Aligning transformer circuit mechanisms to neural representations in relational reasoning

Luke J. Hearne[1,2], Conor Robinson[1,2], Luca Cocchi[1,2], Takuya Ito[3]

[1] QIMR Berghofer Medical Research Institute, Brisbane, QLD, Australia
[2] University of Queensland, School of Biomedical Sciences, Faculty of Medicine.
[3] T.J. Watson Research Center, IBM Research, Yorktown Heights, NY, USA

Abstract

Relational reasoning—the capacity to understand how elements relate to one another—is a defining feature of human intelligence, yet its computational basis remains unclear. Here, we combined human neuroimaging (7T fMRI) and artificial neural network modeling to examine relational reasoning in biological and artificial systems. Using the Latin Square Task, we found that humans and transformers were able to generalize the task reliably, while standard architectures used in cognitive neuroscience could not. Analysing the transformer components revealed distinct computational roles: positional encoding captured the spatial structure of the task and aligned with representations in visual cortex, whereas attention encoded relational structure and mapped onto frontoparietal and default-mode networks. Attention weights tracked the relational complexity of the task, providing a computational analogue of working-memory demands. These results demonstrate convergent computational strategies for reasoning in brains and transformers, highlighting attention-based architectures as powerful models for investigating the neural basis of higher cognition.

Introduction

Relational reasoning is the ability to understand how elements interact to come to a conclusion, decision, or achieve a goal[1,2]. This capacity underpins a plethora of complex thinking skills that contribute to general intelligence[3–5], including understanding analogy, inferring hidden causes, and generalising patterns to novel contexts. In contrast to other animal species, humans are uniquely gifted at relational reasoning due to our increased capacity in processing multiple relations in parallel[1,6]. Although neuroimaging has mapped the neural correlates of relational reasoning[7–11], progress toward understanding the underlying computations has been limited, partly due to the challenge of modeling complex intellectual processes[12].

Modern artificial neural networks have demonstrated impressive performance on a range of complex tasks [13], including forms of reasoning [14–16]. When paired with well-designed cognitive tasks and robust evaluations, these models can help reveal the computational mechanisms underlying specific cognitive processes[17]. When further combined with neural data, these models can adjudicate the similarities and differences between biological and computational implementations of these processes[18,19]. Using this approach, several studies have elucidated the computational mechanisms related to sensory processing[20], memory and navigation[21,22], and context-dependent decision making[17,23–25], among others. While significant progress has been made in bridging computational modeling with cognitive neuroscience, modeling higher-order cognitive processes, such as relational reasoning, remains challenging. This is largely because 1) empirical study of relational reasoning is primarily limited to humans, 2) designing reliable experiments to probe relational reasoning is often challenging in practice, and 3) commonly used neural architectures in neuroscience, like recurrent neural networks and convolutional neural networks, struggle with such complex reasoning tasks.

Recent advances in artificial intelligence have been driven by the computational power of the transformer architecture[13]. In brief, transformers combine three key neural network components: positional encoding (PE), the attention mechanism, and a standard multilayer perceptron (MLP). Similar to how retinal cells receive 2D information from the visual field while maintaining the correct spatial organization of a stimulus, PE preserves the order and arrangement of input tokens (e.g., a spatial receptive field), providing a useful inductive bias for organizing inputs[26]. On the other hand, *attention* allows each element of an input (e.g., each word in a sentence) to query all other elements and decide which ones are relevant[27]. This allows the model to understand relationships between elements in the context of the entire input. Finally, the MLP portion of the transformer – a standard neural architecture – appears to behave as a key-value memory store[28]. While not strictly brain inspired, computational parallels have been drawn between transformers and cortico-thalamic circuits[29], neural visual information routing[30], and models of the hippocampus[31]. For example, transformers can reproduce empirical spatial representations from hippocampal place and grid cells[31]. Nevertheless, exactly how transformers, or any computational model, might implement relational reasoning computations remains unclear, especially given their limited application in cognitive science using well-controlled psychological experiments.

Motivated by this gap, we investigated the computational mechanisms of relational reasoning across various artificial neural networks and compared them to human performance and brain imaging data on an established relational reasoning task: the Latin Square Task (LST)[32]. The LST is a non-verbal spatial relational reasoning task that leverages relational complexity theory[1] to systematically manipulate reasoning difficulty. We found that, while humans reliably solved LST puzzles, both standard MLP and recurrent neural network architectures failed to generalize beyond the training set. In contrast, transformer models were capable of robust generalization. Despite not being biologically inspired, the transformer's PE and attention mechanisms captured computational strategies that paralleled human neurocognitive processes. Specifically, PE captured the spatial structure of the task and aligned with representations in visual cortex, whereas attention tracked the relational complexity of the task and mapped onto frontoparietal and default-mode networks. Moreover, fine-grained analysis of the attention weights suggested that greater attentional 'spread' (i.e., the density of the weights) was associated with more complex reasoning problems, providing a computational analog of the working memory capacity demands of the task. Finally, we illustrate how different layers of transformer representations produce heterogeneous representations that organize from lower-level to higher-order cortical representations. Together, these findings demonstrate that despite limited architectural similarities across brains and transformers, we find that analysis of transformers – and the corresponding representations that its unique computational components produce, such as PE and attention – lead to interpretable representations that reveal converging computational strategies underlying relational reasoning in humans and machines.

Results

The Latin Square Task in humans and neural networks

We start by contrasting performance between humans and neural networks on the Latin Square Task (LST)[32]. The LST has been used in cognitive science to probe relational reasoning in humans, and has been previously shown to correlate with fluid intelligence[33,34]. The task involves inferring missing elements in a grid based on relational constraints, analogous to the game sudoku, which has become a popular and challenging benchmark for AI systems[35]. Specifically, participants are shown a 4x4 grid populated with shapes, blank spaces, and a target question mark. Participants are asked to determine the shape that fits in the target cell, ensuring that each shape only appears once in each row and column. Puzzles can be organised into three increasingly complex reasoning conditions[1] that are dictated by the number of vectors that need to be taken into consideration to solve each given puzzle (an example of each is shown in **Fig. 1A**). Thus, the LST offers a tractable, yet meaningful benchmark for probing the computational mechanisms underlying relational reasoning in both humans and machines, while also being well-suited for measuring human brain activity during task performance.

As expected, human performance on the LST decreased as relational complexity increased ($F(2,78)=55.29$, $p < .001$, $\eta^2_G=.43$, **Fig. 1B**). Follow up *t*-tests confirmed the direction of this effect, 1-vector puzzles ($M=96.67\%$, $SD=4.13\%$) were solved more often than 2-vector puzzles

(*M*=87.01%, *SD*=11.01, *t*(39)=6.78, *p*< .001, *d=1.15)* and 2-vector puzzles were solved more often than 3-vector puzzles (*M*=66.25%, *SD*=22.46, *t*(39)=5.94, *p*< .001, *d=1.16)*. Accuracy in the most difficult condition (3-vector) correlated with performance on a fluid intelligence test (The Raven's Advanced Progressive Matrices, rs=.48, p=.002).

We compared three neural network models; an encoder-only Transformer, a Long Short-Term Memory (LSTM) network, and a multilayer perceptron (MLP, see Methods for details). All three models achieved perfect accuracy in the task in the training dataset (8000 puzzles, **Fig. 1C**). However, in 1000 unseen test puzzles drawn from the same distribution as the training data, model performance dropped substantially for the non-Transformer ANNs (mean performance; Transformer=98.51%, LSTM=60.89%, MLP=86.91%). Finally, the ANNs were tested on the same puzzles that humans completed. These puzzles were designed with low correlation with the training dataset (jaccard index > 0.8), thus representing a difficult out-of-distribution generalization task. While all models performed above chance (chance=0.25%, mean performance; Transformer=97.65%, LSTM=55.49%, MLP=33.02%), only the Transformer could successfully generalize and solve the task comparable to (better than) humans. Main effects of relational complexity were also observed in the transformer (see **Supplementary Material**).

