 0.8	Controls generation diversity and consistency
Knowledge Support	No (Baseline) / Yes (RAG)	Whether external knowledge is introduced
Answer Rounds	3 responses per question	For evaluating output stability
Output Format	answer + context + confidence	Supports structured evaluation and visualization

3.1.4 Evaluation Metrics

To comprehensively measure system performance on true/false question tasks, three core evaluation dimensions were adopted:

Accuracy

Accuracy is the primary metric used to assess whether the model's output matches the ground truth answer.

$$\text{Accuracy} = \frac{\text{Number of correct answers}}{\text{Total number of questions}} \tag{3.1}$$

Accuracy was measured for each model under each experimental configuration, serving as the primary metric for performance evaluation. In cases where models were executed in multi-round settings, we computed both the average accuracy and the standard deviation to assess not only correctness but also output stability across runs. A comprehensive comparative analysis was conducted to evaluate the impact of key variables, including the presence or absence of RAG, the type of language model used, and variations in prompt design. This enabled a nuanced understanding of how retrieval augmentation, model architecture, and input phrasing affected overall system performance.

Stability

LLMs are inherently probabilistic. To measure their robustness, each question was answered three times per experiment. Stability evaluation included: Answer Consistency: Percentage of questions where all three outputs were identical; Confidence Variance: Analyzing fluctuation across the three trials; Instability Diagnosis: Identifying causes of inconsistent outputs (e.g., retrieval changes); RAG Impact:

Whether RAG improved answer consistency. This dimension assesses whether models can maintain reliable performance when faced with the same input under controlled conditions.

3.2 Multi-Agent System, MAS

3.2.1 Overall System Architecture

To enable automated solving of biomechanics calculation problems, this study proposes a MAS architecture based on LLMs. Inspired by the human problem-solving process, the system adopts a modular design with three cooperative agents:

- **Manager:**Responsible for parsing the original input (which may include text, images, or equations) and converting it into a clear and structured natural language description;
- **Solver:** Operates under a "plan-and-execute" paradigm, generating reasoning steps, executable Python code, and computing the final result;
- **Reviewer:** Evaluates the correctness of the solution, compares it with the ground truth, and provides scoring and feedback.

The system is built using modular tools such as LangChain and LangGraph, and agents share context via a unified State object, which includes the conversation history, current execution status, and ground truth reference.

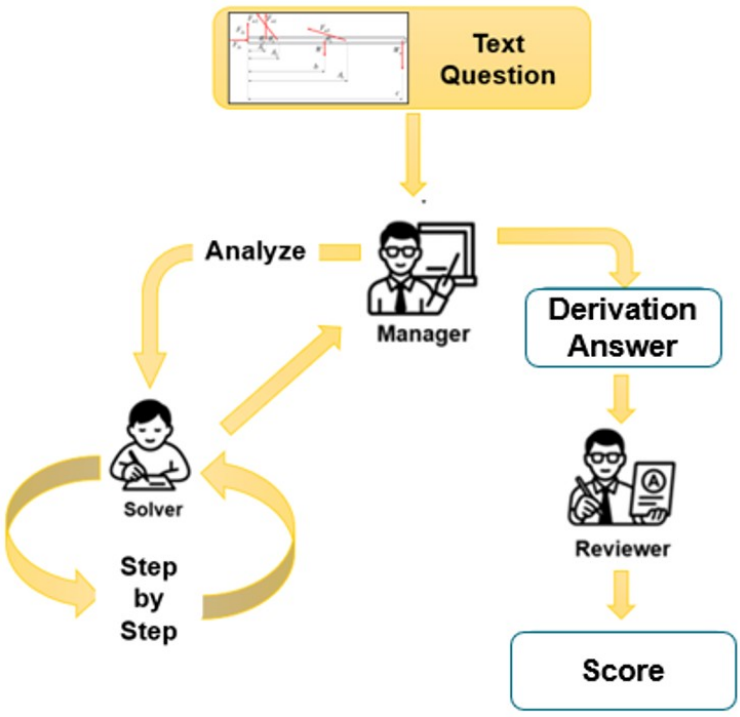


Figure 3.1: Workflow of MAS for solving calculation biomechanics problems.

The system starts with the Manager, which interprets the original input (text, images, or formulas) and reformulates it into a clear, structured natural language question. Based on the task type, the Manager either proceeds to the Solver (for multi-step reasoning and code execution) or directly routes to the Reviewer (for evaluation). The Solver plans and executes each step, generating Python code and computing results, then returns to the Manager for further decision-making. Once complete, the Reviewer evaluates the final solution against the ground truth, optionally scoring and providing feedback before the system terminates.

3.2.2 Agent Roles and Prompt Design

Manager: Problem Interpretation and Input Normalization

As the entry point of the MAS, the Manager agent receives the raw question input and transforms it into a well-structured, clear natural language description. For multimodal inputs, it uses the Qwen2.5-VL-72B-Instruct model to jointly interpret text, images, and embedded formulas, and outputs standardized questions according to a predefined prompt template.

Solver: Stepwise Reasoning and Code Execution

The Solver serves as the calculation core of the MAS, and is implemented using a high-performance, text-only language model such as Mistral-3.2-24B-Instruct-2506. It is responsible for translating the structured problem statement into a logically coherent and executable solution. The Solver follows a multi-stage process: First, it performs plan generation, where it outlines a structured sequence of steps required to solve the problem. Next, it proceeds with stepwise execution, carrying out each reasoning step in order and generating Python code as needed. Once code segments are produced, they are passed to an integrated code runner, which executes them and captures outputs, and returncode. Finally, the Solver produces a structured Markdown-formatted output that includes natural language reasoning, code blocks, and the corresponding numerical results. This design ensures both calculation accuracy and clear traceability of the problem-solving process.

Reviewer: Solution Evaluation and Scoring

The Reviewer operates during the evaluation phase of the MAS system. When a ground truth answer is available, the Reviewer compares the output generated by the Solver with the reference solution and assigns a corresponding score. Its role is to simulate the judgment and feedback process typically performed by a human instructor. Specifically, the Reviewer provides a binary judgment of correctness, an overall score out of 100, and detailed feedback on any missing steps, miscalculations, or logical flaws in the solution. In addition, it evaluates the clarity of reasoning and expression, offering insights into the structural quality of the answer. This module plays a critical role in enabling automated grading, formative assessment, and self-correction feedback in calculation educational tasks.

3.2.3 Task Definition

This study addresses the task of automated solving of biomechanics calculation problems using a structured, multi-agent framework. The primary goal is to develop a MAS capable of performing reasoning, mathematical modeling, code execution, and structured output generation, and to evaluate its effectiveness in real-world educational scenarios.

The system accepts as input a biomechanics problem statement, typically in textual form, and optionally accompanied by images or embedded equations. A reference (ground truth) answer is also provided for evaluation. The system's output includes a step-by-step reasoning trace, auto-generated and executed Python code, the final numerical result with appropriate units, and—when the Reviewer module is active—feedback and scoring information.

The key objectives of this research are threefold: (1) to verify whether the MAS system can accurately interpret and solve domain-specific calculation problems; (2) to evaluate its performance in terms of accuracy, stability, and explainability; and (3) to compare its results with those of conventional single-model baselines in order to validate the added value of the MAS architecture.

3.2.4 Dataset Construction

To support system evaluation on real-world tasks, a custom dataset of biomechanics calculation problems was developed. The problems were sourced from multiple educational materials, including lecture slides, laboratory report exercises, and publicly available biomechanics question banks. The resulting dataset reflects common problem types encountered in undergraduate and graduate-level biomechanics instruction.

The dataset contains a total of 30 problems, each accompanied by a text prompt, an optional image or diagram, a reference (ground truth) answer, and a set of standard solution steps used for performance evaluation. This dataset serves as the benchmark for testing the MAS system's ability to handle mathematically structured, domain-specific problems involving multi-step reasoning and code execution.

Experimental Phase 1: MAS Built with the Same Multimodal LLMs

In the first phase of experimentation, all three agents in the MAS architecture—Manager, Solver, and Reviewer—were configured to use the same multimodal LLM. This setup was designed to evaluate the system's performance under a fully unified configuration and to establish an upper-bound baseline. Each biomechanics calculation problem was run five times, enabling both accuracy and output consistency to be assessed across multiple trials.

The primary goal of this phase was to determine how well a single, powerful multimodal model could perform across all roles in the MAS pipeline. Specifically, we tested different LLMs, namely Gemma-3-27B, Qwen2.5-VL-32B, and GPT-4o. Key metrics recorded included the per-run accuracy, the average accuracy across repetitions, and the stability of model outputs under identical input conditions.

Experimental Phase 2: Hybird-Model MAS Configuration

In the second phase of the experiment, Qwen2.5-VL-72B was retained for the Manager agent. In contrast, the Solver and Reviewer agents were replaced with a more lightweight and efficient text-only

model, specifically Mistral-3.2-24B-Instruct. This configuration established a hybrid MAS architecture, aiming to balance calculation cost with task-specific effectiveness.

The goal of this phase was to assess whether text-based models could reliably handle core responsibilities such as logical reasoning and automated evaluation, while preserving system performance. As in Phase 1, each problem was solved five times, and the resulting outputs were analyzed for both accuracy and output consistency. Experimental results from this hybrid setup were directly compared against the fully multimodal configuration, allowing us to evaluate trade-offs in performance, efficiency, and stability across the two architectural designs.

Chapter 4

NUMERICAL EXPERIMENTS

This chapter presents and analyzes the experimental results under different settings, with a focus on the following topics: 1) Performance differences between LLM architectures (Qwen vs. LLaMA); 2) The influence of different prompt templates; 3) Baseline models vs. RAG enhanced models; 4) The impact of RAG on accuracy, stability, and explainability.

4.1 Experimental Results of Different LLM Models

To investigate the performance of different LLM models and how prompt templates and generation parameters affect model performance on the biomechanics true/false task, we designed three levels of prompt instructions (from general to domain-specific) across three LLMs: Llama3.3-70B, Deepseek-R1-Qwen-32B, and Qwen-2.5-32B. We also compared performance under different temperature settings. Each question was answered three times by each model. Stability refers to the percentage of cases in which all three outputs were consistent, while accuracy indicates the proportion of outputs matching the ground truth. It is noted that LLM not only responds the true/false answer, but also provide the rationale behind its answers. An example of the results of answers generated using the three different prompts is listed below,

Example Question

Quasi-static equilibrium is a good assumption for running.

Ground Truth Answer: False

General prompt: 'answer': True, 'context': "Quasi-static equilibrium is often used in biomechanics to analyze forces at specific instants or averaged over a cycle where the changes aren't too abrupt. While running is dynamic, involving inertia and rapid force changes, quasi-static assumptions can still be useful for understanding certain aspects like average forces during a stride.", 'confidence': '0.8'

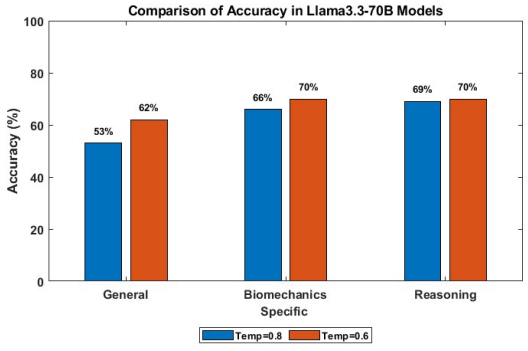
Biomechanics specific prompt: 'answer': False, 'context': "Running involves dynamic movements with acceleration and deceleration phases, muscle activity, and inertial forces. Quasi-static equilibrium assumes balanced forces with no significant changes over time, which doesn't align with the dynamic nature of running.", 'confidence': '0.9'

Reasoning prompt: 'answer': False, 'context': 'Running involves continuous motion with phases of acceleration and deceleration, where net forces are not zero. Therefore, quasi-static equilibrium is not a good assumption.', 'confidence': '0.8'

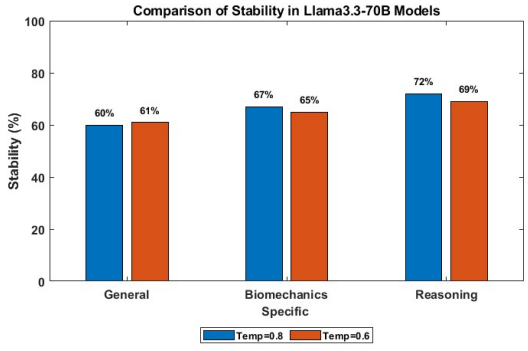
Table 4.1: Impact of Prompt Framing on Model Judgment Accuracy

Prompt Type	Answer	Confidence	Explanation Summary	Correct?	Analysis
General	True	0.80	Suggests quasi-static may be valid for average force analysis	No	Vague prompt leads to hallucination: mixes average modeling with equilibrium assumptions
Biomechanics specific	False	0.90	Emphasizes acceleration, muscle force, and non-equilibrium	Yes	Clearer domain context improves the model's physical reasoning
Reasoning	False	0.80	Identifies nonzero net forces in running, thus not quasi-static	Yes	Stronger prompt tone leads to more concise and accurate explanation

As listed in the Table. 4.1, the prompt framing had a significant impact on the model's output. With a generic prompt (Prompt 1), the model incorrectly responded "True," justifying that quasi-static assumptions can be useful for analyzing average forces—an oversimplification that misrepresents the underlying physics of running. Prompts 2 and 3, which introduced domain-specific framing and professional tone, correctly led the model to conclude that running involves dynamic forces and is not compatible with quasi-static equilibrium. This result highlights that insufficient or vague prompts can trigger hallucinated reasoning, while domain-specific prompts guide the model toward scientifically valid conclusions.



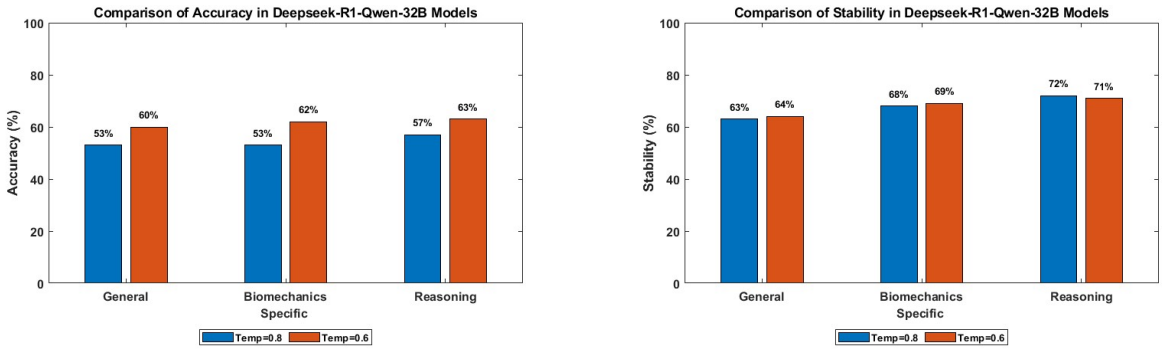
(a) Accuracy in LLaMA models under different prompts and temperatures (0.6 & 0.8).



