

Article

Benchmarking Multimodal Mathematical Reasoning: Prompt Effects, Modality Gaps, and Failure Modes

Gökan Görer, Maria Osipenko *  and Thomas Knispel

Faculty of Business and Economics, Berlin School of Economics and Law, Badensche Str. 50-52,
10825 Berlin, Germany; g.goerer1@gmail.com (G.G.)

* Correspondence: osipenko@hwr-berlin.de

Abstract

Large language models and vision–language models already achieve strong results on reasoning tasks, but their reliability under controlled assessment-style conditions remains insufficiently characterized. This paper presents a benchmark study of multimodal multiple-choice mathematical reasoning using 324 Austrian Mathematical Kangaroo competition problems (2022–2024), including both text-only and diagram-dependent items. We evaluate five state-of-the-art models under a controlled protocol that isolates two factors: input modality and prompt format. We compare a strict short-answer condition requiring a single option label (one_liner) with a structured condition eliciting step-by-step reasoning and an explicit final answer (full) while enforcing deterministic decoding and rule-based answer extraction. Performance is assessed using accuracy, abstention rates, and contest-style scoring, supported by paired and unpaired statistical analyses and a structured error taxonomy. The results show that prompt format is the primary driver of performance: structured prompting yields substantial gains across all the models, particularly on text-only items. In contrast, visual-text problems remain consistently harder, with a robust performance gap that persists across prompting conditions, indicating persistent limitations in visual grounding. Model comparisons are additionally influenced by response strategies, especially abstention behavior under strict output constraints. An error analysis reveals systematic failure modes, including constraint violations, inappropriate strategy selection, and diagram misinterpretation, alongside structured biases in multiple-choice selection under constrained prompting. Overall, the findings demonstrate that measured performance is highly sensitive to the interaction between prompt format and input modality. This underscores the importance of treating prompting, decoding, and answer extraction as integral components of evaluation in assessment-oriented settings, where reliability and reproducibility are central.

Keywords: vision–language models; prompting sensitivity; assessment reliability; competition mathematics



Academic Editor: Carlo Galli

Received: 24 April 2026

Revised: 11 June 2026

Accepted: 22 June 2026

Published: 26 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Despite rapid progress in large language models (LLMs) and vision–language models (VLMs), their reliability under assessment-style conditions remains insufficiently understood (e.g., Hendrycks et al. [1]). In contrast to open-ended benchmarks, standardized evaluation settings require predictions that are not only accurate but also reproducible, scorable, and stable under fixed protocols. This challenge becomes more pronounced in

multimodal tasks, where problems often depend on diagrams, spatial relations, or structured visual inputs (e.g., Lu et al. [2]; Hendrycks et al. [1]). In such cases, models may produce coherent reasoning while arriving at incorrect answers due to misinterpretation of the visual evidence. Recent studies similarly report performance variation across modalities and difficulty levels, with models frequently underutilizing diagrammatic information (e.g., Mondorf and Plank [3]).

This paper investigates these issues in a controlled exam-like setting using Austrian Mathematical Kangaroo (Känguru der Mathematik) multiple-choice problems from 2022 to 2024. We evaluate five LLM/VLM systems on a fixed dataset of 324 items, systematically varying two key factors: input modality (text-only vs. visual-text) and prompt format. Specifically, we compare a strict short-answer condition requiring a single option label (one_liner) with a structured condition that elicits reasoning and a clearly marked final answer (full). All the models are evaluated under deterministic decoding and consistent answer extraction to ensure reproducibility.

This article makes three contributions. First, it introduces a machine-readable benchmark derived from Austrian Mathematical Kangaroo competition materials, including structured answer options, stable identifiers, and linked images for diagram-dependent items. Second, it defines a controlled evaluation protocol that isolates the effects of prompt format and modality while enforcing deterministic inference and parsing. Third, it provides a systematic empirical analysis of model performance, including accuracy, abstention behavior, contest-style scoring, and statistically grounded comparisons across conditions, complemented by a structured qualitative error analysis.

Competition mathematics provides a particularly suitable testbed for this analysis. Multiple-choice contest problems offer unambiguous ground truth, a fixed response space (in our setting: A–E), and carefully constructed distractors such that small errors in constraint interpretation, strategy selection, or arithmetic can directly affect the final answer. At the same time, many items rely on diagrams or spatial structure, requiring models to integrate textual reasoning with visual grounding. This combination enables the analysis of both reasoning performance and systematic failure modes, including cases where explanations are internally coherent but based on incorrect interpretations of the visual input.

A key methodological consideration in this setting is that the evaluation protocol itself becomes part of the measurement instrument. Prompt design, output constraints, and decoding strategies can significantly influence model behavior, affecting both reasoning processes and final predictions. In multiple-choice evaluation, where free-form outputs must be mapped to a single discrete label, deterministic parsing is required for reproducible scoring. Consequently, prompt format, decoding configuration, and answer-extraction rules are treated as first-class components of the experimental design rather than implementation details.

Within this framework, the study addresses the following research questions:

- RQ1 (Overall performance): How do the evaluated models perform (in terms of accuracy and abstention rates) on the Austrian Kangaroo benchmark?
- RQ2 (Prompt-format effect): How does prompt format influence accuracy and abstention behavior?
- RQ3 (Modality effect): What is the performance difference between text-only and visual-text items, and how does it interact with prompt format?
- RQ4 (Failure analysis): What error types characterize model failures (e.g., constraint violations, strategy errors, or diagram-related failures)?

To answer these questions, we construct a structured dataset from Austrian Mathematical Kangaroo PDFs (2022–2024) and evaluate all the models under a controlled protocol that varies only modality and prompt format. Performance is measured using accuracy

(with abstentions counted as non-correct), abstention rates, and contest-style scoring, with subgroup analyses across year, level, and point tier. Statistical robustness is assessed using paired and unpaired methods as appropriate, and aggregate findings are complemented by a qualitative error analysis.

The Mathematical Kangaroo competition was selected because it combines several properties that are valuable for assessment-oriented evaluation. First, the competition covers multiple grade levels and difficulty tiers while preserving a standardized item structure. Second, many problems integrate diagrams, geometric figures, or visual patterns, allowing controlled investigation of multimodal reasoning. Third, the fixed five-option multiple-choice format provides well-defined distractors and objective scoring, making it possible to analyze not only overall accuracy but also systematic error patterns and answer-selection behavior.

The study is limited to Austrian Mathematical Kangaroo multiple-choice items from 2022 to 2024 and five selected models evaluated under two prompt formats and two modalities. The response space is restricted to {A, B, C, D, E} with an optional abstention label (N), and all the outputs are mapped to a single label using deterministic parsing rules.

The remainder of the paper is structured as follows. Section 2 reviews relevant background and related work. Section 3 describes the dataset construction and composition. Section 4 presents the experimental methodology. Section 5 reports the quantitative results and qualitative error analysis. The final section discusses implications, limitations, and future directions for research.

2. Related Works

In this section, we provide the conceptual and empirical background needed to situate this study. First, we introduce the core building blocks of modern LLMs and VLMs, focusing on the architectural ideas that matter for multimodal mathematical reasoning. Then we review prior work on using such models in educational and mathematical settings, including benchmark design and common evaluation practices. The goal is not an exhaustive survey but a focused overview of the most relevant research to the experimental setup and results presented in the later sections.

2.1. Transformer-Based LLMs

Transformers are the foundational architecture of modern LLMs, enabling scalable sequence modeling through self-attention mechanisms that capture dependencies across all the token positions within a layer (e.g., Vaswani et al. [4]). Unlike recurrent architectures, Transformers process sequences in parallel, allowing efficient training on large corpora and supporting strong in-context learning behavior.

At a high level, inputs are represented as token embeddings augmented with positional encodings since attention alone does not encode word order. A standard Transformer layer consists of two main components: multi-head self-attention, which integrates information across tokens, and a position-wise feed-forward network, which transforms token representations independently. In the attention mechanism, hidden states are projected into queries, keys, and values (Q, K, V); attention weights are computed via scaled dot products between queries and keys and applied to aggregate values. Multi-head attention extends this process across multiple learned projections, enabling the model to capture diverse relational patterns within the sequence (e.g., Vaswani et al. [4]).

Most LLMs adopt a decoder-only architecture with causal masking, where each token attends only to preceding tokens, enabling autoregressive next-token prediction. Training is typically based on cross-entropy loss over large text corpora, while inference proceeds through iterative token generation. This setup also underlies in-context prompting, where

instructions and examples provided in the input can steer model behavior without parameter updates (e.g., Brown et al. [5]). At the same time, it implies that outputs are sensitive to prompt structure and decoding settings, such as temperature, which influence determinism and variability.

Model capability is closely tied to scale and training allocation. Empirical scaling laws show predictable performance improvements with increasing model size, data, and compute, although effective scaling requires balancing these factors appropriately (e.g., Hoffmann et al. [6]). As a result, larger models tend to exhibit stronger reasoning performance but may also show increased sensitivity to prompting and evaluation protocols.

These properties are directly relevant for assessment-oriented evaluation. In multiple-choice settings, where outputs must be mapped to discrete labels, small variations in prompt design, output constraints, or decoding configuration can affect both intermediate reasoning and final predictions. This motivates the use of controlled inference settings and deterministic answer extraction, as adopted in this study.

While the above discussion focuses on text-only models, the same Transformer foundation extends naturally to multimodal systems. VLMs incorporate visual inputs by encoding images into token-like representations that can be processed alongside text within the same attention framework. This enables joint reasoning over textual and visual information but also introduces additional challenges related to grounding and the correct interpretation of diagrammatic content, which are central to the tasks considered in this work.

2.2. Vision Encoders and Image Tokenization

In VLMs, images must be converted into vector representations (“visual tokens”) that can be processed jointly with text by a Transformer-based language model. This requires a vision encoder that maps raw pixel inputs to feature vectors, as well as a tokenization scheme that determines the granularity and structure of these representations, such as object regions, image patches, or compressed token sets.

Earlier multimodal pipelines often relied on convolutional backbones combined with region-based representations. For example, convolutional networks such as ResNet produce spatial feature maps (e.g., He et al. [7]), while detector-based approaches extract object-centric region features (e.g., Ren et al. [8]). These region-level features can then be treated as tokens within a multimodal Transformer, enabling object-aware reasoning over images.

More recent approaches avoid explicit object detection and instead tokenize images directly into patches. The Vision Transformer follows this paradigm by splitting an image into fixed-size patches, embedding each patch, and adding positional encodings to preserve spatial structure before processing the sequence with Transformer layers (Dosovitskiy et al. [9]). A key implication of this design is that the number of visual tokens scales with image resolution, leading to increased computational and memory costs for high-resolution inputs.

To interface with a language model, visual features are typically projected into the embedding space of the LLM and incorporated either as additional input tokens or via cross-attention mechanisms. Because long sequences of patch tokens are computationally expensive, many VLM architectures apply token reduction techniques before fusion. Common strategies include resampling spatial feature grids into a fixed number of tokens or using learned query mechanisms to extract a compact set of informative representations (e.g., Li et al. [10]).