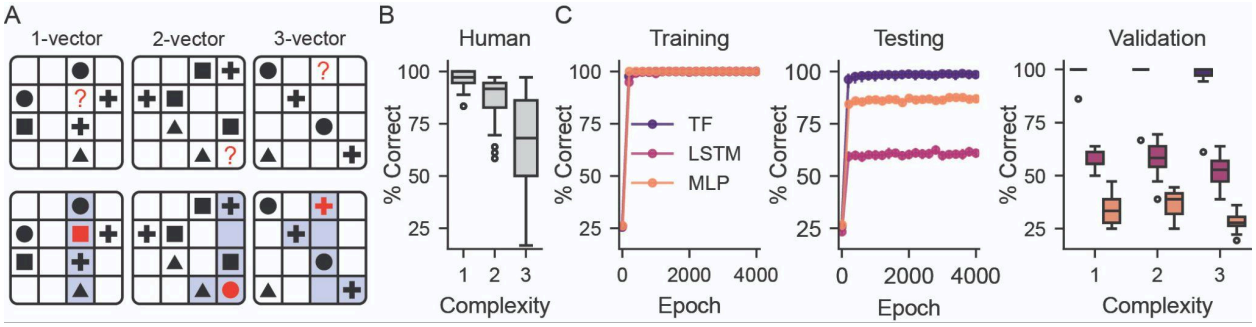

Fig. 1. *Experiment setup and model performance.* **A.** Example puzzles from the Latin Square Task (LST) for 1-, 2- and 3-vector reasoning complexity conditions. Blue highlights in the bottom panel indicate the cells and vectors which contain the required information to solve each problem **B.** Human ($N_{participants}$=40) performance on the task (left). **C.** Average ANN (N_{seeds}=15) task accuracy in the training (left), testing (middle) datasets and validation (right; identical problems to brain imaging data). TF; Transformer, LSTM; Long Short-Term Memory, MLP; Multilayer perceptron.

Mapping architectural components of transformers to neurocognitive mechanisms

Motivated by the question of why transformers could successfully generalize the LST task where other architectures struggled, we next investigated how transformer architectural features contributed to robust relational reasoning. In particular, we evaluated the two architectural components that are unique to the transformer model; the positional encoding (PE) and self-attention mechanism (**Fig. 2A**). PE is a transformer mechanism that provides spatial information about the location of tokens. Since transformers process tokens in parallel rather than sequentially, in the absence of PE, the transformer would not know the correct location of

each symbol in the LST. Since the LST is organized as a 2D grid, we utilized a PE that preserved the pairwise distance between tokens in 2D. Understanding the 2D structure – and therefore the relative positions of columns and rows – is critical to solving each puzzle successfully.

Once the geometry of the tokens (i.e., 2D grid) is accurately processed, the token representation (along with the tag of its location) is sent to the downstream self-attention. There, the self-attention mechanism functions like a dynamic spotlight of visual attention; it evaluates every token's relevance to every other token, assigning "importance" weights for a given pair of tokens. For example, in natural language processing PE aims to embed the order of each word in a sentence, whereas attention aims to identify the relevant relationship *between* words to derive meaning. More concretely, this "spotlight of visual attention" is governed by the dot product of two vectors, for every pair of tokens: the queries (*q*) and keys (*k*) vector (**Fig. 2A**). Functionally, the *Q* vector sends a "query" asking for a specific type of information, and the *K* vector provides information to the query. The dot product then provides an indication of how well the *Q* and *K* vectors matched. In the example provided in **Fig. 2A**, we can see that all tokens received information regarding the probe token "?", though information routing was disproportionately sent to tokens in the row and column of the probe (more information on the Q and K vectors can be found in the **Methods**).

Next, we examined the influence of the transformer's architectural mechanisms on model representations. This was possible by examining the model embeddings after each architectural component (e.g., PE or attention). To estimate the representational content of model embeddings at various stages of processing, we used representational similarity analysis (RSA) [36,37]. RSA enabled us to examine the similarity structure of representations derived from the transformer model and experimenter models of task features (e.g., stimulus, task complexity, and response features; **Fig. 2B**; see **Methods**). To start, we contrasted task representations derived from PE and attention embeddings to each of the experimenter defined task features; we refer to the resulting similarity as representational alignment[38]. We observed a main effect of architectural feature in all three experimenter models; PE demonstrated higher similarity to input stimulus (M_{PE}=0.45, M_{ATTN}=0.31 , $F(1, 14)=575.27$, $p<.001$, $\eta_G^2=.37$), whereas attention activations showed higher similarity to reasoning complexity (M_{PE}=0.02, M_{ATTN}=0.07 , $F(1, 14)=131.59$, $p<.001$, $\eta_G^2=.20$) and motor output models (M_{PE}=0.05, M_{ATTN}=0.22 , $F(1, 14)=144.92$, $p<.001$, $\eta_G^2=.38$) (**Fig. 2C**). Main effects of layers, and interactions between layer and architectural features were also observed (see **Supplementary Table 1**). This analysis confirms the role of PE in organising the visual geometry of the puzzle, and the attention mechanism in relational processing and response generation.

We next sought to assess whether similar patterns of representation emerged in human brain activity and each transformer architectural component **(Fig. 2D)**. We observed significant alignment of representations between specific regions of the primary visual cortex and PE representations (**Fig. 2E,** $p_{FDR} < .05$**, Supplementary Table 2**). For embeddings processed after

the attention mechanism, significant alignment was additionally observed within multimodal brain regions belonging to frontoparietal and default-mode networks ($p_{FDR} < .05$, **Supplementary Table 2**). These results provide evidence that cognitively meaningful representations – PEs with stimulus-related features, and attention with relational complexity-related features – converge across transformers and human brains, despite substantial architectural differences.

