1. Extended Experiment: Multi-Model Comparison and Pareto Trade-Off

To further explore efficiency and accuracy trade-offs, the same 3 × 3 prompt-type × temperature grid was evaluated on an additional local model, **Llama 3.2 (2 GB)**, alongside the original **Mistral 7B** baseline. Both models were tested under identical prompts, normalization, and scoring pipelines to ensure comparability.

Results Summary.
Llama 3.2 achieved notably lower latency (≈ 0.4 s mean) and, unexpectedly, higher accuracy on simpler emoji mappings (up to 60 % in the zero-shot 0.0 condition).
Mistral 7B remained slower (≈ 1.8 s mean) but more stable across prompting strategies.
Few-shot and JSON-constrained conditions maintained the same relative rankings as in the baseline experiment: few-shot prompting balanced accuracy and stability, while JSON-constrained prompting produced valid but semantically weaker outputs.
Accuracy was again largely insensitive to temperature, confirming that **prompt structure dominates over sampling randomness**.

Pareto Analysis.
When plotting accuracy versus mean latency, configurations from both models appeared on the Pareto frontier:

- **Llama 3.2** occupied the *fast-and-accurate* corner, offering rapid responses with surprisingly strong exact-match rates.

- **Mistral 7B** occupied the *balanced-reasoning* region, trading speed for steadier performance on ambiguous emoji sequences.

No single configuration optimized both metrics simultaneously, illustrating a clear **accuracy–latency Pareto trade-off** between smaller and larger local LLMs.

Interpretation.
These results suggest that smaller instruction-tuned models can outperform larger ones on lightweight symbolic tasks where pattern recognition outweighs deep reasoning. However, for longer or more context-dependent inputs, larger models may regain their advantage. This finding reinforces the importance of selecting model size and prompt design according to the computational and reasoning requirements of a local deployment.

2. Multi-Candidate Decoding via Fuzzy Matching

To evaluate whether stochastic diversity can recover near-miss predictions, the model generated $n = 5$ candidates for each emoji at a higher temperature (T = 1.0). Each candidate was compared to the gold title and its aliases using a fuzzy-string similarity metric (Levenshtein ratio), and the best-scoring candidate was selected as the final prediction.

Results.
The fuzzy-matching approach increased effective accuracy from ≈ 26 % (single-shot baseline) to ≈ 40 %, confirming that multiple stochastic samples can recover titles that single deterministic decoding missed. For example, correct titles such as *Jurassic Park*, *Apollo 13*, and *Ratatouille* were identified only among alternate samples. Latency rose roughly linearly with n (≈ 1–3 s per emoji), illustrating a clear **accuracy–efficiency trade-off**.

Interpretation.
The improvement shows that high-temperature sampling enables exploration of alternative phrasings and that lightweight post-selection through fuzzy matching can meaningfully boost local-LLM performance without retraining. However, the additional computational cost makes this technique more suitable for offline batch evaluation rather than real-time inference scenarios.

3. Context-Augmented Decoding (Emoji Glossary)

A final experiment tested whether providing a short emoji-to-meaning glossary as external context could help the model reason about symbolic inputs. A 30-entry glossary (e.g., 🕷️ = spider, 🧙 = wizard, 👩‍🚀 = astronaut) was prepended to the prompt, effectively creating a miniature Retrieval-Augmented Generation (RAG) setup.

Results.
While the glossary increased interpretive consistency, it reduced exact-match accuracy to ≈ 13 %. The model often produced literal translations—such as *"Man-Spider in City"*—instead of canonical movie titles like *Spider-Man*. Latency remained near baseline (≈ 1.2 s). These outcomes indicate that small local models may over-anchor to literal glossaries, prioritizing symbol definitions over contextual reasoning.

Interpretation.
The glossary provided helpful grounding but consumed attention that smaller models needed for inference. Consequently, responses became overly descriptive rather than abstract. Future work could explore more compact, high-level glossaries or semantic embedding retrieval to balance grounding with conceptual flexibility.