The term “tokenization” can also refer to discrete representations obtained via vector quantization, where images are mapped to codebook indices. While this approach supports discrete generative objectives, most contemporary VLM systems operating on document-like inputs (e.g., diagrams or structured figures) rely on continuous patch-based tokens,

sometimes combined with token compression, to preserve fine-grained spatial detail while remaining compatible with Transformer-based decoding (e.g., Radford et al. [11]).

These design choices directly affect how visual information is represented and transmitted to the language model. In particular, token granularity and compression determine the extent to which fine spatial relationships, small symbols, and diagrammatic constraints are preserved. This is especially relevant for mathematical problem-solving tasks, where subtle visual details can be critical for correctness. Limitations at the encoding or tokenization stage may therefore propagate into downstream reasoning errors, even when the language model produces coherent explanations.

This representation bottleneck motivates the next design question: once visual tokens are obtained, how are they integrated with textual representations so that generation can reliably leverage visual evidence rather than defaulting to linguistic priors?

2.3. VLMs and Fusion

VLMs extend Transformer-based language models by conditioning text generation on visual inputs. In this work, the term multimodal LLM (MLLM) refers specifically to VLMs that support open-ended instruction-following generation rather than task-specific prediction. Technically, images are encoded into continuous feature vectors (“visual tokens”), and the central design question becomes how these tokens are integrated into the language model such that fine-grained visual information remains accessible during generation.

The prior work on multimodal learning can be broadly organized by how and where visual and textual representations are combined. The early approaches include dual-encoder architectures that learn aligned image–text embeddings via contrastive objectives (e.g., CLIP, AI [12]), as well as multimodal Transformers that fuse modalities through cross-attention for downstream understanding tasks, such as visual question answering (e.g., UNITER, Chen et al. [13]). While these models produce strong grounded representations, they are not inherently designed for instruction-following generation.

More recent MLLMs instead couple a vision encoder with a pretrained autoregressive LLM and train the combined system for generative tasks. Within this paradigm, fusion mechanisms can be grouped into three main categories. First, token-level concatenation (or prefix-based fusion) inserts projected visual tokens directly into the input sequence alongside text tokens (e.g., Wei et al. [14]). Second, cross-attention architectures preserve the language model’s token stream while introducing dedicated layers that allow textual representations to attend selectively to visual tokens (e.g., Flamingo, Alayrac et al. [15]). Third, bottleneck or adapter-based approaches compress visual information into a small set of learned tokens that act as a soft prompt for the language model (e.g., Nagrani et al. [16]).

These design choices introduce trade-offs between computational efficiency and representational fidelity. In particular, compressing visual tokens reduces memory and latency but may discard fine-grained spatial details that are critical for interpreting diagrams, symbols, or structured layouts. As a result, multimodal performance reflects an end-to-end interaction between visual encoding, fusion, and language generation. For diagram-dependent tasks such as those in the Kangaroo benchmark, errors may originate at any stage of this pipeline, from incomplete visual grounding to incorrect reasoning or final answer selection.

2.4. LLMs and VLMs in Educational Tasks

Educational applications provide a particularly demanding setting for evaluating LLMs and VLMs as they require not only high average performance but also stable, reproducible, and interpretable outputs under well-defined scoring rules. Models are increasingly explored both as learning supports, providing explanations, hints, or feedback, and

as components of assessment workflows, including automated scoring, item generation, and evaluation assistance.

A key advantage of generative models in assessment contexts is their ability to produce both answers and accompanying rationales. However, this flexibility introduces challenges for measurement validity. Model outputs are sensitive to prompt design and output constraints, may contain fluent but incorrect reasoning, and often require deterministic parsing to map free-form text to structured evaluation formats (e.g., Zhuo et al. [17]). Consequently, the evaluation protocol, including prompting strategy and answer extraction, becomes part of the assessment instrument rather than a neutral implementation detail.

One prominent research direction focuses on LLM-based scoring of constructed responses. Prior work shows that structured prompting, including rubric-based instructions and rationale generation (e.g., chain-of-thought prompting), can improve scoring consistency and reduce superficial pattern matching (e.g., Wei et al. [18]). At the same time, these methods highlight that prompt design materially affects both model behavior and scoring reliability. Data limitations further complicate evaluation, and recent approaches use synthetic data generation to augment training sets and mitigate class imbalance (e.g., Fang et al. [19]).

VLMs extend these challenges to multimodal settings, where both the task and the evidence may include visual components such as diagrams, graphs, or geometric figures. Benchmarks such as ScienceQA demonstrate that performance depends on integrating textual reasoning with visual grounding, and that explanation-based prompting can improve results in such settings (e.g., Lu et al. [2]). However, they also underscore that coherent reasoning does not guarantee correctness when visual interpretation is flawed.

For the present study, this literature primarily motivates the evaluation design. Rather than focusing on tutoring or interaction, the emphasis is on assessment-like conditions where outputs must be comparable, reproducible, and directly scorable. This requires controlled prompting, deterministic decoding, and consistent answer extraction. The benchmark and analysis framework developed in this work therefore treat these protocol choices as integral components of the evaluation and explicitly examine how they influence both performance and failure behavior.

2.5. Mathematical Reasoning and Multimodal Benchmarks

Competition-style mathematics problems, such as those in the Kangaroo contest, expose a specific form of brittleness in current models: small errors in constraint interpretation can invalidate an otherwise coherent solution. For example, misreading a condition such as “exactly three digits” as “at most three digits” is sufficient to produce an incorrect answer despite correct downstream reasoning. Unlike open-ended mathematical tasks, where partial correctness may remain hidden, the multiple-choice format with carefully constructed distractors makes such errors directly observable. This property makes Kangaroo-style problems particularly suitable for studying reliability under fixed evaluation protocols.

A complementary line of work focuses on multi-step arithmetic reasoning expressed in natural language. The benchmark of Cobbe et al. [20] contains 8.5K grade-school word problems with annotated solutions and highlights the fragility of reasoning chains: small mistakes in problem setup or intermediate calculations often lead to entirely incorrect outcomes. To address this, the authors propose verification-based strategies, such as generating multiple candidate solutions and selecting among them, as a means of improving robustness.

Benchmark construction itself also plays a critical role in evaluating reasoning. Patel et al. [21] show that several math word problem datasets contain exploitable shortcut cues that allow models to succeed without fully understanding the problem structure. They

introduce SVAMP as a challenge set designed through controlled perturbations of existing problems, explicitly targeting such weaknesses and requiring deeper semantic reasoning.

Prompting and evaluation protocols have emerged as central factors in mathematical reasoning performance. Wei et al. [18] demonstrate that chain-of-thought prompting providing intermediate reasoning steps can substantially improve performance on arithmetic and symbolic reasoning tasks, particularly for larger models. This finding underscores that measured performance is not solely a function of model capability but also of the interaction between model and evaluation protocol. In assessment-style settings, this interaction becomes especially important as different prompt formats (e.g., explanation-based vs. strict short-answer) and answer-extraction rules can systematically alter both accuracy and error patterns.

Finally, a growing body of work addresses multimodal mathematical reasoning, where visual inputs such as diagrams, plots, or tables are integral to the task. Benchmarks such as in Lu et al. [2] extend traditional reasoning tasks by requiring models to integrate textual and visual information. The empirical results show that even strong multimodal models struggle with such tasks, particularly when fine-grained visual interpretation is required. A recurring pattern is that models may produce coherent and well-structured explanations that are nevertheless grounded in incorrect interpretations of diagrams or spatial relationships, leading to incorrect final answers.

Across these benchmarks, a consistent pattern emerges: apparent reasoning performance depends on the interaction between task design, evaluation protocol, and, when applicable, visual grounding. Benchmark structure determines whether shallow heuristics are sufficient, prompting strategies influence reasoning behavior and output format, and multimodal inputs introduce additional sources of error related to perception and grounding. These considerations motivate the use of a competition-style dataset with fixed items, controlled prompting, and explicit modality distinctions, as adopted in this study.

2.6. Prompting Strategies and Output Reliability

Prompting strategies can substantially influence measured reasoning performance and are therefore an important consideration in assessment-oriented evaluation. Large language models adapt their behavior at inference time based on prompt structure alone, making prompt design a key determinant of observed performance (e.g., Brown et al. [5]).

One central design choice concerns the balance between encouraging intermediate reasoning and enforcing output constraints. Reasoning-oriented approaches such as chain-of-thought prompting have been shown to improve performance on multi-step reasoning tasks by eliciting intermediate solution steps (e.g., Wei et al. [18]). Even minimal reasoning cues such as “Let’s think step by step” can produce measurable gains (Kojima et al. [22]). Related work also highlights two-stage approaches that first elicit reasoning and then extract a final answer in a constrained format (Kojima et al. [22]).

At the same time, longer reasoning traces introduce challenges for reliable evaluation. Models may generate plausible but incorrect reasoning or provide ambiguous conclusions that complicate deterministic scoring. Self-consistency sampling addresses this issue by aggregating multiple reasoning paths and selecting the most frequent answer, improving performance on benchmarks such as GSM8K and SVAMP (Wang et al. [23]). However, these gains come at the cost of increased computational expense and reduced reproducibility.

Output constraints provide a complementary mechanism for improving evaluation reliability. Structured generation and constrained decoding can reduce invalid outputs and improve parseability, but overly restrictive constraints may also alter reasoning behavior and reduce task performance (Tam et al. [24]). Consequently, evaluation protocols must balance reasoning support against output standardization.

Overall, the prior work suggests that prompting and output constraints are not merely formatting choices but factors that can materially affect measured performance. This motivates treating prompt format as a controlled experimental variable in assessment-oriented benchmarks.

2.7. Positioning of This Study

This study is positioned at the intersection of multimodal model evaluation, assessment-style benchmarking, and prompt sensitivity analysis in large language and vision–language models. While the prior work has extensively investigated Transformer architectures, vision encoders, and multimodal fusion mechanisms, these components are treated here as established building blocks rather than the focus of the contribution. Instead, this work emphasizes evaluation design and protocol sensitivity as the primary objects of analysis.

From this perspective, multimodal models are treated as fixed systems, and the benchmark is used to probe their reliability and sensitivity under varying interaction conditions. This framing aligns the study with recent work emphasizing evaluation robustness, reproducibility, and protocol sensitivity rather than architectural innovation.

Within this setting, the study makes three contributions. First, it introduces a machine-readable benchmark derived from Austrian Mathematical Kangaroo competition materials (2022–2024), including structured multiple-choice options, stable question identifiers, and linked images for diagram-dependent items. Second, it defines a controlled evaluation protocol that isolates the effects of input modality (text-only vs. visual-text) and prompt format (one_liner vs. full) while enforcing deterministic decoding and rule-based answer extraction to ensure reproducibility. Third, it provides a systematic empirical analysis of model behavior across five state-of-the-art systems, including accuracy, abstention behavior, contest-style scoring, and statistically grounded comparisons across conditions, complemented by a structured qualitative error analysis.

