



# Mapping LLM Performance to Corporate Risk Needs

Jake Brulato

# Evaluation Has Become Complex For Risk



## **Model evaluation is increasingly difficult**

Growing GenAI model inventories have made risk review more complex. At Wells Fargo, inconsistent documentation and varied lifecycle stages across teams slow down validation efforts.



## **Manual review is unsustainable.**

Over **70 models** have been categorized, and new ones emerge regularly. Risk reviewers must still comb through **100+ page** validation reports—slowing down compliance and deployment.

# Creating Standardized Resources to Streamline the Process

## **Evaluate and Benchmark Open-Source LLMs**

- Determine optimal models (e.g., LLaMA, Mistral) suitable for corporate risk validation processes.

## **Develop Visual and Instructional Resources**

- Create accessible guides and prompting strategies for internal validation teams.

## **Explore Advanced Prompting Techniques**

- Adapt frameworks like Reason + Action (ReAct) using internal Retrieval-Augmented Generation (RAG) to enhance model reasoning within compliance guidelines.

# Core Strengths for Model Selection



## Model Suitability

- Long-term viability
- Aligned to avoid bias/toxicity
- Scalable for future needs



## Task Performance

- Handles long contexts
- Strong generation and reasoning
- Reliable under complex tasks



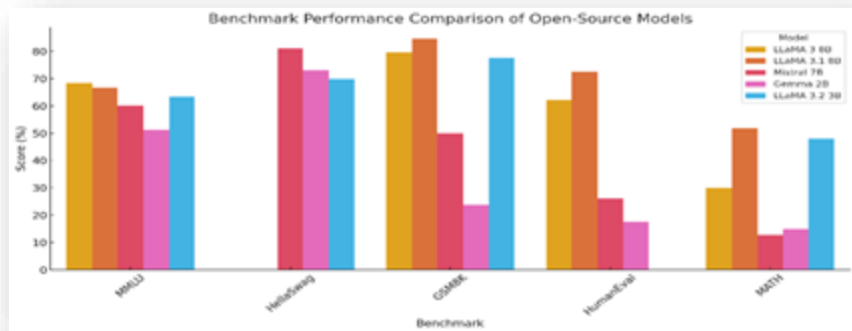
## Longevity

- Compatible with major frameworks
- Active research community
- Applicable use cases past its prime

# Best Models Identified Through Key Benchmarks

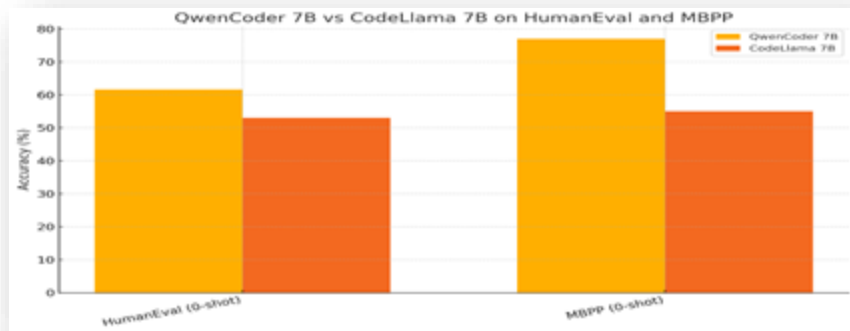
## Evaluated Key Open-Source LLMs:

- ✓ LLaMA series (Meta)
- ✓ Mistral models
- ✓ Gemma models (Google)
- ✓ Qwen series (Alibaba)



## Performance Benchmarks Used:

- 📊 **GSM8K**: Mathematical reasoning
- 💡 **HumanEval**: Code generation accuracy
- 💡 **HellaSwag**: Commonsense reasoning
- ÷ **MATH**: Advanced mathematical problem-solving
- 🐍 **MBPP**: Crowd-sourced Python programming problem



