A mathematical theory of balancing relational generalization and memorization

Luke Cheng [1] **Samuel Lippl** [1]

Abstract

Humans, animals, and modern machine learning models exhibit impressive abilities to learn complex behaviors and generalize these behaviors to unseen situations. This ability requires us to learn rules and regularities that allow for such generalizations. At the same time, in most complex environments, any rule will have its exceptions. How do learning systems balance between learning general regularities and memorizing exceptions? We argue that a lack of task paradigms has hindered the study of this essential ability. To address this gap, we introduce a novel task, transitive inference with exceptions, that tests for relational generalization and memorization of an exception to the relational rule. We then analytically characterize the behavior of a simple, theoretically tractable model of neural network learning (kernel ridge regression) across a broad family of representations and task parameters. We find that these models can balance between relational generalization and memorization, but unlike for transitive inference without an exception, successful generalization is sensitive to the specific representational geometry. We explain why this task is more challenging mechanistically by drawing on our analytical theory. Finally, we validate our theoretical insights in pretrained language models that are finetuned on ordered relations, finding that these models successfully generalize according to the transitive rule, but also make the kinds of systematic mistakes predicted by our theory. Overall, our theory shows how learning systems can balance between relational generalization and memorization, explains how this can go wrong, and emphasizes the need for new task paradigms designed to probe this ability.

[1] Center for Theoretical Neuroscience, Columbia University, New York, NY, 10027, United States. Correspondence to: Luke Cheng <lc3616@columbia.edu>, Samuel Lippl <samuel.lippl@columbia.edu>.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. Copyright 2026 by the authors.

1. Introduction

When learning to perform complex behaviors from limited data, humans, animals, and machine learning models must infer the rules and regularities of their environment (Kemp & Tenenbaum, 2008; Lake et al., 2017) (Fig. 1a). This allows them to generalize their experience to unseen situations involving familiar components, an ability also known as compositional generalization (Hupkes et al., 2020). For example, when learning a new language, we may notice that similar-sounding words (in related languages) often mean the same thing; this can help us learn new words faster. Understanding that certain environments share a stereotyped sequence of actions (e.g. at the airport we usually check our bag and then go through security) lets us better plan for the future (Baldassano et al., 2018). And when learning chess, understanding that a queen is usually worth more to us than a knight will let us avoid making costly mistakes.

Yet in most complex environments, there will be exceptions to these rules (Fig. 1b). For example, "Gift" in German means poison, not gift; we sometimes have to check a bag at the gate rather than before the security check; and in some cases sacrificing a queen can win us the game. Effective behavior requires learning these exceptions in addition to the regular structure (Fig. 1c).

Thus, any learning system deployed in the real world will have to balance learning general rules with memorizing exceptions to those rules. While an extensive theoretical and empirical literature has investigated compositional generalization and memorization in isolation, it has remained unclear how learning systems can do both simultaneously (though see Rumelhart & Mcclelland, 1985). Part of the issue is a lack of suitable task paradigms: while real-world tasks require a balance between rule learning and memorization, simple tasks amenable to theoretical study can often be solved by rules without considering any exceptions.

Here we address this gap, considering relational generalization as an important instance of compositional generalization (Halford et al., 2010; Battaglia et al., 2018). A canonical instance of a relational rule is given by transitivity: $A > B$ and $B > C$ implies $A > C$. Accordingly, testing for the ability to transitively generalize ("transitive inference," TI,

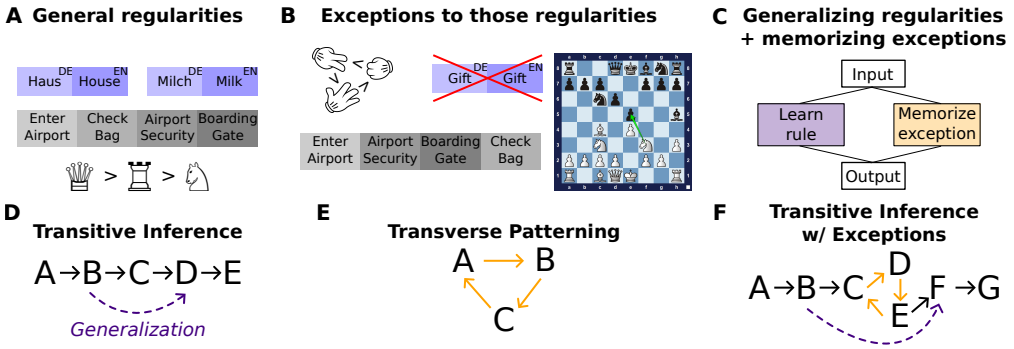

Figure 1. Complex behavior in the real world requires a mixture of general rule learning and memorizing exceptions. **A**, Identifying general regularities allows us to generalize to unseen situations. **B**, Most of these regularities have exceptions. (The chessboard shown here depicts an example of Legal's Mate, a standard example where a player should sacrifice their queen.) **C**, Optimal behavior in the real world requires learning these general regularities while also memorizing exceptions. **D**-**F**, Cognitive task paradigms probing the above abilities. *D*, Transitive inference requires models or subjects to generalize using the transitive rule. *E*, Transverse patterning tests whether subjects or models can learn intransitive relations. *F*, We introduce a novel task: transitive inference with exceptions. This task requires subjects or models to both generalize according to the transitive rule and to memorize exceptions to the transitive rule.

Fig. 1d) has been an important task paradigm for investigating relational reasoning in humans (Piaget, 1928; Bryant & Trabasso, 1971), animals (McGonigle & Chalmers, 1977; Davis, 1992; Vasconcelos, 2008), and (increasingly) neural networks (De Lillo et al., 2001; Di Antonio et al., 2024; Kay et al., 2024; Lippl et al., 2024; Geerts et al., 2025; Liu et al., 2025). Conversely, both humans and animals (as well as neural networks) can also memorize intransitive relations (e.g. rock-paper-scissor-structures, "transverse patterning," TP, Fig. 1e) (Alvarado & Rudy, 1992).

While transitive inference and transverse patterning have been extensively studied in isolation, it has remained unclear whether and how learning systems perform on ordered relations that are mostly transitive but also include intransitive exceptions—a scenario we argue is likely common in complex decision-making tasks (see Appendix A). To study this, we make three major contributions:

1. We introduce a novel task paradigm, transitive inference with exceptions (Fig. 1f). This task is designed to probe relational generalization and memorization in a controlled setting.

2. We analytically characterize a theoretical model of neural network learning (kernel ridge regression) across a general family of representations and task parameters. We find that kernel models can generalize successfully, but that this is sensitive to their representational geometry. We then leverage our analytical theory to mechanistically understand how these models generalize or fail to generalize.

3. Finally, we validate our insights in pretrained language models finetuned on relational tasks.

Overall, our findings demonstrate how simple learning systems can balance compositional generalization and memorization. At the same time, they highlight that compositional tasks that require such a balance are substantially more challenging than tasks that only require learning regularities. As a result, these tasks highlight novel failure modes and impose stronger constraints on which systems will be successful. We suggest that dealing with exceptions is a fundamental challenge for learning systems in the wild and future work should put a stronger emphasis on formulating analytically tractable tasks that capture this complexity.

2. Related Work

Compositional generalization has long been an important evaluation criterion for machine learning systems (Fodor & Pylyshyn, 1988; Lake et al., 2017). It has become increasingly central as modern learning systems have become more capable. While explicit constraints on the compositional operations in a neural network can guarantee compositional generalization (Schug et al., 2024; Wiedemer et al., 2023; 2024; Brady et al., 2025; Jarvis et al., 2023), our work emphasizes that neural networks often need to compositionally generalize in cases where explicit constraints don't work. Relational generalization, as a special instance of compositional generalization, focuses on generalizing to unseen situations by learning how different objects or items are related to each other (Battaglia et al., 2018; Whittington et al., 2020; Webb et al., 2024). Transitive inference (TI; Fig. 1d) is a classical task paradigm for assessing this ability in humans, animals, and neural networks by testing if subjects can generalize using the transitive rule (Vasconcelos, 2008; Jensen, 2017; Geerts et al., 2025; Liu et al., 2025). In contrast, transverse patterning (TP; rock-paper-scissor-structure; Fig. 1e) tests whether subjects are able to memorize fully intransitive relations. In the real world,

many ordered relations may be largely transitive but also involve some exceptions to transitivity, a phenomenon that has been observed in a variety of contexts including competition between species, voting systems, and game theory (Trybuła, 1961; Soliveres & Allan, 2018; Arrow, 1950; Klimenko, 2015). Our work builds on a long-standing literature that uses kernel models to understand neural network generalization (Jacot et al., 2018; Chizat et al., 2019; Canatar et al., 2021a;b; Mallinar et al., 2022; Zhou et al., 2024; Abbe et al., 2024). Prior work has suggested that such a perspective can help us understand finetuning in pretrained language models (Malladi et al., 2023; Afzal et al., 2026)—this is consistent with our findings here. We refer the reader to a more detailed discussion of these related works in Appendix A.

3. Setup

3.1. Task setup

To study relational inference, we consider a set of items $I_1, \ldots, I_n$. To train models on a relation, we present pairs of items (I_j, I_k) and an associated label $y_{j,k} \in \{-1, 1\}$. If $y_{j,k} = 1$, that indicates that $I_j > I_k$; if $y_{j,k} = -1$, that indicates that $I_j < I_k$. Items are represented by orthogonal vectors $x_1, \ldots, x_n \in \mathbb{R}^d$ and each pair of items is represented by a simple concatenation of the two vectors, $x_{j,k} := \begin{pmatrix} x_j & x_k \end{pmatrix}^T \in \mathbb{R}^{2d}$. This is a standard approach for studying compositional and relational inference in neural network models (see e.g. Kay et al., 2024; Lippl et al., 2024; Abbe et al., 2024). We note that all datasets we consider are antisymmetric: if they contain $(x_{j,k}, 1)$, they also contain $(x_{k,j}, -1)$. If a model is trained on this data point, we say that it is trained on $I_j > I_k$.

Transitive inference tests whether subjects are able to infer an underlying ranking $I_1 > \cdots > I_n$. Models are trained on adjacent pairs $I_j > I_{j+1}$, and tested in their performance on (I_j, I_k) for $|j - k| \geq 2$. This tests whether models can use the transitive rule to generalize to new item pairs. **Transverse patterning** prevents models from inferring a consistent ranking system by adding the premise $I_n > I_1$, thus establishing an intransitive, cyclical relation.

Our novel task, **transitive inference with exceptions**, introduces an arbitrary exception $I_p > I_q$ for $p > q$ (and $|p - q| > 1$). This means that the task consists of two ordered sections ($\mathcal{O}^{(1)} := \{I_1, \ldots, I_{q-1}\}, \mathcal{O}^{(2)} := \{I_{p+1}, \ldots, I_n\}$), with an intransitive section in the middle ($\mathcal{L} := \{I_q, \ldots, I_p\}$) (Fig. 1f). We evaluate model predictions on four datasets (Fig. 2):

1. **Memorization of transitive pairs:** Does the model learn the training pairs where at least one item is in $\mathcal{O}^{(j)}$?
2. **Memorization of intransitive pairs:** Does the model learn the training pairs where both items are in $\mathcal{L}$?
3. **Within-section generalization:** Can the model generalize transitively within $\mathcal{O}^{(1)}$ or $\mathcal{O}^{(2)}$, i.e. on $\{(I_j, I_k) | I_j, I_k \in \mathcal{O}^{(i)}; i = 1, 2\}$?
4. **Cross-section generalization:** Can the model generalize transitively across the sections, i.e. on $\{(I_j, I_k) | (I_j \in \mathcal{O}^{(1)} \wedge I_k \in \mathcal{O}^{(2)}) \vee (I_j \in \mathcal{O}^{(2)} \wedge I_k \in \mathcal{O}^{(1)})\}$?