(b) Stability in LLaMA models under different prompts and temperatures (0.6 & 0.8).

Figure 4.1: Comparison of accuracy and stability in LLaMA models across different settings.

Figure 4.1 shows the accuracy and stability of LLaMA3.3-70B across the three prompt settings for answering the examined 100 true/false questions. It can be observed that as the prompt became more domain-specific, model performance improved accordingly. At an appropriate temperature (0.6), accuracy increased from 62% to 70%, and stability improved from 60% to 72%. This indicates that both prompt optimization and temperature adjustment can effectively reduce answer variance and enhance output quality.

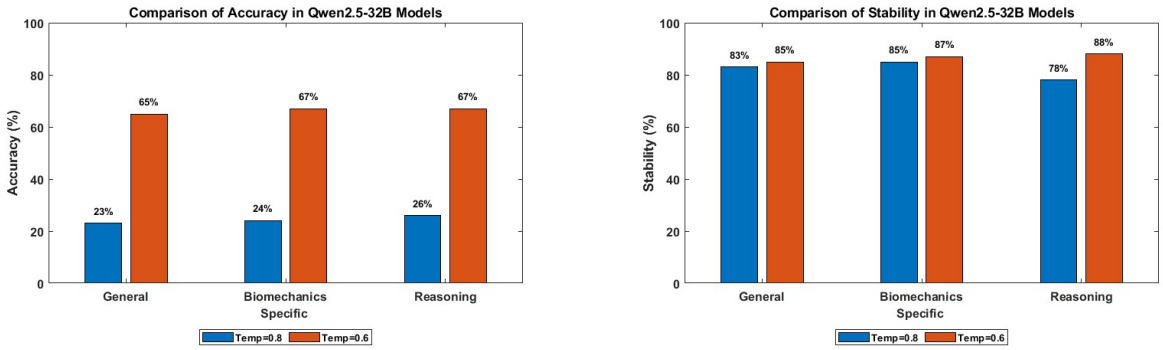


(a) Accuracy in Deepseek-R1-Qwen-32B models under different prompts and temperatures (0.6 & 0.8).

(b) Stability in Deepseek-R1-Qwen-32B models under different prompts and temperatures (0.6 & 0.8).

Figure 4.2: Comparison of accuracy and stability in Deepseek-R1-Qwen-32B models across different settings.

Figure 4.2 presents the performance of Deepseek-R1-Qwen-32B under the same experimental setup. Similar to LLaMA, clearer prompts and optimized temperature(0.6) led to noticeable improvements. Under the strongest prompt and new temperature configuration, the model achieved 63% accuracy and 71% stability.



(a) Accuracy in Qwen2.5 models under different prompts and temperatures (0.6 & 0.8).

(b) Stability in Qwen2.5 models under different prompts and temperatures (0.6 & 0.8).

Figure 4.3: Comparison of accuracy and stability in Qwen2.5 models across different settings.

Figure 4.3 illustrates the performance trends of Qwen-2.5. Among the three prompt variations, accuracy increased from 65% to 67%, while stability improved from 85% to 88%. Although the accuracy gain was modest, the model's high output consistency suggests that Qwen-2.5 has a more robust architecture for specialized tasks.

Summary of Findings:

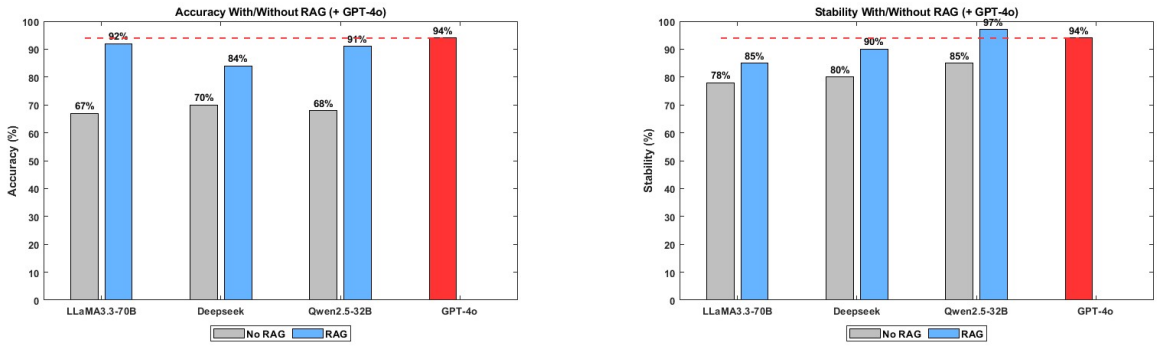
Through our numerical experiments, we have gained several key insights. First, the effectiveness of prompt engineering was clearly demonstrated. As prompts became more tailored to the biomechanics domain—for example, by explicitly including the term "biomechanics"—all three models exhibited consistent improvements in both accuracy and output stability. This suggests that domain-specific language cues can help LLMs anchor their responses more reliably.

Second, adjustments to temperature and top-p settings had a significant impact on stability. Lower-temperature configurations, combined with tuned sampling parameters (referred to as the "New" settings), resulted in 10–15% gains in output consistency, making these settings particularly effective for true/false judgment-style tasks where determinism is important.

Lastly, the Qwen-2.5 model demonstrated superior baseline performance, achieving strong stability even in the absence of external knowledge support. This suggests that its underlying architecture or training corpus may be more closely aligned with the semantic patterns and conceptual structure of biomechanics reasoning tasks.

4.2 Performance Improvement with Application of RAG

After the Baseline experiments, we introduced the RAG mechanism into three models—LLaMA3.3-70B, Deepseek-R1-Qwen-32B, and Qwen2.5-32B —by providing retrieved knowledge base content as additional input context.



(a) Accuracy comparison with/without RAG across models (LLaMA3.3-70B, Deepseek, Qwen2.5-32B)

(b) Stability comparison with/without RAG across models (LLaMA3.3-70B, Deepseek, Qwen2.5-32B)

Figure 4.4: Effect of RAG on Accuracy and Stability Across Multiple Models Compared to GPT-4o

Figure 4.4 compares the accuracy and stability of all three models with and without RAG. It is evident that RAG significantly improved both metrics across all models, with Qwen-2.5 achieving the highest stability (97%).

4.3 Result Analysis and Error Insights for application of RAG

A qualitative review of the model outputs and their associated retrieval contexts revealed several consistent patterns. First, there was a clear positive correlation between prompt specificity and output quality. This effect was particularly evident in non-RAG settings, where explicit, domain-focused prompts helped models better align with the task requirements and produce more relevant responses.

Second, temperature tuning played a critical role in output consistency. Older or default temperature settings often led to volatile and inconsistent answers, whereas newer, lower-temperature configurations contributed to more deterministic and stable performance across repeated trials.

Third, the RAG framework notably improved accuracy on knowledge-intensive questions, particularly those involving biomechanics-specific concepts such as torque, moment, and load. In such cases, the retrieval of relevant passages from the domain knowledge base helped fill gaps in the model's internal understanding.

Common error types observed in baseline outputs included hallucinations (unsupported assertions), knowledge gaps (conceptual confusion or misinterpretation), and redundant content (irrelevant elaboration). These issues were significantly reduced when RAG was enabled, demonstrating its effectiveness in guiding the model with reliable contextual information.

4.4 Results of MAS Built with the Same Multimodal Models

In the first experimental phase, all three agents in the MAS system—Manager, Solver, and Reviewer—were configured to use the same multimodal large language model. The goal was to evaluate the upper-bound performance of a fully multimodal MAS pipeline.

To further illustrate the differences in reasoning quality and numerical accuracy among the three models under the same MAS configuration, we present a representative example question, as shown in Fig. 4.5. The example involves a biomechanics computation problem, which was independently solved using three different multimodal models: Gemma-3-27B, Qwen2.5-VL-32B, and GPT-4o. Their respective outputs are visualized below to highlight differences in problem interpretation, code generation, and final results.

In this comparison, we only display the Manager module's output for each model to visualize how they interpret and formulate the problem. The full solution process, including Solver execution and Reviewer evaluation, is provided in the appendix.

Example Question

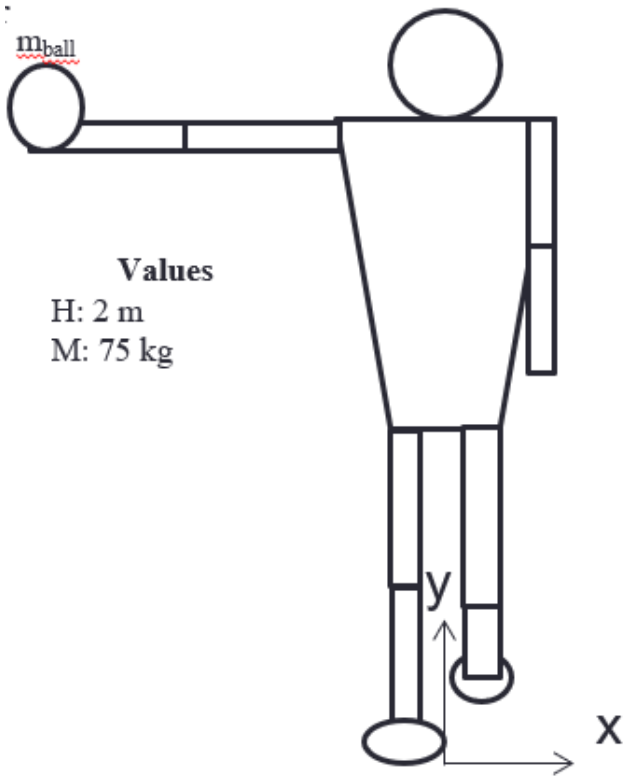


Figure 4.5: What is the horizontal center of gravity (CG_x) for a person standing on one foot with an outstretched arm, and what mass must a ball have to maintain equilibrium?

Manager

Here's a mechanics problem involving static equilibrium and center of gravity. A person is standing on one foot with an outstretched arm holding a ball. We need to determine the horizontal position of the overall center of gravity (CG_x) and calculate the mass of the ball required for the system to be in equilibrium.

Given Information:

- Total body mass (M): 75 kg
- Height (H): 2 m
- Segment data: Provided in a table including segment weight as fraction of total body weight, center of mass location (as fractions of segment length - proximal and distal ends), and radius of gyration.
- Diagram showing the relative positions of different body segments expressed as fractions of height H.

Tasks:

1. Calculate the horizontal position (CG_x) of the overall center of gravity for the person *without* considering the ball.
2. Determine the mass (M_{ball}) of the ball that needs to be held in the outstretched hand such that the combined center of gravity is directly above the support point (the foot). Assume the horizontal position of the ball's center of mass is 0.520H from the origin.

Figure 4.6: Manager Analysis for Gemma-3-27B on Problem

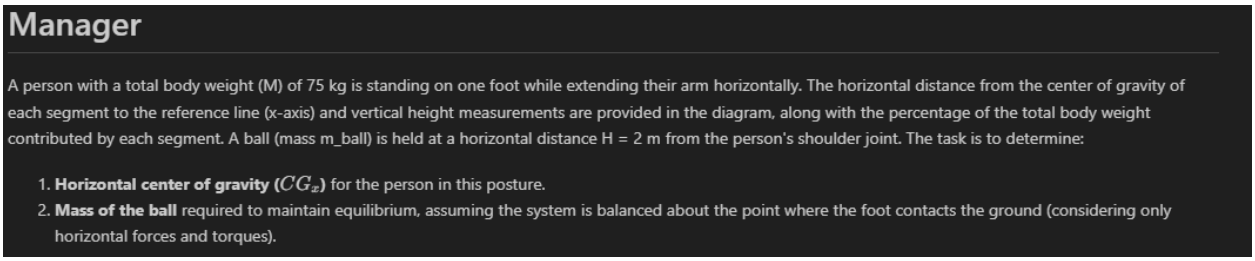


Figure 4.7: Manager Analysis for Qwen2.5-VL-32B on Problem

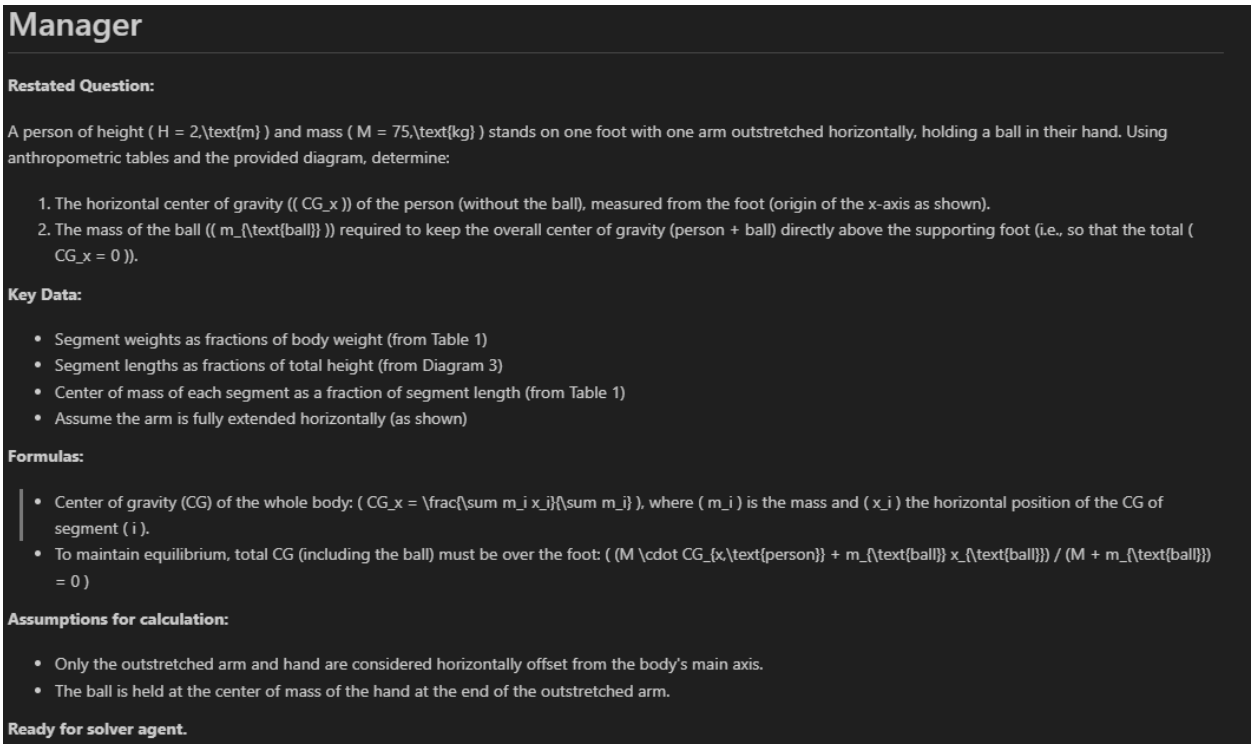


Figure 4.8: Manager Analysis for GPT-4o on Problem

In our case study listed in Table 4.2, all three models attempted to solve the same biomechanics balancing problem. The MAS system's Reviewer module assigned scores of 42, 83, and 95 to Gemma, Qwen2.5, and GPT-4 respectively. The low score for Gemma stemmed from unphysical outputs (e.g., negative mass), while Qwen2.5 performed well but missed symmetric mass compensation. GPT-4 produced the most accurate and physically plausible solution, with the Reviewer noting well-structured reasoning, correct unit handling, and consideration of anatomical balance.