3. Dataset Construction

This section describes the construction of the benchmark dataset used in this study. The dataset is derived from Austrian Mathematical Kangaroo competition materials and transformed into a structured machine-readable set of question records for automated evaluation of LLMs and VLMs. It documents the dataset’s provenance, scope, and key properties, which underpin the experimental methodology described in the following section.

3.1. Data Source and Task Definition

The Mathematical Kangaroo (Känguru der Mathematik, <https://www.kaenguru.at/aufgaben.html>, accessed 12 November 2025) is an international multiple-choice mathematics competition designed to promote mathematical reasoning and problem-solving among pupils. It is coordinated by Kangourou sans Frontières and administered annually across participating countries with standardized age-stratified problem sets. In Austria, the competition is organized by the Verein “Känguru der Mathematik—Österreich,” following the international framework with nationally administered materials and procedures.

The dataset used in this study is constructed from publicly available PDF problem sheets from the Austrian competition years 2022–2024. These materials were obtained via the *matematica.pt* “Kangaroo Questions” archive, which provides downloadable test papers organized by year and level. The PDFs explicitly indicate Austrian provenance (e.g., country labels and references to the national organizer), consistent with the official Austrian competition context. The official Austrian competition website is additionally referenced as a primary source for “Aufgaben & Lösungen”, supporting dataset traceability.

All the problems are presented in English as provided in the source PDFs. We perform no translation or language normalization. This design choice avoids introducing translation artifacts and ensures that observed model performance reflects mathematical reasoning and, where applicable, visual interpretation rather than cross-lingual processing effects.

Each item in the benchmark is a five-option multiple-choice question (A–E). A subset of problems includes diagrams or visual elements that are essential for solving the task; these are explicitly annotated as visual items. Appendix A provides representative examples of both text-only and visual question formats in the standardized evaluation representation used in this study.

The analysis focuses on four Austrian competition levels: Écolier, Benjamin, Kadett, and Junior (corresponding to grades 3–10). These levels were selected because they jointly cover primary and lower-secondary education while maintaining a consistent task structure across years, enabling controlled comparisons. Each selected level is present in all three competition years (2022–2024), supporting cross-year comparability and robustness analysis.

Other Austrian levels (e.g., Felix for grades 1–2 and Student for upper secondary) were excluded to maintain a homogeneous difficulty and curricular range. Including these levels would introduce additional structural and developmental variation, reducing interpretability in cross-level and cross-year comparisons.

3.2. Extraction and Preprocessing Pipeline

This section describes the transformation of Austrian “Känguru der Mathematik” PDF booklets into structured question records suitable for automated evaluation. The objective is to produce a machine-readable format while preserving full provenance information, including year, level, question number, point tier, and source page indices.

Each PDF (corresponding to a specific year and level) is processed sequentially on a page-by-page basis. Extracted page text is concatenated in reading order to reconstruct the full booklet while maintaining a mapping from character spans to originating page numbers. This enables each extracted item to be traced back to its exact source location and supports debugging as well as later alignment with diagram extractions.

Question segmentation is performed on the reconstructed text by detecting numeric question markers (e.g., “1.”, “2.”). Because formatting varies across PDFs (e.g., leading zeros such as “01.” or missing whitespace in constructs like “13.Jelena ...”), segmentation uses a regular expression designed to be robust to such inconsistencies. Front-matter is excluded by ignoring numeric markers prior to the first valid task block, identified via the booklet header preceding the initial problem statement. Each extracted segment is stored verbatim as `chunk_text` to preserve the raw extraction state.

For each item, metadata fields are assigned deterministically: `year`, `level`, and `question_number`. Page indices are derived from the span-to-page mapping, and point values are computed according to the official tier structure (Table 1), which depends jointly on level and question index. Each question is assigned a stable identifier of the form `<year>-<level>-q<NN>` (e.g., 2022-Ecolier-q02), which serves as the primary key across all experimental conditions.

Finally, each chunk is parsed into a structured representation consisting of `question_text` and an `options` object with exactly five entries (A–E). Options are extracted using A–E markers, which removes most formatting artifacts introduced by PDF line breaks. Items that cannot be reliably parsed into the required structure are flagged with `parse_ok = false` and excluded from downstream evaluation to ensure consistency.

The resulting dataset is exported in JSON Lines (JSONL) format, with validation checks for expected item counts per level, unique identifiers, valid option completeness (for

parse_ok = true cases), and consistency of point assignments with Table 1. The following section describes the final schema used for storage and evaluation.

Table 1. Test length and point distribution by level (Austrian Mathematical Kangaroo). The first 1/3 of questions score 3 points, the second 1/3 4 points, and the last 1/3 5 points. Unanswered questions score 0 points; incorrect answers incur a penalty of $-1/4$ of the question's point value.

Level	Grades	Questions	Time_limit_min
Écolier	3–4	24	60
Benjamin	5–6	24	60
Kadett	7–8	30	75
Junior	9–10	30	75

3.3. Dataset Schema and Storage Format

This section defines the final JSONL schema used throughout the study. The design prioritizes compatibility with evaluation pipelines, deterministic joining across experimental conditions, and consistent handling of both text-only and image-dependent items.

The dataset is stored in JSONL format, with one record per line. This format enables efficient streaming, incremental processing, and straightforward merging of outputs across models, prompts, and modalities. All files are encoded in UTF-8 to preserve special characters appearing in problem statements. Each record is indexed by a stable identifier of the form YYYY-Level-qNN (e.g., 2022-Ecolier-q04), which serves as the primary key for alignment with ground truth and across experimental runs.

Each record includes the following core fields: year, level, question_number, and points, where points are assigned according to the official Kangaroo tier structure based on level and question index. The problem statement is stored in question_text, with the leading numeric label removed. Answer options are represented as an options dictionary with exactly five keys ("A" through "E"), each mapping to a string. The correct answer is stored in gold_answer (values in {A, B, C, D, E}). In per-run evaluation outputs, this value is duplicated under the field gold for convenience; both fields refer to the same ground-truth label.

To ensure traceability, optional provenance fields include chunk_text, which stores the raw extracted segment prior to parsing, and page_numbers, a list of source page indices in the original PDF. Modality information is explicitly encoded via the boolean flag requires_image. When requires_image = true, the record includes image_paths, a list of local file paths corresponding to extracted diagrams; otherwise, this field is empty or omitted.

A final integrity flag, parse_ok, indicates whether the item was successfully structured into the required format, in particular whether all five options (A–E) were successfully recovered. Only records with parse_ok = true are included in evaluation, ensuring that parsing artifacts do not affect reported results.

3.4. Visual Subset and Image Handling

A substantial fraction of Mathematical Kangaroo problems rely on non-textual information, including diagrams, grids, geometric constructions, tables, and visual patterns. In such cases, the image is an integral part of the problem specification rather than auxiliary decoration; omitting it can render the task ambiguous, unsolvable, or semantically altered. To ensure valid evaluation, the benchmark explicitly distinguishes between text-only and visual-text items requiring image input. This separation enables controlled comparisons between text-only language models and vision-language models without conflating reasoning limitations with missing-modality input.

An item is labeled as visual when the accompanying diagram or image is necessary to determine the correct solution. This includes cases where the problem explicitly references a figure (e.g., “see diagram”) and where required information is not fully encoded in text. It also includes items where answer options depend on interpreting spatial or structural properties (e.g., shapes, configurations, or sequences) that cannot be resolved without the associated figure. Conversely, items are not labeled as visual if an image is present but redundant, meaning the problem can be solved using only the textual statement and answer options. This conservative labeling strategy avoids inflating the visual subset and preserves the interpretability of modality-based comparisons.

To illustrate the composition of the visual-text subset, Figure 1 presents author-created examples representing diagram categories found in the benchmark. The visual component spans geometric constructions, grid-based reasoning tasks, spatial transformations, and combinatorial or logical arrangement problems. These categories reflect the diversity of visual information required for successful problem solving and highlight that diagrams are not decorative but often essential to constraint specification.

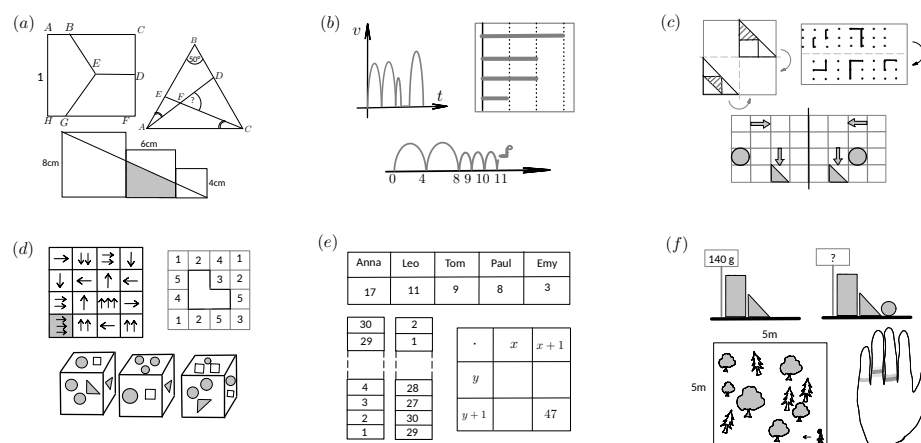


Figure 1. Illustrative author-created examples of diagram types occurring in the visual-text subset: (a) geometric constructions, (b) grids and coordinate systems, (c) spatial transformation tasks, (d) combinatorial and logical arrangement problems, (e) structured visual data (e.g., tables and charts), and (f) contextual illustrations.

The modality distinction is encoded via the boolean field `requires_image`. Text-only items use `requires_image = false` and are evaluated without image input. Visual-text items use `requires_image = true` and include an `image_paths` field containing one or more local file paths to the associated figures. This list structure accommodates rare cases where a question spans multiple pages, although most items require only a single image. Images are stored as lossless PNG files and organized deterministically by year and level. Each file name matches the stable question identifier (e.g., 2022-Eco1ier-q01.png), ensuring straightforward traceability and reproducibility.

During evaluation, visual-text items are presented with the same textual prompt format as text-only items (question stem plus options A–E), augmented with the corresponding image(s) via the model’s multimodal input interface. This design ensures that performance differences primarily reflect access to visual evidence rather than differences in prompt structure. A limitation of this setup is that screenshots may include surrounding textual elements in addition to the diagram; while this reflects realistic document-style inputs, it should be considered when interpreting modality-specific effects.

Consistency checks enforce schema integrity by verifying that text-only items contain no image references and that visual-text items include at least one valid image path. These checks prevent silent failures due to missing or misaligned assets.

3.5. Dataset Statistics and Distribution

This section summarizes the composition of the final benchmark dataset, with an emphasis on its distribution across years, competition levels, and modalities. The goal is to provide transparency regarding dataset structure and to contextualize later experimental results.