For example, if subjects learn $A > B > C$, $E > F > G$, and $C > D > E > C$, an example of the desired cross-section generalization is $B > F$. This type of generalization allows the exception to create a local intransitive loop, without destroying the general transitive rule. We suggest that such a heuristic will yield successful relational generalization in the majority of cases. To ground our suggested heuristic in a concrete, mathematically precise example, we consider the one-on-one win probability of different hands in Texas Hold'em poker: $x \succ y$ iff the hand x is more likely to win against hand y (see Appendix D). This ordered relation contains intransitive loops (Bartholdi & Mikhailov, 2025). In analyzing this example, we found that typically, within-section and cross-section transitive generalization would have resulted in the appropriate generalization (Fig. S20). While other intransitive relations may have a different structure, this provides an instance of a relation for which our suggested generalization is appropriate.

3.2. Model setup

Given an input $x \in \mathbb{R}^{2d}$, we consider a random features model, $\phi(x) = \sigma(Wx) \in \mathbb{R}^h$, with random weights $W \in \mathbb{R}^{h \times 2d}$ and a nonlinearity σ. We denote the kernel induced by ϕ as $K(x, x') := \phi(x)^T \phi(x')$. In the limit of infinite width, $h \to \infty$, the kernel will take on three distinct values (Lippl et al., 2024):

$$K(x_{j,k}, x_{j',k'}) = \begin{cases} \kappa_s & \text{if } j = j' \wedge k = k' \text{ (identical)}, \\ \kappa_o & \text{if } j = j' \vee k = k' \text{ (overlapping)}, \\ \kappa_d & \text{if } j \neq j' \wedge k \neq k' \text{ (distinct)}. \end{cases} \tag{1}$$

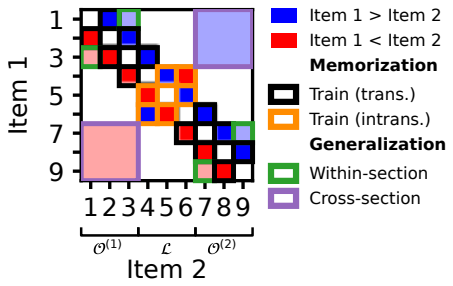

Figure 2. Transitive inference with exceptions. Illustration showing the relevant training and test sets as well as the associated relation. White squares reflect item pairs without an expected generalization.

We call such kernels exchangeable, as the similarity between pairs of items doesn't depend on item identity, only on the overlap between the items. These kernels formalize the idea that we should learn about relations in terms of the interactions between different items rather than the specific feature similarity of those items. Given a dataset $\mathcal{D} = \{(x, y)\}$, we consider predictions made by a linear readout from $\phi(x)$, $f_w(x) := \phi(x)^T w, w \in \mathbb{R}^h$ and consider ridge regression,

$$w^* := \arg\min L(w),$$
$$L(w) := \sum_{x,y\in\mathcal{D}} (f_w(x) - y)^2 + \tfrac{1}{c}\|w\|_2^2. \tag{2}$$

We consider L_2-regularization, as it often prevents overfitting to the training data and can therefore improve generalization at the cost of memorization (Nakkiran et al., 2021; Mei & Montanari, 2022). Additionally, it is particularly relevant to our empirical experiments in Section 5, as we often want to limit the weight changes in finetuning language models, so as to prevent them from forgetting their pretraining data (Kirkpatrick et al., 2017; Li et al., 2024; Béthune et al., 2025). Note that as $c \to \infty$, this network converges to the minimal ℓ_2-norm solution $\min_w \|w\|_2^2$ s.t. $f_w(x) = y$.

We evaluate model performance in terms of the margin $yf(x)$. A positive margin indicates correct generalization and we consider a higher margin as indicative of better performance. More specifically, while we consider deterministic inputs that adhere to exchangeability exactly, in an empirical context (see e.g. Section 5), these inputs might randomly vary. In that case, we assume that a higher margin will yield a more consistent output: intuitively, for points closer to the separating hyperplane, random variance in the input will be more likely to result in a perturbation that changes the prediction. For correct predictions (positive margins), that should result in increased accuracy; for incorrect predictions (negative margins), it should result in further decreases.

We will find that network behavior only depends on two factors: the conjunctivity factor

$$\alpha := 1 - 2\frac{\kappa_o - \kappa_d}{\kappa_s - \kappa_d} \in [0, 1], \tag{3}$$

and the effective regularization parameter

$$\tilde{c} := \frac{c}{\kappa_s - \kappa_d} \in (0, \infty). \tag{4}$$

Intuitively, α captures how nonlinearly entangled the items are in $\phi(x)$. $\alpha = 0$ ("compositional representation") indicates that the two items are represented in orthogonal subspaces; $\alpha = 1$ ("conjunctive representation") indicates that they are fully entangled (Section 4.2). The value of α depends on the nonlinearity, but typical values for a one-hidden-layer network are approximately in the range $(0, 0.3)$ (e.g. a ReLU network has $\alpha \approx 0.15$) (Lippl et al., 2024).

4. Theoretical Results

4.1. Main Theorem

Ranking systems. A canonical approach to encoding an ordered relation is to assign each item a numerical rank and consistently choose the item with the higher rank. Ranking systems are inherently transitive and thus support transitive generalization, but cannot encode intransitive relations.

Intriguingly, we show below that kernel models learn an emergent ranking system, but combine it with an additional conjunctive response that can memorize exceptions. We characterize the model behavior across the entire family of exchangeable representations, regularization strengths, and task parameters:

Theorem 1. *On TI with exceptions with task parameters n, p, q and the training dataset $\mathcal{D}$, the predictions of a kernel model with an exchangeable representation are given by*

$$f(x_{j,k}) := \begin{cases} r_j - r_k & \text{for } x \notin \mathcal{D}, \\ m y_{j,k} + (1-m)(r_j - r_k) & \text{for } x \in \mathcal{D}, \end{cases} \tag{5}$$

where

$$m(\alpha, \tilde{c}) := \tfrac{\alpha}{\alpha + \tilde{c}^{-1}}, \quad r_j(\alpha, \tilde{c}) := r_j^{\mathrm{TI}}(\alpha, \tilde{c}) + r_j^{\mathrm{pert}}(\alpha, \tilde{c}). \tag{6}$$

Here r_j^{TI} is the ranking system emergent for standard TI (S20) and r_j^{pert} (defined in (S21)) is the perturbation to the ranking system arising from the exception $I_p > I_q$ (Fig. 3A). For the training data, the term $m(\alpha, \tilde{c}) y_{j,k}$ demonstrates the conjunctive response that can memorize the exception.

Proof sketch. We analytically solve the ridge regression problem (2) through the dual form, which involves inverting the training set kernel K. We relate this to the inverse for the original TI problem (Lippl et al., 2024), which allows us to express the analytical solution in terms of a perturbation on top of that solution. We then simplify that solution algebraically. For the detailed derivation, see Appendix E.

An encoding model. To provide an intuition for the behavior of our model, we show in Appendix B.2 that any exchangeable representation is rotationally equivalent to a four-hot representation concatenating two vectors representing the first and the second item (the "compositional population"), a vector representing the specific conjunction of the two items (the "conjunctive population"), and a unit that is constantly active (the "bias") (Fig. 3c). Different representational geometries correspond to different weightings of these populations. We can therefore rewrite model behavior in terms of separate readouts from these distinct populations: $f(x_{j,k}) = r_j - r_k + t_{j,k} + b$. The readout from the bias term is zero, the readout from the compositional population implements the ranking system, and the readout

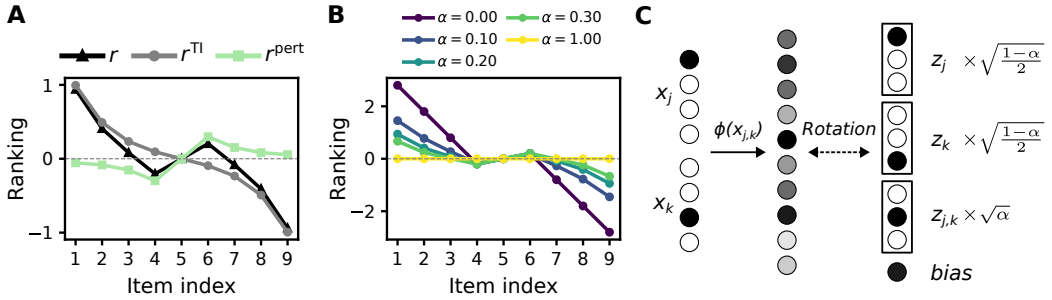

Figure 3. Illustration of the theorem. **A**, The ranking system consists of the original rank from TI, r^{TI}, plus a perturbation r^{pert} (example here uses $\alpha = 0.2$). **B**, Example ranking systems (for $n = 9, p = 6, q = 4, \tilde{c} \to \infty$). **C**, Any exchangeable representation is rotationally equivalent to a four-hot representation where two units represent each input item x_j and x_k separately, one unit represents their conjunction, and one unit is active for all inputs. Different representational geometries are reflected in different weightings of these populations.

from the conjunctive population (which we will call the "conjunctive readout") implements an additional conjunctive response that is only active on the training set. We note that the conjunctive readout allows the model to memorize the intransitive loop.

4.2. The extreme cases: fully compositional or fully conjunctive representations

To get an intuition for model behavior, we first consider a fully additive model ($\alpha = 0$). This corresponds to the case where only the compositional population is active and the model implements a ranking system on both training and test set (as noted in (Lippl et al., 2024)): it infers a rank r_j for each item I_j and computes $f(x_{j,k}) = r_j - r_k$. This is an inherently transitive computation as $f(x_{j,k}) > 0$ (corresponding to the judgement that $I_j > I_k$) is equivalent to $r_j > r_k$. While the model will therefore learn the transitive pairs, it will necessarily make errors on the intransitive pairs.

More specifically, the model will learn a flat ranking system with respect to the intransitive items, while learning the correct ranking system with respect to all other items (Fig. 3b, purple line). The flat ranking system allows the model to distribute its errors evenly across all intransitive pairs—thus minimizing the mean squared error. This means that an additive representation can generalize both within and across sections, but cannot memorize the exceptions.

In the opposite extreme ($\alpha = 1$), only the conjunctive population is active. Such a model can learn arbitrary training data, as it effectively implements a lookup table, $f(x_{j,k}) = t_{j,k}$. But it also has no understanding of how different items are related to each other and is therefore unable to generalize.

Overall, the extreme representations ($\alpha = 0, 1$) represent the extreme strategies described in the introduction: fully rule-based generalization or full memorization. Our insights here are consistent with prior insights on TI and TP. We now turn to the more interesting case of $\alpha \in (0, 1)$, which we will see balances between memorization and generalization.

4.3. Task Performance Depends on Representational Geometry and Regularization

We now use our analytical solution, which is applicable across representations and task parameters, to develop a detailed understanding of training and test performance.

Universal properties of the ranking system. We find that the ranking system has a stereotyped shape: it is always monotonically decreasing on the two transitive lists and typically monotonically *increasing* over the intransitive loop (Fig. 3b). We emphasize that this non-monotonicity in the ranking system only arises for $\alpha \in (0, 1)$; for both extreme cases, the ranking system is flat on the intransitive loop. This demonstrates that a model balancing between relational generalization and memorization will exhibit new and interesting behaviors that go beyond a simple mixture of a model that can only generalize and a model that can only memorize.