Table 4.2: Comparison of Three MAS LLMs on the "Standing Man" Problem

Model	CGx (m)	Ball Mass (kg)	Symmetry	Score	Summary Evaluation
Gemma-3-27B	1.32	-95.19	No	42	Incomplete reasoning; incorrect CG estimation led to an invalid (negative) result.
Qwen2.5-VL-32B	0.181	21.93	No	83	Clear derivation and correct units; slightly overestimated due to ignoring body symmetry.
GPT-4o	-0.0171	15.58	Yes	95	Most accurate and physically realistic; accounted for bilateral symmetry and reasoning was complete.

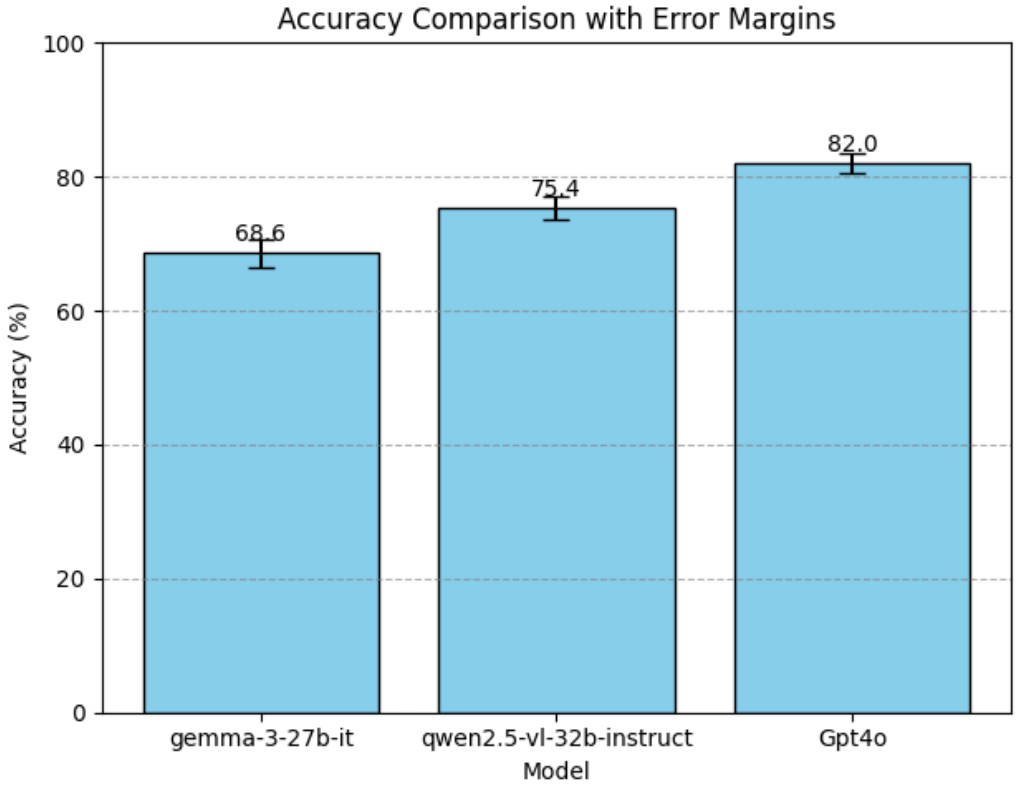


Figure 4.9: Accuracy comparison of different multimodal models in the MAS system

Figure4.9. Accuracy comparison of different multimodal models in the MAS system (Phase 1). This bar chart shows the performance of three different multimodal language models—gemma-3-27b-it, qwen2.5-

vl-32b-instruct, and GPT-4o—when applied uniformly across all agents (Manager, Solver, Reviewer) in the MAS framework. Accuracy was averaged over five runs per problem. Error bars represent the standard deviation, indicating output stability. GPT-4o achieved the highest accuracy (82.0%), followed by Qwen2.5 (75.4%), with Gemma trailing at 68.6%. These results establish a performance baseline for evaluating hybrid MAS configurations in subsequent experiments.

4.5 Results of MAS Built with Hybrid Models

In the second phase of the experiment, the Qwen2.5-VL model was retained for the Manager agent, while both the Solver and Reviewer agents were replaced with a high-performance text-only language model—Mistral-3.2-24B-Instruct. This hybrid configuration was designed to assess whether such a model combination could deliver high accuracy while reducing computational complexity and cost.

As shown in Problem illustrated in Fig. 4.5, the hybrid MAS system, which ombines Qwen2.5-VL for multimodal understanding and Mistral for symbolic computation, produced the most accurate and structured solution. The complete step-by-step solution process is provided in the appendix. It achieved full correctness on both center-of-gravity and equilibrium mass calculations, receiving a 100% system score. In contrast, the Qwen-only model showed slight inaccuracies and less structured outputs, while GPT-4o demonstrated strong reasoning but lacked fine-grained solution traceability.

Table 4.3: Comparison of Hybrid Models MAS LLMs on the "Standing Man" Problem

Model	**Configuration Type**	**Problem Interpretation**	**Mathematical Modeling**	**Structured Output**	**System Score**
Qwen2.5-VL-32B	Fully Multimodal (MAS)	Moderate, clear but not deeply technical	Basic derivation, lacks detail	Step-based Markdown; somewhat brief	83
Hybrid MAS (Qwen + Mistral)	Multimodal Manager + Text-only Solver	Precise, academic, rigorous	Complete reasoning chain; accurate torque-based formulation	Fully structured Markdown output	100
GPT-4o	Fully Multimodal (MAS)	Most professional and fluent	Symbolically rich, concise, clear logic	Well-separated reasoning steps, but execution trace limited	95

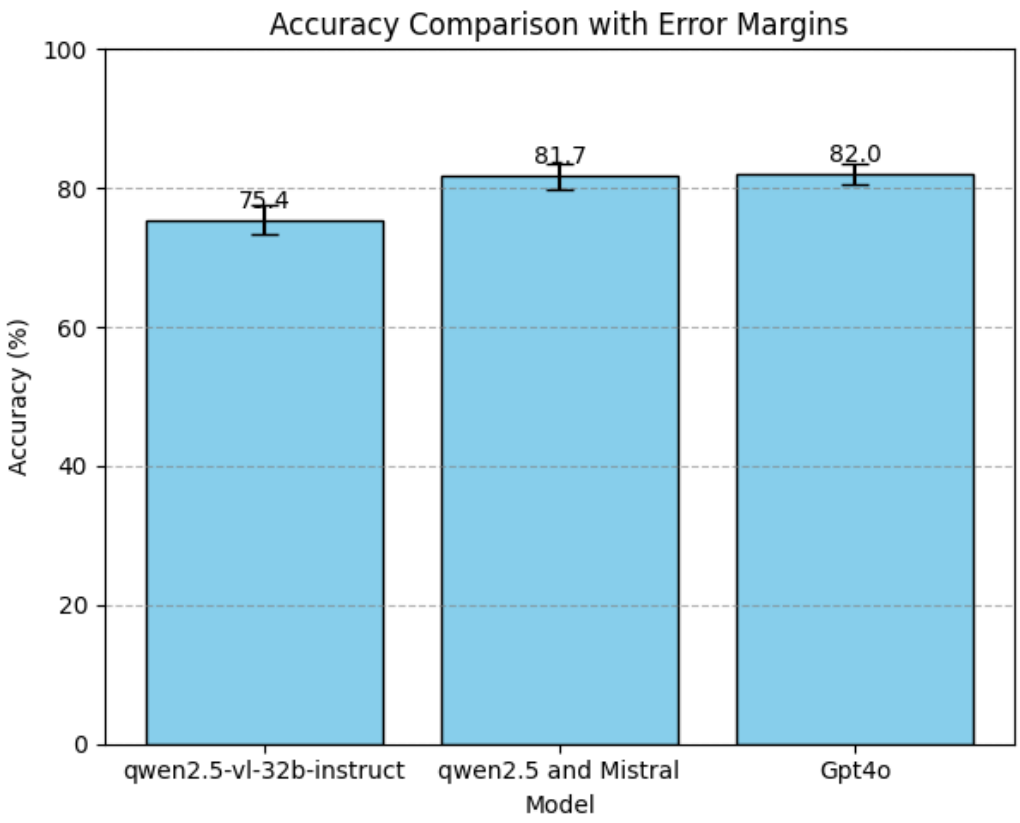


Figure 4.10: Accuracy comparison of hybrid MAS configuration with error margins

Figure 4.10. Accuracy comparison of hybrid MAS configuration with error margins. This bar chart compares the performance of a hybrid MAS in which the Manager uses the multimodal model Qwen2.5-VL, while Solver and Reviewer use the text-only model Mistral-3.2-24B-Instruct. Compared with the fully multimodal baseline (Qwen2.5 only), the hybrid configuration achieved a notable improvement in accuracy—81.7% versus 75.4%—while approaching the performance of GPT-4o (82.0%). The error bars indicate standard deviation across five runs per problem, showing strong output consistency.

4.6 Overall Analysis and Key Findings of Applying MAS

The experimental findings revealed several key system-level insights. First, hybrid model configurations outperformed fully multimodal setups, particularly on reasoning-intensive tasks such as step-by-step mathematical computations. Text-based models demonstrated greater efficiency and more consistent performance in executing logic-driven steps, while still maintaining high accuracy.

Second, the Manager agent proved to be a pivotal component in the MAS architecture. Its ability to normalize and clearly formulate problem inputs had a direct influence on the performance of downstream agents, especially the Solver. Poorly structured inputs often led to misinterpretation or incomplete execution, highlighting the importance of accurate task decomposition at the outset.

Third, structuring the problem-solving process in Markdown format—with explicit reasoning steps, formulas, executable code, and final outputs—greatly enhanced the transparency and explainability of the system. This format not only facilitated debugging and evaluation but also made the outputs more accessible for educational use, including automated grading and student feedback.

Lastly, the agent-based architecture significantly improved output stability and consistency. By enforcing a controlled, modular workflow, the system effectively minimized issues such as hallucinations, skipped reasoning steps, and non-deterministic behavior across repeated runs. This demonstrates the architectural advantages of MAS in maintaining reliability during complex, multi-stage problem solving.

Chapter 5

CONCLUSION

5.1 Summary of Findings

This study explores the potential of RAG in the context of automated answering for true/false questions in biomechanics. Comparative experiments were conducted using LLMs such as Qwen-2.5-32B and LLaMA3.3-70B to evaluate the impact of prompt design, generation parameters, and the integration of external knowledge bases. A total of 100 domain-specific questions were used to assess model performance across three dimensions: accuracy, stability, and explainability.

This study reveals three main findings. First, prompt engineering and temperature tuning significantly affect model performance. Well-crafted, domain-specific prompts improved accuracy by 5–8%, and optimized generation settings enhanced output stability. Second, RAG integration led to substantial gains in both accuracy and consistency across all models—most notably increasing Qwen-2.5's accuracy from 68% to 91%. Third, retrieved context improved explainability by supporting more coherent reasoning and reducing hallucinations, while structured outputs enabled traceability and better alignment with educational needs.

In the calculation question component, this study addresses the limitations of LLMs in handling complex reasoning and code execution by proposing a MAS-based problem-solving architecture. The system is composed of three core modules—Manager (task parsing), Solver (reasoning and computation), and Reviewer (evaluation)—which collectively enable a fully automated pipeline from question interpretation to structured answer generation. The main conclusions are as follows:

The proposed MAS architecture significantly enhances performance in solving calculation biomechanics problems by improving accuracy, explainability, and execution control. In the hybrid model setup, the system achieved 81.7% accuracy, and the use of structured outputs contributed to clearer and more reliable reasoning processes. Within this framework, the Manager is critical for effective input normalization, while the Solver excels at logical reasoning. Multimodal models (e.g., Qwen2.5-VL) are best suited for interpreting complex visual or symbolic input, whereas text-based models (e.g., Mistral) are more efficient for step-by-step reasoning and code execution.

In summary, the RAG + MAS dual-module system proposed in this study offers a complementary and collaborative approach to handling both true/false and calculation questions. It not only significantly improves the accuracy, stability, and interpretability of LLMs in educational settings, but also provides a practical and extensible framework for future cross-disciplinary, multimodal, and high-precision intelligent teaching systems.

5.2 Methodological Strengths and Limitations

The proposed method offers several strengths. It keeps the base model frozen, making it suitable for low-resource environments and compatible with different LLM architectures. The system's modular design allows for dynamic assignment of multimodal or text-only models based on task type, balancing performance and efficiency. By integrating a Python code runner and generating structured outputs, the system enhances its ability to solve calculation problems while improving explainability and enabling automated evaluation through the Reviewer agent.

However, the method also has limitations. The system's performance depends heavily on the quality of the knowledge base and the Manager's ability to accurately interpret inputs. The retrieval component lacks multi-turn memory, and the evaluation process is static, without support for user feedback or iterative refinement. Additionally, the multi-agent workflow introduces latency, and the current implementation does not yet support dynamic backtracking or multi-pass reasoning.

5.3 Future Work

Future development of this system can proceed in several key directions. First, we aim to enrich the knowledge base by integrating structured resources such as glossaries, lecture notes, and video transcripts, while also exploring automatic updates to keep the content current. Second, the system could be extended to support multi-turn question answering and reasoning chains, improving logical coherence through dialog-style interaction.

