ISSN 2723-6471

A Hybrid Fuzzy-LLM Framework for Difficulty Estimation of Math Word Problems: A Data-Driven Human-in-the-Loop Study

Shilpa Kadam[1,*], Jabez Christopher[2], PTV Praveen Kumar[3], Dipak Kumar Satpathi[4]

[1,3,4]*Department of Mathematics, BITS Pilani, Hyderabad Campus, Telangana, 500078, India.*

[2]*Department of Computer Science and Information Systems, BITS Pilani, Hyderabad Campus, Telangana, 500078, India.*

(Received: November 20, 2025; Revised: January 15, 2026; Accepted: March 18, 2026; Available online: April 18, 2026)

Abstract

Assessing the difficulty levels of Math Word Problems (MWPs) is essential for adaptive learning, yet most existing MWP datasets lack standardized difficulty annotations. This paper proposes a decision framework that integrates a 2-tuple Fuzzy Linguistic Decision Model (FLDM) with Large Language Models (LLMs) for automated difficulty estimation. A corpus of over 2,000 MWPs was compiled, of which 200 were annotated by seven instructors and an additional 454 were validated by ten experts. Consensus stability improved markedly (Fleiss' $\kappa = 0.14 \rightarrow$ Cohen's $\kappa = 0.32$), reflecting stronger alignment between expert judgments and the proposed fuzzy 2-tuple aggregation. Sixteen LLM configurations were evaluated, including GPT-3.5, GPT-4o-Mini, Gemini Flash, and LLaMA-3.2 under Zero-Shot, Five-Shot, and RAG settings. GPT-3.5 Zero-Shot achieved the best performance (Precision=0.65, Recall=0.63, F1=0.63), outperforming GPT-4o-Mini and Gemini variants. The validated dataset and linguistic ground truth were integrated into a web-based annotation system (themathbits.com), demonstrating scalability for real-world deployment. The results show that combining human linguistic judgments with fuzzy modeling and LLM inference improves reliability of MWP difficulty estimation, providing a foundation for future adaptive learning platforms.

Keywords: Math Word Problems, Difficulty Estimation, Fuzzy Linguistic Decision Model, Large Language Models, Educational Data, Expert Annotation, Adaptive Learning

1. Introduction

The increasing integration of artificial intelligence (AI) and data-driven technologies into education has transformed how learning content is created, delivered, and assessed. Adaptive learning systems designed to personalize instruction based on learner needs require reliable metadata about educational content to function effectively. One such critical metadata element is the difficulty level of Math Word Problems (MWPs). Difficulty estimation enables personalized learning pathways, supports assessment design, and provides an informed structure for content sequencing. Despite its importance, most existing MWP repositories lack standardized difficulty annotations or grade-level metadata [1]. Datasets such as MathInstruct, MathQA, AQUA-RAT, MetaMathQA, SVAMP, and GSM8K provide rich collections of problems but typically omit explicit difficulty labels. Even large-scale corpora with diverse problem types often present inconsistencies in answers, lexicon patterns, and the cognitive challenge represented by the problems. Only a few datasets such as ASDiv offer problem difficulty, yet even these annotations reveal inconsistencies arising from subjective interpretations across instructors. The absence of reliable, standardized difficulty labels limits the development of scalable educational tools and constrains the evaluation of automated problem-solving and reasoning systems.

Human annotation remains the most accurate way to assign difficulty labels, but it is labour-intensive and inherently subjective. Instructor judgments vary based on experience, pedagogical background, and interpretation of mathematical complexity. Without consensus-building methods, labels derived from individual instructors are difficult to aggregate into a trustworthy ground truth. Standardized testing practices (e.g., SAT, GRE) address this issue through expert panels, pilot testing, and statistical aggregation, highlighting the need for structured methods to produce consensus

***Corresponding author: Shilpa Kadam (p20190508@hyderabad.bits-pilani.ac.in)**
DOI: https://doi.org/10.47738/jads.v7i2.1187
This is an open access article under the CC-BY license (https://creativecommons.org/licenses/by/4.0/).
© Authors retain all copyrights

difficulty levels in educational datasets. Prior work on difficulty estimation in MWPs spans unsupervised, supervised, and semi-supervised approaches. Unsupervised clustering has been used to group MWPs by linguistic and structural features, revealing distinct patterns that may correlate with difficulty [2], [3], [4]. Supervised approaches have leveraged machine learning classifiers such as Random Forest, Naïve Bayes, and gradient boosting, trained on handcrafted linguistic features or word embeddings (Word2Vec, GloVe, FastText, BERT) [5]. These approaches achieved macro F1-scores of approximately 0.25, indicating limited discriminative power. Subsequent embedding-based methods improved performance to around 0.40 macro F1. These methods have shown moderate success, particularly when combined with semantic embeddings. Semi-supervised and human-in-the-loop (HITL) [6] methods further incorporate expert feedback to refine predictions, but they still depend heavily on the quality and consistency of labeled data.

The recent emergence of large language models (LLMs) has introduced new opportunities for automating difficulty estimation. LLMs exhibit strong capabilities in problem solving, mathematical reasoning, and step-by-step explanation generation. Models such as GPT-3.5, GPT-4, MathGPT, and domain-specific variants have demonstrated promising results in problem solving and question generation. However, LLMs still struggle with context alignment, hallucinations, and inconsistent predictions across prompts and configurations. Retrieval-Augmented Generation (RAG) further enhances LLMs by incorporating external knowledge bases to provide problem-relevant context, potentially improving difficulty estimation. Yet the reliability of LLM predictions remains constrained by the availability of high-quality annotated datasets. Challenges include the lack of standardized difficulty annotations in existing MWP datasets, high subjectivity and inconsistency in expert difficulty ratings, limited interpretability and reliability of LLM predictions for educational tasks, and the absence of scalable human-in-the-loop systems for collecting validated difficulty labels.

This study makes several contributions toward understanding and automating difficulty estimation for mathematical word problems (MWPs). First, we introduce a 2-tuple fuzzy linguistic decision framework for aggregating expert judgments. Unlike traditional averaging methods that often lose linguistic nuance, the proposed model preserves both the semantic meaning of difficulty labels and the subtle numerical variation present in human ratings, enabling a more faithful reconciliation of heterogeneous expert opinions.

Second, we perform a systematic evaluation of sixteen large language model (LLM) configurations, spanning Zero-Shot, Five-Shot, and Retrieval-Augmented Generation (RAG) prompting strategies across GPT-3.5, GPT-4o-Mini, Gemini Flash, and LLaMA-3.2. Among these, GPT-3.5 with zero-shot prompting achieves the strongest performance (F1 = 0.63), providing a practical baseline for automated difficulty prediction.

Third, we develop themathbits.com, a web-based annotation platform that supports scalable human-AI collaboration. The platform enables expert labeling, rater-consistency monitoring, and fuzzy aggregation to generate reliable consensus difficulty labels. Using this workflow, we curate a repository of over 2,000 MWPs enriched with expert-validated difficulty and grade metadata, providing a reusable resource for adaptive learning and educational analytics research.

Finally, our empirical analysis shows that fuzzy aggregation significantly stabilizes expert consensus, improving agreement between aggregated labels and individual raters (Cohen's $\kappa = 0.24–0.33$) compared with the raw inter-expert agreement observed prior to aggregation (Fleiss' $\kappa = 0.14$). Together, these results demonstrate the value of combining fuzzy decision modeling with LLM-based analysis for handling inherently subjective educational assessments. We release a curated dataset of more than 2,000 MWPs, each enriched with expert-validated difficulty and grade metadata.

2. Related Work

Math Word Problem (MWP) repositories play an important role across education, artificial intelligence, and computational linguistics. They serve as core resources for machine learning research, benchmark evaluations, automated tutoring systems, and curriculum design. Existing datasets vary widely in scope, problem structure, linguistic patterns, and intended application, yet most lack standardized metadata, particularly difficulty labels and grade-level classifications, limiting their usefulness for adaptive learning and data-driven assessment.

A large number of recent corpora focus primarily on training and evaluating machine reasoning models. Examples include AutoMathText [7], MamooTH [8], MetaMath [9], Lila [10], GSM8K [11], SVAMP [12], MAWPS [13], and ALG514 [14], which provide diverse problem sets ranging from elementary arithmetic to advanced algebra and pre-calculus. However, these datasets typically omit explicit difficulty annotations and often show inconsistencies in answer formats or lexicon variety. As a result, they provide insufficient ground truth for difficulty modeling or educational analytics. In contrast, the ASDiv [15] dataset offers curated problems with broad lexical coverage and includes both problem types and difficulty tags. ASDiv encodes difficulty on a six-level scale, distinguishing basic addition and subtraction (levels 1–2) from more complex problems requiring domain knowledge or multi-step reasoning (levels 5–6), rather than being independently derived from rater consensus or psychometric calibration. Similarly, the MATH dataset [16] includes challenging problems sourced from standardized competitions (AMC, AIME, etc.) with difficulty levels guided by the Art of Problem Solving (AoPS) taxonomy. Yet across datasets, the difficulty scales, grade categorizations, and annotation conventions differ significantly, complicating cross-dataset comparisons.