Across the selected competition years 2022–2024, the dataset contains 324 questions in total, corresponding to the full set of available items within the defined scope. Each year contributes 108 questions, distributed as follows: Écolier (24), Benjamin (24), Kadett (30), and Junior (30). Aggregated across the three years at hand, this yields 72 questions each for Écolier and Benjamin and 90 questions each for Kadett and Junior. No items were excluded due to parsing failures, missing labels, or missing image assets, resulting in full dataset coverage after preprocessing.

The dataset is partitioned into two modality subsets: 89 text-only questions (27.5%) and 235 visual-text questions (72.5%). Table 2 reports the yearly modality distribution. While the total number of items per year is fixed, the proportion of visual-text items varies moderately, ranging from 69.4% (2022) to 75.9% (2023), with 72.2% in 2024.

Table 2. Dataset size by year and modality (final benchmark).

Year	Total	Text-Only	Visual	Text-Only (%)	Visual (%)
2022	108	33	75	30.56	69.44
2023	108	26	82	24.07	75.93
2024	108	30	78	27.78	72.22
Total	324	89	235	27.47	72.53

Modality distribution also varies substantially across competition levels (Table 3). Écolier and Benjamin are strongly skewed toward visual-text items (87.5% and 84.7%, respectively), Kadett exhibits a more balanced composition (67.8% visual-text), and Junior is the most balanced level overall, with a smaller majority of visual-text items (55.6%). Consequently, text-only items are concentrated in higher levels: 40 of 89 text-only questions (44.9%) originate from Junior and 29 (32.6%) from Kadett, while Écolier and Benjamin together contribute 20 items (22.5%). This imbalance is important for later analysis as modality effects may co-vary with difficulty level and problem structure.

Table 3. Dataset size by level and modality (final benchmark).

Level	Total	Text-Only	Visual	Text-Only (%)	Visual (%)
Écolier	72	9	63	12.50	87.50
Benjamin	72	11	61	15.28	84.72
Kadett	90	29	61	32.22	67.78
Junior	90	40	50	44.44	55.56
Total	324	89	235	27.47	72.53

Difficulty is operationalized via the official Kangaroo point tiers (3/4/5), stored in the `points` field. Because each competition level follows a fixed tier structure and the full set of items is included, the dataset is balanced across point tiers in aggregate (108 items per tier across the full dataset). This provides a standardized proxy for difficulty without requiring additional annotation.

Overall, the benchmark is balanced in terms of total coverage across years and levels while exhibiting a deliberately skewed modality distribution favoring visual-text items, particularly in lower grades. This structure supports separate evaluation of text-only and multi-modal performance and enables controlled analysis of how visual grounding affects model

behavior. The following section details the evaluation methodology, including prompting setups, inference configurations, output parsing, and statistical testing procedures.

4. Methodology

This section describes the evaluation methodology used to benchmark LLMs and VLMs on the Kangaroo dataset introduced in Section 3. The evaluation protocol is designed to support controlled comparisons across models, input modalities (text-only vs. visual-text), and prompting conditions while keeping the underlying task fixed: selecting one of five answer options (A–E) for each multiple-choice item.

To examine the effect of prompting under controlled conditions, each model was evaluated under two prompt formats: a strict one liner condition requiring a single option label (*one_liner*) and a full condition requiring explanatory reasoning followed by an explicitly marked final answer (*full*). Self-consistency and other sampling-based approaches were not used because the objective of the study was to assess model behavior under deterministic and reproducible inference settings.

The section further specifies inference settings, output parsing, evaluation metrics, and statistical procedures. We ensure reproducibility through fixed decoding parameters, consistent preprocessing, and structured logging of all the model outputs.

4.1. Models and Experimental Setup

The benchmark evaluates five vision–language models selected to span different providers, parameter scales, and deployment paradigms while remaining comparable under a unified multiple-choice evaluation protocol. The selection combines: (i) a proprietary frontier model accessed via API (gpt-4o, OpenAI et al. [25]), (ii) two Qwen-based models enabling within-family scaling analysis (qwen2p5-vl-32b and qwen3-vl-235b-a22b, Bai et al. [26]; Bai et al. [27]), and (iii) two Pixtral variants distinguishing a fixed snapshot from a continuously updated endpoint (pixtral-large-2411 and Pixtral-Large-Latest, AI [28]). The overview of the models is presented in Table 4.

Table 4. Overview of selected models.

Model	Provider	Parameters	Headquarters	API
gpt-4o	OpenAI	undisclosed	USA	OpenAI
qwen2p5-vl-32b-instruct	Qwen (Alibaba)	32B	China	Fireworks AI
qwen3-vl-235b-a22b-instruct	Qwen (Alibaba)	235B	China	Fireworks AI
pixtral-large-2411	Mistral AI	undisclosed	France	Mistral
pixtral-large-latest	Mistral AI	undisclosed	France	Mistral

This configuration enables separation of protocol effects (prompt format, modality, and decoding constraints) from differences attributable to model family, scale, or provider-specific training regimes while also providing a limited robustness check against version drift in deployed systems.

4.2. Inference Settings and Output Parsing

All the models are evaluated under a uniform inference configuration to ensure comparability across conditions. Each benchmark item is queried once per combination of model, modality, and prompting condition. The evaluation therefore spans two modalities: text-only (question text only) and visual-text (question text plus associated image) and two prompting formats: *one_liner*, which enforces a single label from {A, B, C, D, E} with optional abstention label N, and *full*, which permits free-form reasoning with a required explicit final answer.

Aside from modality and prompt format, all the inputs, templates, and formatting rules are held constant across the models. All the models were queried through their respective APIs using the lowest available sampling temperature (temperature = 0 or the closest available deterministic setting provided by the API). Moreover, maximum generation lengths were set sufficiently high to avoid truncation, particularly in the full condition.

All the outputs are stored in a consolidated JSONL log for full traceability. Each record includes the model identifier, modality, prompt condition, item ID, year, competition level, question number, ground-truth label, raw model output, parsed prediction, and correctness indicator. This structure enables full reconstruction of all reported aggregate statistics from per-item outputs.

Because model outputs may deviate from the required format, we use a deterministic rule-based parser to map each response to a single label. Outputs are first normalized (case-folding and punctuation stripping). In the full condition, the parser prioritizes an explicitly marked final answer (e.g., "FINAL: X") and extracts the corresponding option; if absent or malformed, it falls back to the last valid option token in the response. In the one_liner condition, direct single-character outputs in {A–E, N} are accepted; otherwise, the parser searches for explicit answer cues (e.g., "Answer: C"; "FINAL: D").

If no valid label can be recovered, the response is mapped to N (invalid/abstention). Abstentions are recorded separately as they represent a distinct behavioral outcome from incorrect answers. A final sanity validation step ensures consistency of the evaluation log, including expected row counts and uniqueness of (model $\times$ modality $\times$ prompt $\times$ item) combinations prior to aggregation.

4.3. Evaluation Metrics

We evaluate model performance on the task of predicting the correct option label for each multiple-choice item. The primary metric is accuracy, defined as the proportion of items for which the parsed prediction matches the ground-truth label. Formally, accuracy is computed as the number of correct predictions divided by the total number of evaluated items for a given configuration (model $\times$ modality $\times$ prompt condition).

Given the explicit abstention mechanism (label N), the evaluation additionally reports abstention rate, defined as the proportion of items assigned N by the parser. This metric distinguishes between incorrect predictions and non-committal or non-compliant outputs. In addition, an abstention-conditioned accuracy is reported, defined as accuracy restricted to non-abstained items, i.e., correct predictions divided by the number of answered questions.

All metrics are reported separately across prompting conditions (one_liner, full) and modalities (text-only and visual-text), enabling isolation of effects attributable to output format and input modality. The results are further stratified by dataset attributes capturing structure and difficulty, including competition year (2022–2024), level (Écolier, Benjamin, Kadett, and Junior), and point tier (3/4/5). These stratified analyses provide finer-grained insight into performance variation across difficulty regimes and problem types.

For within-model comparisons across conditions (e.g., one_liner vs. full), evaluations exploit shared item identifiers to construct paired per-question outcomes (correct/incorrect/abstain). This pairing forms the basis for statistical testing and confidence estimation procedures described in the following section.

4.4. Statistical Analysis

The statistical analysis has two objectives: (i) to assess whether observed performance differences reflect systematic effects rather than finite-sample variability in the benchmark and (ii) to quantify uncertainty around reported estimates beyond point accuracy values.

Because each prediction is associated with a stable item identifier, results can be aligned at the question level, enabling paired analyses when the same items are evaluated under different conditions (e.g., *one_liner* vs. *full* for a given model and modality). In contrast, comparisons involving disjoint item subsets (text-only versus visual-text evaluation) are treated as unpaired.

We assess the uncertainty using non-parametric bootstrap resampling over questions (e.g., Card et al. [29]; Efron and Tibshirani [30]). For each statistic of interest (accuracy, abstention rate, or accuracy differences), items are resampled with replacement and the statistic is recomputed over the resampled dataset, yielding an empirical sampling distribution. Paired comparisons are performed by jointly resampling matched item IDs across conditions, while unpaired comparisons resample independently within each subset. Effect estimates are reported together with 95% bootstrap confidence intervals (e.g., Hesterberg [31]; Xu and Harrar [32]).

For hypothesis testing in paired comparisons, McNemar's test is applied (e.g., Dietterich [33]). This test conditions on discordant outcomes and evaluates whether one condition yields significantly more correct responses than the other on the same items. It is used primarily to assess prompt-condition effects within each model and modality. For unpaired comparisons, such as differences between text-only and visual-text subsets within a given model and prompting condition, a chi-square test of independence is used (e.g., Cohen [34]).

Given the large number of comparisons across models, modalities, and prompting conditions, inference is primarily based on effect sizes and confidence intervals, with hypothesis tests treated as complementary evidence. For each major comparison, we report the mean accuracy in each condition, the corresponding accuracy difference, the bootstrap 95% confidence interval for this difference, and the associated *p*-value from the relevant test.

5. Results

This section presents the empirical results of the benchmark evaluation described in Section 4. All the models are evaluated on the same fixed set of Austrian Kangaroo competition items (2022–2024) under two prompting conditions (*one_liner* and *full*) and two modalities (text-only vs. visual-text). Performance is primarily reported using multiple-choice accuracy, defined as the proportion of items for which the parsed prediction matches the gold label. In addition, we report abstention behavior (label N) as an explicit measure of response compliance and uncertainty. Where relevant, the results are complemented with contest-style scoring metrics incorporating official penalties for incorrect answers. Confidence estimates and paired statistical tests are introduced in the subsequent sections to support robustness of comparisons. The analysis concludes with a summary of the key empirical patterns that motivate the qualitative error analysis.

5.1. Overall Performance Across Models

This section provides a high-level comparison of model performance on the Austrian Kangaroo benchmark (2022–2024). All the models are evaluated under two modalities (text-only and visual-text) and two prompting conditions (*one_liner* and *full*). The benchmark consists of 324 items in total, of which 89 are text-only and 235 are visual-text (Table 5). We assess the model performance using accuracy, with abstentions (label N) counted as incorrect for accuracy computation and reported separately as an abstention rate. In Table 5, accuracy is computed over all the evaluated items; abstentions (N) are treated as non-correct outcomes and contribute to the difference between accuracy and 1.0 alongside incorrect answers.