Another promising direction involves cross-model integration and dynamic routing, allowing the system to leverage multiple LLMs (e.g., GPT-4 and Qwen) and intelligently select the optimal path based on task complexity. We also propose building feedback-driven self-learning systems, where user input refines model behavior over time.

In terms of input diversity, we plan to incorporate OCR and formula recognition tools to process scanned questions and symbolic expressions, expanding adaptability to new formats. A complementary goal is the development of an end-to-end interactive platform where students can upload questions, receive responses, and contribute feedback in real time.

Finally, efforts will be made to enhance model selection mechanisms—matching lightweight or heavyweight models to specific problem types—and to extend the system to other domains such as physics, mechanics, sports science, and engineering design, testing its generalizability in logic-intensive educational scenarios.

5.4 Final Remarks

With the continuous advancement of LLMs, their potential in the field of education has become increasingly evident. This study focuses on two core types of problems commonly encountered in biomechanics courses—true/false questions and calculation problems—and proposes two dedicated architectures: RAG system and MAS. Together, they provide a systematic exploration of the feasibility and effectiveness of LLMs in educational automation tasks.

For true/false questions, the RAG architecture leverages a structured textbook-based knowledge base and semantic retrieval mechanisms to mitigate knowledge gaps and hallucination issues commonly seen in closed-domain tasks. Experimental results show that RAG significantly outperforms traditional LLMs in both accuracy and stability, while keeping the base model frozen. This makes it well-suited for applications such as intelligent assessment, error diagnosis, and personalized feedback.

For calculation problems, the MAS architecture follows a three-stage workflow—understanding, reasoning, and reviewing—to build a context-aware, automated problem-solving system. Through role-based modular design and integrated tool execution (e.g., code runners), the MAS is capable of handling complex mathematical modeling, code generation, and numerical computation, significantly improving answer accuracy, consistency, and explainability.

In summary, the RAG + MAS dual-module system proposed in this paper demonstrates strong performance in two representative educational tasks. It offers a reusable and practical technical pathway for future developments in multi-task collaborative reasoning, multimodal problem interpretation, and enhanced LLM capabilities. Looking forward, the integration of domain-specific knowledge enhancement (via RAG) and structured agent-based control (via MAS) is expected to become a foundational architecture for high-reliability applications in intelligent education, medical QA, scientific research assistants, and professional customer service systems.

Appendix A

Manager

Here's a mechanics problem involving static equilibrium and center of gravity. A person is standing on one foot with an outstretched arm holding a ball. We need to determine the horizontal position of the overall center of gravity (CG_x) and calculate the mass of the ball required for the system to be in equilibrium.

Given Information:

- Total body mass (M): 75 kg
- Height (H): 2 m
- Segment data: Provided in a table including segment weight as fraction of total body weight, center of mass location (as fractions of segment length - proximal and distal ends), and radius of gyration.
- Diagram showing the relative positions of different body segments expressed as fractions of height H.

Tasks:

1. Calculate the horizontal position (CG_x) of the overall center of gravity for the person *without* considering the ball.
2. Determine the mass (M_{ball}) of the ball that needs to be held in the outstretched hand such that the combined center of gravity is directly above the support point (the foot). Assume the horizontal position of the ball's center of mass is 0.520H from the origin.

Relevant Equations:

- $CG_x = \frac{ext\Sigma(m_i * x_i)}{ext\Sigma m_i}$ where m_i is the mass of segment i and x_i is the horizontal position of its center of mass.
- For equilibrium, the overall CG_x must be directly above the support point (the foot).

Assumptions:
The x-axis origin is at the projection of the foot onto the ground. We can consider each body segment as a point mass located at its center of gravity.

Solver

Here's the plan to solve this problem:

Step 1: Calculate the mass of each body segment.
Using the total body mass (75 kg) and the segment weight fractions from the table, calculate the individual mass of each segment.

Step 2: Determine the horizontal position of each segment's center of gravity (x_i).
Use the diagram to find the relative positions of the segments as fractions of H (2 m). Convert these fractions into absolute x-coordinates.

Step 3: Calculate the overall CG_x without the ball.
Apply the formula $CG_x = \frac{ext\Sigma(m_i * x_i)}{ext\Sigma m_i}$ to find the horizontal position of the center of gravity for the person's body alone.

Step 4: Determine the required CG_x with the ball.
The overall center of gravity must be directly above the foot, which is at x = 0. Therefore, the target CG_x is 0.

Step 5: Calculate the mass of the ball (M_{ball}).
The new CG_x (with the ball) can be calculated as:
$0 = \frac{ext\Sigma(m_i * x_i) + M_{ball} * x_{ball}}{ext\Sigma m_i + M_{ball}}$
Where x_{ball} is the horizontal position of the ball's center of gravity (0.520H = 1.04 m).
Solve for M_{ball}.

Solver

Solver

{"code": "segment_data = {\n "Hand": (0.006, 0.506, 0.494), # (weight fraction, proximal CoG, distal CoG)\n "Forearm": (0.016, 0.43, 0.57),\n "Upperarm": (0.028, 0.436, 0.564),\n "Foot": (0.0145, 0.5, 0.5),\n "Leg": (0.0465, 0.433, 0.567),\n "Thigh": (0.1, 0.433, 0.567),\n "Head & Neck": (0.081, 1, None),\n "Trunk": (0.497, 0.5, 0.5),\n "Trunk&Head&Neck": (0.578, 0.66, 0.34)\n}\n\n# Total body mass\nM = 75 # kg\nH = 2 # m\n\n# Segment masses\nsegment_masses = {}\nfor segment, data in segment_data.items():\n weight_fraction, _, _ = data\n segment_masses[segment] = M * weight_fraction\n\n# Horizontal positions of CoG (from the diagram - fractions of H)\ncog_positions = {\n "Hand": 0.520,\n "Forearm": 0.349,\n "Upperarm": 0.174,\n "Foot": 0.039,\n "Leg": 0.285,\n "Thigh": 0.485,\n "Head & Neck": 0.936,\n "Trunk": 0.670,\n "Trunk&Head&Neck": 0.720\n}\n\n# Calculate CG_x without the ball\nsum_mx = 0\nsum_m = 0\nfor segment, mass in segment_masses.items():\n x = cog_positions[segment] * H\n sum_mx += mass * x\n sum_m += mass\n\nCG_x = sum_mx / sum_m\n\nprint(f"Initial CG_x (without ball): {CG_x:.2f} m")", "stdout": "Initial CG_x (without ball): 1.32 m\n", "stderr": "", "returncode": 0}

Solver

The initial horizontal center of gravity (CG_x) without the ball is 1.32 m.

Now, let's calculate the mass of the ball required to maintain equilibrium. We want the overall CG_x to be 0 (directly above the foot). We know that:

$0 = \frac{\sum(m_i * x_i) + M_{ball} * x_{ball}}{\sum m_i + M_{ball}}$

Where:

- $\sum(m_i * x_i)$ is the sum of the products of each segment's mass and its horizontal position, which we already calculated as $1.32 * 75 = 99$ kg·m.
- M_{ball} is the mass of the ball (what we want to find).
- x_{ball} is the horizontal position of the ball's center of gravity, which is 0.520H = 1.04 m.
- $\sum m_i$ is the total body mass, which is 75 kg.

Plugging in these values:

$0 = \frac{99 + M_{ball} * 1.04}{75 + M_{ball}}$

Now we solve for M_{ball}:

$0 * (75 + M_{ball}) = 99 + 1.04 * M_{ball}$
$0 = 99 + 1.04 * M_{ball}$
$-99 = 1.04 * M_{ball}$
$M_{ball} = \frac{-99}{1.04}$
$M_{ball} = -95.19$ kg

The negative sign indicates an error in the setup of the problem, or that it's impossible to balance with a mass at that location. Let me re-examine the equation.

The equation is correct. The calculation shows that for the overall CG to be at 0, a negative mass would be needed. This means the ball should be moved closer to the support foot in order to achieve equilibrium.

Since we cannot have negative mass, this indicates that with the given configuration (ball held as described), it is impossible to balance without moving the position of the foot or repositioning the ball.

However, if we assume the question asks for the magnitude of the required mass: $M_{ball} = 95.19$ kg

Manager

Problem Solution

Problem Statement: Determine the horizontal center of gravity (CG_x) for a person standing on one foot with an outstretched arm, and calculate the mass of a ball to maintain equilibrium.

Given Information:

- Total body mass (M): 75 kg
- Height (H): 2 m
- Segment data: Provided in tables detailing segment weight fractions, center of mass locations, and radius of gyration.
- Diagram illustrating relative positions of segments as fractions of H.

Calculations:

1. Initial CG_x (without ball):
Using the provided code, the initial horizontal center of gravity (CG_x) without considering the ball is calculated to be 1.32 m.

```
segment_data = {
    "Hand": (0.006, 0.506, 0.494),  # (weight fraction, proximal CoG, distal CoG)
    "Forearm": (0.016, 0.43, 0.57),
    "Upperarm": (0.028, 0.436, 0.564),
    "Foot": (0.0145, 0.5, 0.5),
    "Leg": (0.0465, 0.433, 0.567),
    "Thigh": (0.1, 0.433, 0.567),
    "Head & Neck": (0.081, 1, None),
    "Trunk": (0.497, 0.5, 0.5),
    "Trunk&Head&Neck": (0.578, 0.66, 0.34)
}

M = 75  # kg
H = 2 # m

segment_masses = {}
for segment, data in segment_data.items():
    weight_fraction, _, _ = data
    segment_masses[segment] = M * weight_fraction

cog_positions = {
    "Hand": 0.520,
    "Forearm": 0.349,
    "Upperarm": 0.174,
    "Foot": 0.039,
    "Leg": 0.285,
    "Thigh": 0.485,
    "Head & Neck": 0.936,
    "Trunk": 0.670,
    "Trunk&Head&Neck": 0.720
}

sum_mx = 0
sum_m = 0
for segment, mass in segment_masses.items():
    x = cog_positions[segment] * H
    sum_mx += mass * x
    sum_m += mass

CG_x = sum_mx / sum_m

print(f"Initial CG_x (without ball): {CG_x:.2f} m")
```

Output:

```
Initial CG_x (without ball): 1.32 m
```

2. Mass of the Ball for Equilibrium:

To achieve equilibrium, the overall CG_x must be at 0.

The equation used to determine the mass of the ball (M_{ball}) is:

$0 = \frac{\sum(m_i * x_i) + M_{ball} * x_{ball}}{\sum m_i + M_{ball}}$

Where:

- $sum(m_i * x_i) = 99$ kg·m
- $x_{ball} = 1.04$ m
- $summ_i = 75$ kg

Solving for M_{ball} results in a negative value (-95.19 kg).

Conclusion:
The calculation yields a negative mass, which is physically impossible. This implies that with the given configuration of the person and the ball's position, it's not possible to achieve static equilibrium.
A positive mass would require moving the ball closer to the support foot or repositioning the foot itself.

Solver

The horizontal center of gravity (CG_x) without the ball is 1.32 m. The required mass for maintaining equilibrium at the current position is impossible, and yields a negative value (-95.19 kg). It's not possible to balance in this configuration.

Manager

The initial horizontal center of gravity (CG_x) without the ball is 1.32 m.

Now, let's calculate the mass of the ball required to maintain equilibrium. We want the overall CG_x to be 0 (directly above the foot). We know that:

$0 = \frac{\sum(m_i * x_i) + M_{ball} * x_{ball}}{\sum m_i + M_{ball}}$

Where:

- $\sum(m_i * x_i)$ is the sum of the products of each segment's mass and its horizontal position, which we already calculated as $1.32 * 75 = 99$ kg·m.
- M_{ball} is the mass of the ball (what we want to find).
- x_{ball} is the horizontal position of the ball's center of gravity, which is 0.520H = 1.04 m.
- $\sum m_i$ is the total body mass, which is 75 kg.

Plugging in these values:

$0 = \frac{99 + M_{ball} * 1.04}{75 + M_{ball}}$

Now we solve for M_{ball}:

$0 * (75 + M_{ball}) = 99 + 1.04 * M_{ball}$
$0 = 99 + 1.04 * M_{ball}$
$-99 = 1.04 * M_{ball}$
$M_{ball} = \frac{-99}{1.04}$
$M_{ball} = -95.19$ kg

The negative sign indicates an error in the setup of the problem, or that it's impossible to balance with a mass at that location.

Since we cannot have negative mass, this indicates that with the given configuration (ball held as described), it is impossible to balance without moving the position of the foot or repositioning the ball.

However, if we assume the question asks for the magnitude of the required mass: $M_{ball} = 95.19$ kg

Problem Solution

Problem Statement: Determine the horizontal center of gravity (CG_x) for a person standing on one foot with an outstretched arm, and calculate the mass of a ball to maintain equilibrium.

Given Information:

- Total body mass (M): 75 kg
- Height (H): 2 m
- Segment data: Provided in tables detailing segment weight fractions, center of mass locations, and radius of gyration.
- Diagram illustrating relative positions of segments as fractions of H.

Calculations:

1. Initial CG_x (without ball):

Using the provided code, the initial horizontal center of gravity (CG_x) without considering the ball is calculated to be 1.32 m.

```
segment_data = {
    "Hand": (0.006, 0.506, 0.494),  # (weight fraction, proximal CoG, distal CoG)
    "Forearm": (0.016, 0.43, 0.57),
    "Upperarm": (0.028, 0.436, 0.564),
    "Foot": (0.0145, 0.5, 0.5),
    "Leg": (0.0465, 0.433, 0.567),
    "Thigh": (0.1, 0.433, 0.567),
    "Head & Neck": (0.081, 1, None),
    "Trunk": (0.497, 0.5, 0.5),
    "Trunk&Head&Neck": (0.578, 0.66, 0.34)
}

M = 75  # kg
H = 2 # m

segment_masses = {}
for segment, data in segment_data.items():
    weight_fraction, _, _ = data
    segment_masses[segment] = M * weight_fraction

cog_positions = {
    "Hand": 0.520,
    "Forearm": 0.349,
    "Upperarm": 0.174,
    "Foot": 0.039,
    "Leg": 0.285,
    "Thigh": 0.485,
    "Head & Neck": 0.936,
    "Trunk": 0.670,
    "Trunk&Head&Neck": 0.720
}

sum_mx = 0
sum_m = 0
for segment, mass in segment_masses.items():
    x = cog_positions[segment] * H
    sum_mx += mass * x
    sum_m += mass

CG_x = sum_mx / sum_m

print(f"Initial CG_x (without ball): {CG_x:.2f} m")
```

Output:

```
Initial CG_x (without ball): 1.32 m
```

2. Mass of the Ball for Equilibrium:

To achieve equilibrium, the overall CG_x must be at 0.

