International Journal of Mathematical, Engineering and Management Sciences
Vol. 11, No. 2, 679-705, 2026
https://doi.org/10.33889/IJMEMS.2026.11.2.028

Automated Estimation of Difficulty Levels in Math Word Problems using Linguistic, Mathematical, and Semantic Features

Shilpa Kadam
Department of Mathematics,
BITS Pilani, Hyderabad Campus, Telangana State, India.
Corresponding author: p20190508@hyderabad.bits-pilani.ac.in

Shoukhi Khan
IBM, Bengaluru, Karnataka, India.
E-mail: shoukhan@in.ibm.com

Jabez Christopher
Department of Department of Computer Science and Information Systems,
BITS Pilani, Hyderabad Campus, Telangana State, India.
E-mail: jabezc@hyderabad.bits-pilani.ac.in

PTV Praveen Kumar
Department of Mathematics,
BITS Pilani, Hyderabad Campus, Telangana State, India.
E-mail: praveen@hyderabad.bits-pilani.ac.in

Dipak Kumar Satpathi
Department of Mathematics,
BITS Pilani, Hyderabad Campus, Telangana State, India.
E-mail: dipak@hyderabad.bits-pilani.ac.in

(Received on December 4, 2025; Revised on January 19, 2026; Accepted on February 4, 2026)

Abstract
Math Word Problems (MWPs) remain challenging for learners due to linguistic complexity, mathematical reasoning demands, and contextual variability. Accurately estimating item difficulty is essential for adaptive learning and automated assessment, yet many existing approaches rely on expert annotation or Item Response Theory (IRT), which are resource-intensive and difficult to scale to new items. This paper proposes IDEA, an integrated data-driven framework that extracts linguistic, mathematical, and semantic embedding features to predict MWP difficulty on a five-level scale. Using 4,244 algebra problems from the MATH dataset, we evaluate multiple feature sets and models, showing that embedding-based representations outperform handcrafted features; on a held-out test set, (Macro-F1 = 0.40 vs. 0.29). Since difficulty levels are ordinal, we additionally report ordinal-aware evaluation: an ordinal regression model achieves MAE = 1.08, quadratic weighted kappa = 0.37, and within-one-level accuracy of 0.71, indicating that most predictions are close even when exact matching is difficult. Model-interpretability analysis using SHAP highlights readability and sentence-structure features as dominant contributors to predicted difficulty; exploratory SEM analysis is included to examine relationships among feature groups but is interpreted cautiously due to limited global fit. Finally, external validation using seven expert ratings and IRT estimates from 61 students suggests variability in human judgment while supporting the practical utility of automated calibration. Overall, IDEA provides a scalable approach to item calibration and helps mitigate cold-start challenges in adaptive learning settings as well as contribute in fine-tuning large language models.

Keywords- Math word problems, Item difficulty, Item response theory, Adaptive learning systems, Data-driven approach, Word embeddings, Difficulty estimation, SHAP.

1. Introduction

The increasing presence of Artificial Intelligence (AI) in education has opened new possibilities for understanding how learners interact with instructional content and how assessments can be designed more intelligently. In both classroom and digital settings, AI-enabled tools are now being used to personalize instruction, generate timely feedback, and help educators interpret large volumes of learner data. One area that has recently received growing attention is the automatic estimation of question difficulty, a problem that sits at the intersection of machine learning, psychometrics, and educational design. Difficulty is not merely a descriptive tag; it influences whether an item is instructionally appropriate, affects the reliability and validity of test scores, and determines how accurately assessments reflect learners' strengths and gaps. A recent survey of automatic difficulty prediction (AlKhuzaey et al., 2024) methods indicate that much of the existing work has centered on language learning, with comparatively little attention given to Mathematics. Only 19% of studies focus on domain-independent contexts, including math word problems (MWPs), despite the fact that MWPs pose unique challenges. In particular, only a small portion of studies address domain-independent settings such as math word problems (MWPs), even though MWPs present distinct challenges: they require learners to construct mathematical meaning from language, integrate contextual information with formal operations, and often bridge multiple concepts within a single narrative. Subtle shifts in wording, mathematical vocabulary, and the number of conceptual "jumps" demanded by the text can substantially alter how difficult a problem feels and how it performs in practice, making consistent calibration difficult without sustained expert effort.

Conventional approaches rely either on expert judgments or on psychometric models such as Item Response Theory (IRT). Both methods have clear limitations: expert ratings are subjective and difficult to scale, while IRT requires large amounts of student response data and can only estimate difficulty after an assessment has been administered. This creates a practical gap-newly created or computer-generated MWPs lack difficulty labels, yet accurate labels are essential for adaptive learning systems, test design, and curriculum planning. Beyond predictive performance, this work addresses a key bottleneck for supervised fine-tuning of language models in education, namely the scarcity and subjectivity of reliable difficulty labels.

AI-based approaches offer a promising alternative because they can analyze problem text directly, taking into account linguistic complexity, mathematical structures, and cognitive demands that influence learners' performance. Adaptive Learning Systems (ALS) depend heavily on accurate difficulty labels to guide content sequencing, provide differentiated instruction, and maintain a smooth learning progression. Before such systems can tailor questions to a learner's proficiency, each question must be reliably calibrated (Wauters et al., 2010). Prior studies have explored how structural and linguistic factors contribute to difficulty, and some have applied machine learning techniques using hand-engineered features or counts of solution steps (Pelánek et al., 2022; Theephoowiang & Chaowicharat, 2022). However, existing work often examines these factors in isolation, and few studies systematically compare traditional feature-based models with modern embedding-based representations, or reconcile machine-generated difficulty with both instructor judgment and IRT-based estimates. In this work, we introduce IDEA (Item Difficulty Estimation and Automation)-a unified, data-driven framework designed to automatically estimate the difficulty of math word problems. IDEA integrates insights from natural language processing, mathematics education, and interpretable machine learning. The contributions of this study are fourfold:

- We develop a comprehensive analytic framework that extracts linguistic features, mathematical vocabulary cues, and semantic embeddings, and applies machine learning models to predict difficulty levels of MWPs.
- We explore feature influence, using SHAP values and Structural Equation Modeling (SEM) to better understand how linguistic and mathematical factors jointly shape item difficulty.
- We conduct a systematic evaluation of a diverse set of machine-learning classifiers on the MATH

dataset, and report a careful comparison of how different feature representations influence predictive performance.
- We examine how closely the model-based difficulty estimates align with instructor ratings and IRT-derived difficulty, highlighting where human judgments vary and where data-driven methods yield more consistent difficulty signals.
- We introduce the IDEA framework as a practical route to estimating difficulty directly from problem statements, without relying on learner-response data or manual labeling. This capability supports scalable dataset development, adaptive assessment design, and downstream fine-tuning of domain-specific models.

Together, these contributions aim not only to improve the automation of difficulty estimation but also to offer educators deeper insight into what makes MWPs challenging for learners. The work also addresses the practical "cold-start" problem in adaptive systems by providing difficulty estimates even in the absence of learner response data. This study operationalizes MWP difficulty using quantifiable properties of problem statements to evaluate how closely automated methods can approximate expert-labeled difficulty. We do not claim to measure students lived difficulty experiences directly; rather, we position this as a computational and measurement-oriented first step. Qualitative factors (e.g., learner prior knowledge, affect, classroom context) are discussed as important but outside the present scope. Section 2 reviews related scholarship, Section 3 describes the data and feature extraction procedures, and Section 4 presents the experimental setup and modeling. Section 5 outlines the performance evaluation of models and in Section 6 we discuss the results and interpretations. Section 7 compares our estimates with traditional approaches, and Section 8 concludes with implications for future research in educational assessment and AI-supported learning.

2. Related Work

Math word problems (MWPs) play a central role in assessing learners' mathematical understanding and their ability to apply concepts in real-world scenarios. MWPs typically present a written narrative describing a situation and ask learners to reason about one or more unknown quantities. Traditionally, the difficulty of such problems has been determined either through expert judgment or through analysis of student performance. With recent advances in machine learning (ML) and AI techniques, however, there has been growing interest in automating this process. Such automation depends heavily on identifying relevant linguistic, mathematical, and cognitive features that influence problem difficulty—an area that has received notable attention in the literature.

Because MWPs are linguistically rich and often verbose, recent research has focused not only on solving or reasoning over MWPs but also on generating them using large language models (LLMs) (Zhang et al., 2019; Kurdi et al., 2020; Lan et al., 2022; Benedetto, 2023). Complementing this trend, several studies have examined how elements of language and mathematical vocabulary contribute to difficulty. A meta-analysis, for instance, reported a strong positive correlation between math vocabulary and learners' mathematical performance (Lin et al., 2021). Other work using non-parametric analysis found statistically significant differences associated with the presence of fractional values, rational-number operations, and their contrast with natural-number reasoning in arithmetic word problems (Sanz et al., 2020). These observations are illustrated by the presence of cue words such as three times, twice, or annually, which often signal underlying mathematical structures in MWPs.

Beyond vocabulary, prior research explores the wider set of factors that contribute to perceived problem difficulty. These studies point to the influence of social background, domain-specific knowledge, prior exposure to mathematical terminology, learning disabilities, reasoning skills, cognitive development, and

reading proficiency (Daroczy et al., 2015). Closely related work has also investigated the role of readability in MWPs; for instance, analyses of mathematics textbooks from Grades 2 to 6 show how linguistic complexity can interact with learners' mathematical understanding and, in turn, affect item difficulty (Acosta-Tello, 2010). In a complementary direction, researchers have classified MWPs into structural templates: such as change, compare, combine, and division–multiplication, and proposed rule-based or hybrid approaches to identify operations, extract quantities, and generate mathematical expressions from text (Mandal & Naskar, 2021). While these efforts have advanced our understanding of what makes a word problem challenging, several studies argue that the primary obstacle is often conceptual understanding rather than vocabulary per se, particularly when students struggle to translate a narrative into a solvable mathematical form (Barbu & Beal, 2010). Supporting this view, confirmatory factor analysis has shown a strong association between math difficulty and text difficulty (Unson, 2021).

Further, we have approached MWP difficulty from a structural and semantic lenses, considering factors such as symbolic representations, linguistic cues, numerical cognition, operator complexity, inconsistent lexicons, and spatial descriptions (Verschaffel et al., 2020). Some studies explore how students identify symbolic relationships in MWPs and how this ability varies with difficulty level (Sunde et al., 2023). Typically, accuracy increases with grade level and prior achievement but decreases as MWP difficulty rises, especially for items involving comparison or multi-step reasoning. Difficulty labeling methods for MWPs generally fall into two categories: (a) expert-derived labels and (b) learner-derived labels via performance metrics. Instructor-based labeling depends heavily on expertise and is time-consuming, while psychometric models such as classical test theory (CTT) and Item Response Theory (IRT) estimate item difficulty from student response patterns (Baker, 2001). Although IRT is widely used, it cannot be applied to newly created or computer-generated questions due to its dependence on large-scale response data. Alternative estimation methods-including proportion correct, learner feedback, expert comparison, and Elo ranking-have been evaluated, with proportion correct demonstrating the strongest correlation with IRT difficulty (Wauters et al., 2012). Evidence also suggests systematic differences in perceived difficulty: students often rate items as harder than teachers do, especially at lower difficulty levels, underscoring the importance of clearer wording and stronger supports for comprehension (Pérez et al., 2019). A modest yet meaningful correlation has also been found between predicted and actual difficulty among standardized test items (Attali et al., 2014). Several earlier works gathered qualitative and survey-based insights to understand the sources of student difficulty. Learners frequently struggle with decoding vocabulary, interpreting graphics, identifying relevant versus irrelevant information, generating number sequences, performing calculations, and conceptualizing solutions (Gooding, 2009, Cetintas et al., 2014; Zollman, 2020). Problem-solving performance has been shown to depend strongly on reading comprehension and arithmetic fluency (Pongsakdi et al., 2016). Other studies emphasize the importance of recognizing mathematical vocabulary and point to calculation skills, operator identification, text comprehension, and reasoning ability as key predictors of success (Sepeng & Madzorera, 2014; de Blas et al., 2021).