Table 5. Overall accuracy by model, modality, and prompt condition, denoted as $\text{Acc}_{\text{prompt condition}}^{\text{Modality}}$ with abstention rates denoted as N.

Model (Short)	$\text{Acc}_{\text{one_liner}}^{\text{Text}}$	$\text{Acc}_{\text{full}}^{\text{Text}}$	$\text{Acc}_{\text{one_liner}}^{\text{Visual}}$	$\text{Acc}_{\text{full}}^{\text{Visual}}$
gpt-4o	0.27 (N = 0.05)	0.57 (N = 0)	0.25 (N = 0)	0.36 (N = 0.04)
qwen2p5-vl	0.28 (N = 0.1)	0.56 (N = 0.1)	0.32 (N = 0.02)	0.38 (N = 0.04)
qwen3-vl	0.44 (N = 0.05)	0.88 (N = 0.06)	0.41 (N = 0)	0.59 (N = 0.01)
pixtral-large	0.16 (N = 0.44)	0.51 (N = 0.16)	0.23 (N = 0.08)	0.3 (N = 0.12)
pixtral-latest	0.17 (N = 0.44)	0.54 (N = 0.16)	0.21 (N = 0.08)	0.29 (N = 0.1)

Several consistent patterns emerge from Table 5. First, qwen3-vl-235b-a22b is the strongest overall model across all the conditions, achieving the highest accuracy in both modalities and prompting regimes. Its best performance is observed on the text-only subset under full prompting, where it reaches 0.88 accuracy, substantially outperforming all the other models.

Second, scaling within the Qwen family is strongly associated with performance gains. qwen3-vl-235b-a22b consistently outperforms qwen2p5-vl-32b across modalities and prompting conditions, indicating a clear benefit of increased model capacity under this evaluation protocol. The substantial performance gap between qwen2.5-vl-32b and qwen3-vl-235b-a22b suggests that capabilities within this model family improve markedly across generations. However, this difference should not be interpreted as a pure scaling effect. The two models differ not only in parameter count but also in training data, post-training procedures, multimodal alignment, and potentially visual-processing components. Consequently, the present benchmark can establish a strong within-family performance improvement, but it cannot determine which architectural or training changes contribute most to the observed gains.

Third, prompting format has a consistent effect across all the models. The full condition yields higher accuracy than one_liner in all cases, although the magnitude of improvement varies by modality. On text-only items, the gains are substantial, with accuracies increasing from the 0.16–0.44 range under one_liner to 0.51–0.88 under full. The largest absolute improvement is observed for qwen3-vl-235b-a22b, which increases from 0.44 to 0.88. On visual-text items, the improvements are smaller but still consistent (e.g., qwen3-vl-235b-a22b: 0.41 $\rightarrow$ 0.59; gpt-4o: 0.25 $\rightarrow$ 0.36). These results suggest that structured reasoning with an explicit final answer improves task performance, particularly in text-only settings.

Fourth, modality effects indicate that visual-text items are generally more challenging under the full condition. Every model exhibits a decrease in accuracy when moving from text-only to visual-text evaluation in this setting (e.g., gpt-4o: 0.57 $\rightarrow$ 0.36; qwen3-vl-235b-a22b: 0.88 $\rightarrow$ 0.59). This pattern is consistent with the additional cognitive load required for diagram interpretation. In contrast, the modality differences under one_liner are smaller and occasionally non-monotonic. However, because text-only and visual-text subsets are disjoint, these differences are descriptive rather than causal; formal comparisons are provided in the dedicated modality analysis.

Finally, abstention behavior varies substantially across models and interacts with prompting format. The pixtral models exhibit high abstention rates under one_liner on text-only items (N = 0.44), which corresponds to their comparatively low accuracy in this condition. Under full prompting, abstention decreases substantially (to 0.16 on text-only), coinciding with improved accuracy. qwen2p5-vl-32b shows moderate but persistent abstention across conditions (e.g., 0.10 on text-only), while gpt-4o exhibits no abstentions on text-only items under full prompting. These patterns highlight abstention as an important secondary indicator of model reliability beyond raw accuracy.

Overall, across models and conditions, three main findings emerge from Table 5. First, full prompting consistently improves performance relative to one_liner, with particularly large gains on text-only items. Second, visual-text items are systematically more challenging under full prompting, leading to lower accuracy across all the models. Third, qwen3-vl-235b-a22b is the strongest overall system, and differences in abstention behavior reveal meaningful variation in response reliability across model families.

The following sections isolate these effects more precisely using paired comparisons for prompting conditions and structured analyses of modality-specific performance differences.

5.2. Effect of Prompt Condition

This section analyzes whether allowing free-form reasoning with an explicit final answer marker (full) improves multiple-choice performance relative to a strict single-token output constraint (one_liner). The comparison isolates the effect of output format while holding task content constant: selecting one of five answer options (A–E), with optional abstention (N). Prompt condition is therefore treated as a primary experimental factor as it directly influences both reasoning traces and response selection behavior.

For each model, the prompt-condition effect is evaluated separately on the text-only subset ($n = 89$) and the visual-text subset ($n = 235$). Within each subset, comparisons are paired at the item level using shared question identifiers, with each model answering the same item under both prompting conditions. Performance is measured using accuracy (proportion of correct predictions), with abstentions (N) treated as incorrect. The effect size is reported as

$$\Delta\text{Acc} = \text{Acc}_{\text{full}} - \text{Acc}_{\text{one_liner}}.$$

Uncertainty is estimated using paired bootstrap resampling over items (95% confidence intervals), and statistical significance is assessed using McNemar’s test on paired correctness outcomes (Table 6).

Table 6. Effect of prompt format (one_liner vs. full) on accuracy (Acc) by model and modality (paired McNemar test; bootstrap 0.95 CI for ΔAcc).

Model (Short)	Modality	Acc _{one_liner}	Acc _{full}	ΔAcc	0.95 CI for ΔAcc	<i>p</i> -Value
gpt-4o	text	0.270	0.573	0.303	[0.169; 0.438]	0.000
gpt-4o	visual	0.247	0.357	0.111	[0.03; 0.191]	0.010
qwen2p5-vl	text	0.281	0.562	0.281	[0.146; 0.416]	0.000
qwen2p5-vl	visual	0.315	0.383	0.068	[−0.009; 0.145]	0.101
qwen3-vl	text	0.438	0.876	0.438	[0.337; 0.551]	0.000
qwen3-vl	visual	0.409	0.587	0.179	[0.094; 0.264]	0.000
pixtral-large	text	0.157	0.506	0.348	[0.225; 0.472]	0.000
pixtral-large	visual	0.230	0.302	0.072	[0; 0.145]	0.071
pixtral-latest	text	0.169	0.539	0.371	[0.247; 0.494]	0.000
pixtral-latest	visual	0.209	0.285	0.077	[0.009; 0.149]	0.041

5.2.1. Text-Only Subset

The text-only subset exhibits a strong and highly consistent effect of prompt condition across all the models. Switching from one_liner to full yields large absolute improvements in accuracy, with all the differences statistically significant under McNemar’s test ($p < 0.001$). Effect sizes range from $\Delta\text{Acc} = 0.2809$ (qwen2p5-vl-32b) and $\Delta\text{Acc} = 0.3034$ (gpt-4o) to $\Delta\text{Acc} = 0.3708$ (pixtral-large-latest) and $\Delta\text{Acc} = 0.4382$ (qwen3-vl-235b-a22b).

The bootstrap confidence intervals for all the models lie well above zero, indicating robust improvements rather than effects driven by a small subset of items (e.g., qwen3-vl-235b-a22b: [0.3371, 0.5506]; gpt-4o: [0.1685, 0.4382]).

Collectively, these results indicate that, for text-only problems, enabling structured reasoning with an explicit final decision substantially increases accuracy across all the evaluated models.

5.2.2. Visual-Text Subset

On the visual-text subset, the full condition also improves performance across all the models, but the effect sizes are smaller and exhibit greater variation in statistical strength. The largest gain is observed for qwen3-vl-235b-a22b ($\Delta\text{Acc} = 0.1787$, 95% CI $[0.0936, 0.2638]$, $p < 0.001$).

In contrast, the weakest effect is observed for qwen2p5-vl-32b, where the improvement is small and not statistically significant under McNemar's test ($\Delta\text{Acc} = 0.0681$, 95% CI $[-0.0085, 0.1447]$, $p = 0.101$). For the remaining models, the improvements are positive but smaller than those observed in the text-only setting.

Overall, the visual-text results indicate a generally positive but attenuated effect of the full prompting condition, with weaker and less uniform statistical support compared to text-only items.

5.2.3. Interpretation

These findings suggest that the full prompting condition primarily improves performance by encouraging more deliberate reasoning and reducing shallow or heuristic answer selection. This effect is strongest in text-based problems, where the task is fully specified in text and requires symbolic or logical reasoning.

In contrast, visual-text problems introduce an additional constraint. Successful performance depends not only on reasoning but also on accurate extraction of relevant information from diagrams, grids, or spatial configurations. When errors stem from perceptual or interpretative failures, additional reasoning scaffolding is less effective, leading to smaller and sometimes non-significant gains.

This interpretation is consistent with the subsequent modality-focused analysis. The qualitative error analysis in Section 5.5 further examines these effects, distinguishing between visual misinterpretation errors and downstream reasoning failures.

In summary, prompt condition is a major determinant of performance in this benchmark. The full condition yields large statistically significant improvements on text-only items across all the models. It yields smaller, but generally positive, improvements on visual-text items. These results demonstrate that output format and reasoning scaffolding materially affect measured accuracy and should be treated as a core dimension of model evaluation in this setting.

5.3. Text-Only vs. Visual Performance

This section analyzes the effect of input modality by comparing performance on text-only items to visual-text items, where the latter require interpretation of an accompanying diagram or figure. Visual-text problems introduce an additional cognitive requirement beyond symbolic reasoning as models must extract relevant constraints from visual input and integrate them with the textual prompt. The modality effect is therefore evaluated separately within each model and prompting condition.

Descriptively, Table 6 indicates a systematic interaction between modality and prompting. Under the full condition, accuracy is consistently higher on text-only items than on visual-text items across all the models, indicating that diagram-based problems remain more difficult even when models are encouraged to reason explicitly. Under one_liner prompting, however, the modality gap is substantially reduced and in some cases reverses direction. This suggests that strict output constraints alter response behavior (e.g., in-

creased heuristic selection or abstention), thereby partially masking underlying differences in task difficulty.

Figure 2 presents a dumbbell plot illustrating accuracy differences between text-only and visual-text modalities across models. Each point corresponds to the mean accuracy over items in the respective subset, and the connecting line visualizes the modality gap. While the shift from `one_liner` to `full` improves performance across both modalities, the magnitude of improvement is systematically larger for text-only items. This asymmetry suggests that the `full` condition primarily enhances reasoning and decision formation, whereas visual-text items remain constrained by an additional bottleneck in visual grounding and diagram interpretation. In other words, improved reasoning benefits both modalities but does not eliminate modality-specific perceptual limitations.