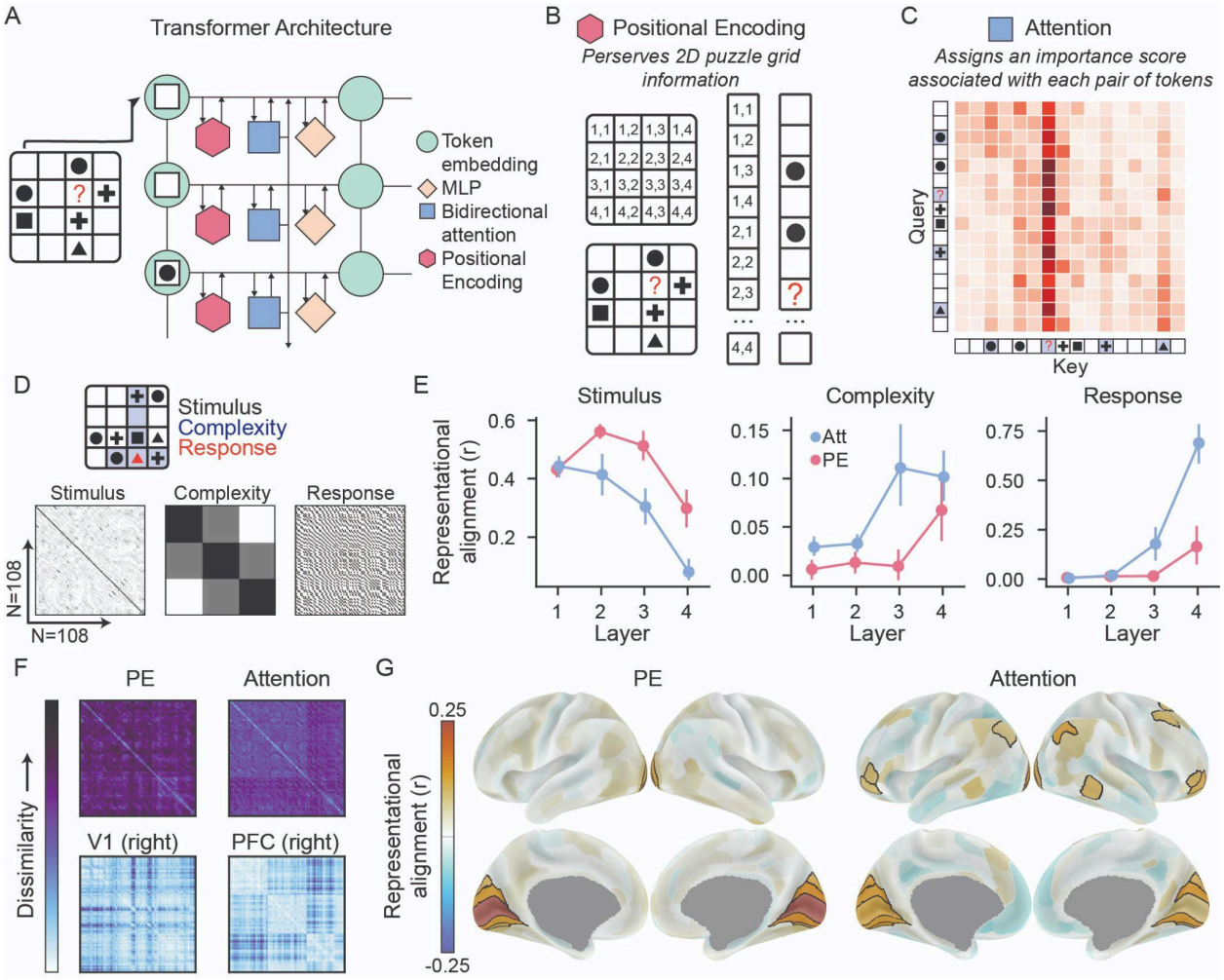

Fig. 2. A. Schematic of the transformer architecture for a single layer. **B.** Schematic of the positional encoding (PE) mechanism. PE information describes the two-dimensional organisation of the puzzle, and therefore the critical rule that each shape can only appear once in every row or column. **C.** Attention scores for the example 1-vector puzzle. Attention scores quantify how strongly each token (puzzle element) relates to all other tokens, guiding the model's focus during information processing. **D.** Experimenter defined models of stimulus, task complexity and response. **E.** Representational alignment between task feature models and transformer (PE and Attention) embeddings. **F.** Example representational dissimilarity matrices derived from the transformer (top row) and fMRI data (bottom row). **G.** Representational alignment between brain and transformer (PE and Attention) embeddings.

Importance of positional encoding on learning meaningful stimulus features

Architecturally, the 2D positional encoding introduced a useful inductive bias, enabling the transformer to capture token order – an essential feature for LST performance. In comparison to the MLP and LSTM, this likely provided the transformer with an advantage in learning and generalizing the LST task, as MLPs and LSTMs were not provided with the input 2D shape in their tokens. (However, LSTMs and MLPs could learn location information from implicit token ordering, since those models are not permutation invariant, unlike the transformer; see **Methods**). To make a fairer comparison between transformers and standard models, we asked whether PE could be learned (or inferred) from data alone, and how learned PE representations might influence performance. Recent work has demonstrated that varying ANN weight initializations can lead to significant differences in the structure of internal representations [39,40]. In particular, weight initializations sampled from a large distribution (e.g., $N(0, \sigma)$ for large σ) tended to produce random, high-dimensional representations (lazy learning), while weight initializations sampled from a small distribution (small σ) tended to learn structured, interpretable representations (rich learning). Within the context of PE for transformers, rather than initialization weights, we initialized learnable parameters with rich (small σ) and lazy (large σ) initializations to 1) assess generalization performance on the out-of-distribution validation set, as well as 2) the structure of the learned representations (**Fig. 3A**)[24].

We found that transformer task accuracy in the out-of-distribution validation set was related to initialization parameters, such that low σ (i.e., rich) models performed significantly better than high σ (lazy) models ($F(19, 266) = 65.93$, $p < .001$, $\eta_G^2 = .73$, **Fig. 3B**). Similar to the human data, there was also a main effect of reasoning complexity ($F(2, 28) = 65.71$, $p < .001$, $\eta_G^2 = .07$), and an interaction between weight initialization and reasoning complexity ($F(38, 532) = 5.85$, $p < .001$, $\eta_G^2 = .06$). The weak complexity-initialization interaction suggests that "lazy" models had particularly poor performance in the most complex reasoning condition.

Next, to better understand why richly learned PEs exhibited better generalization performance, we sought to characterize the learned PE representation. In the previous section, we demonstrated that PE representations that were derived from 2D encodings (i.e., those that preserved the original 2D grid without learning) exhibited high representational alignment with human visual cortex representations (**Fig. 2G**). Thus, we hypothesized that models with *richly learned* PEs (i.e., the models that exhibited improved generalization performance) would learn structured PE representations that had greater alignment with human visual cortex representations. We therefore evaluated whether alignment across fMRI data and transformers was related to the initialization parameter.

Using brain regions of interest defined in the prior analysis (outlined regions in **Fig. 2G** related to PE; **Fig. 3C**) we observed that low σ PE initialization demonstrated higher brain-transformer alignment, specifically in visual regions ($rs = -.82$, $p < .001$, **Fig. 3D**). These results underscore the value of learning structured visual representations (through the PE) for robust relational

reasoning, and show again that transformers – despite their architectural differences – can learn meaningful cognitive and brain-like representations.

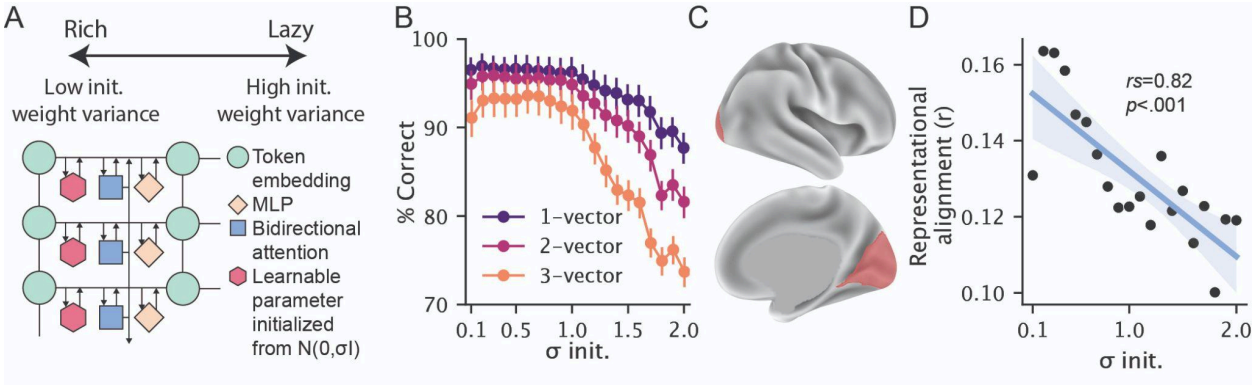

Fig. 3. *Influence of rich and lazy positional encoding weight initialization on generalization performance and brain-transformer representational alignment.* **B**. Task generalization performance when positional encoding (PE) weight initializations (init.) are systematically altered from rich (small σ) to lazy (large σ). **C.** Regions of interest derived from the prior fMRI PE RSA analysis. **D.** Representational alignment between brain and PE within regions of interest across PE initializations.

The computational relevance of attention weights in relational complexity

Our earlier analysis illustrated that the transformer's attention mechanism produced embeddings sensitive to relational complexity using RSA (**Fig. 2E**). This aligns with the architecture, as the attention weights must evaluate the importance of every pairwise interaction between tokens (i.e., elements in the LST grid). Similarly, relational complexity measures the number of vectors and elements needed to accurately perform the task (e.g., 1-vec, 2-vec, and 3-vec problems), therefore taxing working memory demands[1]. Thus, we sought to characterize how the attention weights (in our case 16 tokens x 16 LST tokens) would correlate with task performance and relational complexity. There are several testable hypotheses that lend themselves to interpreting the self-attention mechanism in the LST. We start by focusing on the attention strength of the target ("?") element (equivalent to the red symbols in **Fig. 4A**). We quantified this by deriving the target's key mean score (averaging across queries, see **Methods**). In brief, high query values suggest other puzzle elements are "focusing" on the target ("?"), whereas low values suggest the information flow from other elements to the target is not important for solving the puzzle.