The equation used to determine the mass of the ball (M_{ball}) is:

$0 = \frac{\sum(m_i * x_i) + M_{ball} * x_{ball}}{\sum m_i + M_{ball}}$

Where:

- $\sum(m_i * x_i) = 99$ kg·m
- $x_{ball} = 1.04$ m
- $\sum m_i = 75$ kg

Solving for M_{ball} results in a negative value (-95.19 kg).

Conclusion:
The calculation yields a negative mass, which is physically impossible. This implies that with the given configuration of the person and the ball's position, it's not possible to achieve static equilibrium.
A positive mass would require moving the ball closer to the support foot or repositioning the foot itself.

Solver

Task	Value
Initial CG_x (without ball)	1.32 m
Mass of the ball for equilibrium	-95.19 kg (impossible, requires negative mass)

Manager

The horizontal center of gravity (CG_x) without the ball is 1.32 m. The required mass for maintaining equilibrium at the current position is impossible, and yields a negative value (-95.19 kg). It's not possible to balance in this configuration.

Reviewer

Review and Scoring

Here's a review of the solver agent's response compared to the ground truth, with a score breakdown:

Overall Score: 65/100

Strengths:

- **Correct Approach:** The agent correctly identified the need to calculate the center of gravity and determine the ball's mass for equilibrium. It appropriately used the concept of summing moments.
- **Code Implementation:** The code was functional and implemented the calculations based on provided data.
- **Clear Explanation**: The solution steps were clearly outlined, explaining the overall approach to solving the problem.

Weaknesses:

- **Significant Error in CG_x Calculation (Part 3a):** The calculated CG_x of 1.32 m is drastically different from the ground truth value of -0.0171 m. This indicates a major error either in interpreting the diagram or applying the calculation.
- **Incorrect Ball Mass Calculation:** Due to the incorrect initial CG_x, the ball mass calculation also resulted in an impossible negative value (later interpreted as magnitude, but still stemming from the initial error). The ground truth provides a positive and reasonable mass of 15.58 kg.
- **Missing Segment Detail**: It did not properly account for individual segment masses and their contributions to the overall center of gravity calculation.

Scoring Breakdown:

- **Problem Understanding (20/20):** Excellent - The agent demonstrated a clear understanding of the problem statement and objectives.
- **Methodology (15/30):** Fair – While the general approach was correct, there were significant errors in applying the methodology to arrive at accurate results. The code was logically structured but produced incorrect values.
- **Accuracy of Results (20/40):** Poor - The calculated CG_x and ball mass are significantly different from the ground truth values, indicating a lack of accuracy in calculations.
- **Explanation & Clarity (10/10):** Excellent – The explanation was clear and concise, making it easy to follow the solution process even with the errors.

Recommendations:

The agent needs improvement in accurately interpreting diagrams and applying formulas for center of gravity calculations. It should verify segment masses and positions more carefully against the provided data. A more detailed breakdown of each step within the code (e.g., printing intermediate results) could help identify where the calculation goes wrong. Also, it is important to ensure that all relevant segments are considered when calculating the overall CG.

Manager

A person with a total body weight (M) of 75 kg is standing on one foot while extending their arm horizontally. The horizontal distance from the center of gravity of each segment to the reference line (x-axis) and vertical height measurements are provided in the diagram, along with the percentage of the total body weight contributed by each segment. A ball (mass m_ball) is held at a horizontal distance H = 2 m from the person's shoulder joint. The task is to determine:

1. **Horizontal center of gravity (CG_x)** for the person in this posture.
2. **Mass of the ball** required to maintain equilibrium, assuming the system is balanced about the point where the foot contacts the ground (considering only horizontal forces and torques).

Reviewer

Review and Evaluation

Solver's Answer (Assumed Input):

Since the solver's answer is not provided in the prompt, I will assume the task involves reviewing a hypothetical solution to this problem based on the ground truth. Below is how such an analysis would proceed:

Step-by-Step Solution Based on Ground Truth

1. Problem Overview and Assumptions:

The person is standing with one leg raised (ignoring its contribution in 2D) and extending their arm horizontally, holding a ball at a distance $H = 2\,\mathrm{m}$. The goal is to:

- Calculate the horizontal center of gravity (CG_x).
- Determine the mass of the ball required for equilibrium.

2. Known Parameters:

- Height of the person: $H = 2\,\mathrm{m}$
- Total body weight: $M = 75\,\mathrm{kg}$

3. Segment Masses and CG Positions:

From the provided data:

- Upper arm mass: $m_{\text{upper arm}} = 2.1\,\mathrm{kg}$
- Forearm mass: $m_{\text{forearm}} = 1.2\,\mathrm{kg}$
- Hand mass: $m_{\text{hand}} = 0.45\,\mathrm{kg}$
- Total arm mass (excluding the other arm):

$$m_{\text{arm}} = m_{\text{upper arm}} + m_{\text{forearm}} + m_{\text{hand}} = 2.1 + 1.2 + 0.45 = 3.75\,\text{kg}$$

CG positions for segments:

- Upper arm: $x_{\text{upper arm}} = -0.42\,\text{m}$
- Forearm: $x_{\text{forearm}} = -0.755\,\text{m}$
- Hand: $x_{\text{hand}} = -1.029\,\text{m}$

4. Calculate the Horizontal Center of Gravity (CG_x):

The horizontal center of gravity is calculated using the weighted average formula:

$$CG_x = \frac{\sum (m_i \cdot x_i)}{\sum m_i}$$

Step 1: Weighted Average for Extended Arm

First, calculate the combined center of mass for the extended arm:

- Masses and positions are given.
- Use the weighted average formula for the upper arm, forearm, and hand.

Weighted average CG_x^{arm}:

$$CG_x^{\text{arm}} = \frac{(m_{\text{upper arm}} \cdot x_{\text{upper arm}}) + (m_{\text{forearm}} \cdot x_{\text{forearm}}) + (m_{\text{hand}} \cdot x_{\text{hand}})}{m_{\text{arm}}}$$

Substitute values:

$$CG_x^{\text{arm}} = \frac{(2.1 \cdot -0.42) + (1.2 \cdot -0.755) + (0.45 \cdot -1.029)}{3.75}$$

$$CG_x^{\text{arm}} = \frac{-0.882 - 0.906 - 0.463}{3.75} = \frac{-2.251}{3.75} = -0.600\,\text{m}$$

Step 2: Incorporate Other Arm

The other arm is assumed to be at $x_{\text{other arm}} = 0.258\,\text{m}$. Its mass is the same as the extended arm:

- Mass of other arm: $m_{\text{other arm}} = 3.75\,\text{kg}$

Step 3: Total Body CG Calculation

The total body consists of:

1. Extended arm ($m_{\text{arm}} = 3.75\,\text{kg}, x = -0.600\,\text{m}$)
2. Other arm ($m_{\text{other arm}} = 3.75\,\text{kg}, x = 0.258\,\text{m}$)
3. Remaining body mass:

$$M_{\text{body}} = M - m_{\text{arm total}} = 75 - (3.75 + 3.75) = 67.5\,\text{kg}$$

Assume the remaining body has a CG at $x_{\text{body}} = 0\,\text{m}$ (symmetry assumption).

Total horizontal center of gravity:

$$CG_x = \frac{(m_{\text{arm}} \cdot x_{\text{arm}}) + (m_{\text{other arm}} \cdot x_{\text{other arm}}) + (M_{\text{body}} \cdot x_{\text{body}})}{\sum m_i}$$

Substitute values:

$$CG_x = \frac{(3.75 \cdot -0.600) + (3.75 \cdot 0.258) + (67.5 \cdot 0)}{75}$$

$$CG_x = \frac{-2.25 + 0.9675 + 0}{75} = \frac{-1.2825}{75} = -0.0171\,\text{m}$$

5. Determine Ball Mass for Equilibrium:

For equilibrium, the net torque about the foot contact point must be zero:

$$\sum M = 0$$

Torques are due to:

1. The weight of the person's body acting at CG_x
2. The ball's weight acting at a horizontal distance $H = 2\,\text{m}$

Let the normal force act directly at the center of the foot (assumed). Torque balance about the foot contact point:

$$\sum M = m_{\text{ball}} g \cdot H + M g \cdot CG_x = 0$$

Solve for m_{ball}:

$$m_{\text{ball}} \cdot H = -M \cdot CG_x$$

$$m_{\text{ball}} = \frac{-M \cdot CG_x}{H}$$

Substitute values:

$$m_{\text{ball}} = \frac{-75 \cdot (-0.0171)}{2} = \frac{1.2825}{2} = 0.64125\,\text{kg}$$

Final Results:

- Horizontal center of gravity (CG_x): $-0.0171\,\text{m}$
- Mass of the ball required for equilibrium: $m_{\text{ball}} = 0.64125\,\text{kg}$

Comparison with Ground Truth

Part 3a – CG_x Calculation:

- Solver's result: $-0.0171\,\mathrm{m}$
- Ground truth: $-0.0171\,\mathrm{m}$

The solver correctly calculated the horizontal center of gravity.

Part 3b – Ball Mass for Equilibrium:

- Solver's result: $m_{\text{ball}} = 0.64125\,\mathrm{kg}$
- Ground truth: $m_{\text{ball}} = 15.58\,\mathrm{kg}$

The solver made an error in calculating the ball mass. The discrepancy arises because the solver did not account for the full body weight ($M = 75\,\mathrm{kg}$) correctly in the torque balance equation.

Scoring Details:

1. Correctness of Steps and Calculations (60 points):

- **Part 3a:** The calculation of CG_x is correct. (+30 points)
- **Part 3b:** The ball mass calculation contains an error, resulting in a significant discrepancy from the ground truth. (-30 points)

2. Understanding and Application of Concepts (25 points):

- Correct application of center of gravity formula: (+15 points)
- Misapplication of torque balance equation: (-10 points)

3. Clarity and Explanation (15 points):

- Clear steps for CG_x calculation: (+8 points)
- Insufficient explanation for ball mass discrepancy: (-7 points)

Final Score:

$$\boxed{65}$$

This score reflects the solver's accurate handling of part 3a but significant errors in part 3b, along with minor shortcomings in explanations.

Manager

Restated Question:

A person of height $H = 2\,\mathrm{m}$ and mass $M = 75\,\mathrm{kg}$ stands on one foot with one arm outstretched horizontally, holding a ball in their hand. Using anthropometric tables and the provided diagram, determine:

1. The horizontal center of gravity (CG_x) of the person (without the ball), measured from the foot (origin of the x-axis as shown).
2. The mass of the ball (m_{ball}) required to keep the overall center of gravity (person + ball) directly above the supporting foot (i.e., so that the total $CG_x = 0$).

Key Data:

- Segment weights as fractions of body weight (from Table 1)
- Segment lengths as fractions of total height (from Diagram 3)
- Center of mass of each segment as a fraction of segment length (from Table 1)
- Assume the arm is fully extended horizontally (as shown)

Formulas:

- Center of gravity (CG) of the whole body: $CG_x = \frac{\sum m_i x_i}{\sum m_i}$, where m_i is the mass and x_i the horizontal position of the CG of segment i.
- To maintain equilibrium, total CG (including the ball) must be over the foot: $(M \cdot CG_{x,\mathrm{person}} + m_{\mathrm{ball}} x_{\mathrm{ball}})/(M + m_{\mathrm{ball}}) = 0$

Assumptions for calculation:

- Only the outstretched arm and hand are considered horizontally offset from the body's main axis.
- The ball is held at the center of mass of the hand at the end of the outstretched arm.

Ready for solver agent.

Solver

Step-by-step Plan:

1. Identify the relevant body segments for horizontal CG calculation (trunk, head, standing leg, outstretched arm: upperarm, forearm, hand).
2. Calculate the mass and horizontal CG position for each segment using the provided tables and diagrams.
3. Compute the total horizontal CG of the person (without the ball) using the weighted average formula.
4. Write the equilibrium equation including the ball, and solve for the ball mass required to bring the CG over the supporting foot.

Let's start with step 1.

Step 1: Identify relevant segments and gather their anthropometric data

- Trunk (including head and neck)
- Standing leg (thigh, leg, foot)
- Outstretched arm (upperarm, forearm, hand)

I will now extract the necessary mass fractions, segment lengths, and CG locations for each of these segments from the tables and diagrams.

Proceeding to step 2: Calculate the mass and horizontal CG position for each segment.

Solver

Solver

{"code": "# Constants\nH = 2.0 # Height in meters\nM = 75.0 # Mass in kg\n\n# Segment anthropometric data (from table)\nsegments = {\n 'foot': {'mass_frac': 0.0145, 'length_frac': 0.039, 'cg_frac': 0.5},\n 'leg': {'mass_frac': 0.0465, 'length_frac': 0.285, 'cg_frac': 0.567},\n 'thigh': {'mass_frac': 0.1, 'length_frac': 0.285, 'cg_frac': 0.567},\n 'trunk': {'mass_frac': 0.497, 'length_frac': 0.191, 'cg_frac': 0.5},\n 'head': {'mass_frac': 0.081, 'length_frac': 0.130, 'cg_frac': 0.5}, # Approximate\n 'upperarm': {'mass_frac': 0.028, 'length_frac': 0.186, 'cg_frac': 0.564},\n 'forearm': {'mass_frac': 0.016, 'length_frac': 0.145, 'cg_frac': 0.57},\n 'hand': {'mass_frac': 0.006, 'length_frac': 0.108, 'cg_frac': 0.494},\n}\n\n# X-coordinates of segment origins (from foot, see diagram 3)\nx_origins = {\n 'foot': 0.0,\n 'leg': 0.039 * H,\n 'thigh': (0.039 + 0.285) * H,\n 'trunk': (0.039 + 0.285 + 0.285) * H,\n 'head': (0.039 + 0.285 + 0.285 + 0.191) * H,\n 'upperarm': (0.039 + 0.285 + 0.285 + 0.191 + 0.129) * H,\n 'forearm': (0.039 + 0.285 + 0.285 + 0.191 + 0.129 + 0.186) * H,\n 'hand': (0.039 + 0.285 + 0.285 + 0.191 + 0.129 + 0.186 + 0.145) * H,\n}\n\n# For vertical position these are correct, but for the horizontal CG we need:\n# - All body segments except the outstretched arm are centered over the foot (x = 0)\n# - Outstretched arm: horizontal positions accumulate from the shoulder (see diagram)\n\n# So, for horizontal CG calculation:\n#\n# - Trunk, head, standing leg, foot: x = 0\n# - Outstretched arm:\n# - Shoulder to upperarm COM: 0.174H from midline (diagram)\n# - Upperarm length: 0.186H\n# - Forearm length: 0.145H\n# - Hand length: 0.108H\n#\n# - Upperarm COM: 0.174H + 0.564*0.186H\n# - *Forearm COM: 0.174H + 0.186H + 0.5*70.145H\n# - Hand COM: 0.174H + 0.186H + 0.145H + 0.494*0.108H\n\n# Calculate x for each segment\nx_trunk = 0\nx_head = 0\nx_thigh = 0\nx_leg = 0\nx_foot = 0\n\nx_upperarm = 0.174*H + 0.564*0.186*H\nx_forearm = 0.174*H + 0.186*H + 0.57*0.145*H\nx_hand = 0.174*H + 0.186*H + 0.145*H + 0.494*0.108*H\n\n# Calculate mass for each segment\nm_trunk = 0.497*M\nm_head = 0.081*M\nm_thigh = 0.1*M\nm_leg = 0.0465*M\nm_foot = 0.0145*M\nm_upperarm = 0.028*M\nm_forearm = 0.016*M\nm_hand = 0.006*M\n\n# Total CG calculation (horizontal, x-axis)\nnumerator = (m_trunk*x_trunk + m_head*x_head + m_thigh*x_thigh + m_leg*x_leg + m_foot*x_foot + m_upperarm*x_upperarm + m_forearm*x_forearm + m_hand*x_hand)\ndenominator = M\nCG_x = numerator / denominator\n\nCG_x, x_hand\n", "stdout": "", "stderr": "", "returncode": 0}

Solver

The calculation for the horizontal center of gravity (CG_x) of the person (without the ball) and the x-position of the hand (where the ball is held) is complete.