Beyond dataset compilation, several studies have investigated linguistic or structural patterns in MWPs [3]. Early work analyzed lexical cues, semantic templates, and syntactic structures associated with mathematical reasoning, though these features were not explicitly used for difficulty estimation. More recent machine learning approaches have explored unsupervised clustering of MWPs based on linguistic features, supervised classification using traditional models (Random Forest, LightGBM, Naïve Bayes), and embedding-based methods leveraging Word2Vec, GloVe, FastText, or BERT [5]. These models classify MWPs into predefined difficulty categories but depend heavily on annotated data and often struggle with generalization across diverse problem types. Semi-supervised methods and HITL [6] workflows have been explored to refine predictions by incorporating expert feedback, but these approaches still require substantial manual effort and lack scalable consensus mechanisms. Variability in expert judgments, influenced by background, pedagogy, and learner profiles, presents a challenge for consistent labeling. In related domains, fuzzy linguistic decision-making has been widely applied for group consensus and multi-criteria decision-making [13], [17], [18], [19], offering a potential solution for aggregating heterogeneous ratings without loss of information [20].

Parallel to these advances, large language models (LLMs) have demonstrated strong performance in mathematical reasoning tasks, question answering, and stepwise problem solving. Studies comparing ChatGPT (GPT-3.5) with domain-targeted models such as LLMMathChain show notable improvements in reasoning fluency, yet both struggle with consistent accuracy on word problems. Research on teacher-LLM interaction has revealed misconceptions and superficial engagement among pre-service mathematics educators, illustrating the need for careful integration of LLMs into educational contexts [21], [22]. Retrieval-Augmented Generation (RAG) has been used to improve the accuracy of LLM-generated explanations and responses by grounding them in vetted textbooks or structured mathematical resources [23]. Other work, such as LSTM-based equation parsing [24], automated assessment generation (ItemForge) [25], and hierarchical label-attention models (e.g., DA-20K) [4], has contributed to knowledge extraction and educational content generation. However, none of these efforts examine the automated estimation of difficulty levels, a key component for adaptive learning platforms.

Difficulty estimation itself presents several challenges. Instructor judgments vary widely due to differences in teaching experience, pedagogical philosophy, demographics, and interpretations of mathematical complexity. Difficulty is inherently subjective and multidimensional, encompassing not only mathematical operations but also linguistic structure, contextual knowledge, and cognitive load. These sources of variability underscore the need for consensus-building methods that can aggregate human judgments robustly. Fuzzy linguistic decision models (FLDMs) [26], widely used in group decision-making, multi-criteria evaluation, and subjective assessment domains, provide a structured way to synthesize expert opinions while preserving uncertainty. Prior work on linguistic aggregation, linguistic hierarchies, and 2-tuple fuzzy representations offers theoretical foundations for modeling qualitative judgments and resolving disagreements [27].

In the context of MWPs, difficulty estimation has not yet been systematically integrated with fuzzy decision modeling or validated using multiple expert annotators. Nor has prior work leveraged LLMs to scale the annotation process while maintaining quality through expert feedback loops. This gap motivates the present study, which combines fuzzy

 ISSN 2723-6471

linguistic aggregation, multiple LLM configurations, and a human-in-the-loop validation cycle to create a scalable, reproducible pipeline for MWP difficulty estimation.

3. Methodology

Accurately estimating the difficulty of assessment items is a foundational requirement for adaptive learning systems, computer-based testing, and data-driven instructional design. In adaptive assessments, difficulty estimates guide item selection algorithms, ensuring that learners receive tasks aligned to their current proficiency. Reliable difficulty scores also help educators understand the cognitive demands of each item and maintain fairness across student populations by balancing task complexity. In this work, the terms items, questions, and Math Word Problems (MWPs) are used interchangeably. As large-scale MWP repositories continue to expand, the need for consistent and scalable difficulty annotation becomes increasingly important. While expert annotation remains the gold standard, it is resource-intensive and difficult to sustain across thousands of items. Automated methods particularly those leveraging linguistic models and large language models (LLMs) offer a promising path toward scalable difficulty estimation, provided they are grounded in reliable human judgments. Our decision framework operationalizes this goal by integrating human expertise, fuzzy aggregation, and LLM inference in a unified workflow.

The proposed pipeline consists of six core stages, illustrated conceptually (see Figure 1). This end-to-end process ensures that expert judgments are captured, aggregated, validated, and progressively improved through iterative human-AI collaboration.

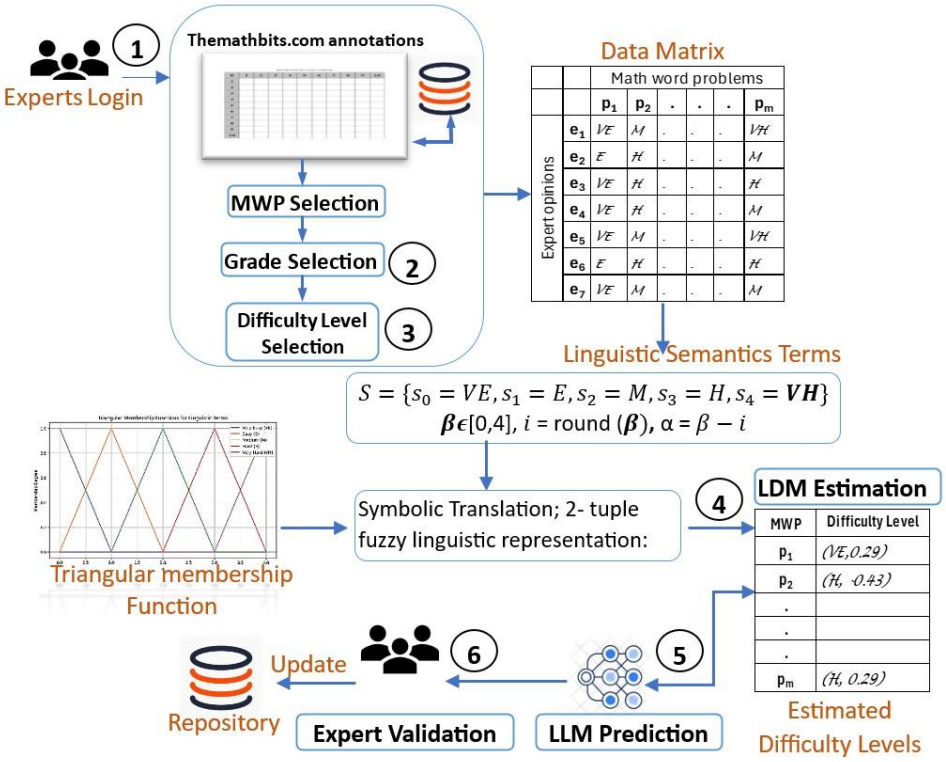

Figure 1. Overview of the proposed human–AI decision workflow for difficulty estimation.

Domain experts begin by registering and logging into the web-based platform, themathbits.com, where mathematical word problems (MWPs) are presented for review through a structured annotation interface. The platform provides clearly defined input fields and simple user controls to support consistent labeling across experts. This interface is designed to guide annotators through the evaluation process while maintaining uniformity in how each problem is assessed. As part of the annotation process, experts first assign each MWP to the appropriate school-grade category using a standardized drop-down menu. These grade annotations provide important pedagogical context, allowing the problems to be interpreted within the framework of curriculum expectations. Capturing grade information also helps ensure that the repository aligns with real classroom progression and can support curriculum-aware applications in the future.

 ISSN 2723-6471

After identifying the grade level, experts label the difficulty of each problem using a five-level semantic scale ranging from very easy to very hard. These ratings collectively form the initial annotation matrix, reflecting the individual judgments of multiple instructors. Because perceptions of difficulty can vary across educators, these ratings naturally contain some degree of disagreement. To reconcile these variations, the annotated difficulty scores are aggregated using a 2-tuple Fuzzy Linguistic Decision Model (FLDM). This aggregation step preserves the semantic meaning of the linguistic labels while also capturing subtle differences in expert opinions. The resulting consensus value represents a stabilized difficulty estimate that serves as the reference label for further analysis.

Once consensus labels are established, multiple large language model (LLM) configurations—including zero-shot, few-shot, and retrieval-augmented generation (RAG) approaches—are applied to predict difficulty levels for the same set of MWPs. These predictions are compared against the fuzzy consensus outputs to assess model alignment and identify configurations that perform well in automated difficulty estimation. Finally, experts review the model predictions, with particular attention to cases where the LLM outputs diverge from the consensus labels. Through this validation step, experts confirm or revise the difficulty annotations as needed. The validated results are then written back to the repository, progressively expanding a curated collection of high-quality MWPs enriched with expert-verified grade and difficulty metadata.

This structured framework creates a scalable annotation ecosystem by combining the strengths of expert intuition, fuzzy linguistic modeling, and LLM-based inference. It also establishes a reproducible methodology for generating high-quality difficulty metadata across diverse mathematical problem types.