We found attention weights were stronger in cells that represent the relationship between the target and other puzzle elements ($t(14)=10.13$, $p<.001$, $d=3.72$, **Fig. 4B**). Next, we tested whether the correct strategy needed to solve one-vector problems could be related to changes in attention scores. Specifically, one-vector problems can *always* be correctly solved by integrating information in the appropriate column *or* row. This means that the transformer's attention mechanism only needs to attend to elements in either a row or column. For example, the 1-vector problem in **Fig. 4A** requires only integration of elements in the third row (cross, circle, triangle, therefore target=square). Therefore, the critical elements for that problem are elements in that row. We found that 'critical' cells have higher attention scores than non-critical

cells ($t(14)=5.81$, $p<.001$, $d=0.86$, **Fig. 4C**). Finally, we tested how attention scores interact with reasoning complexity (**Fig. 4D**). Since increasing reasoning complexity requires increased attention to more LST grid elements (or for humans, greater working memory demands), we hypothesized that increasing LST complexity would require more distributed attention patterns. In transformers this corresponds to a reduction in the average attention weight. This is because transformers apply a softmax to the attention weights, converting them into a probability distribution that sums to 1. Thus, greater distributed attention weights would push the average attention probabilities down, resulting in overall weaker attention scores (see **Methods**). Indeed, consistent with this intuition, we found that attention weights relevant to the target decreased as reasoning complexity increased ($\chi^2(2)=16.13$, Kendall's $W=0.54$, $p < .001$), which was largely driven by increased mean attention scores in 1-vector problems (1-vector vs. 2-vector; $t(14)=4.25$, $d=0.05$, $p<.001$, 1-vector vs. 3-vector; $t(14)=3.11$, $d=0.14$, $p=.008$, 2-vector vs. 3-vector; $t(14)=1.99$, $d=0.10$, $p=.066$). Thus, this analysis provided a computational understanding of how transformers leverage attention weights to solve relational reasoning problems of varying complexity. These findings provide a computational analog of working memory demands (measured in LST relational complexity[1]) in the transformer's attention mechanism.

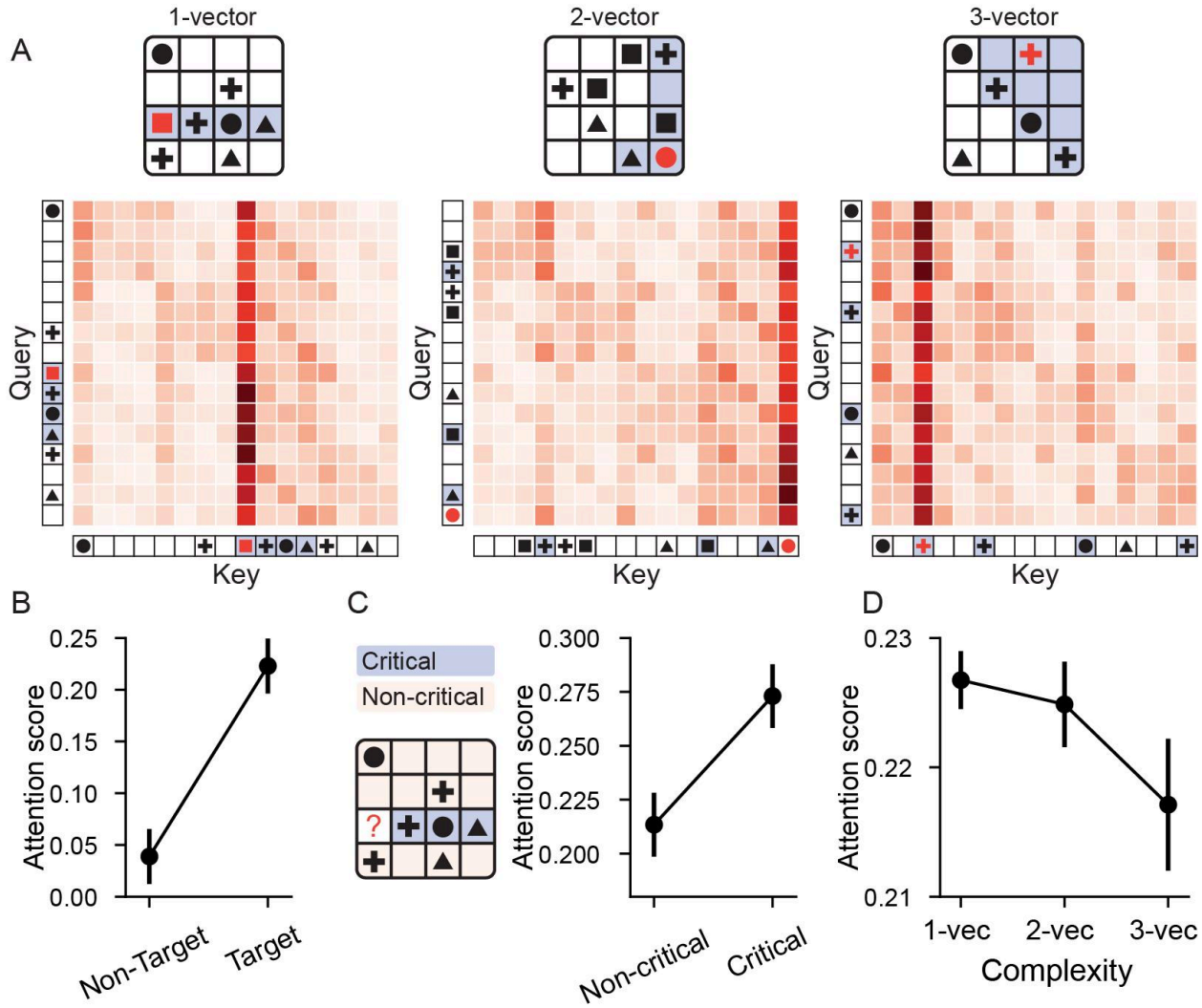

Fig. 4. Computational mechanisms of reasoning complexity in the transformer's attention mechanism. A. Example puzzles from each relational complexity condition and associated attention weight matrices. **B.** Comparison of Target attention scores to all other non-target puzzle elements. The difference in the target versus other elements can be seen visually in the three attention score matrix examples. **C.** Comparison of attention scores between critical and non-critical cells in one-vector problems **D.** Target attention scores as a function of relational complexity. Error bars for panels **B-D** indicate within-seed variability.

Brain-transformer representational alignment across layers

Many prior papers in neuroscience have focused on drawing parallels between hierarchical information processing observed in ANNs and the brain [20]. Thus, as a final exploratory analysis, we examined the relationship between architectural features in the transformer model and brain imaging data across model layers. We examined layer-dependent effects in both PE and attention outputs (**Fig. 5**). Interestingly, PE demonstrated very little change in representational alignment across layers, with significant region effects constrained to the primary visual cortex ($p_{FDR} < 0.05$). In contrast, Attention-fMRI representational alignment was dynamic; appearing in

visual cortex in early layers before progressing to transmodal cortex in later layers ($p_{FDR} < 0.05$, see **Supplementary Table 3**). (Note, in prior analyses we sought to isolate the architectural contributions of PE and attention separately by averaging across layers, whereas here we investigate the effect of PE and attention across layers.) This pattern of results suggests that while PE imposes a strong inductive bias, anchoring the model's representations to preserve stimulus-specific features, attention representations evolve progressively across layers. Notably, in line with previous findings that highlight the involvement of the frontoparietal network (FPN) in relational reasoning (Hearne et al., 2017), our analysis reveals a similar trend: as information propagates through transformer layers, the representations increasingly resemble those found in higher-level FPN regions and diverge from early visual, stimulus-driven patterns. This progression reflects the transformer's ability to organize information hierarchically, transitioning from low-level perceptual features to abstract, relational representations.

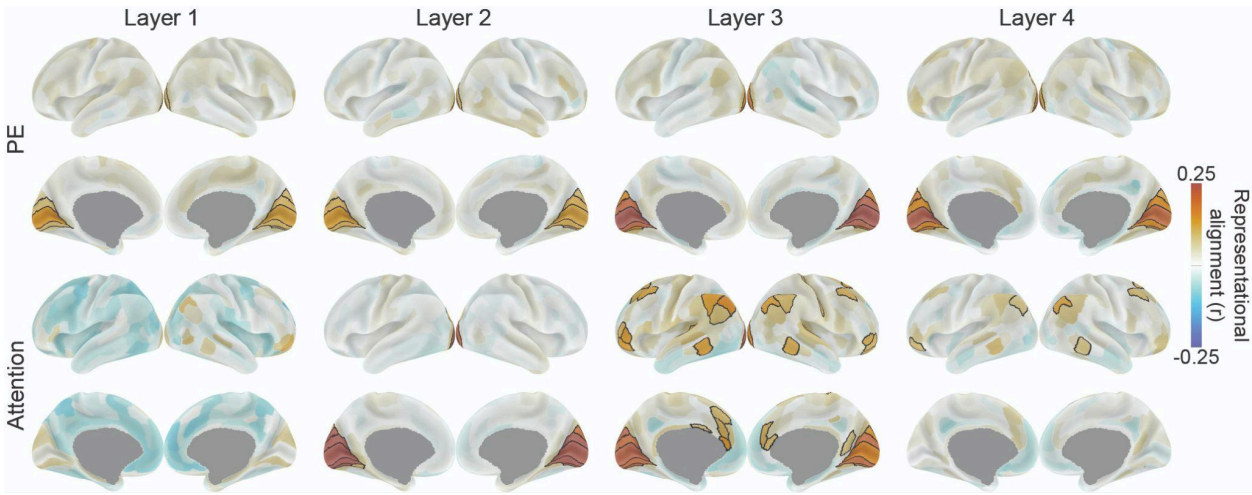

Fig. 5. *Layer-dependent representational alignment between transformer and human fMRI during relational reasoning.* Black borders indicate significant representational alignment (permutation and false discovery rate corrected).

