Beyond Understanding: Evaluating the Pragmatic Gap in LLMs' Cultural Processing of Figurative Language

Mena Attia[1,2]   Aashiq Muhamed[1]   Mai Alkhamissi[1]
Thamar Solorio[2]   Mona Diab[1]

[1]Carnegie Mellon University   [2]MBZUAI
mena.attia@mbzuai.ac.ae, mdiab@andrew.cmu.edu

Abstract

We present a comprehensive evaluation of the ability of large language models (LLMs) to process culturally grounded language, specifically to understand and pragmatically use figurative expressions that encode local knowledge and cultural nuance. Using figurative language as a proxy for cultural competence and local knowledge, we design evaluation tasks for contextual understanding, pragmatic use, and interpretation of connotations in Arabic and English. We evaluate 22 open- and closed-source LLMs on Egyptian Arabic idioms, multidialectal Arabic proverbs, and English proverbs. Our results show a consistent hierarchy: the average accuracy for Arabic proverbs is 4.29% lower than for English proverbs, and the performance for Egyptian idioms is 10.28% lower than for Arabic proverbs. For the pragmatic use task, the accuracy decreases by 14.07% relative to understanding, although providing contextual idiomatic sentences improves the accuracy by 10.66%. Models also struggle with connotative meaning, reaching at most 85.58% agreement with human annotators on idioms with 100% inter-annotator agreement. These findings demonstrate that figurative language serves as an effective diagnostic for cultural reasoning: while LLMs can often interpret figurative meaning, they face challenges in using it appropriately. To support future research, we release *Kinayat*,[1] the first dataset of Egyptian Arabic idioms designed for both figurative understanding and pragmatic use evaluation.

1   Introduction

Large language models (LLMs) have demonstrated remarkable progress in multilingual text understanding and generation. However, their ability to process *culturally grounded meaning*—how language encodes local knowledge, social values, and pragmatic nuance—remains poorly understood.

Proverb	ما تروحش تبيع المية في حارة السقايين
Literal Translation	Don't sell water in the village of water-sellers
Figurative Meaning	Don't try to outsmart the experts.
Historical Context	Historically, water-sellers in Egypt lived together in villages and brought water from the Nile to sell across neighborhoods. New sellers would mistakenly try to sell water in the water-sellers' village, failing to realize the abundance of water there made it an unsuitable market.
Cultural Significance	Featured in the popular song "Haret El-Sa'ayeen" (1966) by Hussein Al-Sayed, sung by Sherifa Fadel, later by Mohammed Mounir with a performance on Arab Idol (2013) surpassing 132 million views on YouTube as of August 2025 (MBC, 2013).

Table 1: Sample Egyptian proverb: water-sellers village.

Figurative language offers a natural lens for studying this capability. Idioms and proverbs are among the most prevalent forms of figurative expression, deeply rooted in collective experience and cultural identity. They rely on shared world knowledge and pragmatic reasoning that extend beyond literal semantics. For example, Table 1 presents an Egyptian proverb. Without contextual knowledge, the reference to the village of water-sellers offers little clue to its intended message or its appropriate use. For models to truly understand language, they must internalize this cultural substrate rather than rely solely on surface statistics or translation mappings.

Most prior studies have focused on figurative comprehension—whether a model can explain an idiom or proverb—but not on pragmatic use, which requires context sensitivity, affective inference, and social appropriateness. In this work, we address these gaps by introducing a comprehensive evaluation framework that uses figurative language as a proxy for cultural competence and knowledge. We design tasks that test models' abilities to (1) interpret figurative expressions, (2) use them appropriately in context, and (3) infer their connotative

---

[1]https://huggingface.co/datasets/menaattia/Kinayat

and affective dimensions. Our evaluation covers 22 open-source and closed-source models spanning multiple architectures and parameter scales. Although we leverage existing resources such as Jawaher (Arabic) and MAPS (English) for proverb interpretation, our central contribution is the introduction of **Kinayat**, a new dataset of Egyptian Arabic idioms annotated for both figurative meaning and pragmatic use. Our work makes the following key contributions:

1. We introduce a *pragmatic use task* to assess the ability of LLMs to employ figurative language appropriately in context;

2. We present a unified cross-lingual evaluation suite that examines figurative interpretation, contextual appropriateness, and connotative inference;

3. We present *Kinayat*, a novel dataset of Egyptian Arabic idioms to evaluate figurative language understanding and pragmatic use.

Our results demonstrate that performance consistently degrades from English proverbs to Arabic proverbs and finally to colloquial idioms, highlighting a systemic weakness in handling culturally-specific figurative language. We find that performance in Arabic proverbs is on average 4.29% lower than in English proverbs, indicating a gap in language performance. Within Arabic, models perform 10.28% worse on idioms from our Kinayat dataset (Egyptian dialect) compared to Modern Standard Arabic (MSA) proverbs.[2] In our novel pragmatic use task, we uncover a significant 'Pragmatics Gap' in LLMs. Our core finding shows that model accuracy drops by an average of 14 percentage points when tasked with applying an idiom in context compared to simply explaining its meaning. That is, the average accuracy is 14.07% lower than in the corresponding understanding task, with a maximum accuracy of 85.33%, suggesting that the use of idioms appropriately in context remains difficult for current models. However, providing the idiom sentence as an additional context in the multiple-choice understanding prompt improves the average precision by 10.66%. Finally, models exhibit notable limitations in grasping connotative meaning, achieving at most 85.58% agreement with humans on samples with 100% inter-annotator agreement.

---

[2]MSA is the standard Arabic used in formal settings, while Egyptian dialect is a spoken vernacular, a related variant of MSA, used in Egypt.

2 Related Work

Recent years have seen a surge of interest in evaluating and improving the ability of LLMs to interpret and generate figurative language. Several benchmarks have been proposed to assess various dimensions of figurative understanding, including metaphor, idiom, proverb, and poetry interpretation. Table 2 provides a comparative summary of these datasets.

Several English-focused datasets have been developed to evaluate literal versus figurative reasoning. The Fig-QA dataset (Liu et al., 2022) frames figurative language understanding as a multiple-choice question answering task. Chakrabarty et al. (2022a) evaluate the ability of LLM to interpret idioms and similes plausibly by continuing narratives. Other efforts include FLUTE (Chakrabarty et al., 2022b) and follow-up work that explores metaphor interpretation using chain-of-thought prompting and psychologically informed reasoning (He et al., 2022; Prystawski et al., 2023; Jang et al., 2023), showing that while LLMs can often identify meaning at the surface level, they frequently fail to capture implicit moral or social connotations. The PUB benchmark (Sravanthi et al., 2024) specifically evaluates pragmatic competence, testing the model's ability to distinguish between literal and contextually appropriate figurative meanings in conversation.

In the multilingual setting, the MAPS dataset (Liu et al., 2024) evaluates figurative understanding in six languages of proverbs, and the ProverbEval dataset (Azime et al., 2025) evaluates LLMs of cultural proverbs for four Ethiopian languages and English, while MABL (Kabra et al., 2023) provides a multilingual benchmark for metaphor and simile comprehension in underrepresented languages.

Figurative language is an integral component of culture, yet Arabic language understanding and cultural benchmarks often omit it entirely. For example, widely used Arabic cultural benchmarks such as AraDiCE (Mousi et al., 2025), CAMEL-Bench (Ghaboura et al., 2025), and CIDAR (Alyafeai et al., 2024) do not include figurative language as an explicit category. Others incorporate it only in limited capacity, such as the ArabCulture dataset (Sadallah et al., 2025), which covers 13 countries and 12 topics with idioms as one topic, but includes only five samples per country, and PALM (Alwajih et al., 2025), which covers 20 diverse topics from 22 Arab countries with proverbs as one of the

Dataset	Languages	Figurative Type
MAPS (Liu et al., 2024)	English, Chinese, German, Russian, Bengali, Indonesian	Proverbs
MABL (Kabra et al., 2023)	Hindi, Indonesian, Javanese, Kannada, Sundanese, Swahili, Yoruba	Metaphors
Fig-QA (Liu et al., 2022)	English	Metaphors
FLUTE (Chakrabarty et al., 2022b)	English	Sarcasm, Simile, Metaphor, and Idioms
PUB (Sravanthi et al., 2024)	English	Implied answers, Presuppositions, Metonymy
Jawaher (Magdy et al., 2025)	Arabic (20 dialects)	Proverbs
Fann or Flop (Alghallabi et al., 2025)	Arabic	Poetry
Kinayat (ours)	Arabic	Idioms

Table 2: Comparison of existing figurative language datasets.

topics.

There are two recent benchmarks that focus specifically on Arabic figurative language: the Jawaher dataset (Magdy et al., 2025), the first large-scale collection of Arabic proverbs in 20 dialects, and Fann or Flop (Alghallabi et al., 2025), a benchmark for the interpretation of Arabic poetry that spans multiple genres and eras. However, the lack of datasets that focus on figurative language highlights the need for more comprehensive evaluation resources that integrate figurative language as an explicit component of cultural understanding.

Recent work has also examined the affective and social dimensions of figurative expressions. Martínez et al. (2024) use LLMs to estimate the valence, arousal, and concreteness of multi-word expressions, offering a route to investigate deeper semantic features. In the Arabic context, Alsiyat and Piao (2020) demonstrate that metaphorical constructions significantly affect sentiment analysis performance, underscoring the need for a connotative understanding in Arabic Natural Language Processing (NLP).

3 Methodology

To provide a comprehensive evaluation of the figurative language, we design a framework of tasks that probe LLM capabilities beyond simple comprehension in Arabic idioms, Arabic proverbs, and English proverbs. Although prior work has benchmarked proverb understanding, our framework introduces a suite of tasks, including negation, contextual reasoning, pragmatic use, and connotation labeling, that have not been systematically applied to the Arabic figurative language before.

Pragmatic Use: We introduce a new task to evaluate the *pragmatic use* of Arabic idioms in LLMs. Using a subset of 150 idioms from the Kinayat dataset, we use a model-in-the-loop approach (Chakrabarty et al., 2022b; Liu et al., 2024), prompting GPT-4.1 (OpenAI, 2025) with each idiom and its explanation to generate a sentence in

Egyptian Arabic, using the prompt shown in Figure 13.

A native Egyptian Arabic speaker then reviews the generated sentences and either accepts them, makes minor changes, or comes up with new sentences to replace them. Of the 150 samples, 77 (51.3%) are modified, either by minor dialectal adjustments, rephrasing, the addition of feminine examples, or complete rewriting of the sentence. The unmodified generated samples appear in first person, first person plural, second person, or third person masculine. Many examples are related to work, with the word شغل "work" appearing 29 times and مدير "manager" appearing 7 times.

As part of the task setup, for each idiom, the annotator selects a plausible but incorrect idiom from the Kinayat set to serve as a distractor in a multiple-choice setting. We then evaluate LLMs, asking them to choose the correct idiom for a given sentence, using the template in Figure 14. Representative examples of this task are shown in Figure 1.

We evaluate the performance of LLMs on the following tasks using zero-shot prompting, with a single evaluation run per experiment. All prompts are included in the appendix A. All prompts are in English to maintain methodological consistency in both English and Arabic datasets in our evaluation, allowing for direct comparison without introducing confounding factors from prompt language variation, and following previous work in multilingual LLM evaluation (Liu et al., 2024; Lin et al., 2022; Muennighoff et al., 2023).

Multiple Choice Question (MCQ) Understanding: To evaluate figurative comprehension, we construct multiple choice questions, inspired by the work of Liu et al. (2024); Kabra et al. (2023), where each item presents two candidate explanations for a given idiom or proverb, one correct and one incorrect.

Because understanding is assessed through a multiple-choice format, the results are inherently

Figure 1: Sample questions from the Pragmatic Use task, along with their English translations.

influenced by the quality and difficulty of incorrect explanations. When an incorrect explanation is implausible or obviously wrong, models can often eliminate it without genuinely understanding the meaning of the idiom or proverb. To try to mitigate this, incorrect explanations are generated using GPT-4.1 with two distinct strategies. First, we use a general prompt template (Figure 7) to produce a semantically plausible but incorrect explanation for each idiom or proverb. Second, we leverage semantic role labeling (SRL) with LLMs (Cheng et al., 2024) to generate more subtly incorrect explanations: we extract the semantic roles from the gold explanation and modify a single role to alter the meaning, using the SRL-based prompt in Figure 8.

Each model is prompted to choose the correct explanation in two separate evaluation settings: once with the general incorrect explanations, and once with the SRL-based alternatives.

Human Verification of Incorrect Distractors
To evaluate the quality of the generated explanations, a native Arabic speaker manually assesses a random sample of 50 incorrect explanations for idioms from Kinayat and another 50 for proverbs from Jawaher. Each explanation is rated on a three-point scale from 0 to 2, where higher scores indicate explanations that are incorrect, but linguisti-

cally and culturally plausible, while lower scores correspond to explanations that are implausible or irrelevant. Detailed annotation guidelines are provided in the appendix B. The average verification scores are 1.72 for idioms and 1.68 for proverbs, confirming that all the generations sampled were indeed incorrect, although they vary in clarity and plausibility. This slight difference suggests that it is slightly easier to produce plausible but incorrect explanations for idioms than for proverbs, although this likely reflects differences in the gold explanations provided as input rather than in the idioms or proverbs themselves.

Contextual Understanding MCQ: We extend the MCQ Understanding Task by providing idiomatic sentences for a subset of Arabic idioms from Kinayat as additional context. Models are then prompted to select the correct explanation given both the idiom and its usage in context, using the prompt shown in Figure 6.

Negation: Building on the work of Liu et al. (2024), we adapt the MCQ understanding task in Arabic datasets by requiring the models to select the *incorrect* explanation rather than the correct one. This inversion introduces a negation component to the task, as models must reject the correct option, which prior work in natural language inference has shown to degrade model performance (Truong et al., 2023; She et al., 2023).

Explanation Generation: Beyond evaluating models' ability to identify the correct explanation, we also assess their capability to *generate* explanations for Arabic idioms and proverbs. Models are prompted to produce explanations using the prompt template shown in Figure 11.

Completing the Proverb: We evaluate the memorization of cultural knowledge of the models by masking the final word of each English and Arabic proverb, following Liu et al. (2024), and prompting the model to complete it using the prompt template shown in Figure 9.

Connotation Understanding: Connotations of each Arabic proverb and idiom are labeled by three native Egyptian Arabic speakers as *positive*, *negative*, or *neutral*. Human annotators are provided with the same prompt template as the models to standardize task instructions. We then evaluate models only on samples with 100% agreement to minimize the impact of connotation subjectivity on

model performance. We use two variants of the task employing the prompt template in Figure 15: (1) predicting the connotation given the proverb or idiom, and (2) predicting the connotation given its explanation. Model predictions are compared with human annotations to calculate accuracy.

4 Experimental Setup

4.1 Datasets

Jawaher (Magdy et al., 2025): We use 198 test samples of multidialectal proverbs across 20 varieties and their Arabic explanations.

MulticulturAl Proverbs and Sayings (MAPS) (Liu et al., 2024): A multilingual dataset of proverbs in 6 languages. We only use the English test set which consists of 394 proverbs.

Kinayat: A Dataset of Egyptian Arabic Colloquial Idioms We introduce the Kinayat dataset, which consists of 325 Egyptian idioms along with their MSA explanations. We extracted them from the book **Al-Kinayat Al-'Amiyya** (Pasha, 1949).[3] Some examples are shown in Table 19 in Appendix F.

Data Preprocessing: During preprocessing, we removed inappropriate or incomplete idioms, resulting in the exclusion of 10 entries from the dataset. Citations appearing in parentheses within the text were deleted to ensure clarity and consistency. We also removed phonetic explanations, as well as similar idioms and sayings that were mentioned with the explanations. Examples and excerpts of poetry were excluded to maintain a focus on the core figurative expressions. Finally, for idioms whose explanations merely referred to an equivalent idiom without further clarification, we supplemented the data by adding in full explanatory text.