3.1. Iterative Validation Procedure

Error! Reference source not found. explains the process. It begins with an initial set of expert annotations $D_0$and alternates between fuzzy linguistic aggregation and LLM-based prediction. Discrepancies are reviewed by experts, who refine the annotations until consensus stabilizes. A stopping threshold of $\epsilon = 0.15$, defined over normalized disagreement, is used to terminate the loop. The expert-FLDM-LLM cycle follows an iterative refinement process summarized below:

Algorithm 1. Human–AI Iterative Validation for Difficulty Consensus

Input:
Initial expert annotations D_0
Output:
Validated difficulty labels

1. ***Initialize Iteration***

$t \leftarrow 0$

2. ***Iterative Refinement***
   - Repeat until convergence or maximum iterations:
   - Compute fuzzy consensus label L_t using the FLDM.
   - Predict difficulty $\hat{L}_t$ using the selected LLM.
   - Compute range-normalized disagreement over the current set:

$$\delta_t \leftarrow \frac{1}{|D_t|} \sum_{p \in D_t} \frac{|L_t(p) - \hat{L}_t(p)|}{4}$$

   (normalization by 4 because difficulty labels are 1–5, so the maximum possible absolute difference is $5 - 1 = 4$).
   - Experts revise annotations to produce D_{t+1}.
   - $t \leftarrow t + 1$
   - If normalized disagreement < ε (ε = 0.15), **stop**.

Return final validated labels.

Disagreement was normalized by the maximum possible difference on the five-level difficulty scale (i.e., 4), ensuring that ε = 0.15 corresponds to an average deviation of less than 0.6 difficulty units. The proposed validation loop offers Improved consensus reliability: FLDM reduces disagreement by retaining granular linguistic information and moderating variability across experts. Scalable difficulty estimation: LLMs accelerate annotation by generating initial

predictions that experts can refine, significantly reducing manual workload. Continuous dataset growth: each iteration expands and strengthens a repository of difficulty-annotated MWPs, supporting downstream adaptive learning and assessment applications.

3.2. Data Repository

To support large-scale difficulty estimation and downstream adaptive learning applications, we constructed a comprehensive repository of Math Word Problems (MWPs) by integrating existing public datasets with newly collected items from educational websites. The initial seed collection consisted of problems from the MATH [16]and ASDiv datasets [15], chosen because they include grade labels and difficulty annotations. These served as the benchmark layer for validating linguistic aggregation and LLM-based predictions. To expand coverage across grades and problem types, we additionally used web crawlers to extract MWPs from high-quality K-12 educational resources such as math-only-math.com and k5learning.com. Several widely used MWP datasets including MathQA [28], GSM8K [11], and ALG514 [14]were also integrated into the platform for annotation and difficulty estimation. Collectively, the repository now contains approximately 24,000 MWPs spanning elementary arithmetic, algebra, geometry, and introductory problem-solving contexts. Each MWP stored in the database includes metadata fields such as problem text, optional solution, source dataset, grade band, and both human-generated and model-generated difficulty levels. The web-based platform (themathbits.com) allows experts to annotate MWPs with two key labels, difficulty level and grade level. Where, the five-level linguistic scale is defined through expert consensus *{1 = Very Easy, 2 = Easy, 3 = Medium, 4 = Hard, 5 = Very Hard} and* grade level: Curriculum bands ranging from Grade 5 to Grade 10.

3.2.1. Annotation Phases

The annotation process proceeded in two phases. Phase 1 - Discrete Difficulty Labels: Seven instructors manually annotated 200 MWPs by selecting a difficulty level from the integer scale (1-5). These labels served as the foundation for initial fuzzy aggregation and inter-rater agreement analysis. Phase 2 - Fine-Grained Subjective Slider: To better capture the nuances of expert perception and reduce information loss due to discretization, we introduced a continuous scale using a slider ranging from 1.0 to 5.0 in increments of 0.1 (e.g., 1.0, 1.1, 1.2, …, 4.9, 5.0). This richer representation allowed deeper analysis of expert variability and enabled the 2-tuple fuzzy linguistic model to operate on more expressive data. More than 2,000 MWPs have been annotated by at least two instructors using this enhanced interface.

Expert annotations were aggregated using either uniform weighting or experience-based weighting, where senior instructors were assigned higher influence during consensus formation. The formal weighting scheme, rationale, and robustness analysis are detailed in section 5.2. So, we have 200 MWPs annotated by seven instructors, another 454 MWPs evaluated by ten instructors and over 2000 MWPs annotated by alteast two instructors on an ongoing basis. Additionally, grade annotations are consolidated using the mode as a descriptive indicator of curricular placement, while difficulty ratings are aggregated independently using fuzzy linguistic modeling. Grade information is not used in difficulty estimation or model evaluation.

3.2.2. Scalability and Quality Control

The repository is designed for extensibility. New MWPs can be bulk-uploaded, annotated, validated, and incorporated into the database through an integrated workflow. Expert activity logs, annotation audits, and aggregated summaries help maintain consistency and ensure high-quality labels. This growing, systematically annotated repository serves both as the training and validation backbone for the proposed difficulty estimation framework and as a reusable resource for future educational data mining research.

3.3. Linguistic Decision-Making Models

In many real-world decision-making scenarios, information cannot be expressed with numerical precision and must instead be described qualitatively. This is particularly true for judgments based on human perception, expertise, or subjective evaluation, such as assessing the difficulty of a mathematics problem or evaluating a student's conceptual understanding. In such cases, linguistic descriptions (e.g., very easy, difficult, moderately challenging) are more natural and interpretable than absolute numeric scores.

Linguistic modeling offers a principled way to formalize and compute with such qualitative information. A linguistic variable is defined by a set of ordered linguistic terms and an associated semantic mapping that assigns each term a fuzzy meaning [26]. Formally, a linguistic variable is represented as a five-tuple $(L, H(L), U, G, M)$, where $L$is the variable name, $H(L)$the term set, $U$the universe of discourse, $G$the syntactic generation rule, and $M$the semantic rule that maps each term to a fuzzy subset of U[17], [20]. This structure allows linguistic assessments to be manipulated, aggregated, and interpreted in a systematic way.

The linguistic approach has been widely applied in domains such as artificial intelligence, human decision-making, education, marketing, and software evaluation [29], [30], [31]. Its appeal lies in its ability to bridge qualitative human reasoning with formal computational processes. Within this family of approaches, Computing with Words (CWW) and fuzzy linguistic computational models provide the foundation for representing expert opinions when numeric data is insufficient or unavailable [27], [32]. A central challenge in the present study is the aggregation of difficulty judgments assigned by multiple instructors to Math Word Problems (MWPs). Since difficulty is inherently subjective, expert annotations often vary due to differences in teaching experience, grade-level familiarity, and interpretation of cognitive complexity. Directly averaging numerical encodings of linguistic labels (e.g., mapping 1–5 to difficulty levels) may oversimplify the nuanced opinions contained in these annotations [33], [34].

Linguistic Decision Models (LDMs) offer a structured solution: they allow experts to express judgments using linguistic terms while enabling the system to aggregate these inputs into a consensus difficulty rating [35]. The decision-making process consists of aggregation and exploitation, where aggregation combines subjective labels from multiple experts into a unified assessment, while exploitation uses the aggregated output to make a final decision or derive an interpretable label. Several linguistic computational models have been proposed, including membership-function-based models, type-2 fuzzy sets, and symbolic approaches [36]. However, many of these suffer from information loss during the translation and re-translation steps required for aggregation.

To overcome this limitation, we employ the 2-tuple fuzzy linguistic model, which preserves the granularity of expert assessments by storing the symbolic term and a numerical "translation value" that captures the deviation from the nearest linguistic term. This model ensures that no information is lost during aggregation, a key advantage for educational applications where subtle differences in perceived problem difficulty may carry instructional significance. In this study, the five-level linguistic term set used for annotating MWPs is (e.g., Very Easy, Easy, Medium, Hard, Very Hard). Expert annotations collected through themathbits.com form the input to the LDM, which aggregates these labels using linguistic computational rules to produce a stable and interpretable difficulty estimate [35], [37]. These aggregated ratings also serve as linguistic ground truth for evaluating and improving LLM-based difficulty prediction models. In this study, "linguistic ground truth" refers to the fuzzy consensus label obtained by aggregating multiple expert difficulty judgments using the 2-tuple linguistic decision model; it represents a consensus-based reference rather than an objective or deterministic label. By implementing the 2-tuple fuzzy linguistic model within this workflow, the system accommodates subjective variation among experts while generating consistent and reproducible difficulty labels suitable for large-scale educational analytics and downstream machine learning.

3.3.1. Linguistic Computational Model

A linguistic computational model provides a formal mechanism for manipulating qualitative information expressed through linguistic terms. Let $S = \{l_0, l_1, \dots, l_h\}$ denote an ordered linguistic term set of odd cardinality $h + 1$, where each term $l_i \in S$ represents a possible value of a linguistic variable (e.g., *Very Easy*, *Easy*, *Medium*, *Hard*, *Very Hard*). During aggregation and computation, intermediate results typically take the form of numerical values $\alpha \in [0, h]$. Because these numerical results must ultimately be mapped back to one of the linguistic terms, an approximation function is required. Formally, the linguistic computational process can be expressed as:

$$S^n \xrightarrow{C} [0, h] \xrightarrow{app_2(\cdot)} \{0, \dots, h\} \to S, \tag{1}$$

C is a symbolic linguistic aggregation operator that combines multiple expert inputs, $app_2(\cdot)$is an approximation function that maps a numerical intermediate result to the nearest linguistic term index in $\{0, \dots, h\}$, the final step retrieves the corresponding linguistic label in S. For this model to behave consistently, the linguistic term set S must satisfy several fundamental properties:

Ordered is defined as

$$l_i \geq l_j \text{if and only if } i \geq j. \quad (2)$$

This ensures that the terms correspond to a natural ranking (e.g., Very Easy < Easy < Medium < Hard < Very Hard).