# Best Performing Models by Category

<u>Model</u>	<u>Category</u>	<u>Key Strength</u>
<u>LLaMA 3.1 8B Instruct</u>	Text Generation & Reasoning	<ul style="list-style-type: none"><li>• Top Scores:</li><li>• 84.5% GSM8K</li><li>• 72.6% <u>HumanEval</u></li><li>• 51.9% MATH</li></ul>
Qwen2-VL-Instruct 7B	Vision-Language Tasks	<ul style="list-style-type: none"><li>• Best image + text pairing</li><li>• Strong multilingual document QA</li><li>• Effective information extraction</li></ul>
<u>QwenCoder 7B</u>	Code Generation	<ul style="list-style-type: none"><li>• Code generation and debugging</li><li>• Documentation summarization</li><li>• Task automation support</li></ul>

# Organizing Model Insights with the Prompt Hub

- Organized key prompting parameters, errors, best practices, and task examples for open-source models.
- Logged token limits, generation issues, and recommended settings to guide validation workflows.
- Allowing validators and users in corporate risk to avoid any issues and decide which model is best for their current needs.

Models	Token Limits	Parameter Changes	Testing Errors	Best Prompting Practices	Paraphrase Prompt	Summarization Prompt	Grammar Check	Q/A Prompt
Llama 3 - 7B, 8B	• Llama 3: 8,192	Temperature: .7 top_p: .9 max_new_tokens: 10000 do_sample: True	<p>Llama 3 sometimes has failed to halt in its generation, making incomplete or too lengthy responses.</p> <p>Depending on what was prompted, the style control can be inconsistent, defining specifics with context examples gives it the ability to be more through in generation.</p> <p>Sometimes repeats generation, define in prompt to be unique and avoid repetition.</p>	<p>• Best to keep the prompt limited to a single task based on token length for text generation.</p> <p>• Create concise constraints to avoid overloading the prompt.</p> <p>• Error correction prompts help with paraphrasing and checking, Llama 3 handles these well.</p> <p>• Binary questions for accuracy to ensure their is clarity when making decisions.</p> <p>• If complexity is desired, make instructional prompts that clearly outline the structure of the task and the output.</p> <p>• Few Shot and Zero Shot prompting work well but anything beyond that like Chain-of-Thought prompting will underperform.</p>	<p>• Llama 3 and its instruct are specifically tuned for instructional tasks because of its robust instruction following ability in text generation.</p> <p>• Examples:</p> <p>• "Rewrite this text with [formal/creative/informal] tone: Original: "This device works efficiently under heavy load" Formal: "the device operates with exceptional efficiency under high workloads" Now rephrase this: "[Insert Text]"</p> <p>• "Reword this text in the style of the following examples: Example 1: "He quickly solved the problem" -&gt; "He resolved the issue in no time." Example 2: "Her voice was soothing the crowd" -&gt; "Her voice calmed everyone present" Now, paraphrase: "[insert text]"</p>	<p>• Strong contextual understanding allows it to adapt different styles and lengths, condensing it into concise summaries.</p> <p>• Examples:</p> <p>• "Summarize the text into a hierarchy of main points and subpoints. Use this format as a guide: Main point: The park is a popular destination. Subpoint: Know for its diverse wildlife. Subpoint: Offers scenic trails and paths. Now Summarize this text like it: [Insert Text]"</p> <p>• Summarize the following text into different lengths of 500/200/100 words:</p> <p>• Write an abstractive summary of the following text or extract the main important key sentences.</p>	<p>• Llama 3's pretraining allows it to have diverse grammar checking abilities and multi layer error detection.</p> <p>• Examples:</p> <p>• "Here's how grammar corrections are explained: Example: "She go to the market yesterday" Correction: "She went to the market yesterday" Explanation: The verb 'go' should be in the past tense to match the time indicator 'yesterday' Now check the grammar of this text and explain your corrections: [Insert Text]"</p> <p>• "Identify and fix grammar mistakes in this text without changing its meaning: "</p>	<p>• Examples:</p> <p>• "Answer questions using the guidance:  Example 1: Q: "What is the capital of France" A: "The capital of France is Paris" Example 2: Q: "How does photosynthesis work?" A: "Photosynthesis is the process by which plants convert sunlight into energy." Now answer this question: "What is the capital of France?"</p> <p>• "If the user's question lacks ask a clarifying question first, examples as guidance:  Example: Q: "What are the effects of climate change?" A: "Could you specify whether you're asking about global or regional effects?" Now answer this question: "What are the effects of climate change?"</p>

# Model Insights with Presentations – AIA Seminar

- Easier on the eyes, condensed into a format for users to view at specific points or glance where I'm pointing.
- Focuses on the performances and outputs from the seven main models.