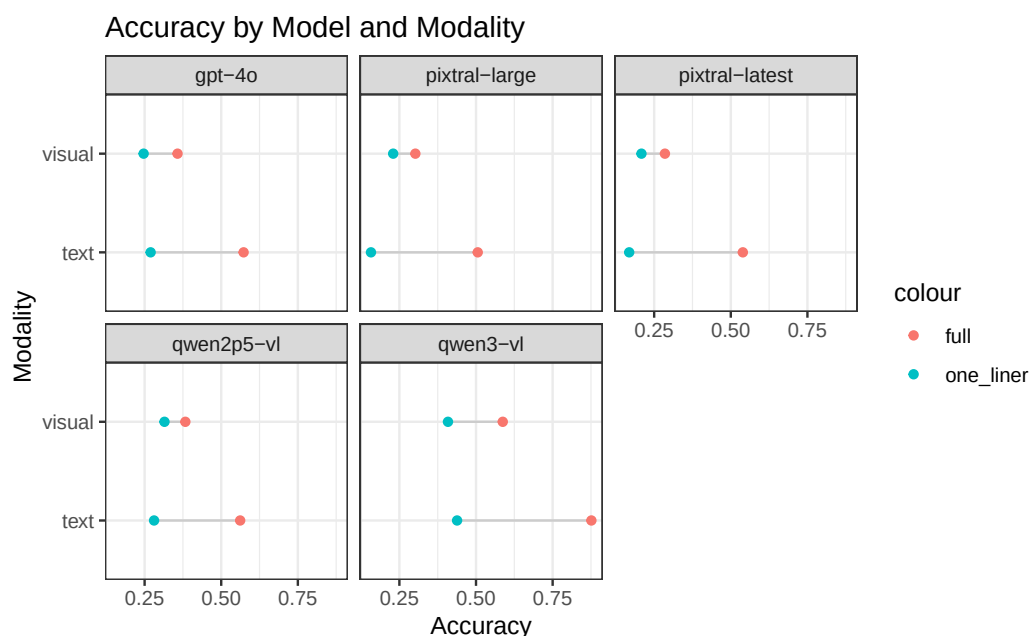


Figure 2. Prompt-condition effect (`one_liner` vs. `full`): accuracy by model and modality.

To quantify these effects, modality differences are tested within each model and prompting condition using chi-square tests of independence alongside bootstrap estimates of the accuracy difference, defined as

$$\Delta \text{Acc}^{\text{Mod}} = \text{Acc}^{\text{Visual}} - \text{Acc}^{\text{Text}}.$$

Because text-only and visual-text subsets contain disjoint items ($n_{\text{text}} = 89, n_{\text{visual}} = 235$), these comparisons are unpaired and interpreted as differences between item distributions rather than item-level effects. Bootstrap confidence intervals are computed at the 95% level.

The inferential results in Table 7 align closely with the descriptive patterns. Under the `full` condition, all the models exhibit a substantial negative modality gap, with visual-text accuracy lower than text-only accuracy by $\Delta \text{Acc}^{\text{Mod}} \approx 0.18$ to $\Delta \text{Acc}^{\text{Mod}} \approx 0.29$. All the corresponding bootstrap confidence intervals lie entirely below zero, and chi-square tests indicate statistically significant associations between modality and correctness across models.

Table 7. Effect of question modality (text vs. visual) on accuracy (Acc) by model and prompt format (unpaired chi-squared test; bootstrap 0.95 CI for $\Delta\text{Acc}^{\text{Mod}}$).

Model (Short)	Prompt	Acc ^{Text}	Acc ^{Visual}	$\Delta\text{Acc}^{\text{Mod}}$	0.95 CI for $\Delta\text{Acc}^{\text{Mod}}$	p-Value
gpt-4o	full	0.573	0.357	−0.216	[−0.335; −0.093]	0.000
gpt-4o	one_liner	0.270	0.247	−0.023	[−0.133; 0.084]	0.673
qwen2p5-vl	full	0.562	0.383	−0.179	[−0.302; −0.056]	0.004
qwen2p5-vl	one_liner	0.281	0.315	0.034	[−0.076; 0.14]	0.553
qwen3-vl	full	0.876	0.587	−0.289	[−0.382; −0.193]	0.000
qwen3-vl	one_liner	0.438	0.409	−0.030	[−0.152; 0.093]	0.628
pixtral-large	full	0.506	0.302	−0.203	[−0.323; −0.083]	0.001
pixtral-large	one_liner	0.157	0.230	0.072	[−0.022; 0.161]	0.153
pixtral-latest	full	0.539	0.285	−0.254	[−0.371; −0.136]	0.000
pixtral-latest	one_liner	0.169	0.209	0.040	[−0.056; 0.133]	0.420

Under the *one_liner* condition, the modality differences are small and inconsistent in sign, with confidence intervals overlapping zero in all cases. Correspondingly, chi-square tests do not indicate statistically reliable modality effects. This indicates that the modality gap becomes both larger and more statistically robust under structured reasoning conditions, while it is effectively attenuated under strict output constraints.

Overall, these results suggest that *one_liner* prompting suppresses performance sufficiently in the text-only condition to partially obscure modality differences, whereas *full* prompting restores a clearer separation between modalities by improving reasoning performance without resolving visual grounding limitations. This interpretation is consistent with the prompt-condition analysis in the previous section.

The next section examines whether these modality effects vary systematically across competition year, level, and point tier.

5.4. Summary of Results

This chapter evaluated five LLM/VLM systems on the Austrian Kangaroo benchmark (2022–2024), consisting of 324 multiple-choice items (89 text-only and 235 visual-text) under two prompting conditions (*one_liner* and *full*). Performance is reported primarily as accuracy, with abstentions (N) treated as incorrect and reported separately (Table 5). Inferential analyses include paired bootstrap confidence intervals and McNemar’s test for prompt-condition effects and chi-square tests with unpaired bootstrap estimates for modality comparisons.

Across all the conditions, qwen3-vl-235b-a22b is the strongest-performing model, clearly separating from the rest of the systems (Table 5). gpt-4o and qwen2p5-vl-32b form a mid-tier group, with relative ranking depending on modality and prompting condition. The pixtral models exhibit lower raw accuracy overall, which is strongly influenced by their abstention behavior: under *one_liner*, they abstain substantially more frequently than other models, mechanically reducing accuracy.

Conditioned on answering (i.e., excluding abstentions), both pixtral variants perform comparably to or slightly above gpt-4o on the text-only subset in both prompting conditions, indicating that part of their apparent underperformance is attributable to response strategy rather than purely reasoning ability.

Prompt format is a dominant driver of performance: *full* prompting consistently improves accuracy relative to *one_liner*, with especially large gains on text-only items. This effect is statistically supported by paired bootstrap estimates and McNemar tests and is consistent with the prompt-condition analysis in Section 5. In contrast, visual-text

items remain systematically more difficult than text-only items under full prompting, with robust modality gaps confirmed by unpaired statistical tests. This indicates that diagram grounding introduces a persistent bottleneck that is not eliminated by improved reasoning scaffolding.

Overall, the results suggest two dominant and complementary sources of error: (i) reasoning and decision-structure failures, which are substantially reduced under full prompting, and (ii) visual grounding and diagram interpretation failures, which persist across models and prompting regimes. Section 5.5 provides a qualitative analysis of these failure modes and relates them to observable patterns such as abstention behavior and systematic answer biases.

5.5. Error Analysis

While aggregate accuracy quantifies overall performance, it does not explain why models fail on this benchmark. This section therefore provides a structured error analysis at two levels. First, it presents a qualitative categorization of failure modes based on manual inspection of a stratified sample of incorrect predictions. Second, it analyzes systematic patterns in incorrect multiple-choice selections to characterize biases in answer choice behavior. Together, these analyses distinguish failures arising from reasoning and constraint handling from those due to visual grounding and diagram interpretation, providing context for the reliability differences observed in previous sections.

5.5.1. Error Taxonomy

This analysis is based on model outputs generated by the evaluation pipeline described in Section 4. A prediction is considered incorrect if the parsed final answer differs from the gold label (A–E). Abstentions (label N) are treated as a separate outcome class and are excluded from analyses conditioned on incorrect option selection.

To characterize failure mechanisms, we manually inspect a stratified sample of 30 incorrect cases drawn across models and modalities (text-only and visual-text). Each sample includes the full input (question text, answer options, and image where applicable), the raw model output, and the parsed prediction. The analysis assigns exactly one primary error category per case, corresponding to the dominant factor responsible for the incorrect answer. When multiple issues are present, the annotation reflects the most consequential failure mode. This single-label scheme ensures comparability across categories and avoids double counting.

The following taxonomy is used throughout this section:

- *constraint_miss*: The model fails to correctly incorporate a key constraint from the problem statement (e.g., “at least,” “exactly,” boundary conditions, or exclusions), resulting in a logically coherent but invalid solution.
- *strategy_error*: The model selects an inappropriate or incomplete solution strategy (e.g., applying an unsuitable heuristic, omitting necessary case distinctions, or choosing a method misaligned with the problem structure).
- *diagram_misinterpretation*: The model constructs a correct reasoning process based on an incorrect interpretation of the diagram or visual structure (e.g., spatial relations, adjacency, directionality, or counting elements in a figure).
- *arithmetic_error*: The model follows a correct high-level strategy but makes a low-level computational mistake (e.g., arithmetic operations, algebraic simplification, or counting errors) that changes the final answer.
- *visual_mistake*: The model fails in processing the visual input without a specific misinterpretation of structure (e.g., missing relevant visual elements, ignoring annotations, or failing to integrate the image into reasoning).

- *input_parsing_error*: The model exhibits confusion about the multiple-choice format (e.g., mixing option text into reasoning or producing responses that are inconsistent with the provided options) despite correct dataset-level parsing and extraction.

Each item is labeled with a single dominant error category by three independent expert annotators. Inter-annotator agreement measured by Fleiss' κ (Conger [35]) was 0.29, indicating fair agreement. This is consistent with the fine-grained and partially subjective nature of the error taxonomy (especially distinctions between constraint- and strategy-related errors). Final labels were obtained via majority vote, with remaining ties resolved through expert consensus. The analysis is intended as qualitative characterization of failure modes rather than statistical inference over error frequencies.

5.5.2. Error Distribution

Table 8 summarizes the distribution of error types over the inspected sample ($n = 30$). The dominant failure mode is *constraint_miss*, indicating that a substantial fraction of incorrect answers arise not from inability to perform multi-step reasoning but from incomplete incorporation of problem constraints. In many cases, models follow a plausible solution path but omit a critical condition that invalidates the result.

The second most frequent category is *strategy_error*, where models commit early to an unsuitable solution approach and fail to recover even when intermediate steps indicate mismatch with the problem structure. This suggests that errors are often driven by early-stage methodological decisions rather than local computational failure.

