What underlies rapid learning and systematic generalization in humans?

Andrew Joohun Nam
Department of Psychology
Stanford University
ajhnam@stanford.edu

James L. McClelland
Department of Psychology
Stanford University
jlmcc@stanford.edu

Abstract

Humans can sometimes learn a procedure from one or a small number of examples and then apply what they have learned to a much larger range of examples. Here, we explore this ability as it arises in learning a strategy from Sudoku. Participants naive to Sudoku went through a tutorial explaining a procedure for solving a specific instance of a general type of situation that arises in Sudoku without reference to a general rule. They then received practice with explanatory feedback on examples closely related to the one in the tutorial. Most of those who acquired the procedure from this experience did so within a small number of examples and, in a subsequent test, readily transferred what they had learned to examples outside the distribution of training examples. These participants' learning was better characterized as a series of discrete transitions than as a continuous shift across strategies, and most of these participants described a reliably identifiable, valid strategy when asked to report how they solved one final puzzle. However, less than half of participants succeeded in acquiring the procedure, and success was associated with education, particularly basic mathematics education: no participants who reported having taken neither high-school algebra nor geometry successfully acquired the procedure. We present these findings as constraints on computational principles guiding models of human intelligence for learning generalizable reasoning skills.

1 Introduction

The capacity for people to take advantage of regularities in domains such as natural language has raised the question of whether the explicit representation of compositional structure is an inherent property of the human mind or is a consequence of relevant experiences [13, 32], with each perspective endorsed by classical and emergentist, neural network approaches to cognition respectively. While the latter models make minimal a priori structural assumptions about the internal representations and transformations of information, classical symbolic approaches build compositional structure into the model, attributing shortcomings of early neural network models to failures to do so [13, 23]. Although deep learning models have found groundbreaking successes in image recognition [7], natural language translation [40], and super-human game playing abilities [26, 36, 41], the criticisms regarding systematicity still persist. When put to the test, even the most impressive advancements in AI in the past decade fail to match up to humans' rate of learning [38], requiring massive data sets or self-generated experience (e.g. through self-play in game-playing models), and often fail to generalize outside the range of their training examples [20, 34]. In contrast, humans can sometimes learn something new and generalize broadly from one or a few training examples [2, 37, 21], motivating researchers to find alternative methods to induce systematicity. While some approaches, such as auxiliary loss signals [39, 16], attempt to address the issue while allowing for general model architecture, others have suggested hybrid models with architectures with a priori constraints to build in compositional representations and relevant domain knowledge that promote human-level fast learning and out-of-sample generalization [22].

However, despite the growing interest in systematicity and generalization, the base of empirical data focusing specifically on rapid human learning and generalization from one or a few examples is relatively thin. Existing studies of learning from one or a few examples rely on domains such as hand-written alphabets [21] or cultural practices [2] – domains where the participants can bring extensive prior experience to bear. Even in logical reasoning, an exemplary domain for the role of systematicity in human thought [13], humans can fail to exhibit consistency with basic laws of valid

logical inference [43], instead exhibiting strong dependence on the context of the reasoning problems they are asked to solve (see [18] for a review). Thus, many questions remain about the basis on which humans can learn abstractly and generalize to novel instances.

A central issue that requires deeper investigation is the role of instructions and explanations in humans' ability to exploit systematic structure within a domain. When understanding a new conceptual structure such as a cultural practice or scientific procedure [2], it has been observed that experimenter-provided explanation is a significant factor in allowing participants to learn from a single example. Humans can also learn to play Atari games through explicit instructions about the games [38], something that contemporary deep neural network models generally do not do since they rely instead on gradual, gradient-based learning driven solely by signals that specify either the correct response in a situation or by a reward outcome. This explicit instruction-driven learning has been suggested to result in more robust generalization compared to bottom-up learning that generalizes gradually from accumulation of similar experiences [35]. Moreover, even in tasks without instructions, people with formal education have been observed to more successfully infer abstract properties than people with only informal education, such as in identifying relevant and irrelevant features within classification tasks [8]. Indeed, it has been theorized that systematic reasoning is first acquired through a form of formal education which then transfers to enable spontaneous generation [42].

The present work seeks to contribute to the further investigation of these issues by exploring the following questions about rapid learning and out-of-domain transfer in humans in an abstract reasoning puzzle task. First, after successfully learning to solve the puzzles through a brief tutorial and a systematically constrained set of training examples, how well do humans generalize to out-of-distribution samples? A Classical model with built-in invariances would suggest that such resulting behavioral measures would be indistinguishable from within-distribution samples, whereas an emergentist model would suggest a possible performance decrement. Second, how rapidly do humans successfully acquire the solution strategy? More powerful inductive biases enable learning with fewer samples by constraining the learned representations and transformations. Thus, highly sample-efficient learning should suggest more sophisticated methods of incorporating information from individual trials, such as by leveraging explanations. Lastly, what factors contribute to people's ability to learn abstract procedures? If indeed formal education and use of instructions and explanations are powerful enablers for systematic reasoning, participants with more education and explanatory descriptions of their strategies should correlate with higher task performance.

We address these questions through a novel task which we call the Hidden Single puzzle, based on a solving technique of the same name in the puzzle Sudoku, which requires the solver to use the digits already present in a grid and the principle of mutual exclusivity to deduce the content of a single designated empty cell (see Section 2). The task is appealing as a domain in which to explore the general features of human systematic reasoning ability, since simple solution techniques, such as the Hidden Single technique as presented in our experiment, can be described in simple explanatory language without the need to appeal to technical concepts. Moreover, the task is characterized by the same symmetries, group properties, and combinatorics that characterize Sudoku in general [12, 33], allowing procedural transformations, such as re-assigning the roles of digits, shuffling rows and/or columns, and rotating or transposing the grid. This allows us both to explore the process of learning within a controlled task subspace and to assess how well learning generalizes outside of the narrow range of examples used in the tutorial and in an initial practice phase of the experiment. We contribute to the further understanding of the human ability to exploit instructions and explanation by using an explanatory instructional tutorial combined with feedback explaining why an answer is incorrect as participants learn the task through the tutorial and subsequent practice phase of the experiment.

We recruited 271 participants on Amazon Mechanical Turk after screening out many others who demonstrated or attested to prior Sudoku experience (See SI Section 1.1). They were presented a tutorial building up to the Hidden Single technique through a self-paced tutorial explaining the technique using a single example. To study the induction of a systematically generalizable strategy, we avoided abstract statements that describe the

Hidden Single technique in terms of a general principle or rule, and did not suggest that this technique should work in any other context than in the example provided. After completing 25 practice puzzles with explanatory feedback but with limited variation in puzzle features, participants were tested on 64 puzzles with outside-of-sample variations on several feature dimensions.

We found that only 1/3 of participants learned to consistently solve the puzzles by the end of the practice phase. Using a combination of accuracy, response time, response type, and survey measures, we provide evidence for the following four points about the performance and prior education of these successful learners. First, these participants could systematically generalize to puzzles outside of the training distribution, albeit with selective performance costs we will detail. Second, the learning was better characterized as a sequence of discrete transitions reaching mastery within about 10 trials rather than as incremental changes in solution strategies. Third, by the end of the experiment, most who acquired the strategy could give a valid description of their solution to a puzzle. Finally, self-reported education, and particularly education in basic high school mathematics, was associated with successful learning.

As a point of comparison with human performance, we compare human learning and generalization to that of a state-of-the-art neural network architecture designed to solve Sudoku [30]. We trained this network with the same restricted range of examples that we used with human participants in the tutorial and practice phase of our experiment. Unlike human participants, the neural network model exhibited perfect and immediate generalization to all feature variations that the model was designed for but no generalization whatsoever to variations in a feature that the original authors had not built into the model. Moreover, we also found that the model's learning dynamics were significantly slower and less data-efficient compared to our human participants, a pattern often seen in neural networks. We bring these findings together in the *Discussion*, noting that the standard approach in contemporary deep learning-based AI, as exemplified by the network in [30], does not seem well suited to capturing several aspects of human performance. We conclude the paper by proposing alternatives to domain-specific design to improve models for abstract reasoning and systematicity with three potentially synergistic future directions: exploiting sequential attention in task-agnostic neural network architectures, incorporating additional modalities to enable instruction-following and explanation, and promoting abstraction by employing meta-learning across multiple tasks that demonstrate generalizable procedures through specific examples.

2 Experiment Design

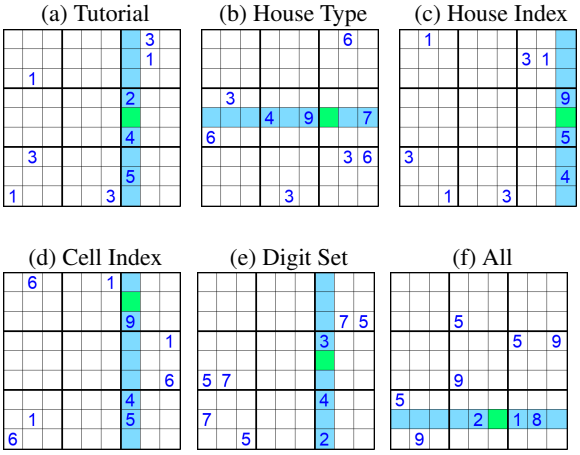


Figure 1: Examples of puzzles a participant might see, given the participant's individualized, random assignment of the training house type, house index, cell index, and digit set. (a) The full Hidden Single puzzle configuration as the participant might see it part way through the tutorial, conforming to the house type, house index, cell index, and containing the target (3) and the distractor (1) digits, both drawn from the training digit set. Examples used in the practice phase and control examples from the test phase would all use the same house type, house index, cell index, and training digit set, in this case the digits (1, 3, 6, 8). (b-f) Test phase examples with divergences from the tutorial and practice phase examples. (b-e) Puzzles with single feature divergences. HT = house type; HI = house index; CI = cell index; DS = digit set. (f) Puzzle with all possible features diverging from the tutorial example in (a).