Generalization. Because the ranking system is always monotonically decreasing within the two transitive lists (Appendix E.8), we can immediately conclude that any $\alpha > 0$ will yield **successful within-section generalization**. For a suitable representational geometry, the model is also capable of **cross-section generalization** (e.g. Fig. 4A, middle and right). However, this is not guaranteed (e.g. Fig. 4A, left). Specifically, the ranking system's non-monotonocity in the intransitive section propagates into the transitive sections as well: the pre-exception items' ranks are biased downwards and the post-exception items' ranks are biased upwards. For a sufficiently large α or a sufficiently small $\tilde{c}$, this means that the model makes systematic errors in generalization, predicting, in particular, that $I_{q-1} < I_{p+1}$ (Fig. 4B). We emphasize that kernel models never make such mistakes for standard TI (for $\alpha < 1$); this highlights that more composi-

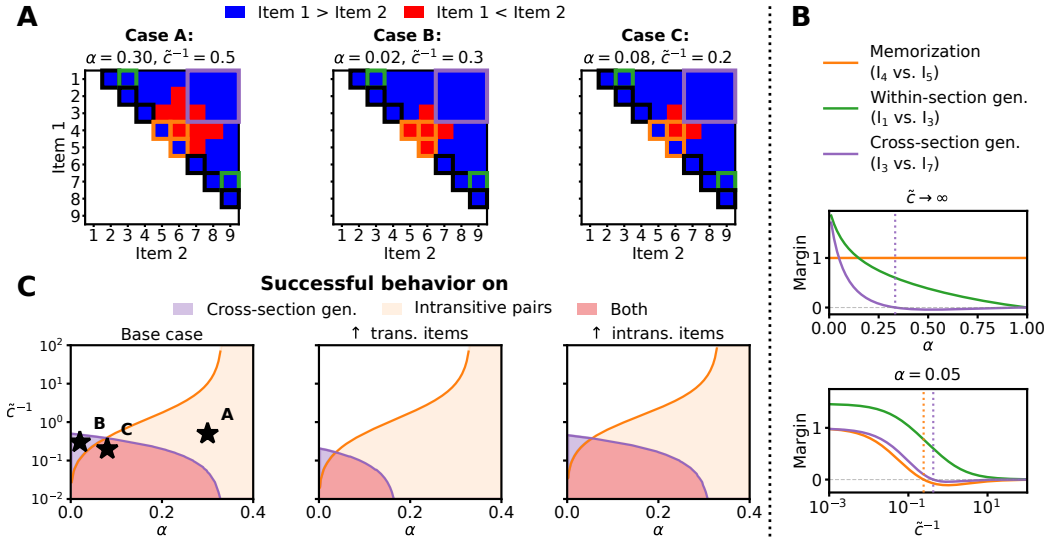

Figure 4. Behavior of the kernel model across task space. **A**, Generalization behavior for $n = 9, p = 6, q = 4$. We show three distinct parameter settings that lead to: successful memorization, but errors in generalization (case A); successful generalization, but errors in memorization (case B); and successful memorization and generalization (case C). **B**, Changes in the margin as a function of α or $\tilde{c}$. Values where the margin becomes negative (indicating systematic errors) are highlighted with dashed vertical lines. **C**, Phase diagrams which show the effect of changing α and $\tilde{c}^{-1}$ on performance.

tional representations (with small α) may actually be more important for tasks involving exceptions.

Memorization. For $\alpha \in (0, 1)$ and $\tilde{c} \to \infty$, the model will memorize all training data points. However, as $\tilde{c}$ decreases, the model eventually starts making systematic errors on the exception pairs (for all task variants except TP): while it still gets (I_p, I_q) right, it starts getting (I_q, I_{q+1}) and (I_{p-1}, I_p) wrong (Fig. 4B, bottom). Our analytical solution shows that this is because as $\tilde{c}$ decreases, $m(\alpha, \tilde{c})$ becomes smaller, meaning that the ranking system more strongly influences training set performance. The errors therefore arise from the non-monotonicity in the ranking system. Notably, for smaller α, the model makes errors for a broader range of $\tilde{c}$ values. This demonstrates that more conjunctive representations will more easily account for an intransitive structure. We emphasize that on transverse patterning, the model never makes any systematic errors, regardless of $\tilde{c}$. This demonstrates that task structures that partially exhibit relational structure (such as TI with exceptions) can make memorization of exceptions more challenging.

Overall, our analysis demonstrates two things: On the one hand, even a relatively simple learning model can balance relational generalization with memorization of exceptions. On the other hand, this is much more challenging than *only* learning a relation (as in TI), or only having to memorize (as in TP). We summarize these insights in terms of a single diagram demonstrating the regions of the $(\alpha, \tilde{c})$ where the model successfully generalizes and memorizes, and the regions where it fails to do one of the two or both (Fig. 4C, left). A notable and surprising takeaway from our analysis is that L_2-regularization (which aids in-distribution generalization) is actually counterproductive for out-of-distribution generalization and can induce systematic errors. Furthermore, our analysis highlights that TI with exceptions gives rise to a trade-off between more conjunctive representations (better for memorization) and more compositional representations (better for generalization).

Effects of task parameters. We now investigate the impact of changing different task parameters. We focus on cases where the intransitive loop is still located in the middle of the list (i.e. $q + 1 = n - p$), and vary either the length of the intransitive loop ("↑ intransitive items") or the length of the two transitive sections ("↑ transitive items") (Fig. 4C, middle and right). While memorization ability is not strongly affected by either change, additional generalizable items renders cross-section generalization much more challenging, as indicated by the area of successful generalization being much smaller. Thus, increasing the length of the transitive sections makes generalization much more challenging, whereas increasing the length of the intransitive loop has a much weaker effect.

4.4. A Mechanistic Explanation for the Non-Monotonic Ranking System

The central challenge to both cross-section generalization and memorization of intransitive pairs is the non-monotonicity in the ranking system. We now leverage our analytical solution to mechanistically explain this non-monotonicity. We base our intuition around the encoding model we provided above, which expressed model

behavior in terms of distinct latent populations responding to single components and component conjunctions: $f(x_{j,k}) = r_j - r_k + t_{j,k}$. We express the effect of L_2-regularization directly in terms of this ranking system, $C(r,t) := \frac{2}{1-\alpha}\|r\|_2^2 + \frac{1}{\alpha}\|t\|_2^2$ (see Appendix B.2). Intuitively, more additive representations will incur a higher cost for using $t_{j,k}$ and more conjunctive representations will incur a higher cost for using r_j. For simplicity, we focus on $\tilde{c} \to \infty$. Thus, the model needs to match the prescribed labels exactly while minimizing $C(r,t)$.

We now focus on two different subsets of the training data: First, if the model is only fit to the two transitive sections, the emergent ranking system will be centered around zero in both segments (Fig. S6A, solid blue line). This is because centering these ranking systems around zero minimizes $\|r\|_2^2$. Second, if the model is only fit to the intransitive loop, the ranking system will be flat at zero, as any other ranking will increase $\|t\|_2^2$ (Fig. S6B).

The non-monotonicity arises from the interaction between these two incentives: when using the first solution, the model would substantially increase its cost on t when memorizing the intransitive loop (Fig. S6B). Yet, when using the second solution, the ranking systems will be shifted far away from zero—incurring increased costs on the ranking system. The solution therefore finds a compromise between these two systems, resulting in a non-monotonic ranking system, albeit less extreme than the parallel ranking systems.

How does α change this tradeoff? For small α, memorization is extremely costly whereas the ranking system is comparatively cheap. Thus, the model will focus on minimizing the memorization cost and keep the ranks of the intransitive loop relatively flat. In contrast, for large α, memorization is cheap and the ranking system costly, yielding a more non-monotonic ranking system.

While we focused on $\tilde{c} \to \infty$, this intuition extends to smaller $\tilde{c}$ as well. In that case, mistakes made by the ranking system can not only be compensated by the conjunctive coefficient, but can also be absorbed by the loss function. This similarly increases the non-monotonicity in the ranking system.

5. Finetuning Behavior in Pretrained Language Models

We now evaluate our theoretical predictions in pretrained language models (PLMs), which we finetune on novel hierarchies. We consider either the likely winner between two sports teams (which are sampled from either real team names or fake team names; $n = 100$ random seeds per setting), or the likely winner between two hands in Texas Hold'em Poker (using either the ground truth or the exact reverse; $n = 50$ random seeds each). We present each relation as a question/answer prompt (see Fig. 5A). We consider three distinct finetuning strategies: 1) full finetuning (full FT), 2) low-rank adaptation (LoRA, (Hu et al., 2022)), and 3) finetuning of the linear readout head with L_2-regularization with respect to the initial values (linear probe). We consider four different language models (Llama3.2-1B/3B (Touvron et al., 2023) and Qwen3.5-2B/4B (Qwen Team, 2026); for full finetuning, we only consider the two smaller models). We summarize our findings here and describe the setup and results in more detail in Appendix C.

As noted in Section 3.2, we connect the accuracy metrics of the PLMs to the predictions made by the kernel model in two ways: 1) The kernel model making the correct/incorrect prediction is analogous to the PLM behavior having above/below chance accuracy. 2) The overall margin may predict the overall magnitude of the accuracy.

We emphasize that our relational learning task is a practically relevant scenario: Language models are getting exposed to a lot of relational information in their training data. Extending these relations to novel combinations of items would let them generalize to a broad range of new inputs (Battaglia et al., 2018).

We first evaluated the models on a transitive inference task without any exceptions. We found that the models generalized successfully on this task (Fig. 5B; Appendix C.2). Moreover, their generalization behavior shared important characteristics with that of humans and animals (e.g. lower accuracy for intermediate item comparisons, see Appendix C.3). Finally, PLMs generalized better on tasks that were consistent with their prior knowledge (i.e. poker) than tasks that were inconsistent (i.e. poker with a reversed hierarchy), a phenomenon that has previously been observed in both PLMs and humans (Evans et al., 1983; Dasgupta et al., 2022; Lampinen et al., 2024) (see Appendix C.6). Overall, this demonstrates that PLMs are able to compositionally generalize over relational data that they were finetuned on and contributes to a growing literature evaluating Transformers and PLMs on transitive inference (Wu & Deng, 2025; Geerts et al., 2025; Liu et al., 2025). We note that linear probing and full finetuning generally led to higher performance than LoRA (Fig. S11) (Biderman et al., 2024).

Next, we considered models finetuned on TI with exceptions. Consistent with our theory, we found that models broadly generalized above chance within-section (Fig. 5C,D; Appendix C.4), whereas memorization of the exception pairs and cross-section generalization was much more inconsistent, with some models having high performance and others making systematic errors (indicated by accuracies below 50%). Thus, finetuned PLMs can balance relational generalization and memorization, but some aspects of generalization (pinpointed by our theory) are brittle.

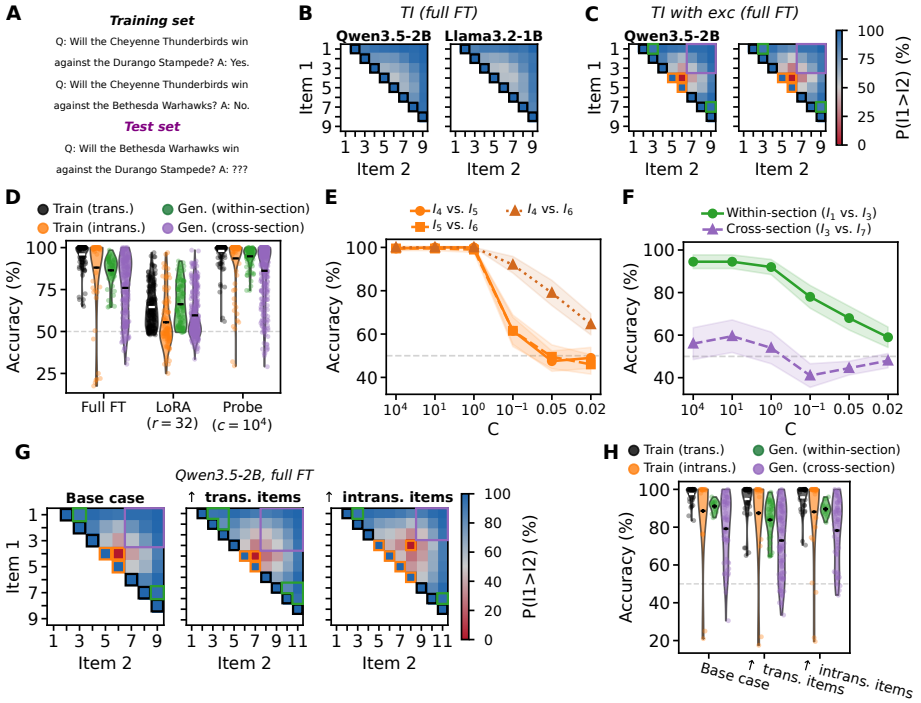

Figure 5. Finetuning pretrained language models (PLMs) on relational data with exceptions. **A**, Example of the data the models are finetuned on. **B**,**C**, Generalization behavior of different PLMs on *B*, TI and *C*, TI with exceptions ($n = 9, p = 6, q = 4$). Here we plot performance on fake sports team, other modalities see Figs. S8, S12. For *C*, see Fig. 2 for the desired generalization. **D**, Accuracy on distinct subsets of the data across the different finetuning methods. Each dot corresponds to a specific rank comparison, task modality, and model. We summarize the data in terms of the mean and 95% binomial confidence intervals (often too small to see). **E**, **F**, Performance of linear probing in Qwen3.5-4B on *E*, exception pairs and *F*, generalization pairs. Shaded regions represent 95% binomial confidence intervals. **G**, **H**, Generalization behavior for the base task, a task with additional generalizable items ($n = 11, p = 7, q = 5$), or a task with additional exception items ($n = 11, p = 8, q = 4$). Heatmaps show fake teams (see Fig. S18, S19 for other cases).