4.2 Models

Our experiments consider a broad selection of LLMs, encompassing both open-source and closed-source options, as well as Arabic and non-Arabic models of varying size. We evaluate a total of 22 models.

The open-source multilingual models evaluated include LLaMA 3.1 (8B, 70B-Instruct) (Grattafiori et al., 2024), Gemma 2 (9B, 27B-IT) (Team et al., 2024b), Qwen-2.5 (7B, 14B, 32B-Instruct) (Qwen et al., 2025), Aya-Expanse (8B, 32B) (Dang et al.,

2024), and Mistral-7B-Instruct-v0.3 (Jiang et al., 2023). We further evaluate a diverse set of open-source Arabic models, such as Jais-family-6p7b-chat, Jais-family-13b-chat (Sengupta et al., 2023), ALLaM-7B-Instruct-preview (Bari et al., 2025), SILMA-9B-Instruct-v1.0 (silma-ai, 2024), Fanar-1-9B-Instruct (Team et al., 2025), and AceGPT-v2-8B-Chat (Huang et al., 2024).

The closed-source models included in our evaluation are GPT-4o and 4o-mini (OpenAI et al., 2024), Gemini 1.5 Flash (Team et al., 2024a), Gemini 2.5 Flash Lite Preview (Comanici et al., 2025), as well as Claude 3.5 Sonnet (Anthropic, 2024) and Sonnet 4 (Anthropic, 2025).

4.3 Evaluation

We evaluate LLM performance across the different tasks using the `lm-eval` framework (Gao et al., 2024), which computes accuracy using log-likelihood for open-source models and model-generated outputs for closed-source APIs. For open-source models, `lm-eval` was configured with vLLM (Kwon et al., 2023) and tensor parallelism to distribute compute across 4 GPUs. Our code and evaluation scripts are publicly available on GitHub.[4]

For multiple-choice tasks, we report the mean *accuracy $\pm$ standard error* as the primary evaluation metrics. For generation-based tasks, we employ two complementary evaluation methods:

1. **BERTScore** (Zhang et al., 2020): Measures semantic similarity between model-generated explanations and gold references using contextual embeddings from `bert-base-multilingual-cased`, with the language set to Arabic (`lang="ar"`). We report the F1 score as the main evaluation metric.

2. **LLM-as-a-Judge** (Zheng et al., 2023): Uses a LLM to assess how well the generated explanation aligns with the intended meaning of the gold explanation. We use GPT-4.1 as the default system to score all models except for the evaluation of the GPT models, which are instead judged using Claude-3.5-Sonnet to avoid self-evaluation bias (Wataoka et al., 2024). The scoring prompt is shown in Figure 12.

3. **Human Evaluation:** We also conduct human evaluation on a subset of the generations to

---

[3]The book is publicly available by the publisher.

validate the reliability of the LLM-as-a-Judge scores.

5 Results

5.1 Knowledge and Understanding

Distinguishing Between Correct and Incorrect explanations In Figure 2, we observe a clear difficulty hierarchy in understanding MCQ, with the highest model performance on MAPS with context (95.66%), followed by MAPS (90.86%), Jawaher (86.57%) and lowest on Kinayat (76.29%). Arabic proverbs consistently yield lower performance than English proverbs, both with context (95.66% vs. 86.57%) and without context (90.86% vs. 86.57%), indicating a persistent gap in cross-lingual and cultural understanding. One contributing factor may be differences in context lengths; English proverbs exhibit a higher mean sentence-length distribution than Arabic proverbs (6.52 vs. 5.21), providing richer lexical and syntactic context that can help models. Furthermore, Arabic idioms are more challenging than Arabic proverbs (average accuracy of 76.29% vs. 86.57%), possibly because proverbs tend to be almost frozen making them easier to memorize as they could have been seen in the training data as opposed to idioms. Additionally, idioms occur in shorter and less informative contexts, as shown by their length distributions in Figure 17 (Appendix C). Detailed results for the MCQ task and subsequent tasks are presented in Appendix D.

Table 7 presents the results for the negation variant of the MCQ task. When models were asked to identify the *incorrect* explanation rather than the correct one. This inversion led to a notable performance drop: for idioms, the average accuracy decreased from 76.29% to 70.97%, and for proverbs from 86.57% to 82.71%. Interestingly, GPT-4o outperformed the Claude models in this more challenging negation task, despite the Claude models achieving higher accuracy in the original MCQ understanding task.

Knowledge and Memorization of Proverbs Results for the proverb completion task are presented in Table 8. Claude 3.5 achieved the highest accuracy for both English and Arabic proverbs, with scores of 93.91% and 36.36%, respectively. On average, the performance of the model was substantially higher for English proverbs (75.43%) compared to Arabic proverbs (10.65%). The low accuracy of Arabic completions suggests limited memorization, which may reflect a lower represen-

tation of Arabic proverbs in the models' training data. This gap in completion accuracy mirrors the performance differences observed in the proverb understanding tasks, suggesting that greater exposure to English proverbs during training may provide models with stronger lexical and phrasal cues that facilitate both memorization and understanding. Despite low memorization, models perform well in the understanding task, suggesting that they can still reason effectively without relying on memorization (Liu et al., 2024).

Model Ability to Generate Explanations Claude 3.5 Sonnet achieved the highest scores in both evaluation metrics, BERTScore-F1 and LLM-as-a-Judge, for both proverbs and idioms. For proverbs, it scored 0.70 on BERTScore-F1 and 3.89 on the LLM-as-a-Judge scale (1–5). For idioms, it scored 0.68 and 2.93, respectively. Although higher BERTScore-F1 values did not always correspond to higher LLM-as-a-Judge scores, the highest BERTScore-F1 outputs were consistently aligned with the highest human-aligned ratings. Overall, idioms proved to be more challenging for models to generate explanations for, with lower average scores across both metrics. Specifically, the average BERTScore-F1 and LLM-as-a-Judge scores were 0.65 and 2.19 for idioms, compared to 0.68 and 3.06 for proverbs.

LLM-as-a-Judge Verification To validate the reliability of the LLM-as-a-Judge scoring procedure, we include human evaluation on a representative subset of 220 generated explanations, corresponding to 10 explanations per model across the 22 evaluated models. This subset was sampled to cover a range of scores and idioms, enabling assessment of whether the automated judgments align with human judgments of correctness and plausibility. The average LLM-as-a-Judge score on this subset was 2.15, compared to an average human score of 1.94, indicating slight overestimation by the automated evaluator (0.21 points). In terms of agreement, human and LLM-as-a-Judge scores matched exactly in 62.27% of cases; in 29.09% of cases, the human score was lower than the LLM score, while in 8.64% of cases, the human score was higher. Nevertheless, both evaluations consistently reflect low explanation quality overall. These results demonstrate that the LLM-as-a-Judge scores are broadly consistent with human evaluation, supporting their use at scale in our analysis.
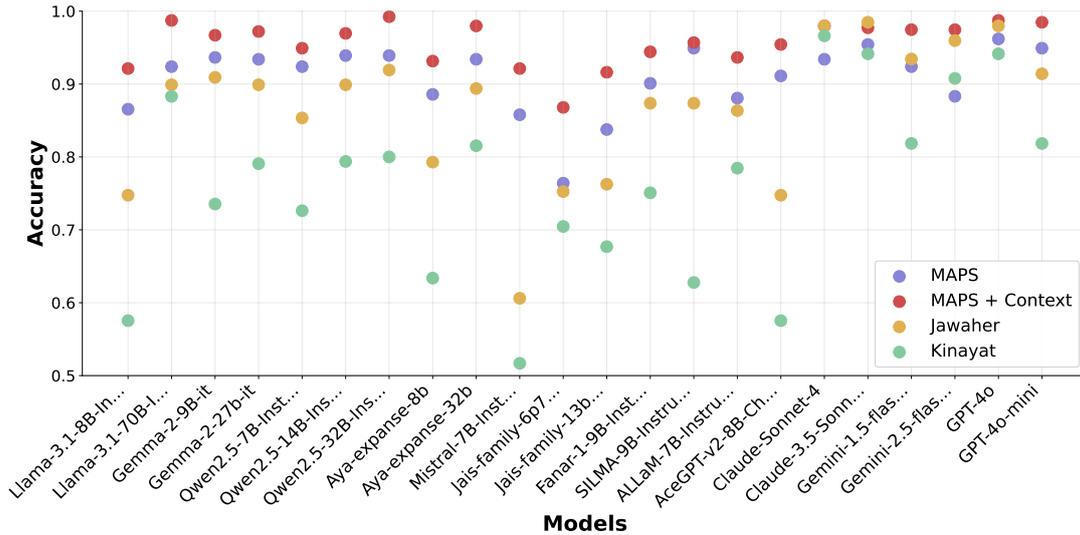
Figure 2: Accuracy on MCQ Understanding task across different test sets.

5.2 Pragmatic Use

As shown in Table 3, the highest accuracy of 85.33% on the pragmatic use task was achieved by Claude 3.5 Sonnet. The average accuracy on pragmatic use across all models was 64.45%, which is 14.07% lower than the average accuracy on the understanding task for the same set of 150 idioms. When the sentence containing the idiom was added as context to the MCQ understanding prompt, the average accuracy increased by 10.66% to 89.18%. These results demonstrate a performance gap between understanding and pragmatic use even for frontier models, indicating that using idioms pragmatically in context is more challenging than choosing the correct explanation from multiple choice options.
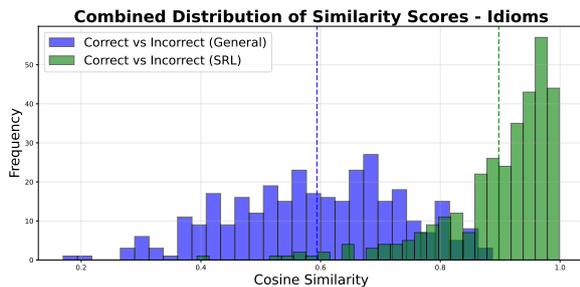
Figure 3: Cosine similarity (↑) between the correct and incorrect explanation choices for idioms (Kinayat).

5.3 Connotations

Annotating connotation in Arabic figurative expressions reveals inherent subjectivities in cultural expressions. Assigning clear sentiment values is often reductive, as interpretation varies widely across individuals and groups (Sap et al., 2022; Aroyo and Welty, 2015). Out of the 198 multidialectal proverbs, 105 (53.03%) achieved full inter-

annotator agreement on their connotations, while 104 out of 150 Egyptian idioms (69.3%) reached full agreement, underscoring the inherent subjectivity of connotation judgments. To reduce the confounding effect of subjectivity of connotations, models were evaluated only on the samples where all three annotators agreed, ensuring a focus on stronger connotations. Claude-3.5-Sonnet achieved the highest accuracy in labeling the connotations of idioms (85.58% accuracy) and proverbs (74.04% accuracy), while Claude-Sonnet-4 achieved the highest accuracy in labeling the connotations of explanations, with accuracy scores of 89.42% for idioms and 86.54% for proverbs. On average, accuracy of explanations was higher by 22.11% for idioms and 18.46% for proverbs, which is expected given that the purpose of the explanations is to clarify figurative expressions. These results suggest that models struggle with connotations, which could be related to how they perform less on the pragmatic use task. All connotation results are presented in Table 12 (Appendix D).

6 Ablation and Linguistic Variation Analysis

Variants of Incorrect Distractors To better understand the semantic relationship between correct and incorrect explanations, we compute the cosine similarity between their sentence embeddings (using the `paraphrase-multilingual-mpnet-base-v2` model (Reimers and Gurevych, 2019))—comparing correct vs. incorrect explanations generated using the general prompt (Figure 7), and correct vs. incorrect explanations using the SRL prompt

7244

| Model	Pragmatic Use	Understanding	Contextual Understanding
Llama-3.1-8B-Instruct	$0.5400^{\pm0.0408}$	$0.6000^{\pm0.0401}$	$0.7867^{\pm0.0336}$
Llama-3.1-70B-Instruct	$0.6333^{\pm0.0395}$	$0.8867^{\pm0.0260}$	$0.9667^{\pm0.0147}$
Gemma-2-9B-it	$0.6000^{\pm0.0401}$	$0.7867^{\pm0.0336}$	$0.8933^{\pm0.0253}$
Gemma-2-27b-it	$0.6933^{\pm0.0378}$	$0.8133^{\pm0.0319}$	$0.9533^{\pm0.0173}$
Qwen2.5-7B-Instruct	$0.5267^{\pm0.0409}$	$0.7733^{\pm0.0343}$	$0.8667^{\pm0.0278}$
Qwen2.5-14B-Instruct	$0.6467^{\pm0.0392}$	$0.8400^{\pm0.0300}$	$0.9667^{\pm0.0147}$
Qwen2.5-32B-Instruct	$0.7133^{\pm0.0370}$	$0.8200^{\pm0.0315}$	$0.9267^{\pm0.0214}$
Aya-expanse-8b	$0.5600^{\pm0.0407}$	$0.6533^{\pm0.0390}$	$0.8467^{\pm0.0295}$
Aya-expanse-32b	$0.7533^{\pm0.0353}$	$0.8467^{\pm0.0295}$	$0.9400^{\pm0.0195}$
Mistral-7B-Instruct-v0.3	$0.4400^{\pm0.0407}$	$0.4867^{\pm0.0409}$	$0.6267^{\pm0.0396}$
Jais-family-6p7b-chat	$0.5667^{\pm0.0406}$	$0.7067^{\pm0.0373}$	$0.8333^{\pm0.0305}$
Jais-family-13b-chat	$0.6067^{\pm0.0400}$	$0.7867^{\pm0.0336}$	$0.8133^{\pm0.0319}$
Fanar-1-9B-Instruct	$0.5800^{\pm0.0404}$	$0.7733^{\pm0.0343}$	$0.9000^{\pm0.0246}$
SILMA-9B-Instruct-v1.0	$0.5600^{\pm0.0407}$	$0.5933^{\pm0.0402}$	$0.8533^{\pm0.0290}$
ALLaM-7B-Instruct-preview	$0.6867^{\pm0.0380}$	$0.8533^{\pm0.0290}$	$0.9333^{\pm0.0204}$
AceGPT-v2-8B-Chat	$0.4667^{\pm0.0409}$	$0.5667^{\pm0.0406}$	$0.7400^{\pm0.0359}$
Claude-Sonnet-4	$0.8333^{\pm0.0305}$	$\mathbf{0.9533}^{\pm0.0173}$	$0.9733^{\pm0.0132}$
Claude-3.5-Sonnet	$\mathbf{0.8533}^{\pm0.0290}$	$\mathbf{0.9533}^{\pm0.0173}$	$0.9667^{\pm0.0147}$
Gemini-1.5-flash	$0.6533^{\pm0.0390}$	$0.8400^{\pm0.0300}$	$0.9333^{\pm0.0204}$
Gemini-2.5-flash-lite	$0.7800^{\pm0.0339}$	$0.9333^{\pm0.0204}$	$\mathbf{0.9867}^{\pm0.0094}$
GPT-4o	$0.8400^{\pm0.0300}$	$\mathbf{0.9533}^{\pm0.0173}$	$0.9667^{\pm0.0147}$
GPT-4o-mini	$0.6467^{\pm0.0392}$	$0.8533^{\pm0.0290}$	$0.9467^{\pm0.0184}$
Average	0.6445	0.7852	0.8918

Table 3: Evaluation results for Pragmatic Use, Understanding, and Understanding with Context on a subset of 150 sample idioms from the Kinayat dataset.