The Hidden Single task (Figure 1) follows the same constraints as Sudoku: each row, column, and 3x3 box contains exactly one of each number from 1 to 9. However, the Hidden Single task simplifies the puzzle to finding the *target* digit that must go into a *goal* cell. The puzzles are constructed such that there is just one digit not already in the blue highlighted row or column that can be placed in the goal cell but not in any other cell. We procedurally generated every puzzle (See *Methods*) to have controlled variations while also maintaining standardized difficulty. Puzzles always contain 5 different

digits called *hints*, one of which is the target and another of which is the *distractor*. The target and distractor digits each have 3 instances arranged in the grid in a way that prevents participants from performing reliably based on perceptually obvious heuristics. The other three hints, called *in-house* digits, each occur once in the highlighted house. The remaining 4 of 9 digits, called *absent* digits, did not appear in the grid.

There are four experimentally controlled variable features for each puzzle. The *house type* is the type of house (row or column) to apply the Hidden Single technique to, indicated by whether it is a row or column that is highlighted in blue. The *house index* is the house to apply the Hidden Single technique to, also indicated by the blue highlighting. The *cell index* indicates which cell to solve for within a house, indicated by the cell highlighted in green. Lastly, the *digit set* is the set of 4 digits from which the target and distractor digits are drawn. The digits used in each puzzle are determined by first selecting a digit set, then assigning a number from the set as the target digit and a different number from the same set as the distractor digit. The three in-house digits are then randomly selected from the remaining 7 digits.

At the start of the experiment, each participant was randomly assigned a specific house type, house index, goal cell index, and training digit set to be used in the tutorial and practice puzzles. For each participant, a *transfer* set of four digits was then selected from the 5 digits remaining.

The *tutorial phase* of the experiment began with the one-sentence description of Sudoku stated previously. The tutorial then walked the participant through a sequence of screens presenting the logic of the Hidden Single technique, but without reference to it as such. Instead, the tutorial used different colors to highlight specific elements in the grid and refer to them, e.g. 'the purple cell' or the 'blue row' (see SI Figure 2), and did not make any indication that an abstraction of the pattern of reasoning could be applied as a general strategy. Throughout the sequence, participants were required to enter responses that, if incorrect, resulted in an explanation and a requirement to correct the error before proceeding.

Following the tutorial phase, each participant was given 25 puzzles to solve as part of a *practice phase*, without any mention of a relationship between the puzzles in this phase and the one they encountered in the tutorial. For each puzzle in this phase, participants were allowed unlimited attempts and time, with the goal of giving them the best chance of mastering the Hidden Single technique. All puzzles in this phase shared the same house type, house index, cell index, and digit set as the tutorial, only varying in the hint locations and the choices of specific digits to serve in the various hint roles, subject to the constraints imposed by the digit set. During this phase, all incorrect attempts produced detailed explanations specific to the puzzle and given response, referring to the particular digits and hints for why the response was incorrect.

Next, in the *test phase*, participants received 64 puzzles to solve with only one attempt and a 2-minute time limit for each puzzle. Here, feedback only indicated whether the response on a given trial was correct. In this phase, puzzles varied in terms of whether a particular feature (digit set, house type, house index, cell index) was changed or unchanged in the puzzle compared to its value during the tutorial and practice phases, yielding 16 possible puzzle conditions (including control puzzles which share all features with the tutorial and practice phase puzzles), with trial order carefully counterbalanced (see *Methods*). In presenting the results, we consider the effects of digit set, house type, and *goal position*, specifying whether the absolute position of the target cell was changed or not from its absolute position in the tutorial and practice phases.

3 Results

Although 271 participants completed the experiment, many did not succeed in acquiring a successful solution strategy from the combined learning experience provided by the tutorial and practice phases of the experiment. Since our interest focuses primarily on the transfer performance of the successful learners, whom we call *solvers*, we used a regression analysis applied to the practice phase data to identify these participants (see *Methods*).

This method identified 88 solvers, leaving 183 participants who we refer to henceforth as *non-solvers*, although some of these did eventually achieve high accuracy and a few others may have found a partially successful strategy (See SI Figure 11). In what follows, we focus primarily on the performance of the solvers, contrasting their performance with that of non-solvers in certain cases.

3.1 Identifying effects of systematic variation

We examined whether changing the digit set, house type, and goal cell position affected solvers' accuracy and response times across the 64 test phase trials. We present our evidence comparing the results in the first 16 trials and the remaining 48 sets separately to identify effects that may potentially be short-lived. Overall, there was substantial transfer across all variations, as detailed below. All analyses in this section were pre-registered after extensive pilot testing, with the exception of the analysis examining the effect of goal position. We first performed the committed analyses that were based on the cell index and house index variables; this resulted in a complex pattern that could be better understood by recoding the conditions in terms of the goal position variable. See SI Section 3 for preregistered but unreported regressions.

3.1.1 Digit sets

Solvers were able to transfer immediately when tested with target and distractor digits selected from a set never used in either of these roles during the tutorial or practice phases. Indeed, the effect of a change in the digit set, either for accuracy or response time, was negligible: all estimates were strikingly close to 0, which fell well within the 95% HDI, as shown in Table 1 and Figure 2, panels A and C.

3.1.2 House type

Solvers were able to apply what they had learned after a switch in house type, albeit with a small initial reduction in accuracy to 85% correct and a substantial initial increase in response time as shown in Figure 2. Although the 95% HDI for the effect of a change in house type on accuracy includes 0, we note that about 90% of the probability mass is below 0 and an effect of similar size with 0 falling outside the 95% HDI was obtained in pilot work reported in the preregistration of the current study [27]. Thus, while this small accuracy decrement is likely to be a real effect, it is noteworthy that it is small and very short-lived: the effect was more prominent in the first half of the first block of 16 trials than in the second (see Figure 2B) and is numerically reversed in trials 17-64.

Response time increased by 0.319 $\log_2$ seconds in the first 16 trials, or roughly a 25% increase from an average of 19.59 seconds to 24.44 seconds. The effect on RT decreases rapidly, down to 0.055 $\log_2$ seconds, or a 3.9% increase from 15.04 seconds to 15.63 seconds, in the last 48 trials and, as shown in Figure 2D, the effect appears to be gone by the end. We found no noteworthy difference for the direction of the house type shift, from row to column or column to row (see SI Section 2.4.1).

3.1.3 Goal position

A change in goal cell position produced a slowing of response times, and there was a trend toward a slight impact on accuracy toward the end of the test phase. For response times, we found main effects of 0.115 $\log_2$ seconds (8.3% increase) in the first 16 trials and of 0.089 $\log_2$ seconds (6.4% increase) in the last 48 trials. The 95% HDI did not include 0 in both cases. Numerically, a change in goal position was associated with a slight increase in accuracy in the first 16 test trials but 0 fell well inside the HDI for this effect. The decrease in accuracy in the last 48 trials was small in percentage terms (-0.314 logits, or less than 2% from a baseline over 95% correct), but the HDI for this effect just barely included 0.

In interpreting the effect of goal position, we note that, throughout the test phase, 25% of the puzzles used the same goal cell as the puzzles during the tutorial and practice phases, whereas when the goal position was changed, there were many other cells where it could end up, each with less than 25% probability. Thus, the persistent effect of goal position might result from a justifiable bias in attention toward the most common goal cell location, producing a small cost when attention must be deployed to a less likely position. (No such difference in relative likelihood applies either to the house type or the digit set variables, since both house types and both digit sets are used in the test problems with equal frequency.)

3.2 Dynamics of strategy acquisition

Having established the high level of transferability of the knowledge that solvers acquired, we next sought to characterize the dynamics of solvers' strategy acquisition by fitting a hidden Markov model (HMM) to the practice phase data. The model specifies participants' initial strategy distribution for the first practice trial, how this distribution changes from trial to trial via a strategy transition probability matrix, and the probabilities of different types of responses via a response emission matrix.

