Can Large Language Models Generalize Analogy Solving Like Children Can?

Claire E. Stevenson[1], Alexandra Pafford[1], Han L. J. van der Maas[1,2], Melanie Mitchell[2]

[1]Psychological Methods, University of Amsterdam, the Netherlands
[2]Sante Fe Institute for Complexity, Sante Fe, AZ, USA
c.e.stevenson@uva.nl, h.l.j.vandermaas@uva.nl, mm@santafe.edu

Abstract

In people, the ability to solve analogies such as ''body: feet:: table: ?'' emerges in childhood, and appears to transfer easily to other domains, such as the visual domain ''(:) :: < : ?''. Recent research shows that large language models (LLMs) can solve various forms of analogies. However, can LLMs generalize analogy solving to other domains like people can? To investigate this, we had children, adults, and LLMs solve a series of letter-string analogies (e.g., a b : a c :: j k : ?) in the Latin alphabet, in a near transfer domain (Greek alphabet), and a far transfer domain (list of symbols). Children and adults easily generalized their knowledge to unfamiliar domains, whereas LLMs did not. This key difference between human and AI performance is evidence that these LLMs still struggle with robust human-like analogical transfer.

1 Introduction

You may be familiar with the analogy ''consciousness is like an iceberg''. Here, people intuitively infer the below-the-surface depth and complexity of consciousness by relating it to an iceberg, whose mass is mostly found under water, just as our subconscious dwells under our conscious minds. This ability emerges in childhood (Goddu et al., 2020; Gentner, 1988; Stevenson and Hickendorff, 2018). However, it is a subject of debate whether analogical reasoning has emerged in Large Language Models (LLMs) (Webb et al., 2023; Lewis and Mitchell, 2025; Hodel and West, 2023; Webb et al., 2024). More importantly, are LLMs able to solve analogies at this level of conceptual abstraction and generalize to novel domains (Mitchell, 2021; Shiffrin and Mitchell, 2023)? In this study, we investigate analogical transfer at two levels of abstraction (near and far), and compare LLM performance not only to adults, but also to children, who are still developing analogical reasoning abilities. We ask the question: Can LLMs can generalize analogy solving like children can?

Analogical reasoning, the process of applying a known concept to understand something new through relational similarity, is fundamental to the way people think and learn (Holyoak, 2012; Gentner and Hoyos, 2017). This is because we humans can easily generalize—that is, transfer principles discovered in one domain to new domains that share varying degrees of similarity with the original (Doumas et al., 2022). This can be principles in near contexts that are similar in terms of concrete attributes (e.g., shape, ''a pyramid is like an iceberg'') or in farther contexts that are only similar in terms of abstract relations (e.g., abstraction of depth, ''consciousness is like an iceberg'') (Barnett and Ceci, 2002). Near analogies tend to be easier for both adults and children to solve than far analogies (Johnson et al., 2025; Jones et al., 2022; Thibaut and French, 2016). And, in general, adults are better at solving analogies than children. But, when the required domain knowledge and a causal framing are present, then children can solve analogies such as ''body is to feet as table is to ?'' as early as the 3–4 years-old (e.g., Goddu et al., 2020; Goswami, 1991). And when analogies are presented in a more challenging or far context, young children tend to revert to associative strategies, e.g., replying 'egg' to 'dog is to doghouse as chicken is to ?' instead of 'chicken coop' (Stevenson and Hickendorff, 2018; Gentner, 1988; Thibaut and French, 2016).

There are many tasks used to study analogical reasoning and transfer in people, from verbal to geometric to scene analogy problems (e.g., Ichien et al., 2020; Richland et al., 2006; Mulholland

Transactions of the Association for Computational Linguistics, vol. 14, pp. 612–626, 2026. https://doi.org/10.1162/TACL.a.614
Action Editor: Benoit Crabbé. Submission batch: 5/2025; Revision batch: 9/2025; Published 4/2026.
© 2026 Association for Computational Linguistics. Distributed under a CC-BY 4.0 license.

et al., 1980). However, many of these tasks are either not suitable for children (e.g., verbal analogies may contain unfamiliar words or relations for children) or to LLMs (e.g., visual analogies designed for children are still difficult for today's multimodal models [Yiu et al., 2024]). Therefore, we need a domain that is text-based, but doesn't require domain knowledge beyond what a typical child or LLM would know. Letter-string analogies fit the bill as they require very little domain knowledge and offer an idealized scenario to examine analogical reasoning in a ''pure, uncontaminated way'' (Hofstadter, 1984, p. 3). In these puzzles, a string of letters is transformed according to one or more rules, and the task is to use analogy and apply the same transformations to a new string. For example, ''If abc changes to abd, what should pqr change to?'' (Mitchell, 2021).

Letter-string analogy solving has been studied in human adults and LLMs. For example, Webb et al. (2023) showed that GPT-3 is able to solve letter-string analogies better than college students. Lewis and Mitchell (2025) showed that GPT-models solved letter-string analogies at about 60% accuracy in the Latin alphabet domain, somewhat below the level of adults they tested. Interestingly, Lewis and Mitchell (2025) and Hodel and West (2023) found that GPT-3's performance degraded when presented with these same analogies using an alphabet of shuffled letters. Moreover, Lewis and Mitchell (2025) showed that GPT-models had great difficulty solving letter-string analogies in an unfamiliar alphabet of symbols, whereas people did not. As such, there is conflicting evidence of whether LLMs can generalize analogy solving to novel domains (Lewis and Mitchell, 2025; Webb et al., 2024; Hodel and West, 2023), something that comes easily to adults (e.g., Thibaut et al., 2022; Doumas et al., 2022), and that even children appear capable of when domains share structural similarities (Chen, 1996; Gentner and Toupin, 1986; Bobrowicz et al., 2020; Holyoak et al., 1984). Thus, while there is some evidence to suggest that LLMs can solve letter-string analogies at around the same level as people, it is unclear whether these models understand the problem and are actually using analogical reasoning (Opiełka et al., 2024; Johnson et al., 2025; Moskvichev et al., 2023).

In this study, we investigate whether LLMs can generalize analogy solving to new domains like adults and 8-year-old children can at two levels of abstraction. To this end, we compare how adults, children, and LLMs generalize analogy solving on the letter-string task to both near (Greek alphabet) and far (Symbol list) domains.