Discussion

We investigated the cognitive and computational mechanisms of relational reasoning. To do so, we analyzed representations of both ANNs and human task fMRI data during LST performance. While standard ANN architectures used in computational neuroscience failed at generalizing the LST (i.e., MLPs, LSTMs), transformers were able to generalize the LST reliably. By isolating the representations produced by the transformer's unique components (through averaging) – namely its PE and attention mechanism – we characterized the precise representations produced by the transformer for successful LST performance. Specifically, PE produced representations that were highly sensitive to the visual, grid structure of the LST puzzle. On the other hand, the attention mechanism produced representations that aligned with the relational complexity conditions of the LST. We further found that the representations produced by each architectural component could be similarly observed in the human brain; PE and attention representations were representationally aligned to neural systems responsible for

visual and higher-order cognitive control representations, respectively. When focusing on inter-layer transformer representations rather than intra-layer architectural components, we found a systematic transition from lower- to higher-order neural representations across successive layers, mirroring hierarchical processing observed in other biological systems (e.g.,[20]). Our findings build a comprehensive account of both the computational mechanisms required to perform and generalize relational reasoning in the LST, and how they correspond to those in the human brain.

Prior studies in computational neuroscience that have bridged ANNs with neural data have primarily used architectures such as recurrent neural networks (RNNs), MLPs, or convolutional neural networks (CNNs)[20,23,25,41–44]. The predominance of these models reflects their relative simplicity, as well as their historical roots in neuroscience[45–47], whereas the transformer architecture was developed independently from neuroscience[13]. Nevertheless recent studies have identified potential links between transformers and specific neural systems, such as the hippocampus[31], visual cortex[30], cortico-thalamic circuits[29], and language networks[48]. In the present study, we extend this line of research by investigating the neural computations of specific transformer components – PE and attention – while relating them to whole-cortex fMRI data during relational reasoning.

We focus on the two architectural components that are unique to transformers: PE and attention. PE has been shown to be a strong and inductive bias for transformers. While most implementations of PEs employed by transformers are optimized for 1D text[13,49–51], a number of studies have demonstrated the importance of PE for more structured tasks (rather than natural language), such as arithmetic[52–54], or tasks requiring multiple dimensions (e.g., 2D tasks[55]). Here we illustrated the importance of either having a PE that preserved the original 2D arrangement of the LST (in contrast to the other neural network architectures, which do not), or *learning* the correct 2D arrangement using a learnable PE parameter (**Fig. 3**). Critically, solving the LST is significantly more challenging given an unordered sequence (e.g., if the LST grid was flattened from 2D to 1D, **Fig. 2B**). Thus, tracking the positional arrangement of symbols in the LST is essential to capturing their relational structure, a function fulfilled by the transformer's attention mechanism. Supporting this, we found that the representations produced from the attention mechanism correlate with the relational complexity of the LST items (**Fig. 2E**). Moreover, attention weights became increasingly dense as problem complexity grew (**Fig. 4D**), reflecting greater allocation of importance (or "attention") across tokens. This provided a computational analog of the working memory capacity demands required for the LST task. Together, these findings suggest that PE and attention work in concert: PE anchors tokens to their spatial positions, and attention leverages these positions in order to provide a relational understanding of LST problems.

Although transformers have been predominantly developed to power large language models, their broader applicability across domains – such as within neuroscience – remains relatively nascent. While transformers were not designed with biological plausibility in mind, this does not preclude their utility as computational models for understanding neural systems and brain computation[19]. In particular, the construction and manipulation of transformer representations offers a rich substrate for investigating principles of biological computation and representations. Here we illustrate how attention-based computation, position-sensitive encodings, and hierarchical abstractions – all hallmarks of transformer models – offer valuable

insights into the principles of relational reasoning in biological information processing. Though other studies have used simpler models to model cognitive functions in neuroscience[17,56], we found that such models were unable to generalize a more sophisticated reasoning task like the LST. Importantly, in this study we leverage the unique ability of transformers (as an artificial neural architecture) and humans (as a model organism) to perform the LST to study the computational underpinnings of complex relational reasoning. Thus, even in the absence of direct neurobiological inspiration, transformer-based models may serve as useful tools for providing a computational language to describe complex cognition through careful analysis of their internal mechanisms and representations.

Our findings reveal converging computational strategies for relational reasoning in humans and transformers. This opens up several promising directions for future research. First, while the LST is a complex and sophisticated task within neuroscience research, transformers have demonstrated the ability to learn even more complex reasoning tasks that have yet to be explored within neuroscience, such as Chess[57], Go[58], and the abstract reasoning corpus[59]. Though designing feasible experiments to collect neural data for these tasks will be challenging, characterizing the neural mechanisms underlying reasoning across a diverse set of complex tasks that are both challenging for humans and artificial systems offers exciting directions for future work. Second, although standard implementations of MLPs and LSTMs failed to generalize on the LST, it is possible that non-standard training set ups, learning regimes, or architectural variants may confer improved performance. A systematic investigation of these possibilities could clarify what additional parameters are needed to enhance MLPs and LSTMs to natively perform the LST. Finally, though we use machine learning models to investigate the computational mechanisms of relational reasoning in humans, future work should explore more circuit-based, and biologically meaningful mechanisms through the use of biophysical models or, where possible, neural data collected from other animal models.

In summary, we investigated the computational mechanisms key for the relational reasoning required in the LST using artificial neural network models and fMRI data. Among commonly used neural architectures, we found that the transformer architecture generalized the LST most accurately. Despite having origins outside of neuroscience, the transformer architecture offered a powerful computational framework for modeling relational reasoning and its neural correlates. By analyzing the contributions of PE and attention to LST generalization, we revealed how these components jointly support the representation and manipulation of structured information, paralleling visual and cognitive control systems in the human brain. Moreover, the hierarchical progression of transformer representations across layers mirrored cortical processing hierarchies. This suggests shared algorithmic strategies in performing the LST in biological and gradient-descent-optimized artificial systems, underscoring the potential of transformer models as tools for probing the computational mechanisms of cognition. We hope this study inspires further work in leveraging more powerful artificial models to investigate the neural correlates of more sophisticated cognitive tasks.

Methods

Participants

Sixty-five healthy, right-handed adult participants completed the LST while undergoing fMRI. Of these, 40 were included in the final analysis (mean age = 23 years old; age range = 18 - 33; 60% female). Four participants were excluded due to MR scanning errors, one was excluded due to an unexpected brain structure abnormality, another was excluded for low accuracy in the behavioral task (total score more than three standard deviations below average) and 19 participants were excluded due to excessive head movement (note a stringent head motion threshold was employed - described below in the Neuroimaging section). Participants provided written informed consent and were eligible if they were aged between 18 - 35, with no previous reported history of mental health or neurological disorders. Ethics approval was granted by the University of Queensland Human Research Ethics Committee. This dataset has been used in previous publications[9,60,61].

Latin Square Task (LST) experiment design

The LST is a nonverbal relational reasoning task[32]. Each puzzle is a two-dimensional 4x4 grid populated with shapes (squares, triangles, circles, and crosses) and a target question mark. Participants must determine the shape of the target, ensuring each shape appears only once in every row and column, similar to Sudoku. Puzzles were organised into three conditions with increasing reasoning complexity[1]:

- One-vector: linking puzzle elements (shapes) across a single row or column
- Two-vector: linking elements across both a row and a column
- Three-vector: linking elements across three rows or columns.