Despite these rich insights, relatively little work has focused on directly predicting the difficulty level of MWPs. Some studies considered structural attributes-for example, number of steps, numerical complexity, logarithmic operations, symbol variety, or student familiarity-to classify difficulty (Lee & Heyworth, 2000). Others applied regression models such as SVCs, ANNs, and Naïve Bayes using symbolic or step-count features, evaluating performance primarily through Mean Absolute Error (MAE) (Theephoowiang & Chaowicharat, 2022). However, features like the number of solution steps often require expert annotation and may be unreliable for automatically generated MWPs. More importantly, existing work rarely addresses the cold-start problem, where new items must be labeled without student-response data. Additionally, unlike prior work that emphasizes algorithmic novelty, this study focuses on problem-level novelty by addressing cold-start difficulty estimation and systematically comparing machine learning predictions with

expert judgment and psychometric models.

Our work extends this literature by adopting a fully data-driven approach grounded in natural language processing to extract linguistic, structural, and semantic features, contributing to the taxonomy outlined in **Figure 1**. We propose a comprehensive framework for identifying difficulty-influencing attributes, selecting suitable classifiers, and evaluating predictive performance. The IDEA framework encapsulates the end-to-end process-from preprocessing and feature extraction to embedding generation, classifier training, and model validation. A curated sample of MWPs representing all difficulty levels is used to benchmark the models, and results are further compared against traditional approaches. The following sections describe the components of this framework and present key findings from our analyses.

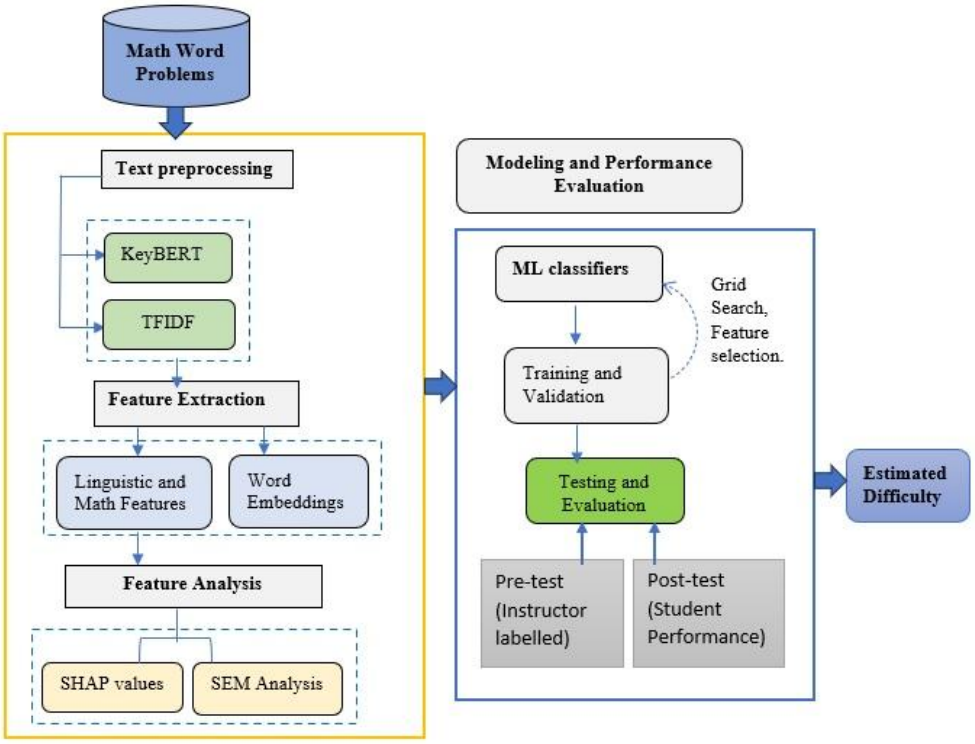

Figure 1. IDEA: item difficulty estimation and automation framework.

3. Data Pre-processing, Feature Extraction and Analysis

Estimating item difficulty supports several core tasks in assessment design and adaptive learning, including sequencing practice questions, calibrating tests, and gauging learners' knowledge levels. Recently, this problem has attracted increased attention in language learning, particularly with the advent of large language models (LLMs), as highlighted in recent survey papers (Benedetto, 2023; AlKhuzaey et al., 2024). In mathematics education, understanding the factors that influence the difficulty of MWPs is equally important, especially for automation and large-scale deployment. This involves examining number patterns, symbols, equations, expressions, mathematical vocabulary, and linguistic complexity. While these aspects have been studied primarily in relation to learner performance, there remains significant scope to employ them directly for automated difficulty estimation. In this section, we describe the dataset, pre-processing pipeline, feature extraction, and feature analysis methods used to support item difficulty estimation.

3.1 Dataset

The Mathematics Aptitude Test of Heuristics (MATH) dataset comprises 12,500 problems (7,500 training and 5,000 test) curated to support automated solving of math word problems (Hendrycks et al., 2021). The problems were collected from Khan Academy and competitive exams such as AMC 10, AMC 12, and AIME, which assess the problem-solving abilities of learners in K-12 settings. The dataset spans a range of subjects and difficulty levels. The seven subjects include Pre-algebra, Algebra, Number Theory, Counting and Probability, Geometry, Intermediate Algebra, and Pre-calculus. Problem difficulty is encoded on a 5-point scale (1–5) following the Art of Problem Solving (AoPS) convention. According to the MATH documentation, the labels correspond to: 1 (Beginner), 2 (Motivated/Novice Beginner), 3 (Intermediate), 4 (High-level), and 5 (Expert). Automatically classifying MWPs remains challenging, partly due to the difficulty of obtaining large, reliably labeled datasets. In this study, we focus on algebraic word problems, which are conceptually simple in form but highly diverse in structure and linguistic realization, making them nontrivial for machine classification. The difficulty levels (1–5) are ordinal While some subjectivity in difficulty labeling is expected, we adopt the MATH difficulty annotations as our ground truth baseline.

The MATH dataset does not provide an explicit item-level mapping from each problem to a specific grade. Consequently, we do not stratify results by grade or claim that "advanced topics" are uniformly more difficult, since difficulty may also reflect language demands, multi-step reasoning, and unfamiliar representations. To maintain a consistent evaluation context, we administered the post-test to a single cohort of Grade 10 students. Accordingly, our findings should be interpreted relative to this Grade 10 cohort and the dataset labels rather than as a cross-grade comparison; establishing grade/topic mappings and validating difficulty across cohorts remains future work.

3.2 Text Pre-processing

From the MATH dataset, we combined MWPs from the Pre-algebra, Intermediate Algebra, and Algebra folders, yielding 4,244 MWPs in the training partition and 3,591 MWPs in the test partition. For this study, we use the 4,244 MWPs from the training folder as our primary dataset. Each word problem is originally stored as a *.json file. We read these files using Python and converted them into tabular form, saving them as Train.csv and Test.csv. After an initial inspection of the dataset, we discarded MWPs with missing difficulty labels and those with more than 50 words reducing to 3238 MWPs. The distribution of MWPs by difficulty level is shown in **Table 1**. Feature analysis plays a crucial role in understanding which attributes influence model performance, guiding both feature selection and feature engineering. By identifying irrelevant or redundant features, we can reduce overfitting and improve generalization. In this work, we use unsupervised techniques such as KeyBERT and TF-IDF, as well as supervised approaches such as Structural Equation Modeling (SEM) and SHAP (SHapley Additive exPlanations), to uncover attributes that are most predictive of difficulty levels in MWPs.

Table 1. Distribution of MWPs in MATH dataset.

Difficulty level	Train folder	Test folder
1	362	273
2	836	506
3	944	680
4	985	722
5	1117	780

KeyBERT: To explore how keyword usage varies with difficulty, we applied KeyBERT to the MWPs at each difficulty level separately. Except for a small set of common terms, we observed considerable variation in the prominent keywords between levels, as shown in **Table 2**. For example, words such as *point*, *function*,

and *equation* are more frequent in level 5 items, whereas words such as *sum*, *difference*, and *probability* are more typical at level 1. This suggests that lexical cues can reflect increasing conceptual and symbolic complexity. Certain terms such as *value* and *number* appear frequently across all levels.

Table 2. Keywords identified for each difficulty level based on KeyBERT.

Difficulty level	Keywords extracted using KeyBERT
1	sum, probability, fractional, large, perimeter, double, difference, percent, divisible, positive, figure, obtuse, etc.
2	sum, square, multiple, mean, subtract, coordinate, product, solve, compute, angle, prime, etc.
3	perimeter, circle, average, smallest, divided, expression, sequence, degree, quadratic, total, probability, evaluate, etc.
4	factor, integer, polynomial, equation, prime, square, intercept, solution, quadratic, minimum, constant, root, etc.
5	graph, parabola, fraction, sequence, point, function, integer, root, equation, number, solve, etc.

TF–IDF: Term Frequency–Inverse Document Frequency (TF–IDF) scores provide another perspective on term importance. TF–IDF assigns a weight to each term based on its frequency within a document and its rarity across the entire collection. We computed TF–IDF scores for all MWPs and ranked terms by their contribution within each difficulty level **Table 3**. The resulting keyword profiles again highlight meaningful distinctions between levels and support the hypothesis that lexical patterns, especially mathematical vocabulary, are informative for difficulty classification.

Table 3. Keywords identified for each difficulty level based on TF-IDF.

Difficulty level	Keywords extracted using TF-IDF matrix
1	sum, integer, perimeter, product, divisible, compute, odd, greatest, measure, result, value, solutions, second, distance, odd, minute, xcirc, hour, second, zero, etc.
2	subtract, factor, product, square, angle, prime, coordinate, area, simplify, solution, value, area, etc.
3	root, graph, square, factor, solve, perimeter, add, largest, evaluate, number, divisor, equation, asymptote, rectangle, etc.
4	polynomial, product, diagonal, rectangle, graph, equation, greatest, smallest, intersect, remainder, triangle, etc.
5	nearest, divisor, average, center, coefficient, lowest, origin, quadratic, range, minimum, maximum, etc.

Overall, the KeyBERT and TF–IDF analyses indicate that certain keywords and lexical fields are associated with higher or lower difficulty levels, providing a useful starting point for systematic feature derivation.

3.3 Feature Extraction

Prior work has proposed hybrid approaches that combine machine learning with rule-based methods to classify MWPs into categories such as *change*, *combine*, *compare*, and *equalize* based on textual semantics (Mandal & Naskar, 2021). Other studies have shown that complexity, defined in terms of sentence structure, can significantly influence perceived difficulty (Pelánek et al., 2022). MWPs have also been grouped according to situation types (e.g., proportion, unity, rate, sum, motion), repeated number occurrence, expression length, and expression depth, particularly in evaluating transformer-based solvers (Chen et al., 2022). Building on these insights and our keyword analysis, we derive three main families of features: mathematical, linguistic, and embedding-based.