We then considered the impact of regularization. We first considered training set behavior, focusing on the intransitive loop (Figs. 5e, S16). We found that stronger regularization generally decreased accuracy, with substantially faster decreases on $x_{4,5}$ and $x_{5,6}$ than $x_{4,6}$, as predicted by our theory. We then considered generalization behavior, comparing the impact of regularization on within-section and cross-section generalization (exemplified by $x_{1,3}$ vs. $x_{3,7}$, the most challenging generalization pair according to our theory, see Fig. 4B). We again observed the behavior predicted by our theory: as regularization becomes stronger, predictions on $x_{3,7}$ eventually become systematically incorrect (Figs. 5F, S17). In contrast, within-section accuracy decreases as regularization increases (consistent with the kernel model's output becoming smaller), but never goes below chance.

Finally, we varied the task parameters, either increasing the number of generalizable items ($n = 11, p = 7, q = 5$) or increasing the length of the intransitive loop ($n = 11, p = 8, q = 4$). Consistent with our theory, we found that increasing the number of generalizable items yielded a stronger decrease in performance than increasing the number of exception items (Fig. 5G,H, Fig. S15). Overall, these results indicate that our theory captures key aspects of relational generalization and memorization in pretrained language models finetuned on relational data.

6. Discussion

Complex behavior in the real world requires not just inferring general rules, but also memorizing exceptions to those rules. To study how learning systems balance these opposing demands, we have introduced a novel task paradigm grounded in the relational learning literature—transitive inference with exceptions. We analytically characterized the behavior of kernel models on this task, finding that a successful balance between relational generalization and memorization was sensitive to the models' representational geometry, and subsequently validated our theory in pretrained language models finetuned on novel relations.

Our theory highlights (and our experiments confirm) that exceptions render successful generalization much more challenging. This highlights that task paradigms like TI with exceptions can 1) help us understand the conditions under which learning systems generalize on such complex tasks and 2) can serve as a benchmark for the development of

learning systems that avoid their pitfalls. Better learning systems could, for example, separate the learning of general structure and exceptions into distinct streams of processing (Schapiro et al., 2017) or use meta-learning to imbue the appropriate inductive biases (Lake & Baroni, 2023).

We focused on kernel models with exchangeable representations. Kernel models are an influential model in deep learning theory, but simplify important aspects of neural network learning. Future work should investigate how feature-learning neural networks generalize on TI with exceptions. Exchangeable representations capture the idea that we should be able to relationally generalize over different items without any particular knowledge about the items themselves; by relaxing this assumption, we could investigate how prior knowledge (e.g. about poker) could affect relational reasoning (Dasgupta et al., 2022; Dominé et al., 2023; Lampinen et al., 2024). Finally, we have considered one specific instance of compositional generalization. Expanding this to a broader battery of tasks could highlight important distinctions and convergences in balancing generalization and memorization across different compositional structures. Our conceptual and theoretical framework provides a foundation for such future advances. Overall, we suggest that explicitly modeling the effect of exceptions to general rules is critical for understanding and improving compositional generalization in the real world.

Acknowledgments

We thank Larry Abbott, Ethan Hwang, Chris Iyer, Berkan Ottlik, Erica Shook, and Marcus Triplett for helpful discussions and comments. We thank Justin Buck for help in brainstorming intransitivities in games and Kollin Wasserlein for generating fictional sports team names. This work was supported by a grant from the Simons Foundation International 542939SPI, LFA; SFI-AN-NC-GB-Culmination-00003215-01.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abbe, E., Bengio, S., Lotfi, A., and Rizk, K. Generalization on the unseen, logic reasoning and degree curriculum. *Journal of Machine Learning Research*, 25(331):1–58, 2024.

Afzal, Z. R., Esmaeilbeig, T., Soltanalian, M., and Ohannessian, M. I. Linearization explains fine-tuning in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=tdwRIP6NG2.

Alvarado, M. C. and Rudy, J. W. Some properties of configural learning: an investigation of the transverse-patterning problem. *Journal of Experimental Psychology: Animal Behavior Processes*, 18(2):145, 1992.

Anguita, N., Locatello, F., Saxe, A. M., Mondelli, M., Mancini, F., Lippl, S., and Domine, C. A theory of how pretraining shapes inductive bias in fine-tuning. *arXiv preprint arXiv:2602.20062*, 2026.

Arrow, K. J. A difficulty in the concept of social welfare. *Journal of political economy*, 58(4):328–346, 1950.

Astur, R. S. and Sutherland, R. J. Configural learning in humans: The transverse patterning problem. *Psychobiology*, 26(3):176–182, 1998.

Baldassano, C., Hasson, U., and Norman, K. A. Representation of real-world event schemas during narrative perception. *Journal of Neuroscience*, 38(45):9689–9699, 2018.

Bar-Hillel, M. and Margalit, A. How vicious are cycles of intransitive choice? *Theory and decision*, 24(2):119–145, 1988.

Bartholdi, L. and Mikhailov, R. The topology of poker. *Games and Economic Behavior*, 2025.

Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

Barzilai, D., Kornowski, G., and Shamir, O. Beyond benign overfitting in nadaraya-watson interpolators. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=XbAtacZeEp.

Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

Berglund, L., Tong, M., Kaufmann, M., Balesni, M., Stickland, A. C., Korbak, T., and Evans, O. The reversal curse: LLMs trained on "a is b" fail to learn "b is a". In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GPKTIktA0k.

Béthune, L., Grangier, D., Busbridge, D., Gualdoni, E., marco cuturi, and Ablin, P. Scaling laws for forgetting during finetuning with pretraining data injection. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=vWMij23BmQ.

Biderman, D., Portes, J., Ortiz, J. J. G., Paul, M., Greengard, P., Jennings, C., King, D., Havens, S., Chiley, V., Frankle, J., Blakeney, C., and Cunningham, J. P. LoRA learns less and forgets less. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=aloEru2qCG. Featured Certification.

Boix-Adsera, E., Saremi, O., Abbe, E., Bengio, S., Littwin, E., and Susskind, J. M. When can transformers reason with abstract symbols? In *The Twelfth International Conference on Learning Representations*, 2023.

Brady, J., von Kügelgen, J., Lachapelle, S., Buchholz, S., Kipf, T., and Brendel, W. Interaction asymmetry: A general principle for learning composable abstractions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=cCl10IU836.

Bryant, P. E. and Trabasso, T. Transitive inferences and memory in young children. *Nature*, 1971.

Buss, L. Competitive intransitivity and size-frequency distributions of interacting populations. *Proceedings of the National Academy of Sciences*, 77(9):5355–5359, 1980.

Canatar, A., Bordelon, B., and Pehlevan, C. Out-of-distribution generalization in kernel regression. *Advances in Neural Information Processing Systems*, 34:12600–12612, 2021a.

Canatar, A., Bordelon, B., and Pehlevan, C. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021b.

Chatterji, N. S., Long, P. M., and Bartlett, P. L. The interplay between implicit bias and benign overfitting in two-layer linear networks. *Journal of machine learning research*, 23(263):1–48, 2022.

Chizat, L. and Bach, F. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on learning theory*, pp. 1305–1338. PMLR, 2020.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. *Advances in neural information processing systems*, 32, 2019.

Conrey, B., Gabbard, J., Grant, K., Liu, A., and Morrison, K. E. Intransitive dice. *Mathematics Magazine*, 89(2): 133–143, 2016.

Dasgupta, I., Lampinen, A. K., Chan, S. C., Sheahan, H. R., Creswell, A., Kumaran, D., McClelland, J. L., and Hill, F. Language models show human-like content effects on reasoning tasks. *arXiv preprint arXiv:2207.07051*, 2022.

Davis, H. Transitive inference in rats (rattus norvegicus). *Journal of Comparative Psychology*, 106(4):342, 1992.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., and Dolan, R. J. Model-based influences on humans' choices and striatal prediction errors. *Neuron*, 69(6):1204–1215, 2011.

De Lillo, C., Floreano, D., and Antinucci, F. Transitive choices by a simple, fully connected, backpropagation neural network: implications for the comparative study of transitive inference. *Animal Cognition*, 4(1):61–68, 2001.

Di Antonio, G., Raglio, S., and Mattia, M. A geometrical solution underlies general neural principle for serial ordering. *Nature Communications*, 15(1):8238, 2024.

Dominé, C. C. J., Braun, L., Fitzgerald, J. E., and Saxe, A. M. Exact learning dynamics of deep linear networks with prior knowledge. *Journal of Statistical Mechanics: Theory and Experiment*, 2023(11):114004, 2023.

Dusek, J. A. and Eichenbaum, H. The hippocampus and transverse patterning guided by olfactory cues. *Behavioral neuroscience*, 112(4):762, 1998.

Elmoznino, E., Jiralerspong, T., Bengio, Y., and Lajoie, G. Towards a formal theory of representational compositionality. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=fXCfT7ErvL.

Evans, J. S. B. and Perry, T. S. Belief bias in children's reasoning. *Cahiers de Psychologie Cognitive/Current Psychology of Cognition*, 1995.

Evans, J. S. B., Barston, J. L., and Pollard, P. On the conflict between logic and belief in syllogistic reasoning. *Memory & cognition*, 11(3):295–306, 1983.

Fodor, J. A. and Pylyshyn, Z. W. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2): 3–71, 1988.

Geerts, J., Chan, S., Clopath, C., and Stachenfeld, K. Relational reasoning and inductive bias in transformers trained on a transitive inference task. *arXiv preprint arXiv:2506.04289*, 2025.

Gilpin, M. E. Limit cycles in competition communities. *The American Naturalist*, 109(965):51–60, 1975.

Grosenick, L., Clement, T. S., and Fernald, R. D. Fish can infer social rank by observation alone. *Nature*, 445(7126): 429–432, 2007.

Gunasekar, S., Lee, J., Soudry, D., and Srebro, N. Characterizing implicit bias in terms of optimization geometry. In *International Conference on Machine Learning*, pp. 1832–1841. PMLR, 2018.

Halford, G. S., Wilson, W. H., and Phillips, S. Relational knowledge: The foundation of higher cognition. *Trends in cognitive sciences*, 14(11):497–505, 2010.

Hu, E. J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Hu, G.-Y. and O'Connell, R. F. Analytical inversion of symmetric tridiagonal matrices. *Journal of Physics A: Mathematical and General*, 29(7):1511–1513, 1996.

Huisman, J. and Weissing, F. J. Biodiversity of plankton by species oscillations and chaos. *Nature*, 402(6760): 407–410, 1999.

Hupkes, D., Dankers, V., Mul, M., and Bruni, E. Compositionality decomposed: How do neural networks generalise? *Journal of Artificial Intelligence Research*, 67: 757–795, 2020.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Jarvis, D., Klein, R., Rosman, B., and Saxe, A. M. On the specialization of neural modules. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Fh97BDaR6I.