2 Method

We compared 42 children (7–9 year-olds), 62 adults, and 55 runs of each of four LLMs (Claude-3.5 Anthropic [2024]; Gemma-2 27B Gemma Team [2024]; GPT-4o OpenAI [2023]; and Llama-3.1 405B Touvron et al. [2023]) on a set of letter-string analogies under three alphabet conditions: Latin, Greek, and a Symbol list.

2.1 Materials

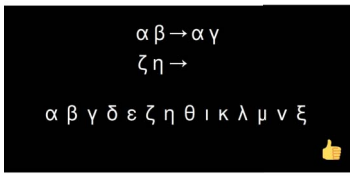
2.1.1 Letter-String Analogy Task

Letter-string analogies, pioneered by Hofstadter (1984), are a type of proportional analogy (A is to B as C is to D) involving alphabetic strings. For example, ''If the string of letters *abc* changes to *abd*. How would you change the string *pqrs* in the 'same way'?'' (Mitchell, 2021). Such letter-string analogies can be solved in multiple ways. For example, shifting the last letter and responding *pqrt* is what people tend to prefer (and what we consider ''correct'' in this context). However, another possible solution could be *pqrd*, where a literal rule is applied, namely, replacing the last letter with *d*.

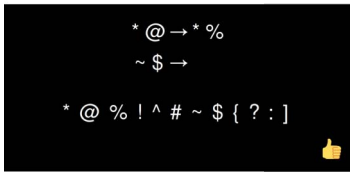
Rules. There are several possible transformations from A to B and generalizations from A to C as described in Webb et al. (2023). We use only the simplest transformations of successor (one and two after), predecessor (one before) and repetition, and the generalizations are limited to shifting in the alphabet and letter repetitions—rules that children are expected to be familiar with.

Alphabets. For each of the items in the Latin alphabet we also created a near transfer version using the Greek alphabet and a far transfer version in our invented Symbol alphabet: ∗ @ % ! ^ # ~ $ { ? = : (see Figure 1 for an example). We chose the Greek alphabet as near transfer domain because Greek symbols are somewhat visually similar to the Latin alphabet, but otherwise unfamiliar to the children in our study. We presented actual Greek symbols to humans, but chose the written version (alpha, beta, etc.) for LLMs based on their ability to list the Greek alphabet in this form upon request. We chose to use an ordered list of

(a) Latin



(b) Greek



(c) Symbol

Figure 1: Letter-string analogy task item 1 in (a) baseline alphabet: Latin, (b) near transfer alphabet: Greek, and (c) far transfer alphabet: Symbol.

Symbols for far transfer, because it is an unfamiliar 'alphabet' that neither people nor the LLMs had seen before in this context, but at the same time were both able to process (i.e., the children can identify differences visually and for the LLMs these are common symbol keys on a keyboard). The constructed items for each alphabet were kept consistent, where the same transformations and generalizations from item 1 of the Latin alphabet were also used for item 1 of the Greek and Symbol alphabets. See Table 1 in the Results section for an overview of all items.

2.1.2 Human Data Collection

Procedure. Both children and adults completed the task in a browser. They were first shown the Latin alphabet and told that they would solve puzzles with these. For adults there was a simple example with feedback as the study was carried out fully online. For children, the interface was explained and demonstrated in person. Participants then solved two simple practice items without feedback (used to ensure understanding of task). Then for each alphabet, they were shown the list of letters/symbols and told they would again solve puzzles using these letters/symbols, where the Greek and Symbol alphabets were referred to as ''secret code'' letters for children. There were five items for each alphabet, with 15 items total.

Adults. We collected adult data online from fluent English speakers through the Prolific research participant recruitment platform. We recruited 68 adults of 18 years or older (M = 24.0 years, SD = 7.33 years, 50% female) who had completed secondary education or higher and resided in the Netherlands or neighboring countries (as children were recruited in the Netherlands). We also required that they have no language disorders and have (corrected-to-) normal vision to ensure they could see/process the task, that they use a device at least 2× a week (to ensure digital fluency), and that they have a 95% or higher approval rating on Prolific to ensure high quality data from the participants. Based on the pre-registered exclusion criteria for adults (answering >80% of items), 6 adults were excluded.

Children. Data was collected from 44 children (7–9 year-olds, M = 8.26 years, SD = 0.67 years) at a local school on an electronic tablet. The recruited school is a public Montessori school and emphasizes natural materials and does not use tablets or computers in this age group. All children from the participating classrooms were included in the study, as language disorders are generally not yet assessed in this age group in the Netherlands. The researchers gave spoken instructions given the limited reading abilities in this age group. The children then completed the task independently. We excluded two children, because they did not complete the task.

2.1.3 LLM Data Collection

We collected data from LLMs from four types of models: Anthropic's Claude-3 and Claude-3.5; Google's Gemma-2-9B and Gemma-2 27B; Open AI's GPT-3, GPT-3.5, GPT-4, and GPT-4o; and Meta's Llama-3.1-8B, Llama-3.1-70B, Llama-3.1-405B. For each model type, the newest and largest model had the best performance. Therefore, to provide clear and concise results our main results comparing human and LLM performance report on this selected set of models: Anthropic's Claude-3.5, Google's Gemma-2 27B, Open AI's GPT-4o, and Meta's Llama-3.1 405B. A brief overview of the results of other models can be found in Section 4; the full dataset is available in our GitHub repository.

Procedure. We presented the analogies in chat completion mode using Python APIs from Anthropic for Claude models, from Open AI for GPT

Item ID	Alphabet	A	B	C	D	AB Rule	AC Rule
A	Practice	a	b	j	k	successor_1	shift
B	Practice	c d	c d d	j k	j k k	repeat_1	shift
1	Latin	a b	a c	g h	g i	successor_1	shift
2	Latin	c d	c c e e	m n	m m o o	successor_1, repeat_2	shift
3	Latin	e f	e h	k l	k n	successor_2	shift
4	Latin	d e	d f f	g h	g i i	successor_1, repeat_1	shift
5	Latin	c d	b d	m m n n	l l n n	predecessor_1	shift, repeat_2
1	Greek	α β	α γ	ζ η	ζ ϑ	successor_1	shift
2	Greek	γ δ	γ γ ϵ ϵ	κ λ	κ κ μ μ	successor_1, repeat_2	shift
3	Greek	ϵ ζ	ϵ ϑ	ι κ	ι μ	successor_2	shift
4	Greek	η ϑ	η ι ι	λ μ	λ ν ν	successor_1, repeat_1	shift
5	Greek	β γ	α γ	ν ν ξ, ξ	μ μ ξ, ξ	predecessor_1	shift, repeat_2
1	Symbol	* @	* %	∼ $	∼ {	successor_1	shift
2	Symbol	% !	% % ^ ^	= :	= =))	successor_1, repeat_2	shift
3	Symbol	@ %	@ ^	# ∼	# {	successor_1	shift
4	Symbol	! ^	! # #	$ {	$ = =	successor_1, repeat_1	shift
5	Symbol	^ #	! #	= = : :	{ { : :	predecessor_1	shift, repeat_2

Table 1: Base item set administered to adults, children, and LLMs.

models and from Together AI for the remaining models, which are all open source. We specified a temperature of 0 for near-deterministic data collection and set the maximum number of tokens to 10.