## Llama 3 Vs. 3.1 Vs. 3.2 - Paraphrasing (Refined)

### Llama 3: Straightforward baseline, lacks breakdowns and nuanced explanations

The Central Analytics Support & Enablement (CASE) platform serves over 20,000 data professionals, offering features like the intent-driven chatbot CASEY, the collective intelligence framework, conditional ticketing, and MS Graph-enabled search. This unified platform enables product and support teams to efficiently address customer needs, manage documentation, and handle inquiries.

### Llama 3.1: Expanded and integrated descriptions with clear distinctions

Our enterprise support platform, Central Analytics Support & Enablement (CASE), serves over 20,000 data professionals, including data scientists, analysts, and developers. CASE features a chatbot, "CASEY," powered by MS Copilot Studio, as well as the Collective Intelligence framework, a conditional ticketing system, and MS Graph-enabled search. This integrated platform enables product and support teams to efficiently manage customer needs, document product knowledge, and respond to inquiries in a unified environment.

### Llama 3.2: Organized into structured format, making it more engaging

The Central Analytics Support & Enablement (CASE) platform serves over 20,000 data professionals, providing a unified solution for product and support teams. Key features include:

- CASEY, a non-Gen AI chatbot built on MS Copilot Studio
- The Collective Intelligence (CI) framework
- A conditional ticketing system
- MS Graph-enabled search

Strikes a balance between detail and simplicity, and changed format without instruction

- With a more specific goal in mind focusing on clarity and conciseness, we can see that more notable changes occur, specifically with format or length.

9

## Overall View - Llama 3 Vs. 3.1 Vs. 3.2

Key Features	Llama 3 - Instruct	Llama 3.1 - Instruct	Llama 3.2 - Instruct
Parameter Sizes	7B/8B	7B/8B	2B/3B
Token Limit	8192	128k	128k
Deployment	Cloud/Hardware	Cloud/Hardware	Cloud/Hardware
Use Cases	Chatbot/Content Generation	Decision Support, Complex Query Resolution	Interactive Services, small context or limited reasoning
Text Generation	Yes	Yes	Yes
Advanced Reasoning	No	Yes	Yes
Extended Context	No	Yes	Yes

14







# Model Insights with Presentations – Technical

- More dense, designed for users to deep dive in for more clarification and research, features the benchmarks and side by side comparisons.
- Filled with examples and technical errors from internal testing.

## Llama 3 - Overall Summary Review

- Model created by Meta, with a context length of 8,192 tokens that can be processed. Maximum number of tokens per response is 2048.
- Created April 4th, 2024, with high MMLU, G5M8K, and HumanEval.
- Provides strong performance in NLP tasks when asked to Summarize, Paraphrase, and Grammar Check.
- Likely because the llama 3 generation is newer, it is more robust in natural text generation near the same size as Mistral 3 7B from the last generation.
- Models from 10 months ago, like llama 2 and Mistral 3 7b do not perform the same.

	 GLOWS 1.0 BENCHMARK	 GLOWS 1.1 BENCHMARK	 GLOWS 1.2.30	 MYSTICAL BENCHMARK
<b>BRAC</b> Establishing GfL knowledge acquisition in open chat and few-shot settings	66.4%	62.7%	63.4%	62.7%
<b>BRAC</b> A suite of long math diagnosis and multistep benchmarks	Benchmark not available	Benchmark not available	Benchmark not available	Benchmark not available
<b>InfuGen</b> A challenging sentence completion benchmark	Benchmark not available	Benchmark not available	89.0%	89%
<b>CODE</b> Code-challenge math problems benchmark	76.6%	84.5%	77.7%	89%
<b>HumanEval</b> A dataset of human-written prompts for software engineering tasks	62.2%	72.6%	Benchmark not available	26.2%
<b>MAW</b> Benchmark performance on Math problems requiring use of context of difficulty and 7 sub-questions	30%	51.9%	40%	52.7%

## Side-By-Side Specifics – Llama Models

## Llama 3

- Considered the baseline, tries to include as much as possible.
- Has the ability to go further if prompted correctly, will keep a lot of jargon.
- Tends to keep it limited to simple structure text generation.
- Focuses on listing the features but makes it too long impacting readability.
- Works for simple information or clarification but will have more difficulty in long handing in specific contexts.
- Recommend:
  - Short paraphrasing, summarization, and grammar checking
  - Q/A and Q/AQC is possible but would best be done by another model.