Jensen, G. Serial learning. 2017.

Ji, Z., Dudík, M., Schapire, R. E., and Telgarsky, M. Gradient descent follows the regularization path for general losses. In *Conference on Learning Theory*, pp. 2109–2136. PMLR, 2020.

Johnson, J., Hariharan, B., Van Der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Jr., R. P. S. The paradox of nontransitive dice. *The American Mathematical Monthly*, 101(5):429–436, 1994. doi: 10.1080/00029890.1994.11996968. URL https://doi.org/10.1080/00029890.1994.11996968.

Karkada, D., Turnbull, J., Liu, Y., and Simon, J. B. Predicting kernel regression learning curves from only raw data statistics. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nn5Vf6GEsV.

Kay, K., Biderman, N., Khajeh, R., Beiran, M., Cueva, C. J., Shohamy, D., Jensen, G., Wei, X.-X., Ferrera, V. P., and Abbott, L. Emergent neural dynamics and geometry for generalization in a transitive inference task. *PLOS Computational Biology*, 20(4):e1011954, 2024.

Kemp, C. and Tenenbaum, J. B. The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31):10687–10692, 2008.

Kerr, B., Riley, M. A., Feldman, M. W., and Bohannan, B. J. Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors. *Nature*, 418(6894):171–174, 2002.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Klimenko, A. Y. Intransitivity in theory and in the real world. *Entropy*, 17(6):4364–4412, 2015.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Lake, B. and Baroni, M. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pp. 2873–2882. PMLR, 2018.

Lake, B. M. and Baroni, M. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121, 2023.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences*, 40:e253, 2017.

Lampinen, A. K. and Ganguli, S. An analytic theory of generalization dynamics and transfer learning in deep linear networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryfMLoCqtQ.

Lampinen, A. K., Dasgupta, I., Chan, S. C., Sheahan, H. R., Creswell, A., Kumaran, D., McClelland, J. L., and Hill, F. Language models, like humans, show content effects on reasoning tasks. *PNAS nexus*, 3(7):pgae233, 2024.

Lepori, M., Serre, T., and Pavlick, E. Break it down: Evidence for structural compositionality in neural networks. *Advances in Neural Information Processing Systems*, 36: 42623–42660, 2023.

Li, H., Ding, L., Fang, M., and Tao, D. Revisiting catastrophic forgetting in large language model tuning. In *Findings of the association for computational linguistics: EMNLP 2024*, pp. 4297–4308, 2024.

Lippl, S. and Lindsey, J. Inductive biases of multi-task learning and finetuning: multiple regimes of feature reuse. *Advances in Neural Information Processing Systems*, 37: 118745–118776, 2024.

Lippl, S. and Stachenfeld, K. When does compositional structure yield compositional generalization? a kernel theory. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=FPBce2P1er.

Lippl, S., Kay, K., Jensen, G., Ferrera, V. P., and Abbott, L. A mathematical theory of relational generalization in transitive inference. *Proceedings of the National Academy of Sciences*, 121(28):e2314511121, 2024.

Liu, A., Prior, H., Balasubramaniam, G., Moroshko, R., Zait, A., Labzovsky, I., Karmon, D., Dasgupta, I., Stachenfeld, K., and Marino, K. Recoglab: a framework testing relational reasoning & cognitive hypotheses on llms. In *The Thirteenth International Conference on Learning Representations*, 2025.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Malladi, S., Wettig, A., Yu, D., Chen, D., and Arora, S. A kernel-based view of language model fine-tuning. In *International Conference on Machine Learning*, pp. 23610–23641. PMLR, 2023.

Mallinar, N., Simon, J., Abedsoltan, A., Pandit, P., Belkin, M., and Nakkiran, P. Benign, tempered, or catastrophic: Toward a refined taxonomy of overfitting. *Advances in neural information processing systems*, 35:1182–1195, 2022.

May, R. M. and Leonard, W. J. Nonlinear aspects of competition between three species. *SIAM journal on applied mathematics*, 29(2):243–253, 1975.

McGonigle, B. O. and Chalmers, M. Are monkeys logical? *Nature*, 267(5613):694–696, 1977.

Mei, S. and Montanari, A. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022.

Mitchell, E., Finn, C., and Manning, C. Challenges of acquiring compositional inductive biases via meta-learning. In Guyon, I., van Rijn, J. N., Treguer, S., and Vanschoren, J. (eds.), *AAAI Workshop on Meta-Learning and MetaDL Challenge*, volume 140 of *Proceedings of Machine Learning Research*, pp. 138–148. PMLR, 09 Feb 2021. URL https://proceedings.mlr.press/v140/mitchell21a.html.

Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N. D., and Gershman, S. J. The successor representation in human reinforcement learning. *Nature human behaviour*, 1(9):680–692, 2017.

Nakkiran, P., Venkat, P., Kakade, S. M., and Ma, T. Optimal regularization can mitigate double descent. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=7R7fAoUygoa.

Nelli, S., Braun, L., Dumbalska, T., Saxe, A., and Summerfield, C. Neural knowledge assembly in humans and neural networks. *Neuron*, 111(9):1504–1516, 2023.

Pahikkala, T., Waegeman, W., Tsivtsivadze, E., Salakoski, T., and De Baets, B. Learning intransitive reciprocal relations with kernel methods. *European Journal of Operational Research*, 206(3):676–685, 2010. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2010.03.018. URL https://www.sciencedirect.com/science/article/pii/S0377221710002171.

Piaget, J. Judgment and reasoning in the child. 1928.

Press, O., Zhang, M., Min, S., Schmidt, L., Smith, N. A., and Lewis, M. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Redhardt, F., Akram, Y., and Schug, S. Scaling can lead to compositional generalization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=hZt0daVIZi.

Rumelhart, D. E. and Mcclelland, J. L. On learning the past tenses of english verbs. Technical report, 1985.

Schapiro, A. C., Turk-Browne, N. B., Botvinick, M. M., and Norman, K. A. Complementary learning systems within the hippocampus: a neural network modelling approach to reconciling episodic memory with statistical learning. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1711), 2017.

Schug, S., Kobayashi, S., Akram, Y., Wolczyk, M., Proca, A. M., von Oswald, J., Pascanu, R., Sacramento, J., and Steger, A. Discovering modular solutions that generalize compositionally. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=H98CVcX1eh.

Sinervo, B. and Lively, C. M. The rock–paper–scissors game and the evolution of alternative male strategies. *Nature*, 380(6571):240–243, 1996.

Soliveres, S. and Allan, E. Everything you always wanted to know about intransitive competition but were afraid to ask, 2018.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19 (70):1–57, 2018.

Tibbetts, E. A., Agudelo, J., Pandit, S., and Riojas, J. Transitive inference in polistes paper wasps. *Biology letters*, 15(5), 2019.

Tong, W. L. and Pehlevan, C. Learning richness modulates equality reasoning in neural networks. In *8th Annual Conference on Cognitive Computational Neuroscience*, 2025. URL https://openreview.net/forum?id=QaDfpS2dg2.

Tong, W. L., Cakar, E., and Pehlevan, C. Boule or baguette? a study on task topology, length generalization, and the benefit of reasoning traces. *arXiv preprint arXiv:2602.14404*, 2026.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Trybuła, S. On the paradox of three random variables. *Applicationes Mathematicae*, 5(4):321–332, 1961.

Tsigler, A. and Bartlett, P. L. Benign overfitting in ridge regression. *Journal of Machine Learning Research*, 24 (123):1–76, 2023.

Vasconcelos, M. Transitive inference in non-human animals: An empirical and theoretical analysis. *Behavioural Processes*, 78(3):313–334, 2008.

Vyas, N., Bansal, Y., and Nakkiran, P. Empirical limitations of the ntk for understanding scaling laws in deep learning. *Transactions on Machine Learning Research*, 2023.

Webb, T. W., Frankland, S. M., Altabaa, A., Segert, S., Krishnamurthy, K., Campbell, D., Russin, J., Giallanza, T., O'Reilly, R., Lafferty, J., et al. The relational bottleneck as an inductive bias for efficient abstraction. *Trends in Cognitive Sciences*, 28(9):829–843, 2024.

Whittington, J. C., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., and Behrens, T. E. The tolman-eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183(5):1249–1263, 2020.

Wiedemer, T., Mayilvahanan, P., Bethge, M., and Brendel, W. Compositional generalization from first principles. *Advances in Neural Information Processing Systems*, 36: 6941–6960, 2023.

Wiedemer, T., Brady, J., Panfilov, A., Juhos, A., Bethge, M., and Brendel, W. Provable compositional generalization for object-centric learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=7VPTUWkiDQ.

Woodworth, B., Gunasekar, S., Lee, J. D., Moroshko, E., Savarese, P., Golan, I., Soudry, D., and Srebro, N. Kernel and rich regimes in overparametrized models. In *Conference on Learning Theory*, pp. 3635–3673. PMLR, 2020.

Wu, B., Fang, J., Zeng, X., Liang, S., and Zhang, Q. Adaptive compositional continual meta-learning. In *International Conference on Machine Learning*, pp. 37358–37378. PMLR, 2023.

Wu, W. and Deng, W. Transitive inference in large language models and prompting intervention. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

Yang, S., Gribovskaya, E., Kassner, N., Geva, M., and Riedel, S. Do large language models latently perform multi-hop reasoning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10210–10229, 2024.

Zeithamova, D. and Preston, A. R. Flexible memories: differential roles for medial temporal lobe and prefrontal cortex in cross-episode binding. *Journal of Neuroscience*, 30(44):14676–14684, 2010.

Zhou, L., Simon, J. B., Vardi, G., and Srebro, N. An agnostic view on the cost of overfitting in (kernel) ridge regression. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=YrTI2Zu0dd.

A. Extended Related Work

Compositional generalization. As modern machine learning models have become increasingly capable, generalizing to unseen situations has become an increasingly important challenge (Abbe et al., 2024; Press et al., 2023; Berglund et al., 2024; Fodor & Pylyshyn, 1988; Lake et al., 2017; Hupkes et al., 2020; Lake & Baroni, 2018; Johnson et al., 2017; Elmoznino et al., 2025; Lepori et al., 2023). Compositional or systematic generalization tests whether models are able to generalize to novel combinations of familiar components (Hupkes et al., 2020). Relational learning paradigms, in particular, test whether models can draw on their understanding of how these different components are related to generalize to novel combinations (Halford et al., 2010; Battaglia et al., 2018; Whittington et al., 2020; Webb et al., 2024). In deep neural networks, explicit, pre-specified constraints on the compositional operation can guarantee compositional generalization (Jarvis et al., 2023; Wiedemer et al., 2023; 2024; Brady et al., 2025; Schug et al., 2024); alternatively, these compositional principles can be meta-learned (Wu et al., 2023; Lake & Baroni, 2023; Redhardt et al., 2025) though see (Mitchell et al., 2021, though see). These studies usually consider tasks that follow an exact rule; we introduce a task paradigm that highlights the need for compositional generalization even in cases where explicit constraints would not work.

Kernel models. As described in Section 3.2, linear readout models trained with L_2-regularization ("ridge regression") can be understood in terms of the kernel induced by their representation. Ridge regression is both a common statistical learning method itself and captures the behavior of other popular learning algorithms. In particular, linear readout learning with gradient descent (from zero initialization) converges to the same solution (Soudry et al., 2018; Gunasekar et al., 2018; Ji et al., 2020). Moreover, an influential line of work in deep learning theory approximates neural networks by their first-order Taylor expansion (the "neural tangent kernel"). For large initial weights or wide neural networks, this approximation remains accurate throughout training, meaning that kernel models capture neural network learning exactly in this regime (Jacot et al., 2018; Chizat et al., 2019). As a result, kernel models have been a highly influential framework for theoretically understanding neural network learning (Canatar et al., 2021a;b; Mallinar et al., 2022; Zhou et al., 2024; Karkada et al., 2026), including in compositional generalization settings (Abbe et al., 2024; Lippl & Stachenfeld, 2025). In finetuning, in particular, weight changes are often very small, suggesting that kernel models might be a particularly relevant framework (Malladi et al., 2023; Afzal et al., 2026). Our findings support this, as our theory accurately captures finetuning behavior in pretrained models. In other contexts, inductive biases that are not captured by kernel models (and that our theory thus doesn't speak to) play an important role for generalization (Chizat & Bach, 2020; Woodworth et al., 2020; Vyas et al., 2023; Boix-Adsera et al., 2023; Tong & Pehlevan, 2025), including in finetuning (Lampinen & Ganguli, 2019; Kumar et al., 2022; Lippl & Lindsey, 2024; Anguita et al., 2026).