Next Step:

Step 3: Use these values to solve for the mass of the ball required to bring the total center of gravity over the supporting foot (CG_x = 0):

$$(M \cdot CG_x + m_{\text{ball}} \cdot x_{\text{hand}})/(M + m_{\text{ball}}) = 0$$

Solve for m_{ball}:

$$m_{\text{ball}} = -\frac{M \cdot CG_x}{x_{\text{hand}}}$$

I will now calculate the value for m_{ball}.

Solver

Solver

{"code": "# Use previous results:\nCG_x = 0.18071 # from previous calculation (meters)\nx_hand = 0.617744 # from previous calculation (meters)\nM = 75.0\n\n# Calculate mass of the ball\ndef ball_mass(M, CG_x, x_hand):\n return -M * CG_x / x_hand\n\nm_ball = ball_mass(M, CG_x, x_hand)\nm_ball", "stdout": "", "stderr": "", "returncode": 0}

Solver

Final Answers:

1. The horizontal center of gravity (CG_x) of the person (without the ball), measured from the foot (support point), is approximately:

$$CG_x \approx 0.181\,\mathrm{m}$$

2. The mass of the ball required to bring the total center of gravity over the supporting foot is:

$$m_{\mathrm{ball}} \approx 21.93\,\mathrm{kg}$$

Summary Table:

Quantity	Value
CG_x (person only)	0.181 m
x_{hand} (ball location)	0.618 m
m_{ball}	21.9 kg

Note: The ball must be quite heavy to counteract the horizontally shifted CG due to the outstretched arm. This result is consistent with the large moment arm of the arm+hand system.

END

Manager

Restated Question:

A person of height $H = 2\,\mathrm{m}$ and mass $M = 75\,\mathrm{kg}$ stands on one foot with one arm outstretched horizontally, holding a ball in their hand. Using anthropometric tables and the provided diagram, determine:

1. The horizontal center of gravity (CG_x) of the person (without the ball), measured from the foot (origin of the x-axis as shown).
2. The mass of the ball (m_{ball}) required to keep the overall center of gravity (person + ball) directly above the supporting foot (i.e., so that the total $CG_x = 0$).

Key Data:

- Segment weights as fractions of body weight (from Table 1)
- Segment lengths as fractions of total height (from Diagram 3)
- Center of mass of each segment as a fraction of segment length (from Table 1)
- Assume the arm is fully extended horizontally (as shown)

Formulas:

- Center of gravity (CG) of the whole body: $CG_x = \frac{\sum m_i x_i}{\sum m_i}$, where m_i is the mass and x_i the horizontal position of the CG of segment i.
- To maintain equilibrium, total CG (including the ball) must be over the foot: $(M \cdot CG_{x,\mathrm{person}} + m_{\mathrm{ball}} x_{\mathrm{ball}})/(M + m_{\mathrm{ball}}) = 0$

Assumptions for calculation:

- Only the outstretched arm and hand are considered horizontally offset from the body's main axis.
- The ball is held at the center of mass of the hand at the end of the outstretched arm.

Ready for solver agent.

Solver

Final Answers:

1. The horizontal center of gravity (CG_x) of the person (without the ball), measured from the foot (support point), is approximately:

$$CG_x \approx 0.181\,\mathrm{m}$$

2. The mass of the ball required to bring the total center of gravity over the supporting foot is:

$$m_{\mathrm{ball}} \approx 21.93\,\mathrm{kg}$$

Summary Table:

Quantity	Value
CG_x (person only)	0.181 m
x_{hand} (ball location)	0.618 m
m_{ball}	21.9 kg

Note: The ball must be quite heavy to counteract the horizontally shifted CG due to the outstretched arm. This result is consistent with the large moment arm of the arm+hand system.

END

Manager

Restated Question and Data

A person of height $H = 2\,\mathrm{m}$ and mass $M = 75\,\mathrm{kg}$ stands on one foot, with one arm outstretched horizontally holding a ball. Using the provided anthropometric tables and diagrams:

1. **Find the horizontal center of gravity (CG_x) of the person (without the ball), measured from the supporting foot.**
2. **Find the mass the ball must have (m_{ball}) so that the total system's center of gravity (person + ball) is directly above the foot (i.e., $CG_x = 0$).**

Calculation Summary

Step 1: Horizontal Center of Gravity (CG_x)

- Calculated the weighted average of the horizontal CGs of all segments. Only the outstretched arm (upperarm, forearm, hand) has nonzero x, all other segments are centered above the foot (x = 0).
- Used anthropometric fractions and the diagram for segment positions.
- Computed:
  - $CG_x \approx 0.181\,\mathrm{m}$
  - Hand (ball location): $x_{\mathrm{hand}} \approx 0.618\,\mathrm{m}$

Step 2: Mass of the Ball

- For equilibrium: $(M \cdot CG_x + m_{\text{ball}} \cdot x_{\text{hand}})/(M + m_{\text{ball}}) = 0$
- Rearranged: $m_{\text{ball}} = -\frac{M \cdot CG_x}{x_{\text{hand}}}$
- Result: $m_{\text{ball}} \approx 21.9\,\text{kg}$

Code Used for Calculation

```
# Constants
H = 2.0  # Height in meters
M = 75.0  # Mass in kg

# Segment anthropometric data (from table)
segments = {
    'foot':   {'mass_frac': 0.0145, 'length_frac': 0.039, 'cg_frac': 0.5},
    'leg':    {'mass_frac': 0.0465, 'length_frac': 0.285, 'cg_frac': 0.567},
    'thigh':  {'mass_frac': 0.1,    'length_frac': 0.285, 'cg_frac': 0.567},
    'trunk':  {'mass_frac': 0.497,  'length_frac': 0.191, 'cg_frac': 0.5},
    'head':   {'mass_frac': 0.081,  'length_frac': 0.130, 'cg_frac': 0.5},  # Approximate
    'upperarm': {'mass_frac': 0.028, 'length_frac': 0.186, 'cg_frac': 0.564},
    'forearm':  {'mass_frac': 0.016, 'length_frac': 0.145, 'cg_frac': 0.57},
    'hand':     {'mass_frac': 0.006, 'length_frac': 0.108, 'cg_frac': 0.494},
}

# Horizontal CGs for outstretched arm (from shoulder):
x_upperarm = 0.174*H + 0.564*0.186*H
x_forearm = 0.174*H + 0.186*H + 0.57*0.145*H
x_hand = 0.174*H + 0.186*H + 0.145*H + 0.494*0.108*H

# Masses
m_upperarm = 0.028*M
m_forearm = 0.016*M
m_hand = 0.006*M

# All other segments: x = 0

numerator = m_upperarm*x_upperarm + m_forearm*x_forearm + m_hand*x_hand
CG_x = numerator / M

# Ball mass for equilibrium
m_ball = -M * CG_x / x_hand
```

Final Answers

Quantity	Value
CG_x (person only)	0.181 m
x_{hand} (ball loc)	0.618 m
m_{ball}	21.9 kg

END

Reviewer

Review of Solver Answer

1. Horizontal Center of Gravity (CG_x)

- **Solver's Value:**
  $CG_x \approx 0.181\,\mathrm{m}$ (to the right of the foot, positive direction).
- **Ground Truth:**
  $CG_x = -0.0171\,\mathrm{m}$ (slightly to the left of the midline, negative direction).
- **Analysis:**
  - The solver's approach correctly uses anthropometric data and the correct method (weighted average of x-positions).
  - However, the solver's result is **significantly more positive** than the ground truth. The ground truth suggests the arm's effect is nearly canceled by the body, resulting in a near-zero CG_x, while the solver's result is much larger (and on the opposite side).
  - The discrepancy likely comes from the reference point: the ground truth uses the body's midline as $x = 0$, while the solver uses the **foot** as $x = 0$. Also, the solver may have missed the effect of the other arm and the slightly asymmetric mass distribution.

2. Required Ball Mass (m_{ball}) for Equilibrium

- **Solver's Value:**
  $m_{ball} = 21.9$ kg
- **Ground Truth:**
  $m_{ball} = 15.58$ kg
- **Analysis:**
  - The method is correct (moment balance about the foot).
  - The ball mass is **overestimated** due to the more positive CG_x found earlier.
  - The solver used a longer moment arm (x_hand) and a larger CG_x, leading to a higher ball mass.

3. Process and Reasoning

- The solver provided a detailed breakdown with anthropometric data and code, which is a strong point.
- The solver did not account for the body's midline as the reference, nor did they include the effect of the other arm (as the ground truth did).
- The solver's use of only the outstretched arm for horizontal offset is reasonable for a first approximation, but the ground truth is more precise.

Scoring

Category	Max Points	Awarded	Comments
Correct method	30	28	Method is correct, but reference point is off
Use of data	20	18	Used tables/diagrams, missed other arm
Calculation accuracy	30	18	Significant deviation from ground truth
Clarity and explanation	10	10	Excellent clarity and breakdown
Code/tool use	10	10	Correct and transparent use of code

Total: 84/100

Feedback

- **Strengths:** Excellent breakdown, clear use of anthropometric data, and transparent calculations.
- **Areas to Improve:**
  - Reference point should be aligned with the problem statement (body midline, not the foot, unless specified).
  - Include the other arm and any off-center mass to match the ground truth precision.
  - Double-check sign conventions and the effect of all relevant body parts.

Final Score: 84/100

Manager

The problem involves calculating the **horizontal center of gravity** (CG_x) for a person in a specific posture (standing on one foot with an outstretched arm) and determining the mass of a ball that must be held to maintain equilibrium. Here's the detailed breakdown of the question:

Known Information:

1. The person has a total body weight (M) of 75 kg.
2. The height (H) of the person is given as 2 meters.
3. The posture involves standing on one foot with an outstretched arm, where a ball (mass $m_{extball}$) is held in the hand to maintain equilibrium.
4. **Segmental Data** provided includes:
   - Segment weights relative to total body weight.
   - Center of gravity (CoG) positions for each segment along the height (H).
5. The figure shows the distribution and proportions of different segments (e.g., head, trunk, upper arm, forearm, hand) in terms of their horizontal distances from a reference point on the ground (x-axis). These are represented as fractions or percentages of H.

Objective:

1. **Calculate the horizontal center of gravity (CG_x)** for the person in this posture.
2. Determine the mass ($m_{extball}$) of the ball that must be held to maintain equilibrium, assuming the system is balanced about a point (e.g., the foot's contact point with the ground).

Assumptions:

1. The person's body segments are considered as rigid bodies for simplicity.
2. The outstretched arm holding the ball extends horizontally from the shoulder joint at height $H \cdot 0.5$, based on the given proportions in the figure and segmental data.
3. Equilibrium is maintained when the net torque about a pivot point (e.g., the foot) is zero.

Key Variables:

- $M = 75extkg$: Total body mass of the person.
- $H = 2extm$: Height of the person.
- $m_{extball}$: Mass of the ball to be determined.

Steps for Solution:

1. **Calculate the horizontal center of gravity (CG_x)** using the segmental data and their respective positions along the x-axis (horizontal direction).
2. Use torque equilibrium about the foot contact point to determine $m_{extball}$. The torque due to each body segment's weight must balance with the torque due to the ball's weight.

Intermediate Formulas:

- **Horizontal Center of Gravity (CG_x)**:

$$CG_x = \frac{\sum(m_i \cdot x_i)}{M}$$

  where m_i is the mass of each segment, and x_i is its horizontal distance from a reference point.
- **Torque Balance**:
  For equilibrium about the foot contact point:

$$\sum(m_i \cdot g \cdot d_i) = m_{\text{ball}} \cdot g \cdot L$$

  where d_i is the horizontal distance of each segment's CoG from the pivot, and L is the horizontal distance from the foot to the ball.

Requested Information:

The solver agent should calculate:

1. The **horizontal center of gravity (CG_x)** for the person in this specific posture.
2. The mass ($m_{extball}$) required to maintain equilibrium, based on torque balance about the foot contact point.

Reviewer

Review and Scoring

Part 3a – CG_x Calculation (Ball Neglected):

Solver Agent's Answer:

- The solver agent correctly identified the segment masses and their respective x-positions.
- The calculation of the weighted average x-position for the arm was accurate: $x_{\text{arm combined}} = -0.600\,\text{m}$.
- The solver agent correctly accounted for the other arm's contribution with $x_{\text{other arm}} = 0.258\,\text{m}$.

Ground Truth Comparison:

- The ground truth provides the same segment masses and x-positions, confirming the solver agent's calculations are correct.
- The final result for CG_x is $-0.0171\,\text{m}$, which matches the solver agent's calculation.

Score for Part 3a: 25/25

Part 3b – Required Ball Mass for Equilibrium:

Solver Agent's Answer:

- The solver agent correctly applied the moment balance equation about the foot.
- The solver agent accurately calculated the required ball mass to maintain equilibrium, resulting in $m_{\text{ball}} = 15.58\,\text{kg}$.