Prompt. The LLM general instruction was as follows: `We are going to do puzzles with the letters or symbols '[Latin alphabet|Greek alphabet|Symbol list]'. Example 'if a changes to b, then j changes to k'.`

Per item the LLMs received the instruction and item as follows: `The [letter|symbol] list is '[Latin alphabet|Greek alphabet|Symbol list]'. If [A] changes to [B], what does [C] change to ?`

Pre-pending Previous Conversation. Also, following Webb et al.'s (2023) approach to administering verbal analogies and digit matrices, all previous conversation with the LLM was pre-pended to each successive item so that the models could learn while testing just as people could. This seemed especially important because the exact same items with the same rules were applied in the same order from one alphabet set to the next. We also ran the tasks without pre-pending previous conversation, which generally resulted in lower LLM performance (see Appendix C).

Prompting Templates. We tested 5 different prompt templates for presenting items to LLMs, as prompt engineering can change the LLMs' performance on the task. Results are reported for the best performing template as shown above: `If A changes to B, what does C change to?` See Table 8 for more details on the different templates and results.

Differences between Human and LLM Procedures. To keep the conditions for the LLM data collection similar to that of people and, especially to fairly compare LLM results to those of children, we presented all analogies in a zero-shot setting using the same instructions that we spoke to the children. There were two exceptions. First, with children we referred to the Greek and Symbol alphabets as ''secret code letters'', whereas this was 'alphabet' and '(ordered) list', respectively, for adults and LLMs. Second, the LLMs received the worked example 'if a changes to b, then j changes to k' that humans did not receive.

Item Variants for LLMs. To enable robust comparisons between individual LLMs and groups of people, we adopted a similar methodology to Webb et al. (2023) and administered approximately as many variants of the task to each LLM as we had people who solved it. To do so, we created variants of each item by systematically shifting all of the characters in the item. For example, ''a b: a c:: l m: ?'' became ''b c: b d:: m n: ?''. For each of the 5 items per alphabet administered to humans (see Table 1), we created 4 item variants, i.e., shifted 1–4 elements

to the left and/or right. We then systematically selected item variants to create 55 unique parallel testlets (required number based on power analysis from pre-registration; Note: we administered each testlet as a 'conversation' containing multiple messages to recreate how we tested humans, see 2.1.3). This allowed us to have robust estimates of LLM performance, while creating some variation in LLM data and enabling us to compute standard errors for statistical analyses.

3 Results

We use mixed ANOVAs to (1) compare performance between our between-subjects participant groups (Adults, Children, and each of the LLMs) on the Latin alphabet and (2) test whether each participant group could generalize analogy solving by performing similarly across alphabets (i.e., our repeated within-subjects factor). All plots show the means and standard errors as error bars.

3.1 RQ1: How Well Do LLMs Solve Letter-String Analogy Problems in the Latin Alphabet Compared to Adults and Children?

We expected LLMs to be able to solve letter-string analogies with the Latin alphabet at the same level as adults (Webb et al., 2023) and that both adults and LLMs would outperform children (Thibaut and French, 2016) (hypotheses H1a-c). Similar to what we expected, adults and some LLMs, except Google's Gemma-2 27B and Anthropic's Claude 3.5, performed better than children in the Latin alphabet domain. Open AI's GPT-4o performed similarly to adults, followed closely by Meta's Llama-3.1 405B. See Figure 2 and Table 2 for more detailed results.

3.2 RQ2: How Well Do Adults, Children, and LLMs Generalize Letter-String Analogy Solving from Latin to Greek (near) and Symbol (far) Alphabets?

As expected, adults and children performed similarly across alphabets (see Figure 2). But, as we suspected, LLM performance indeed degraded in less familiar alphabets (ANOVA results shown in Table 3). More specifically, for each model, performance degraded significantly from the Latin to Greek alphabet (posthoc Bonferonni-corrected t-test results all $p < .001$, except for Llama-3.1 405B $p = 0.012$) and then again from the Greek alphabet to the Symbol list (posthoc Bonferonni-corrected t-test results all $p < .001$).

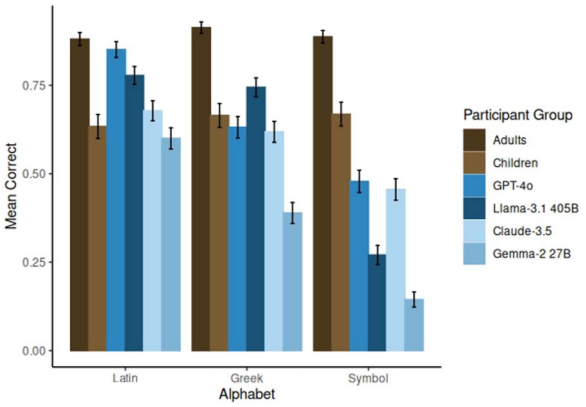


Figure 2: Human vs LLM performance on letter-string analogies across alphabet domains.

3.3 RQ3: Why Can't LLMs Generalize Letter-String Analogy Solving, Like Children?

3.3.1 Performance by Item

To understand why the LLM's had trouble generalizing letter-string analogy solving, we examined item-by-item performance. Table 4 shows an overview. Here we see that the LLMs and humans perform best on item 1, which involves only the first successor transformation, and worst on item 5, that involves both the predecessor transformation and repetition generalization. Item 2 also involves the same repetition rule as item 5, but was solved better by LLMs and children; therefore, it appears that the predecessor rule is what gives both LLMs and children the most trouble. The other item people and LLMs have relatively more trouble with is item 3. This item involves the second successor rule. In sum, the predecessor and second successor rules appear to be the most difficult rules from our item set for people and LLMs to apply.