Mathematical Features: Unlike traditional text pre-processing pipelines, which typically remove stop words and punctuation, we retain the full text to extract mathematical symbols, quantities (both numeric and textual), and expressions, in addition to mathematical cue words. Our working hypothesis is that heavier use of mathematical vocabulary, long or nested expressions, larger magnitudes (e.g., large fractions or numbers), and complex symbolic structure are indicative of higher difficulty. Using regular expressions in Python, we compute features such as: number of mathematical symbols per sentence, number of numeric values per sentence, number of math vocabulary terms per sentence, and total counts of mathematical cue

words. Since many MWPs in the MATH dataset contain LaTeX-encoded expressions, we detect and decode/normalize LaTeX (e.g., fractions, roots, comparison operators) prior to feature extraction. We also construct a mathematical vocabulary lexicon that includes, but is not limited to: cue words (e.g., *divide*, *fraction*, *proportion*), phrases (e.g., *improper fraction*, *greater than*, *twice as much*, *half of*), abbreviations (e.g., *min* for minutes), symbols (e.g., «, %, *, -,,), number words (e.g., *zero*, *four*, *ten*), quantity descriptors (e.g., *more*, *less*, *little*), and spatial terms (e.g., *below*, *under*, *end*). The normalized text is then sentence-split and tokenized in a symbol-preserving manner so that mathematical operators and special characters remain available for downstream extraction. This lexicon, coupled with associated statistics, provides a compact yet informative representation of the mathematical content of each problem and is consistent with complexity indicators discussed in (Pelánek et al., 2022).

Linguistic Features: A substantial body of research suggests that linguistic features have a strong influence on the difficulty of MWPs, motivating the development of readability indices and other quantitative measures of textual complexity. Readability formulas for English often incorporate variables such as sentence length, word length, vocabulary difficulty, and syllable counts. Tools like Coh-Metrix and Linguistic Inquiry and Word Count (LIWC) have been used to examine how linguistic properties of mathematics story problems relate to student performance (Walkington et al., 2015). More recently, the LXPER index has been proposed, using 35 lexical features to define the readability of English texts for L2 learners (Lee & Lee, 2020).

In our earlier work, we used Coh-Metrix and mathematical features to classify MWPs by difficulty level and performed a comparative analysis (Kadam et al., 2023). In this study, we compute seven standard readability scores-Flesch, Flesch-Kincaid, Automated Readability Index, Gunning Fog, Coleman-Liau Index, Dale-Chall Index, Linsear Write, and SMOG-to investigate their association with MATH difficulty levels (Graesser et al., 2004). Alongside these, we derive twelve additional structural features: word count, equation count, presence of tabular presentation, presence of images or diagrams, count of mathematical symbols, count of mathematical vocabulary terms, average words per sentence, average sentence length, count of decimal numbers, total number of numeric tokens, and counts of single-, double-, and triple-digit numbers. Approximately 10% of the MWPs contain more than 50 words. To limit extreme text lengths, based on the word-count distribution we retain only those MWPS that have utmost 50 words. We again use regular expressions to detect expressions and equations. The feature *Tabular Question* is coded as 'Yes' if the problem statement includes a table and 'No' otherwise, while *Image* indicates presence or absence of diagrams, plots, or other graphical content. Symbol counts and math vocabulary counts are derived from the lexicon described above. Although the algebra subset is predominantly text-only, a small fraction of MWPs include tables (e.g., value tables) or figures/plots. We include Tabular Question and Image as binary indicators to capture these cases when present and to keep the feature taxonomy extensible to other MWP corpora where tables/diagrams are more common. These indicators are lightweight, interpretable signals that capture format-driven cognitive load (information extraction + representation), which is a key contributor to perceived and empirical difficulty in MWPs.

Word Embeddings: To apply classifiers to text data, we require numeric representations of words, sentences, or entire problem statements. While linguistic and mathematical features capture specific, interpretable patterns, word embeddings offer dense, high-dimensional representations that encode semantic relationships learned from large text corpora. Traditional feature extraction methods, such as Bag-of-Words (BoW) and TF-IDF, treat words largely as independent tokens. In contrast, modern word embedding methods leverage neural architectures to capture semantic similarity and contextual usage patterns.

In this work, we use TF-IDF, Word2Vec, fastText, GloVe, and BERT-based embeddings to generate vector representations of MWPs. Prior studies have used BERT and related models to study difficulty prediction and to model relationships between question embeddings, concept embeddings, and student proficiency embeddings, but these have not explicitly focused on difficulty classification of MWPs (Cheng et al., 2019; Zhou & Tao, 2020). We use 300-dimensional embeddings for Word2Vec, fastText, and GloVe. Each embedding model captures different aspects of the text. Word2Vec treats each token as an atomic unit and learns embeddings via local context windows, while fastText represents words as compositions of character -grams, which is advantageous for handling rare or morphologically complex words. GloVe embeddings are trained on global word co-occurrence statistics, providing non-contextual but semantically rich representations. Word2Vec is pre-trained on the Google News corpus, and fastText uses one-million-word vectors trained on Wikipedia 2017, UMBC webbase, and the statmt.org news dataset (16B tokens). We use 300-dimensional GloVe embeddings trained on Wikipedia 2014 (glove.6B.zip). For BERT, we use 768-dimensional contextual embeddings trained on the Google Books corpus. For instance, minimal pre-processing is typically performed in case of BERT, beginning with basic Unicode and whitespace normalization, Convert common patterns to consistent text tokens: \frac{a}{b} to a / b or fraction a b; \sqrt{x} to sqrt x; \times to times or *, Keep comparison tokens: <=, >=, etc.; leave numbers as is, no stop-word removal as BERT uses them for context, use model tokenizer (bert-base-uncased), etc. In case of GloVe, we additionally, standardize, km, kilometers to kilometer; Rs./₹ to INR; aggregate word vectors to get a single problem vector: mean pooling, etc. Each MWP is thus mapped to a continuous vector space that encapsulates its semantic content, enabling downstream classification.

3.4 Feature Analysis

In the previous subsections, we described how math, linguistic, and embedding-based features were derived. We now analyze how these features relate to difficulty levels using model-based approaches.

SHAP: SHapley Additive exPlanations (SHAP) quantify the contribution of individual features to the predictions of a given model, where the model is Random Forest. Here, we employ SHAP to understand which features drive the classification decisions and to guide feature selection. Positive SHAP values indicate a positive contribution of a feature to predicting a given class (difficulty level), whereas negative values indicate a negative contribution. The magnitude of the SHAP value reflects the strength of this contribution, allowing us to rank features by importance in a transparent, model-agnostic manner. Features such as word count, Linsear Write formula, math vocabulary count, Automated Readability Index, and equation count exhibit strong positive contributions to predicting level~5 items as presented in **Figure 2**. In contrast, low values of *Double digit count* are associated with higher difficulty in this class. Similar class-specific patterns for all difficulty levels are presented in **Figure A2** of the Appendix. A global summary of feature importance across difficulty levels is shown in **Figure A1**, where SHAP values are plotted on the x-axis and features on the y-axis, with color indicating the difficulty level. We observe that several linguistic features have greater impact on difficulty classification than purely mathematical features. Note that, due to zero-based indexing in Python, difficulty levels are labeled 0-4 in these plots.

SEM: Structural Equation Modeling (SEM) allows us to model relationships between observed features and latent constructs. Based on the literature, we define two latent variables: *NmF* (numeric factors) and *EAF* (English factors), and examine their effects on item difficulty. We estimated the proposed two-factor measurement model and the corresponding structural model (Difficulty_level regressed on NmF and EAF) on the full sample (N = 3,938). Standard SEM fit indices indicate that the current measurement specification provides inadequate global fit: for the CFA, CFI = 0.60, TLI = 0.49, RMSEA = 0.21 ($\chi^2(43) = 7565.38$, $p < .001$); for the full SEM, CFI = 0.60, TLI = 0.49, RMSEA = 0.19 ($\chi^2(52) = 7831.01$, $p < .001$). These values fall below commonly accepted thresholds (e.g., CFI/TLI ≥ 0.90 and RMSEA ≤ 0.08), indicating that

the hypothesized indicator-to-construct mapping does not fully capture the observed covariance structure. Despite the weak global fit as shown in **Table A4** of Appendix, the estimated structural paths suggest that both constructs are positively associated with difficulty (standardized effect directions consistent across models), with Linguistic showing a larger association than Math in our current specification (Difficulty_level ~ Linguistic: $\beta = 0.51$, $p < .001$; Difficulty_level ~ Math: $\beta = 0.27$, $p < .001$). We also report the indicator–construct relationships (loadings) for transparency; in our implementation these appear in the parameter table as regressions of indicators on latent variables (indicator ~ latent), which is mathematically equivalent to the conventional lavaan notation (latent =~ indicator). As a follow-up improvement, we plan to refine the measurement model by reducing redundancy among highly correlated readability indices, evaluating alternative indicator sets (and/or allowing theoretically justified cross-loadings), and re-estimating fit using an estimator better suited to non-normal and binary indicators (e.g., categorical SEM / WLSMV in future work). These results consistently support the conclusion that linguistic features play a more prominent role than purely mathematical features in predicting item difficulty. **Table 4** summarizes features that exert positive and negative impacts on each difficulty level, combining insights from SHAP and SEM analyses. In summary, SHAP provides fine-grained, model-based explanations of feature contributions, and SEM offers a complementary, latent-variable perspective on the relationships between feature groups and difficulty. Both analyses underscore the strong influence of linguistic factors relative to purely mathematical characteristics. In the next section, we describe the experimental setup and supervised models used to estimate difficulty levels.

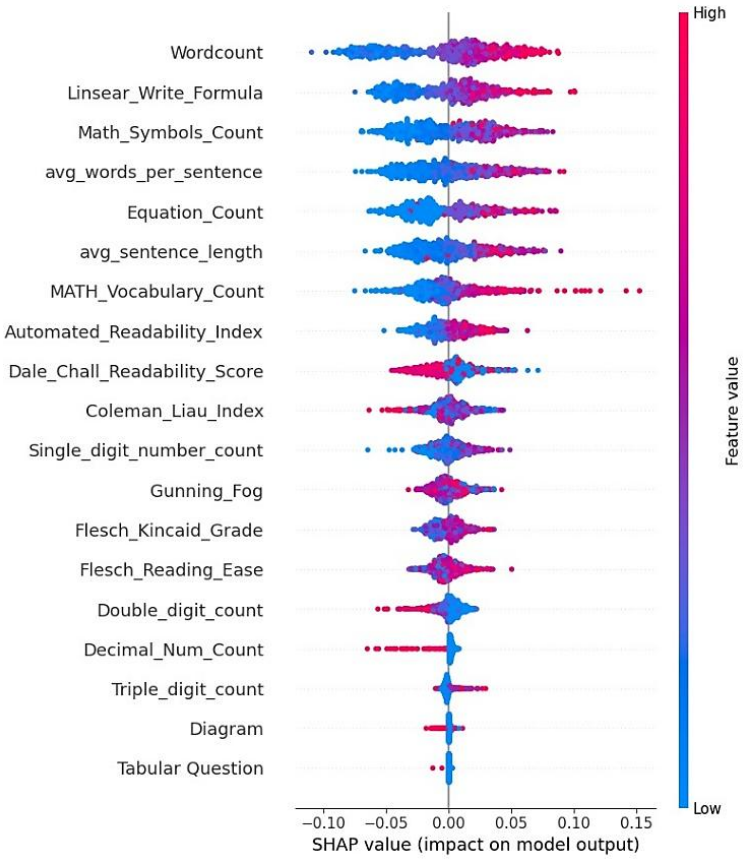

Figure 2. SHAP values for difficulty level 5.