To specify the model we classified possible responses into four categories: 1. *target*: the correct

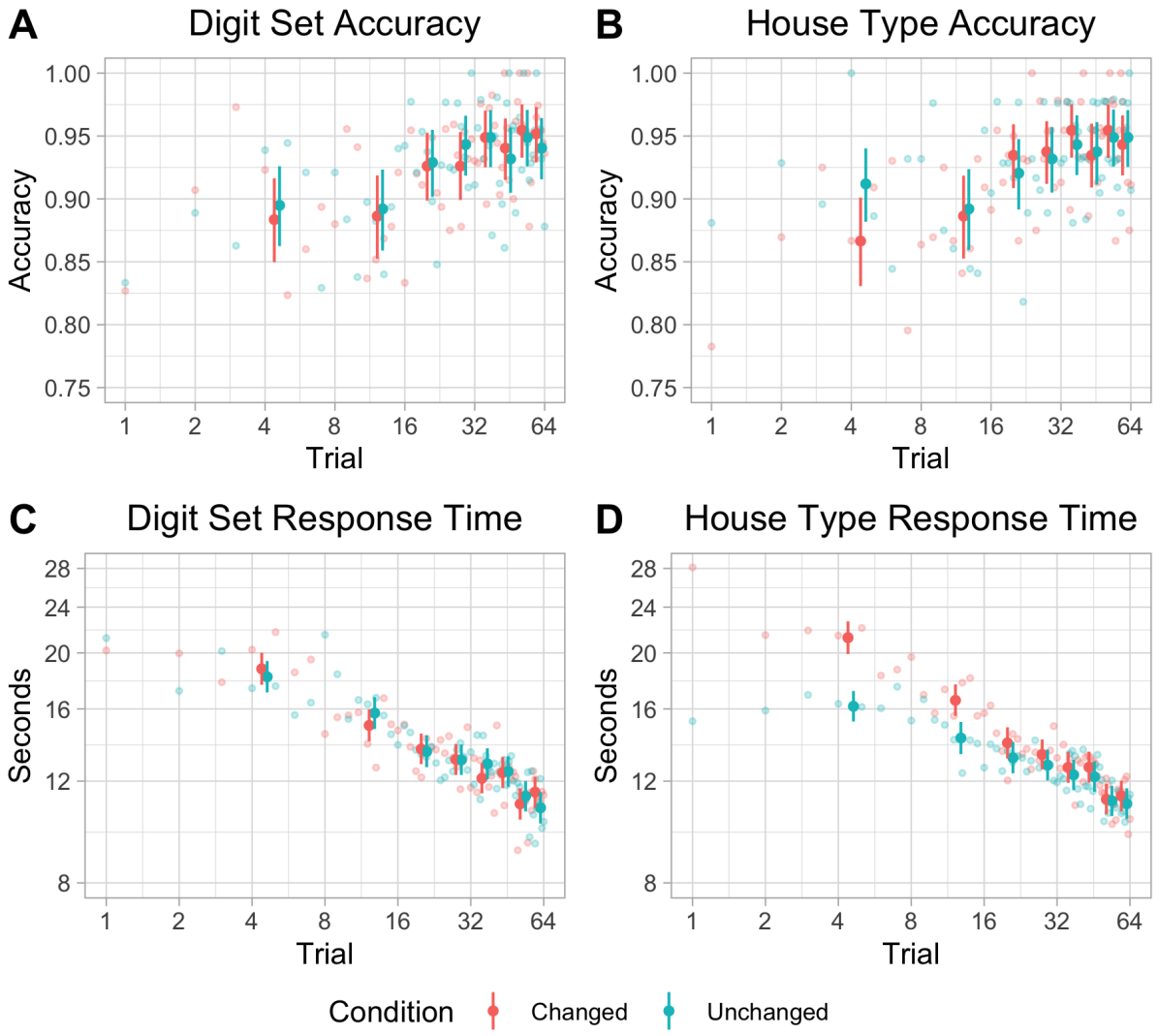
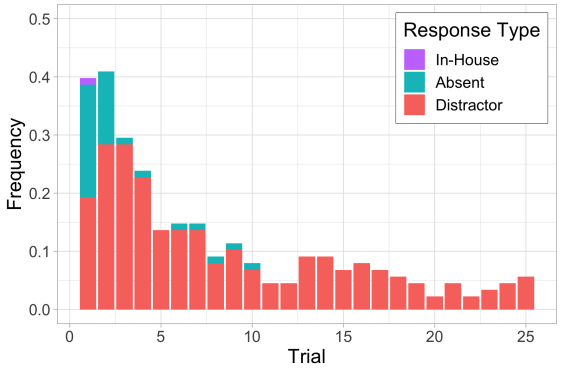


Figure 2: Mean accuracy and response times. Darker points indicate means and error bars indicate 95% highest density intervals for sets of 8 trials. Lighter points indicate means at individual trials. Values for response times computed in log-space. Only trials with correct responses included for response time plots.

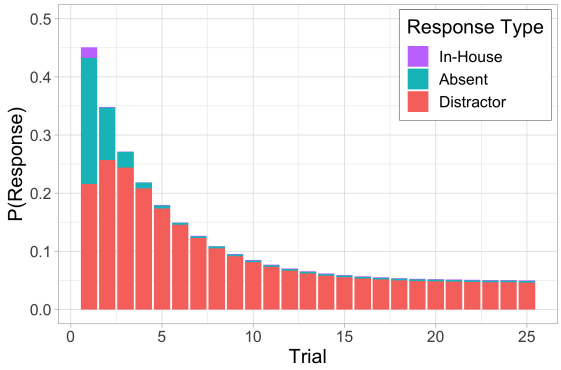
Table 1: Test phase regression coefficient estimates and 95% highest density interval (HDI) lower (L) and upper (U) bounds for effects of a change in digit set (DS), house type (HT) and goal cell position (GP) on dependent variables. Accuracy (Acc.) coefficients are presented in logits and response time (RT) measure in coefficients are presented in $\log_2$(seconds).

		Trials 1-16			Trials 17-64		
Term	DV	Estimate	HDI-L	HDI-U	Estimate	HDI-L	HDI-U
DS	Acc.	-0.098	-0.472	0.268	0.020	-0.248	0.268
DS	RT	-0.022	-0.086	0.043	0.000	-0.032	0.032
HT	Acc.	-0.302	-0.666	0.059	0.095	-0.170	0.364
HT	RT	0.319	0.258	0.379	0.055	0.022	0.088
GP	Acc.	0.165	-0.240	0.570	-0.314	-0.668	0.019
GP	RT	0.115	0.040	0.189	0.089	0.050	0.128

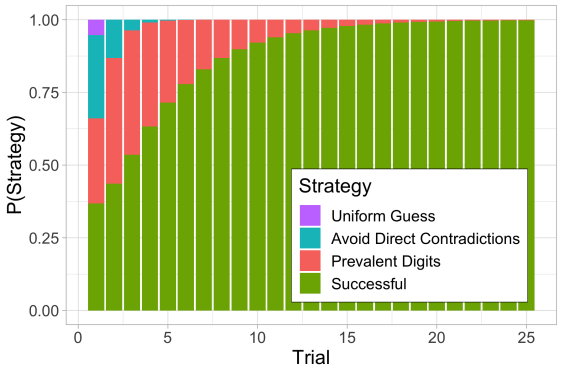
(a) Actual incorrect responses



(b) Inferred probabilities of incorrect responses



(c) Inferred distributions of strategies



(d) Probability of successful strategy

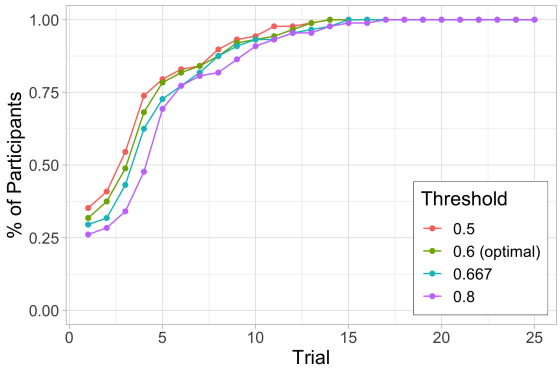


Figure 3: (a) Actual frequency of incorrect response types (b) Inferred probabilities of incorrect response types by the aggregate model (c) Inferred distributions of strategies by the aggregate model (d) Percent of participants inferred by individually fitted models to be using the successful strategy at each trial using different decision threshold values.

digit for the goal cell, 2. *distractor*: the incorrect digit which also occurs 3 times in the puzzle, 3. *absent*: any digit that does not appear in the puzzle at all, and 4. *in-house*: any digit that already appears in the blue-highlighted house. In each puzzle, there are 1, 1, 4, and 3 digits in each category respectively. In the example in Figure 1a, these digits are $\{3\}, \{1\}, \{6, 7, 8, 9\}, \{2, 4, 5\}$.

We also define 4 classes of strategies that participants may have used to generate their responses in each puzzle: 1. *uninformed guess* (UG): responses are completely unconstrained by the hints in the puzzle, selecting from any of the 9 digits. 2. *avoid direct contradictions* (ADC): responses only avoid in-house digits, selecting from any of the remaining 6 digits. 3. *prevalent digits* (PD): distractor and