(Figure 8), separately for proverbs and idioms. The resulting similarity distributions, shown in Figure 3 for idioms (and Figure 18 for proverbs in Appendix C), reveal noticeable shifts in distribution, but the impact on overall performance was minor. For proverbs, average accuracy slightly decreased with SRL-based distractors (86.57% → 86.36%), while for idioms, performance slightly improved (76.29% → 78.52%). This suggests that the semantic closeness of distractors can vary depending on the generation strategy, but does not largely affect model performance. For the generation of incorrect distractors in the multiple-choice understanding task, the LLM (GPT-4.1) was prompted only with the correct explanation, not with the idiom or proverb itself. This prevents the model from using knowledge of the idiom/proverb in generating incorrect explanations. Moreover, GPT-4.1 was not among the models evaluated in the experiments. Importantly, despite distractors being generated by a GPT-family model, Claude models consistently outperformed GPT models on MCQ understanding for both idioms and proverbs, indicating that the use of GPT-4.1 did not advantage GPT models or introduce evaluation bias.

Dialectal Breakdown The dialectal breakdown for the MCQ understanding results in Table 10 for the Jawaher dataset is shown in Tables 14 and 15 (as well as Figures 20 and 21 in Appendix D. The average of the two tables is presented in Figure

4 and Table 16. The top performing dialects on average were: UAE, MSA, Libya, and Oman. The lowest performing dialects were: Sudan, Mauritania, Qatar, and Yemen. Notably, UAE, Libya, and Oman performed unexpectedly well despite being low-resource dialects, while Egypt, a high-resource dialect, achieved only a mid-range score. One possible explanation is that certain proverbs in the Jawaher dataset are culturally shared across multiple Arab countries, particularly Gulf and Levantine varieties, whereas others are highly localized or unique. Country-level performance varied depending on whether the incorrect distractor was generated with a general or SRL-based prompt, but UAE and MSA consistently ranked in the top four, while Sudan and Mauritania consistently ranked among the lowest three.
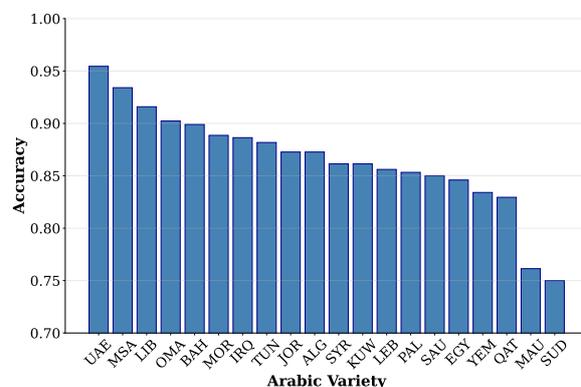
Figure 4: Country-level breakdown of MCQ Understanding Average Accuracy.

Arabic vs. Multilingual Models Table 13 in Appendix D summarizes the average performance of open source multilingual, open source Arabic, and closed source models in all tasks. Multilingual models generally outperform Arabic models in English tasks by an average of 4.72% and Arabic tasks by a smaller margin of 2.19%. However, this trend is not consistent across all multilingual models; for example, Mistral-7B-Instruct performed worse than the Arabic models on most tasks. Closed-source models achieve the highest performance overall, surpassing both Arabic and multilingual models, which may in part be due to their larger size, as their exact parameter counts are not disclosed but they are estimated to be larger than the open-source models evaluated, where the latter models range from 6.7B to 32B parameters. However, Arabic models show notable strengths, outperforming in completion and pragmatic use tasks and achieving slightly higher average scores in generation. The higher completion performance of Arabic models could be partly explained by their pretraining on Arabic text, which likely exposed them to more Arabic proverbs during training.

Model Size A weak positive correlation between model size and performance was observed for understanding tasks, with values of R^2 ranging from 0.189 to 0.424, as shown in Figure 23 and summarized in Table 17 (Appendix E). The most efficient model for the Arabic task, measured as performance per billion parameters, was ALLaM-7B-Instruct-preview, while the most efficient model for the English task was Qwen2.5-7B-Instruct. The pragmatic use task exhibited a stronger correlation with model size ($R^2 = 0.6$) compared to the MCQ understanding task ($R^2 = 0.265$), with ALLaM-7B-Instruct-preview again being the most efficient model.

6.1 Error Analysis

We conducted error analysis on the top eight most frequently incorrectly answered idioms across three tasks—*MCQ Understanding*, *MCQ Understanding with Context*, and *Pragmatic Use*—using a 150-sample subset of Kinayat (corresponding to Table 3). We report the number of models (out of 22) that failed on each idiom in Tables 20, 21, and 22 in Appendix G. Our analysis reveals that providing idiomatic context sentences substantially improves performance for certain idioms in the *MCQ Under-*

standing task. For instance, وَلَع shows dramatic improvement (error frequency: 9→1), and عَنْدُه الدُنيَا بِالْحَلْخَال improves notably (9→4). However, contextualization does not consistently benefit all idioms, as performance remains poor for خَبَر الصَباحْ رَبَاحْ, and سَكْرِةْ يَنِيْ, أبْيَض despite the addition of context. In contrast, error patterns for the *Pragmatic Use* task differ markedly from those observed in *MCQ Understanding*, indicating a dissociation between comprehension and pragmatic use capabilities. Models that correctly identify idiom meanings often fail to appropriately use those same idioms in context, highlighting the limitations in pragmatic competence discussed in Section 5.2.

7 Conclusion

We presented a comprehensive evaluation of large language models' ability to understand and pragmatically use figurative expressions that encode local knowledge and social nuance. We evaluated diverse LLMs on Egyptian Arabic idioms, multi-dialectal Arabic proverbs, and English proverbs across tasks assessing contextual understanding, pragmatic use, and connotation interpretation. Results revealed a consistent performance hierarchy, with English proverbs outperforming Arabic proverbs, and both outperforming Egyptian idioms. Pragmatic use emerged as significantly more challenging than understanding, and models struggled to capture connotative meaning appropriately. Figurative language thus serves as an effective diagnostic for cultural reasoning, revealing that while LLMs often interpret figurative meaning, they face major challenges in using it appropriately. To support future research, we released *Kinayat*, the first dataset of Egyptian Arabic idioms designed for both figurative understanding and pragmatic use evaluation. Future work should extend pragmatic evaluation to proverbs and idioms in multiple dialects, and assess pragmatic use in free-form generation to better capture the ability of models to produce culturally and contextually appropriate figurative expressions.

Limitations

Scope Although our study provides new insights into the ability of large language models to understand and use Egyptian Arabic idioms and proverbs, it is important to acknowledge several limitations. First, our benchmark focuses exclusively on Egyptian idioms. Although Egyptian Arabic is a high-resource dialect with significant representation online and in available datasets, the observed challenges that models face in the pragmatic use of figurative language underscore the necessity for similar resources and benchmarks in other Arabic dialects, many of which are lower-resource and may present even greater challenges. Additionally, since the source used for our idiom collection was published in 1949, some idioms may be outdated as language evolves over time, and idiom usage varies across generations and social groups. However, this does not necessarily render the idioms obsolete: many remain in active use among particular speaker communities. That said, we took steps to mitigate this concern in our experiments. Specifically, our pragmatic evaluation was conducted on a subset of 150 more familiar idioms out of 325, ensuring that the core findings are not driven by highly unfamiliar expressions.

Second, our evaluation is limited to idioms and proverbs, which represent only a subset of figurative language phenomena. Figurative language encompasses a broader range of expressions, including metaphors, similes, hyperboles, irony, sarcasm, and other culturally specific figures of speech. Future work should extend the evaluation to these additional forms to provide a more comprehensive assessment of model capabilities in the understanding and generation of Arabic figurative language.

Connotations Limitations The connotations task also carries assumptions and limitations: the human labeling of idioms and proverbs can be somewhat subjective, particularly when the connotation is weak or context-dependent.

This ambiguity reflects broader challenges in NLP annotation, where guidelines may fail to account for the multiplicity of cultural perspectives, leading to mismatches between annotator intent and dataset construction (Liu et al., 2025). Research has repeatedly shown that annotator bias—shaped by personal and cultural background—can skew labeled data, embedding specific worldviews into machine learning systems and perpetuating representational inequalities (Sap et al., 2022; Bender et al., 2021; Aroyo and Welty, 2015). Attempts to enforce consistency through guidelines and inter-annotator agreement often overlook that language is deeply contextual and values are not universally shared (Bender et al., 2021).

Anthropological critiques urge moving beyond the pursuit of an "objective" gold standard in annotation, advocating instead for multi-layered or perspectivist models that document the diversity of meanings and cultural nuances present in a corpus (Bender et al., 2021; Aroyo and Welty, 2015). Struggling with annotation from an anthropological standpoint underscores the importance of recognizing cultural context, annotator subjectivity, and the limits of value assignment in NLP, serving as a reminder that data is more than numbers: it is a reflection of lived sociocultural realities (Liu et al., 2025; Aroyo and Welty, 2015).

Ethical Considerations

Offensive Content Elimination The complete list of idioms in the Al-Kinayat Al-'Amiyya (Pasha, 1949) book was manually reviewed and seven samples were removed that we considered inappropriate or offensive.

Licenses The Kinayat dataset is released under a CC-BY 4.0 license[5], permitting use, distribution, and adaptation with attribution.

Annotations Two of the annotators are coauthors of this paper, while the third annotator was compensated for their time at a rate of 50 USD for an estimated three hours of work, which is slightly above the minimum wage in the US. No personal information from annotators is included in the Kinayat dataset.

References

Wafa Alghallabi, Ritesh Thawkar, Sara Ghaboura, Ketan More, Omkar Thawakar, Hisham Cholakkal, Salman Khan, and Rao Muhammad Anwer. 2025. Fann or flop: A multigenre, multiera benchmark for arabic poetry understanding in llms. *Preprint*, arXiv:2505.18152.

Israa Alsiyat and Scott Piao. 2020. Metaphorical expressions in automatic Arabic sentiment analysis. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4911–4916, Marseille, France. European Language Resources Association.

---

[5] https://creativecommons.org/licenses/by/4.0/

Fakhraddin Alwajih, Abdellah El Mekki, Samar Mohamed Magdy, AbdelRahim A. Elmadany, Omer Nacar, El Moatez Billah Nagoudi, Reem Abdel-Salam, Hanin Atwany, Youssef Nafea, Abdulfattah Mohammed Yahya, Rahaf Alhamouri, Hamzah A. Alsayadi, Hiba Zayed, Sara Shatnawi, Serry Sibaee, Yasir Ech-chammakhy, Walid Al-Dhabyani, Marwa Mohamed Ali, Imen Jarraya, and 25 others. 2025. Palm: A culturally inclusive and linguistically diverse dataset for Arabic LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32871–32894, Vienna, Austria. Association for Computational Linguistics.

Zaid Alyafeai, Khalid Almubarak, Ahmed Ashraf, Deema Alnuhait, Saied Alshahrani, Gubran Abdulrahman, Gamil Ahmed, Qais Gawah, Zead Saleh, Mustafa Ghaleb, Yousef Ali, and Maged Al-shaibani. 2024. CIDAR: Culturally relevant instruction dataset for Arabic. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12878–12901, Bangkok, Thailand. Association for Computational Linguistics.

Anthropic. 2024. Introducing claude 3.5 sonnet.

Anthropic. 2025. Introducing claude 4.

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Mag.*, 36(1):15–24.

Israel Abebe Azime, Atnafu Lambebo Tonja, Tadesse Destaw Belay, Yonas Chanie, Bontu Fufa Balcha, Negasi Haile Abadi, Henok Biadglign Ademtew, Mulubrhan Abebe Nerea, Debela Desalegn Yadeta, Derartu Dagne Geremew, Assefa Atsbiha Tesfu, Philipp Slusallek, Thamar Solorio, and Dietrich Klakow. 2025. ProverbEval: Exploring LLM evaluation challenges for low-resource language understanding. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6250–6266, Albuquerque, New Mexico. Association for Computational Linguistics.

M Saiful Bari, Yazeed Alnumay, Norah A. Alzahrani, Nouf M. Alotaibi, Hisham Abdullah Alyahya, Sultan AlRashed, Faisal Abdulrahman Mirza, Shaykhah Z. Alsubaie, Hassan A. Alahmed, Ghadah Alabduljabbar, Raghad Alkhathran, Yousef Almushayqih, Raneem Alnajim, Salman Alsubaihi, Maryam Al Mansour, Saad Amin Hassan, Dr. Majed Alrubaian, Ali Alammari, Zaki Alawami, and 7 others. 2025. AL-Lam: Large language models for arabic and english. In *The Thirteenth International Conference on Learning Representations*.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Tuhin Chakrabarty, Yejin Choi, and Vered Shwartz. 2022a. It's not rocket science: Interpreting figurative language in narratives. *Transactions of the Association for Computational Linguistics*, 10:589–606.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022b. FLUTE: Figurative language understanding through textual explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7139–7159, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ning Cheng, Zhaohui Yan, Ziming Wang, Zhijie Li, Jiaming Yu, Zilong Zheng, Kewei Tu, Jinan Xu, and Wenjuan Han. 2024. Potential and limitations of llms in capturing structured semantics: A case study on srl. *Preprint*, arXiv:2405.06410.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3290 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, and 26 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *Preprint*, arXiv:2412.04261.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. The language model evaluation harness.

Sara Ghaboura, Ahmed Heakl, Omkar Thawakar, Ali Husain Salem Abdulla Alharthi, Ines Riahi, Abduljalil Radman, Jorma Laaksonen, Fahad Shahbaz Khan, Salman Khan, and Rao Muhammad Anwer. 2025. CAMEL-bench: A comprehensive Arabic LMM benchmark. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1970–1980, Albuquerque, New Mexico. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur

Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Qianyu He, Sijie Cheng, Zhixu Li, Rui Xie, and Yanghua Xiao. 2022. Can pre-trained language models interpret similes as smart as human? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7875–7887, Dublin, Ireland. Association for Computational Linguistics.

Huang Huang, Fei Yu, Jianqing Zhu, Xuening Sun, Hao Cheng, Song Dingjie, Zhihong Chen, Mosen Alharthi, Bang An, Juncai He, Ziche Liu, Junying Chen, Jianquan Li, Benyou Wang, Lian Zhang, Ruoyu Sun, Xiang Wan, Haizhou Li, and Jinchao Xu. 2024. AceGPT, localizing large language models in Arabic. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8139–8163, Mexico City, Mexico. Association for Computational Linguistics.

Hyewon Jang, Qi Yu, and Diego Frassinelli. 2023. Figurative language processing: A linguistically informed feature analysis of the behavior of language models and humans. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9816–9832, Toronto, Canada. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Anubha Kabra, Emmy Liu, Simran Khanuja, Alham Fikri Aji, Genta Winata, Samuel Cahyawijaya, Anuoluwapo Aremu, Perez Ogayo, and Graham Neubig. 2023. Multi-lingual and multi-cultural figurative language understanding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8269–8284, Toronto, Canada. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, and 2 others.

2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chen Liu, Fajri Koto, Timothy Baldwin, and Iryna Gurevych. 2024. Are multilingual LLMs culturally-diverse reasoners? an investigation into multicultural proverbs and sayings. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2016–2039, Mexico City, Mexico. Association for Computational Linguistics.

Chen Cecilia Liu, Iryna Gurevych, and Anna Korhonen. 2025. Culturally aware and adapted NLP: A taxonomy and a survey of the state of the art. *Transactions of the Association for Computational Linguistics*, 13:652–689.

Emmy Liu, Chenxuan Cui, Kenneth Zheng, and Graham Neubig. 2022. Testing the ability of language models to interpret figurative language. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4437–4452, Seattle, United States. Association for Computational Linguistics.

