

Which is the best free AI model for Excel formulas: Gemma 4, GPT-4o, or Llama 4?

Google's Gemma 4, OpenAI's GPT-4o (free version), and Meta's Llama 4 all represent a new generation of AI models capable of handling Excel formula requests with surprising efficiency.

If you work professionally with Excel, you've probably experimented with AI-powered formula generation. ChatGPT is often recommended, and it's actually quite good. But ChatGPT isn't the only option anymore, and it's not always the best choice—especially if you want something free, private, or that can run on your own hardware.

Google's Gemma 4, OpenAI's GPT-4o (free version), and Meta's Llama 4 all represent a new generation of AI models capable of handling Excel formula requests with surprising efficiency. After spending the past two weeks running structured tests on all three to find out which model truly delivers the most reliable Excel results, this guide shares the findings, including the exact statements used and the formulas each model returns.

Direct Comparison: 5 Practical Excel Formula Problems

Each model is given five identical problems, using the same wording and context. Each problem describes a real-world Excel scenario with column references and expected results. Results are evaluated based on three criteria: Formula accuracy (does it produce the correct result?), quality of explanation (does the model provide a clear and logical explanation?), and ability to handle exceptions (does it account for empty cells, errors, or unusual input?).

Test 1: SUMIFS function with multiple conditions

Prompt:

Tôi có một bảng doanh số bán hàng trong Excel. Cột A là tên người bán hàng, cột B là khu vực (North, South, East, West), cột C là danh mục sản phẩm và cột D là số tiền bán hàng. Viết công thức cho ô F2 để tính tổng doanh số bán hàng tại khu vực North cho danh mục Electronics.

Gemma 4: Returns `=SUMIFS(D:D, B:B, "North", C:C, "Electronics")`. Correct the first time. Clearly explains how SUMIFS evaluates multiple criteria at once. Also suggests using table reference formatting as an alternative.

GPT-4o : Returns =SUMIFS(D:D, B:B, "North", C:C, "Electronics"). The exact same formula. Added a longer explanation with step-by-step analysis of each argument. Also proactively mentioned that SUMIFS is not case-sensitive.

Llama 4 : Returns =SUMIFS(D:D, B:B, "North", C:C, "Electronics"). Correct. The explanation is more complete but concise than the other two options. It doesn't mention case sensitivity or alternatives.

Conclusion : All three answered this question correctly. SUMIFS is widely used in training data, so this is easy to understand. GPT-4o provided the most thorough explanation.

Test 2: Nested IF functions for classification

Prompt :

C?t E ch?a ?i?m thi (0-100). Vi?t công th?c cho ô F2 ?? phân lo?i t?ng ?i?m: 90 tr? lên là Distinction, 75-89 là Merit, 60-74 là Pass, d?? i 60 là Fail.

Gemma 4 : Returns =IF(E2>=90, "Distinction", IF(E2>=75, "Merit", IF(E2>=60, "Pass", "Fail"))). Exactly. It also suggests using IFS as a neater alternative for Microsoft 365 users and mentions wrapping the formula in IFERROR to handle empty cells.

GPT-4o : Returns a similar nested IF structure. Also proposes an IFS scheme. Goes further by proposing a switch-based method and explaining why nested IF functions evaluate from top to bottom, which is important for thresholding.

Llama 4 : Returns a nested IF function correctly. It doesn't mention IFS or SWITCH. The explanation is fairly basic but doesn't address exceptions like empty cells or non-numeric data.

Conclusion : All are correct. Gemma 4 and GPT-4o both provide suggestions for handling practical exceptions. The explanation in GPT-4o is the most educational.

Test 3: INDEX-MATCH with multiple criteria

Prompt :

Tôi có m?t b?ng giá. C?t A là tên s?n ph?m, c?t B là kích th?? c (Small, Medium, Large), và c?t C là giá. Vi?t công th?c cho ô G2 ?? tra c ?u giá c?a s?n ph?m trong E2 và kích th??c trong F2. Có th? có các tên s?n ph?m trùng l?p v?i các kích th??c khác nhau.

Gemma 4 : Returns the formula =INDEX(C:C, MATCH(1, (A:A=E2)*(B:B=F2), 0)) along with a note stating that it is an array formula and needs to be entered using the **Ctrl + Shift + Enter** key combination in older versions of Excel, or it will work correctly in Microsoft 365. Accurate and clearly explained.

GPT-4o : Returns the same INDEX-MATCH method. Also suggests an XLOOKUP option using auxiliary columns and a FILTER-based method for Microsoft 365. Very thorough, but the additional options can be overwhelming for beginners.

Llama 4 : Returns the formula `=INDEX(C:C, MATCH(E2&F2, A:A&B:B, 0))`. This string concatenation method works but is less robust – it can produce false matches when product names and sizes match the same string (e.g., "LargeBox" + "Small" matches "Large" + "BoxSmall"). The model does not indicate this limitation.