3.3.2 Next-Previous Letter Task

We designed the Next-Previous Letter Task to check that the LLMs had the requisite knowledge of predecessor and successor to solve letter-string analogies. For this new task we provided an ordered list of letters/symbols and asked the LLMs what the previous and next two letters were given a specific letter. Each rule was tested 5 times, resulting in 20 items total.

Participant Group	n	Latin		Greek		Symbol	
		Mean	SD	Mean	SD	Mean	SD
Adults	62	0.88	0.16	0.91	0.13	0.89	0.23
Children	41	0.62	0.22	0.66	0.23	0.67	0.30
Claude-3.5	54	0.68	0.18	0.62	0.21	0.46	0.24
Gemma-2 27B	54	0.60	0.24	0.39	0.20	0.14	0.15
GPT-4o	54	0.85	0.18	0.63	0.21	0.48	0.18
Llama-3.1 405B	54	0.79	0.16	0.74	0.19	0.27	0.20

Table 2: Descriptive statistics on letter-string analogy performance by Participant Group and Alphabet.

Participant Group	Effect	DFn	DFd	F	p Adjusted
Adults	1.59	96.9	0.95	1.000	
Children	2.00	76.0	0.27	1.000	
Claude-3.5	1.65	87.6	29.5	<.001	
Gemma-2 27B	2.00	106.0	88.2	<.001	
GPT-4o	2.00	100.0	55.0	<.001	
Llama-3.1 405B	1.70	90.1	135.0	<.001	

Table 3: Post hoc ANOVA Results for main Alphabet effect by Participant Group.

	Item				
Participant Group	1	2	3	4	5
Adults	0.97 (0.18)	0.94 (0.25)	0.82 (0.39)	0.94 (0.24)	0.81 (0.40)
Children	0.85 (0.36)	0.75 (0.44)	0.52 (0.50)	0.76 (0.43)	0.38 (0.49)
Claude-3.5	0.90 (0.30)	0.65 (0.48)	0.54 (0.50)	0.62 (0.49)	0.20 (0.40)
Gemma-2 27B	0.62 (0.49)	0.27 (0.45)	0.38 (0.49)	0.37 (0.48)	0.25 (0.43)
GPT-4o	0.92 (0.27)	0.73 (0.45)	0.45 (0.50)	0.78 (0.41)	0.39 (0.49)
Llama-3.1 405B	0.83 (0.37)	0.62 (0.49)	0.53 (0.50)	0.57 (0.50)	0.44 (0.50)

Table 4: Mean proportion correct (SD) by Participant Group for each Item.

X	Correct	Rule
c	d	next_1
c	e	next_2
d	c	prev_1
e	c	prev_2

Table 5: Next-Previous Letter Task: Example items from the Latin alphabet.

Figure 3: LLM performance by rule type across alphabet domains.

We used this optimized prompt: `Here is an ordered list of letters or symbols [Latin alphabet|Greek alphabet|Symbol list]. Which letter or symbol is [one|two] [before|after] [X]?` See Table 5 for rules and example items.

As can be seen in Figure 3, all models do best when asked to identify the next or previous letter and worse when it concerns identifying two before or two after. Furthermore, Claude-3.5 performed well and similarly in all three domains, which is in contrast to its letter-string analogy performance that degrades from baseline to near to far domains. Similarly, GPT-4o performs well on the Next-Previous task in the Latin and Greek domain, but in the analogy task, its performance

A	B	C	D	Rule AB
c	d	h	i	successor_1
c	e	h	j	successor_2
d	c	h	g	predecessor_1
e	c	h	f	predecessor_2
c d	c d d	h i	h i i	repetition_1
c d	c c d d	h i	h h i i	repetition_2

Table 6: Rule Check Task: Example Items From the Latin Alphabet.

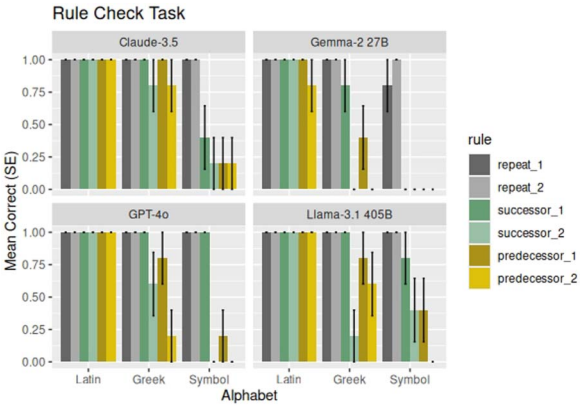


Figure 4: Rule-specific performance across alphabet domains for LLMs.

degrades from Latin to Greek. For Llama-3.1 405B transfer from the Latin to Greek to Symbol domain is similar across tasks, where in both tasks it does well with the Latin and Greek alphabets, but not the Symbol alphabet. Gemma-2 27B's performance is surprisingly more spotty here in the Greek domain than the Symbol domain. In conclusion, these results could explain why the LLMs have trouble with item 3, involving the second successor, but the results do not explain why they have trouble with item 5 involving the predecessor.

3.3.3 Rule Check Task

To better pinpoint why the LLMs had difficulty generalizing to other alphabets, we created a simplified version of the original analogy task that explicitly tested each rule in isolation. LLM prompts were exactly the same as in our original letter-string task, see Section 2.1.3.

The rules were: (1) successor_1, the next letter; (2) successor_2, letter two places after; (3) predecessor_1, the previous letter; (4) predecessor_2, letter two places before; (5) repetition_1, repeating the last letter; and (6) repetition_2, repeating both letters. Each rule was tested five times. See Table 6 for examples.

As Figure 4 shows, the LLMs we tested can solve all rules in the Latin alphabet and have no problem with repetition rules in the Greek and Symbol domains. The successor and predecessor rules were solved to differing degrees in the Greek alphabet, with Claude-3.5 performing best followed by GPT-4o. All models had trouble with the successor_2 and predecessor rules in the Symbol alphabet, where only the successor_1 rule sometimes formed an exception. This makes sense given the predict-the-next-token goal that LLMs are trained on (McCoy et al., 2024).