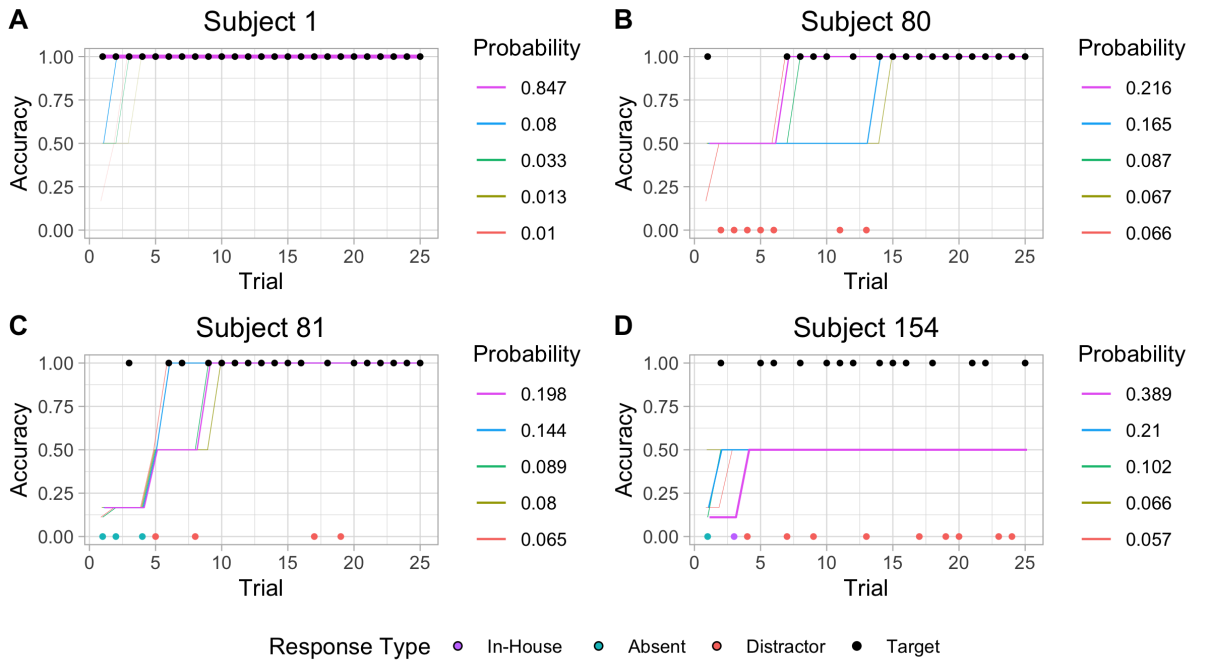


Figure 4: Responses of selected solvers (panels a-c) and one non-solver (panel d) during the practice phase and their top 5 Viterbi paths from individually fitted HMMs according to their posterior probabliities. Target (correct) responses are placed at accuracy of 1 whereas incorrect responses are placed at 0. Viterbi path lines are placed horizontally at positions corresponding to the probability that responses based on the strategy will be correct, e.g. successful at 1 and prevalent digits at 0.5, ignoring strategy execution errors. Line thicknesses indicate path posterior probabilities. The fit to the non-solver used an HMM based on aggregate response profiles of non-solvers. See SI Section 4.1.

target digits are selected with equal probabilities. 4. *successful* (S): the target digit is consistently selected. Ignoring execution errors, the four strategy classes are expected to produce correct responses with probabilities 11.1%, 16.7%, 50%, and 100% respectively. Note that there are multiple specific strategies that can produce responses consistent with each of the listed strategy classes, e.g. always choosing the larger between the distractor and target digits for the PD class (see Section 3.3.1 and SI Section 1.4 for more details).

We fit the model to the aggregate responses across all solvers (Figure 3a) and inferred the aggregate strategy evolution profile (Figure 3c) from the fitted staring strategy distribution, transition matrix, and response emission matrix (SI Section 4.1). As the figure shows, the inferred initial strategy distribution specifies that 36.8% of the responses were based on a successful strategy on the first practice trial, with 29.3%, 28.6% and 5.3% based on PD, ADC, and UG strategies respectively. Use of the UG and ADC strategies rapidly disappears, and the transition to a successful strategy is about 90% complete by trial 10. The model's corresponding expected response distribution, shown in Figure 3b, captures the main features of the pattern of the participants' actual error responses shown in Figure 3a. The residual distractor responses shown are attributed to strategy execution errors that occur with a probability of 0.046 under successful strategies.

3.2.1 Discrete vs incremental transitions

The pattern of change in strategy use over time could arise from an incremental change process, albeit a fairly rapid one, or from discrete strategy changes that occur at different trials for different participants. To explore these possibilities further, we examined the response profiles of individual participants (see examples in Figure 4) and compared the likelihood of the participants' data under a *discrete transition* hypothesis and several variants of alternative *incremental transition* hypotheses (see SI Section 4.3).

Under the discrete hypothesis, each participant begins the practice phase using a strategy from one of the four strategy classes with the initial strategy distribution representing the proportion of participants using each class. On each trial, the participant

either remains in the same class or switches to a superior strategy class, according to the probabilities in the aggregate strategy transition matrix. Under the incremental transition hypotheses, participants can rely on a weighted superposition of multiple strategies and the transition matrix represents the rate of transfer of weight from inferior to superior strategies across trials. We estimated the likelihoods of each participant's response patterns for each hypothesis and found that the data are more likely under the discrete hypothesis than under any of the variants of the incremental hypothesis, with a Bayes factor of 51.71 favoring the discrete hypothesis over the most successful variant of the incremental hypothesis.

3.2.2 Learning efficiency

It is evident from the aggregate data that most solvers learned successful strategies well before the end of the practice phase (Figure 3c), demonstrating an impressive sample efficiency relative to what is usually seen when training a neural network. Under the assumption that participants' actually made discrete transitions, we can estimate when each participant actually began using the successful strategy by fitting separate discrete transition-based hidden Markov models to each individual participant's data and examining the resulting estimates of the location of individual participants transition boundaries.

Along with the response profiles of four example participants shown in Figure 4, we include colored lines indicating the 5 most likely candidate state transition trajectories for each of these participants. The uncertainty in these trajectories, particularly in the timing of the transition to a successful strategy exhibited for the participants in panels b and c, arises because both target and distractor responses occur with equal probability under PD stategies, and distractor responses occasionally occur as errors under successful strategies. Even with an error-free participant like the one in panel a, we cannot be certain some initial trials were not based on the PD strategy.

Given these uncertainties, it is not possible to know exactly when a particular solver transitioned to a successful strategy. Instead, we classified whether or not the estimate of the probability that the participant was using a successful strategy exceeded a confidence threshold, and selected a value for the threshold to minimize bias in these estimates by fitting the model to simulated data (see SI Section 4.4). As shown in Figure 3d, 31.8% of the participants were inferred to be using the successful strategy from the first trial of the practice phase with the optimal (minimum bias) confidence threshold of 60%. By the same criterion, 48.9% of participants were using a successful strategy by trial 3, 78.4% by trial 5, and 93.2% by trial 10. These estimates do depend on the assumptions of the discrete transition model, but even if some solvers are completing the transition to a successful strategy incrementally, the results appear to be consistent with the summary that the transition is over 90% complete by trial 10 of the practice phase.

3.3 Explicitness of acquired strategies

Following the test phase, participants solved one last puzzle and were then asked a series of questions to assess their solution strategies and their ability to describe them. Questions consisted of both free response and multiple choice questions organized as a branching tree intended to successively delineate each participants' strategy. Of these questions, we focus here on the first, most open ended free-response question because all participants were asked to respond to it regardless of responses to any other part of the questionnaire and their responses to it could not have been influenced by later, more specific questions. We also present results on participants' self-reports of attained education. The complete responses of all participants to all questions they were asked are available in data tables available online. In SI Section 5.2, we present results from all of the multiple choice questions.

For the following analyses, we sought to ensure that the verbal reports we considered were obtained from participants whose behavioral profiles were consistent with either successful or PD guessing strategies. Therefore, we screened out solvers who failed to maintain a high level of accuracy throughout the test phase and non-solvers whose pattern of responses were suggestive of strategies other than PD guessing. This left 84 participants in the group we call *persistent-solvers* and, coincidentally, exactly 84 participants in the group we call *PD-guessers*. All 168 participants responded with the target or distractor digits to the questionnaire puzzle.

3.3.1 Responses to the first free-response question

We gave the following prompt for the first free-response question: "Explain as clearly as possible the steps you went through to choose your answer. Please be as detailed as possible so that someone else could replicate your strategy by following your response." A set of 9 categories of possible bases for choosing a response digit were developed and refined by the authors, and a second rater unfamiliar with the details of the study was recruited. 20 participants' responses were used to refine and calibrate the ratings scheme. Disagreements which appeared to reflect the ambiguity of some of the participants' responses were allowed to remain unresolved. Finally, one of the authors and the second rater independently classified the responses of the 148 remaining participants, and the full set of both raters' 168 ratings were then used as the basis for the final assignment of each participant's response (see *Methods) and SI Section 5.3.3*.

The raters placed each participant's response into one of 9 categories. Categories V1, V2, and V3, corresponded to *valid* bases for responding that would yield the correct answer to the participant's given puzzle, based on the rules of Sudoku and the specific constraints employed in constructing the puzzles used in the experiment. Categories U1 and U2 were used for *uncertain* responses. U1 was used for responses that could have been valid, but were described too vaguely to be certain or to assign the strategy to any of the valid responses. U2 was used for responses from which the participants' procedures could not be discerned, either due to unclear or missing descriptions. Categories I1, I2, and I3 covered *invalid* bases that would not reliably give the correct answer. Category M was used for *missing* or otherwise completely uninformative responses. A 10th 'other' option was provided but was never used by either rater. Although all 168 responses were rated, we focus only on the responses of participants who correctly solved the final puzzle to avoid the possibility that differences between the ratings of responses by persistent solvers and PD guessers could be attributed to the rater's awareness of the correctness of the solution. Thus, the results reported in Figure 5 are based on the responses of the 80 persistent-solvers and 42 PD-guessers who solved the questionnaire puzzle correctly. Each bar represents the proportion of persistent solvers' or PD guessers' ratings (treating each rater's rating of each of the participants as a separate rating). The concordance of the two rater's ratings (number of agreements divided by number of participants rated) was 69.6% overall. For solvers, we further asked whether the raters agreed on the assignment of the strategy at the superordinate level (Valid, Uncertain, Invalid, or Missing), and the agreement on this level was 91.3%. Disagreements generally involved the unclear rating from one of the raters (See SI Section 5.3.4).

The key result that emerged from this analysis was the finding that 79% of persistent solver's ratings fell into one of the valid solution types while only 12% of PD guesser's strategies fell into any of these categories (χ^2 = 99.12, df = 1, p < .001). Figure 5a displays the distributions of ratings across all 9 ratings categories separately for persistent solvers and PD guessers. These differences are also statistically significant (χ^2 = 125.86, df = 8, p < .001).