Conclusion : Both Gemma 4 and GPT-40 produce a more robust multiplication-based array method. Gemma 4's string concatenation method works in most cases but has a minor bug it doesn't address. This is a critical exception type in production spreadsheets.

Test 4: Extract text between delimiters

Prompt :

C?t A ch?a mã ??n hàng ???c ??nh d?ng nh? 'ORD-12345-UK'. Vi?t công th? c cho ô B2 ?? trích xu?t ch? ph?n s? (12345) t? mã, gi? s? ??nh đ? ng luôn là TI?N T?-S?-H?U T? v?i d?u g?ch ngang làm d?u phân cách.

Gemma 4 : Returns `=MID(A2, FIND("-",A2)+1, FIND("-",A2,FIND("-",A2)+1)-FIND("-",A2)-1)`. Exactly. Clearly explains the nested FIND logic, noting that the second FIND statement starts searching after the first hyphen.

GPT-4o : Returns the exact same MID-FIND formula. Also suggests a TEXTSPLIT solution for Microsoft 365: `=TEXTSPLIT(A2, "-", , 2)` to get the second segment. Additionally, it suggests using the VALUE() function to convert the extracted text to numbers if needed for calculations.

Llama 4 : Returns the same MID-FIND formula but with slightly different variable names in the explanation, making it harder to follow. The formula itself is correct.

Conclusion : All three provide accurate formulas. GPT-4o's TEXTSPLIT proposal is a truly helpful addition for Microsoft 365 users. Gemma 4's explanation is the clearest.

Test 5: Monthly Reset Cumulative Total

Prompt :

C?t A ch?a ngày tháng và c?t B ch?a s? ti?n doanh thu hàng ngày. Vi?t công th?c cho C2 t?o t?ng l?y ??k? doanh thu reset v? 0 vào ??u m? i tháng.

Gemma 4 : Returns a cumulative total `=SUMIFS(B$2:B2, MONTH(A$2:A2), MONTH(A2), YEAR(A$2:A2), YEAR(A2))`. Accurate and elegant. The extended range with a fixed starting row creates a cumulative total, and the MONTH/YEAR criterion ensures it's reset every month. It's also worth noting that the data must be sorted by date for this to work correctly.

GPT-4o : Returns a method based on SUMPRODUCT: `=SUMPRODUCT((MONTH(A$2:A2)=MONTH(A2))*(YEAR(A$2:A2)=YEAR(A2))*B$2:B2)`. Also correct, and avoids the potential complications of array formulas in older versions of Excel. Provided a good explanation but did not mention the requirement to sort by date.

Llama 4 : Returns `=SUMPRODUCT((MONTH(A2 : A2) =MONTH(A2)) * (B2 : B2))`. This formula ignores the YEAR check, meaning it will incorrectly match January 2025 with January 2026 if the data spans multiple years. This is a significant error for any dataset longer than 12 months.

Conclusion : Gemma 4 is the cleanest and most complete model. GPT-4o is accurate but deficient in alignment. Llama 4 has a serious logic error. This test revealed the largest quality gap between the models.

Which model handles exceptions better?

The real difference lies in the ability to handle exceptions. Simple formulas—SUMIFS, basic IF, VLOOKUP—work well on all three models. Problems arise when the requirements involve multi-step logic, date constraints, potential data inconsistencies, or version-specific functions.

Gemma 4 is most consistent in proactively alerting to potential problems. When creating a cumulative sum formula, it noted the need to sort by date without being asked. When using an array formula template, it mentioned the **Ctrl + Shift + Enter** requirement for older versions of Excel. This contextual awareness makes a real difference when you're building spreadsheets that others will use.

GPT-4o excels at providing alternative methods. For almost any request, it offers two or three ways to solve the same problem, often including modern Microsoft 365 functions along with versions compatible with older versions. This is incredibly useful if you're working with teams that have multiple versions of Excel. However, it sometimes misses practical considerations (like sorting requirements) while still maintaining syntactic completeness.

Llama 4 is weakest in exceptional cases. The string concatenation method for INDEX-MATCH and the lack of YEAR checking in the cumulative total are seemingly correct errors that actually cause real problems in production. If you use Llama 4 for Excel formulas, plan to scrutinize all output more carefully.

Which free AI model is the most accurate for Excel formulas?

In testing across five common formula types, both Gemma 4 and GPT-4o (free version) scored highly in accuracy. Gemma 4 performed better with complex nested logic and multi-criteria lookups, while GPT-4o offered slightly better explanations in simpler language. Llama 4 competed well with simpler formulas but struggled with exceptions in advanced scenarios.

You finished reading the article "**Which is the best free AI model for Excel formulas: Gemma 4, GPT-4o, or Llama 4?**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.