Samar Mohamed Magdy, Sang Yun Kwon, Fakhraddin Alwajih, Safaa Taher Abdelfadil, Shady Shehata, and Muhammad Abdul-Mageed. 2025. JAWAHER: A multidialectal dataset of Arabic proverbs for LLM benchmarking. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 12320–12341, Albuquerque, New Mexico. Association for Computational Linguistics.

Gonzalo Martínez, Juan Diego Molero, Sandra González, Javier Conde, Marc Brysbaert, and Pedro Reviriego. 2024. Using large language models to estimate features of multi-word expressions: Concreteness, valence, arousal. *Preprint*, arXiv:2408.16012.

MBC. 2013. Mohamed Mounir & Nancy Ajram - حارة السقايين | Arab Idol. YouTube video, accessed August 8, 2025.

Basel Mousi, Nadir Durrani, Fatema Ahmad, Md. Arid Hasan, Maram Hasanain, Tameem Kabbani, Fahim Dalvi, Shammur Absar Chowdhury, and Firoj Alam. 2025. AraDiCE: Benchmarks for dialectal and cultural capabilities in LLMs. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4186–4218, Abu Dhabi, UAE. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao,

M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

OpenAI. 2025. Introducing gpt-4.1 in the api. OpenAI.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Ahmad Taymour Pasha. 1949. *Al-Kinayat Al-'Amiyya*, 2016 edition. Mu'assasat Hindawi. Available in EPUB, PDF, KFX formats.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large language models sensitivity to the order of options in multiple-choice questions. *Preprint*, arXiv:2308.11483.

Ben Prystawski, Paul Thibodeau, Christopher Potts, and Noah D. Goodman. 2023. Psychologically-informed chain-of-thought prompts for metaphor understanding in large language models. *Preprint*, arXiv:2209.08141.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Abdelrahman Sadallah, Junior Cedric Tonga, Khalid Almubarak, Saeed Almheiri, Farah Atif, Chatrine Qwaider, Karima Kadaoui, Sara Shatnawi, Yaser Alesh, and Fajri Koto. 2025. Commonsense reasoning in Arab culture. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7695–7710, Vienna, Austria. Association for Computational Linguistics.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2022. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5884–5906, Seattle, United States. Association for Computational Linguistics.

Neha Sengupta, Sunil Kumar Sahu, Bokang Jia, Satheesh Katipomu, Haonan Li, Fajri Koto, William Marshall, Gurpreet Gosal, Cynthia Liu, Zhiming Chen, Osama Mohammed Afzal, Samta Kamboj, Onkar Pandit, Rahul Pal, Lalit Pradhan, Zain Muhammad Mujahid, Massa Baali, Xudong Han, Sondos Mahmoud Bsharat, and 13 others. 2023. Jais and jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models. *Preprint*, arXiv:2308.16149.

Jingyuan S. She, Christopher Potts, Samuel R. Bowman, and Atticus Geiger. 2023. ScoNe: Benchmarking negation reasoning in language models with fine-tuning and in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1803–1821, Toronto, Canada. Association for Computational Linguistics.

silma-ai. 2024. Silma 9b instruct v1.0. https://huggingface.co/silma-ai/SILMA-9B-Instruct-v1.0.

Settaluri Sravanthi, Meet Doshi, Pavan Tankala, Rudra Murthy, Raj Dabre, and Pushpak Bhattacharyya. 2024. PUB: A pragmatics understanding benchmark for assessing LLMs' pragmatics capabilities. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12075–12097, Bangkok, Thailand. Association for Computational Linguistics.

Fanar Team, Ummar Abbas, Mohammad Shahmeer Ahmad, Firoj Alam, Enes Altinisik, Ehsannedin Asgari, Yazan Boshmaf, Sabri Boughorbel, Sanjay Chawla, Shammur Chowdhury, Fahim Dalvi, Kareem Darwish, Nadir Durrani, Mohamed Elfeky, Ahmed Elmagarmid, Mohamed Eltabakh, Masoomali Fatehkia, Anastasios Fragkopoulos, Maram Hasanain, and 23 others. 2025. Fanar: An arabic-centric multimodal generative ai platform. *Preprint*, arXiv:2501.13944.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, and 1118 others. 2024a. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy

Jerome, and 179 others. 2024b. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Thinh Hung Truong, Timothy Baldwin, Karin Verspoor, and Trevor Cohn. 2023. Language models are not naysayers: an analysis of language models on negation benchmarks. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 101–114, Toronto, Canada. Association for Computational Linguistics.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. 2024. Self-preference bias in LLM-as-a-judge. In *Neurips Safe Generative AI Workshop 2024*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. *Preprint*, arXiv:2309.03882.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

A Prompt Templates

The prompts used across our evaluation tasks are illustrated in Figures 5–15. Figure 5 presents the prompt for the MCQ understanding task, while Figure 6 shows the version that includes the idiom's sentence context. Figures 7 and 8 display the prompts used to generate incorrect distractors for the MCQ task, the latter leveraging semantic role labeling. Figure 9 illustrates the prompt used to complete proverbs by predicting the final word, and Figure 10 depicts the negated version of the MCQ understanding task. The generation-based prompt used to produce incorrect explanations is shown in Figure 11, with their evaluation guided by the judging prompt in Figure 12. Figure 13 displays the prompt used to generate example sentences containing idioms from Kinayat prior to human post-editing. Finally, Figures 14 and 15 show the prompts for the pragmatic use and connotation labeling tasks, respectively.

In all prompt templates that include the word "proverb", they refer to items from the Jawaher dataset of proverbs. For the Kinayat dataset of idioms, the word "proverb" in the templates was replaced with "idiom" to match the dataset content.

You are tasked with selecting the correct explanation for the following proverb. Choose the correct explanation from the options provided. Only output the letter corresponding to the correct answer and nothing else.

Proverb: `[PROVERB]`

Options: A. `[OPTION 1]`
        B. `[OPTION 2]`

Answer:

Figure 5: Prompt used for the MCQ understanding task.

You are tasked with selecting the correct explanation for the following idiom, given the idiom in a sentence for context. Choose the correct explanation from the options provided. Only output the letter corresponding to the correct answer and nothing else.

Idiom: `[IDIOM]`

Sentence: `[SENTENCE]`

Options: A. `[OPTION 1]`
        B. `[OPTION 2]`

Answer:

Figure 6: Prompt used for the Contextual MCQ Idiom Explanation task.

Given the following correct explanation, generate an incorrect explanation that sounds plausible and is not trivially incorrect. Only output the incorrect explanation and nothing else

Correct explanation: [correct explanation]

Figure 7: Prompt used for the generating incorrect explanantions.

Your task is to generate an incorrect explanation from the provided correct explanation by following the given steps:
1- Find the semantic role labels for the sentence.
2- Change one of the semantic role labels.
3- Generate an explanation using the new semantic role labels.
Only output the result in the following JSON format:
```
{semantic_role_labels: ,
new_labels: ,
new_sentence:}
```

Correct explanation: [correct explanation]

Figure 8: Prompt used for generating incorrect explanations using semantic role labeling.

You are tasked with completing the proverb with the last word. Output the next word only.

Incomplete Proverb: [incomplete proverb]
Answer:

Figure 9: Prompt used for last-word proverb completion.

You are tasked with selecting the incorrect explanation for the following proverb. Choose the incorrect explanation from the options provided. Only output the letter corresponding to the incorrect answer and nothing else.

Proverb: [PROVERB]

Options: A. [OPTION 1]
             B. [OPTION 2]

Answer:

Figure 10: Prompt used for the MCQ negation task.

Your task is to explain the meaning of the following Arabic proverb. Provide a clear and concise explanation in Arabic, highlighting its figurative meaning and any cultural or contextual significance.
Only output the Arabic explanation and nothing else.

Proverb: [PROVERB]
Arabic Explanation:

Figure 11: Prompt used for the task of generating explanations.

You are an expert in Arabic language and culture. Your task is to evaluate how well a generated explanation matches the intended meaning of the reference explanation of an Arabic idiom/proverb. Below is a reference (gold) explanation and a generated explanation.

Rate the accuracy of the generated explanation based on how well it preserves the intended meaning of the idiom/proverb.

Gold Explanation: {gold_explanation}
Generated Explanation: {generated_explanation}

Use the following rating scale:
5 = *Excellent:* Perfectly matches the gold explanation in meaning.
4 = *Good:* Minor omissions or phrasing differences, but the meaning is well preserved.
3 = *Fair:* Partial understanding, some inaccuracies or missing key aspects.
2 = *Poor:* Significant misunderstanding or loss of core meaning.
1 = Very Poor: Completely incorrect or irrelevant explanation.

Only output a numerical rating and nothing else.
Rating (1–5):

Figure 12: Prompt used for LLM-as-a-judge rating of idiom/proverb explanation quality.

Generate a sample sentence using the following idiom in the correct context in the Egyptian Arabic dialect given the following explanation of the idiom.

Idiom: {idiom}
Explanation: {explanation}

Only output the sentence and nothing else.

Figure 13: Prompt for generating an Egyptian Arabic sentence using a given idiom in context.

Determine the connotation of the following Arabic proverb or explanation. Classify the connotation as Positive, Negative, or Neutral based on the following guidelines:

Positive Connotation: It conveys optimism, hope, praise, or beneficial outcomes. It highlights virtues such as kindness, success, loyalty, or happiness. It encourages or celebrates desirable behaviors or outcomes.

Negative Connotation: It expresses pessimism, caution, loss, or undesirable consequences. It highlights flaws, mistakes, or risks and often reflects on the dangers or negative results of certain actions.

Neutral Connotation: It provides general advice or observation without invoking strong feelings or judgment.

Proverb/Explanation: [PROVERB OR EXPLANATION]

Only output the connotation and nothing else.
Connotation:

Figure 15: Prompt used for connotation classification of Arabic proverbs and their explanations.

Your task is to fill in the blank with the correct idiom.
Choose the correct idiom from the options provided. Only output the letter corresponding to the correct answer and nothing else.

Sentence: [sentence with blank]

Options: A. [OPTION 1]
           B. [OPTION 2]

Answer:

Figure 14: Prompt used for the idiom pragmatic use task.

B Verification of Incorrect Explanations

Figure 16 shows the annotation guidelines for verifying the quality of the generated incorrect explanations used as distractors in the MCQ understanding task, while Table 4 provides examples of generated explanations that received scores of 2, 1, and 0, along with the rationale for each score.

Task Overview A single annotator evaluated the quality of automatically generated *incorrect explanations* for idioms and proverbs. The goal was to determine whether each explanation was both *plausible* and *incorrect*, ensuring that it functioned as a meaningful distractor rather than a trivial or nonsensical option.

Inputs For each sample, the annotator was provided with: the idioms/proverbs, the correct explanations (true meaning), and the corresponding generated incorrect explanations.

Annotation Guidelines Each incorrect explanation was rated on a **0–2 scale** based on its plausibility, distinctness from the correct explanation, and linguistic coherence.

Use the following rating scale:
2 = *High-Quality Incorrect (Plausible but Wrong):* The explanation is clearly incorrect yet plausible. It makes sense linguistically and culturally, could realistically confuse a reader, and differs semantically from the correct meaning.
1 = *Medium-Quality Incorrect (Too Similar or Slightly Off):* The explanation is partially incorrect, implausible or too close in meaning to the correct one (e.g., paraphrase or mild variation). It shows partial understanding but fails to be a strong distractor.
0 = *Low-Quality Incorrect (Implausible or Unrelated):* The explanation is either correct, or clearly irrelevant, nonsensical, or incomplete. It fails to make sense in context and would not plausibly be mistaken for the correct meaning.

Figure 16: Guidelines for human verification of incorrect explanations generated by GPT-4.1.

Idiom	Correct Explanation	Incorrect Generation	Score	Rationale
جَسِ الْمَخَاضَهْ	كناية عن استطلاع الأمر، وهي في معنى سَبْرِ الغَوْرِ.	كناية عن إدراك التفاصيل الدقيقة، وهي في معنى تتبع الأثرِ.	2	The generation is incorrect but remains semantically plausible.
سَيِبْهْ يِرِنْ	ترك الدار خالية تصفر، هذا الأصل في الكناية، ثم كنوا بها عن ترك الشخص وشأنه فيما يعمل أو يتكلم به، وإهماله منفردًا يناجي نفسه في وحدته.	تُركَتِ الدار خالية تصفر، أي أنها امْتِلأت بالناس وازدحمت بالأصوات، ثم صار يُكْنى بذلك عن مواكبة الآخرين للمرء في أفعاله وأقواله ومشاركته في كل شأن من شُؤُونه.	1	The generation is incorrect and weakly plausible, as it contradicts the literal meaning by interpreting "empty" as "full."
جَابْهَا فِي قُبّهْ	أي علَقْها أو ربطها في قبّ قميص ذلك الشخص، أي طوّق بها عنقه وألصقها به، والمراد التهمة يتحايل بعضهم حتى يلصقها بشخص.	أي وضعها في جيب قميصه ليحفظها من الضياع، أي أبقاها بالقرب من صدره كعلامة على البراءة والثقة، والمقصود أن التهمة يبعدها البعض عن أنفسهم كي لا تلتصق بهم.	0	The generation is implausible and semantically incoherent with the idiom's intended meaning.

Table 4: Examples of model-generated explanations receiving scores of 2, 1, and 0, along with rationales. These examples illustrate the annotation guidelines used for explanation verification (see Figure 16).

C Data Analysis

Figure 17 shows the sequence length distributions for Arabic idioms (Kinayat), Arabic proverbs (Jawaher), and English proverbs (MAPS), with corresponding statistics in Table 5, highlighting that Arabic proverbs exhibit a higher mean sentence length than Arabic idioms, and English proverbs exhibit a higher mean sentence length than Arabic proverbs.

To assess semantic similarity, we compute cosine similarity between sentence embeddings using the `paraphrase-multilingual-mpnet-base-v2` model (Reimers and Gurevych, 2019), comparing correct and incorrect explanations generated with the general prompt (Figure 7) and the SRL-based prompt (Figure 8). As shown in Figure 18, the incorrect explanations produced by the general prompt result in a roughly normal-shaped distribution (mean = 0.6596), whereas the SRL-based prompt leads to a highly right-skewed distribution (mean = 0.9239).

Figure 19 presents the distribution of cosine similarity between the two idiom options in the pragmatic use task, with a mean similarity of 0.743.

Statistic	Kinayat	Jawaher	MAPS
Samples	325	198	394
Mean	2.79	5.21	6.52
Median	3.0	5.0	6.0
Std	1.08	1.98	2.64
Range	1 − 8	2 − 12	3 − 26

Table 5: Dataset statistics for `Kinayat`, `Jawaher`, and `MAPS` (`English`).
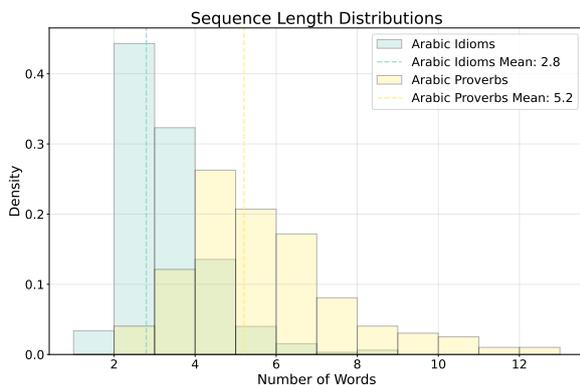
Figure 17: Sequence length distributions for Idioms and Proverbs.