Transitive inference and transverse patterning. Transitive inference (TI) paradigms assess whether subjects are able to use the transitive rule to generalize learned relations. This is a canonical task for studying relational generalization in humans and animals (from monkeys and rats to wasps and fish) (Bryant & Trabasso, 1971; McGonigle & Chalmers, 1977; Davis, 1992; Grosenick et al., 2007; Tibbetts et al., 2019) (see also related task paradigms, e.g. (Zeithamova & Preston, 2010; Daw et al., 2011; Momennejad et al., 2017; Yang et al., 2024; Tong et al., 2026)). Transverse patterning is a common task for studying whether animals can learn nonlinear relations (Alvarado & Rudy, 1992; Astur & Sutherland, 1998; Dusek & Eichenbaum, 1998). TI has been used to assess relational generalization in neural networks (De Lillo et al., 2001; Nelli et al., 2023; Di Antonio et al., 2024; Lippl & Lindsey, 2024; Kay et al., 2024), including in-context learning in transformers and pretrained language models (PLMs) (Geerts et al., 2025; Liu et al., 2025; Wu & Deng, 2025). We expand on this work by studying relational generalization as a result of in-weight learning in PLMs.

Intransitive loops in relational orders. Intransitive loops commonly arise when a multi-dimensional comparison structure is projected into a low-dimensional space (Trybuła, 1961; Bar-Hillel & Margalit, 1988; Klimenko, 2015). A canonical example is inter-species competition, where complementary strengths and vulnerability might mean that A out-competes B, B out-competes C, and C out-competes A (May & Leonard, 1975; Gilpin, 1975; Buss, 1980; Sinervo & Lively, 1996; Huisman & Weissing, 1999; Kerr et al., 2002; Soliveres & Allan, 2018). Further, ranked-choice voting system potentially give rise to intransitive preference orderings (Arrow, 1950). In game theory, these concepts have been explored with "intransitive dice" (Jr., 1994; Conrey et al., 2016); along these lines, we expand on a prior analysis of choice intransitivity in poker (Bartholdi & Mikhailov, 2025). Overall, this highlights that potential intransitivity is ubiquitous in ordered (and at first glance transitive) relations in complex mathematical systems and the real world. Relatedly, (Pahikkala et al., 2010) previously developed a specific kernel method for learning intransitive relations. While they focused on empirically fitting benchmark datasets, we analytically characterize the out-of-distribution generalization performance arising from a very similar kernel approach that can be used to characterize the generalization behavior of a random features model.

Benign overfitting. The closest concept in classical in-distribution generalization to exceptions is label noise. While memorizing label noise does not have as clear benefits as memorizing exceptions, many modern machine learning algorithms nevertheless overfit their training data. Intriguingly, while classical perspectives suggest that overfitting makes generalization worse, recent work has identified certain regimes (including ridge regression with certain types of kernels) in which networks overfit their data without becoming worse at generalization ("benign overfitting") (Bartlett et al., 2020; Chatterji et al., 2022; Mallinar et al., 2022; Tsigler & Bartlett, 2023; Barzilai et al., 2026). Our work here emphasizes that "almost additive" representation exhibit benign memorization of exceptions, but only for no explicit regularization.

B. Encoding Model and Mechanistic Intuition

B.1. Encoding Model

We define a four-hot representation that represents any item $x_{j,k}$ as

$$z = \begin{pmatrix} s_1 z_j^T & s_1 z_k^T & s_2 z_{j,k}^T & s_0 z_b \end{pmatrix} \in \mathbb{R}^{2n+n^2+1}, \tag{S7}$$

where $z_j \in \mathbb{R}^n$ is the one-hot vector with an entry in the j-th position and $z_{j,k}$ is a one-hot vector with an entry in the $(j-1)n+k$-th position (or more intuitively, the one-hot vector representing each specific combination of items). Finally $z_b = 1$. For a representation $H \in \mathbb{R}^{n^2 \times h}$ of our pairs of items with associated k_d, k_o, k_s, we set

$$s_0 = \sqrt{k_d}, \quad s_1 = \sqrt{k_o - k_d}, \quad s_2 = \sqrt{(k_s - k_d) - 2(k_o - k_d)}. \tag{S8}$$

We write the complete feature matrix of z as

$$Z \in \mathbb{R}^{n^2 \times 2n+n^2+1} \tag{S9}$$

Then, HH^T, i.e. the kernel induced by H, is equivalent to the kernel induced by the four-hot representation. This means that we can rotate between these representations (note that this is a direct consequence of the fact that both matrices have identical left-singular vectors):

$$H = ZR, \quad Z = HR^T, \tag{S10}$$

where $R \in \mathbb{R}^{(n^2 \times 2n+n^2+1) \times h}$ is an orthonormal matrix. (Note that R is not square and we mean more specifically a matrix with orthogonal rows if $h \geq (n^2 \times 2n + n^2 + 1)$ and orthogonal columns if $h \leq (n^2 \times 2n + n^2 + 1)$.)

B.2. Mechanistic Intuition

We note that we can rewrite readouts from H, as readouts from Z:

$$f(x_{j,k}) = H_{j,k} w = Z_{j,k} \tilde{w} \text{ for } \tilde{w} := Rw. \tag{S11}$$

Because R is a rotation,

$$\|\tilde{w}\|_2^2 = \|w\|_2^2. \tag{S12}$$

We now split up $\tilde{w}$ into the ranking system, the conjunctive coefficient and the bias:

$$\tilde{w} = \begin{pmatrix} \tilde{r}^{(1)T} & \tilde{r}^{(2)T} & \tilde{t}^T & b \end{pmatrix}. \tag{S13}$$

We note that by the same symmetry arguments used in our proof, $\tilde{r}^{(1)} = -\tilde{r}^{(2)} =: \tilde{r}$ and $b = 0$. Thus, we can write

$$f(x_{j,k}) = s_1(\tilde{r}_j - \tilde{r}_k) + s_2 \tilde{t}_{j,k}. \tag{S14}$$

We now set

$$r_j := s_1 \tilde{r}_j, \quad t_{j,k} := s_2 \tilde{t}_{j,k}. \tag{S15}$$

Conveniently, this allows us to express the costs on w in terms of costs on the effective ranking system and the conjunctive coefficient:

$$\begin{aligned} \|w\|_2^2 = s_1^{-2}\|r\|_2^2 + s_2^{-2}\|t\|_2^2 &= \frac{1}{k_o - k_d}\|r\|_2^2 + \frac{1}{(k_s - k_d) - 2(k_o - k_d)}\|t\|_2^2 \\ &\propto \frac{k_s - k_d}{k_o - k_d}\|r\|_2^2 + \frac{k_s - k_d}{(k_s - k_d) - 2(k_o - k_d)}\|t\|_2^2. \end{aligned} \tag{S16}$$

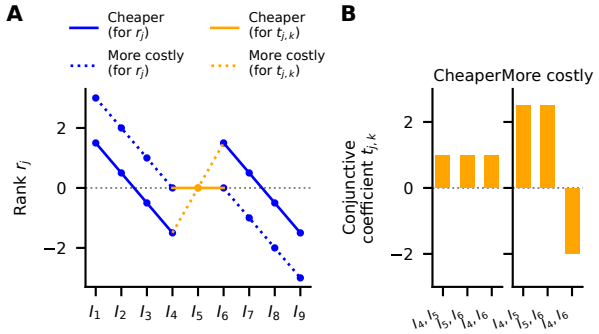

Figure S6. Illustration of the intuition. **A**, Two possible ranking systems, one of which is more costly and one of which is cheaper. **B** The values of the conjunctive coefficient, in order to encode the intransitive loop given the two different ranking systems. If we prioritize costs on the ranking system, we want to center the ranks of the section before and after the intransitive loop (solid blue line as opposed to dotted blue line). Thus, wanting to minimize the norm of the ranking system confers an inductive bias towards non-monotonicity. However, this means that the conjunctive coefficients need to counteract the ranking systems predictions, which incurs a substantially larger norm. To minimize the norm of the conjunctive coefficients, we want to keep the intransitive section flat (solid orange line as opposed to dotted orange line). This confers successful cross-section generalization. Because in practice we optimize a mixture of the ranking system and memorization cost, we always arrive at a non-monotonic ranking system, albeit a more attenuated one. A more compositional representations means that the conjunctive coefficient is more costly and hence the model prioritizes a flat ranking system. A more conjunctive representation means that the ranking system is more costly and hence the model prioritize a more non-monotonic ranking system.

This means that

$$\|w\| \propto C(r,t) := \frac{2}{1-\alpha}\|r\|_2^2 + \frac{1}{\alpha}\|t\|_2^2, \tag{S17}$$

which demonstrates that we can express the weight norm in terms of a relative weight on the ranking system and the conjunctive coefficients. This allows us to compare the relative incentives enforced by these different terms, as in Fig. S6 and Section 4.4.

C. Experiments with Pretrained Language Models

C.1. Detailed Setup

C.1.1. Task modalities.

We consider four task modalities: sports teams (either real or fake) and poker hands (either correct or reversed).

Sports teams. In this case, the relation is whether team A is likely to win against team B, presented as "Q: Will the [team A] win against the [team B]? A: [Yes/No]." To ensure that the models are biased to responding with Yes or No rather than a different format even before pretraining, we provide two "anchor questions" that indicate the format. To this end, we randomly sample a winning and a losing team (neither of which are a part of the actual hierarchy) and present them as additional preceding question. In total, this gives rise to the following format:

Q: Will the [anchor winner] win against [anchor loser]? A: Yes.
Q: Will the [anchor loser] win against [anchor winner]? A: No.
Q: Will the [team A] win against [team B]? A: [Yes/No].

We sampled teams from one of two lists. To minimize the degree of prior expectation on this relation, we considered sports teams with fictional names ("fake teams"). To control for any potential confusion arising from such fictional names, we additionally also considered real MLB teams ("real teams"). Below we list all names used in either of these modalities:

- **Fake teams:** The Harlem Renaissance, Marysville Mudskippers, Ogden Railspikes, Cheyenne Thunderbirds, Colorado Springs Quake, Fort Huachuca Javelina, Honolulu Pu'ali, Cambridge Charlies, Bethesda Warhawks, Grand Junction Dunes, Biloxi Shrimp, Ocean Springs Sandhill Crane, Gulfport Swampmen, Everett Yeti, McLean Majors, Morningside Midnight Run, Pawnee Fire, Carson City Cougars, Boise Noise, Silverton Slides, Durango Stampede, Ouray Morays,

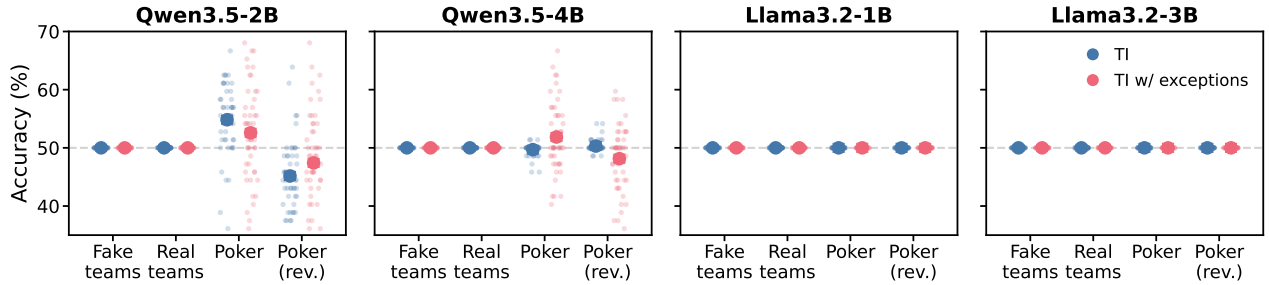

Figure S7. Accuracy for all models prior to finetuning for all four tasks. Line ranges represent 95% binomial confidence intervals (may be too small to see).