Table 4. Features that positively or negatively influence difficulty level.

Difficulty level	Positive impact	Negative impact
1	MATH Vocabulary Count	Dale–Chall Readability Score, Flesch Reading Ease, Decimal number count
2	Triple digit count, MATH Vocabulary Count	Dale–Chall Readability Score, Decimal number count, Diagram
3	Decimal number count, Dale–Chall Readability Score, Diagram	Word count, Flesch Reading Ease
4	Gunning Fog Index	Dale–Chall Readability Score, Decimal number count, Diagram
5	Word count, Linsear Write Formula, MATH Vocabulary Count, Math symbols, Equation count, Automated Readability Index, Flesch Reading Ease, Triple digit count, etc	Decimal number count, Dale–Chall Readability Score, Coleman–Liau Index, Diagram, etc

4. Experimental Setup and Supervised Modeling

Building on the features and analyses described above, we now outline the supervised modeling setup used to predict MWP difficulty. Our goal is to compare traditional feature-based approaches with embedding-based representations across a diverse set of classifiers, and to identify models that offer robust performance for multi-class difficulty prediction. We first construct a comprehensive feature set for each MWP, combining linguistic, syntactic, structural, and mathematical attributes, along with multiple types of word embeddings. To accommodate the heterogeneity of these feature types and the high dimensionality of embeddings, we experiment with several families of classifiers, including both individual and ensemble methods.

Train-Validation Split: After applying all pre-processing steps, the dataset of 4,244 MWPs is reduced to 3,238 usable items (after removing MWPs having more than 50 words and missing difficulty levels). To reduce dependence on a single random split and to address small-sample testing concerns, we treat the cleaned training partition as the primary evaluation bed and report stratified k-fold cross-validation results (k = 5) in addition to 70:30 train-test split. After applying all pre-processing and filtering steps Section 3, the training partition is reduced to N = 3,238 usable items. We perform stratified 5-fold cross-validation so that each fold preserves the original class distribution. We construct a small external sanity-check set by sampling 10 items from each difficulty level from the official MATH test partition (50 MWPs total). Given the limited size of this external set, we treat it as auxiliary evidence (out-of-distribution check) rather than the primary basis for performance claims.

Classifiers: Data-driven decision-making in educational assessment typically benefits from exploring multiple modeling approaches rather than relying on a single algorithm. To this end, we implement a set of standard classifiers and ensemble methods to evaluate prediction performance: Decision Trees (DT), Gaussian Naïve Bayes (GNB), Support Vector Classifier (SVC), Adaboost (ADA), Gradient Boosting Machines (GBM), Nearest Neighbours (*k*-NN), XGBoost (XGB), and Random Forests (RF). Some classifiers (e.g., DT and feedforward neural networks) are known to be prone to overfitting, particularly when sample sizes are modest and feature spaces are large. Instance-based methods such as *k*-NN can be sensitive to outliers and require more computational effort as the number of features grows. SVCs may underperform in the presence of substantial noise, and GNB can struggle when feature distributions deviate strongly from its independence assumptions. Given that the class distribution in **Figure 4** is moderately imbalanced, we use 'balanced' class weights where applicable to reduce bias toward more frequent difficulty levels. Our objective is to identify one or more classifiers that offer consistently strong macro-level performance across the 5-class difficulty prediction task. Additionally, we reformulated difficulty prediction as an ordinal regression problem to respect the ordinal nature of 1-5 levels.

Feature Selection and Hyperparameter Tuning: After the initial round of modeling, we use feature importance scores from the RF classifier to refine the feature set. The least influential features (those with minimal impact on prediction performance) are dropped, as illustrated in **Figure 3**, which shows the RF feature importance profile. This step helps to reduce dimensionality and mitigate overfitting, while maintaining predictive power.

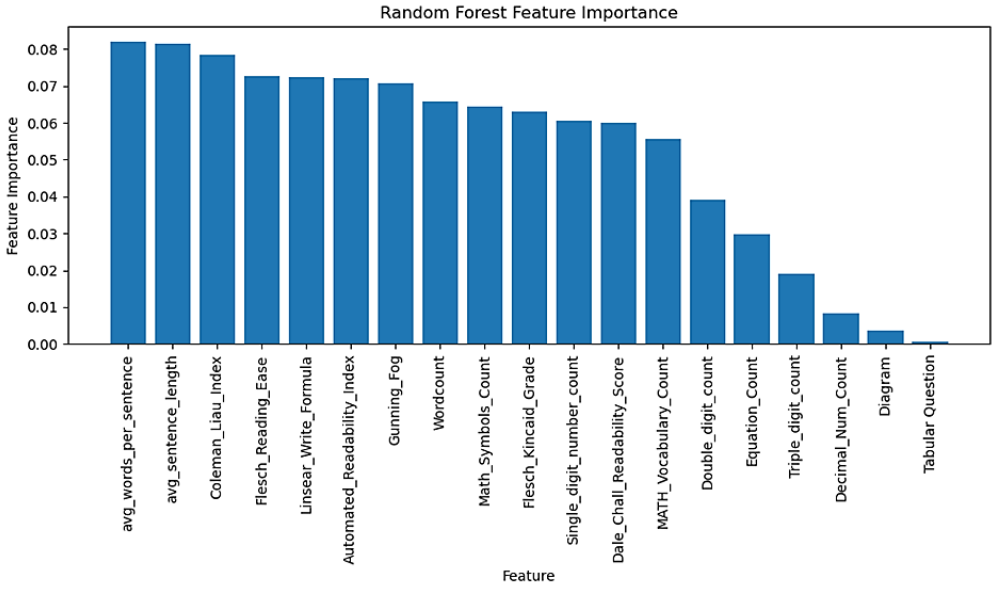

Figure 3. Feature importance plot from random forest classifier.

To further improve performance, we conducted hyperparameter tuning using 5-fold cross-validation and GridSearchCV in Python. For the RF model, we search over a grid of values for maximum tree depth, minimum samples per leaf, and number of trees, specified as: max_depth: [None, 3, 5, 7], min_samples_leaf: [1, 3, 5], and n_estimators: [50, 100, 150], min_samples_split: [2, 5, 10], max_features: ["sqrt", "log2", None], class_weight: [None, "balanced"]. The best configuration identified by the grid search is RandomForestClassifier(max_depth=3, n_estimators=50, random_state=100). The macro F1-score obtained from the this model was 0.26 which is lower than 0.4. We report and compare the performance of all models under different feature configurations as shown in **Table A2** of Appendix. Additionally, we highlight the relative strengths of feature-based and embedding-based approaches.

5. Performance Evaluation

We evaluate the proposed models using standard multi-class classification metrics derived from the confusion matrix: accuracy, precision (P), recall (R), and F1-score (1). Although accuracy is commonly reported, it can be misleading under class imbalance, where performance may be dominated by frequent classes. For this reason, we emphasize the F1-score, which balances precision and recall and is more informative when class distributions are unequal.

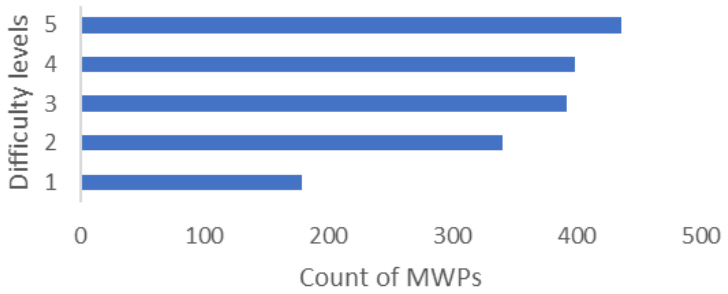

Figure 4. shows the distribution of algebra MWPs across five difficulty levels.

$$\boldsymbol{F1\ Score} = \frac{\boldsymbol{2*(P*R)}}{\boldsymbol{P+R}} \quad (1)$$

$$Macro\ F1-score = \frac{\sum_1^n F1_i}{n} \quad (2)$$

In this paper, we report *Macro F1-score* (2) as the primary metric because it assigns equal weight to each difficulty level, thereby reflecting performance across all classes rather than favoring majority classes. For a 5-class classification task, a random-guess baseline corresponds to a Macro F1-score of approximately 0.20; therefore, results meaningfully above this threshold indicate performance better than chance.

Evaluation Summary: Achieving strong Macro F1-scores in a multi-class setting indicates that a model discriminates effectively across difficulty levels while maintaining a reasonable balance between precision and recall. Using handcrafted linguistic and mathematical features, the best performance on the held-out 50-MWP test set reaches a Macro F1-score of 0.23 **Table 5** with Decision Trees showing clear signs of overfitting (near-perfect training performance but poor generalization). After feature reduction and re-training, performance improves modestly: SVC reaches 0.29 and GBM reaches 0.26 on the test set **Table 6.** However, further improvements through GridSearchCV were limited when using only handcrafted features. Additionally, we trained an ordinal logistic regression model (LogisticIT; mord) using 5-fold stratified cross-validation with inner-fold tuning of the regularization parameter (alpha ∈ {0.01, 0.1, 1, 10, 50}) to avoid selection bias. On aggregated out-of-fold predictions (N = 3,238), the model achieved macro-F1 score of 0.243, MAE = 1.08 and quadratic weighted kappa (QWK) = 0.37, with within-one-level accuracy ($|\hat{y}-y| \leq 1$) of 0.71. These ordinal-aware metrics complement multiclass accuracy/F1 by penalizing larger mistakes more heavily and better reflect the ordered nature of the target.

Table 5. F1-scores from linguistic and mathematical features with 70:30 split and 5-fold cross validation on train data.

F1-scores based on 70:30 split					
Data	**DT**	**GNB**	**SVC**	**kNN**	**RF**
Train	0.53	0.19	0.49	0.54	0.95
Validation	0.30	0.18	0.31	0.32	0.34
50 MWPs	**0.23**	0.15	0.19	0.2	0.21
GridSearchCV with 5-folds					
Data	**DT**	**GNB**	**SVC**	**kNN**	**RF**
Mean CV	0.97	0.22	0.49	0.44	0.88
50 MWPs	**0.28**	0.19	0.27	0.19	0.24

Table 6. F1-scores after feature reduction.

Data	RF	ADA	SVC	GBM	XGB
Train	0.99	0.33	0.35	0.59	0.99
Validation	0.36	0.30	0.29	0.30	0.34
50 MWPs	0.24	0.25	**0.29**	**0.26**	0.24

Table 7. F1-scores from word-embedding features.