## Llama 3.1

- Adds more depth to generation, describing specifics like user roles and the amount of people (20k data scientists, etc.)
- More clarity and better sentence structure in the output.
- Seamless transitions to other topics, making it more refined than Llama 3
- Highlights most of the technical features and their use cases.
- Most robust, handles large streams of data making very good for large documents.
- Recommend:
  - Most tasks, especially with large document streams chunked properly.
  - Q/A and QAQC works best along with the other prompt types.

## Llama 3.2

- Latest model with smallest parameter count, designed for resource constrained environments.
- Strikes a balance between detail and simplicity, making well thought and concise prompts.
- Language reasoning is good but will hallucinate if you don't specify that you want to keep the original text. (See Grammar Refined for example)
- A simpler less refined prompt worked better, keeping the original text with improvements.
- Recommendation:
  - Zero-Shot to Few-Shot concise prompts for paraphrasing and summarization.
  - Grammar Checking, Q/A, and QAQC best with Zero-Shot as those tend to hallucinate if too wordy.

# ReAct Prompting: Enhancing Reasoning and Action

- ReAct (Reason + Action) prompting combines logical reasoning ("Thought") with actionable steps ("Action"). Introduced by Yao et al. (2022), it improves model output quality.
- ReAct enables step-by-step reasoning and environment interaction (e.g., fact lookups) — boosting problem-solving depth and factual accuracy.
- ReAct achieves the highest Exact Match (EM) and accuracy among prompting strategies tested.
- Outperforms Chain-of-Thought (CoT) by a large margin on complex question answering tasks.

Prompt Method <sup>a</sup>	HotpotQA (EM)	Fever (Acc)
Standard	28.7	57.1
CoT (Wei et al., 2022)	29.4	56.3
CoT-SC (Wang et al., 2022a)	33.4	60.4
Act	25.7	58.9
ReAct	27.4	60.9
CoT-SC → ReAct	34.2	<b>64.6</b>
ReAct → CoT-SC	<b>35.1</b>	62.0
<b>Supervised SoTA<sup>b</sup></b>	67.5	89.5

Source: Adapted from Yao et al. (2022)

# Recreating ReAct Prompting Within Internal Constraints

- Problem:  
Internal restrictions prevent using external tools for "Action."
- Solution:  
Developed an internal Retrieval-Augmented Generation (RAG) system to replicate search functionality.
- Framework replicated ReAct structure but functioned more like CoT due to limitations.
- Shared RAG logic across teams to enhance their implementations.

### Thought

The task seems straightforward - comparing the key features of two plans, specifically focusing on how they handle healthcare costs. Upon reviewing the given text, I notice that there's some overlap between the HSA Plan and the Copay Plan with HRA regarding their structure and payment mechanisms. This implies that understanding how these similarities differ might hold the key to identifying the primary distinctions between the two plans.

### Action: Step 1

I'll start by looking into the way the plans deal with annual deductibles and out-of-pocket expenses. It appears that for the HSA Plan, charges for in-network services do not affect the annual deductible and out-of-pocket limits when compared to out-of-network expenses. On the contrary, the Copay Plan with HRA allows for cross-accumulation between its in-network Tiers 1 and 2 deductibles.

### Observation

This comparison suggests that the HSA Plan maintains separate deductibles and out-of-pocket limits for in-network and out-of-network expenses, whereas the Copay Plan with HRA combines them through cross-accumulation.

### Action: Step 2

Considering the unique characteristics of each plan, my next move would be to examine how prescription drugs fit into their respective frameworks. According to the provided content, the HSA Plan requires meeting the annual deductible before applying copays to prescription medications, except for those listed under preventive drug therapy. In contrast, the Copay Plan with HRA does not require meeting the deductible before copays apply to prescription medications.