Details of the fMRI implementation of the LST have been described previously[9,34]. In brief, 108 LST items (36 per relational complexity condition) were presented in the MR session across three runs. Participants viewed each puzzle for five seconds, followed by a brief pause and then a response screen. Motor responses were counterbalanced across participants, ensuring that an equal number of participants had the same shape-response mapping. Stimuli were presented using PsychToolBox 3.0[62]. In addition to the LST, participants also completed the Raven's Advanced Progressive Matrices with a 40 minute time limit as a standard measure of fluid intelligence[63].

Neuroimaging

Acquisition

High resolution fMRI data were acquired on a 7 Tesla Siemens scanner equipped with a 32-channel head coil at the Centre of Advanced Imaging, The University of Queensland, Brisbane. Whole-brain echo-planar images were collected with the following parameters: isotropic 2 mm voxels, TR = 586ms, TE = 23ms, flip angle = 40°, FOV = 208mm, 55 axial slices, multiband acceleration factor = 5. Structural scans (MP2RAGE) used in the preprocessing

pipeline were acquired with the following parameters: isotropic 0.75 mm voxels, TR = 4300ms, TE = 3.44 ms, and 256 slices.

Minimal preprocessing and denoising

Neuroimaging data were minimally preprocessed using a standard fMRIPrep pipeline (version 21.0.1)[64 60]. In brief, this pipeline consisted of skull stripping, susceptibility distortion correction, coregistration, slice time correction and spatial normalisation (full details in **Supplementary Material**). The analyses in this manuscript focus on multi-vertex patterns, therefore any participants exceeding 2 mm of head motion in the six rotation and translation parameters were excluded from further analysis ($N_{excluded}$=19, as in[60]). Additional denoising was performed by regressing fourteen nuisance signals (mean cerebrospinal fluid and white matter signals, along with the six motion parameters and their derivatives). Denoising was performed using the *clean* function in Nilearn[65].

fMRI task activation estimation

Individual-level brain activations were estimated by conducting vertex-wise general linear model (GLM) analysis in grayordinate space[66]. In line with prior work[67], a separate GLM was conducted for each puzzle in the experiment (N=108). Each design matrix included two regressors; one for the puzzle of interest, and one for all other puzzles in the experiment. Each puzzle was modelled as a five second boxcar regressor convolved with the Statistical Parametric Mapping [68] canonical hemodynamic response function. Three additional regressors were included to account for each run of the experiment. Task GLMs were performed using the *run_glm* function in Nilearn [65]. In subsequent analysis of fMRI data we adopted the Glasser cortical parcellation [66] by analyzing vertices within each parcel. We selected this parcellation because it provides a better description of task-related brain organisation when contrasted with atlases based solely on resting-state fMRI [69].

Artificial Neural Networks (ANN)

We trained three ANNs with distinct architectures to perform the LST. Three thousand training LST puzzles were generated for each of the one-, two-, or three-vector conditions. The aforementioned 108 fMRI experiment puzzles were used for validation and comparison to empirical data. We ensured that the similarity between the training and validation puzzles was low (Jaccard dissimilarity > 0.8). For all models, we used the Adam optimizer with a learning rate of 0.0001 without regularization (no dropout or weight decay). We trained on 15 independent seeds for a fixed number of epochs (4000). Each model trained on 8000 of the original 9000 puzzles, holding out 1000 puzzles as an additional test set.

Transformer

The input to the transformer was a one-hot encoding of the symbol for each element in the grid. Since there were five possible symbols, each token was represented as a 5-dimensional vector. If the grid element was empty, the token was a vector of zeros. We used a standard encoder-only transformer architecture with four layers and model embedding dimension of 160

[13]. The context window for the model was 16 tokens long, given the two-dimensional 4 × 4 structure of the LST. Each model contained fully-connected bidirectional attention. For simplicity (and interpretability) we limited the number of attention heads to 1. We used the standard position-wise MLP in the transformer with a single hidden layer with an expansive embedding dimension that was four times larger than the token embedding dimension (160*4) and a GELU nonlinearity. For PE, we used a 2D absolute PE and a separate learnable embedding (described in more detail below). Both these PE variants were incorporated to the token embedding via vector addition.

More formally, let X be a 16×160 matrix (number of tokens × embedding dimensions) that denotes our input tokens, P be our PE matrix of the same dimensions, $\mathbf{A}$ be the attention mechanism (defined here as a function), and $\mathbf{M}$ be the MLP (defined as a function). Then, a single transformer layer is computed as

$$X \leftarrow \mathbf{M}(\mathbf{A}(X + P) + X + P) + \mathbf{A}(X + P)$$

where the additional $X + P$ and $\mathbf{A}(X + P)$ terms are standard residual connections that are common in transformers. We define the attention mechanism $\mathbf{A}$ as $\mathbf{A}(X) = Softmax(\frac{QK^{\top}}{\sqrt{d_k}})V$, where $Q = XW_Q$, $K = XW_K$, and $V = XW_V$ with W_Q, W_K, W_V as learnable $d \times d$ (i.e., 160 x 160) matrices. The Softmax is a row-wise computation (applied to row vectors) defined as $Softmax(x)_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$, where x is a row vector in the $n \times n$ attention matrix ($n = 16$ for the LST task), and yields probability values (summed to 1) for all elements in the row vector. The intuition is that the highest probability value is assigned to the token that is most strongly attended to. The function $\mathbf{M}$ is a learnable MLP specified above with a single hidden layer. Note that we assume LayerNorms, a common normalization technique in transformers, to be carried out after functions $\mathbf{A}$ and $\mathbf{M}$. We use standard placements of the LayerNorms (as specified in Vaswani et al., 2017), which can be found in the code (to be made publicly available). For training, we used the Adam optimizer with a learning rate of 0.0001.

Positional encoding (PE) in transformers

In this analysis we explored the impact of PE on downstream model performance and representations. The two-dimensional LST puzzle grid can be explicitly modelled in the PE of the transformer. In this context, the PE reflects the model's ability to grasp the 2D task structure—without an understanding of the puzzle's square layout, it cannot access the row and column information needed to solve it visually. Two-dimensional absolute PEs are a generalization of 1d-fixed presented in prior work[13]. The primary distinction is that, rather than using a single position variable where $1 < pos < 16$, there are two position variables: $1 \leq pos_w \leq 4$ and $1 \leq pos_h \leq 4$, representing the width and height of the grid, respectively.

PEPositional encoding for a row w is defined by:

$$PE_{(pos_w, 2i)} = sin(\frac{pos_w}{10000^{2i/d_{model}}})$$

$$PE_{(pos_w, 2i+1)} = cos(\frac{pos_w}{10000^{2i/d_{model}}})$$

PEPositional encoding for a column h is defined by

$$PE_{(pos_h,2i)} = sin(\frac{pos_h}{10000^{2i/d_{model}}})$$
$$PE_{(pos_h,2i+1)} = cos(\frac{pos_h}{10000^{2i/d_{model}}})$$

For a 2D PE, half of the embedding dimensionality is allocated to encoding rows, and the other half to columns. Thus, for $d_{model} = 160$, embedding dimensions 0–79 are reserved for row encoding and embedding dimensions 80–159 are reserved for column encoding. This PE formulation provides a strong inductive bias for 2D reasoning problems.

However, this inductive bias may not be a fair comparison with our control ANNs, which do not have 2D information explicit into the model architecture. Unlike transformers with a 2D PE, MLPs and LSTMs need to learn a 2D representation through weight updates via gradient descent. We therefore used a transformer with no positional inductive bias, and instead a learnable free PE parameter (represented as a 160-dimensional vector per token). Prior work in deep learning theory suggests that the choice of weight initialization can influence both the structure of learned representations and task generalization [39,40,70]. Specifically, smaller initialization norms tend to promote more structured internal representations. We assessed whether these findings would extend to learnable transformer PEs initialized from different Normal distributions, controlling for the standard deviation (i.e., norm). For each token embedding, we initialized a learnable PE parameter from a multivariate Normal distribution, denoted $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma} = \sigma\mathbf{I}$, and $\mathbf{I}$ denotes the identity matrix scaled by σ. As in the prior analysis, we trained 15 independent seeds across 4000 epochs. All models converged in training. We then tested whether there was an interaction between reasoning complexity condition and initialization norm in regards to model generalisation accuracy.