D Additional Results

MCQ Positional Selection Bias Research has shown that LLMs can favor certain answer choices
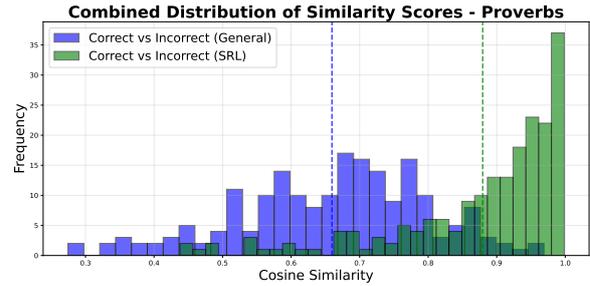
Figure 18: Cosine similarity (↑) between the correct and incorrect explanation choices for proverbs (Jawaher).
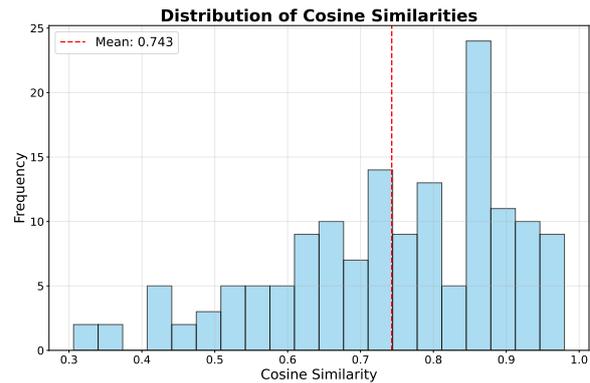
Figure 19: Cosine similarity (↑) between the correct and incorrect idiom choices for the pragmatic use task.

due to token-level prior probabilities (Zheng et al., 2024; Pezeshkpour and Hruschka, 2023). We ran the evaluation with both permutations of the correct and incorrect answers. When the correct answer was always listed first, models performed better (91.95% vs. 79.65% average accuracy), as shown in Table 6. For the remaining experiments, the order of the options was randomized to mitigate this bias.

Knowledge and Understanding: Table 9 reports MCQ understanding performance across all datasets (MAPS, Kinayat, and Jawaher), while Table 10 compares results on Kinayat and Jawaher under both the general and SRL-based prompting strategies. The highest accuracy is observed on English proverbs with context (95.66%), followed by English proverbs without context (90.86%), multi-dialectal Arabic proverbs (86.57%), and Egyptian Arabic idioms, which yield the lowest accuracy (76.29%). Table 7 presents results for the task in which models are asked to identify the incorrect explanation (generated using the general prompt), showing a decline in performance compared to selecting the correct explanation. Table 8 summarizes results for the completion task, where a substantial performance gap emerges between Arabic and

Model	MCQ A	MCQ B	Difference
Llama-3.1-8B-Instruct	$0.8182^{\pm0.0275}$	$0.6667^{\pm0.0336}$	0.1515
Gemma-2-9B-it	$0.9343^{\pm0.0176}$	$0.8788^{\pm0.0233}$	0.0556
Qwen2.5-7B-Instruct	$0.9141^{\pm0.0200}$	$0.7828^{\pm0.0294}$	0.1313
Qwen2.5-14B-Instruct	$0.9192^{\pm0.0194}$	$0.9091^{\pm0.0205}$	0.0101
Aya-expanse-8b	$0.8838^{\pm0.0228}$	$0.6667^{\pm0.0336}$	0.2172
Aya-expanse-32b	$0.9293^{\pm0.0183}$	$0.8838^{\pm0.0228}$	0.0455
Mistral-7B-Instruct-v0.3	$0.9293^{\pm0.0183}$	$0.2475^{\pm0.0307}$	0.6818
Jais-family-6p7b-chat	$0.6212^{\pm0.0346}$	$0.8182^{\pm0.0275}$	-0.1970
Fanar-1-9B-Instruct	$0.9444^{\pm0.0163}$	$0.8434^{\pm0.0259}$	0.1010
SILMA-9B-Instruct-v1.0	$0.9444^{\pm0.0163}$	$0.7475^{\pm0.0310}$	0.1970
ALLaM-7B-Instruct-preview	$0.9141^{\pm0.0200}$	$0.8737^{\pm0.0237}$	0.0404
AceGPT-v2-8B-Chat	$0.9697^{\pm0.0122}$	$0.5202^{\pm0.0356}$	0.4495
Claude-Sonnet-4	$0.9848^{\pm0.0087}$	$0.9798^{\pm0.0100}$	0.0051
Claude-3.5-Sonnet	$0.9949^{\pm0.0051}$	$0.9747^{\pm0.0112}$	0.0202
Gemini-1.5-flash	$0.9545^{\pm0.0148}$	$0.9091^{\pm0.0205}$	0.0455
GPT-4o	$0.9899^{\pm0.0071}$	$0.9646^{\pm0.0132}$	0.0253
GPT-4o-mini	$0.9848^{\pm0.0087}$	$0.8737^{\pm0.0237}$	0.1111
Average	0.9195	0.7965	0.1230

Table 6: Accuracy $(\uparrow)^{\pm\text{stderr}\,(\downarrow)}$ on the positional bias MCQ understanding task for variants A and B, and their difference (A – B) on the Jawaher dataset (general prompt used for generating incorrect choices) for a subset of models.

English proverb completion (10.64% vs. 75.43%).

Country Breakdown: Figures 20 and 21 illustrate country-level accuracies in descending order for the MCQ Understanding task using the general and SRL-based prompts, respectively. Mauritania and Sudan consistently appear among the lowest-performing dialects in both settings, whereas UAE and MSA remain within the top-performing group across both prompting strategies. Tables 14 and 15 provide detailed country-level results for the Jawaher dataset under each prompt condition, and Table 16 reports the average accuracy across the two settings.

Pragmatic Use: Figure 22 visualizes model performance on the pragmatic use, MCQ understanding, and contextual MCQ understanding tasks using a 150-idiom subset from the Kinayat dataset, corresponding to the results reported in Table 3. A consistent performance gradient emerges, with the pragmatic use task yielding the lowest scores, followed by MCQ understanding, and the highest performance observed in contextual MCQ understanding.

Connotations: Table 12 presents the connotation task results for the Jawaher and Kinayat datasets, restricted to samples with full inter-annotator agreement (105 entries for Jawaher and 104 for Kinayat). Among all models, the Claude family (Claude Sonnet 4 and Claude 3.5 Sonnet) achieved the highest performance. Overall, proverb connotations were slightly harder to identify than idiomatic ones, with lower average model agreement (49.71% vs. 50.10%). However, models demonstrated stronger

performance when inferring connotations from the explanations, achieving average accuracies of 68.17% for proverbs and 72.21% for idioms.

Overall Performance: Table 13 presents the average performance of multilingual open-source, Arabic open-source, and closed-source models across all tasks. On average, multilingual open-source models slightly outperformed Arabic models on most tasks; however, Arabic models led on proverb completion, Kinayat MCQ (SRL-prompt), Kinayat pragmatic use, and explanation generation for both proverbs and idioms. These averages, however, mask variation within each category. For example, the multilingual model Mistral-7B-Instruct performed below most Arabic models on several tasks. In contrast, closed-source models consistently outperformed both Arabic and multilingual open-source models across all evaluations. One potential contributing factor is model scale, as closed-source systems are generally assumed to be larger than the open-source models evaluated, though their exact sizes are not publicly disclosed.

Model	Kinayat	Jawaher
Llama-3.1-8B-Instruct	$0.6492^{\pm0.0265}$	$0.7424^{\pm0.0312}$
Llama-3.1-70B-Instruct	$0.5508^{\pm0.0276}$	$0.6919^{\pm0.0329}$
Gemma-2-9B-it	$0.6031^{\pm0.0272}$	$0.8384^{\pm0.0262}$
Gemma-2-27b-it	$0.8123^{\pm0.0217}$	$0.9192^{\pm0.0194}$
Qwen2.5-7B-Instruct	$0.6277^{\pm0.0269}$	$0.7677^{\pm0.0301}$
Qwen2.5-14B-Instruct	$0.7908^{\pm0.0226}$	$0.8737^{\pm0.0237}$
Qwen2.5-32B-Instruct	$0.8000^{\pm0.0222}$	$0.9141^{\pm0.0200}$
Aya-expanse-8b	$0.5754^{\pm0.0275}$	$0.7020^{\pm0.0326}$
Aya-expanse-32b	$0.8092^{\pm0.0218}$	$0.8586^{\pm0.0248}$
Mistral-7B-Instruct-v0.3	$0.4923^{\pm0.0278}$	$0.5808^{\pm0.0352}$
Jais-family-6p7b-chat	$0.6062^{\pm0.0271}$	$0.7374^{\pm0.0314}$
Jais-family-13b-chat	$0.5846^{\pm0.0274}$	$0.7020^{\pm0.0326}$
Fanar-1-9B-Instruct	$0.5815^{\pm0.0274}$	$0.7980^{\pm0.0286}$
SILMA-9B-Instruct-v1.0	$0.5354^{\pm0.0277}$	$0.8131^{\pm0.0278}$
ALLaM-7B-Instruct-preview	$0.8000^{\pm0.0222}$	$0.8990^{\pm0.0215}$
AceGPT-v2-8B-Chat	$0.5969^{\pm0.0273}$	$0.7374^{\pm0.0314}$
Claude-Sonnet-4	$0.9446^{\pm0.0127}$	$0.9697^{\pm0.0122}$
Claude-3.5-Sonnet	$0.9169^{\pm0.0153}$	$0.9747^{\pm0.0112}$
Gemini-1.5-flash	$0.7785^{\pm0.0231}$	$0.8939^{\pm0.0219}$
Gemini-2.5-flash-lite-preview-06-17	$0.8308^{\pm0.0208}$	$0.9242^{\pm0.0189}$
GPT-4o	$0.9477^{\pm0.0124}$	$0.9798^{\pm0.0100}$
GPT-4o-mini	$0.7785^{\pm0.0231}$	$0.8788^{\pm0.0233}$
Average	$0.7097^{\pm0.0000}$	$0.8271^{\pm0.0000}$

Table 7: Accuracy $(\uparrow)^{\pm\text{stderr}\,(\downarrow)}$ on the task of selecting the incorrect explanation for Kinayat and Jawaher datasets.

Model	MAPS	Jawaher
Llama-3.1-8B-Instruct	$0.6954^{\pm 0.0232}$	$0.0253^{\pm 0.0112}$
Llama-3.1-70B-Instruct	$0.8756^{\pm 0.0166}$	$0.0707^{\pm 0.0183}$
Gemma-2-9B-it	$0.8046^{\pm 0.0200}$	$0.0556^{\pm 0.0163}$
Gemma-2-27b-it	$0.6751^{\pm 0.0236}$	$0.0808^{\pm 0.0194}$
Qwen2.5-7B-Instruct	$0.6421^{\pm 0.0242}$	$0.0253^{\pm 0.0112}$
Qwen2.5-14B-Instruct	$0.7792^{\pm 0.0209}$	$0.0354^{\pm 0.0132}$
Qwen2.5-32B-Instruct	$0.8401^{\pm 0.0185}$	$0.0606^{\pm 0.0170}$
Aya-expanse-8b	$0.5051^{\pm 0.0252}$	$0.0556^{\pm 0.0163}$
Aya-expanse-32b	$0.7107^{\pm 0.0229}$	$0.1364^{\pm 0.0245}$
Mistral-7B-Instruct-v0.3	$0.7614^{\pm 0.0215}$	$0.0000^{\pm 0.0000}$
Jais-family-6p7b-chat	$0.2995^{\pm 0.0231}$	$0.0303^{\pm 0.0122}$
Jais-family-13b-chat	$0.3604^{\pm 0.0242}$	$0.0152^{\pm 0.0087}$
Fanar-1-9B-Instruct	$0.8046^{\pm 0.0200}$	$0.1111^{\pm 0.0224}$
SILMA-9B-Instruct-v1.0	$0.8756^{\pm 0.0166}$	$0.0354^{\pm 0.0132}$
ALLaM-7B-Instruct-preview	$0.7107^{\pm 0.0229}$	$0.1313^{\pm 0.0241}$
AceGPT-v2-8B-Chat	$0.7690^{\pm 0.0213}$	$0.0808^{\pm 0.0194}$
Claude-Sonnet-4	$0.9340^{\pm 0.0125}$	$0.2980^{\pm 0.0326}$
Claude-3.5-Sonnet	$\mathbf{0.9391}^{\pm 0.0121}$	$\mathbf{0.3636}^{\pm 0.0343}$
Gemini-1.5-flash	$0.9061^{\pm 0.0147}$	$0.1212^{\pm 0.0233}$
Gemini-2.5-flash-lite-preview-06-17	$0.8782^{\pm 0.0165}$	$0.2273^{\pm 0.0299}$
GPT-4o	$0.9340^{\pm 0.0125}$	$0.2879^{\pm 0.0323}$
GPT-4o-mini	$0.8934^{\pm 0.0156}$	$0.0960^{\pm 0.0210}$
Average	0.7543	0.1065

Table 8: Evaluation results of completion task (accuracy $(\uparrow)^{\pm \text{stderr}(\downarrow)}$) for different models on MAPS and Jawaher datasets.
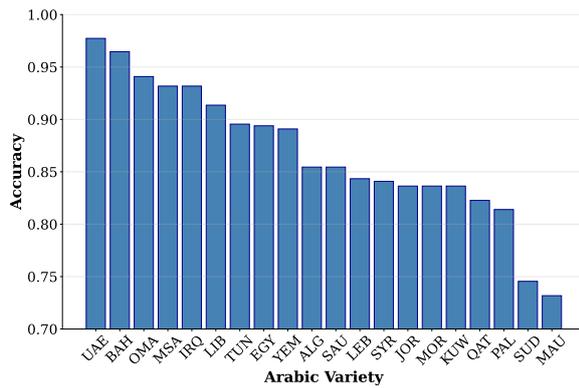
Figure 20: Country-level breakdown of MCQ Understanding Accuracy (incorrect distractor generated with general prompt).
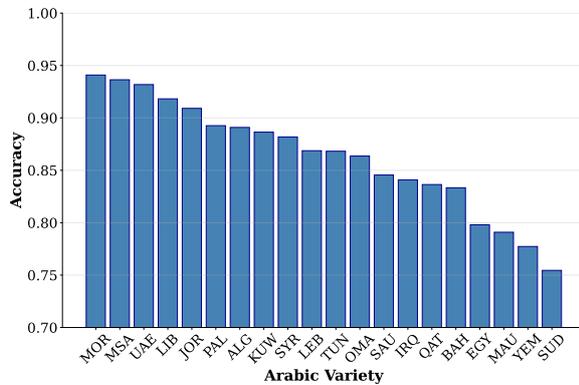
Figure 21: Country-level breakdown of MCQ Understanding Accuracy (incorrect distractor generated with SRL-based prompt).