Embeddings	Data	DT	RF	ADA	GBM	SVC	k-NN	GNB
Word2Vec	Train	0.357	1.000	0.333	0.859	0.282	0.507	0.225
	Validation	0.278	0.345	0.290	0.342	0.267	0.285	0.214
	50 MWPs	0.297	0.728	0.199	0.703	0.159	**0.379**	0.196
GLoVe	Train	0.390	0.971	0.338	0.853	0.400	0.488	0.320
	Validation	0.274	0.310	0.282	0.327	0.315	0.322	0.267
	50 MWPs	0.239	0.781	0.329	0.611	0.285	**0.400**	**0.392**
FastText	Train	0.356	1.000	0.337	0.872	0.259	0.501	0.227
	Validation	0.286	0.334	0.272	0.305	0.249	0.292	0.217
	50 MWPs	0.271	0.818	0.270	0.677	0.186	0.339	0.166
BERT	Train	0.403	1.000	0.346	0.855	0.571	0.490	0.214
	Validation	0.294	0.342	0.287	0.332	0.358	0.321	0.238
	50 MWPs	0.120	0.239	0.159	0.206	0.118	0.124	0.191

Error analysis indicates that most mistakes occur between neighbouring levels. Adjacent-level errors ($|\hat{y}-y| = 1$) account for 0.39 of cases, while larger deviations ($|\hat{y}-y| \geq 2$) occur in 0.29 of cases. The confusion matrix shows substantial overlap among mid-to-high difficulty categories (especially levels 3–5), suggesting that boundaries between adjacent levels are not sharply separable using the current feature set alone. Overall, these findings support the interpretation that the model frequently predicts a nearby difficulty level even when exact matching is challenging, consistent with an ordinal and potentially subjective labeling scheme.

In contrast, when using word embeddings as representations of MWPs, several models show substantial training-set overfitting (particularly RF and GBM), but certain classifier–embedding combinations generalize better. Notably, the *k*-NN classifier achieves the strongest generalizable performance, reaching a Macro F1-score of 0.40 with GLoVe embeddings which is larger when compared to most other models. We further conducted sensitivity analysis by implementing GridSearch{'n_neighbors' = {3.5.7.15 'metric'= {cosine, Euclidean 'weights': {uniform, distance_weighted}} and found the best param model {'metric': 'cosine', 'n_neighbors': 15, 'weights': 'distance'} as updated in the **Table 7**. These results suggest that embedding-based representations capture semantic and linguistic characteristics relevant to difficulty more effectively than handcrafted features alone.

6. Results and Analysis

The Macro F1-score is used as the primary metric for interpreting model performance and comparing results across feature settings. In addition to aggregated performance, per-class precision and recall offer insight into which difficulty levels are easier or harder for the model to identify. **Tables 8**, **9** and **10** present detailed classification reports for the *k*-NN classifier (with the best-performing embedding configuration) on the training, validation sets and, test sets, respectively.

A key observation from the per-class results is that model performance varies across difficulty levels. On the 50-item test set, the model achieves an overall accuracy of 0.76 (macro F1 = 0.756; weighted F1 = 0.762). Performance is strongest for difficulty levels 1 and 5, both of which show high F1-scores (0.857

each) with correspondingly high recall (0.900 for level 1; 0.818 for level 5). In contrast, mid-range difficulty levels exhibit weaker and less consistent behavior: level 3 attains high precision (0.857) but reduced recall (0.600), indicating that many true level-3 problems are being confused with neighboring categories; level 4 is the most challenging class (precision = 0.545; recall = 0.667; F1 = 0.600), suggesting greater overlap with adjacent difficulty levels and/or fewer distinctive cues for that category. Overall, these patterns support the interpretation that difficulty is only partially separable from text alone, with most confusion likely occurring between adjacent levels—an expected outcome when labels are ordinal and may reflect subjective or context-dependent judgments. This performance gap aligns with our interpretability findings (SHAP and SEM), which suggest that linguistic properties exert a substantial influence on difficulty estimation. At the same time, handcrafted features remain valuable because they provide interpretability and educational insight, whereas embeddings act largely as black-box representations.

Table 8. Classification report for train data from (*k*-NN) classifier.

Class	Precision	Recall	F1-score	Support
1	0.4074	0.4802	0.4408	252
2	0.4472	0.5469	0.4920	565
3	0.4753	0.3875	0.4269	622
4	0.5215	0.4910	0.5058	668
5	0.5823	0.5670	0.5746	649
Accuracy			**0.4960**	**2756**
Macro Avg	0.4867	0.4945	**0.4880**	2756
Weighted Avg	0.4997	0.4960	0.4954	2756

Table 9. Classification report for validation data from (*k*-NN) classifier.

Class	Precision	Recall	F1-score	Support
1	0.2015	0.2700	0.2308	100
2	0.3234	0.3655	0.3432	238
3	0.2972	0.2671	0.2814	277
4	0.2709	0.2720	0.2715	250
5	0.4373	0.3849	0.4094	317
Accuracy			**0.3198**	**1182**
Macro Avg	0.3061	0.3119	**0.3072**	1182
Weighted Avg	0.3264	0.3198	0.3218	1182

Table 10. Classification report for test data from (*k*-NN) classifier.

Class	Precision	Recall	F1-score	Support
1	0.8182	0.9	0.8571	10
2	0.7273	0.8	0.7619	10
3	0.8571	0.6	0.7059	10
4	0.5455	0.6667	0.6	9
5	0.9	0.8182	0.8571	11
Accuracy			0.76	50
Macro Avg	0.7696	0.757	**0.7564**	50
Weighted Avg	0.7767	0.76	0.7616	50

While other explainability techniques such as LIME, permutation importance, partial dependence plots, or attention-based analyses could also be explored, many of these methods either provide purely local explanations, assume feature independence, or are tightly coupled to specific model architectures. In contrast, SHAP and SEM were chosen for their stability, model-agnostic nature, and ability to support both predictive and theory-driven interpretation. Future work may systematically compare these methods, particularly in conjunction with neural and transformer-based models. These findings motivate future work

exploring more advanced language models for difficulty estimation while balancing accuracy with explicability.

7. Performance Comparison

To validate the proposed ML-based difficulty estimation, we compare classifier outputs against two commonly used traditional approaches: (i) expert judgment (pre-test) and (ii) Item Response Theory (IRT) analysis using learner responses (post-test). This comparison is important because difficulty labels in educational assessment are often influenced by subjectivity (experts) and by empirical learner performance (IRT), and these perspectives do not always align. In what follows, we describe the pre-test and post-test settings and then evaluate the fit and implications of standard IRT models (Baker, 2001; Wauters et al., 2012).

7.1 The Pre-test and Post-test Models

Pre-test (expert judgment): To obtain expert ratings of item difficulty, we administered a structured Google Form containing 50 MWPs and collected independent difficulty-level assignments from seven instructor who have over 15 years of experience teaching high school mathematics. While expert consensus is typically considered a reasonable proxy for item difficulty, the ratings in our sample exhibited substantial variability **Table 12**. To quantify inter-rater agreement, we computed Fleiss' Kappa, which is widely used to measure reliability among multiple raters. The resulting value was below 0.2, indicating low agreement and underscoring the inherent subjectivity of manual difficulty annotation in MWPs. This finding aligns with earlier observations that perceived difficulty can vary across raters depending on interpretation of language, assumed prerequisite knowledge, and expectations about solution strategies.

Post-test (IRT-based difficulty): For the IRT analysis, we created a test paper consisting of the same 50 items and administered it to 61 high-school students, collecting their item-level responses. Since problems in the MATH dataset are sourced from Olympiad-style exams intended for students from Grade 8 to 10, we selected the student cohort accordingly. The goal of this post-test evaluation is to estimate item difficulty from observed learner performance and to compare IRT-based difficulty parameters with the predictions produced by ML classifiers.

IRT provides a probabilistic framework for modeling the relationship between learner ability and item characteristics, including difficulty and discrimination. Consider a test with items taken by students. Suppose a multiple-choice test consisting of k items is taken by n subjects (students/learners), let Yij denote the correct or incorrect response of ith student on jth item $y = (y_{11}, y_{12}, \dots, y_{nk})$. $\{Y_{ij}\}$ could be assumed as a Bernoulli random variable where $\{p_{ij}\}$ = 0 or 1 and probability of success $p_{ij} = Prob(Y_{ij} = 1)$. Another well-known model is item response theory where the two-parameter probit model is defined as: $p_{ij} = \Phi(a_j, \theta_i - b_j)$ where Φ is a standard normal cdf. The parameters are also called as latent traits that are unknown and must be estimated based on the responses in a test. The students' ability to perform well in the test is represented as θ_i, a_j, measures the ability of item to discriminate between good and bad students and $b_j$measures the difficulty of a particular item. The parameter c is the probability of getting the item correct by guessing alone. The IRT models are as follows by Equation (3), where:

$$p_{ii} = \varphi(a_j(\theta_i - b_j)) \quad (3)$$

where, b is the difficulty parameter a is the discrimination parameter c is the guessing parameter and θ is the ability level. The three-parameter variant further includes a guessing parameter capturing the probability of answering correctly by chance. In this study, we estimate item difficulty using one- (4), two- (5), and three-parameter IRT (6) models and compare the resulting difficulty estimates with ML predictions.

The IRT models considered are:

$$P(\theta) = \frac{1}{1+ e^{-(\theta-b)}} \ (one - parameter/Rasch\ model) \quad (4)$$

$$P(\theta) = \frac{1}{1+ e^{-\alpha(\theta-b)}} \ (two - parameter) \quad (5)$$

$$P(\theta) = c + (1-c) * \frac{1}{1+ e^{-(\theta-b)}} \ (three - parameter/Rasch\ model) \quad (6)$$

$a: \rightarrow discrimation\ parameter; b: \rightarrow difficulty\ parameter; c: \rightarrow guessing\ parameter,$
$\theta \rightarrow abilit\ level.$

Given student-item interaction data, latent ability parameters and item parameters can be estimated by maximizing the likelihood of the observed response matrix. While IRT supports a wide range of diagnostic questions (e.g., item redundancy, information coverage across ability levels, and fairness), our primary interest in this paper is to extract and interpret item difficulty estimates and compare them to the outputs of ML classifiers. We next describe model fitting and selection for IRT.

7.2 Model Fit

The one-parameter IRT model (Rasch model) assumes that items differ only in difficulty, and that all items discriminate equally across learners. The two-parameter model relaxes this assumption by allowing items to vary in both difficulty and discrimination. The three-parameter model further incorporates guessing effects, which can be relevant for tests where students may answer correctly by chance. Although our items are graded as correct/incorrect (dichotomous), we estimate all three models to evaluate which best captures the observed response patterns.

We used the ltm library in R to fit these models and conduct statistical comparisons (Pimentel & Villaruz, 2020). To compare model fit, we apply ANOVA to test whether the improvement in fit from one model to another is statistically significant. First, we compare the Rasch model and the 2-parameter model. The null hypothesis assumes no significant difference between the two models. The ANOVA output yields **Table 11**, leading us to reject the null hypothesis and conclude that the 2-parameter model fits significantly better than the Rasch model. Next, we compare the 2-parameter and 3-parameter models. Here, the ANOVA output yields, and we fail to reject the null hypothesis, suggesting that the 3-parameter model does not provide a statistically meaningful improvement over the 2-parameter model for this dataset.