Control ANNs

Two additional ANNs were assessed in their ability to complete the LST, i) a feedforward fully-connected MLP and, ii) a bidirectional Long Short-Term Memory (LSTM) network. Each of the control models contained an input layer, four hidden layers of 160 units, and an output layer. Both ANNs were implemented with ‘out-of-the-box’ PyTorch models, and are included with the code [71]. Despite all models and seeds converging during training, these models did not exhibit meaningful generalization to unseen LST puzzles.

The input to the MLP was a concatenation of the one-hot encodings of all grid elements. As there were 5 possible symbols (per grid element) and 16 possible grid locations, this yielded a 80-dimensional input vector. The input to the LSTM (as a sequence model) was to present each LST grid element (as a 5-dimensional vector) in sequence, both forwards and backwards (i.e., bidirectionally). In other words, the LSTM recurrently iterated 32 (16*2) times prior to forming a decision. This contrasts with the transformer, where each token is processed separately (but in parallel) and PE is applied separately to each token. In the MLP and LSTM, grid position information is implicitly encoded, as each token has its own set of unique input channels/weights in the MLP, and each token is processed in ordered sequence in the LSTM. Thus, MLPs and LSTMs are conceptually similar to transformers with a learnable PE, as those

models need to learn which set of input channels/tokens belong to which grid position implicitly via weight updates. Note, however, that the learnable PE in transformers is a vector parameter, whereas in MLPs and LSTMs it learns grid position through weight updates on the input embeddings. In the absence of the ability to accurately learn the correct 2D positional structure, the formulation of the MLP effectively encodes a randomized PE of grid elements, as all input channels are orthogonal to each other at initialization.

Representational Similarity Analysis (RSA)

We used representational similarity analysis (RSA) to contrast experimenter task models, transformer model activations and fMRI brain activations [72]. Three task models were contrasted; stimulus, complexity and response (shown in **Fig. 2B**). The stimulus model was calculated with $1 - jaccard\ index$ between each of the 108 one-hot encoded puzzles. Thus, similar puzzle geometries had scores closer to 0 and vice versa. The complexity model represented the different conditions of the experiment whereby each condition was coded to be maximally similar (0), adjacent conditions where dissimilar (0.5, e.g. one-vector vs. two vector) and one-vector and three-vector puzzles were maximally dissimilar (1). Finally, as in the stimulus model, the response model was calculated with $1 - jaccard\ index$ between each of the 108 one-hot encoded solutions. For the transformer and fMRI data we used *rsatoolbox* to generate representations [37]. Specifically, given there is no noise in the measurement of transformer neural activity, we used the squared Euclidean distance to derive representational dissimilarity matrices (RDMs), and then averaged these values across the 15 seeds. This was performed for activations produced by both the PE and Attention mechanisms (i.e., qkv). For the fMRI data, since each puzzle was presented once to each participant, we performed cross-validation across-participants (rather than *within participant*) using the squared Mahalanobis distance ("crossnobis" distance). This resulted in a single RDM per brain region in the parcellation.

The upper triangle of the RDMs were compared using Pearson correlation; we refer to this as "representational alignment", consistent with prior studies[73]. For the comparison between task models and transformer RDMs, these values were submitted to a repeated-measures ANOVA, with model seeds treated as subjects. For the comparison between transformer and brain imaging, significance was ascribed using a permutation method, whereby the condition labels (i.e., puzzles) were reordered within the transformer RDMs, and then compared with the empirical fMRI RDM. This was repeated 10,000 times to generate a null distribution of unrelated RDMs for every brain region[72]. For a given brain region, if the true correlation exceeded the 95th percentile, the null hypothesis was rejected. In addition to this procedure, we performed false discovery rate (FDR) correction across the 360 brain regions. We also examined the relationship between brain-transformer (PE) representational alignment and initialization norms within regions of interest which had shown an effect in the main analysis (i.e., the primary visual cortex).

Attention score analysis

In a final analysis we sought to relate attention weights (qk values) to cognitive features of the LST. Attention weights are the token by token (in our case 16 x 16) matrices denoting how each puzzle element relates to one another. These values were averaged over layers. To start, we assessed whether attention weights were higher for the target element ("?") in each puzzle. To do so, we compared the average the target's key (k) value across its queries (q) (i.e., the mean of $qk_{:,T}$ where T is the target index) to non-target values. Next, to further examine this effect, we examined whether the critical cells in a puzzle had higher attention weights than non-critical cells. Critical cells were defined as elements in the grid that directly contributed to the prediction of the element in the target location. We focussed on one-vector puzzles where the critical cells are always within the same row or column as the target. We extracted attention weights relevant to critical cells and the target (the mean of $qk_{critical,T}$) and contrasted them with non-critical attention weights. Lastly, we tested whether the target attention weights (the mean of $qk_{:,T}$) differed across the three relational complexity conditions. We therefore came up with a statistic that measured the density of the attention matrix, described below. Because the sum of each attention weight across *keys* is summed to 1 (due to the softmax function, which transforms raw attention weights to probability scores), the "denser" the attention weights (i.e., the more tokens attended to), the lower the average attention score is for each token. While the sum across keys (row sum) is fixed (as they sum to 1), the query (column sum) is not. Therefore, by computing the sum across queries (column sum), we could statistically infer that the lower the attention weights across queries, the denser the attention weights would be.

References

1. Halford, G. S., Wilson, W. H. & Phillips, S. Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and brain sciences* **21**, 803–831 (1998).
2. Waltz, J. A. *et al.* A system for relational reasoning in human prefrontal cortex. *Psychological science* **10**, 119–125 (1999).
3. Cattell, R. B. A culture-free intelligence test. I. *Journal of Educational Psychology* **31**, 161 (1940).
4. Raven, J. C. *Raven's Progressive Matrices (1938): Sets A, B, C, D, E*. (Australian Council for Educational Research, Melbourne, 1938).
5. Holyoak, K. J. & Lu, H. Emergence of relational reasoning. *Current Opinion in Behavioral Sciences* **37**, 118–124 (2021).
6. Penn, D. C. & Povinelli, D. J. Causal cognition in human and nonhuman animals: A comparative, critical review. *Annu. Rev. Psychol.* **58**, 97–118 (2007).

7. Wendelken, C., Nakhabenko, D., Donohue, S. E., Carter, C. S. & Bunge, S. A. “Brain is to thought as stomach is to??”: investigating the role of rostrolateral prefrontal cortex in relational reasoning. *Journal of cognitive neuroscience* **20**, 682–693 (2008).
8. Krawczyk, D. C., McClelland, M. M. & Donovan, C. M. A hierarchy for relational reasoning in the prefrontal cortex. *Cortex* **47**, 588–597 (2011).
9. Hearne, L. J., Cocchi, L., Zalesky, A. & Mattingley, J. B. Reconfiguration of Brain Network Architectures between Resting-State and Complexity-Dependent Cognitive Reasoning. *Journal of Neuroscience* **37**, 8399–8411 (2017).
10. Cocchi, L. *et al.* Complexity in relational processing predicts changes in functional brain network dynamics. *Cerebral Cortex* **24**, 2283–2296 (2014).
11. Robinson, C., Cocchi, L., Ito, T. & Hearne, L. Relational Integration Demands Are Tracked by Temporally Delayed Neural Representations in Alpha and Beta Rhythms Within Higher‐Order Cortical Networks. *Human Brain Mapping* **46**, e70272 (2025).
12. Knowlton, B. J., Morrison, R. G., Hummel, J. E. & Holyoak, K. J. A neurocomputational system for relational reasoning. *Trends in cognitive sciences* **16**, 373–381 (2012).
13. Vaswani, A. *et al.* Attention is All you Need. in *Advances in Neural Information Processing Systems* vol. 30 (Curran Associates, Inc., 2017).
14. Brown, T. *et al.* Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020).
15. Geerts, J., Chan, S., Clopath, C. & Stachenfeld, K. Relational reasoning and inductive bias in transformers trained on a transitive inference task. *arXiv preprint arXiv:2506.04289* (2025).
16. Lewkowycz, A. *et al.* Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems* **35**, 3843–3857 (2022).
17. Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T. & Wang, X.-J. Task representations in neural networks trained to perform many cognitive tasks. *Nature neuroscience* **22**, 297–306 (2019).
18. Churchland, P. S. & Sejnowski, T. J. Blending computational and experimental neuroscience. *Nature Reviews Neuroscience* **17**, 667–668 (2016).