3.4 Role of education

We next considered the relationship between education and participants' ability to solve the puzzles. First, using self-report data on highest level of education pursued, we fitted a regression model to predict the total number of puzzles solved across both practice and test phases, finding that the number of years of education was a small but significant predictor of puzzles solved (β = 1.46, 95% HDI = [0.45, 2.45], R^2 = 0.032). Next, we fitted a second regression using self-report data of various math courses taken and found that of the 9 different topics, only high-school algebra (β = 9.68, 95% HDI = [2.87, 16.78]) and high-school geometry (β = 9.81, 95% HDI = [3.78, 16.03]) were significant independent predictors while none of the others were significant independently. *None* of the 40 participants who reported having taken neither HS algebra nor HS geometry were solvers. In further analyses combining years of education and math courses (See SI Section 2.5), we found that education predicted a small amount of independent variance when combined with HS algebra and geometry (R^2 = 0.168 with education, R^2 = 0.152 without, Bayes factor = 11.517), but its significance is lost when considered together with all of the math course variables (R^2 = 0.213 with education, R^2 = 0.206 without, Bayes factor = 2.624).

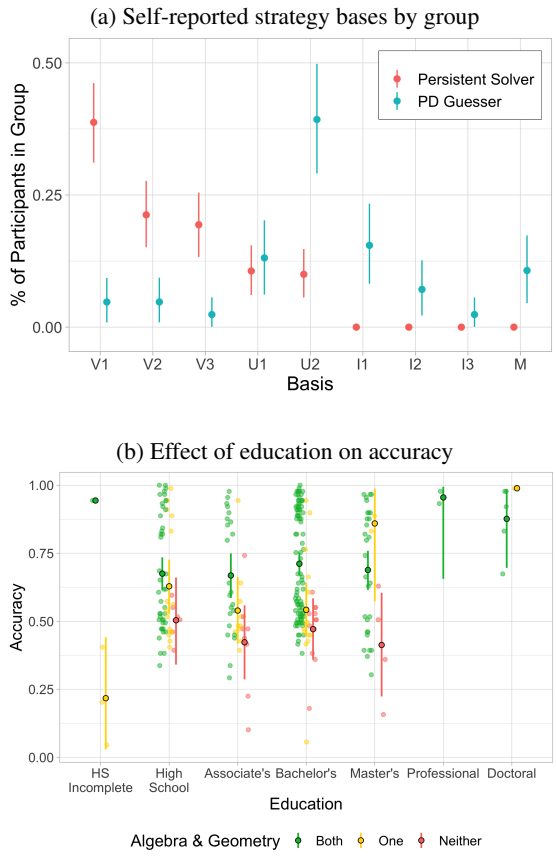


Figure 5: (a) Bases for choosing between prevalent digits by group. (b) Overall accuracy across both the practice and test phases of the experiment by highest education. Color shows whether participant has had formal education in both, one, or neither of algebra and geometry. Darker, bordered points represent group means and lighter points represent individual participants. All error bars show 95% highest density intervals (groups with N < 2 excluded).

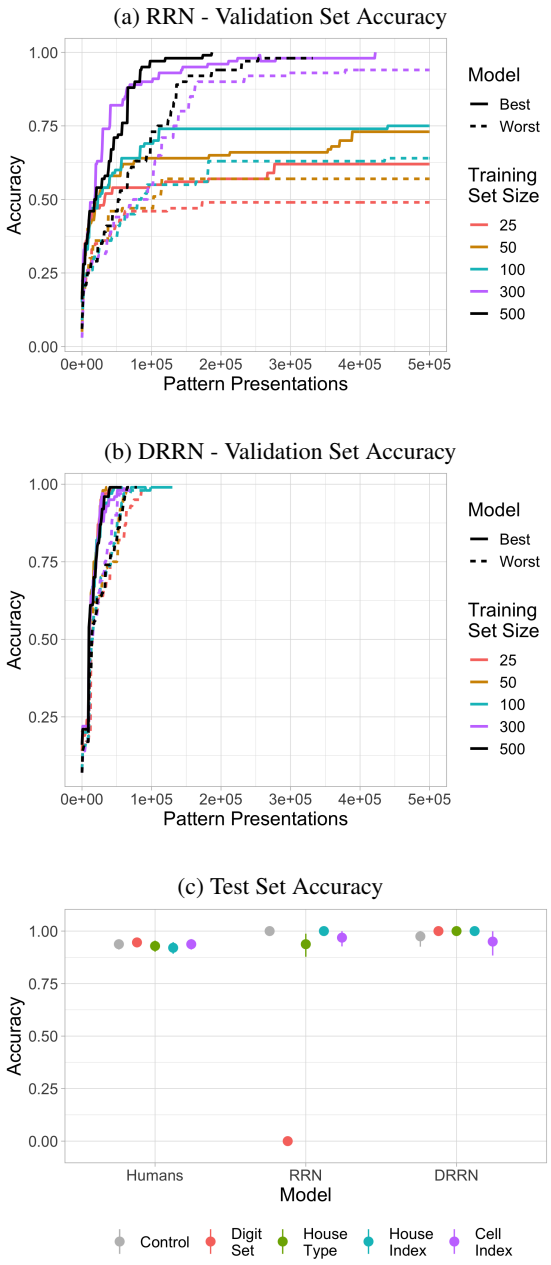


Figure 6: (a-b) Recurrent Relational Network's (RRN) and Digit-invariant RRN's (DRRN) validation accuracies on held-out puzzles with the same features as training samples. The x-axis represents total number of training puzzles presented to the model and the y-axis represents cumulative maximum accuracy during training. 10 models were trained for each training set size. *Best* and *Worst* lines indicate the highest and lowest cumulative maximum accuracy among the 10 model instances respectively. (c) Test set accuracy by feature for human solvers, the RRN, and the DRRN. Error bars indicate 95% highest density intervals.

4 Model Results

To compare how contemporary neural network models learn and generalize solving the Hidden Single puzzles, we replicated and adapted a model that is state-of-the-art in solving Sudoku puzzles [30] (see *Methods* and SI Section 6). The Recurrent Relational Network (RRN) uses a relational message passing scheme where in each time step, it computes for each cell in the grid an update instruction for each other cell that the cell shares a house with. As we shall see, the model achieves systematicity with respect to the positional variables by sharing the same connection weights to each cell from all of the relevant constraining cells, thereby remaining invariant to the particular cell it solves for and the particular relevant cell constraining it (see SI Section 6 for details).

Using the same set of restrictions on the practice puzzles as described in our human experiments, we trained the model with varying numbers of training samples, stopping if and when it reached 99% ac-

curacy for puzzles with the same features. We then tested the trained model with the same systematic variations we considered for human participants. Comparing the model's training and generalization results to human solvers, we observed two key properties of the network.

First, given the same restrictions on the particulars of the training puzzles as described in our human experiments, the model generalized immediately, performing nearly perfectly on problems with changes to the positional features (house type, house index, and cell index) but could not solve a single puzzle with changed digit sets as shown in Figure 6c. Second, the model trained inefficiently compared to the human participants, requiring 300 unique puzzles with tens of thousands of total pattern presentations to gradually reach accuracies comparable to the solvers (Figure 6a).

We then modified the architecture described in [30] to also induce invariance to digits by extending the weight-sharing scheme across the 9 digits (see *Methods* and SI Section 6.4). This version of the model required only 25 training examples to reach accuracies comparable to solvers, but still required 40,000 total pattern presentations to train (Figure 6b). As expected, the Digit-invariant Recurrent Relational Network (DRRN) achieved the same very high level of systematic generalization for puzzles with changed digit sets as it did in the other three feature variations.

5 Discussion

We have presented evidence that participants who learned a novel problem solving strategy from a brief tutorial and practice session can systematically extend what they have learned to problems varying along several different dimensions outside the range of the tutorial and practice problems. Accuracy on examples outside the training range on the digit set, house type, or target cell position used in training was over 85% correct at the start of the test phase.

For most of the participants classified as solvers, learning was nearly complete within 10 trials, and one-third of solvers may have used a successful strategy as early as the first practice trial after the tutorial. For other solvers, the findings are consistent with the hypothesis that they made very rapid, perhaps even discrete-step transitions as they acquired more successful strategies during the practice phase. Even if transitions may not have been completely discrete for some solvers, they were clearly quite rapid. Thus, with explicit instruction and explanatory feedback during practice, many participants acquired a new problem solving skill quickly and in a way that they could extend to new examples varying from training examples in several different ways. We also found that most solvers were able to describe a valid solution strategy when prompted.

While some participants learned a successful strategy very quickly, the ability to do so is far from universal. Only about a third of our participants met the criteria for classification as solvers by the end of the practice phase of the experiment. We note that the criterion used to identify solvers was quite a conservative one, and an examination of performance across both the practice and test phases reveals an additional 22 participants whose test phase accuracy exceeded 75% correct (see SI Section 7.1). Treating these participants as solvers would still leave half the participants failing to perform at a level significantly above the 50% PD-guessing level. Importantly, a rudimentary formal exposure to mathematics appeared to be crucial for success: none of the participants lacking both high-school algebra and geometry backgrounds met our criteria for classification as solvers. Such exposure was not sufficient for success, however, as only 44% of participants who reported having taken both classes were solvers.

It is likely that a participant's success rate in acquiring successful solution strategies depends on the particular features of the tutorial. On the one hand, since we were interested in induction of generalizable knowledge rather than simply learning to follow an explicit rule, we did not directly provide a description of the Hidden Single technique in the form of a general rule, as online Sudoku tutorials often do, and such direct instruction using an explicit generalizable rule could increase the success rate further. On the other hand, we did not rely solely on the standard reinforcement-based or supervised learning approach employed in most neural network modeling work, because we were also interested in the ability to learn from instructions and explanatory feedback. Although further research should confirm this, we expect that most participants would learn the Hidden Single strategy far more slowly, if at all, under these learning conditions, consistent with other evidence of a role for explanations and instructions in facilitating rapid

learning [2, 38].