Consistent with the above results, the AIC values also favor the 2-parameter model over both the 1-parameter and 3-parameter alternatives. Therefore, for subsequent interpretation of item difficulty, we focus on the estimated item parameters from the 2-parameter model. The corresponding item coefficients are shown in **Table A1** of the Appendix. In addition to numeric fit indices, the ltm package produces diagnostic visualizations such as item characteristic curves (trace plots), which provide intuitive insights into item behavior. **Figure A3** presents trace plots for the fitted 2-parameter model. Difficult items typically exhibit curves shifted to the right, indicating that higher ability levels are required to achieve a high probability of a correct response, whereas easier items shift left. The plots also reveal that some items provide more information at higher ability ranges, while others span a broader range. For example, Item 1 shows high discrimination, while Item 2 shows relatively low discrimination. We summarize key observations based on the parameters presented in **Table A1** in the Appendix.

Observations: Based on the estimated item parameters from the 2-parameter IRT model **Table A1** of Appendix, we make the following observations:

- For a subset of items (MQ2, MQ8, MQ20, MQ21, MQ49, MQ50), both the difficulty parameter (b) and

discrimination parameter (a) are negative. This implies that the probability of answering these items correctly decreases as respondent ability increases an atypical pattern in IRT. Such behavior may arise due to item ambiguity, misalignment between item wording and expected reasoning strategies, or noise in student responses. These items warrant further qualitative review before being used in calibrated assessments.

- Items MQ1, MQ4, and MQ7 exhibit high discrimination (a) but negative difficulty values (b). This combination suggests that although these items effectively distinguish between higher- and lower-ability learners, they are relatively easy overall. Such patterns may indicate well-constructed items that sharply separate learners near the lower end of the ability spectrum, or alternatively, inconsistencies between perceived and empirical difficulty.
- Thirty-seven items have negative difficulty parameters (b), indicating a range of easier difficulty levels (very easy to moderate). Using the estimated values, thresholds could be defined to map continuous IRT difficulty estimates onto discrete difficulty categories.
- Items MQ23, MQ47, and MQ48 show relatively large positive difficulty parameters, indicating higher difficulty, coupled with moderate discrimination. These items align more closely with classical expectations of challenging MWPs.
- Using the cross-tabulation between instructor judgments and MATH dataset labels **Table 12**, we computed class-wise F1-scores of 44%, 26%, 30%, 53%, and 18% for difficulty levels 1 through 5, respectively. When compared to these values, the Macro F1-scores obtained from the *k*-NN classifier **Table 11** exceed expert judgment performance for difficulty levels 1, 2, 3, and 5, indicating that the ML-based approach offers more consistent labeling than subjective expert ratings.
- Further, we examine the columns *Math Dataset Label* and *2P-P(x=1|z=0)* from **Table A1**. After sorting by dataset label, we apply a threshold of 75% to the predicted probability values, encoding items as 1 if the threshold is exceeded and 0 otherwise. Under this heuristic, we obtain agreement rates of 70%, 80%, 70%, 70%, and 20% for difficulty levels 1 through 5, respectively. While these values are encouraging, especially for lower and mid-range difficulty levels, further validation is required to establish robust thresholding strategies.

Table 11. ANOVA results.

ANOVA (IRTRasch vs. IRT2P)						
Model	**AIC**	**BIC**	**Log.Lik**	**LRT**	**df**	**P-value**
IRTRasch	1861.69	1949.08	-879.84			
IRT2P	1738.13	1909.48	-769.06	221.56	49	< 0.001
ANOVA (IRT2P vs. IRT3P)						
IRTRasch	1861.69	1949.08	-879.84			
IRT2P	1787.94	2044.98	-743.97	50.18	50	0.466

Table 12. Cross-tabulation of instructor judgment versus MATH dataset labels.

Instructor label	**1**	**2**	**3**	**4**	**5**
1	5	5	5	1	1
2	2	3	0	2	5
3	2	1	3	1	3
4	1	1	2	5	1
5	0	0	0	0	1

In addition, item information curves **Figure A4** illustrate the precision with which each item estimates ability (θ). Items with steeper and higher information curves are more effective at discriminating between learners with different ability levels and are therefore preferable in test construction. **Table A3** of Appendix,

shows overall model fit indices from the IRT calibration: M2 (limited-information goodness-of-fit) with degrees of freedom (df) and p-value (p), RMSEA with its 90% confidence interval (RMSEA_5, RMSEA_95), and additional approximate fit indices (SRMSR, TLI, CFI). Lower M2, RMSEA, and SRMSR, and higher TLI/CFI, indicate better fit. The results show a progressive improvement in global fit from Rasch/1PL → 2PL → 3PL, suggesting that allowing item discrimination (2PL) and guessing (3PL) better captures response patterns in the cohort. To analyze whether the cohort ability distribution supports stable estimation, in case of 3PL model, Sample size (persons): N = 61, mean ≈ -0.041, SD ≈ 1.046, Range: min = -2.43, max = 2.16, Concentration: 10th percentile ≈-1.33, median≈-0.048, 90th percentile ≈1.22. So about 80% of the cohort lies in roughly [-1.33, 1.22]. The distribution is roughly centered near 0 with a reasonable spread (~1 SD), and most respondents are not piled at one single ability value. That generally supports stable estimation around the region where most abilities lie. however, tails are sparse ($|\theta| \geq 2$: 5/61), so precision at extremes is limited.

The key limitation of IRT-based approaches is that estimated item difficulty depends on the specific cohort used for calibration. As a result, difficulty estimates may not generalize well to new student populations. Overall, the comparison across MATH dataset labels, instructor judgments, and IRT-based estimates reveals substantial discrepancies in difficulty labeling. These findings reinforce the need for scalable, data-driven approaches. The proposed IDEA framework addresses this challenge by providing consistent difficulty estimates directly from item content, while also mitigating the cold-start problem inherent in traditional psychometric methods.

8. Conclusions and Future Research

Most prior research on Math Word Problems (MWPs) has focused on identifying factors that make problems difficult for learners or on developing instructional interfaces that assist problem solving. Such work often implicitly assumes that reliable difficulty labels are already available, typically through expert annotation or learner-response data. The post-test was administered at a 40-year-old English-medium school affiliated with the Central Board of Secondary Education (CBSE), India. We note that perceived difficulty can depend on educational context (e.g., medium of instruction, curriculum alignment, and prior exposure to word problems); however, we did not collect detailed student- or school-level covariates (e.g., prior achievement, socioeconomic background, or access to coaching) to quantify these effects. Therefore, the post-test findings should be interpreted as reflecting difficulty judgments within this specific institutional context, and broader generalization across settings is left for future work. This study addresses a foundational and under-explored question: *can we estimate MWP difficulty directly from the problem statement itself*, especially in cold-start situations where neither expert consensus nor learner data is available? To address this, we proposed the **IDEA** framework, which combines linguistic features, mathematical vocabulary analysis, semantic word embeddings, and supervised machine learning to estimate difficulty levels of MWPs. Across our experiments, semantic embeddings captured difficulty-related regularities more reliably than handcrafted linguistic or mathematical features on their own. Among the evaluated classifiers, *k*-NN and SVC exhibited the most consistent generalization performance, particularly when combined with GloVe and Word2Vec embeddings. Given the five-class formulation of the task and mild class imbalance, Macro F1-score was adopted as the primary evaluation metric to ensure balanced assessment across difficulty levels.

A further practical takeaway is that transformer embeddings such as BERT did not offer consistent gains over simpler embedding methods for this task. This suggests that, for MWPs, difficulty may be tied more to relatively stable semantic and linguistic patterns than to deeper contextual modelling. It also reflects an implementation concern: transformer tokenisation and pooling choices can fragment mathematical symbols and LaTeX-like expressions, weakening representations when the input contains formula-style text. Taken

together, these findings point to a realistic trade-off between accuracy, interpretability, and computational cost an important consideration for educational systems that must operate at scale or under resource constraints.

The MATH dataset provides difficulty annotations but does not include explicit grade-level mappings, which limits direct alignment with curriculum standards. For real-world educational use, grade-wise or curriculum-aligned difficulty labels are essential. Future research should therefore focus on collecting or curating datasets that associate MWPs with both difficulty levels and instructional stages. Another open question concerns the optimal granularity of difficulty labels-whether three-level (Easy, Medium, Hard), five-level, or finer categorizations are most effective for adaptive learning and assessment systems. As future work, we plan to (i) expand the related work section to include education literature on student experiences and common barriers in solving word problems (e.g., language demands, schema/strategy selection, and translation from text to mathematical representation), and (ii) conduct a mixed-methods extension that triangulates ML-based predictions with structured teacher judgments and student feedback (surveys and/or interviews). This would enable us to examine where automated difficulty estimates align with learners' perceptions, where they diverge, and which qualitative factors explain those gaps—thereby improving both the validity and the educational usefulness of automated difficulty modeling.

Beyond immediate predictive performance, this work has important implications for supervised fine-tuning of language models in education. While large language models have shown promise in solving and generating MWPs, supervised fine-tuning for difficulty-aware behavior fundamentally depends on the availability of reliable labeled data. As demonstrated by the low inter-rater agreement among instructors in this study, such labels are often subjective, inconsistent, and costly to obtain. The proposed IDEA framework directly addresses this bottleneck by enabling automated, data-driven difficulty estimation without relying on learner-response data or manual annotation. In this sense, difficulty estimation should be viewed as an enabling step rather than an end task-supporting dataset creation, adaptive assessment design, and future fine-tuning of domain-specific language models.

Finally, while this study focuses on algebra MWPs, the proposed framework is general and can be extended to other mathematical domains such as arithmetic, calculus, combinatorics, probability, and statistics. A major challenge remains the limited availability of labeled datasets across these domains. Nevertheless, this work demonstrates that automated, interpretable, and scalable difficulty estimation is feasible and provides foundational infrastructure for building difficulty-aware AI systems in mathematics education.

Appendix

Table A1. The item difficulty, item discrimination and the probability of correct response from IRT 2-parameter model.