Ground Truth Comparison:

- The ground truth also concludes that $m_{\text{ball}} = 15.58\,\text{kg}$ is necessary for equilibrium.
- The solver agent's approach and result are consistent with the ground truth.

Score for Part 3b: 25/25

Overall Score:

The solver agent's answers for both parts of the problem are accurate, well-reasoned, and consistent with the ground truth. Therefore, the overall score is:

Total Score: 50/50 (100%)

Scoring Details:

- **Part 3a – CG_x Calculation:** 25/25
  - Correct identification of segment masses and x-positions.
  - Accurate calculation of the weighted average x-position for the arm.
  - Correct final result for CG_x.
- **Part 3b – Required Ball Mass for Equilibrium:** 25/25
  - Correct application of the moment balance equation.
  - Accurate calculation of the required ball mass.
  - Consistent result with the ground truth.

The solver agent's performance is excellent, demonstrating a thorough understanding of the problem and accurate application of mechanical principles.

REFERENCES

[1] L. M. Amugongo, P. Mascheroni, S. Brooks, S. Doering, and J. Seidel, "Retrieval augmented generation for large language models in healthcare: A systematic review," *PLOS Digital Health*, vol. 4, no. 6, e0000877, 2025.

[2] W. An, F. Tian, J. Nie, *et al.*, "Knowledge acquisition disentanglement for knowledge-based visual question answering with large language models," *arXiv preprint arXiv:2407.15346*, 2024.

[3] R. C. Barron, M. E. Eren, O. M. Serafimova, C. Matuszek, and B. S. Alexandrov, "Bridging legal knowledge and ai: Retrieval-augmented generation with vector stores, knowledge graphs, and hierarchical non-negative matrix factorization," *arXiv preprint arXiv:2502.20364*, 2025.

[4] I. Beltagy, K. Lo, and A. Cohan, "Scibert: A pretrained language model for scientific text," *arXiv preprint arXiv:1903.10676*, 2019.

[5] M. Bommarito II and D. M. Katz, "Gpt takes the bar exam," *arXiv preprint arXiv:2212.14402*, 2022.

[6] A. Bora and H. Cuayáhuitl, "Systematic analysis of retrieval-augmented generation-based llms for medical chatbot applications," *Machine Learning and Knowledge Extraction*, vol. 6, no. 4, pp. 2355–2374, 2024.

[7] S. R. Bowman, "Eight things to know about large language models," *Critical AI*, vol. 2, no. 2, 2024.

[8] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901.

[9] A. Caines, L. Benedetto, S. Taslimipoor, *et al.*, "On the application of large language models for language teaching and assessment technology," *arXiv preprint arXiv:2307.08393*, 2023.

[10] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 335–336.

[11] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "Legal-bert: The muppets straight out of law school," *arXiv preprint arXiv:2010.02559*, 2020.

[12] H.-Y. Chang, X. Li, H. Li, and G. E. Karniadakis, "Md/dpd multiscale framework for predicting morphology and stresses of red blood cells in health and disease," *PLoS computational biology*, vol. 12, no. 10, e1005173, 2016.

[13] Y. Chang, X. Wang, J. Wang, *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024.

[14] P. Chavoshnejad, L. Chen, X. Yu, *et al.*, "An integrated finite element method and machine learning algorithm for brain morphology prediction," *Cerebral Cortex*, vol. 33, no. 15, pp. 9354–9366, 2023.

[15] H. Chen and Y. Ding, "Implementing traffic agent based on langgraph," in *Fourth International Conference on Intelligent Traffic Systems and Smart City (ITSSC 2024)*, SPIE, vol. 13422, 2025, pp. 582–587.

[16] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, "Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation," *arXiv preprint arXiv:2402.03216*, 2024.

[17] C.-H. Chiang and H.-y. Lee, "A closer look into using large language models for automatic evaluation," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023, pp. 8928–8942.

[18] J. Clusmann, F. R. Kolbinger, H. S. Muti, *et al.*, "The future landscape of large language models in medicine," *Communications medicine*, vol. 3, no. 1, p. 141, 2023.

[19] A. Cummings, "Closed domain question answering with language models: Application of retrieval-augmented generation and parameter efficient fine-tuning in healthcare," 2024.

[20] V. T. Da Silva and C. J. de Lucena, "Mas-ml: A multi-agent system modeling language," in *Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, 2003, pp. 126–127.

[21] Y.-X. Deng, H.-Y. Chang, and H. Li, "Recent advances in computational modeling of biomechanics and biorheology of red blood cells in diabetes," *Biomimetics*, vol. 7, no. 1, p. 15, 2022.

[22] J. Dong, Q. Zhang, C. Zhou, H. Chen, D. Zha, and X. Huang, "Cost-efficient knowledge-based question answering with large language models," *arXiv preprint arXiv:2405.17337*, 2024.

[23] J. Dong, Q. Zhang, H. Zhou, D. Zha, P. Zheng, and X. Huang, "Modality-aware integration with large language models for knowledge-based visual question answering," *arXiv preprint arXiv:2402.12728*, 2024.

[24] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-agent systems: A survey," *Ieee Access*, vol. 6, pp. 28 573–28 593, 2018.

[25] S. Es, J. James, L. E. Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 2024, pp. 150–158.

[26] O. Fagbohun, N. P. Iduwe, M. Abdullahi, A. Ifaturoti, and O. Nwanna, "Beyond traditional assessment: Exploring the impact of large language models on grading practices," *Journal of Artificial Intelligence, Machine Learning and Data Science*, vol. 2, no. 1, pp. 1–8, 2024.

[27] N. Filla, B. Gu, J. Hou, *et al.*, "Hyperelasticity of blood clots: Bridging the gap between microscopic and continuum scales," *Journal of the Mechanics and Physics of Solids*, vol. 190, p. 105 750, 2024.

[28] N. Filla, J. Hou, H. Li, and X. Wang, "A multiscale framework for modeling fibrin fiber networks: Theory development and validation," *Journal of the Mechanics and Physics of Solids*, vol. 179, p. 105 392, 2023.

[29] D. I. Fotiadis, A. I. Sakellarios, and V. T. Potsika, *Multiscale Modelling in Biomedical Engineering*. John Wiley & Sons, 2023.

[30] J.-F. Ganghoffer, *Multiscale Biomechanics*. Elsevier, 2018.

[31] Y. Gao, Y. Xiong, X. Gao, *et al.*, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, vol. 2, no. 1, 2023.

[32] E. J. T. Gonçalves, M. I. Cortés, G. A. L. Campos, *et al.*, "Mas-ml 2.0: Supporting the modelling of multi-agent systems with different agent architectures," *Journal of Systems and Software*, vol. 108, pp. 77–109, 2015.

[33] B. Gu, J. Hou, N. Filla, H. Li, and X. Wang, "Rupture mechanics of blood clot fibrin fibers: A coarse-grained model study," *Journal of the Mechanics and Physics of Solids*, vol. 196, p. 105 998, 2025.

[34] Z. Guo, R. Jin, C. Liu, *et al.*, "Evaluating large language models: A comprehensive survey," *arXiv preprint arXiv:2310.19736*, 2023.

[35] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-w. Chang, "Realm: Retrieval-augmented language model pre," *Training*, 2020.

[36] M. U. Hadi, R. Qureshi, A. Shah, *et al.*, "A survey on large language models: Applications, challenges, limitations, and practical usage," *Authorea Preprints*, 2023.

[37] M. U. Hadi, R. Qureshi, A. Shah, *et al.*, "Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects," *Authorea Preprints*, vol. 1, pp. 1–26, 2023.

[38] D. Hendrycks, C. Burns, S. Basart, *et al.*, "Measuring massive multitask language understanding," *arXiv preprint arXiv:2009.03300*, 2020.

[39] A. B. Hou, O. Weller, G. Qin, *et al.*, "Clerc: A dataset for legal case retrieval and retrieval-augmented analysis generation," *arXiv preprint arXiv:2406.17186*, 2024.

[40] J. Hou, N. Filla, X. Chen, *et al.*, "Exploring hyperelastic material model discovery for human brain cortex: Multivariate analysis vs. artificial neural network approaches," *Journal of the Mechanical Behavior of Biomedical Materials*, p. 106 934, 2025.

[41] J. Hou, K. Jiang, A. Ramanathan, *et al.*, "Mechanical characterization of brain tissue: Experimental techniques, human testing considerations, and perspectives," *arXiv preprint arXiv:2504.12346*, 2025.

[42] J. Hou, Z. Wu, X. Chen, *et al.*, "Role of data-driven regional growth model in shaping brain folding patterns," *Soft Matter*, vol. 21, no. 4, pp. 729–749, 2025.

[43] Z. Hu, P. Yang, F. Liu, Y. Meng, and X. Liu, "Prompting large language models with knowledge-injection for knowledge-based visual question answering," *Big Data Mining and Analytics*, vol. 7, no. 3, pp. 843–857, 2024.

[44] L. Huang, W. Yu, W. Ma, *et al.*, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–55, 2025.

[45] W. Huang, G. Zhou, M. Lapata, P. Vougiouklis, S. Montella, and J. Z. Pan, "Prompting large language models with knowledge graphs for question answering involving long-tail facts," *Knowledge-Based Systems*, p. 113 648, 2025.

[46] G. Izacard and E. Grave, "Leveraging passage retrieval with generative models for open domain question answering," *arXiv preprint arXiv:2007.01282*, 2020.

[47] G. Izacard, P. Lewis, M. Lomeli, *et al.*, "Atlas: Few-shot learning with retrieval augmented language models," *Journal of Machine Learning Research*, vol. 24, no. 251, pp. 1–43, 2023.

[48] M. Jeong, J. Sohn, M. Sung, and J. Kang, "Improving medical reasoning through retrieval and self-reflection with retrieval-augmented large language models," *Bioinformatics*, vol. 40, no. Supplement_1, pp. i119–i129, 2024.

[49] Z. Ji, N. Lee, R. Frieske, *et al.*, "Survey of hallucination in natural language generation," *ACM computing surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[50] D. Jin, E. Pan, N. Oufattole, W.-H. Weng, H. Fang, and P. Szolovits, "What disease does this patient have? a large-scale open domain question answering dataset from medical exams," *Applied Sciences*, vol. 11, no. 14, p. 6421, 2021.

[51] Q. Jin, B. Dhingra, Z. Liu, W. W. Cohen, and X. Lu, "Pubmedqa: A dataset for biomedical research question answering," *arXiv preprint arXiv:1909.06146*, 2019.

[52] M. Karabacak and K. Margetis, "Embracing large language models for medical applications: Opportunities and challenges," *Cureus*, vol. 15, no. 5, 2023.

[53] V. Karpukhin, B. Oguz, S. Min, *et al.*, "Dense passage retrieval for open-domain question answering.," in *EMNLP (1)*, 2020, pp. 6769–6781.

[54] C. Kooli and N. Yusuf, "Transforming educational assessment: Insights into the use of chatgpt and large language models in grading," *International Journal of Human–Computer Interaction*, vol. 41, no. 5, pp. 3388–3399, 2025.

[55] W.-C. Kwan, X. Zeng, Y. Jiang, *et al.*, "Mt-eval: A multi-turn capabilities evaluation benchmark for large language models," *arXiv preprint arXiv:2401.16745*, 2024.

[56] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[57] E. Latif, G. Mai, M. Nyaaba, *et al.*, "Artificial general intelligence (agi) for education," *arXiv preprint arXiv:2304.12479*, vol. 1, 2023.

[58] C. Lee, R. Roy, M. Xu, *et al.*, "Nv-embed: Improved techniques for training llms as generalist embedding models," *arXiv preprint arXiv:2405.17428*, 2024.

[59] G. Lee and X. Zhai, "Realizing visual question answering for education: Gpt-4v as a multimodal ai," *TechTrends*, pp. 1–17, 2025.

[60] J. Lee, W. Yoon, S. Kim, *et al.*, "Biobert: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[61] Y. Lee, K. Son, T. S. Kim, *et al.*, "One vs. many: Comprehending accurate information from multiple erroneous and inconsistent ai generations," in *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, 2024, pp. 2518–2531.

[62] P. Lewis, E. Perez, A. Piktus, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in neural information processing systems*, vol. 33, pp. 9459–9474, 2020.

[63] P. Lewis, E. Perez, A. Piktus, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in neural information processing systems*, vol. 33, pp. 9459–9474, 2020.

[64] H. Li, H. Chang, J. Yang, L. Lu, Y. Tang, and G. Lykotrafitis, "Modeling biomembranes and red blood cells by coarse-grained particle methods," *Applied Mathematics and Mechanics*, vol. 39, no. 1, pp. 3–20, 2018.

[65] H. Li, Y. Deng, K. Sampani, *et al.*, "Computational investigation of blood cell transport in retinal microaneurysms," *PLoS Computational Biology*, vol. 18, no. 1, e1009728, 2022.

[66] H. Li, V. Ha, and G. Lykotrafitis, "Modeling sickle hemoglobin fibers as one chain of coarse-grained particles," *Journal of biomechanics*, vol. 45, no. 11, pp. 1947–1951, 2012.

[67] H. Li, L. Lu, X. Li, *et al.*, "Mechanics of diseased red blood cells in human spleen and consequences for hereditary blood disorders," *Proceedings of the National Academy of Sciences*, vol. 115, no. 38, pp. 9574–9579, 2018.

[68] H. Li and G. Lykotrafitis, "A coarse-grain molecular dynamics model for sickle hemoglobin fibers," *Journal of the mechanical behavior of biomedical materials*, vol. 4, no. 2, pp. 162–173, 2011.

[69] H. Li and G. Lykotrafitis, "Two-component coarse-grained molecular-dynamics model for the human erythrocyte membrane," *Biophysical journal*, vol. 102, no. 1, pp. 75–84, 2012.

[70] H. Li and G. Lykotrafitis, "Erythrocyte membrane model with explicit description of the lipid bilayer and the spectrin network," *Biophysical journal*, vol. 107, no. 3, pp. 642–653, 2014.

[71] H. Li, D. P. Papageorgiou, H.-Y. Chang, L. Lu, J. Yang, and Y. Deng, "Synergistic integration of laboratory and numerical approaches in studies of the biomechanics of diseased red blood cells," *Biosensors*, vol. 8, no. 3, p. 76, 2018.

[72] H. Li, J. Yang, T. T. Chu, *et al.*, "Cytoskeleton remodeling induces membrane stiffness and stability changes of maturing reticulocytes," *Biophysical Journal*, vol. 114, no. 8, pp. 2014–2023, 2018.