5.1 Implications for understanding the basis of human systematic reasoning abilities

As discussed in the *Introduction*, there is an important tradition that proposes that humans rely on built-in systems that support systematic reasoning and generalization [13] and allow for rapid, and researchers continue to suggest that built-in commitment to compositional representations may underlie transferable learning based on one or a few examples [22]. We have found in this investigation, however, that the ability to acquire one systematic reasoning strategy – the Hidden Single strategy – from a brief explanatory tutorial and to generalize it systematically is restricted to a subset of adult participants, all of whom had at least some prior exposure to high-school level mathematics. This observation is consistent with the idea that the systematic reasoning ability exhibited by solvers is one that is acquired through learning the systematic reasoning skills taught in mathematics classes. Based on this, we suggest that we should seek to understand systematic reasoning and generalization as an acquired ability that depends on relevant experience, not as an inevitable built-in component available as a core element of human cognitive abilities. While we have only considered a single domain in this study, and our findings may not apply across all domains in which humans exhibit the ability to learn quickly and to generalize, we suggest that it would be worthwhile for researchers studying human learning and reasoning in a wide range of domains to consider the possible role of relevant prior experience very carefully.

Our investigations also demonstrate a strong association between the ability to acquire and use the Hidden Single strategy correctly and the ability to describe a valid solution strategy. Although solvers did not always provide explanations that were clear and detailed enough to fully identify a valid strategy, we note that not a single one of the persistent solvers' explanations was rated by either rater as unambiguously invalid. Thus, it is possible that persistent solvers always possessed at least vaguely cohesive explanations, even if they did not articulate them fully or clearly in all cases [9, 2]. Indeed, the facts that mathematics is often taught through explanatory tutorials and subsequent practice; that encouraging students to explain enhances mathematics learning outcomes [5]; and that our tutorial and feedback during practice were explanation-based all point toward the possibility that learning in settings in which procedures are taught through language and in which both learner and teacher as expected to produce and understand explanations may play a role in supporting rapid acquisition of systematically generalizable reasoning and problem-solving skills.

In summary, we take our findings to suggest that future research should seek to understand how humans can *acquire* the ability to reason systematically from instruction and explanation in combination with problem-solving practice. In this regard, our work dovetails with decades-old research pointing out the potent role of education in establishing the ability to learn from instruction and explanation versus mere observation and practice [35].

5.2 Sources of domain-specific constraints

It has long been argued that the ability to learn quickly from limited information depends crucially on having the right inductive bias at the outset of a new learning experience, and a long-standing perspective holds that domain-specific inductive biases are often available from birth, in the form of such things as core systems for number [11], a human language acquisition device with a built-in universal grammar [6], or initial core-knowledge systems for intuitive physics and psychology [22]. Neural network models such as convolutional neural networks exploit such inductive biases, as well. The Recurrent Relational Network applied to Sudoku by [30] exemplifies this approach. The connection weights to each constrained cell from each constraining neighbor cell were completely shared, allowing it to generalize perfectly across all of the spatial variables we considered, including the house type variable, and we were able to extend this approach, allowing for for systematic generalization across digits as well.

However, building in the specific connectivity required to capture the specific constraints of a particular problem can specialize a model too much. The Sudoku RRN of [30], could not, for example, learn to exploit constraints of a puzzle with similar rules that depend on relations between all pairs of diagonally adjacent squares. Similarly, building in equivalent treatment of all digits as we did would prevent it from being able to solve other games, such as Ken-Ken, which depend critically on the arithmetic combining possibilities of particular dig-

its. This is, of course, an issue for other modeling approaches as well. While we would not deny the utility of building in domain specific constraints on learning and generalization in basic sensory-motor and survival-related domains, the need that humans must be able to master new, human invented domains points toward the utility of seeking solutions that avoid strong dependence on strong, built-in, domain specific biases.

A more recent approach to achieving rapid learning and transfer in neural networks relies on learning to learn, or meta-learning, by training a network with a series of similar tasks with shared features [17], and a reliance on meta-learning is among the approaches endorsed by Lake et al. [22]. However, it is typical for meta-learning to be strongly targeted to a single task domain, so that a dedicated neural network is applied to acquiring the inductive biases relevant to a single specific task domain. What we believe is needed instead is a single, integrated learning system that is applied simultaneously to learn the full range of tasks a human learner might be exposed to. Below we turn to a consideration of some features of what such a system might look like.

5.3 The path forward for neural network models of human systematic generalization and transfer

Based on our analyses of human success and failure in learning and extending a problem-solving skill, we propose that future neural network models should (1) exploit a task-agnostic architecture, (2) combine multiple modalities to allow instruction and explanation to be integrated with task-based learning and (c) seek to promote systematic generalization through learning multiple tasks in which individual numeric values are used as instances of a broader class.

First, if abstract reasoning is indeed a learned ability rather than innate in humans, models should contain task-agnostic general architectures rather than features optimized for any specific task. One architectural feature that such a system could rely upon is sequential attention [40, 14, 31]. Rather than applying the same connection-based knowledge simultaneously to all cells in a Sudoku grid as the RRN Sudoku network does, we propose that systematicity can emerge if the network is allowed to approach problem solving through sequential steps, selectively attending to different parts of a task or problem, e.g. a subset of cells in Sudoku, using a single set of connection weights that are shared by applying them sequentially to the set of cells in the current focus of attention. While we are firm believers in the idea that humans can engage in a parallel, mutual constraint satisfaction process, and that such a process characterizes aspects of perception, language processing, and many other cognitive systems [25, 24], the extent of this parallelism is likely limited in ways that artificial neural networks need not be, such as the number of cells and digits that can be simultaneously considered. When humans solve Sudoku puzzles, it seems more likely that connections are reused across time using attention to facilitate routing relevant inputs through the appropriate pathways [28, 29], albeit with some degree of parallel processing within each step. Thus, we favor a hybrid approach of parallel and sequential processes where attentional mechanisms bind relevant variables to generalizable computations as appropriate.

Second, noting the importance of learning through instructions and explanations, we propose that neural networks should encompass additional modalities beyond those strictly relevant to task performance in a particular learning environment. While visual and language inputs and outputs are common in contemporary models, these modalities are typically included only when they are relevant to the task itself, e.g. visual-question answering tasks, and not as channels for conveying instructions and explanations. The RRN, for instance, only possesses inputs and outputs for the 9x9 grids as these are the only interfaces necessary to solve the puzzles. Humans, however, can learn to play using just the description of the game and can learn more advanced methods by consulting an instructor or even a reference guide of known Sudoku techniques, accelerating their training far beyond what could be achieved through pure trial-and-error loss signals. Thus, to build models that learn as humans do, we propose that model modalities be expanded not only as relevant to executing the task but to offer additional teaching signals during training. Similarly, we believe that training models to provide valid explanations for their decisions, similar to how students are encouraged to show their work in math classes in addition to the final answer, would further enhance learning and abstract generalization. Indeed, learning through instructions can allow the model to rapidly acquire task-relevant

inductive biases, thus maintaining flexibility in its ability to generalize to new tasks and variations, but doing so without slowly training over additional large datasets.

Finally, if systematic abstract reasoning is to be learned, then any single task should be thought of as one among a larger set where variable and filler relations are reinforced. Adults with relevant educational backgrounds can enter novel reasoning tasks with strong inductive biases to help them learn rapidly and infer which features can be generalized and which cannot, and we propose that meta-learning to abstract may help induce similar priors. On this view, the reason why a change in digits had no visible effect is because algebraic abstraction of digits is explicitly taught in math education and frequently applied in procedures that can be applied to arbitrary variables, such as solving an equation with one unknown, are required. Transposing grids, however, is less likely to be an operation that people would need to perform often, consistent with the fact that solvers did not generalize perfectly under this transformation. Thus, in conjunction with the first two points, we consider mechanisms such as systematicity through sequential attention and learning through explanations to be the broader meta-principles that the model could learn.

One model that begins to address these goals is GPT-3 [3]. After extensive training to predict words in text based on a long window of prior text, it has demonstrated some capacity to bind variables and execute simple instructions, relying on a relatively generic transformer architecture [40]. However, [34] found that a sequence-to-sequence transformer trained to solve 41 different modules of mathematics problems struggled to learn the more complex modules. While there may be many reasons for the model's limitations, a direct sequence-to-sequence transformation may impede the model's capacity to perform long multiplication compared to a more suitable representation in columnar format, and may thus benefit from a more interactive environment that allows restructuring the task. To us, a promising path forward would be to extend generic architectures such as the transformer to process and produce actions in a visuospatial environment while using language to receive and generate instructions and explanations as partially implemented in [1]. This model could learn not only naturalistically, but also from a structured curriculum of mathematical problem-solving tasks requiring following instructions and receiving and producing explanations. We hope that this might bring us considerably closer than we are now toward a model that captures the human ability to learn new problem solving skills that exhibit out of domain transfer after learning through instructions and explanations from a small number of examples.