3.3.4 Error Analysis

In general, when solving letter-string analogies there are often multiple rules that could underlie the change from A to B (Hofstadter and Mitchell, 1994). Because we use very short strings, there are generally only two clearly correct responses. We considered the rules that people would generally prefer when responding, to be ''correct'', such as if **ab** changes to **ac**, then **gh** changes to **gi**. However, the literal rule of replacing the last letter with **c**, with response **gc** could also be considered correct.

Error Categories. To examine errors in more detail we created a set of categories based on those from Lewis and Mitchell (2025) and extended these to account for common errors in children (Stevenson and Hickendorff, 2018). In the **Literal rule** category, the change from A to B is literally copied to C such as **a b : a c c :: g h : g c c** rather than providing the more common response of **g i i**. In the **One rule** category, the response is partially correct, but only (part of) one of the rules in the problem was applied, such as in responses to the previous example, **g h h** (only repetition applied) or **g i** (only successor applied). Partially correct responses are common in children when problem load supersedes processing capacity (Stevenson and Hickendorff, 2018). In the **Incorrect rule** category, one of the other rules from our item set (i.e., successor, predecessor, repetition) was applied; for example, if the successor rule was used instead of the predecessor rule. For the **Copy rule**, the A, B or C term was copied as copying the C-term is common in young children (Stevenson and Hickendorff, 2018; Opiełka et al., 2024). Finally, all remaining erroneous responses were placed in the **Other rule** category. Given that our

Participant Group	Correct	Literal Rule	One Rule	Incorrect Rule	Copy Rule	Other Rule
Adults	0.89	0.00	0.02	0.00	0.00	**0.09**
Children	0.66	0.00	0.06	0.01	0.00	**0.23**
Claude-3.5	0.58	0.05	**0.19**	0.08	0.01	0.09
Gemma-2 27B	0.38	**0.21**	0.12	0.05	0.02	**0.22**
GPT-4o	0.65	**0.13**	0.07	0.02	0.00	**0.12**
Llama-3.1 405B	0.60	0.08	0.10	0.02	0.00	**0.20**

Table 7: Proportions of error categories by participant group. *Note*: 5% of children's responses were empty.

task was less complex than in Lewis and Mitchell (2025) (i.e., shorter strings, fewer rules), we were able to automatically code these categories.

Table 7 shows that adults and children did not use the Literal rule, whereas all models used it sometimes. For Gemma-2 27B and GPT-4o the Literal rule was one of the most common error types. The One rule was used most often in errors by Claude-3.5. The Incorrect and Copy rules were not used very often by people or models. And the Other rules were used most often by all, except Claude-3.5.

String Distance between ''Correct'' and ''Erroneous'' Responses. For each erroneous response we computed the Levenshtein string distance, also known as optimal string alignment distance, from the expected ''correct'' response to the given response. This distance counts the minimum number of edit operations (insertion, deletion, substitution) needed to change one string into the other. Here we investigate whether there are differences in mean Levenshtein distance between adults, children and LLMs for ''erroneous'' responses. Figure 5 shows that that the Levenshtein distance for ''erroneous'' responses is greater for children on all alphabets than for LLMs. For adults, this is only the case for the Symbol alphabet. For LLMs the Levenshtein distance hovers just under the 2 for all alphabets. Note also that the standard errors for LLMs are also much smaller, but this is because the adults and sometimes children (Greek, Symbol alphabets) had far fewer ''erroneous'' responses to sample from. These results tells us that when children provide ''erroneous'' answers their responses tended to differ largely from the expected response. For example, three children responded 'm m' to the item 'If c d changes to b d, what does m m n n change to?', which has a Levenshtein distance of 6 from the expected response 'l l n n'. The LLMs

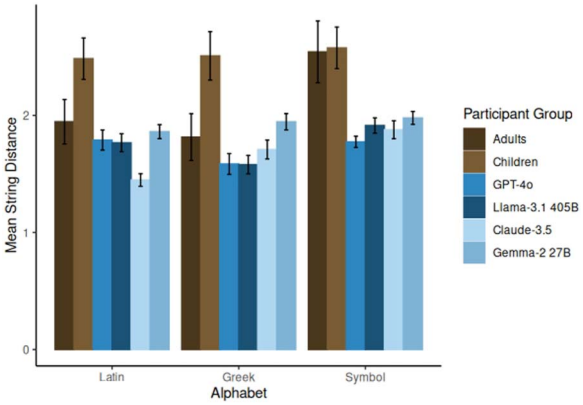


Figure 5: Mean Levenshtein string distance between incorrect and expected responses.

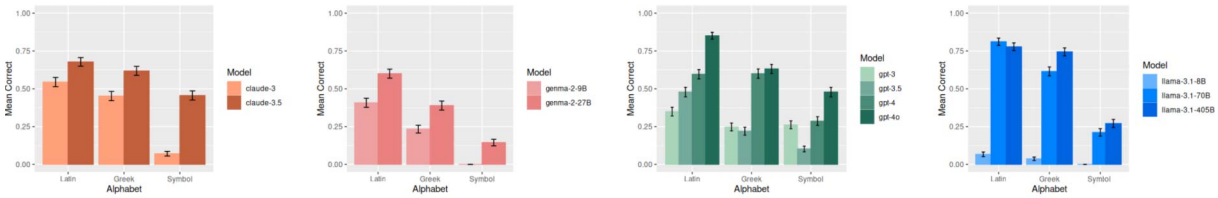
tended to provide 1 or 2 expected letters and 1 or 2 unexpected ones. For example, on the same item (and its variants) six GPT-4o runs provided 'l m m n' as a response, with a Levenshtein distance of 2 from the expected response.

4 RQ4: What is the Effect of Model Size on Letter-String Analogy Performance?

In general, larger LLMs perform better on reasoning tasks than smaller LLMs (Wei et al., 2022; Huang and Chang, 2022). As Figure 6 shows, typical scaling laws generally appear to hold for how well LLMs generalize analogy solving in the letter-string domain. Especially in the Symbol domain, we observe a marked performance increase from smaller to larger models.

5 Discussion

Our main finding is that the LLMs we tested, using the same prompts given to children, were not able to generalize letter-string analogy solving like children can. LLMs perform at or above the level of children on letter-string analogies in the familiar Latin alphabet, but their performance



(a) Claude (b) Gemma (c) GPT (d) Llama

Figure 6: Effect of LLM size on proportion correct in letter-string analogy solving across alphabets Latin, Greek, and Symbol.

on these same problems reduces somewhat when using the Greek alphabet (near transfer) and deteriorates greatly when using our Symbol alphabet (far transfer).