The negation of a linguistic term is defined as

$$\text{neg}(l_i) = l_{h-i}, \quad (3)$$

which reverses its position in the ordered scale.

Maximum Operator: $\max(l_i, l_j) = l_i \text{if } i \geq j$ and Minimum Operator: $\min(l_i, l_j) = l_i \text{if } i \leq j$ (4)

While this computational model provides a method for aggregating linguistic information, it suffers from an important drawback: information loss. Specifically, when numerical intermediate results are rounded or approximated back to the nearest linguistic index, the underlying nuance in the experts' assessments may be lost. This limitation becomes significant in educational contexts, where small differences in perceived difficulty can meaningfully influence instructional design and adaptive testing. To overcome these limitations, the 2-tuple fuzzy linguistic model was introduced. This model augments each linguistic term with a numerical "translation value" that captures the deviation from the nearest linguistic label. As a result, the 2-tuple representation preserves information throughout the aggregation process and enables more accurate and interpretable difficulty estimations.

In this work, we adopt the 2-tuple fuzzy linguistic decision model to consolidate instructor-provided difficulty ratings for Math Word Problems. The model's ability to retain granularity in expert judgments makes it particularly suitable for establishing high-quality ground truth for downstream LLM-based difficulty prediction.

3.3.2. The 2-Tuple Fuzzy Linguistic Symbolic Representation Model

We define the linguistic term set used for difficulty ratings as:

$$S = \{l_0 = \text{"Very Easy"}, l_1 = \text{"Easy"}, l_2 = \text{"Moderate"}, l_3 = \text{"Hard"}, l_4 = \text{"Very Hard"}\}.$$

Definition and Motivation: When aggregating linguistic labels (e.g., multiple instructor ratings), symbolic linguistic models often map intermediate results to numerical values $\beta \in [0, h]$, where $h$is the cardinality of the scale minus one. Classical linguistic models require rounding β to the nearest linguistic index, resulting in information loss.

To overcome this limitation, 2-tuple fuzzy linguistic representation was introduced [36]. The key idea is to represent a linguistic value not only by its closest linguistic term l_i, but also by a translation parameter αcapturing the deviation from that term. Formally, let $S = \{l_0, \dots, l_h\}$and let $\beta \in [0, h]$be the numerical result of symbolic aggregation. Define:

$$i = \text{round}(\beta), \text{ the index of the closest linguistic term,} \quad (5)$$

$$\alpha = \beta - i, \text{ the symbolic translation, where } \alpha \in [-0.5, 0.5). \quad (6)$$

Then, the pair (l_i, α) constitutes the 2-tuple linguistic representation, preserving both categorical and numerical information. Table 1 shows a sample of three MWPs annotated by four experts, their mapped numerical scale values, the aggregated numerical value, and the final 2-tuple output. This methodology is advantageous in educational assessment scenarios where human raters often disagree on the exact difficulty category.

Table 1. Example of 2-Tuple Fuzzy Linguistic Aggregation for MWP Difficulty Ratings

MWP ID	Expert Ratings	Mapped Values	Aggregated Value	2-Tuple Output	Final Label
101	Easy, Medium, Medium, Hard	2, 3, 3, 4	3.00	(Medium, 0.00)	Medium
102	Very Easy, Easy, Easy, Easy	1, 2, 2, 2	1.75	(Easy, −0.25)	Easy
103	Medium, Hard, Medium, Hard	3, 4, 3, 4	3.50	(Medium, 0.50)	Medium–Hard

Variability among instructor ratings is common (see Figure 2), and the 2-Tuple model provides a mathematically rigorous yet interpretable mechanism to reconcile these differences.

ISSN 2723-6471

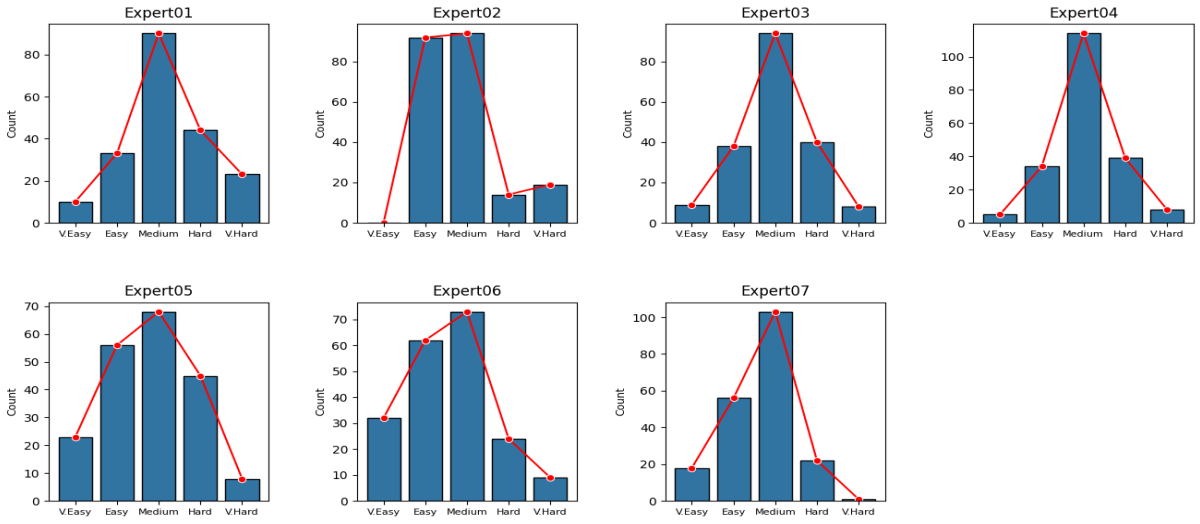

Figure 2. distribution of instructor rating

The model defines two transformation operators, namely the direct mapping and the inverse mapping, which establish a relationship between numerical values and their 2-tuple linguistic representations. The direct mapping operator, denoted as $\Delta : [0, h] \rightarrow S \times [-0.5, 0.5)$, transforms a numerical value β into a 2-tuple representation (l_i, α). The linguistic index i is obtained by rounding β to the nearest integer, while the symbolic translation α represents the deviation between β and its closest linguistic index.

$$\Delta(\beta) = (l_i, \alpha), i = \text{round}(\beta),\ \alpha = \beta - i. \tag{7}$$

$$\text{Boundary handling follows: } \alpha = \begin{cases} \beta - i, & \text{if } i \neq 0, h, \\ \beta, & \text{if } i = 0, \\ \beta - h, & \text{if } i = h. \end{cases} \tag{8}$$

The inverse mapping operator, denoted as Δ^{-1}, converts a 2-tuple (l_i, α) back into its equivalent numerical value by combining the linguistic index and its symbolic translation. Given a 2-tuple (l_i, α), the inverse operator retrieves its equivalent numeric value: $\Delta^{-1}(l_i, \alpha) = i + \alpha$. This formulation ensures that no information is lost during aggregation, one of the main advantages over traditional linguistic approximation methods.

Each instructor's difficulty rating is first treated as a categorical linguistic label. For each MWP, the ratings are aggregated (using a simple or weighted mean), producing a numeric value β. This value is then converted into its 2-tuple representation (l_i, α), which preserves the nearest difficulty category, and the fine-grained deviation representing consensus strength. This enables more interpretable and precise modeling of expert agreement critical for establishing reliable ground truth for downstream LLM training and evaluation. The following algorithm (**Error! Reference source not found.**) summarizes the process used to compute 2-tuple values for the annotated MWPs.

Algorithm 2. Compute 2-Tuple Fuzzy Linguistic Representation

Input:
Difficulty matrix $D = [d_{i,j}]$, where $i = 1, \ldots, n$ denotes MWPs; $j = 1, \ldots, m$ denotes experts; $d_{i,j} \in \{1,2,3,4,5\}$.

Output:
2-tuple representations $(s_i, \alpha_i)$for each MWP.

For each problem p_i: do
a. Compute the average expert rating: $\bar{d}_i = \frac{1}{m} \sum_{j=1}^{m} d_{i,j}$
b. Identify the nearest linguistic term index: $s_i = \text{round}(\bar{d}_i)$.
c. Compute the translation value: $\alpha_i = \bar{d}_i - s_i$.
d. Form the 2-tuple representation: (s_i, α_i).
End For

Return the set of 2-tuples for all MWPs.

ISSN 2723-6471

Prior research has applied similar fuzzy linguistic models in various domains, including competency evaluation [38], e-learning personalization, and group decision-making [13]. However, their application to difficulty estimation for MWPs in combination with LLM predictions remains novel. By integrating the 2-Tuple model with the LLM evaluation framework we establish a hybrid human-AI pipeline that enhances both the reliability and scalability of difficulty classification. The adoption of LDM in this context also supports the Human-in-the-Loop paradigm ensuring that automated predictions remain aligned with expert consensus while allowing for continuous refinement of the model through iterative feedback.