Model	MAPS	MAPS + Context	Jawaher	Kinayat
Llama-3.1-8B-Instruct	$0.8655^{\pm0.0172}$	$0.9213^{\pm0.0136}$	$0.7475^{\pm0.0310}$	$0.5754^{\pm0.0275}$
Llama-3.1-70B-Instruct	$0.9239^{\pm0.0134}$	$0.9873^{\pm0.0056}$	$0.8990^{\pm0.0215}$	$0.8831^{\pm0.0179}$
Gemma-2-9B-it	$0.9365^{\pm0.0123}$	$0.9670^{\pm0.0090}$	$0.9091^{\pm0.0205}$	$0.7354^{\pm0.0245}$
Gemma-2-27b-it	$0.9340^{\pm0.0125}$	$0.9721^{\pm0.0083}$	$0.8990^{\pm0.0215}$	$0.7908^{\pm0.0226}$
Qwen2.5-7B-Instruct	$0.9239^{\pm0.0134}$	$0.9492^{\pm0.0111}$	$0.8535^{\pm0.0252}$	$0.7262^{\pm0.0248}$
Qwen2.5-14B-Instruct	$0.9391^{\pm0.0121}$	$0.9695^{\pm0.0087}$	$0.8990^{\pm0.0215}$	$0.7938^{\pm0.0225}$
Qwen2.5-32B-Instruct	$0.9391^{\pm0.0121}$	$\mathbf{0.9924}^{\pm0.0044}$	$0.9192^{\pm0.0194}$	$0.8000^{\pm0.0222}$
Aya-expanse-8b	$0.8858^{\pm0.0160}$	$0.9315^{\pm0.0127}$	$0.7929^{\pm0.0289}$	$0.6338^{\pm0.0268}$
Aya-expanse-32b	$0.9340^{\pm0.0125}$	$0.9797^{\pm0.0071}$	$0.8939^{\pm0.0219}$	$0.8154^{\pm0.0216}$
Mistral-7B-Instruct-v0.3	$0.8579^{\pm0.0176}$	$0.9213^{\pm0.0136}$	$0.6061^{\pm0.0348}$	$0.5169^{\pm0.0278}$
Jais-family-6p7b-chat	$0.7640^{\pm0.0214}$	$0.8680^{\pm0.0171}$	$0.7525^{\pm0.0307}$	$0.7046^{\pm0.0253}$
Jais-family-13b-chat	$0.8376^{\pm0.0186}$	$0.9162^{\pm0.0140}$	$0.7626^{\pm0.0303}$	$0.6769^{\pm0.0260}$
Fanar-1-9B-Instruct	$0.9010^{\pm0.0151}$	$0.9442^{\pm0.0116}$	$0.8737^{\pm0.0237}$	$0.7508^{\pm0.0240}$
SILMA-9B-Instruct-v1.0	$0.9492^{\pm0.0111}$	$0.9569^{\pm0.0102}$	$0.8737^{\pm0.0237}$	$0.6277^{\pm0.0269}$
ALLaM-7B-Instruct-preview	$0.8807^{\pm0.0164}$	$0.9365^{\pm0.0123}$	$0.8636^{\pm0.0245}$	$0.7846^{\pm0.0228}$
AceGPT-v2-8B-Chat	$0.9112^{\pm0.0144}$	$0.9543^{\pm0.0105}$	$0.7475^{\pm0.0310}$	$0.5754^{\pm0.0275}$
Claude-Sonnet-4	$0.9340^{\pm0.0125}$	$0.9797^{\pm0.0071}$	$0.9798^{\pm0.0100}$	$\mathbf{0.9662}^{\pm0.0100}$
Claude-3.5-Sonnet	$0.9543^{\pm0.0105}$	$0.9772^{\pm0.0075}$	$\mathbf{0.9848}^{\pm0.0087}$	$0.9415^{\pm0.0130}$
Gemini-1.5-flash	$0.9239^{\pm0.0134}$	$0.9746^{\pm0.0079}$	$0.9343^{\pm0.0176}$	$0.8185^{\pm0.0214}$
Gemini-2.5-flash-lite-preview-06-17	$0.8832^{\pm0.0162}$	$0.9746^{\pm0.0079}$	$0.9596^{\pm0.0140}$	$0.9077^{\pm0.0161}$
GPT-4o	$\mathbf{0.9619}^{\pm0.0097}$	$0.9873^{\pm0.0056}$	$0.9798^{\pm0.0100}$	$0.9415^{\pm0.0130}$
GPT-4o-mini	$0.9492^{\pm0.0111}$	$0.9848^{\pm0.0062}$	$0.9141^{\pm0.0200}$	$0.8185^{\pm0.0214}$
Average	0.9086	0.9566	0.8657	0.7629

Table 9: Evaluation results (accuracy $(\uparrow)^{\pm \text{stderr}\,(\downarrow)}$) of multiple choice understanding task on different test sets. The incorrect explanation choices for Jawaher and Kinayat for the results shown here were generated with the general prompt.

Model	Jawaher (General)	Jawaher (SRL)	Kinayat (General)	Kinayat (SRL)
Llama-3.1-8B-Instruct	$0.7475^{\pm0.0310}$	$0.7475^{\pm0.0310}$	$0.5754^{\pm0.0275}$	$0.6400^{\pm0.0267}$
Llama-3.1-70B-Instruct	$0.8990^{\pm0.0215}$	$0.9141^{\pm0.0200}$	$0.8990^{\pm0.0215}$	$0.8492^{\pm0.0199}$
Gemma-2-9B-it	$0.9091^{\pm0.0205}$	$0.8838^{\pm0.0228}$	$0.7354^{\pm0.0245}$	$0.7692^{\pm0.0234}$
Gemma-2-27b-it	$0.8990^{\pm0.0215}$	$0.8939^{\pm0.0219}$	$0.7908^{\pm0.0226}$	$0.8123^{\pm0.0217}$
Qwen2.5-7B-Instruct	$0.8535^{\pm0.0252}$	$0.8687^{\pm0.0241}$	$0.7262^{\pm0.0248}$	$0.7815^{\pm0.0230}$
Qwen2.5-14B-Instruct	$0.8990^{\pm0.0215}$	$0.9293^{\pm0.0183}$	$0.7938^{\pm0.0225}$	$0.8431^{\pm0.0202}$
Qwen2.5-32B-Instruct	$0.9192^{\pm0.0194}$	$0.9545^{\pm0.0148}$	$0.8000^{\pm0.0222}$	$0.8308^{\pm0.0208}$
Aya-expanse-8b	$0.7929^{\pm0.0289}$	$0.7677^{\pm0.0301}$	$0.6338^{\pm0.0268}$	$0.7323^{\pm0.0246}$
Aya-expanse-32b	$0.8939^{\pm0.0219}$	$0.9242^{\pm0.0189}$	$0.8154^{\pm0.0216}$	$0.8185^{\pm0.0214}$
Mistral-7B-Instruct-v0.3	$0.6061^{\pm0.0348}$	$0.5960^{\pm0.0350}$	$0.5169^{\pm0.0278}$	$0.5262^{\pm0.0277}$
Jais-family-6p7b-chat	$0.7525^{\pm0.0307}$	$0.7929^{\pm0.0289}$	$0.7046^{\pm0.0253}$	$0.7262^{\pm0.0248}$
Jais-family-13b-chat	$0.7626^{\pm0.0303}$	$0.7879^{\pm0.0291}$	$0.6769^{\pm0.0260}$	$0.7446^{\pm0.0242}$
Fanar-1-9B-Instruct	$0.8737^{\pm0.0237}$	$0.8737^{\pm0.0237}$	$0.7508^{\pm0.0240}$	$0.7446^{\pm0.0242}$
SILMA-9B-Instruct-v1.0	$0.8737^{\pm0.0237}$	$0.8485^{\pm0.0255}$	$0.6277^{\pm0.0269}$	$0.7415^{\pm0.0243}$
ALLaM-7B-Instruct-preview	$0.8636^{\pm0.0245}$	$0.8586^{\pm0.0248}$	$0.7846^{\pm0.0228}$	$0.8462^{\pm0.0200}$
AceGPT-v2-8B-Chat	$0.7475^{\pm0.0310}$	$0.7576^{\pm0.0305}$	$0.5754^{\pm0.0275}$	$0.6585^{\pm0.0263}$
Claude-Sonnet-4	$0.9798^{\pm0.0100}$	$\mathbf{0.9848}^{\pm0.0087}$	$\mathbf{0.9662}^{\pm0.0100}$	$0.9077^{\pm0.0161}$
Claude-3.5-Sonnet	$\mathbf{0.9848}^{\pm0.0087}$	$0.9646^{\pm0.0132}$	$0.9415^{\pm0.0130}$	$0.9046^{\pm0.0163}$
Gemini-1.5-flash	$0.9343^{\pm0.0176}$	$0.8990^{\pm0.0215}$	$0.8185^{\pm0.0214}$	$0.8738^{\pm0.0184}$
Gemini-2.5-flash-lite-preview-06-17	$0.9596^{\pm0.0140}$	$0.9343^{\pm0.0176}$	$0.9077^{\pm0.0161}$	$0.8431^{\pm0.0202}$
GPT-4o	$0.9798^{\pm0.0100}$	$0.9596^{\pm0.0140}$	$0.9415^{\pm0.0130}$	$\mathbf{0.9138}^{\pm0.0156}$
GPT-4o-mini	$0.9141^{\pm0.0200}$	$0.8586^{\pm0.0248}$	$0.8185^{\pm0.0214}$	$0.7662^{\pm0.0235}$
Average	0.8657	0.8636	0.7629	0.7852

Table 10: Accuracy $(\uparrow)^{\pm \text{stderr}\,(\downarrow)}$ on Jawaher and Kinayat datasets with incorrect explanations generated by a general vs. SRL variant prompt.
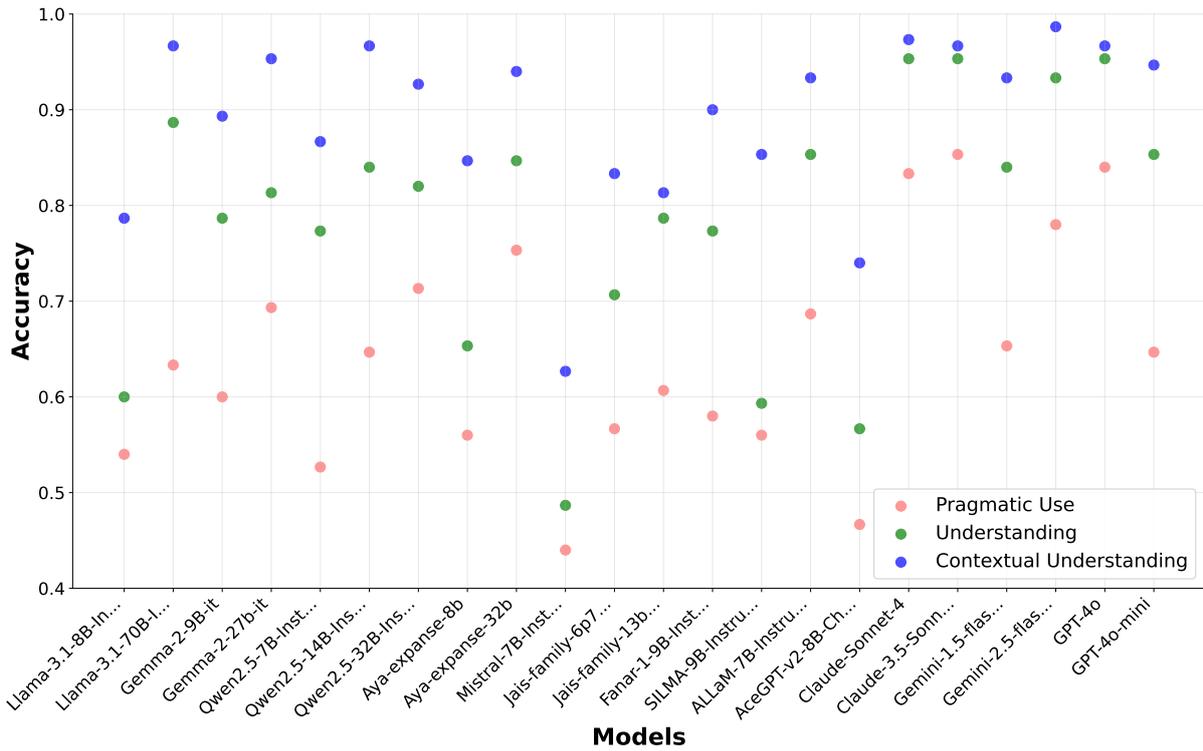
Figure 22: Accuracy (↑) on Pragmatic Use and Understanding tasks on 150 samples from the Kinayat dataset.

Model	Jawaher		Kinayat	
	BERT-F1	**LLM-Judge**	**BERT-F1**	**LLM-Judge**
Llama-3.1-8B-Instruct	$0.5988^{\pm0.0022}$	$2.1111^{\pm0.0701}$	$0.5714^{\pm0.0015}$	$1.5292^{\pm0.0350}$
Llama-3.1-70B-Instruct	$0.6816^{\pm0.0021}$	$3.1869^{\pm0.0934}$	$0.6486^{\pm0.0017}$	$2.1138^{\pm0.0531}$
Gemma-2-9B-it	$0.6774^{\pm0.0022}$	$2.9293^{\pm0.0899}$	$0.6492^{\pm0.0018}$	$2.0862^{\pm0.0452}$
Gemma-2-27b-it	$0.6893^{\pm0.0019}$	$3.0455^{\pm0.0921}$	$0.6793^{\pm0.0012}$	$2.1877^{\pm0.0517}$
Qwen2.5-7B-Instruct	$0.6759^{\pm0.0020}$	$2.5303^{\pm0.0793}$	$0.6584^{\pm0.0012}$	$2.0646^{\pm0.0477}$
Qwen2.5-14B-Instruct	$0.6651^{\pm0.0031}$	$2.8485^{\pm0.1071}$	$0.6560^{\pm0.0013}$	$2.1662^{\pm0.0518}$
Qwen2.5-32B-Instruct	$0.6737^{\pm0.0023}$	$3.2020^{\pm0.0974}$	$0.6541^{\pm0.0016}$	$2.2092^{\pm0.0566}$
Aya-expanse-8b	$0.6794^{\pm0.0018}$	$2.9343^{\pm0.0859}$	$0.6633^{\pm0.0012}$	$2.1815^{\pm0.0512}$
Aya-expanse-32b	$0.6820^{\pm0.0019}$	$3.4646^{\pm0.0935}$	$0.6615^{\pm0.0014}$	$2.4123^{\pm0.0595}$
Mistral-7B-Instruct-v0.3	$0.6695^{\pm0.0019}$	$2.1313^{\pm0.0528}$	$0.6373^{\pm0.0019}$	$1.7415^{\pm0.0335}$
Jais-family-6p7b-chat	$0.6762^{\pm0.0028}$	$2.2071^{\pm0.0703}$	$0.6226^{\pm0.0028}$	$1.5200^{\pm0.0419}$
Jais-family-13b-chat	$0.6598^{\pm0.0033}$	$2.4495^{\pm0.0812}$	$0.6267^{\pm0.0023}$	$1.7108^{\pm0.0430}$
Fanar-1-9B-Instruct	$0.6799^{\pm0.0019}$	$3.3535^{\pm0.0899}$	$0.6687^{\pm0.0011}$	$2.3200^{\pm0.0555}$
SILMA-9B-Instruct-v1.0	$0.6856^{\pm0.0026}$	$2.5556^{\pm0.0824}$	$0.6581^{\pm0.0025}$	$1.6923^{\pm0.0417}$
ALLaM-7B-Instruct-preview	$0.6893^{\pm0.0021}$	$3.2727^{\pm0.0902}$	$0.6675^{\pm0.0011}$	$2.3077^{\pm0.0606}$
AceGPT-v2-8B-Chat	$0.6730^{\pm0.0017}$	$3.0606^{\pm0.0870}$	$0.6455^{\pm0.0010}$	$2.0031^{\pm0.0450}$
Claude-Sonnet-4	$0.6990^{\pm0.0018}$	$3.7778^{\pm0.0848}$	$0.6795^{\pm0.0012}$	$2.6646^{\pm0.0687}$
Claude-3.5-Sonnet	$\mathbf{0.6998}^{\pm0.0017}$	$\mathbf{3.8939}^{\pm0.0887}$	$\mathbf{0.6798}^{\pm0.0013}$	$\mathbf{2.9262}^{\pm0.0724}$
Gemini-1.5-flash	$0.6700^{\pm0.0020}$	$3.5152^{\pm0.0911}$	$0.6360^{\pm0.0028}$	$2.4769^{\pm0.0587}$
Gemini-2.5-flash-lite-preview-06-17	$0.6827^{\pm0.0020}$	$3.6818^{\pm0.0921}$	$0.6684^{\pm0.0013}$	$2.8400^{\pm0.0700}$
GPT-4o	$0.6811^{\pm0.0022}$	$3.6667^{\pm0.0982}$	$0.6560^{\pm0.0011}$	$2.9046^{\pm0.0819}$
GPT-4o-mini	$0.6813^{\pm0.0019}$	$3.4394^{\pm0.1029}$	$0.6552^{\pm0.0010}$	$2.1231^{\pm0.0682}$
Average	0.6759	3.0572	0.6520	2.1901

Table 11: Explanation generation scores using BERTScore-F1 (↑) and LLM-as-a-judge (↑) for **Jawaher** and **Kinayat**.