Q ID	MATH Dataset Label	Instructor Label	% Correct Responses	Difficulty (Rasch)	2P Difficulty (b)	2P Discrimination (a)	2P P(x = 1 | z = 0)
Q8	1	1	31.1	0.708597	-49.4464	-0.01607	0.311216
Q20	2	1	13.1	1.757733	-6.00105	-0.32741	0.12295
Q2	1	1	13.1	1.757707	-4.08773	-0.49935	0.114945
Q50	5	2	47.5	-0.02537	-3.59935	-0.02791	0.474907
Q21	3	4	34.4	0.556066	-3.57927	-0.18576	0.339645
Q49	5	4	21.3	1.213968	-3.36855	-0.41559	0.197825
Q3	1	4, 2,1	80.3	-1.66843	-2.02733	0.856291	0.850176
Q1	1	1	90.2	-2.44975	-1.38963	31.65048	1
Q7	1	3	86.9	-2.14715	-1.3735	25.48454	1
Q4	1	1	85.2	-2.01439	-1.3624	30.22269	1

Table A1 continued…

Q17	2	2	77	-1.46509	-1.35701	1.400988	0.870022
Q25	3	1	68.9	-1.01515	-1.317	0.748263	0.728191
Q15	2	2	75.4	-1.36949	-1.22045	1.568263	0.871467
Q29	3	1	78.7	-1.56499	-1.21469	2.732053	0.965062
Q14	2	1	77	-1.46553	-1.1335	3.572072	0.982858
Q13	2	1	75.4	-1.36949	-1.13187	2.086011	0.913812
Q16	2	1	70.5	-1.1004	-1.11904	1.147603	0.783167
Q28	3	1	75.4	-1.36939	-1.08848	2.776288	0.953555
Q39	4	1	75.4	-1.36939	-1.07776	3.306315	0.97244
Q18	2	3	72.1	-1.18764	-1.0609	1.601404	0.845394
Q36	4	2	70.5	-1.10039	-0.99205	1.577346	0.827041
Q9	1	1	67.2	-0.93189	-0.97176	1.086843	0.741954
Q45	5	3	67.2	-0.93188	-0.9685	1.094655	0.742725
Q22	3	3	72.1	-1.18739	-0.96086	3.191412	0.955489
Q5	1	2	70.5	-1.10035	-0.95583	1.826807	0.851462
Q46	5	2	70.5	-1.10035	-0.92974	2.12862	0.87858
Q12	2	1	70.5	-1.10035	-0.91815	2.348834	0.896283
Q41	5	2	68.9	-1.01529	-0.86656	2.108055	0.861376
Q27	3	1	68.9	-1.01603	-0.84264	5.413855	0.989666
Q10	1	3	68.9	-1.01542	-0.84157	3.404476	0.946095
Q32	4	3	68.9	-1.01554	-0.84154	3.441202	0.947643
Q31	4	4	67.2	-0.93193	-0.83168	1.712732	0.806035
Q42	5	2	57.4	-0.46365	-0.82396	0.421423	0.585947
Q11	2	4	65.6	-0.85046	-0.80748	1.381114	0.7531
Q35	4	4	67.2	-0.93208	-0.78992	2.485549	0.876899
Q24	3	4	67.2	-0.9321	-0.78822	2.570152	0.883485
Q26	3	3	63.9	-0.77068	-0.6632	3.715904	0.921608
Q34	4	4	62.3	-0.69229	-0.62496	1.788034	0.753515
Q6	1	2,4	62.3	-0.69229	-0.60641	2.508096	0.820676
Q30	3	3	62.3	-0.69229	-0.60328	3.243053	0.876152
Q37	4	2, 3	60.7	-0.61512	-0.58029	1.451834	0.698989
Q44	5	3	57.4	-0.46364	-0.54655	0.77227	0.603982
Q43	5	1	57.4	-0.46364	-0.43467	1.561262	0.663434
Q40	4	3	57.4	-0.4637	-0.42397	2.714102	0.75964
Q19	2	2	54.1	-0.31534	-0.31686	0.985928	0.577471
Q33	4	4	50.8	-0.16942	-0.16931	1.847808	0.577583
Q38	4	4, 3, 2	50.8	-0.16954	-0.15052	1.161668	0.543602
Q47	5	5	45.9	0.046657	0.036107	1.690395	0.484746
Q23	3	1, 2, 4	23	1.122977	7.198432	0.167974	0.229852
Q48	5	2	18	1.411352	35.60461	0.042463	0.180662

Table A2. The hyperparameter search space used for each classifier and the final settings used in this study. When a parameter is not listed as tuned, the scikit-learn default value is used.

Model	**Hyperparameter search grid**	**Final setting used**
Random Forest	n_estimators: [200, 500, 800], max_depth: [None, 10, 20, 40], min_samples_split: [2, 5, 10], min_samples_leaf: [1, 2, clf__max_features: ["sqrt", "log2", None], class_weight: [None, "balanced"],	(max_depth=3, n_estimators=50, random_state=100).
Decision Trees	criterion: ["gini", "entropy", "log_loss"], max_depth: [None, 5, 10, 20, 40], min_samples_split: [2, 5, 10], min_samples_leaf: [1, 2, 4], class_weight: [None, "balanced"],	Best Params: {'clf__class_weight': 'balanced', 'clf__criterion': 'entropy', 'clf__max_depth': 10, 'clf__min_samples_leaf': 2, 'clf__min_samples_split': 10}
SVC	kernel: ["linear", "rbf", "poly"], C: [0.1, 0.5, 1.0, 2.0, 5.0, 10.0], gamma: ["scale", "auto"], class_weigh: [None, "balanced"],	Best Params: {'clf__C': 10.0, 'clf__class_weight': 'balanced', 'clf__gamma': 'scale', 'clf__kernel': 'rbf'}
GNB	var_smoothing": np.logspace(-12, -7, 6)	Best Params: {'clf__var_smoothing': 1e-07}
kNN	n_neighbors: [3, 5, 7], weights: ["uniform", "distance"], algorithm: ["auto", "kd_tree", "brute"], leaf_size: [20, 30], metric": ["minkowski", "cosine"], p: [1, 2]	Best Params: {'clf__algorithm': 'auto', 'clf__leaf_size': 20, 'clf__metric': 'cosine', 'clf__n_neighbors': 7, 'clf__p': 1, 'clf__weights': 'distance'}

Table A3. Global fit diagnostics for competing IRT models (Rasch/1PL, 2PL, 3PL).

IRT	M2	df	p	RMSEA	RMSEA_5	RMSEA_95	SRMSR	TLI	CFI
Rasch_stats	2048.993	1224	0	0.1059885	0.09711767	0.1130164	0.2082024	0.8864328	0.8865256
2PL_stats	1871.034	1175	0	0.09936208	0.09012058	0.1068217	0.1364149	0.9001895	0.904263
3PL_stats	1430.423	1125	1.36E-09	0.06726648	0.05565474	0.07695762	0.1137273	0.9542562	0.9579904

Table A4. Exploratory analysis of SEM global fit statistics that indicate inadequate fit.

Metrics	**Values**		
'CFI':	0.548		
'GFI':	0.546		
'AGFI':	0.483		
'NFI':	0.546		
'TLI':	0.485		
'RMSEA':	0.170		
'AIC':	71.159		
'BIC':	322.296		
'LogLik':	4.420		
lval	**rval**	**Estimate**	**Std**
Flesch_Reading_Ease	Linguistic	1	0.015415
Flesch_Kincaid_Grade	Linguistic	-58.048279	-0.895355
Gunning_Fog	Linguistic	-37.929459	-0.585417
Automated_Readability_Index	Linguistic	-57.121605	-0.880612
Coleman_Liau_Index	Linguistic	-34.625147	-0.533833
Dale_Chall_Readability_Score	Linguistic	-2.358149	-0.036358
Linsear_Write_Formula	Linguistic	-35.832239	-0.552333
Wordcount	Math	1	0.195627
Equation_Count	Math	3.170095	0.620063
Tabular_Question	Math	0.096744	0.018922
Diagram	Math	2.951762	0.577352
Math_Symbols_Count	Math	4.569025	0.89379
MATH_Vocabulary_Count	Math	0.917356	0.179416
avg_words_per_sentence	Math	4.172399	0.816166
Decimal_Num_Count	Math	2.871833	0.561712
Single_digit_number_count	Math	4.543034	0.888697
Double_digit_count	Math	2.332763	0.456295
Triple_digit_count	Math	0.847546	0.165768
lval	**rval**	**Estimate**	**Std**
Difficulty_level	Linguistic	-20.778388	-0.248463
Difficulty_level	Math	0.745286	0.113034

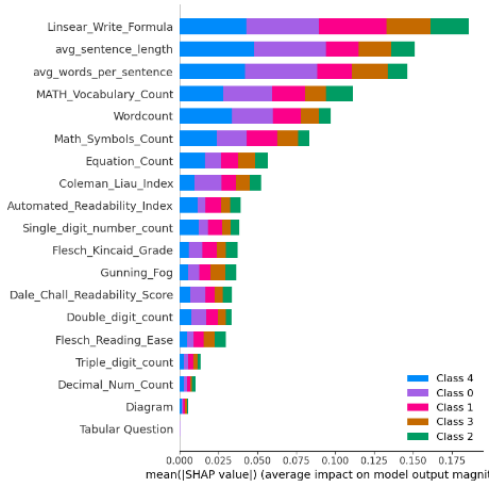

Figure A1. Class-wise SHAP feature importance for difficulty prediction. Mean absolute SHAP values (stacked by class) for the top input features, showing each feature's average contribution to the model's output magnitude across the five difficulty levels.

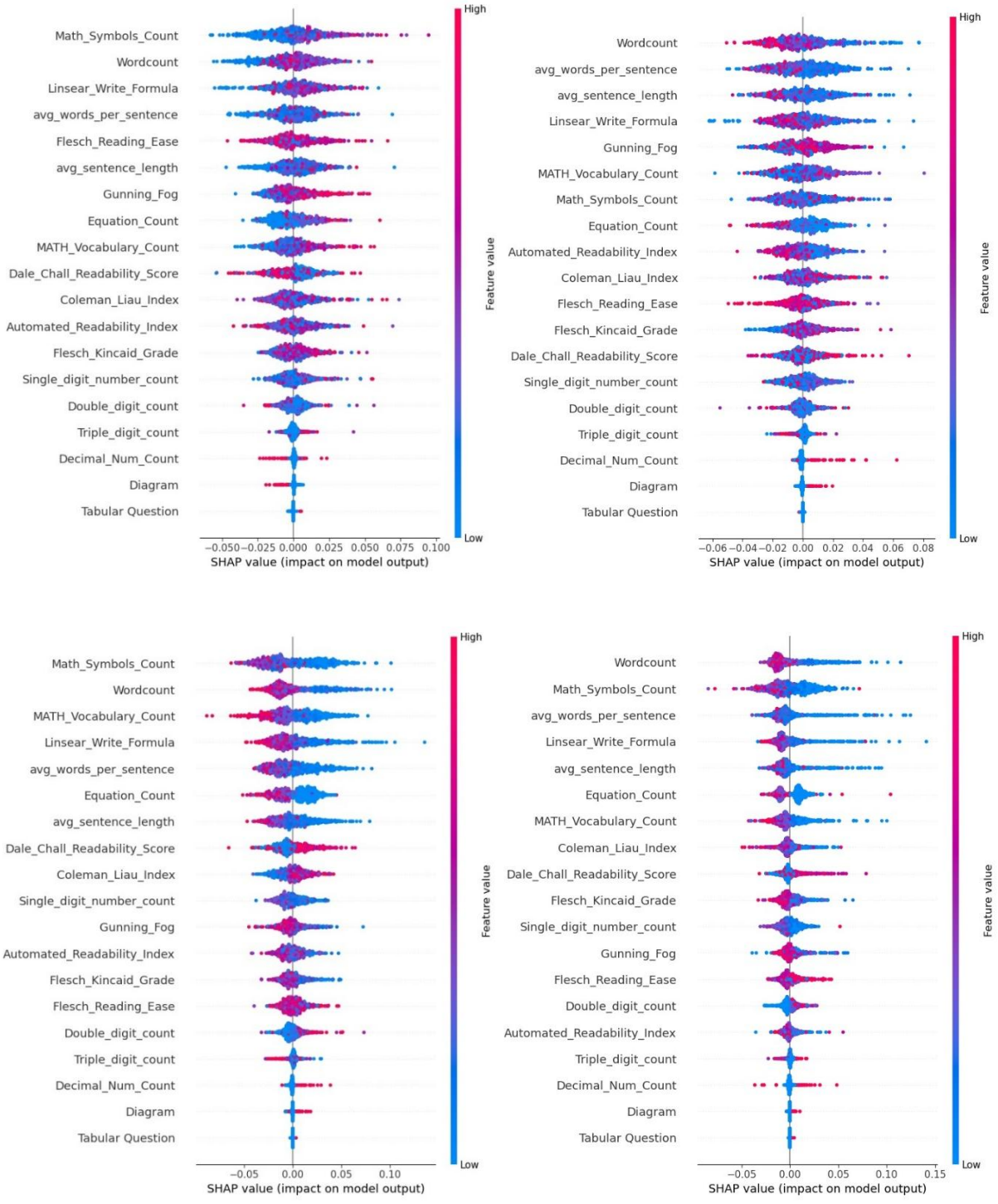

Figure A2. SHAP values by difficulty level 1, 2, 3 and 4 respectively.