5.4 Conclusion

Our findings contribute both to understanding how humans learn and generalize abstract strategies and to identifying where humans and machine learning algorithms diverge in these respects. While machine learning can provide tools for inducing systematicity that can be superior to humans by certain measures, we also emphasize the importance of considering potential trade-offs and limitations. With the characteristics of human systematic reasoning in mind, we hope to push models of intelligence towards a direction that draws inspiration from the mechanisms that enable humans to learn efficiently and generalize robustly. Many of the attributes of human reasoning that we have observed remain incompletely understood, and the details of how to implement them in models are exciting further directions for both human and machine intelligence that we look forward to addressing in future research.

6 Methods

6.1 Participants

All experiments were conducted through Amazon Mechanical Turk. In our pre-registration, we committed to collecting samples until we had at least 75 solvers, but because we ran the experiment in batches (with the expectation that only a small random fraction of participants would be solvers), we ended up collecting a slightly higher sample size than we had originally intended. Overall, 1985 people entered the study, receiving a small base payment for doing so. 1714 were filtered out through a diagnostic puzzle and questions about their prior Sudoku experience, leaving 271 participants that completed the experiment (participants were offered a bonus for solving the diagnostic puzzle to minimize incorrect attempts; see SI Section 1.1 for full details). Those not filtered out continued on to the tutorial and remaining phases of the experiment. Participants received a bonus for completing each section of the tutorial and for correctly solving

puzzles on the first attempt during the practice and test phases. Participants were not notified of the purpose of the experiment.

6.2 Participant subselection

The following criteria were used to classify focused subsets of participants for different analyses. For exact regression formulas and coefficients, see SI Section 2.1.

Solver vs non-solver: We fitted a logistic mixed effects model to the practice phase data to predict the correctness of the participants' responses at each trial. Using the predicted accuracy at the 25th trial, we used a decision boundary of 0.8 to classify solvers and non-solvers. This method of distinguishing solvers from non-solvers was preregistered.

Persistent-solver vs PD-guesser: We fitted a logistic mixed effects model to the test phase data to predict the correctness of the participants' responses at each trial. Using the predicted accuracy at the 64th trial, we included solvers with scores of at least 0.8 for persistent-solvers and non-solvers with scores of at most 0.6 for PD-guessers. For PD-guessers, we also required that at least 58 out of 64 test phase responses were either the distractor or target digits and that they solved exactly 3, 4, or 5 puzzles in the final 8 trials of the test phase.

The non-solvers who were screened out include some participants who performed well below the 50% correct level expected from the PD strategy, some others who may have been late and/or inconsistent solvers, and yet others who may have been PD-guessers whose pattern of responding deviated from the PD-guesser profile by chance, just as a set of tosses of a fair coin will produce a result outside the 95% confidence interval for the observed probability of heads 5% of the time. Taken together our criteria were stringent enough that they may well have screened out quite a few true PD-guessers.

6.3 Behavioral experiment

The experiment program was implemented using Facebook's React, hosted on Amazon AWS, and deployed using Psiturk [10, 15]. For brevity, we only describe here the practice phase, test phase, and the general flow of the questionnaire phase. For more specific details on the diagnostic measures, tutorial sequence, and the questionnaire, see SI Section 1.

With the exception of some puzzles used in the tutorial, all puzzles were generated using the same procedure. To make puzzles nontrivial and encourage deductive reasoning, every puzzle was generated with at least 1 box constraint (a hint that shares a box with 3 of the empty cells in the house, simultaneously constraining all 3) and 2 orthogonal constraints (hints that share a column or row with an empty cell in the target house for puzzles with row and column house types respectively). 3 cells in the highlighted house were filled with random remaining digits. Because this would require at least 3 hints that share digits with the target, we added a distractor digit with 3 hints that constrain the same box and one of the 2 other unconstrained cells, making both the target and distractor digits salient as potential candidate target digits. Puzzles could be interacted with by selecting a cell with the computer cursor and typing a digit. Only clicking the 'Submit' button would commit the response, thus allowing participants to change their responses if desired before submitting.

All 25 puzzles in the practice phase shared the same house type, house index, cell index, and digit set as the tutorial. The 64 puzzles in the test phase were counterbalanced using randomized balanced Latin squares such that in each of the 8 sets of 8 trials, each of the 8 combinations of unchanged vs. changed house type, house index, and cell index conditions would appear once as the i^{th} puzzle, and that for each combination *A* and *B*, *A* would appear before and after *B* exactly 4 times. 4 trials in each set of 8 trials were also assigned the each of the unchanged vs. changed digit set conditions such that in every 16 trials, all 16 combinations of changed or unchanged house type, house index, cell index, and digit set would appear exactly once. In trials with house type changed and either the house or cell index changed, the house type was changed first and the house or cell index changed with respect to the new house type. For instance, if HT and HI conditions were applied to a row puzzle with the goal cell at (3, 5), one resulting puzzle could be a column puzzle with the goal cell at (3, 1).

Participants were allowed unlimited time and number of attempts to solve puzzles during the practice phase, and were additionally provided detailed explanations customized to their puzzles and error types following each error (see SI Section 1.3). Correct responses following errors also triggered explanations. During the test phase, however, participants were allowed up to 2 minutes and only

a single attempt at solving each puzzle, and no feedback was provided except whether they were correct or incorrect. Upon submitting a response, participants had 10 seconds to continue viewing the puzzle before the program would automatically proceed to the next puzzle. Correct responses allowed the participants to skip ahead to the next puzzle while incorrect responses paused the screen for the full 10 seconds.

At the beginning of the questionnaire, participants were asked 3 attention check questions. They were then given a new puzzle to solve, randomly generated for each participant, with the constraints that the puzzle shared the same house type and digit set as the tutorial and practice phase puzzles, but the house index and cell index were set such that they differed from the tutorial while ensuring that the goal cell fell within the center 3x3 box. Although no feedback was provided for this puzzle, the puzzle (with the participant's response) always remained on display for the remainder of the questionnaire, allowing the participant to refer to it as necessary.

Following the puzzle, participants were asked for their confidence that their responses were correct as a percentage between 0 and 100 in increments of 5% and then answered a first free response question in which they were asked to explain as clearly as possible the steps they went through to obtain their answer to the puzzle so that someone reading their response could reproduce the sequence of steps. Each participant who had chosen either the target or the distractor then entered a branching questionnaire, following a path that depended on their answers to forced choice questions that were interleaved with free response questions they could reach depending on their answers to the forced choice questions. At the end of the questionnaire, all participants then received one final free response question asking for further comments or clarification of previous answers. For full details, see SI Section 1.4.

6.4 Regressions

The regressions to predict accuracy on the last trial of the practice and test phases were fitted using logistic mixed-effects models as pre-registered. However, we changed the regression method to Bayesian for consistency with the remainder of the paper. Classifications were consistent between using Bayesian and non-Bayesian predictors. All regressions were fitted using the BRMS package in R [4]. See SI Sections 2 and 3 for exact regression formulas and tables of coefficients.

All reported coefficients on accuracy models are in logits. All reported coefficients on response time models are in $\log_2$(seconds). Only trials with correct responses were used to fit the response time models. In each model, we accounted for improvements through practice using a $\log_2$(trial) term and for individual variations through random effect intercepts for each participant. For exact formulas and the full table of statistical measures including models using all 64 trials, see SI Sections 2 and 3.

6.5 Highest density intervals

The 95% HDIs of binary variables such as correctness in Figure 2 and questionnaire responses in Figure 5 were estimated by sampling 100,000 times from resulting Beta distributions. The 95% HDIs of multinomial variables in Figure 5a were estimated by sampling 100,000 times from resulting Dirichlet distributions. The 95% HDIs of continuous variables such as response times in Figure 2 and accuracies in Figure 5b were estimated using Bayesian regressions.

6.6 Hidden Markov model

The aggregate model for finding priors was written in PyTorch and was optimized by minimizing the cross-entropy loss with respect to participants' actual responses. We used Adam [19] to perform gradient descent with learning rate = 0.01 over 2000 epochs. Using the same solver and non-solver classification as the test phase, we fitted a separate aggregate model for each group. The initial state distribution vector π, transition matrix A and likelihood matrix B optimized through the aggregate model were then used to fit the individualized HMM for each participant. We allowed for the possibility of errors in executing strategies in each class, defined as producing a response that should not be produced under the strategy class, such as an in-house response when using a prevalent digits strategy. To account for these occasional strategy-inconsistent responses, we added additional error terms in the emission matrix. We also assumed that participants would always transition toward better strategies, thus constraining the model such that participants could not transition from a better strategy to a worse one. See SI Section 4.1 for exact formulas and fitted values.

6.7 Discrete vs Incremental Transitions

To calculate the Bayes factor between the discrete and incremental hypotheses, we estimated the likelihood of each participant's response profile by simulating the distribution of response profiles under each hypothesis. We simulated individual response trajectories across the practice phase under the discrete and incremental transition hypotheses by starting each trajectory with a random sample of the four strategies according to the fitted starting strategy distribution. For the remaining 24 trials, we repeatedly applied the transition matrix to get the subsequent strategy distributions.

For discrete transition samples, we randomly sampled from these distributions to determine a discrete strategy for each trial, and conditioned the response choice on that trial and the strategy used on the next trial on the sampled strategy. For incremental transition samples, we iteratively multiplied the strategy weight vector with the transition matrix to get the subsequent trial's strategy weight vector: $\pi_{t+1} = \pi_t \mathbf{A}$ where π_t is the strategy weight vector at trial t and $\mathbf{A}$ is the transition matrix. Thus, the strategy weight vectors were left as weighted combinations for all but the first trial. To sample responses, we marginalized the emission probabilities across all four strategies, i.e. $\pi_t \mathbf{B}$ where $\mathbf{B}$ is the emission matrix, and sampled from the resulting emission distribution.