Model	Proverb	Proverb Explanation	Idiom	Idiom Explanation
Llama-3.1-8B-Instruct	$0.5577^{\pm 0.0489}$	$0.7019^{\pm 0.0451}$	$0.7404^{\pm 0.0432}$	$0.7692^{\pm 0.0415}$
Llama-3.1-70B-Instruct	$0.2788^{\pm 0.0442}$	$0.7019^{\pm 0.0451}$	$0.2115^{\pm 0.0402}$	$0.8269^{\pm 0.0373}$
Gemma-2-9B-it	$0.6058^{\pm 0.0482}$	$0.7981^{\pm 0.0396}$	$0.7692^{\pm 0.0415}$	$0.8654^{\pm 0.0336}$
Gemma-2-27b-it	$0.6250^{\pm 0.0477}$	$0.7885^{\pm 0.0402}$	$0.8269^{\pm 0.0373}$	$0.8750^{\pm 0.0326}$
Qwen2.5-7B-Instruct	$0.5288^{\pm 0.0492}$	$0.7788^{\pm 0.0409}$	$0.5481^{\pm 0.0490}$	$0.8365^{\pm 0.0364}$
Qwen2.5-14B-Instruct	$0.6442^{\pm 0.0472}$	$0.7788^{\pm 0.0409}$	$0.7692^{\pm 0.0415}$	$0.8558^{\pm 0.0346}$
Qwen2.5-32B-Instruct	$0.5673^{\pm 0.0488}$	$0.8173^{\pm 0.0381}$	$0.5769^{\pm 0.0487}$	$0.8558^{\pm 0.0346}$
Aya-expanse-8b	$0.3462^{\pm 0.0469}$	$0.6058^{\pm 0.0482}$	$0.2596^{\pm 0.0432}$	$0.5769^{\pm 0.0487}$
Aya-expanse-32b	$0.5962^{\pm 0.0483}$	$0.7308^{\pm 0.0437}$	$0.3462^{\pm 0.0469}$	$0.7788^{\pm 0.0409}$
Mistral-7B-Instruct-v0.3	$0.5096^{\pm 0.0493}$	$0.7404^{\pm 0.0432}$	$0.4423^{\pm 0.0489}$	$0.8558^{\pm 0.0346}$
Jais-family-6p7b-chat	$0.2019^{\pm 0.0396}$	$0.2692^{\pm 0.0437}$	$0.0865^{\pm 0.0277}$	$0.1442^{\pm 0.0346}$
Jais-family-13b-chat	$0.1635^{\pm 0.0364}$	$0.1635^{\pm 0.0364}$	$0.0865^{\pm 0.0277}$	$0.0865^{\pm 0.0277}$
Fanar-1-9B-Instruct	$0.4327^{\pm 0.0488}$	$0.7115^{\pm 0.0446}$	$0.3750^{\pm 0.0477}$	$0.8077^{\pm 0.0388}$
SILMA-9B-Instruct-v1.0	$0.3654^{\pm 0.0474}$	$0.7308^{\pm 0.0437}$	$0.2500^{\pm 0.0427}$	$0.8654^{\pm 0.0336}$
ALLaM-7B-Instruct-preview	$0.2308^{\pm 0.0415}$	$0.3173^{\pm 0.0459}$	$0.1923^{\pm 0.0388}$	$0.1923^{\pm 0.0388}$
AceGPT-v2-8B-Chat	$0.4808^{\pm 0.0492}$	$0.7308^{\pm 0.0437}$	$0.2596^{\pm 0.0432}$	$0.7788^{\pm 0.0409}$
Claude-Sonnet-4	$0.7115^{\pm 0.0446}$	$\mathbf{0.8654}^{\pm 0.0336}$	$0.8365^{\pm 0.0364}$	$\mathbf{0.8942}^{\pm 0.0303}$
Claude-3.5-Sonnet	$\mathbf{0.7404}^{\pm 0.0432}$	$0.8077^{\pm 0.0388}$	$\mathbf{0.8558}^{\pm 0.0346}$	$0.9038^{\pm 0.0290}$
GPT-4o	$0.7115^{\pm 0.0446}$	$0.7981^{\pm 0.0396}$	$0.8462^{\pm 0.0356}$	$0.8558^{\pm 0.0346}$
GPT-4o-mini	$0.6442^{\pm 0.0472}$	$0.7981^{\pm 0.0396}$	$0.7404^{\pm 0.0432}$	$0.8173^{\pm 0.0381}$
Average	0.4971	0.6817	0.5010	0.7221

Table 12: Accuracy (↑) and standard error (↓) for the connotations of proverbs, their explanations (Jawaher dataset), idioms, and their explanations (Kinayat dataset).

Task	Multilingual Average	Arabic Average	Closed-Source Average
MAPS MCQ	0.9129	0.8739	0.9344
MAPS MCQ Context	0.9560	0.9294	0.9797
MAPS Completion	0.7126	0.6366	0.9141
English Average Accuracy	0.8605	0.8133	0.9428
Jawaher MCQ (general)	0.8356	0.8123	0.9588
Jawaher MCQ (SRL)	0.8406	0.8199	0.9335
Jawaher MCQ Negation	0.7997	0.7811	0.9369
Jawaher Completion	0.0527	0.0673	0.2323
Kinayat MCQ (general)	0.7097	0.6867	0.8990
Kinayat MCQ (SRL)	0.7504	0.7436	0.8682
Kinayat MCQ Negation	0.6844	0.6174	0.8662
Kinayat Pragmatic Use	0.6081	0.5778	0.7678
Kinayat MCQ (150 samples)	0.7356	0.7133	0.9144
Kinayat MCQ Context (150 samples)	0.8674	0.8456	0.9622
Arabic Average Accuracy	0.6884	0.6665	0.8339
Jawaher Generation BERTScore-F1	0.6679	0.6770	0.6857
Kinayat Generation BERTScore-F1	0.6478	0.6530	0.6625
Arabic Average BERTScore-F1	0.6579	0.6650	0.6741
Jawaher Generation LLM-Judge	2.7997	2.8480	3.6625
Kinayat Generation LLM-Judge	2.0643	2.0683	2.6559
Arabic Average LLM-Judge	2.4320	2.4581	3.1592

Table 13: Performance results (↑) across different tasks and evaluation metrics (Multilingual Average excludes Llama-3.1 70B Instruct model).

Model	EGY	MSA	UAE	JOR	MAU	PAL	ALG	SYR	IRQ	LEB	SAU	YEM	MOR	QAT	SUD	KUW	OMA	TUN	BAH	LIB
Llama-3.1-8B-Instruct	0.778	0.900	1.000	0.700	0.400	0.455	0.800	0.700	1.000	0.444	0.800	0.700	0.800	0.900	0.400	0.700	0.900	0.700	1.000	0.900
Llama-3.1-70B-Instruct	0.778	1.000	1.000	1.000	0.900	0.909	1.000	1.000	0.900	0.778	1.000	0.900	1.000	0.900	0.900	0.800	0.900	0.900	0.778	1.000
Gemma-2-9b-it	0.889	0.900	1.000	0.900	0.900	0.909	1.000	1.000	1.000	0.889	1.000	1.000	0.800	0.700	0.900	0.800	0.900	0.900	0.889	0.900
Gemma-2-27b-it	1.000	0.900	1.000	0.800	0.800	0.909	0.800	1.000	0.900	0.778	1.000	0.900	0.800	0.800	0.800	0.900	1.000	1.000	1.000	1.000
Qwen2.5-7B-Instruct	1.000	1.000	0.900	0.900	0.600	0.909	0.800	0.800	0.700	0.889	0.800	1.000	0.800	0.800	0.700	0.800	1.000	0.900	1.000	0.800
Qwen2.5-14B-Instruct	0.889	0.900	1.000	0.800	0.900	0.818	0.700	0.800	0.900	1.000	0.900	0.900	0.900	0.900	1.000	0.800	0.900	0.900	1.000	1.000
Qwen2.5-32B-Instruct	1.000	1.000	1.000	0.800	0.800	0.909	0.900	1.000	0.900	1.000	1.000	0.900	0.800	0.900	0.800	0.900	1.000	0.900	1.000	0.900
Aya-expanse-8b	0.889	1.000	1.000	0.700	0.600	0.727	0.700	0.800	0.900	0.556	0.500	0.900	1.000	0.600	0.700	1.000	0.900	0.800	1.000	0.900
Aya-expanse-32b	0.889	1.000	1.000	0.900	0.600	0.818	0.800	0.900	1.000	1.000	0.800	1.000	0.800	0.800	0.900	0.900	0.900	1.000	1.000	1.000
Mistral-7B-Instruct	0.889	0.800	0.900	0.700	0.300	0.636	0.500	0.600	0.700	0.556	0.400	0.300	0.800	0.500	0.400	0.700	0.700	0.500	0.667	0.700
Jais-family-6p7b-chat	0.556	1.000	0.900	0.600	0.800	0.455	0.700	0.500	0.900	0.556	0.900	0.700	0.600	0.700	0.500	0.800	0.900	1.000	1.000	0.900
Jais-family-13b-chat	0.889	1.000	0.900	0.600	0.500	0.636	0.900	0.700	1.000	0.889	0.500	0.800	0.800	0.800	0.400	0.700	0.800	0.700	1.000	0.700
Fanar-1-9B-Instruct	0.778	0.900	1.000	0.900	0.700	1.000	0.800	0.800	0.900	1.000	1.000	0.900	0.800	0.800	0.600	0.800	1.000	1.000	1.000	0.900
SILMA-9B-Instruct	0.778	0.900	1.000	0.900	0.600	0.818	1.000	0.800	1.000	0.889	0.800	1.000	0.800	1.000	0.700	0.900	1.000	1.000	1.000	1.000
ALLaM-7B-Instruct	1.000	0.900	1.000	0.900	1.000	0.545	0.800	0.900	1.000	0.667	0.900	0.900	0.800	0.900	0.700	0.900	0.800	1.000	0.889	0.800
AceGPT-v2-8B-Chat	0.778	0.700	0.900	0.800	0.400	0.545	0.800	0.800	0.800	0.778	0.600	0.800	0.900	0.900	0.500	0.800	0.900	0.700	0.778	0.900
Claude-Sonnet-4	1.000	1.000	1.000	0.900	1.000	1.000	1.000	1.000	1.000	0.889	1.000	1.000	0.900	0.900	1.000	1.000	1.000	1.000	1.000	1.000
Claude-3.5-Sonnet	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.900	1.000	0.900	1.000	1.000	1.000	1.000	1.000
Gemini-1.5-flash	0.889	0.900	1.000	1.000	0.800	0.909	1.000	0.900	1.000	1.000	0.900	0.900	0.900	0.900	0.800	0.900	1.000	1.000	1.000	1.000
Gemini-2.5-flash-lite	1.000	1.000	1.000	1.000	0.900	1.000	0.900	1.000	0.900	1.000	1.000	1.000	0.800	0.900	0.900	1.000	1.000	1.000	1.000	1.000
GPT-4o	1.000	0.900	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.900	0.800	1.000	1.000	1.000	0.800	1.000	1.000
GPT-4o-mini	0.889	1.000	1.000	0.800	0.600	0.909	0.900	0.900	1.000	1.000	1.000	0.900	1.000	0.800	0.800	1.000	1.000	0.800	1.000	1.000
Average	0.798	0.936	0.932	0.909	0.791	0.893	0.891	0.882	0.841	0.869	0.845	0.777	0.941	0.836	0.755	0.886	0.864	0.868	0.833	0.918

Table 14: Accuracy (↑) of models across Arabic dialects and countries (with the general prompt).

Model	EGY	MSA	UAE	JOR	MAU	PAL	ALG	SYR	IRQ	LEB	SAU	YEM	MOR	QAT	SUD	KUW	OMA	TUN	BAH	LIB
Llama-3.1-8B-Instruct	0.444	1.000	0.800	1.000	0.700	0.455	0.900	0.700	0.700	0.778	0.900	0.800	0.900	0.800	0.600	0.600	0.600	0.800	0.667	0.700
Llama-3.1-70B-Instruct	1.000	0.900	0.900	0.800	0.700	0.818	0.800	0.900	0.600	1.000	1.000	0.700	1.000	0.600	0.600	0.900	0.800	0.800	0.667	0.900
Gemma-2-9b-it	0.778	1.000	0.900	1.000	0.900	0.818	1.000	0.800	0.900	0.889	0.800	0.900	1.000	0.800	0.700	0.900	0.900	0.900	0.778	0.900
Gemma-2-27b-it	0.667	1.000	1.000	1.000	0.800	1.000	1.000	1.000	0.800	0.778	0.900	0.900	1.000	0.700	0.900	0.900	1.000	1.000	0.778	1.000
Qwen2.5-7B-Instruct	0.667	1.000	1.000	0.900	0.600	1.000	0.900	0.900	0.700	1.000	0.900	0.800	0.900	0.800	0.800	0.700	0.900	1.000	0.778	1.000
Qwen2.5-14B-Instruct	0.778	1.000	1.000	1.000	0.800	1.000	0.900	1.000	0.800	0.889	1.000	0.900	1.000	1.000	0.800	1.000	0.900	0.900	1.000	1.000
Qwen2.5-32B-Instruct	0.889	1.000	1.000	1.000	1.000	0.909	0.900	0.900	1.000	1.000	1.000	0.900	1.000	0.900	0.800	1.000	1.000	1.000	0.889	1.000
Aya-expanse-8b	0.667	0.800	0.800	0.800	0.800	0.818	0.700	0.900	0.700	0.889	0.500	0.700	0.900	0.600	0.700	0.900	0.800	0.700	0.778	0.900
Aya-expanse-32b	0.889	1.000	1.000	1.000	0.800	0.909	1.000	0.900	0.900	0.889	0.900	0.900	1.000	0.900	1.000	1.000	0.900	0.900	0.889	0.900
Mistral-7B-Instruct	1.000	0.500	0.800	0.500	0.600	0.909	0.700	0.500	0.600	0.556	0.500	0.300	0.700	0.400	0.300	0.800	0.700	0.600	0.444	0.500
Jais-family-6p7b-chat	0.778	0.900	0.900	0.700	0.700	0.545	0.900	0.900	0.600	0.667	0.800	0.700	0.900	0.800	0.700	0.900	0.900	0.700	0.778	1.000
Jais-family-13b-chat	0.556	0.800	0.900	0.900	0.700	0.909	0.900	0.900	0.800	0.778	0.800	0.600	0.900	0.800	0.700	0.800	0.700	0.600	0.889	0.700
Fanar-1-9B-Instruct	0.778	1.000	0.900	1.000	0.700	1.000	0.800	1.000	0.900	0.889	0.700	0.600	0.900	0.800	0.800	1.000	0.900	0.900	0.889	0.900
SILMA-9B-Instruct	0.889	1.000	0.800	1.000	0.800	0.818	1.000	0.900	1.000	1.000	1.000	0.700	0.900	0.800	0.700	0.900	0.800	0.900	0.667	0.900
ALLaM-7B-Instruct	0.889	0.900	0.900	0.900	0.800	0.818	0.700	0.900	0.800	0.778	1.000	0.800	0.900	0.900	0.600	0.900	0.700	0.900	1.000	0.900
AceGPT-v2-8B-Chat	0.778	0.700	0.900	0.800	0.400	0.909	0.800	0.800	0.700	0.778	0.300	0.800	0.800	0.800	0.700	0.900	0.800	0.900	0.667	0.900
Claude-Sonnet-4	1.000	1.000	1.000	1.000	0.900	1.000	1.000	0.900	1.000	1.000	1.000	1.000	1.000	1.000	0.900	1.000	1.000	1.000	1.000	1.000
Claude-3.5-Sonnet	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.900	1.000	0.900	0.900	1.000	1.000	1.000	1.000
Gemini-1.5-flash	0.778	1.000	1.000	0.900	0.900	0.909	0.900	0.900	0.900	0.889	0.900	0.800	1.000	0.900	0.800	0.900	1.000	0.900	0.667	1.000
Gemini-2.5-flash-lite	0.889	1.000	1.000	0.800	0.800	1.000	0.900	1.000	0.900	1.000	1.000	1.000	0.800	0.900	0.900	1.000	0.900	1.000	1.000	1.000
GPT-4o	1.000	0.900	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.900	0.800	1.000	1.000	1.000	1.000	1.000	1.000
GPT-4o-mini	0.889	1.000	1.000	0.800	0.900	0.909	0.800	0.800	1.000	0.889	0.800	0.600	1.000	0.800	0.800	0.800	0.800	0.700	1.000	1.000
Average	0.808	0.932	0.927	0.900	0.782	0.888	0.882	0.877	0.827	0.879	0.845	0.768	0.941	0.823	0.741	0.891	0.859	0.864	0.828	0.914

Table 15: Accuracy (↑) of models across Arabic dialects and countries (with incorrect explanations generated with SRL).