To summarize, the linguistic term set comprised five ordered labels $\{l_0, \ldots, l_4\}$, each associated with a symbolic translation $\alpha \in [-0.5, 0.5)$to capture intra-category variation. Expert annotations in Phase 2 were collected using a continuous slider with a granularity of 0.1, enabling finer discrimination between adjacent difficulty levels. For models based on the extension principle [39], triangular membership functions were normalized, and representativeness weights were set to $(P_1, P_2, P_3) = (0.2, 0.6, 0.2)$, consistent with guidelines proposed in [26], [35]. The continuous ratings are directly consumed by the 2-tuple fuzzy linguistic decision model without prior rounding, allowing fractional deviations to be preserved during aggregation; rounding to discrete difficulty classes is performed only after fuzzy consensus estimation for alignment with LLM outputs.

The expert weights $w_j$were assigned based on instructional experience to reflect differences in pedagogical expertise. Senior instructors (more than 15 years of teaching experience) were allocated a weight of 0.20, while mid-level and junior instructors were each assigned a weight of 0.12, with normalization ensuring that $\sum_j w_j = 1$. This weighting strategy emphasizes the contribution of more experienced educators while retaining balanced representation across raters. A sensitivity analysis demonstrated that perturbing the weights by ±10% resulted in changes of less than 2% in the aggregated difficulty scores, indicating that the F LDM consensus output is robust to moderate variations in expert weighting. For comparison, we implement several commonly used aggregation approaches for combining expert difficulty ratings. A simple baseline method is the simple average, where the unweighted mean of the expert scores d_ijis computed and then rounded to the nearest linguistic label l_i. A related approach is the weighted average, in which ratings from more experienced or senior instructors are assigned higher weights to reflect their domain expertise. Another alternative is the Computing with Words (CWW) framework, which aggregates expert opinions using triangular membership functions defined over the linguistic term set S [27], [28]. The extension principle provides another mechanism for aggregation by minimizing the Euclidean distance between the aggregated value and the membership functions associated with each linguistic term [31]. The extension principle provides another mechanism for aggregation by minimizing the Euclidean distance between the aggregated value and the membership functions associated with each linguistic term [40].

3.4. Large Language Models

Large Language Models (LLMs) are advanced deep learning systems based on the transformer architecture, capable of recognizing, summarizing, translating, predicting, and generating text, code, and other modalities such as images, audio, and video. They have transformed natural language processing by enabling a wide spectrum of applications, including question answering, reasoning, content generation, and dialogue systems [41], [42], [25]. In education, LLMs support personalized tutoring, automated grading, question generation, and accessibility enhancement, thereby expanding opportunities for adaptive and inclusive learning environments.

Despite these advances, LLMs face limitations in quantitative analysis, multi-step reasoning, and domain-specific problem solving. These challenges are especially evident in mathematics education, where problems require precise logical sequencing and symbolic manipulation. Recent work has sought to address these shortcomings through domain-specific models such as MathGPT, which are fine-tuned on mathematical corpora to improve reasoning accuracy [43].

In the context of Math Word Problems (MWPs), existing research has primarily focused on problem classification, equation extraction, solution generation [14], [24], [44], and visual reasoning [45], [46], [47]. However, the estimation of difficulty levels remains underexplored, despite its importance for personalized learning and curriculum design.

Our work addresses this gap by leveraging LLMs to predict the difficulty level of MWPs, complementing human expert annotations obtained through linguistic decision-making models. Difficulty estimation with LLMs requires explicit

prompting because general-purpose models do not contain built-in difficulty taxonomies. We structure prompts by: (1) stating the task objective (difficulty estimation), (2) defining the difficulty scale (1–5), (3) optionally providing grade-level guidance, and (4) specifying the expected output format.

Prompting strategies including zero-shot, one-shot, few-shot, and chain-of-thought were employed to map MWPs to five predefined difficulty levels. Initial experiments revealed substantial divergence across models, with only 44% agreement observed on a 50-problem sample, underscoring the influence of both training domain and prompting strategy. A standard in-context learning paradigm was used to construct prompts with three components: {Role}, {Task}, and {Demonstration}. Zero-shot classification, which requires no prior examples, proved particularly effective when annotated data were limited, while one-shot and few-shot settings incorporated one or more labeled instances for context. Table 2 summarizes the comparative strengths and limitations of each prompting method, including chain-of-thought prompting, which enhanced interpretability but underperformed in accuracy.

Table 2. Summary of Labelling Strategies for Difficulty Classification

Method	Strengths	Limitations
Zero-Shot Prompting	Requires no labeled data; scalable to new tasks; efficient baseline.	May rely on superficial linguistic cues; provides only moderate accuracy.
One-Shot Prompting	Adapts quickly using a single example; simple to implement.	Highly sensitive to the quality of the single exemplar; outputs may vary across prompts.
Few-Shot Prompting	Offers better generalization by utilizing multiple curated demonstrations; captures richer context.	Requires high-quality examples; increases prompt length; may reduce consistency. Sensitive to exemplar selection; examples may anchor predictions to surface features rather than difficulty semantics.
Chain-of-Thought Reasoning	Produces step-wise reasoning, improving interpretability and modeling cognitive processes.	Encourages solution-centric reasoning, leading the model to conflate procedural length with pedagogical difficulty; this misaligns with instructor judgments that emphasize conceptual familiarity and grade-level expectations.

4. Results and Discussions

Accurately classifying the difficulty levels of MWPs in line with human judgment is essential for scalable automated educational tools. We evaluated multiple LLMs, including GPT-3.5-Turbo (Zero-Shot, Five-Shot), GPT-4o.Mini (Zero-Shot, Five-Shot), and RAG variants (RAG-4oMini, RAG-3.5-Turbo) in addition to GPT-5. Model performance was compared against the benchmark difficulty levels determined via the 2-tuple Fuzzy Linguistic Symbolic Representation model. Table 3 summarizes results across Precision, Recall, Accuracy, and F1-score. GPT-3.5-ZS achieved the best performance (F1 = 0.63), aligning closely with the fuzzy model benchmark. Notably, Zero-Shot models outperformed both Five-Shot and RAG variants, which traditionally benefit from contextual information. This suggests that, for well-structured datasets, the directness of Zero-Shot inference may be advantageous.

Figure 3. Distribution of difficulty levels for 454 MWPsFigure 3 shows the distribution of predictions. GPT-3.5-ZS and GPT-4o.Mini-ZS mirrored the ground truth closely, primarily classifying MWPs as *Medium* or *Easy*. Five-Shot models tended to overpredict *Hard* problems, deviating from the benchmark. Reliability was assessed via agreement with human expert labels and inter-model consistency. Zero-Shot configurations not only achieved the highest F1-scores but also demonstrated stable performance across repeated trials. While RAG and Five-Shot models incorporated additional context, they often deviated from the benchmark, particularly overestimating the *Hard* category.

Table 3. Performance metrics: 2-tuple fuzzy (weighted avg.) LDM

LLM Model	Precision	Recall	Accuracy	F1-Score
GPT-3.5-ZS	0.65	0.63	0.63	0.63

ISSN 2723-6471

GPT-4.oMini-ZS	0.58	0.53	0.53	0.55
GPT-3.5-5S	0.54	0.25	0.25	0.34
GPT-4.oMini-5S	0.46	0.37	0.37	0.41
GPT-5-ZS	0.37	0.52	0.48	0.43
GPT-5-CoT	0.37	0.38	0.37	0.38
RAG-4oMini	0.64	0.30	0.30	0.40
RAG-3.5-Turbo	0.50	0.52	0.52	0.51

The reported metrics are descriptive summaries from a single evaluation run and are not accompanied by confidence intervals or significance testing; they are intended to indicate relative alignment trends rather than statistically validated performance differences.

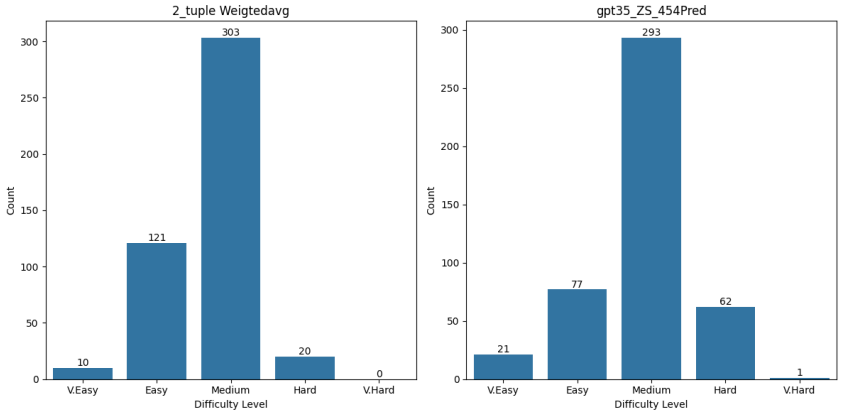

Figure 3. Distribution of difficulty levels for 454 MWPs

To further investigate classification tendencies, we generated ~1000 MWPs with GPT-4o across all difficulty levels and compared predictions from GPT-4o.Mini, LLaMA-3.2, and Gemini Flash (Zero-Shot, Five-Shot, RAG), GPT-5 and GPT-5-CoT. Table 4 shows that most models tended to under-classify *Very Hard* and *Hard* problems, shifting them toward *Medium* or *Easy*. For example, GPT-4o-ZS produced only 1 *Very Hard* case versus 210 in the original set, while LLaMA3.2-RAG significantly overclassified *Hard* MWPs. These shifts suggest difficulties in capturing subtle complexity features, possibly due to differences in model architecture, parameter count, or training corpora.