Table 8. Distribution of manually coded error categories in the qualitative sample ($n = 30$).

Category	Proportion
<i>constraint_miss</i>	0.500
<i>strategy_error</i>	0.200
<i>diagram_misinterpretation</i>	0.167
<i>arithmetic_error</i>	0.067
<i>visual_mistake</i>	0.033
<i>input_parsing_error</i>	0.033

In the visual-text subset, *diagram_misinterpretation* appears as a persistent and distinct source of error, reflecting a modality-specific bottleneck. In these cases, reasoning may be internally consistent, but it is grounded in an incorrect interpretation of spatial or structural visual information.

By contrast, *arithmetic_error* is relatively rare, indicating that purely computational mistakes contribute less to overall failure than higher-level issues in constraint handling, strategy selection, or visual grounding. *visual_mistake* and *input_parsing_error* occur infrequently and primarily reflect peripheral failures in perception or format adherence rather than core reasoning limitations.

5.6. Representative Failure Cases

To complement the aggregate error distribution, this subsection presents representative failure cases that illustrate how dominant error modes manifest at the instance level. The objective is not exhaustive coverage of all observed mistakes but rather the construction of interpretable exemplars that clarify the mechanisms underlying the primary failure categories identified in the manual analysis. For each case, we provide the original problem statement (as presented to the model), the model's verbatim response, and a concise diagnosis linking the observed behavior to the corresponding error class.

Example 1

The following example illustrates a case categorized as *constraint_miss*, where the model misinterprets the core task objective.

Problem statement (Source: Känguru der Mathematik, <https://www.kaenguru.at/aufgaben.html>, accessed 12 November 2025): “Beate arranges the five cards so that the smallest nine-digit number is created. Which card is furthest on the right? (a) 4 (b) 8 (c) 31 (d) 59 (e) 107”.

The task requires arranging five cards to form the smallest possible nine-digit number and then identifying which card appears at the far right. The model instead interprets the problem as a local ordering task over individual numeric values rather than a global concatenation problem over digit strings. As a result, it optimizes an incorrect objective and applies an inappropriate heuristic.

Model response:

“To create the smallest nine-digit number using the cards, we should arrange the numbers in ascending order based on their value. The cards are:

– 4 (A) – 8 (B) – 31 (C) – 59 (D) – 107 (E)

Arranging these in ascending order gives us: 4, 8, 31, 59, 107. Therefore, the card furthest on the right is 107, which corresponds to option E.

FINAL: E”

This is a *constraint_miss* because the model fails to incorporate the key requirement that the numbers must be concatenated to form a single nine-digit integer before comparison. Under the correct interpretation, the optimal construction is (107314598), implying that the rightmost card is 8 (option B).

The error is characteristic of a broader pattern in which models rely on surface-level lexical cues (e.g., “smallest number”) and default to sorting heuristics without modeling the compositional structure required by the task specification.

Example 2

The following example illustrates a *strategy_error*, where the model adopts an unduly restricted solution strategy that ignores a permitted operation.

Problem statement ((Source: Känguru der Mathematik, <https://www.kaenguru.at/aufgaben.html>, accessed 12 November 2025)): “Pieter has a parcel that weights 445 g and the following eight weights: 500, 200, 200, 100, 50, 20, 20, 5. He places the parcel on the right scale pan (see picture [omitted]). Pieter is allowed to place weights on either side of the scale. What is the minimum number of weights that he needs to balance the scale? (a) 2 (b) 3 (c) 4 (d) 6”.

In this problem, the task involves balancing a scale using a set of available weights, with the explicit allowance that weights may be placed on either side of the scale. The optimal solution requires exploiting this flexibility. However, the model implicitly assumes that all the weights must be placed on the same side as the parcel, thereby solving a more constrained variant of the problem than the one specified.

Model response:

“To balance the scale, Pieter needs to place weights on the left side to match the weight of the parcel (445 g) on the right side. We need to find the minimum number of weights that can sum up to 445 g using the given weights: 500 g, 200 g, 200 g, 100 g, 50 g, 20 g, 20 g, and 5 g.

1. Start with the largest weights to minimize the number of weights used.
2. Use the 200 g weight: $445 - 200 = 245$ g remaining.

3. Use another 200 g weight: $245 - 200 = 45$ g remaining.
4. Use the 50 g weight: $45 - 50 = -5$ g (we overshoot, so we need to adjust).
5. Use the 5 g weight to correct the overshoot: 0 g remaining.

Thus, the weights used are: 200 g, 200 g, 50 g, and 5 g. This totals 4 weights.

FINAL: C''

This is best classified as a *strategy_error* since the failure arises from an unnecessarily restrictive solution space: the model ignores the possibility of placing weights on both pans, which is essential for obtaining the optimal solution.

The correct solution exploits this symmetry: placing 500 g on one side and 50 g + 5 g on the other side yields a balanced configuration with the parcel, requiring only three weights in total (answer B).

While the model's computation contains secondary inconsistencies (e.g., overshoot correction), the decisive failure is the initial strategic assumption that reduces the problem to a one-sided subset-sum formulation rather than a two-sided balance problem.

To further investigate visual failures, we refine the qualitative error analysis by focusing specifically on visual-text items. Within the manually annotated sample, visual-related errors are dominated by diagram misinterpretations. Diagram misinterpretation occurs when the model correctly follows a reasoning strategy but operates on an incorrect interpretation of spatial relations, such as relative positioning, symmetry, or connectivity in geometric or grid-based figures.

Example 3

To illustrate typical diagram-related failure modes, we present representative cases of diagram misinterpretation errors. The diagrams shown in Figure 3 are author-created reconstructions based on the corresponding benchmark items and are intended to preserve the visual structure relevant to the model outputs discussed below. In each case, correct interpretation of the diagram is essential for solving the problem.

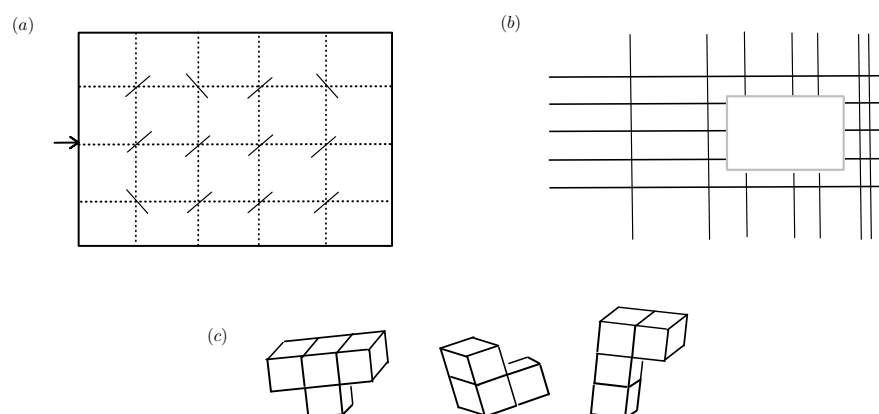


Figure 3. Representative examples of diagram misinterpretation errors: (a) incorrect interpretation of mirror orientations, (b) errors in an irregular grid-completion task, and (c) misperception of 3D figures. The diagrams are author-created reconstructions based on Austrian Mathematical Kangaroo problems and are included for illustrative and analytical purposes.

- Model response to diagram processing shown in panel (a) of Figure 3: “[...] The grid has 3 rows and 4 columns of cells. Each cell contains a diagonal mirror. Let’s label the cells by row and column:
 - Row 1 (top): mirrors at (1,1), (1,2), (1,3), (1,4)—all are ‘/’ (as shown in the image)
 - Row 2 (middle): mirrors at (2,1), (2,2), (2,3), (2,4)—all are ‘\’ (as shown)

- Row 3 (bottom): mirrors at (3,1), (3,2), (3,3), (3,4)—all are ‘/’ [...]

In this case, the model incorrectly assigns consistent orientations to mirrored elements across the grid despite heterogeneous orientations in the diagram. This indicates a systematic misreading of local visual features, leading to an incorrect global structure and final answer.

- Model response to diagram processing shown in panel (b) of Figure 3: “[...] The missing section in the grid is a 2×2 square, which means it has:
 - 2 horizontal lines at the top and bottom.
 - 2 vertical lines on the left and right. [...]

Here, the model incorrectly infers a regular 2×2 structure where the diagram contains irregular spatial constraints. This reflects a failure to accurately parse the true grid configuration, resulting in an invalid counting and matching.

- Model response to diagram processing shown in panel (c) of Figure 3: “We are given three distinct 3D parts made of unit cubes [...]. First, let’s analyze the three given parts:
 1. Part 1: Looks like a 2×2 base with one cube sticking out on the side—total 5 cubes.
 - Structure: Imagine a 2×2 square (4 cubes), and one cube attached to the side of one of the cubes in the square. So it’s like a “T” or “L” shape extended.
 2. Part 2: Looks like a $2 \times 2 \times 2$ cube missing one corner—total 7 cubes.
 - Structure: 3 cubes in a row, with 2 stacked on top of the first two—forming a kind of “stair” or “corner” shape.
 3. Part 3: Looks like a 3×1 column with one cube sticking out on the side—total 4 cubes.
 - Structure: 3 cubes stacked vertically, and one cube attached to the side of the middle cube.

Wait—let’s count the cubes in each part more carefully. Actually, looking at the image:

- Part 1: 4 cubes in a row, with one cube attached below the second cube → total 5 cubes.
- Part 2: 2 cubes side by side, with 2 cubes stacked on top of the left one → total 4 cubes.
- Part 3: 3 cubes stacked vertically, with one cube attached to the side of the top cube → total 4 cubes. [...]

In this example, the model produces a plausible verbal decomposition of the 3D objects but repeatedly miscounts or misidentifies cube arrangements. Even after self-correction attempts, the inferred structures remain inconsistent with the visual input, indicating difficulty in reliably grounding 3D spatial configurations.

Overall, these examples indicate that spatial and structural details in diagrams are not consistently and reliably extracted, even when the model produces fluent and internally coherent reasoning. This prevents correct downstream reasoning since it is conditioned on an inaccurate visual representation.

These findings suggest that performance degradation on visual-text items is primarily driven by limitations in visual grounding rather than arithmetic or symbolic reasoning. In particular, the main bottleneck appears to lie in the accurate extraction and integration of diagrammatic information into the reasoning process.

In summary, these examples illustrate several recurring and structurally distinct failure mechanisms. The first involves constraint integration failures, where models misread or partially ignore structural requirements that define the problem’s solution space. The second involves strategy restriction failures, where models prematurely commit to an

incomplete or overly constrained solution framework, excluding valid operations that are necessary for optimal solutions. The third involves visual grounding failures, where models misinterpret or incompletely extract information from diagrams, leading to downstream reasoning based on incorrect spatial or structural representations. This includes errors in reading geometric relations, miscounting or misidentifying elements in grids, and misconstructing 3D structures from visual input.