19. Doerig, A. *et al.* The neuroconnectionist research programme. *Nature Reviews Neuroscience* 1–20 (2023) doi:10.1038/s41583-023-00705-w.
20. Yamins, D. L. K. *et al.* Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences* **111**, 8619–8624 (2014).
21. Banino, A. *et al.* Vector-based navigation using grid-like representations in artificial agents. *Nature* **557**, 429–433 (2018).
22. Cueva, C. J. & Wei, X.-X. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. https://openreview.net/forum?id=B17JTOe0- (2018).
23. Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature* **503**, 78–84 (2013).
24. Flesch, T., Juechems, K., Dumbalska, T., Saxe, A. & Summerfield, C. Orthogonal representations for robust context-dependent task performance in brains and neural networks. *Neuron* **0**, (2022).
25. Ito, T., Yang, G. R., Laurent, P., Schultz, D. H. & Cole, M. W. Constructing neural network models from brain data reveals representational transformations linked to adaptive behavior. *Nature Communications* **13**, 673 (2022).
26. Ito, T. *et al.* Learning interpretable positional encodings in transformers depends on initialization. in (2025).
27. Bahdanau, D., Cho, K. & Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).
28. Geva, M., Schuster, R., Berant, J. & Levy, O. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913* (2020).
29. Granier, A. & Senn, W. Multihead self-attention in cortico-thalamic circuits. *arXiv preprint arXiv:2504.06354* (2025).
30. Adeli, H., Sun, M. & Kriegeskorte, N. Transformer brain encoders explain human high-level visual responses. *arXiv preprint arXiv:2505.17329* (2025).
31. Whittington, J. C. R., Warren, J. & Behrens, T. E. J. Relating transformers to models and neural representations of the hippocampal formation. Preprint at http://arxiv.org/abs/2112.04035 (2022).

32. Birney, D. P., Halford, G. S. & Andrews, G. Measuring the influence of complexity on relational reasoning: The development of the Latin Square Task. *Educational and Psychological Measurement* **66**, 146–171 (2006).

33. Birney, D. P. & Bowman, D. B. An experimental-differential investigation of cognitive complexity. *Psychological Test and Assessment Modeling* **51**, 449 (2009).

34. Hearne, L. J., Birney, D. P., Cocchi, L. & Mattingley, J. B. The Latin Square Task as a Measure of Relational Reasoning: A Replication and Assessment of Reliability. *European Journal of Psychological Assessment* 1–7 (2019) doi:10.1027/1015-5759/a000520.

35. Seely, J., Imajuku, Y., Zhao, T., Cetin, E. & Jones, L. Sudoku-Bench: Evaluating creative reasoning with Sudoku variants. *arXiv preprint arXiv:2505.16135* (2025).

36. Diedrichsen, J. & Kriegeskorte, N. Representational models: A common framework for understanding encoding, pattern-component, and representational-similarity analysis. *PLOS Computational Biology* **13**, e1005508 (2017).

37. van den Bosch, J. J. *et al.* A Python Toolbox for Representational Similarity Analysis. *bioRxiv* 2025–05 (2025).

38. Sucholutsky, I. *et al.* Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018* (2023).

39. Chizat, L., Oyallon, E. & Bach, F. On Lazy Training in Differentiable Programming. *arXiv:1812.07956 [cs, math]* http://arxiv.org/abs/1812.07956 (2020).

40. Woodworth, B. *et al.* Kernel and rich regimes in overparametrized models. in 3635–3673 (PMLR, 2020).

41. Cohen, J. D., Dunbar, K. & McClelland, J. L. On the control of automatic processes: a parallel distributed processing account of the Stroop effect. *Psychological review* **97**, 332 (1990).

42. Russo, A. A. *et al.* Neural Trajectories in the Supplementary Motor Area and Motor Cortex Exhibit Distinct Geometries, Compatible with Different Classes of Computation. *Neuron* https://doi.org/10.1016/j.neuron.2020.05.020 (2020) doi:10.1016/j.neuron.2020.05.020.

43. Khaligh-Razavi, S.-M. & Kriegeskorte, N. Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. *PLOS Computational Biology* **10**, e1003915 (2014).

44. Kar, K., Kubilius, J., Schmidt, K., Issa, E. B. & DiCarlo, J. J. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nature neuroscience* **22**, 974–983 (2019).

45. Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics* **36**, 193–202 (1980).

46. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* **65**, 386–408 (1958).

47. McCulloch, W. S. & Pitts, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics* **5**, 115–133 (1943).

48. Tuckute, G. *et al.* Driving and suppressing the human language network using large language models. *Nature Human Behaviour* **8**, 544–561 (2024).

49. Shaw, P., Uszkoreit, J. & Vaswani, A. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155* (2018).

50. Raffel, C. *et al.* Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research* **21**, 1–67 (2020).

51. Su, J. *et al.* Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* **568**, 127063 (2024).

52. Saxton, D., Grefenstette, E., Hill, F. & Kohli, P. Analysing Mathematical Reasoning Abilities of Neural Models. (2019) doi:10.48550/arXiv.1904.01557.

53. Delétang, G. *et al.* Neural networks and the chomsky hierarchy. *arXiv preprint arXiv:2207.02098* (2022).

54. Shen, R. *et al.* Positional description matters for transformers arithmetic. *arXiv preprint arXiv:2311.14737* (2023).

55. Ito, T. *et al.* Learning interpretable positional encodings in transformers depends on initialization. in (2025).

56. Driscoll, L. N., Shenoy, K. & Sussillo, D. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nature Neuroscience* **27**, 1349–1363 (2024).

57. Silver, D. *et al.* A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **362**, 1140–1144 (2018).
58. Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. *nature* **529**, 484–489 (2016).
59. Chollet, F. On the Measure of Intelligence. (2019).
60. Robinson, C. N., Cocchi, L., Ito, T. & Hearne, L. J. Temporally delayed representations in alpha and beta rhythms in higher-order cortical networks track increasing relational integration demands. *bioRxiv* 2024–10 (2024).
61. Shine, J. M. *et al.* The Low-Dimensional Neural Architecture of Cognitive Complexity Is Related to Activity in Medial Thalamic Nuclei. *Neuron* **104**, 849-855.e3 (2019).
62. Brainard, D. H. & Vision, S. The psychophysics toolbox. *Spatial vision* **10**, 433–436 (1997).
63. Bors, D. A. & Stokes, T. L. Raven's Advanced Progressive Matrices: Norms for First-Year University Students and the Development of a Short Form. *Educational and Psychological Measurement* **58**, 382–398 (1998).
64. Esteban, O. *et al.* fMRIPrep: a robust preprocessing pipeline for functional MRI. *Nature Methods* 1 (2018) doi:10.1038/s41592-018-0235-4.
65. Abraham, A. *et al.* Machine learning for neuroimaging with scikit-learn. *Frontiers in neuroinformatics* 14 (2014).
66. Glasser, M. F. *et al.* A multi-modal parcellation of human cerebral cortex. *Nature* **536**, 171–178 (2016).
67. Mumford, J. A., Turner, B. O., Ashby, F. G. & Poldrack, R. A. Deconvolving BOLD activation in event-related designs for multivoxel pattern classification analyses. *Neuroimage* **59**, 2636–2643 (2012).
68. Friston, K. J. *et al.* Statistical parametric maps in functional imaging: a general linear approach. *Human brain mapping* **2**, 189–210 (1994).
69. Schaefer, A. *et al.* Local-Global Parcellation of the Human Cerebral Cortex from Intrinsic Functional Connectivity MRI. *Cereb Cortex* **28**, 3095–3114 (2018).

70. Jacot, A., Gabriel, F. & Hongler, C. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. *arXiv:1806.07572 [cs, math, stat]* http://arxiv.org/abs/1806.07572 (2020).

71. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32**, (2019).

72. Kriegeskorte, N., Mur, M. & Bandettini, P. Representational similarity analysis - connecting the branches of systems neuroscience. *Frontiers in systems neuroscience* **2**, 4 (2008).

73. Ito, T. & Murray, J. D. *Multi-Task Representations in Human Cortex Transform along a Sensory-to-Motor Hierarchy*. 2021.11.29.470432 https://www.biorxiv.org/content/10.1101/2021.11.29.470432v1 (2021) doi:10.1101/2021.11.29.470432.