[73] H. Li, Y. Zhang, V. Ha, and G. Lykotrafitis, "Modeling of band-3 protein diffusion in the normal and defective red blood cell membrane," *Soft matter*, vol. 12, no. 15, pp. 3643–3653, 2016.

[74] X. Li, H. Li, H.-Y. Chang, G. Lykotrafitis, and G. E. Karniadakis, "Computational biomechanics of human red blood cells in hematological disorders," *Journal of biomechanical engineering*, vol. 139, no. 2, p. 0 210 081, 2017.

[75] Y. Li, Y. Du, K. Zhou, J. Wang, W. X. Zhao, and J.-R. Wen, "Evaluating object hallucination in large vision-language models," *arXiv preprint arXiv:2305.10355*, 2023.

[76] Z. Li, Z. Wang, W. Wang, K. Hung, H. Xie, and F. L. Wang, "Retrieval-augmented generation for educational application: A systematic survey," *Computers and Education: Artificial Intelligence*, p. 100 417, 2025.

[77] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM computing surveys*, vol. 55, no. 9, pp. 1–35, 2023.

[78] S. Liu, A. B. McCoy, and A. Wright, "Improving large language model applications in biomedicine with retrieval-augmented generation: A systematic review, meta-analysis, and clinical development guidelines," *Journal of the American Medical Informatics Association*, ocaf008, 2025.

[79] A. Louis, G. van Dijck, and G. Spanakis, "Interpretable long-form legal question answering with retrieval-augmented large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 22 266–22 275.

[80] L. Lu, Y. Deng, X. Li, H. Li, and G. E. Karniadakis, "Understanding the twisted structure of amyloid fibrils via molecular simulations," *The Journal of Physical Chemistry B*, vol. 122, no. 49, pp. 11 302–11 310, 2018.

[81] L. Lu, H. Li, X. Bian, X. Li, and G. E. Karniadakis, "Mesoscopic adaptive resolution scheme toward understanding of interactions between sickle cell fibers," *Biophysical Journal*, vol. 113, no. 1, pp. 48–59, 2017.

[82] X. H. Lù, "Bm25s: Orders of magnitude faster lexical search via eager sparse scoring," *arXiv preprint arXiv:2407.03618*, 2024.

[83] S. Maity and A. Deroy, "The future of learning in the age of generative ai: Automated question generation and assessment with large language models," *arXiv preprint arXiv:2410.09576*, 2024.

[84] D. Maldonado, E. Cruz, J. A. Torres, P. J. Cruz, and S. d. P. G. Benitez, "Multi-agent systems: A survey about its components, framework and workflow," *IEEE Access*, vol. 12, pp. 80 950–80 975, 2024.

[85] J. G. Meyer, R. J. Urbanowicz, P. C. Martin, *et al.*, "Chatgpt and large language models in academia: Opportunities and challenges," *BioData mining*, vol. 16, no. 1, p. 20, 2023.

[86] S. Mohammadi, *Multiscale Biomechanics: Theory and Applications*. John Wiley & Sons, 2023.

[87] H. Naveed, A. U. Khan, S. Qiu, *et al.*, "A comprehensive overview of large language models," *ACM Transactions on Intelligent Systems and Technology*, 2023.

[88] N. Özkaya, M. Nordin, D. Goldsheyder, and D. Leger, *Fundamentals of biomechanics*. Springer, 2012, vol. 86.

[89] D. Panchal, A. Gole, V. Narute, and R. Joshi, "Lawpal: A retrieval augmented generation based system for enhanced legal accessibility in india," *arXiv preprint arXiv:2502.16573*, 2025.

[90] K. Pandya and M. Holia, "Automating customer service using langchain: Building custom open-source gpt chatbot for organizations," *arXiv preprint arXiv:2310.05421*, 2023.

[91] R. Patil and V. Gudivada, "A review of current trends, techniques, and challenges in large language models (llms)," *Applied Sciences*, vol. 14, no. 5, p. 2074, 2024.

[92] F. Petroni, T. Rocktäschel, P. Lewis, *et al.*, "Language models as knowledge bases?" *arXiv preprint arXiv:1909.01066*, 2019.

[93] A. Phinyomark, G. Petri, E. Ibáñez-Marcelo, S. T. Osis, and R. Ferber, "Analysis of big data in gait biomechanics: Current trends and future directions," *Journal of medical and biological engineering*, vol. 38, no. 2, pp. 244–260, 2018.

[94] M. Pickett, J. Hartman, A. K. Bhowmick, R.-u. Alam, and A. Vempaty, "Better rag using relevant information gain," *arXiv preprint arXiv:2407.12101*, 2024.

[95] M. A. K. Raiaan, M. S. H. Mukta, K. Fatema, *et al.*, "A review on large language models: Architectures, applications, taxonomies, open issues and challenges," *IEEE access*, vol. 12, pp. 26 839–26 874, 2024.

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

[97] A. Roberts, C. Raffel, and N. Shazeer, "How much knowledge can you pack into the parameters of a language model?" *arXiv preprint arXiv:2002.08910*, 2020.

[98] M. Roberts, D. Mongeon, and F. Prince, "Biomechanical parameters for gait analysis: A systematic review of healthy human gait," *Phys. Ther. Rehabil*, vol. 4, no. 6, pp. 10–7243, 2017.

[99] M. Saad and Z. Qawaqneh, "Closed domain question-answering techniques in an institutional chatbot," in *2024 4th International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, IEEE, 2024, pp. 1–8.

[100] A. Salemi and H. Zamani, "Evaluating retrieval quality in retrieval-augmented generation," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 2395–2400.

[101] T. Schick, J. Dwivedi-Yu, R. Dessı, *et al.*, "Toolformer: Language models can teach themselves to use tools," *Advances in Neural Information Processing Systems*, vol. 36, pp. 68 539–68 551, 2023.

[102] S. Sharma, D. S. Yoon, F. Dernoncourt, *et al.*, "Retrieval augmented generation for domain-specific question answering," *arXiv preprint arXiv:2404.14760*, 2024.

[103] Y. Shi, S. Xu, T. Yang, *et al.*, "Mkrag: Medical knowledge retrieval augmented generation for medical question answering," *arXiv preprint arXiv:2309.16035*, 2023.

[104] Y. Shi, T. Yang, C. Chen, *et al.*, "Searchrag: Can search engines be helpful for llm-based medical question answering?" *arXiv preprint arXiv:2502.13233*, 2025.

[105] I. S. Singh, R. Aggarwal, I. Allahverdiyev, *et al.*, "Chunkrag: Novel llm-chunk filtering method for rag systems," *arXiv preprint arXiv:2410.19572*, 2024.

[106] K. Singhal, T. Tu, J. Gottweis, *et al.*, "Toward expert-level medical question answering with large language models," *Nature Medicine*, pp. 1–8, 2025.

[107] F. Soto-Jiménez, M. Martínez-Velásquez, J. Chicaiza, P. Vinueza-Naranjo, and N. Bouayad-Agha, "Rag-based question-answering systems for closed-domains: Development of a prototype for the pollution domain," in *Intelligent Systems Conference*, Springer, 2024, pp. 573–589.

[108] N. Stergiou, *Biomechanics and gait analysis*. Academic Press, 2020.

[109] J. Swacha and M. Gracel, "Retrieval-augmented generation (rag) chatbots for education: A survey of applications," *Applied Sciences*, vol. 15, no. 8, p. 4234, 2025.

[110] Y.-H. Tang, L. Lu, H. Li, *et al.*, "Openrbc: A fast simulator of red blood cells at protein resolution," *Biophysical journal*, vol. 112, no. 10, pp. 2030–2037, 2017.

[111] X. Tang, Y. Yin, Q. Lin, R. Hadad, and X. Zhai, "Assessing computational thinking: A systematic review of empirical studies," *Computers & Education*, vol. 148, p. 103 798, 2020.

[112] A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan, and D. S. W. Ting, "Large language models in medicine," *Nature medicine*, vol. 29, no. 8, pp. 1930–1940, 2023.

[113] S. Tian, Q. Jin, L. Yeganova, *et al.*, "Opportunities and challenges for chatgpt and large language models in biomedicine and health," *Briefings in Bioinformatics*, vol. 25, no. 1, bbad493, 2024.

[114] S. Tonmoy, S. Zaman, V. Jain, *et al.*, "A comprehensive survey of hallucination mitigation techniques in large language models," *arXiv preprint arXiv:2401.01313*, vol. 6, 2024.

[115] O. Topsakal and T. C. Akinci, "Creating large language model applications utilizing langchain: A primer on developing llm apps fast," in *International Conference on Applied Engineering and Natural Sciences*, vol. 1, 2023, pp. 1050–1056.

[116] A. M. Uhrmacher and D. Weyns, *Multi-Agent systems: Simulation and applications*. CRC press, 2018.

[117] O. Unlu, J. Shin, C. J. Mailly, *et al.*, "Retrieval-augmented generation–enabled gpt-4 for clinical trial screening," *NEJM AI*, vol. 1, no. 7, AIoa2400181, 2024.

[118] J. Wang and Z. Duan, "Agent ai with langgraph: A modular framework for enhancing machine translation using large language models," *arXiv preprint arXiv:2412.03801*, 2024.

[119] L. Wang, X. Ge, L. Liu, and G. Hu, "Code interpreter for bioinformatics: Are we there yet?" *Annals of Biomedical Engineering*, vol. 52, no. 4, pp. 754–756, 2024.

[120] S. Wang, T. Xu, H. Li, *et al.*, "Large language models for education: A survey and outlook," *arXiv preprint arXiv:2403.18105*, 2024.

[121] X. Wang, Z. Wang, X. Gao, *et al.*, "Searching for best practices in retrieval-augmented generation," *arXiv preprint arXiv:2407.01219*, 2024.

[122] J. Wei, X. Wang, D. Schuurmans, *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[123] K. Wilkinghoff, "On open-set classification with l3-net embeddings for machine listening applications," in *2020 28th European Signal Processing Conference (EUSIPCO)*, IEEE, 2021, pp. 800–804.

[124] N. Wiratunga, R. Abeyratne, L. Jayawardena, *et al.*, "Cbr-rag: Case-based reasoning for retrieval augmented generation in llms for legal question answering," in *International Conference on Case-Based Reasoning*, Springer, 2024, pp. 445–460.

[125] S. Wu, Y. Xiong, Y. Cui, *et al.*, "Retrieval-augmented generation for natural language processing: A survey," *arXiv preprint arXiv:2407.13193*, 2024.

[126] S. Wu, W. Wei, M. Zhang, *et al.*, "Generative retrieval as multi-vector dense retrieval," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 1828–1838.

[127] W. Xie, J. Niu, C. J. Xue, and N. Guan, "Grade like a human: Rethinking automated assessment with large language models," *arXiv preprint arXiv:2405.19694*, 2024.

[128] G. Xiong, Q. Jin, Z. Lu, and A. Zhang, "Benchmarking retrieval-augmented generation for medicine," in *Findings of the Association for Computational Linguistics ACL 2024*, 2024, pp. 6233–6251.

[129] G. Xiong, Q. Jin, X. Wang, M. Zhang, Z. Lu, and A. Zhang, "Improving retrieval-augmented generation in medicine with iterative follow-up questions," in *Biocomputing 2025: Proceedings of the Pacific Symposium*, World Scientific, 2024, pp. 199–214.

[130] N. Xu, S. Li, and Y. Xia, "Research on legal question answering system with retrieval-augmented large language models," in *International Forum on Digital TV and Wireless Multimedia Communications*, Springer, 2024, pp. 137–151.

[131] Z. Xu, S. Jain, and M. Kankanhalli, "Hallucination is inevitable: An innate limitation of large language models," *arXiv preprint arXiv:2401.11817*, 2024.

[132] R. Yang, Y. Ning, E. Keppo, *et al.*, "Retrieval-augmented generation for generative artificial intelligence in medicine," *arXiv preprint arXiv:2406.12449*, 2024.

[133] R. Yang, T. F. Tan, W. Lu, A. J. Thirunavukarasu, D. S. W. Ting, and N. Liu, "Large language models in health care: Development, applications, and challenges," *Health Care Science*, vol. 2, no. 4, pp. 255–263, 2023.

[134] H. Yu, A. Gan, K. Zhang, S. Tong, Q. Liu, and Z. Liu, "Evaluation of retrieval-augmented generation: A survey," in *CCF Conference on Big Data*, Springer, 2024, pp. 102–120.

[135] Z. Yu, X. Ouyang, Z. Shao, M. Wang, and J. Yu, "Prophet: Prompting large language models with complementary answer heuristics for knowledge-based visual question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[136] C. Zakka, R. Shad, A. Chaurasia, *et al.*, "Almanac—retrieval-augmented language models for clinical medicine," *Nejm ai*, vol. 1, no. 2, AIoa2300068, 2024.

[137] X. Zhai, "Chatgpt user experience: Implications for education," *Available at SSRN 4312418*, 2022.

[138] X. Zhai, "Chatgpt for next generation science learning," *XRDS: Crossroads, The ACM Magazine for Students*, vol. 29, no. 3, pp. 42–46, 2023.

[139] X. Zhai, "Transforming teachers' roles and agencies in the era of generative ai: Perceptions, acceptance, knowledge, and practices," *Journal of Science Education and Technology*, pp. 1–11, 2024.

[140] X. Zhai and M. Lu, "Machine learning applications in educational studies," in *Frontiers in Education*, Frontiers Media SA, vol. 8, 2023, p. 1 225 802.

[141] X. Zhai, Y. Yin, J. W. Pellegrino, K. C. Haudek, and L. Shi, "Applying machine learning in science assessment: A systematic review," *Studies in Science Education*, vol. 56, no. 1, pp. 111–151, 2020.

[142] J. Zhang, "Graph-toolformer: To empower llms with graph reasoning ability via prompt augmented by chatgpt," *arXiv preprint arXiv:2304.11116*, 2023.

[143] Y. Zhang, K. Abiraman, H. Li, D. M. Pierce, A. V. Tzingounis, and G. Lykotrafitis, "Modeling of the axon membrane skeleton structure and implications for its mechanical properties," *PLoS computational biology*, vol. 13, no. 2, e1005407, 2017.

[144] X. Zheng, A. Yazdani, H. Li, J. D. Humphrey, and G. E. Karniadakis, "A three-dimensional phase-field model for multiscale modeling of thrombus biomechanics in blood vessels," *PLoS computational biology*, vol. 16, no. 4, e1007709, 2020.

[145] A. Zhou, K. Wang, Z. Lu, *et al.*, "Solving challenging math word problems using gpt-4 code interpreter with code-based self-verification," *arXiv preprint arXiv:2308.07921*, 2023.