Table 4. Predictions by LLMs across difficulty levels

Model	**VE**	**E**	**M**	**H**	**VH**
GPT-4o-Gen	200	210	205	206	210
GPT-4o-ZS	258	339	387	46	1
GPT-4o-5S	496	230	248	35	22
GPT-4o-RAG	328	306	231	136	30
GPT-4oMini-ZS	178	280	439	133	1
GPT-4oMini-5S	381	185	397	53	15
GPT-4oMini-RAG	271	251	228	264	17
GPT-5-ZS	311	277	344	94	5
GPT-5-CoT	269	621	134	7	0
GemFlash-ZS	135	314	551	31	0
GemFlash-5S	339	325	353	13	1

GemFlash-RAG	289	335	258	141	8
LLM3.2-5S	159	660	198	2	12
LLM3.2-ZS	200	179	649	0	3
LLM3.2-RAG	150	360	40	477	4

Overall, Zero-Shot GPT-3.5-Turbo emerged as the most reliable, suggesting that for certain structured educational datasets, simplicity in inference can outperform more context-heavy approaches. Few-shot and RAG configurations introduce auxiliary signals-such as exemplar reasoning patterns or retrieved problems with differing curricular assumptions-that can shift the model's focus toward surface complexity rather than pedagogical difficulty. Similar degradation effects due to exemplar bias and over-conditioning have been reported in prior prompt-learning studies [48], [49]. For ordinal educational judgments grounded in expert consensus, such contextual interference can outweigh potential gains from additional information.

An important pattern observed across models is the systematic under-prediction of the 'Very Hard' category. This behavior can be attributed to a combination of class imbalance in the annotated dataset, limited exposure of LLMs to expert-defined extreme difficulty cases, and reliance on surface-level cues that do not fully reflect expert notions of cognitive complexity. Constraining model outputs to discrete labels without uncertainty calibration may further bias predictions toward central difficulty levels. These findings motivate further exploration of hybrid and ensemble strategies to leverage the strengths of different LLM architectures while improving classification fidelity for higher-complexity MWPs.

4.1. Expert validation

This study validates the 2-Tuple Fuzzy Linguistic Symbolic Representation model as a robust benchmark for estimating MWP difficulty levels from expert annotations. Among the evaluated LLMs, GPT-3.5-Turbo-ZS achieved the highest alignment with fuzzy model predictions, making it a strong candidate for automated classification in educational contexts. To test scalability, GPT-3.5-Turbo-ZS was applied to an additional 500 MWPs randomly sampled from a larger dataset. Recognizing the subjectivity inherent in both human and AI-generated labels, we adopted a HITL strategy [6], integrating expert oversight to maintain accuracy, contextual relevance, and ethical compliance. The HITL process was operationalized through the web-based platform themathbits.com, allowing instructors to annotate MWPs via an intuitive slider interface for difficulty levels.

For evaluation, 10 experts independently annotated the difficulty levels of the 500 MWPs, yielding 454 fully labeled problems. The fuzzy model inferred consensus difficulty levels, revealing a majority classified as *Medium*, followed by *Easy*, *Hard*, and *Very Easy*, with none as *Very Hard*. GPT-3.5-Turbo-ZS predictions mirrored this distribution closely, confirming its reliability for this task (see Figure 4***).*** While results are promising, expanding expert diversity and refining annotation guidelines could further enhance the robustness.

Fleiss' κ was used to quantify agreement among multiple experts prior to aggregation, while Cohen's κ was computed post-aggregation to assess alignment between individual expert ratings and the fuzzy consensus label; the two statistics therefore serve complementary, not directly comparable, roles.

ISSN 2723-6471

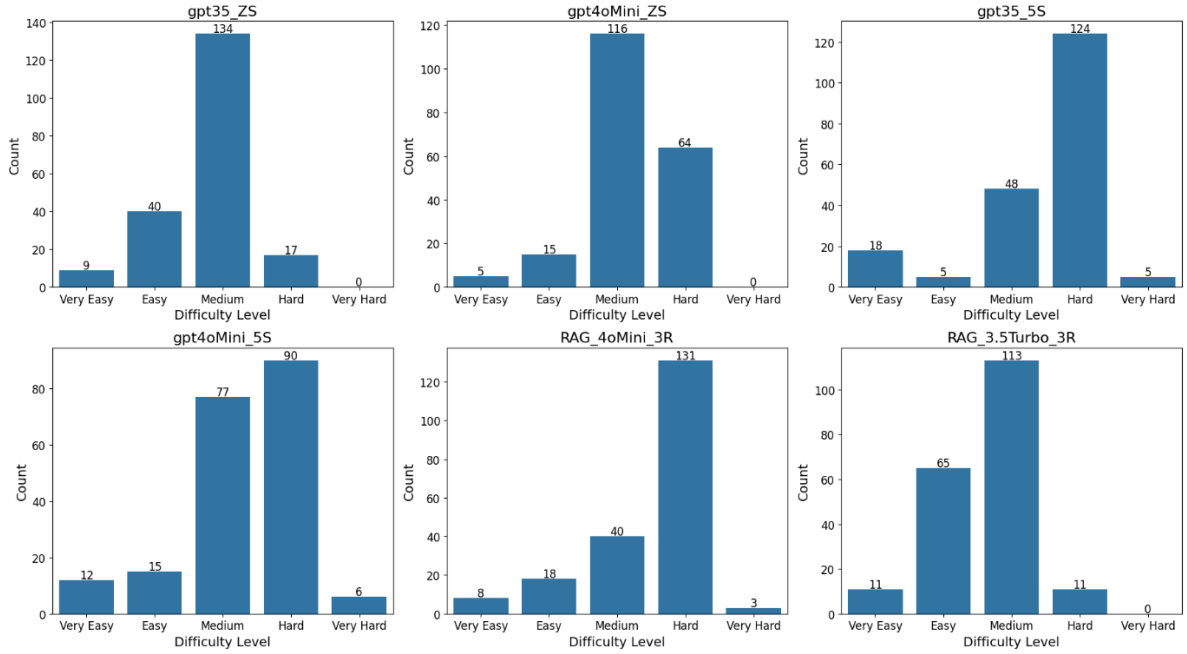

Figure 4. LLM predictions across difficulty levels

4.2. Application: themathbits.com

The platform supports three user types with distinct roles and permissions. Administrators manage registrations, approve accounts, assign credentials, and oversee all edits, logs, and annotations. Instructors can update or add MWPs and perform bulk uploads via Excel templates. Regular users update difficulty levels sequentially from the provided lists. After login, users view MWPs along with metadata including type, description, solution, grade, difficulty, and resource name. Edits are performed through drop-down menus for grade and difficulty. To ensure annotation reliability, clear guidelines are provided, administrative monitoring identifies erratic label changes, and periodic reports highlight deviations for calibration.

The system aggregates results using the mode of grade selections and the average of difficulty ratings. Administrators can export user-specific annotations and activity logs, supporting traceability and reducing bias through broad expert participation. Although time-intensive, this structured, multi-expert annotation process ensures a clean, reliable, and scalable dataset, advancing the long-term goal of building a global repository of graded MWPs with validated difficulty levels.

4.3. Limitations and Future Work

The proposed decision framework explored MWP difficulty-level classification using state-of-the-art LLMs, benchmarked against expert-labeled ground truth. GPT-3.5-Turbo and GPT-4.o.mini were implemented with a temperature of 0 to ensure deterministic outputs and were evaluated under both zero-shot and five-shot settings. For five-shot inference, a representative set of five MWPs one from each difficulty category was curated to guide the models. Evaluation focused on MWPs consistently labeled by human experts, providing a robust benchmark for comparison. Additional models, including Gemini, LLaMA-3, and Mistral, were also tested but demonstrated weaker reasoning coherence for classification.

Interestingly, CoT prompting underperformed compared to zero-shot and few-shot settings, yielding macro-averaged precision, recall, and F1 of approximately 0.38. While CoT improved interpretability by providing explicit reasoning traces, its predictions often diverged from expert-annotated ground truth. Empirically, CoT prompting produces a marked redistribution of predictions toward lower difficulty categories, with minimal assignments to Hard and Very Hard classes Table 4. This distributional shift is accompanied by lower macro-averaged performance (see Table 3), suggesting that CoT alters the model's difficulty calibration rather than improving agreement with expert consensus.In contrast, zero-shot and few-shot prompting map more directly onto the predefined label set, producing more consistent

 ISSN 2723-6471

outputs. These findings suggest that although CoT enhances transparency, its reasoning style does not always align with consensus-based difficulty judgments, highlighting the need for domain-specific fine-tuning or constrained output formats when applying CoT for difficulty classification.