### Observation

These findings indicate that the HSA Plan imposes stricter requirements for accessing prescription medication benefits than the Copay Plan with HRA, where copays begin immediately without needing to satisfy a deductible.

### Final Answer

In summary, the major differences between the Copay Plan with HRA and the HSA Plan lie in their approaches to handling annual deductibles and out-of-pocket expenses, particularly when it comes to combining in-network and out-of-network expenses. Additionally, the HSA Plan places stricter conditions on accessing prescription medication benefits compared to the Copay Plan with HRA. These distinct features highlight the varying strategies employed by each plan to manage patient costs and ensure healthcare accessibility.assistant

Sources:

[1] Relevance score: 0.53

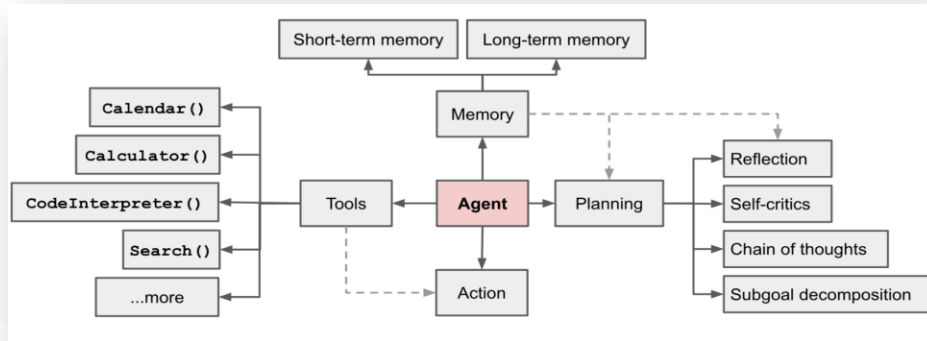
Preview: Once enrolled in COBRA, let BenefitConnect™ | COBRA know about any changes in addresses of family members (see "Plan contact information" below). You should also keep a copy, for your records, of any notices you send to the COBRA Administrator

[2] Relevance score: 0.53

Preview: To receive this second qualifying event extension of COBRA continuation coverage, BenefitConnect™ | COBRA must be notified of the second qualifying event within 60 days after the second qualifying event occurs to maintain extension rights under COBRA. To notify BenefitConnect™ | COBRA of the second qualifying event, call 1-877-29-COBRA (1-877-292-6272) (858-314-5188 International only) or online at <https://cobra.ehr.com>. Failure to notify BenefitConnect™ | COBRA within 60 days of the second qualifying event will make the qualified beneficiary ineligible for the extension rights under COBRA. Are there other coverage options besides COBRA continuation coverage? Yes

# Next Steps: Recreating Agentic Workflow Internally

- Adapt agent workflows internally to meet security and compliance requirements.
- Integrate memory and planning modules using internal tools and retrieval systems.
- Simulate tool usage (e.g., search, calculators) without external API dependencies.
- Enable agent reasoning patterns like reflection, subgoal decomposition, and self-critique.



Source: Adapted from Weng (2023)

Questions?

# References

- Yao et al., 2022. *ReAct: Synergizing Reasoning and Acting in Language Models*. (arXiv: [2210.03629](https://arxiv.org/abs/2210.03629))
- Weng, L. (2023, June 23). *LLM-powered autonomous agents*. Lilian Weng's Blog. <https://lilianweng.github.io/posts/2023-06-23-agent/>
- Rozière, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., Rapin, J., Kozhevnikov, A., Copet, J., ... Synnaeve, G. (2023). *Code Llama: Open foundation models for code* (arXiv preprint arXiv:2308.12950). arXiv. <https://arxiv.org/pdf/2308.12950>
- Qwen Team. (2024, March 7). *CodeQwen 1.5: Advancing code generation and understanding*. Qwen Blog. <https://qwenlm.github.io/blog/codeqwen1.5/>
- Prompt Hackers. (2024). *Llama 3 8B vs Claude 3.5 Sonnet: Model comparison*. Prompt Hackers. <https://www.prompthackers.co/compare/llama-3-8b/claude-3.7-sonnet>