Because multiplying the same transition matrices would produce the same 4 strategy trajectories (one for each starting strategy), we introduced additional variability in the simulated trajectories by sampling from 4 Dirichlet distributions. To estimate the Dirichlet parameters, we calculated the expected number of times each strategy transition was observed according to the fitted starting strategy distribution and transition matrix over all 88 participants and 25 trials:

$$88 * \sum_{t=1}^{25} \pi_1 \mathbf{A}^{*t-1}$$

where $\mathbf{A}^*$ is the fitted transition matrix. We applied a similar sampling procedure for the emission probability matrix using the expected number of times each response was observed under each strategy:

$$88 * \sum_{t=1}^{25} \pi_1 \mathbf{A}^{*t-1} \mathbf{B}^*$$

where $\mathbf{B}^*$ is the fitted emission matrix.

Using N = 100,000 sampled strategy trajectories and their resulting likelihood values $P(\text{response}_t|\text{strategy}_{h,i,t})$ where $\text{strategy}_{h,i,t}$ is the strategy distribution under hypothesis h for sample i at trial t, we calculated the marginalized likelihood of each subject's response sequence S_s under the hypothesis according to the formula:

$$P(S_s|\text{hypothesis}) = \frac{1}{N} \sum_{i=1}^{N} \prod_{t=1}^{25} P(\text{response}_{s,t}|\text{strategy}_{h,i,t})$$

The same set of sampled starting strategies, transition matrices, and emission probability matrices were used for sampling response profiles under both the discrete and incremental transition hypotheses.

6.8 Questionnaire ratings

One of the authors (JLM) considered all participants' strategy descriptions given to the first free response question and developed a draft set of rating categories. Together the authors developed the persistent solver and PD guesser screening criteria, leaving 168 strategy description responses to rate. Both authors then rated the same set of 20 of these responses, then met to refine the categories and create rating instructions. Author AJN served as one of the two raters. A Master's student in Computer Science collaborating with JLM but otherwise uninvolved in the project was recruited for pay to serve as a second rater. This rater went through the experiment as a participant would, read the instructions prepared by the authors, and rated the 20 examples mentioned above. Final adjustments to the ratings categories were made in discussion among the two raters and JLM. Author AJN and the second rater then finalized their ratings of the 20 example participants without requiring full agreement and then proceeded to rate the responses of the remaining 148 participants. For each participant, the puzzle and the participant's digit response were shown to the rater and, as an attention check, the rater first judged whether or not the participant had correctly solved the puzzle. The actual correctness and the participant's strategy description were then revealed. The raters then judged whether both, one, or neither of the two prevalent digits were mentioned in the strategy description. Next, if the participant did not correctly solve the puzzle, the rater identified whether or not the participant demonstrated awareness of the error. Lastly, the

rater selected among 10 options for describing the participant's basis for choosing between the prevalent digits as inferred from the written response. A second option was allowed to be selected if the rater judged that there was a close second choice. For the full set of rating categories and additional procedural details, see SI Section 5.3.2.

6.9 Recurrent relational network

We implemented the RRN model with some simplifications (e.g. fewer hidden layers, smaller embedding sizes, etc.) due to computational resource constraints. Although our version had a lower success rate producing complete solutions compared to the original, it exhibited over 97% correct performance in completing individual cells in held-out test Sudoku problems (see SI Section 6).

We then adapted the model to the Hidden Single puzzles by modifying the loss function to only compare the cells relevant to the task and made additional minor simplifications. We then constructed the Digit invariant RRN (DRRN) by developing a scheme that used a single set of shared connection weights to compute each digit's value in each cell across all cells and digits. Since this created 9^3 separate computational nodes, further simplifications were required. The DRRN was able to acquire the Hidden Singles task from far fewer training examples than the RRN, in addition to demonstrating perfect out of distribution transfer (see SI Section 6.4).

To test for systematic generalization, we used the models that were trained with 500 puzzles until at least 99% validation accuracy. We compared puzzles that had at just one feature varied rather than all the combinations, resulting in 20 test puzzles for each model. The results shown in Figure 6c were averaged across all 10 instances of each model.

7 Pre-registration, Data and Code Availability, and Compliance

The pre-registration can be found at https://osf.io/smf4b/.

All relevant data and code for this study can be found at https://github.com/andrewnam/hidden_singles_public. Instructions on navigating and using the repository can be found in the README.

This study was approved by the Stanford University Institutional Review Board Panel on Non-Medical Human Subjects, Ethics Committee 7029.

References

[1] Josh Abramson, Arun Ahuja, Iain Barr, Arthur Brussee, Federico Carnevale, Mary Cassin, Rachita Chhaparia, Stephen Clark, Bogdan Damoc, Andrew Dudzik, et al. Imitating interactive intelligence. *arXiv preprint arXiv:2012.05672*, 2020.

[2] Woo-kyoung Ahn, William F Brewer, and Raymond J Mooney. Schema acquisition from a single example. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18(2):391, 1992.

[3] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[4] Paul-Christian Bürkner. BRMS: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, 80(1):1–28, 2017.

[5] Michelene T. H. Chi and Ruth Wylie. The ICAP framework: Linking cognitive engagement to active learning outcomes. *Educational Psychologist*, 49(4):219–243, 2014.

[6] N. Chomsky. *Language and Mind*. Mouton, The Hague, 1968.

[7] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3642–3649. IEEE, 2012.

[8] Michael Cole, John Gay, Joseph Glick, and Sharp Donald. The cultural context of learning and thinking: An exploration in experimental anthropology. 1971.

[9] Gerald DeJong and Raymond Mooney. Explanation-based learning: An alternative view. *Machine learning*, 1(2):145–176, 1986.

[10] David Eargle, Todd Gureckis, Alexander S. Rich, John McDonnell, and Jay B. Martin. psiTurk: An open platform for science on Amazon Mechanical Turk, March 2021.

[11] Lisa Feigenson, Stanislas Dehaene, and Elizabeth Spelke. Core systems of number. *Trends in cognitive sciences*, 8(7):307–314, 2004.

[12] Bertram Felgenhauer and Frazer Jarvis. Mathematics of sudoku I. *Mathematical Spectrum*, 39(1):15–22, 2006.

[13] Jerry A Fodor, Zenon W Pylyshyn, et al. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71, 1988.

[14] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016.

[15] Todd M Gureckis, Jay Martin, John McDonnell, Alexander S Rich, Doug Markant, Anna Coenen, David Halpern, Jessica B Hamrick, and Patricia Chan. psiturk: An open-source framework for conducting replicable behavioral experiments online. *Behavior research methods*, 48(3):829–842, 2016.

[16] Felix Hill, Adam Santoro, David GT Barrett, Ari S Morcos, and Timothy Lillicrap. Learning to make analogies by contrasting abstract relational structure. *arXiv preprint arXiv:1902.00120*, 2019.

[17] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439*, 2020.

[18] Philip N Johnson-Laird. Mental models and deduction. *Trends in cognitive sciences*, 5(10):434–442, 2001.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning*, pages 2873–2882. PMLR, 2018.

[21] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[22] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

[23] Gary F Marcus. *The algebraic mind: Integrating connectionism and cognitive science*. MIT press, 2003.

[24] James L. McClelland, Felix Hill, Maja Rudolph, Jason Baldridge, and Hinrich Schütze. Placing language in an integrated understanding system: Next steps toward human-level performance in neural language models. *Proceedings of the National Academy of Sciences*, 117(42):25966–25974, 2020.

[25] James L McClelland and David E Rumelhart. An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychological review*, 88(5):375, 1981.

[26] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[27] Andrew Nam and James L McClelland. A study in extensible algorithmic thinking - hidden singles, Dec 2020.

[28] Allen Newell, John C Shaw, and Herbert A Simon. Report on a general problem solving program. In *IFIP congress*, volume 256, page 64. Pittsburgh, PA, 1959.

[29] Allen Newell and Herbert A Simon. Computer simulation of human thinking. *Science*, 134(3495):2011–2017, 1961.

[30] Rasmus Palm, Ulrich Paquet, and Ole Winther. Recurrent relational networks. In *Advances in Neural Information Processing Systems*, pages 3368–3378, 2018.

[31] Scott Reed and Nando De Freitas. Neural programmer-interpreters. *arXiv preprint arXiv:1511.06279*, 2015.

[32] Timothy T Rogers and James L McClelland. Parallel distributed processing at 25: Further explorations in the microstructure of cognition. *Cognitive science*, 38(6):1024–1077, 2014.

[33] Ed Russell and Frazer Jarvis. Mathematics of sudoku II. *Mathematical Spectrum*, 39(2):54–58, 2006.

[34] David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. In *International Conference on Learning Representations*, 2019.

[35] Sylvia Scribner and Michael Cole. Cognitive consequences of formal and informal education. *Science*, 182(4112):553–559, 1973.

[36] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[37] Andreas Stuhlmuller, Joshua B Tenenbaum, and Noah D Goodman. Learning structured generative concepts. Cognitive Science Society, 2010.

[38] Pedro A Tsividis, Thomas Pouncy, Jacqueline L Xu, Joshua B Tenenbaum, and Samuel J Gershman. Human learning in atari. 2017.

[39] Ivan I Vankov and Jeffrey S Bowers. Training neural networks to encode symbols enables combinatorial generalization. *Philosophical Transactions of the Royal Society B*, 375(1791):20190309, 2020.

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[41] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

[42] Lev S Vygotsky. *Thought and language*. 1934.

[43] Peter C Wason. Reasoning about a rule. *Quarterly journal of experimental psychology*, 20(3):273–281, 1968.