Why can't LLMs generalize when solving letter-string analogies? For some LLMs, this appears to be because they do not meet underlying requisites, such as indicating the predecessor or second successor of a letter in sequence. This would make sense given the predict-the-next-token goal that LLMs are trained on (McCoy et al., 2024). We tested this using the Next-Previous letter task, where models were explicitly prompted with an ordered list of letters or symbols and asked to identify the (second) successor or predecessor to a given letter or symbol. These results explain why the LLMs have trouble with analogies involving a second successor. But, the LLMs had little trouble identifying predecessors in the Next-Previous Letter task, so these results do not fully explain why LLM performance degrades from Latin to Greek and Symbol domains.

The problem with LLM transfer from the Latin to other domains seems to lie in that the conceptual abstraction of what constitutes an alphabet, such as being an ordered sequence, does not flexibly map to less familiar domains like it does in people. Evidence for this comes from the Rule Check task, where we tested LLMs on each rule in isolation. Here repetition rules could easily be applied to novel alphabets. This makes sense because repeating a character in a string can be done without knowing the alphabet. In contrast, LLMs had more trouble with predecessor and second successor rules. Both require an alphabet that is encoded as an ordered list of letters/symbols and an abstraction of what constitutes *previous* and *next*. This result aligns with previous work where GPT models could solve letter-string analogies with familiar alphabets in their standard order, but for shuffled alphabets performance dropped drastically, whereas for people performance remained the same (Hodel and West, 2023; Lewis and Mitchell, 2025). We noted that in the Greek domain the letters were also ordered by unicode value, but in our Symbol domain they were not, which could perhaps explain why Greek items were easier. So, to check whether order was also a factor in our Symbol domain, we adapted the task to make the Symbol alphabet also ordered by their unicode values. However, this adaptation resulted only in some improvement in the Claude and Gemma models, and our findings still held (see Appendix D).

We also investigated which kinds of errors people and LLMs made. This is important because letter-string analogies, like many four-term visual analogies, apply ambiguous rules (e.g., Opiełka et al., 2024), and can be solved correctly in multiple ways (Hofstadter and Mitchell, 1994). The two main ways to solve the items in our task were what we considered the ‘‘correct’’ way (e.g., a b : a c c :: g h : g i i) and the ‘‘literal’’ way (e.g., a b : a c c :: g h : g c c). People did not use the ‘‘literal’’ rule, whereas the models all did to varying degrees (ranging from 5-21%). The other main difference between human and LLM errors was that children's erroneous responses were generally more distant (Levenshtein string distance) from the ‘‘correct’’ response than those of LLMs. This could be because children reverted to non-analogical strategies that we didn't account for in our error coding scheme, given that this is the first time letter-string analogies have been administered to children.

Based on our results, LLMs appear unable to create on-the-fly representations of novel alphabets in the context of the letter-string analogies as well as the next-previous letter task—despite

being given the ordered list of letters/symbols before each item. This inability was clear for both larger and smaller LLMs, although relative performance did scale with model size. A possible explanation lies in work studying the internal representations of LLMs, where abstract concepts like ‘‘antonym’’ show invariant linear representations, but ‘‘previous’’ and ‘‘next’’ do not (Opiełka et al., 2025). It appears that LLMs require an in-weight linear representation of an alphabet to successfully solve letter-string analogies. For novel alphabets, next-token-prediction does help them solve analogies with simple repetition and successor rules, but not with more complex rules and not at the level of children. Indeed, Webb et al. (2024) found that GPT-4 can only perform these abstractions by creating and executing code to map the novel alphabet to new positions and compute previous and next letters. This is of course very different from how children solve such problems.

In contrast, our results show that in children, familiarity with letters or symbols does not influence letter-string analogy solving. As such, our results add to the accumulating evidence that questions whether reasoning actually occurs in these LLMs (Wu et al., 2024; Gendron et al., 2024; Razeghi et al., 2022). Interestingly, in 1980, Schank concluded that there wasn't much intelligence in artificial intelligence given its limited ability to generalize. Similarly, Doumas et al. (2022) argue that robust analogical transfer is a uniquely human ability. Based on our findings so far we concur, and now ask the question: Is generalization to unfamiliar domains indeed what separates human general intelligence from that of artificial general intelligence? The challenge now is to create uncontaminated far generalization tasks that AI models have not been trained on to answer this question.

Acknowledgments

This research was funded by the the Dutch Research Council (NWO) project "Learning to solve analogies: Why do children excel where AI models fail?" with project number 406.22.GO.029 awarded to Claire Stevenson. We thank Veerle Vijverberg and Talea Sibum for their assistance with data collection. We also thank the University of Amsterdam CreAI Lab for fruitful discussions related to this work and the anonymous reviewers and the action editor for their helpful comments.

References

Anthropic. 2024. Introducing the next generation of claude. *Anthropic Preprint*.

Susan M. Barnett and Stephen J. Ceci. 2002. When and where do we apply what we learn?: A taxonomy for far transfer. *Psychological Bulletin*, 128(4):612. https://doi.org/10.1037/0033-2909.128.4.612, PubMed: 12081085

Katarzyna Bobrowicz, Felicia Lindström, Marcus Lindblom Lovén, and Elia Psouni. 2020. Flexibility in problem solving: Analogical transfer of tool use in toddlers is immune to delay. *Frontiers in Psychology*, 11:573730. https://doi.org/10.3389/fpsyg.2020.573730, PubMed: 33123052

Zhe Chen. 1996. Children's analogical problem solving: The effects of superficial, structural, and procedural similarity. *Journal of Experimental Child Psychology*, 62(3):410–431. https://doi.org/10.1006/jecp.1996.0037, PubMed: 8691121

Leonidas A. A. Doumas, Guillermo Puebla, Andrea E. Martin, and John E. Hummel. 2022. A theory of relation learning and cross-domain generalization. *Psychological Review*. https://doi.org/10.1037/rev0000346, PubMed: 35113620

Gemma Team. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Gael Gendron, Qiming Bao, Michael Witbrock, and Gillian Dobbie. 2024. Large language models are not strong abstract reasoners. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence*, IJCAI-24. International Joint Conferences on Artificial Intelligence Organization.