Munich Monks, Hamburg Raiders, Cologne Horribles, Mount Erie Field Devils, Waynesboro Knights, Oakland Visigoths, Gainesville Gators, Des Moines Shuckers

- **Real teams:** Boston Red Sox, New York Yankees, Toronto Blue Jays, Chicago White Sox, Cleveland Guardians, Detroit Tigers, Kansas City Royals, Minnesota Twins, Houston Astros, Los Angeles Angels, Oakland Athletics, Seattle Mariners, Texas Rangers, Atlanta Braves, Miami Marlins, New York Mets, Philadelphia Phillies, Washington Nationals, Chicago Cubs, Cincinnati Reds, Milwaukee Brewers, Pittsburgh Pirates, St. Louis Cardinals, Arizona Diamondbacks, Colorado Rockies, Los Angeles Dodgers, San Diego Padres, San Francisco Giants

Poker hands. The possible hands are any pair of two cards, represented by their ranks (written in order as A, K, Q, J, T, 9,..., 2) as well as whether they have matching suits (indicated by a "s") or not (indicated by a "o"). If the two ranks are matched, we know that their suits are not matched and we don't specify that further. For example an ace and a queen with different suits as represented as AQo. We note that this is standard poker terminology and we therefore expect that the language models are familiar with it. We consider as our relation whether hand A is more likely to win against hand B in a one-on-one match-up ("winningness") (see Appendix D). We present the relation in the following format:

We consider pre-flop all-in heads-up.
Q: Is AA likely to win against 72o? A: Yes.
Q: Is 72o likely to win against AA? A: No.
Q: Is [hand A] likely to win against [hand B]? A: [Yes/No].

Unlike for the sports team, we use the ground truth (according to an equity matrix we generated using `eval7`, see Appendix D). We either consider that relation directly, or invert it (i.e. such that worse hands are indicated as winning against better hands, "poker (reversed)"). This allows us to evaluate the degree to which real-world information influences the pretrained language models' judgment. To sample the relevant relations, we specifically sample cases where the winning hand has a probability between 51% and 60% of winning against the losing hand, so as to make the task sufficiently challenging. For TI, we make sure to sample an actually transitive hierarchy, checking each pairwise relation. In contrast, for TI with exceptions, we only control for the training set pairs. We sample these sets by first sampling a particular hand and then sequentially generating the next hand such that it satisifes the constraints of the hierarchy. If no hand satisfies those constraints, we consider a different sample of starting hand and start again. We confirmed that models had some (but not much prior knowledge) about the poker hands (Fig. S7).

Task hierarchies We consider four task hierarchies: TI ($n = 9$) and TI with exceptions ($n = 9, p = 6, q = 4$, $n = 11, p = 7, q = 5$, and $n = 11, p = 8, q = 4$). In total, this means that we consider sixteen different tasks (4 task hierarchies combined with four task modalities).

Evaluation We consider $n = 100$ randomly generated datasets for the real and fake sports teams and $n = 50$ randomly generated datasets for the regular and reversed Poker hierarchies. To evaluate the models, we both compute the probability on each training and generalization pair that the assign on "yes" vs. "no." For the cases where we have a ground truth, we additionally determine whether the model was correct (i.e. assigned a higher probability to the correct answer). We compute accuracies by averaging over all random seeds.

C.1.2. Model setup

We consider three finetuning strategies: full finetuning, low-rank adaptation (LoRA), and linear probing. In all cases, we use AdamW (Loshchilov & Hutter, 2019) with parameters $\beta_1 = 0.9, \beta_2 = 0.999$ and a weight decay of 10^{-2} (as is default in the PyTorch implementation). We emphasize that our objective here was not to find the hyperparameters that worked best, but rather to use representative parameters to understand the default behavior arising from finetuning. We train on crossentropy over the yes/no tokens specifically and do not use the rest of presented sequence as part of our objective.

We consider four recent pretrained language models: Llama3.2-1B/3B and Qwen3.5-2B/4B. We only apply full finetuning to the smaller two models, giving rise to ten total finetuning setups.

Full finetuning. We finetune all weights in the model, with a learning rate of 10^{-6} (for Qwen3.5-2B) and $2 \cdot 10^{-6}$ (for Llama3.2-1B). We use a batch size of 32 and finetune for 100 epochs.

Low-rank adaptation (LoRA). We apply LoRA to all parameters in the attention and MLP layers. We separately consider ranks $r = 4, 8, 16, 32, 64$ and set $\alpha = 2r$. By default, we consider $r = 32$, but we analyze the effect of different ranks below. We use a batch size of 16 and finetune for 10 epochs.

Linear probing. We only train the linear readout head of the model, using L_2-regularization with respect to the readout weights. For the inverse scaling factor of the regularization c, we use the values $c = 10^4, 10, 1, 0.1, 0.05, 0.02$. By default we consider extremely small regularization, $c = 10^4$. We use a batch size of 16 and train for 50 epochs.

C.1.3. Reproducibility

The code for reproducing the experiments described in Section 5 can be found here: `https://github.com/sflippl/ti-with-exceptions`. We ran all experiments on a Slurm cluster using A40 GPUs. Each experiment (which involved running 100 seeds) took between 2–6h. We ran a total of 464 experiments requiring between 1000-2700h of compute in total.

C.2. Transitive Inference

We first trained models on transitive inference ($n = 9$). To evaluate model behavior on this task, we draw a distinction between three types of pairs: adjacent items (which the model was trained on), non-adjacent items that include a terminal item ("Gen. (terminal)") and non-adjacent items that only include internal items ("Gen. (internal)"). This is motivated by the fact that the former generalization set can be solved by simply learning to always indicate that the first item is larger and that the last item is smaller, without requiring actually relating the items to each other. In contrast, internal items are necessarily context-dependent in their relation. Thus, transitive generalization is more accurately captured by the ability to transitively generalize on the internal items.

Full finetuning yielded successful transitive generalization. Figure S8 depicts the accuracy of models fully finetuned on TI (further, see Fig. S11A, top row, for summarized performance). We find that the models broadly generalize above chance across all task modalities.

LoRA. We found that low-rank adaptation largely yielded more modest gains in performance. For most cases, transitive generalization (including on the internal items) was still above chance, however. We note that LoRA performed particularly badly on Llama3.2-1B. It is possible that further hyperparameter optimization or longer training would yield stronger benefits from LoRA.

We then evaluated the impact of varying rank on generalization (Fig. S11B). We found that intermediate ranks (16 or 32) tended to perform best. However, for no parameter choices did the models exhibit systematic errors (i.e. below-chance accuracy). This is consistent with the fact that kernel models also never make such systematic errors.

Linear probing. Finally, we again observed strong transitive generalization for linear probing (Fig. S11C). We found that stronger generalization decreased performance on both training and generalization. Again, model performance never went below chance.

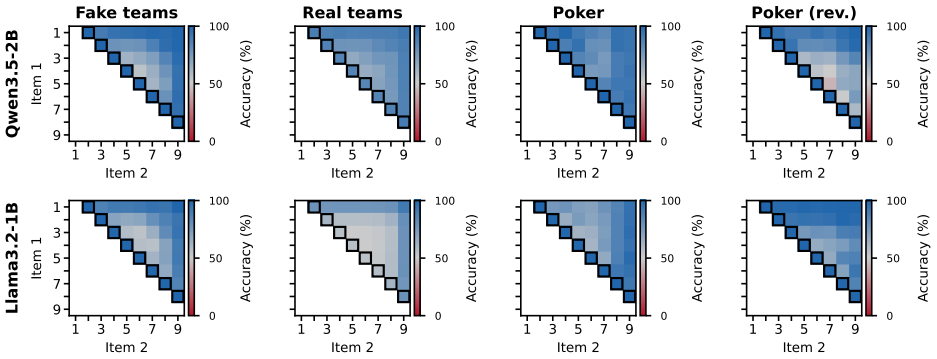

Figure S8. Accuracy for all models and tasks for full finetuning.

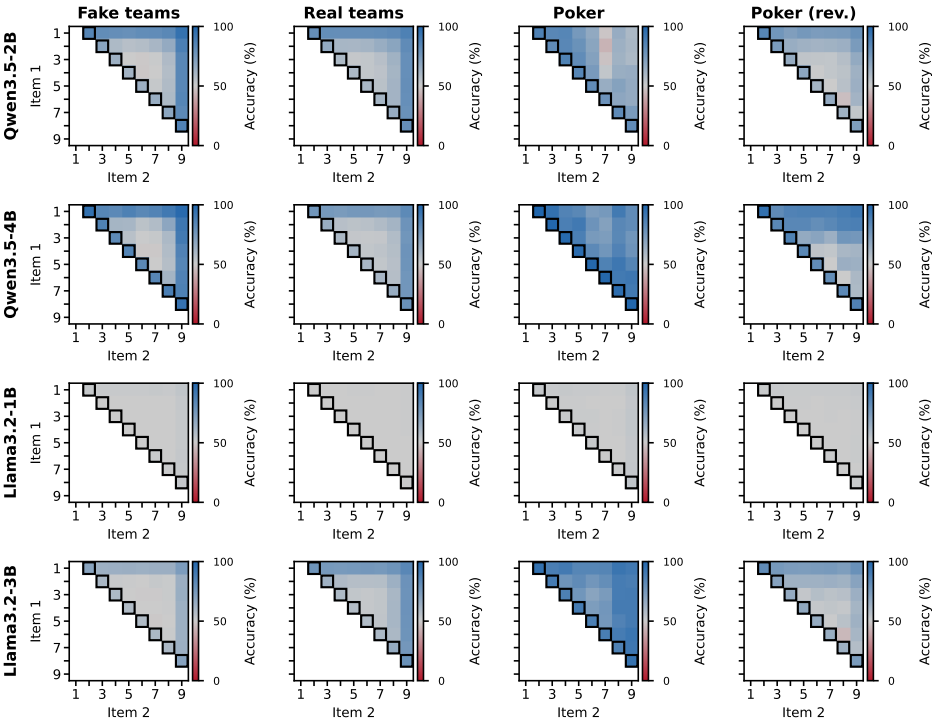

Figure S9. Accuracy for all models finetuned with LoRA (rank=32).

C.3. Consistent behavioral effects on transitive inference

On both full finetuning (Fig. S8) and linear probing (Fig. S10), we observed that generalization on the sports teams becomes better as a function of the symbolic distance between the two items (i.e. the distance of their ranks). Moreover, the models are also better at relatively more terminal items than more internal items. Notably, all of these behavioral effects parallel effects robustly observed in humans and animals as well (Vasconcelos, 2008; Jensen, 2017). On the poker task modalities, behavior in the case of full finetuning looks a little different and models are better at generalizing for later items in the hierarchy (or, identically, earlier items in the reversed hierarchy). We speculate that this is a consequence of the prior knowledge the language models have about poker. In contrast, for linear probing, we still observe the same behavioral effects for the poker modality, indicating that prior knowledge affects linear probing differently from full finetuning.