Model	EGY	MSA	UAE	JOR	MAU	PAL	ALG	SYR	IRQ	LEB	SAU	YEM	MOR	QAT	SUD	KUW	OMA	TUN	BAH	LIB
Llama-3.1-8B-Instruct	0.611	0.950	0.900	0.850	0.550	0.455	0.850	0.700	0.850	0.611	0.850	0.750	0.850	0.850	0.500	0.650	0.750	0.750	0.833	0.800
Llama-3.1-70B-Instruct	0.833	0.950	1.000	0.900	0.900	0.955	1.000	0.950	0.850	0.778	1.000	0.950	0.900	0.900	0.950	0.750	0.950	0.950	0.889	0.900
Gemma-2-9b-it	0.833	0.950	0.950	0.950	0.900	0.864	1.000	0.900	0.950	0.889	0.900	0.950	0.900	0.750	0.800	0.850	0.900	0.900	0.833	0.900
Gemma-2-27b-it	0.833	0.950	1.000	0.900	0.800	0.955	0.900	1.000	0.850	0.778	0.950	0.850	1.000	0.800	0.750	0.900	0.950	1.000	0.889	1.000
Qwen2.5-7B-Instruct	0.833	1.000	0.950	0.900	0.600	0.955	0.850	0.850	0.700	1.000	0.900	0.800	0.850	0.800	0.700	0.900	1.000	1.000	0.778	1.000
Qwen2.5-14B-Instruct	0.833	0.950	1.000	0.900	0.850	0.909	0.900	1.000	0.850	0.944	0.950	0.950	0.950	0.900	0.900	0.900	0.950	0.900	1.000	1.000
Qwen2.5-32B-Instruct	0.944	1.000	1.000	0.900	0.900	0.955	0.900	0.950	0.950	1.000	1.000	0.900	1.000	0.900	0.800	1.000	1.000	1.000	0.944	0.950
Aya-expanse-8b	0.778	0.900	0.900	0.750	0.700	0.773	0.700	0.850	0.800	0.722	0.500	0.700	0.900	0.600	0.700	0.900	0.800	0.700	0.778	0.900
Aya-expanse-32b	0.889	1.000	1.000	0.950	0.700	0.864	0.900	0.900	0.950	0.944	0.850	0.950	0.900	0.850	0.950	0.950	0.950	0.900	0.944	0.950
Mistral-7B-Instruct	1.000	0.500	0.800	0.600	0.450	0.773	0.600	0.550	0.650	0.556	0.450	0.300	0.750	0.450	0.350	0.750	0.700	0.550	0.556	0.600
Jais-family-6p7b-chat	0.667	0.950	0.900	0.650	0.750	0.500	0.800	0.700	0.750	0.611	0.850	0.700	0.750	0.800	0.650	0.750	0.900	0.850	0.889	0.950
Jais-family-13b-chat	0.722	0.900	0.900	0.750	0.600	0.773	0.900	0.800	0.900	0.833	0.650	0.700	0.850	0.800	0.550	0.750	0.750	0.650	0.944	0.700
Fanar-1-9B-Instruct	0.778	1.000	0.900	1.000	0.700	1.000	0.800	1.000	0.900	0.889	0.700	0.600	0.900	0.800	0.800	1.000	0.900	0.900	0.944	0.900
SILMA-9B-Instruct	0.833	1.000	0.900	0.950	0.700	0.818	1.000	0.850	1.000	1.000	1.000	0.700	0.900	0.800	0.700	0.900	0.800	0.900	0.667	0.900
ALLaM-7B-Instruct	0.944	0.900	0.950	0.900	0.900	0.682	0.750	0.900	0.900	0.722	0.950	0.850	0.900	0.900	0.650	0.900	0.750	0.950	0.944	0.850
AceGPT-v2-8B-Chat	0.778	0.700	0.900	0.800	0.400	0.727	0.800	0.800	0.750	0.778	0.450	0.800	0.800	0.800	0.700	0.900	0.850	0.800	0.722	0.900
Claude-Sonnet-4	1.000	1.000	1.000	0.950	0.950	1.000	1.000	0.950	1.000	1.000	1.000	1.000	1.000	1.000	0.900	1.000	1.000	1.000	1.000	1.000
Claude-3.5-Sonnet	0.944	1.000	1.000	1.000	0.950	1.000	1.000	1.000	1.000	1.000	1.000	0.950	0.950	0.950	0.900	0.950	1.000	0.950	1.000	1.000
Gemini-1.5-flash	0.833	0.950	1.000	0.950	0.850	0.909	0.950	0.900	0.950	0.944	0.900	0.850	0.950	0.900	0.800	0.900	1.000	0.950	0.833	1.000
Gemini-2.5-flash-lite	0.944	1.000	1.000	0.900	0.850	1.000	0.900	1.000	0.900	1.000	1.000	1.000	0.800	0.900	0.900	1.000	0.900	1.000	1.000	1.000
GPT-4o	0.944	0.950	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.900	0.800	1.000	1.000	1.000	1.000	1.000	1.000
GPT-4o-mini	0.889	1.000	1.000	0.800	0.750	0.909	0.850	0.850	1.000	0.944	0.900	0.750	1.000	0.800	0.800	0.900	0.900	0.750	1.000	1.000
Average	0.846	0.934	0.955	0.873	0.761	0.853	0.873	0.861	0.886	0.856	0.850	0.834	0.889	0.830	0.750	0.861	0.902	0.882	0.899	0.916

Table 16: Accuracy (↑) of models across Arabic dialects and countries (average of two runs).

E Ablation

Table 17 presents the relationship between model size and performance accuracy for open-source models ranging from 6.7B to 32B parameters on the MCQ Understanding task across different datasets, while Figure 23 provides the corresponding statistical correlation analysis illustrating the strength of this relationship. Across datasets, correlations were assessed using a significance threshold of $\alpha = 0.05$. For MAPS, the weak positive trend did not reach statistical significance ($p = 0.1053$), whereas adding context (MAPS + Context) resulted in a significant correlation ($p = 0.0086$), suggesting that contextualization enhances the link between model size and performance. Jawaher showed a borderline, yet non-significant, trend ($p = 0.0529$), while Kinayat demonstrated a statistically significant correlation ($p = 0.0178$), indicating that larger models more reliably benefit in this dataset.
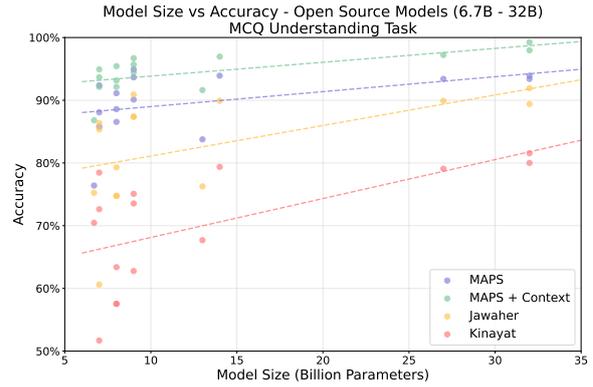
Similarly, Table 18 reports the relationship between model size and accuracy for the Pragmatic Use tasks in Kinayat, with Figure 24 visualizing the emerging correlation pattern, further supporting the observation that larger models generally exhibit stronger pragmatic competence in figurative language understanding. Here, model size demonstrates a moderate and statistically significant correlation with pragmatic competence ($R^2 = 0.600$, $p = 0.0007$), in contrast to a very weak but significant trend observed in MCQ Understanding ($R^2 = 0.265$, $p = 0.0497$) and Contextual Understanding ($R^2 = 0.265$, $p = 0.0495$). These findings suggest that larger models not only benefit more clearly from scale in pragmatic reasoning but also gain modestly in interpretive understanding tasks.

Figure 23: Model size vs accuracy ($\uparrow$) of open-source models (6.7B to 32B) on the MCQ Understanding task.

Figure 24: Model size vs accuracy ($\uparrow$) of open-source models (6.7B to 32B) on the Understanding and Pragmatic Use tasks.

F Sample Idioms

Table 19 presents a sample of idioms from the Kinayat dataset along with their corresponding explanations.

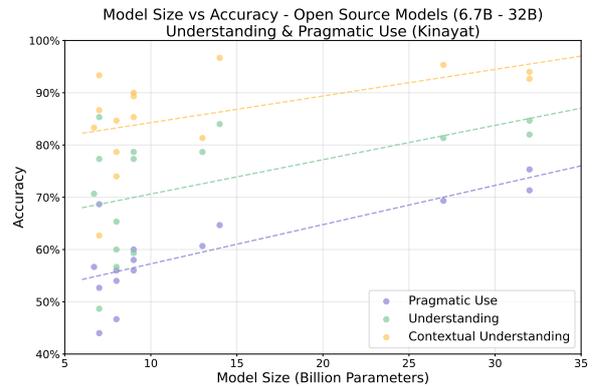| Task	R²	Slope	P-value	Interpretation
MAPS	0.189	0.0024	0.1053	Very weak correlation
MAPS + Context	0.424	0.0022	0.0086	Weak correlation
Jawaher	0.259	0.0049	0.0529	Very weak correlation
Kinayat	0.361	0.0062	0.0178	Weak correlation

Table 17: Statistical correlation analysis for different datasets: goodness of fit, slope, and significance testing.

Task	R²	Slope	P-value	Interpretation
Pragmatic Use	0.600	0.0075	0.0007	Moderate correlation
Understanding	0.265	0.0066	0.0497	Very weak correlation
Contextual Understanding	0.265	0.0051	0.0495	Very weak correlation

Table 18: Statistical correlation analysis for Kinayat: goodness of git, slope, and significance testing.

Idiom	Explanation
الدُّنْيَا بِتِضْرَبْ وِتِقْلَبْ	كناية عن كثرة ازدحام الناس في مكان وذهابهم ومجيئهم فيه.
رِجِعْ إيدْ وَرا وإيدْ قُدَّامْ	أي رجع يحرِّك ذراعيه ولا يحمل شيئًا، كناية عن الرجوع بالخيبة.
زَوِّدِ المِبَلَّهْ طِينْ	يريدون بالمبلة موضع نقع الكتان، وإذا زادها طينًا فقد زادها فسادًا، كناية عن زيادة الشيء الفاسد فسادًا.
ثَم عَلَى ضَهْرْ إيدُهْ	يقولون: ما شمتش على ضهر إيدي كناية عن عدم معرفة الغيب، أي لم يكن لهذا الخبر رائحة على ظهر يدي فأشمها فمن أين لي معرفته.
عِرِفْهَا وِهِيَّ طَايْرَهْ	المراد الكلمة يسرع قائلها بها، كناية عن شدة الذكاء، أي عرف ما يقال وأدرك معناه من أوَّل وهلة.
عِينِي عِينَكْ	كناية عن الجهر بالشيء بين الناس.
غِرِق في شِبْرْ مَيَّهْ	كناية عن الارتباك من العجز وقلة الحيلة.
لَتَّ وْعَجْن	كناية عن كثرة الكلام وتطويله وإعادة ما قيل، ومعنى اللت والعجن معروف.
لِسَانُه طَوِيلْ	كناية عن السفاهة والتطاول بالقذع على الناس وتعود ذلك، والمراد هنا الطول المعنويُّ.
مَا تِنْبَلِّش فِي بُقُّهْ فُولَهْ	البُق الفم، أي لا تبلُّ في فمه فولة، كناية عن عدم كتمان السرِّ والتسرُّع في إفشائه، فالكلمة لا تستقر في فمه كأنها الباقلاءة يسرع في إخراجها قبل أن تبلَّ بريقه.
مَاشِي عَلى قِشْرْ بِيض	كناية عن المتباطئ الحذر في مشيه، أي كأنه في تباطئه ماشٍ على بيض يخشى أن يكسر قشره بوطئه عليه.
نَشِّف الرِّيقْ	التنشيف عندهم: التجفيف، كناية عن المضايقة الشديدة بالمماطلة.

Table 19: Examples of Egyptian Arabic idioms from the Kinayat dataset and their explanations.

G Error Analysis

Tables 20, 21, and 22 summarize the most frequent error patterns observed across 22 models on the 150-idiom KINAYAT subset. Table 20 reports the top eight error types in MCQ idiom understanding before and after the addition of contextual sentences, while Table 21 focuses on the most common errors in MCQ contextual understanding. Finally, Table 22 contrasts the dominant pragmatic-use errors with those observed in MCQ understanding, highlighting differences in error distributions across evaluation tasks.

Idiom	MCQ	MCQ w/ Context
خَبَر أَبْيَض	10	10
سَكْرَة يَنِي	10	10
الصَباحْ رَبَاحْ	10	11
إيدْ مِنْ وَرَا وِإيدْ مِنْ قُدَأْمْ	9	3
وَلَع	9	1
عَنْدُه الدُنيَا بالْحُلْخَالْ	9	4
إلى حيثْ	8	7
حَطِ ضْبَاعُه فِي الشَقْ	8	4

Table 20: Most frequent errors in MCQ Understanding on the 150-idiom Kinayat subset, before and after adding contextual sentences (22 models).

Idiom	MCQ w/ Context
الصَباحْ رَبَاحْ	11
خَبَر أَبْيَض	10
سَكْرَة يَنِي	10
دَمُه يُلْطُشْ	9
عِينِي عِينَكْ	8
عَمَل البَحْرِ طْحِينَه	8
عَلَى قَفَاه	7
إلى حيثْ	7

Table 21: Most frequent errors in MCQ Contextual Understanding on the 150-idiom Kinayat subset (22 models).

Idiom	Pragmatic Use	MCQ
اِبْنْ حَرَامْ	15	0
أَشْكَرَه خَبَر	15	3
جَوَازِة نَصَارَى	15	4
إتْأْوِب عَ النَأْمُوش	14	3
يَا مَوْلَايْ كَمَا خَلَقْتِني	14	6
وَلَع لهْ قَنْدِيلْ	13	3
جَابْهَا في قُبِّتْه	13	4
عَمَل البَحْرِ طْحِينَه	13	6

Table 22: Comparison of the most frequent pragmatic-use errors against MCQ Understanding errors on the 150-idiom Kinayat subset (22 models).