Dedre Gentner. 1988. Metaphor as structure mapping: The relational shift. *Child Development*, 59(1):47–59. https://doi.org/10.1111/j.1467-8624.1988.tb03194.x

Dedre Gentner and Christian Hoyos. 2017. Analogy and abstraction. *Topics in Cognitive Science*, 9(3):672–693. https://doi.org/10.1111/tops.12278, PubMed: 28621480

Dedre Gentner and Cecile Toupin. 1986. Systematicity and surface similarity in the

development of analogy. *Cognitive Science*, 10(3):277–300. https://doi.org/10.1016/S0364-0213(86)80019-2

Mariel K. Goddu, Tania Lombrozo, and Alison Gopnik. 2020. Transformations and transfer: Preschool children understand abstract relations and reason analogically in a causal task. *Child Development*, 91(6):1898–1915. https://doi.org/10.1111/cdev.13412, PubMed: 32880903

Usha Goswami. 1991. Analogical reasoning: What develops? A review of research and theory. *Child Development*, 62(1):1–22. https://doi.org/10.1111/j.1467-8624.1991.tb01511.x

Damian Hodel and Jevin West. 2023. Response to ''Emergent analogical reasoning in large language models''. *arXiv preprint arXiv:2308.16118*. Response to Webb et al. (2023), Nature Human Behaviour.

Douglas R. Hofstadter. 1984. The Copycat project: An experiment in nondeterminism and creative analogies, Massachusetts Institute of Technology.

Douglas R. Hofstadter and Melanie Mitchell. 1994. The copycat project: A model of mental fluidity and analogy-making. In *Advances in Connectionist and Neural Computation Theory*, volume 2, pages 31–112. Ablex, Norwood, NJ.

Keith J. Holyoak. 2012. Analogy and relational reasoning. In Keith J. Holyoak and Robert G. Morrison, editors, *The Oxford Handbook of Thinking and Reasoning*, pages 234–259. https://doi.org/10.1093/oxfordhb/9780199734689.013.0013

Keith J. Holyoak, Ellen N. Junn, and Dorrit O. Billman. 1984. Development of analogical problem-solving skill. *Child Development* 2042–2055. https://doi.org/10.2307/1129778, PubMed: 6525888

Jie Huang and Kevin Chen-Chuan Chang. 2022. Towards reasoning in large language models: A survey. *CoRR*, abs/2212.10403.

Nicholas Ichien, Hongjing Lu, and Keith J. Holyoak. 2020. Verbal analogy problem sets: An inventory of testing materials. *Behavior Research Methods*, 52(5):1803–1816. https://doi.org/10.3758/s13428-019-01312-3, PubMed: 31898298

Tamar Johnson, Mathilde ter Veen, Rochelle Choenni, Han van der Maas, Ekaterina Shutova, and Claire E. Stevenson. 2025. Do large language models solve verbal analogies like children do? In *Proceedings of the 29th Conference on Computational Natural Language Learning*, pages 627–639. https://doi.org/10.18653/v1/2025.conll-1.40

Laura L. Jones, Matt J. Kmiecik, John L. Irwin, and Robert G. Morrison. 2022. Differential effects of semantic distance, distractor salience, and relations in verbal analogy. *Psychonomic Bulletin & Review*. https://doi.org/10.3758/s13423-022-02062-8, PubMed: 35132581

Martha Lewis and Melanie Mitchell. 2025. Evaluating the robustness of analogical reasoning in large language models. *Transactions on Machine Learning Research*. https://doi.org/10.1007/978-3-031-76646-6_4

R. Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L. Griffiths. 2024. Embers of autoregression show how large language models are shaped by the problem they are trained to solve. *Proceedings of the National Academy of Sciences*, 121(41):e2322420121. Edited by Richard Shiffrin. https://doi.org/10.1073/pnas.2322420121, PubMed: 39365822

Melanie Mitchell. 2021. Abstraction and analogy-making in artificial intelligence. *Annals of the New York Academy of Sciences*, 1505(1):79–101. https://doi.org/10.1111/nyas.14619, PubMed: 34173249

Arsenii Kirillovich Moskvichev, Victor Vikram Odouard, and Melanie Mitchell. 2023. The conceptARC benchmark: Evaluating understanding and generalization in the ARC domain. *Transactions on Machine Learning Research*.

Timothy M. Mulholland, James W. Pellegrino, and Robert Glaser. 1980. Components of geometric analogy solution. *Cognitive Psychology*, 12(2):252–284. https://doi.org/10.1016/0010-0285(80)90011-0, PubMed: 7371379

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Gustaw Opiełka, Hannes Rosenbusch, and Claire E. Stevenson. 2025. Analogical reasoning inside large language models: Concept vectors and the limits of abstraction. *arXiv preprint arXiv:2503.03666*.

Gustaw Opiełka, Hannes Rosenbusch, Veerle Vijverberg, and Claire E. Stevenson. 2024. Do large language models solve ARC visual analogies like people do? In *Proceedings of the 46th Annual Conference of the Cognitive Science Society*.

Yasaman Razeghi, Robert L. Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 840–854. https://doi.org/10.18653/v1/2022.findings-emnlp.59

Lindsey E. Richland, Robert G. Morrison, and Keith J. Holyoak. 2006. Children's development of analogical reasoning: Insights from scene analogy problems. *Journal of Experimental Child Psychology*, 94(3):249–273. https://doi.org/10.1016/j.jecp.2006.02.002, PubMed: 16620867

Roger C. Schank. 1980. How much intelligence is there in artificial intelligence? *Intelligence*, 4:1–14. https://doi.org/10.1016/0160-2896(80)90002-1

Richard Shiffrin and Melanie Mitchell. 2023. Probing the psychology of AI models. *Proceedings of the National Academy of Sciences*, 120(10):e2300963120. https://doi.org/10.1073/pnas.2300963120, PubMed: 36857344

Claire E. Stevenson and Marian Hickendorff. 2018. Learning to solve figural matrix analogies: The paths children take. *Learning and Individual Differences*, 66:16–28. https://doi.org/10.1016/j.lindif.2018.04.010

Jean-Pierre Thibaut and Robert M. French. 2016. Analogical reasoning, control and executive functions: A developmental investigation with eye-tracking. *Cognitive Development*, 38:10–26. https://doi.org/10.1016/j.cogdev.2015.12.002

Jean-Pierre Thibaut, Yannick Glady, and Robert M. French. 2022. Understanding the what and when of analogical reasoning across analogy formats: An eye-tracking and machine learning approach. *Cognitive Science*, 46(11):e13208. https://doi.org/10.1111/cogs.13208, PubMed: 36399055