These findings reinforce that MWP difficulty classification is a nuanced, context-dependent process, even among experts. To mitigate rigid labeling, the 2-tuple Fuzzy Linguistic Decision Model was applied, capturing degrees of agreement and providing a consensus-oriented approximation. While difficulty is not a static property, such a standardized initial estimation enables metadata generation, curriculum alignment, and benchmarking for adaptive learning systems. The observed performance gap between LLMs and human experts particularly the deterministic behavior of GPT-3.5-Turbo and GPT-4.o.mini under zero-shot configurations suggests that fine-tuning may yield substantial improvements. Since few-shot and RAG configurations underperformed, future work should focus on domain-specific adaptation of LLMs for MWP difficulty prediction. A systematic qualitative error analysis (e.g., exemplar ablations, retrieval audits, and prompt-failure categorization) is planned as future work.

This study presents encouraging results but also several limitations that constrain generalizability. First, the annotated dataset remains modest in scale 200 MWPs rated by seven instructors in Phase 1 and 454 MWPs evaluated by ten instructors in Phase 2. This size was sufficient for methodological exploration but limits broader applicability. Expanding coverage across grades, curricula, cultural contexts, and languages is essential for building a representative corpus. Second, expert annotations exhibited relatively low agreement, underscoring the inherent subjectivity of difficulty judgments. This variability sets an upper bound on achievable model performance and highlights the need for improved rater calibration. The observed mismatch between grade level and perceived difficulty also reflects the multidimensional nature of difficulty shaped not only by cognitive demand but also by linguistic complexity and domain familiarity. Third, current LLM performance indicates that off-the-shelf models are not yet optimized for difficulty estimation. While GPT-3.5-Turbo zero-shot performed best, few-shot and RAG variants underperformed, suggesting limited domain adaptation. Moreover, the study did not examine supervised fine-tuning or parameter-efficient approaches (e.g., LoRA, QLoRA), which may substantially enhance performance when trained on fuzzy-aggregated labels and richer contextual features such as solution-step complexity or key mathematical constructs. Finally, ethical and robustness considerations were outside the present scope. Small perturbations in wording can mislead LLMs, raising concerns about reliability, fairness, and the potential propagation of bias. Future systems must incorporate explainable AI, adversarial defenses, and safeguards to ensure equitable use in educational environments.

From an operational perspective, the proposed framework is best deployed as a decision-support system rather than a fully automated grader. Instructors can use LLM-generated difficulty estimates as initial signals, which are then refined through fuzzy consensus modeling to accommodate pedagogical context and expert judgment.

Future research will focus on expanding the repository to more than 10,000 MWPs through collaborations with schools and educational institutions, while simultaneously strengthening annotation quality via automated rater-consistency monitoring and the integration of fuzzy consensus thresholds into the workflow. To enhance model accuracy and robustness, domain-specific LLM fine-tuning using supervised or parameter-efficient methods on fuzzy-aggregated labels will be explored. Further, difficulty estimation will be enriched through multi-attribute modeling that incorporates linguistic complexity, algebraic depth, and structural features of MWPs. Finally, robustness and safety will be prioritized by developing adversarial detection mechanisms, improving explainability, and mitigating bias to ensure reliable and equitable deployment in educational settings. By unifying expert insight, fuzzy consensus methods, and domain-adapted LLMs, this work lays a foundation for scalable, reliable, and equitable difficulty estimation systems that can support personalized learning across diverse educational settings.

5. Conclusions

Accurate estimation of MWP difficulty is essential for adaptive and personalized learning; however, most publicly available datasets lack difficulty and grade annotations, limiting their pedagogical utility. This study introduced a framework that combines the 2-tuple Fuzzy Linguistic Decision Model with LLM predictions, benchmarked against expert annotations. Using a dataset of 200 MWPs labeled by seven instructors, subjective ratings were aggregated to derive consensus difficulty levels, and multiple LLMs were evaluated under zero-shot and few-shot settings. Among

these, the zero-shot GPT-3.5-Turbo achieved the highest F1-score of 63%, outperforming GPT-4.o.mini, Gemini, LLaMA-3, Chain-of-Thought, and RAG-based approaches.

To support scalability, we developed *themathbits.com*, a collaborative annotation platform enabling expert labeling and iterative validation of LLM predictions. While the results demonstrate the feasibility of LLM-assisted difficulty classification, several challenges remain, including variability in expert opinions, limited dataset diversity, and broader ethical considerations surrounding automated assessment. Collectively, the findings illustrate both the promise and the complexity of building a reliable, AI-supported ecosystem for difficulty estimation in mathematics education.

6. Declarations

6.1. Author Contributions

Conceptualization: S.K., J.C., P.P.K., and D.K.S.; Methodology: S.K., J.C., P.P.K., and D.K.S.; Software: S.K.; Validation: S.K., J.C., P.P.K., and D.K.S.; Formal Analysis: S.K., J.C., P.P.K., and D.K.S.; Investigation: S.K., J.C., P.P.K., and D.K.S.; Resources: J.C., P.P.K., and D.K.S.; Data Curation: J.C., P.P.K., and D.K.S.; Writing Original Draft Preparation: S.K., J.C., P.P.K., and D.K.S.; Writing Review and Editing: S.K., J.C., P.P.K., and D.K.S.; Visualization: S.K.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

No funding was received for the research and authorship of this article. Partial reimbursement was provided solely to cover publication-related expenses. This support had no role in the study design, experimentation, analysis, interpretation of results, or manuscript preparation.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

6.7. Acknowledgements

The authors gratefully acknowledge the subject matter experts for their invaluable participation in annotating the Math Word Problems. Their time, expertise, and careful judgments were essential in building the dataset and ensuring the quality of the study.

References

[1] R. Koncel-Kedziorski, S. Roy, A. Amini, N. Kushman, and H. Hajishirzi, "MAWPS: a math word problem repository," *arXiv*, vol. 2016, no. Jun., pp. 1152–1157, 2016, doi: 10.18653/v1/N16-1136.

[2] G. Daroczy, M. Wolska, W. D. Meurers, and H. C. Nuerk, "Word problems: a review of linguistic and numerical factors contributing to their difficulty," *Front. Psychol.*, vol. 6, no. Mar., pp. 1–12, 2015, doi: 10.3389/fpsyg.2015.00348.

[3] L. Verschaffel, S. Schukajlow, J. Star, and W. Van Dooren, "Word problems in mathematics education: a survey," *ZDM Math. Educ.*, vol. 52, no. 1, pp. 1–16, 2020, doi: 10.1007/s11858-020-01130-4.

[4] Z. Ding, X. Wang, Y. Wu, G. Cao, and L. Chen, "Tagging knowledge concepts for math problems based on multi-label text classification," *Expert Syst. Appl.*, vol. 267, no. Jan., pp. 1–12, 2025, doi: 10.1016/j.eswa.2024.126232.

[5] S. Kadam, P. K. Srungaram, S. D. Y, M. S. S. S. R, P. Praveen, S. Pappu, and D. K. Satpathi, "Analysis of linguistics and math features for classification of math word problems," *Int. J. Manag. Appl. Sci.*, vol. 9, no. 8, pp. 18–22, 2023.

[6] E. M. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and Á. Fernández-Leal, "Human-in-the-loop machine learning: a state of the art," *Artif. Intell. Rev.*, vol. 56, no. 4, pp. 3005–3054, 2023, doi: 10.1007/s10462-022-10246-w.

[7] Y. Zhang, Y. Luo, Y. Yuan, and A. C.-C. Yao, "Autonomous data selection with language models for mathematical texts," *arXiv*, vol. 2024, no. Feb., pp. 1–12, 2024, doi: 10.48550/arXiv.2402.07625.

[8] X. Yue, X. Qu, G. Zhang, Y. Fu, W. Huang, H. Sun, Y. Su, and W. Chen, "MAmmoTH: building math generalist models through hybrid instruction tuning," *arXiv*, vol. 2023, no. Sep., pp. 1–12, 2023, doi: 10.48550/arXiv.2309.05653.

[9] L. Yu, W. Jiang, H. Shi, J. Yu, Z. Liu, Y. Zhang, J. T. Kwok, Z. Li, A. Weller, and W. Liu, "MetaMath: bootstrap your own mathematical questions for large language models," *arXiv*, vol. 2023, no. Sep., pp. 1–12, 2023, doi: 10.48550/arXiv.2309.12284.

[10] S. Mishra, M. Finlayson, P. Lu, L. Tang, S. Welleck, C. Baral, T. Rajpurohit, O. Tafjord, A. Sabharwal, P. Clark, and A. Kalyan, "LĪLA: a unified benchmark for mathematical reasoning," *arXiv*, vol. 2022, no. Dec., pp. 10074–10092, 2022, doi: 10.18653/v1/2022.emnlp-main.689.

[11] K. Schulman, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman, "Training verifiers to solve math word problems," *arXiv*, vol. 2021, no. Oct., pp. 1–12, 2021, doi: 10.48550/arXiv.2110.14168.

[12] A. Patel, S. Bhattamishra, and N. Goyal, "Are NLP models really able to solve simple math word problems?" *arXiv*, vol. 2021, no. Jun., pp. 2080–2094, 2021, doi: 10.18653/v1/2021.naacl-main.168.

[13] R. M. Rodríguez and L. Martínez, "An analysis of symbolic linguistic computing models in decision making," *Int. J. Gen. Syst.*, vol. 42, no. 1, pp. 121–136, 2013, doi: 10.1080/03081079.2012.710442.

[14] K. Kushman, Y. Artzi, L. Zettlemoyer, and R. Barzilay, "Learning to automatically solve algebra word problems," *arXiv*, vol. 2014, no. Jun., pp. 271–281, 2014, doi: 10.3115/v1/P14-1026.