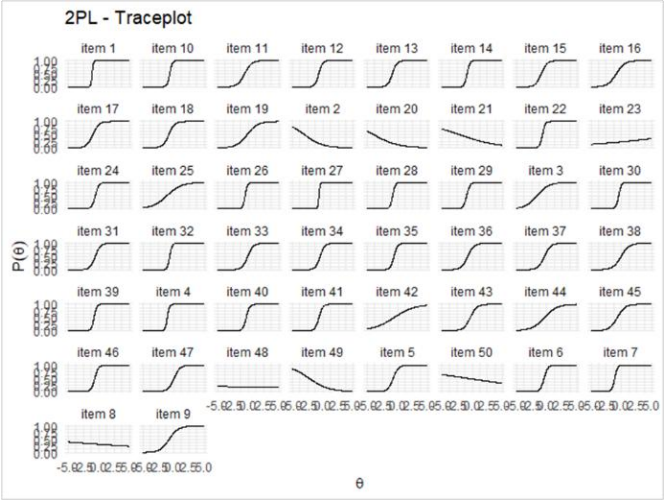

Figure A3. 2 PL traceplot.

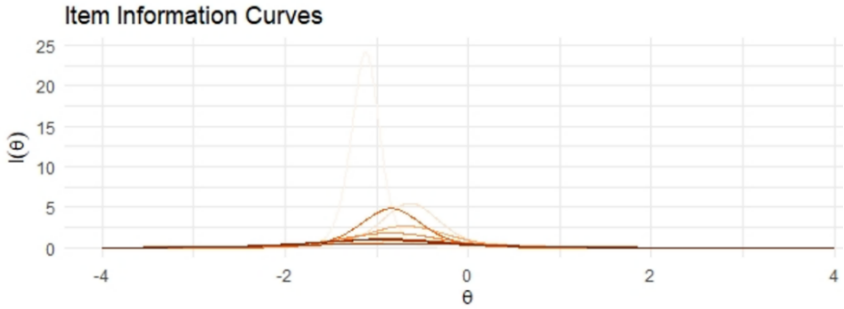

Figure A4. Avg. words per sentence and word count have greater influence.

Conflicts of Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome. We confirm that the manuscript has been read and approved by all named authors and that no other persons have satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us. We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property. We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). She is responsible for communicating with the other authors about progress, submissions of revisions, and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the corresponding author.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that help improve the quality of this work

AI Disclosure

During the preparation of this work the author(s) used generative AI in order to improve the language of the article. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

Acosta-Tello, E. (2010). Making mathematics word problems reliable measures of student mathematics abilities. *Journal of Mathematics Education*, *3*(1), 15-26.

AlKhuzaey, S., Grasso, F., Payne, T.R., & Tamma, V. (2024). Text-based question difficulty prediction: a systematic review of automatic approaches. *International Journal of Artificial Intelligence in Education*, *34*(3), 862-914.

Attali, Y., Saldivia, L., Jackson, C., Schuppan, F., & Wanamaker, W. (2014). Estimating item difficulty with comparative judgments. *ETS Research Report Series*, *2014*(2), 1-8.

Baker, F.B. (2001). *The basics of item response theory. second edition.* ERIC Publications. Washington, DC.

Barbu, O.C., & Beal, C.R. (2010). Effects of linguistic complexity and math difficulty on word problem solving by English learners. *International Journal of Education*, *2*(2), E6.

Benedetto, L. (2023). A quantitative study of NLP approaches to question difficulty estimation. In *International Conference on Artificial Intelligence in Education* (pp. 428-434). Springer Nature, Switzerland. https://doi.org/10.1007/978-3-031-36336-8_67.

Cetintas, S., Si, L., Xin, Y.P., Zhang, D., Park, J.Y., & Tzur, R. (2014). A joint probabilistic classification model of relevant and irrelevant sentences in mathematical word problems. *arXiv preprint arXiv:1411.5732*.

Chen, S., Wang, P., Zhou, M., Wang, Z., & He, B. (2022). A comparative analysis of math word problem solving on characterized datasets. In *2022 International Conference on Intelligent Education and Intelligent Research* (pp. 162-168). IEEE. Wuhan, China.

Cheng, S., Liu, Q., Chen, E., Huang, Z., Huang, Z., Chen, Y., Ma, H., Hu, G., & Hu, G. (2019). Dirt: deep learning enhanced item response theory for cognitive diagnosis. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2397-2400). Association for Computing Machinery. New York.

Daroczy, G., Wolska, M., Meurers, W.D., & Nuerk, H.C. (2015). Word problems: a review of linguistic and numerical factors contributing to their difficulty. *Frontiers in Psychology*, *6*, 348. https://doi.org/10.3389/fpsyg.2015.00348.

de Blas, G.D., Gómez-Veiga, I., & García-Madruga, J.A. (2021). Arithmetic word problems revisited: cognitive processes and academic performance in secondary school. *Education Sciences*, *11*(4), 155. https://doi.org/10.3390/educsci11040155.

Gooding, S. (2009). Children's difficulties with mathematical word problems. *Proceedings of the British Society for Research into Learning Mathematics*, *29*(3), 31-36.

Graesser, A.C., McNamara, D.S., Louwerse, M.M., & Cai, Z. (2004). Coh-metrix: analysis of text on cohesion and language. *Behavior Research Methods, Instruments, & Computers*, *36*(2), 193-202. https://doi.org/10.3758/BF03195564.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., & Steinhardt, J. (2021). Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Kadam, S., Srungaram, P.K., Dheeraj, S.Y., Manish, S.S.S.R., Praveen, P.T.V., Pappu, S., & Satpathi, D.K. (2023). Analysis of linguistics and math features for classification of math word problems: insights and future directions. *International Journal of Management and Applied Science, 9*(8), 18-22.

Kurdi, G., Leo, J., Parsia, B., Sattler, U., & Al-Emari, S. (2020). A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education*, *30*(1), 121-204.

Lan, Y., Wang, L., Zhang, Q., Lan, Y., Dai, B.T., Wang, Y., Zhang, D., & Lim, E.P. (2022). Mwptoolkit: an open-source framework for deep learning-based math word problem solvers. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 36, No. 11, pp. 13188-13190). Canada. https://doi.org/10.1609/aaai.v36i11.21723.

Lee, B.W., & Lee, J. (2020). LXPER index 2.0: improving text readability assessment model for L2 English students in Korea. *arXiv preprint arXiv:2010.13374*.

Lee, F.L., & Heyworth, R. (2000). Problem complexity: a measure of problem difficulty in algebra by using computer. *Education Journal-Hong Kong-Chinese University Of Hong Kong*, *28*(1), 85-108.

Lin, X., Peng, P., & Zeng, J. (2021). Understanding the relation between mathematics vocabulary and mathematics performance: a meta-analysis. *The Elementary School Journal*, *121*(3), 504-540.

Mandal, S., & Naskar, S.K. (2021). Classifying and solving arithmetic math word problems—an intelligent math solver. *IEEE Transactions on Learning Technologies*, *14*(1), 28-41.

Pelánek, R., Effenberger, T., & Čechák, J. (2022). Complexity and difficulty of items in learning systems. *International Journal of Artificial Intelligence in Education*, *32*(1), 196-232. https://doi.org/10.1007/s40593-021-00252-4.

Pérez, E.V., Santos, L.M.R., Pérez, M.J.V., de Castro Fernández, J.P., & Martín, R.G. (2012). Automatic classification of question difficulty level: teachers' estimation vs. students' perception. In *2012 frontiers in Education Conference Proceedings* (pp. 1-5). IEEE. Seattle, WA, USA.

Pimentel, J. L., & Villaruz, M.L.A. (2020). Comparison of item difficulty estimates in a basic statistics test using ltm and CTT software packages in R. *International Journal of Advanced Computer Science and Applications*, *11*(3), 367-372.

Pongsakdi, N., Laine, T., Veermans, K., Hannula-Sormunen, M.M., & Lehtinen, E. (2016). Improving word problem performance in elementary school students by enriching word problems used in mathematics teaching. *NOMAD Nordic Studies in Mathematics Education*, *21*(2), 23-44.

Sanz, M.T., López-Iñesta, E., Garcia-Costa, D., & Grimaldo, F. (2020). Measuring arithmetic word problem complexity through reading comprehension and learning analytics. *Mathematics*, *8*(9), 1556. https://doi.org/10.3390/math8091556.

Sepeng, P., & Madzorera, A. (2014). Sources of difficulty in comprehending and solving mathematical word problems. *International Journal of Educational Sciences*, *6*(2), 217-225.

Sunde, P.B., Bjerre, M., Sunde, P., & Pind, P. (2023). Word problems, item difficulty and low performers. In *Mathematical Cognition and Learning Society Conference 2023*. Loughborough University, UK. https://www.the-mcls.org/mcls-2023.

Theephoowiang, K., & Chaowicharat, E. (2022). Difficulty level estimation of mathematics problems using machine learning. In *Proceedings of the 2022 4th International Conference on Image, Video and Signal Processing* (pp. 231-237). ACM Digital Library. https://doi.org/10.1145/3531232.3531266.

Unson, J.C. (2021). Vocabulary and identification of information: difficulties and challenges in word problems solving. *International Journal of Scientific Research and Management*, *9*(08), 338-357.

Verschaffel, L., Schukajlow, S., Star, J., & Van Dooren, W. (2020). Word problems in mathematics education: a survey. *ZDM Mathematics Education*, *52*(1), 1-16. https://doi.org/10.1007/s11858-020-01130-4.

Walkington, C., Clinton, V., Ritter, S.N., & Nathan, M.J. (2015). How readability and topic incidence relate to performance on mathematics story problems in computer-based curricula. *Journal of Educational Psychology*, *107*(4), 1051. https://doi.org/10.1037/edu0000036.

Wauters, K., Desmet, P., & Van Den Noortgate, W. (2010). Adaptive item-based learning environments based on the item response theory: Possibilities and challenges. *Journal of Computer Assisted Learning*, *26*(6), 549-562. https://doi.org/10.1111/j.1365-2729.2010.00368.x.

Wauters, K., Desmet, P., & Van Den Noortgate, W. (2012). Item difficulty estimation: an auspicious collaboration between data and judgment. *Computers & Education*, *58*(4), 1183-1193.

Zhang, D., Wang, L., Zhang, L., Dai, B.T., & Shen, H.T. (2019). The gap of semantic parsing: a survey on automatic math word problem solvers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(9), 2287-2305. https://doi.org/10.1109/TPAMI.2019.2914054.

Zhou, Y., & Tao, C. (2020). Multi-task BERT for problem difficulty prediction. In *2020 International Conference on Communications, Information System and Computer Engineering* (pp. 213-216). IEEE. Kuala Lumpur, Malaysia.

Zollman, A. (2009). Mathematical graphic organizers. *Teaching Children Mathematics*, *16*(4), 222-229.