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Taylor Webb, Keith J. Holyoak, and Hongjing Lu. 2023. Emergent analogical reasoning in large language models. *Nature Human Behaviour*, 7(9):1526–1541. https://doi.org/10.1038/s41562-023-01659-w, PubMed: 37524930

Taylor Webb, Keith J. Holyoak, and Hongjing Lu. 2024. Evidence from counterfactual tasks supports emergent analogical reasoning in large language models. *arXiv preprint arXiv:2404.13070*. https://doi.org/10.1093/pnasnexus/pgaf135, PubMed: 40432905

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani

Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2024. Reasoning or reciting? Exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, volume 1 of *NAACL-HLT '24*, pages 1819–1862, Mexico City, Mexico. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.naacl-long.102

Eunice Yiu, Maan Qraitem, Charlie Wong, Anisa Noor Majhi, Yutong Bai, Shiry Ginosar, Alison Gopnik, and Kate Saenko. 2024. Kiva: Kid-inspired visual analogies for testing large multimodal models. *arXiv preprint arXiv:2407.17773*.

A Open Science Practices

Ethics This study was approved by the ethics board at the University of Amsterdam, Social and Behavioural Sciences on May 24, 2023 (ID: FMG-2495).

Preregistration The hypotheses, methods, and analyses were preregistered on the Open Science Framework (OSF) prior to ethical approval. These were updated on July 26, 2023, to accommodate new methods for LLM data collection. The main deviation from preregistration was using different LLMs than previously specified.

Data Availability Preregistration, materials, data, and code are publicly available on the project's OSF repository: `https://osf.io/jdty3/`. A direct link to all data and code can be found here: `https://github.com/cstevenson-uva/llm_letterstring_generalization/`.

B LLM Prompt Engineering Results

We administered each letter-string analogy item to LLMs using 5 different prompt templates, as prompt engineering can change the LLMs' performance on the task. The templates were as follows.

1. If a b c changes to a b d, what does i j k change to?
2. a b c is to a b d, as i j k is to ?
3. a b c $\longrightarrow$ a b d \n e f g $\longrightarrow$
4. Let's try to complete the pattern:\n [a b c] [a b d] \n [i j k] [
5. [a b c] [a b d] \n [i j k] [

As can be seen in Figure or Table 8, template 1, derived from Mitchell (2021) worked best overall. Template 4, the best template found by Webb et al. (2023) worked well in Latin and Greek alphabets, but not as well for the Symbol list, which makes sense because [and] are symbols themselves. Our results are based on template 1.

Model	Template 1	Template 2	Template 3	Template 4	Template 5
Claude-3.5	0.82 (0.10)	**0.88 (0.08)**	0.71 (0.11)	0.53 (0.13)	0.71 (0.11)
Gemma-2 27B	**0.59 (0.12)**	**0.59 (0.12)**	0.41 (0.12)	0.41 (0.12)	0.29 (0.11)
GPT-4o	**0.82 (0.10)**	0.71 (0.11)	0.71 (0.11)	0.71 (0.11)	0.71 (0.11)
Llama-3.1 405B	**0.71 (0.11)**	0.59 (0.12)	0.59 (0.12)	0.59 (0.12)	0.35 (0.12)
Total	**0.74 (0.05)**	0.69 (0.06)	0.60 (0.06)	0.56 (0.06)	0.52 (0.06)

Table 8: Prompt template performance mean correct (SE) for selected models.

C LLM Results Without Previous Messages

We readministered the items from the template comparison (see Appendix B) to examine whether it was better to administer the items one-by-one or to include all previous message history, i.e., the previous items and their responses.

As can be seen in Table 9, it was generally advantageous to include previous message history versus not. Of the LLMs we tested, there may be two possible exceptions to look out for in future work. Both Gemma-2 27B and Llama-3.1 405B had significantly higher accuracy (i.e., no overlapping SE margins) without message history on the Symbol alphabet. In both cases, the main result of lower performance on Greek and Symbol alphabets versus Latin alphabet still holds.

D Ordered Symbol Task

We re-administered the items from the template comparison (see Appendix B) to examine whether ordering the symbols by unicode value would improve the models' performance on the Symbol alphabet.

Model	Latin		Greek		Symbol	
	Incl. History	No History	Incl. History	No History	Incl. History	No History
Claude-3.5	**0.88 (0.07)**	0.65 (0.10)	**0.80 (0.08)**	0.75 (0.10)	**0.40 (0.10)**	0.20 (0.09)
Gemma-2 27B	0.64 (0.10)	**0.73 (0.08)**	**0.48 (0.10)**	0.43 (0.09)	0.04 (0.04)	**0.20 (0.07)**
GPT-4o	**0.84 (0.07)**	0.73 (0.08)	**0.64 (0.10)**	0.43 (0.09)	**0.60 (0.10)**	0.57 (0.09)
Llama-3.1 405B	**0.76 (0.09)**	0.67 (0.09)	0.56 (0.10)	**0.67 (0.09)**	0.20 (0.08)	**0.43 (0.09)**

Table 9: Mean Correct (SE) for LLMs with versus without Message History.

Model	Latin	Greek	Symbol (unordered)	Symbol (ordered)
Claude-3.5	0.84 (0.07)	0.72 (0.09)	0.40 (0.10)	**0.72 (0.09)**
Gemma-2 27B	0.64 (0.10)	0.48 (0.10)	0.04 (0.04)	**0.36 (0.10)**
GPT-4o	0.76 (0.09)	0.60 (0.10)	**0.60 (0.10)**	**0.60 (0.10)**
Llama-3.1 405B	0.72 (0.09)	0.68 (0.10)	**0.20 (0.08)**	**0.28 (0.09)**
Total	0.74 (0.44)	0.62 (0.49)	0.31 (0.47)	**0.49 (0.50)**

Table 10: Mean (SE) correct for LLMs with versus without Symbols were ordered by unicode value.

This did result in improved performance in Claude 3.5 and Gemma 2, where Claude 3.5 improved to the same performance level of the Greek alphabet. For GPT-4o and Llama 3.1 there were no significant improvements from the reordering. In all cases our main finding—that performance degraded from Latin to the Greek and Symbol alphabets—still held. However, in future experiments using a Symbol domain, it is important to realize that LLMs generally benefit by ordering symbols by unicode value.