[15] S.-Y. Miao, C.-C. Liang, and K.-Y. Su, "A diverse corpus for evaluating and developing English math word problem solvers," *arXiv*, vol. 2020, no. Jul., pp. 975–984, 2020, doi: 10.18653/v1/2020.acl-main.92.

[16] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt, "Measuring mathematical problem solving with the MATH dataset," *arXiv*, vol. 2021, no. Dec., pp. 1–12, 2021.

[17] F. Herrera and L. Martínez, "An approach for combining linguistic and numerical information based on the 2-tuple fuzzy linguistic representation model," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 8, no. 5, pp. 539–562, 2000, doi: 10.1142/S0218488500000381.

[18] R. A. Carrasco, M. F. Blasco, and E. Herrera-Viedma, "A 2-tuple fuzzy linguistic RFM model and its implementation," *Procedia Comput. Sci.*, vol. 55, no. Jan., pp. 1340–1347, 2015, doi: 10.1016/j.procs.2015.07.118.

[19] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Inf. Sci.*, vol. 8, no. 3, pp. 199–249, 1975, doi: 10.1016/0020-0255(75)90036-5.

[20] F. Herrera and E. Herrera-Viedma, "Choice functions and mechanisms for linguistic preference relations," *Eur. J. Oper. Res.*, vol. 120, no. 1, pp. 144–161, 2000, doi: 10.1016/S0377-2217(98)00383-X.

[21] F. Soygazi and D. Oguz, "An analysis of large language models and LangChain in mathematics education," *arXiv*, vol. 2024, no. Jan., pp. 1–12, 2024, doi: 10.1145/3633598.3633614.

[22] F. Dilling, "Using large language models to support pre-service teachers' mathematical reasoning—an exploratory study on ChatGPT as an instrument for creating mathematical proofs in geometry," *Front. Artif. Intell.*, vol. 7, no. Jan., pp. 1–12, 2024, doi: 10.3389/frai.2024.1460337.

[23] Z. Levonian, C. Li, W. Zhu, A. Gade, O. Henkel, M.-E. Postle, and W. Xing, "Retrieval-augmented generation to improve math question-answering: trade-offs between groundedness and human preference," *arXiv*, vol. 2023, no. Oct., pp. 1–12, 2023, doi: 10.48550/arXiv.2310.03184.

[24] K. Zaporojets, G. Bekoulis, J. Deleu, T. Demeester, and C. Develder, "Solving arithmetic word problems by scoring equations with recursive neural networks," *Expert Syst. Appl.*, vol. 174, no. Jan., pp. 1–12, 2021, doi: 10.1016/j.eswa.2021.114704.

[25] R. Meissner, A. Pögelt, K. Ihsberner, M. Grüttmüller, S. Tornack, A. Thor, N. Pengel, H.-W. Wollersheim, and W. Hardt, "LLM-generated competence-based e-assessment items for higher education mathematics: methodology and evaluation," *Front. Educ.*, vol. 9, no. Jan., pp. 1–12, 2024, doi: 10.3389/feduc.2024.1427502.

[26] F. Herrera and E. Herrera-Viedma, "Linguistic decision analysis: steps for solving decision problems under linguistic information," *Fuzzy Sets Syst.*, vol. 115, no. 1, pp. 67–82, 2000, doi: 10.1016/S0165-0114(99)00024-X.

[27] F. Herrera and L. Martínez, "Computing with words in decision making," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 746–752, 2000, doi: 10.1109/91.890332.

[28] A. Amini, S. Gabriel, S. Lin, R. Koncel-Kedziorski, Y. Choi, and H. Hajishirzi, "MathQA: towards interpretable math word problem solving with operation-based formalisms," *arXiv*, vol. 2019, no. Jun., pp. 2357–2367, 2019, doi: 10.18653/v1/N19-1245.

[29] T. Malhotra and A. Gupta, "A systematic review of developments in the 2-tuple linguistic model and its applications in decision analysis," *Soft Comput.*, vol. 27, no. 3, pp. 1871–1905, 2023, doi: 10.1007/s00500-020-05031-2.

[30] C.-K. Law, "Using fuzzy numbers in educational grading system," *Fuzzy Sets Syst.*, vol. 83, no. 3, pp. 311–323, 1996, doi: 10.1016/0165-0114(95)00298-7.

[31] H.-M. Lee, "Group decision making using fuzzy sets theory for evaluating the rate of aggregative risk in software development," *Fuzzy Sets Syst.*, vol. 80, no. 3, pp. 261–271, 1996, doi: 10.1016/0165-0114(95)00201-4.

[32] F. Herrera, E. Herrera-Viedma, S. Alonso, and F. Chiclana, "Computing with words in decision making: foundations, trends and prospects," *Fuzzy Optim. Decis. Making*, vol. 8, no. 4, pp. 337–364, 2009, doi: 10.1007/s10700-009-9065-2.

[33] L. Martínez, R. M. Rodríguez, and F. Herrera, "The 2-tuple linguistic model: computing with words in decision making," *Springer*, vol. 2015, no. Jan., pp. 1–12, 2015, doi: 10.1007/978-3-319-24714-4_7.

[34] L. Martínez, "Computing with words in linguistic decision making: analysis of linguistic computing models," *IEEE Access*, vol. 2010, no. Dec., pp. 5–8, 2010, doi: 10.1109/ISKE.2010.5680783.

[35] F. Herrera, E. Herrera-Viedma, and J. L. Verdegay, "A rational consensus model in group decision making using linguistic assessments," *Fuzzy Sets Syst.*, vol. 88, no. 1, pp. 31–49, 1997, doi: 10.1016/S0165-0114(96)00047-4.

[36] L. M. Herrera, "A model based on linguistic 2-tuples for dealing with multigranular hierarchical linguistic contexts in multi-expert decision-making," *IEEE Trans. Syst. Man Cybern.*, vol. 31, no. 2, pp. 227–234, 2001, doi: 10.1109/3477.915345.

[37] F. Herrera and L. Martínez, "A model based on linguistic 2-tuples for dealing with multigranular hierarchical linguistic contexts in multi-expert decision-making," *IEEE Trans. Syst. Man Cybern.*, vol. 31, no. 2, pp. 227–234, 2001, doi: 10.1109/3477.915345.

[38] Z. Xu and H. Wang, "On the syntax and semantics of virtual linguistic terms for information fusion in decision making," *Inf. Fusion*, vol. 34, no. Jan., pp. 43–48, 2017, doi: 10.1016/j.inffus.2016.06.002.

[39] D. Dubois and H. Prade, "Fuzzy sets and systems: theory and applications," *Fuzzy Sets Syst.*, vol. 1997, no. Jan., pp. 1–12, 1997, doi: 10.1016/S0076-5392(09)60129-6.

[40] R. R. Yager, "An approach to ordinal decision making," *Int. J. Approx. Reason.*, vol. 12, no. 3–4, pp. 237–261, 1995, doi: 10.1016/0888-613X(94)00035-2.

[41] A. Scarlatos and A. Lan, "Tree-based representation and generation of natural and mathematical language," *arXiv*, vol. 2023, no. Feb., pp. 1–12, 2023, doi: 10.48550/arXiv.2302.07974.

[42] J. Jaeho and S. Lee, "Large language models in education: a focus on the complementary relationship between human teachers and ChatGPT," *Educ. Inf. Technol.*, vol. 2023, no. Jan., pp. 1–12, 2023, doi: 10.1007/s10639-023-11834-1.

[43] J. Ahn, R. Verma, R. Lou, D. Liu, R. Zhang, and W. Yin, "Large language models for mathematical reasoning: progresses and challenges," *arXiv*, vol. 2024, no. Feb., pp. 1–12, 2024, doi: 10.48550/arXiv.2402.00157.

[44] H. M. Javad, H. Hannaneh, and E. O. Nate, "Learning to solve arithmetic word problems with verb categorization," *arXiv*, vol. 2014, no. Oct., pp. 523–533, 2014, doi: 10.3115/v1/D14-1058.

[45] Y. Wang, X. Liu, and S. Shi, "Deep neural solver for math word problems," *arXiv*, vol. 2017, no. Sep., pp. 845–854, 2017, doi: 10.18653/v1/D17-1088.

 ISSN 2723-6471

[46] L. Wang, D. Zhang, C. Chen, and P. Liang, "Program induction by rationale generation: learning to solve and explain algebraic word problems," *arXiv*, vol. 2017, no. Jul., pp. 158–167, 2017, doi: 10.18653/v1/P17-1015.

[47] X. Sun, X. Li, J. Li, F. Wu, S. Guo, T. Zhang, and G. Wang, "Text classification via large language models," *arXiv*, vol. 2023, no. May, pp. 1–12, 2023, doi: 10.48550/arXiv.2305.08377.

[48] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, and L. Zettlemoyer, "Rethinking the role of demonstrations: what makes in-context learning work?" *arXiv*, vol. 2022, no. Dec., pp. 11048–11064, 2022, doi: 10.18653/v1/2022.emnlp-main.759.

[49] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and N. Graham, "Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing," *arXiv*, vol. 2023, no. Jul., pp. 1–12, 2023, doi: 10.48550/arXiv.2107.13586.