Taken together, these patterns indicate that errors are not primarily random or computational in nature but instead arise from systematic breakdowns in either problem specification handling, strategy selection, or visual information integration.

6. Discussion and Conclusions

This study presents a controlled evaluation of five large language and vision–language models on Austrian Mathematical Kangaroo problems (2022–2024; 324 items). The design systematically varies input modality (text-only vs. visual-text) and prompting format (*one_liner* vs. *full*) while holding decoding and answer-extraction constant. This enables a controlled analysis of how evaluation protocol and multimodal input jointly shape measured performance.

Across experiments, three consistent effects emerge. First, prompt format is the strongest driver of performance. Full prompting consistently improves accuracy over *one_liner* across all the models, with especially large gains on text-only items. This indicates that benchmark outcomes are highly sensitive to inference protocol and cannot be interpreted solely as reflections of underlying model capability. Rather, performance emerges from the interaction between model behavior and output constraints, which can either facilitate or suppress reasoning. Second, visual-text problems constitute a persistent and robust difficulty regime. All the models show a substantial drop in accuracy on visual-text items under full prompting, and gains from full prompting are systematically smaller in this modality. This suggests that, while structured prompting improves symbolic reasoning, it does not resolve failures in visual grounding and diagram interpretation, which remain a distinct bottleneck in multimodal reasoning. Third, abstention behavior meaningfully affects reported accuracy. Models differ in their tendency to abstain under strict output constraints, directly influencing accuracy when abstentions are counted as incorrect. This introduces a trade-off between decisiveness and correctness and highlights the importance of reporting both overall accuracy and accuracy conditional on answering.

Taken together, these results show that benchmark scores are jointly determined by reasoning capability, visual grounding, and evaluation protocol. Structured prompting improves performance primarily by enabling more complete reasoning processes but does not eliminate errors arising from misinterpretation of visual inputs. At the same time, strict output formats systematically alter response behavior, including answer-selection patterns and abstention frequency, thereby shaping aggregate results.

An important implication is that benchmark scores should not be interpreted as direct measurements of intrinsic reasoning ability. The substantial differences observed between the *one_liner* and *full* conditions demonstrate that model rankings and performance estimates can depend materially on prompting choices, output constraints, and answer-extraction procedures. Consequently, benchmark design should explicitly account for prompt sensitivity and protocol robustness. Reporting results across multiple prompting conditions may provide a more informative and reliable characterization of model behavior than a single benchmark score. From this perspective, prompt sensitivity is itself an important dimension of model reliability and should be considered when comparing systems and interpreting benchmark results.

Error analysis reinforces this interpretation. Failures are structured rather than random, dominated by constraint violations, premature strategy commitment, and diagram misinterpretation. These patterns indicate that errors arise from both incomplete problem specification handling and limitations in visual integration rather than low-level computational inability. The qualitative error analysis is based on a small stratified sample ($n = 30$), which limits statistical generalization. The error analysis should be interpreted as characterizing representative failure modes rather than population-level error frequencies.

Several further limitations apply. The dataset is restricted to Austrian competition mathematics problems (2022–2024), limiting generalizability. Accuracy aggregates heterogeneous failure modes into a single metric and is not psychometrically calibrated. In multimodal settings, differences in image preprocessing and model interfaces introduce potential variability. Additionally, prior exposure to similar competition problems during pretraining cannot be excluded.

The findings suggest several implications for evaluation and modeling. First, the evaluation protocol should be treated as a primary factor in model assessment as small changes in prompting can substantially shift results. Second, multimodal reasoning failures are primarily driven by visual grounding rather than symbolic reasoning, indicating a need for improved diagram interpretation mechanisms. Finally, abstention behavior should be explicitly modeled as it introduces a fundamental trade-off between accuracy and coverage that is not captured by standard metrics.

In conclusion, this study demonstrates that performance on structured mathematical benchmarks is jointly determined by reasoning ability, visual grounding, and evaluation protocol. While modern models perform strongly under supportive prompting, their behavior remains highly sensitive to format constraints and degrades consistently in visual reasoning settings. The main contribution is a reproducible evaluation framework that makes these dependencies explicit, emphasizing that benchmark accuracy should be interpreted as a conditional measure shaped by both model and protocol rather than a standalone capability estimate.

Author Contributions: Conceptualization, G.G.; methodology, G.G.; software, G.G.; validation, M.O. and T.K.; formal analysis, G.G. and M.O.; investigation, G.G.; data curation, G.G., M.O. and T.K.; writing—original draft preparation, G.G., M.O. and T.K.; writing—review and editing, M.O. and T.K.; visualization, M.O.; supervision, M.O. and T.K.; project administration, M.O. and T.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The Python code and the data for conducting the analysis are available at <https://github.com/GokanGorer/kangaroo-math-model-eval>, accessed on 12 March 2026.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; Steinhardt, J. Measuring Mathematical Problem Solving With the MATH Dataset. In Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks, Online, 6–14 December 2021; Volume 1.
2. Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.W.; Galley, M.; Gao, J. MATHVISTA: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts. *arXiv* **2024**, arXiv:2310.02255. [[CrossRef](#)]
3. Mondorf, P.; Plank, B. Beyond Accuracy: Evaluating the Reasoning Behavior of Large Language Models—A Survey. *arXiv* **2024**, arXiv:2404.01869. [[CrossRef](#)]
4. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.

5. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models Are Few-Shot Learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS '20)*; Curran Associates, Inc.: Red Hook, NY, USA, 2020; pp. 1877–1901.
6. Hoffmann, J.; Borgeaud, S.; Mensch, A.; Buchatskaya, E.; Cai, T.; Rutherford, E.; de Las Casas, D.; Hendricks, L.A.; Welbl, J.; Clark, A.; et al. Training Compute-Optimal Large Language Models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)*; Curran Associates, Inc.: Red Hook, NY, USA, 2022; pp. 30016–30030.
7. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [\[CrossRef\]](#)
8. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2015; Volume 28.
9. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929. [\[CrossRef\]](#)
10. Li, J.; Li, D.; Savarese, S.; Hoi, S. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*; PMLR: New York, NY, USA, 2023; pp. 19730–19742.
11. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*; PMLR: New York, NY, USA, 2021; pp. 8748–8763.
12. AI, O. CLIP: Connecting Text and Images. 2022. Available online: <https://openai.com/index/clip/> (accessed on 12 March 2026).
13. Chen, Y.C.; Li, L.; Yu, L.; Kholy, A.E.; Ahmed, F.; Gan, Z.; Cheng, Y.; Liu, J. UNITER: UNiversal Image-TExt Representation Learning. *arXiv* **2020**, arXiv:1909.11740. [\[CrossRef\]](#)
14. Wei, X.; Yang, G.; Zhou, J.; Yang, M.; Li, L.; Zhang, K.; Qiu, C. Dynamic Embedding of Hierarchical Visual Features for Efficient Vision-Language Fine-Tuning. *arXiv* **2025**, arXiv:2508.17638. [\[CrossRef\]](#)
15. Alayrac, J.B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. Flamingo: A Visual Language Model for Few-Shot Learning. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)*; Curran Associates, Inc.: Red Hook, NY, USA, 2022; pp. 23716–23736.
16. Nagrani, A.; Yang, S.; Arnab, A.; Jansen, A.; Schmid, C.; Sun, C. Attention Bottlenecks for Multimodal Fusion. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2021; Volume 34, pp. 14200–14213.
17. Zhuo, J.; Zhang, S.; Fang, X.; Duan, H.; Lin, D.; Chen, K. ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024*; Al-Onaizan, Y., Bansal, M., Chen, Y.N., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 1950–1976. [\[CrossRef\]](#)
18. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.H.; Le, Q.V.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)*; Curran Associates, Inc.: Red Hook, NY, USA, 2022; pp. 24824–24837.
19. Fang, M.; Wan, X.; Lu, F.; Xing, F.; Zou, K. MathOdyssey: Benchmarking Mathematical Problem-Solving Skills in Large Language Models Using Odyssey Math Data. *Sci. Data* **2025**, *12*, 1392. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. Training Verifiers to Solve Math Word Problems. *arXiv* **2021**, arXiv:2110.14168. [\[CrossRef\]](#)
21. Patel, A.; Bhattamishra, S.; Goyal, N. Are NLP Models Really Able to Solve Simple Math Word Problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., Zhou, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 2080–2094. [\[CrossRef\]](#)
22. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large Language Models Are Zero-Shot Reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)*; Curran Associates, Inc.: Red Hook, NY, USA, 2022; pp. 22199–22213.
23. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; Zhou, D. Self-Consistency Improves Chain of Thought Reasoning in Language Models. *arXiv* **2023**, arXiv:2203.11171. [\[CrossRef\]](#)
24. Tam, Z.R.; Wu, C.K.; Tsai, Y.L.; Lin, C.Y.; Lee, H.y.; Chen, Y.N. Let Me Speak Freely? A Study On The Impact Of Format Restrictions On Large Language Model Performance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*; Derroncourt, F., Preotiuc-Pietro, D., Shimorina, A., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 1218–1236. [\[CrossRef\]](#)
25. OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; et al. GPT-4 Technical Report. *arXiv* **2024**, arXiv:2303.08774. [\[CrossRef\]](#)

26. Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. Qwen2.5-VL Technical Report. *arXiv* **2025**, arXiv:2502.13923. [[CrossRef](#)]
27. Bai, S.; Cai, Y.; Chen, R.; Chen, K.; Chen, X.; Cheng, Z.; Deng, L.; Ding, W.; Gao, C.; Ge, C.; et al. Qwen3-VL Technical Report. *arXiv* **2025**, arXiv:2511.21631. [[CrossRef](#)]
28. AI, M. Pixtral Large | Mistral AI. 2024. Available online: <https://docs.mistral.ai/capabilities/vision> (accessed on 12 March 2026).
29. Card, D.; Henderson, P.; Khandelwal, U.; Jia, R.; Mahowald, K.; Jurafsky, D. With Little Power Comes Great Responsibility. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Webber, B., Cohn, T., He, Y., Liu, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 9263–9274. [[CrossRef](#)]
30. Efron, B.; Tibshirani, R.J. *An Introduction to the Bootstrap*; Chapman and Hall/CRC: New York, NY, USA, 1994. [[CrossRef](#)]
31. Hesterberg, T.C. What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum. *Am. Stat.* **2015**, *69*, 371–386. [[CrossRef](#)] [[PubMed](#)]
32. Xu, J.; Harrar, S.W. Accurate Mean Comparisons for Paired Samples with Missing Data: An Application to a Smoking-Cessation Trial. *Biometr. J. Biom. Z.* **2012**, *54*, 281–295. [[CrossRef](#)] [[PubMed](#)]
33. Dietterich, T.G. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Comput.* **1998**, *10*, 1895–1923. [[CrossRef](#)] [[PubMed](#)]
34. Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed.; Routledge: New York, NY, USA, 2013. [[CrossRef](#)]
35. Conger, A.J. Integration and Generalization of Kappas for Multiple Raters. *Psychol. Bull.* **1980**, *88*, 322–328. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.