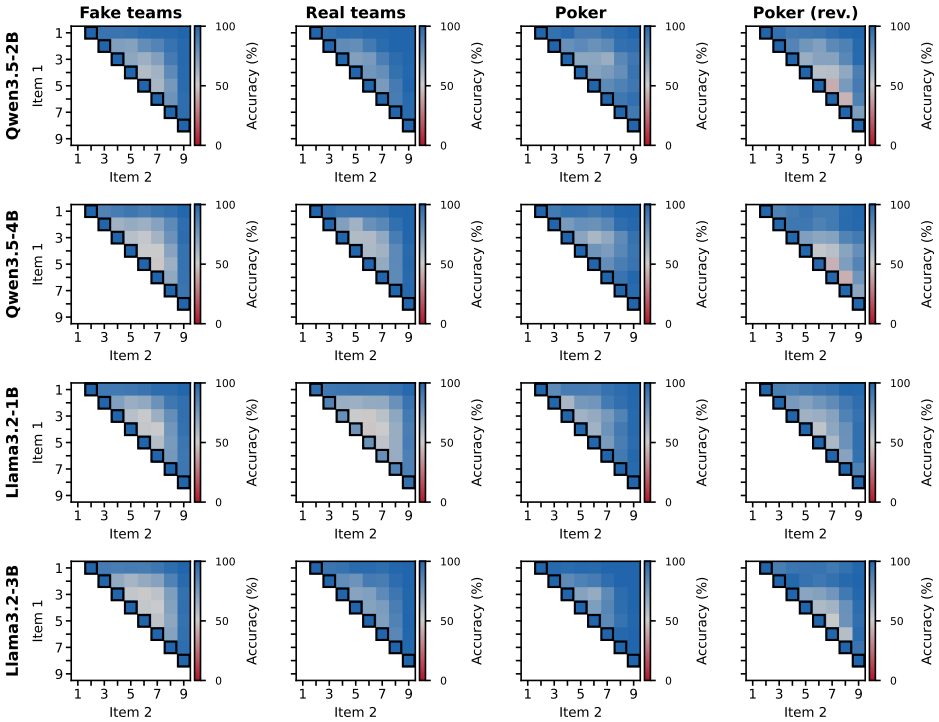

Figure S10. Accuracy for all models finetuned with the linear probe ($c = 10^4$).

C.4. Transitive inference with exceptions (base case)

To complement Fig. 5D in the main text, Figs. S12,S13,S14 depict generalization performance across the different finetuning methods, task modalities, and models. As indicated by Fig. 5 within-section generalization is highly reliable, whereas cross-section generalization is variable and sometimes even below chance. We again observe that LoRA performs much worse on the task. We summarize the overall performance in Fig. S15, which shows, in particular, that models are substantially worse at cross-section generalization than within-section generalization.

To complement Fig. 5E, we find that across all models and task modalities, training performance on I_4 vs. I_5 and I_5 vs. I_6 is lower than on I_4 vs. I_6, as predicted by our theory (Fig. S16). Moreover only the former two data points are ever below chance—again as predicted by our theory. We note that on the linear probe, the regularization coefficient behaves highly similar to the predictions made by the kernel model: as regularization becomes stronger, performance on the training data points becomes worse. (This is perhaps not too surprising given the close analogy between these models.) In contrast, rank in LoRA does not have such a consistent effect. Rather, performance often becomes better for lower rank. Thus, the effects of varying the rank in LoRA are not captured by our theoretical model. However, we emphasize that the coarse trends we highlighted above still hold true.

Finally, to complement Fig. 5F, we find that across all models and task modalities, generalization performance within-section is better than cross-section (Fig. S16). Moreover, only cross-section generalization sometimes goes below chance. On linear probing, c generally has the effect predicted by our theory: both within-section and cross-section generalization performance decreases for stronger regularization and for cross-section generalization, performance can even intermittently go below chance.

C.5. Varying task parameters

To expand on Fig. 5G,H, Figs. S18 and S19 depict the generalization behavior on the two additional task variants.

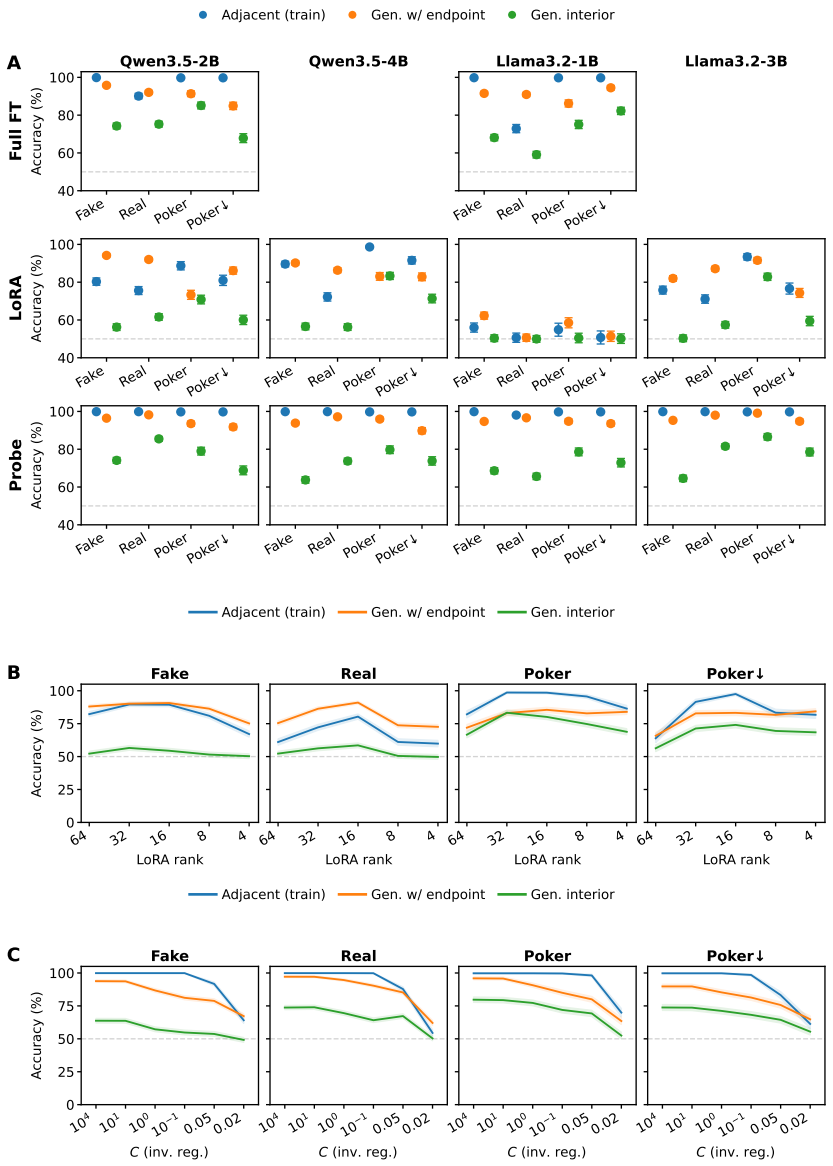

Figure S11. Summary of behavior on TI. Shaded regions/line ranges represent 95% binomial confidence intervals (may be too small to see). **A**, Accuracy before and after finetuning across all tasks and for different finetuning methods. **B**, Accuracy on LoRA as a function of rank. **c**, Accuracy on the linear probe as a function of the regularization coefficient. **C**, Accuracy as a function of symbolic distance.

C.6. Content effects in the poker modality

Humans and PLMs seem to draw logical inferences that are consistent with their prior knowledge more easily, than ones that are inconsistent (Evans et al., 1983; Evans & Perry, 1995; Dasgupta et al., 2022; Lampinen et al., 2024). We generally found that this was true for the PLMs finetuned on TI as well: models trained on poker tended to perform better than models trained on the reverted version of poker (Fig. S11A). A surprising exception is given by Llama3.2-1B, which showed the opposite effect and performed better in its generalization on the reversed poker hierarchy. For all other finetuning setups, however, we consistently found that models were worse at generalizing over the reversed poker hierarchy than the non-reversed poker hierarchy. Overall, we therefore found that models, in almost all cases of TI, showed a content effect. In contrast, we did not consistently observe a content effect with TI with exceptions. This highlights that introducing more complicated relational or logical rules that potentially contain exceptions as well may differentially affect content effects in PLMs.

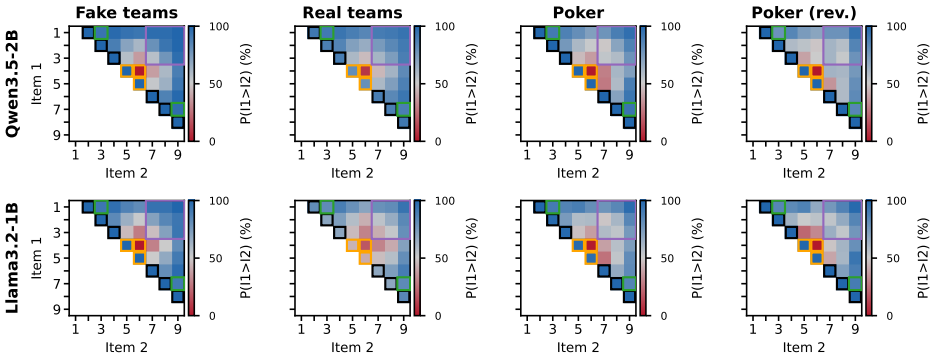

Figure S12. Model predictions for all tasks and models for full finetuning.

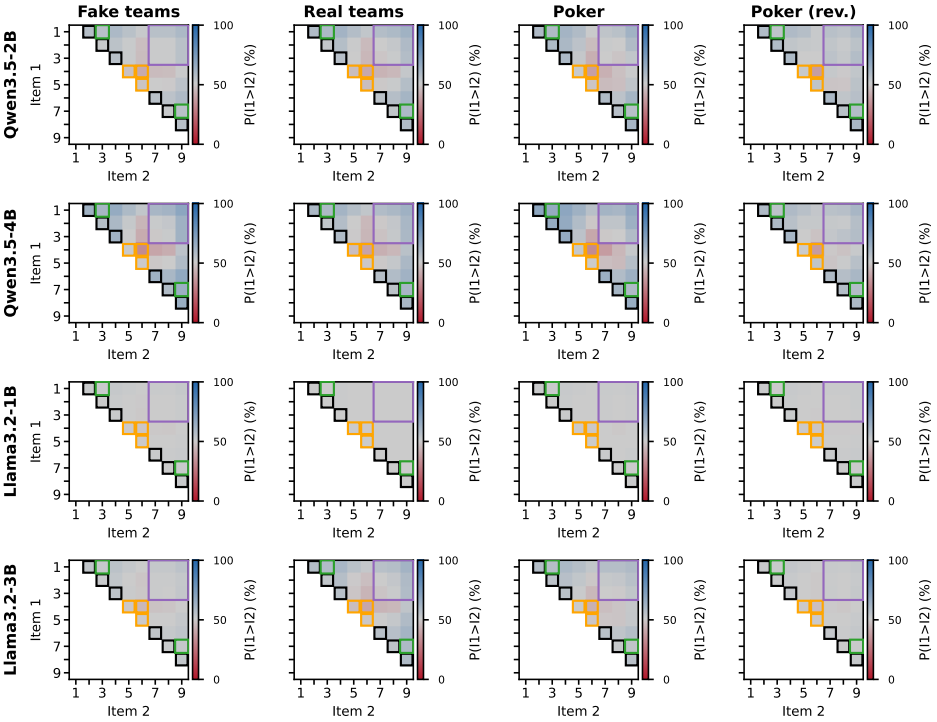

Figure S13. Model predictions for all tasks and models for LoRA (rank=32).

C.7. Open questions

Overall, we observed that our theoretical model provided a striking account of the finetuning behavior of pretrained language models. Below, we expand on two questions our empirical experiments have raised, that our theory cannot address.

How does prior knowledge impact transitive generalization? In practice, relational generalization will interact with prior knowledge. In particular, on TI we observed that models performed better on the poker hierarchy than the reversed poker hierarchy (see Section C.6). Our theory currently cannot account for such effects of prior knowledge: deviating from the exchangeability assumption might allow us to address this point. This could potentially help us better understand why these content effects appear to be less salient for TI with exceptions.

How can we understand the role of rank for relational generalization through LoRA? We found that the rank in low-rank adaptation played a different role from the regularization coefficient in ridge regression (though both are commonly

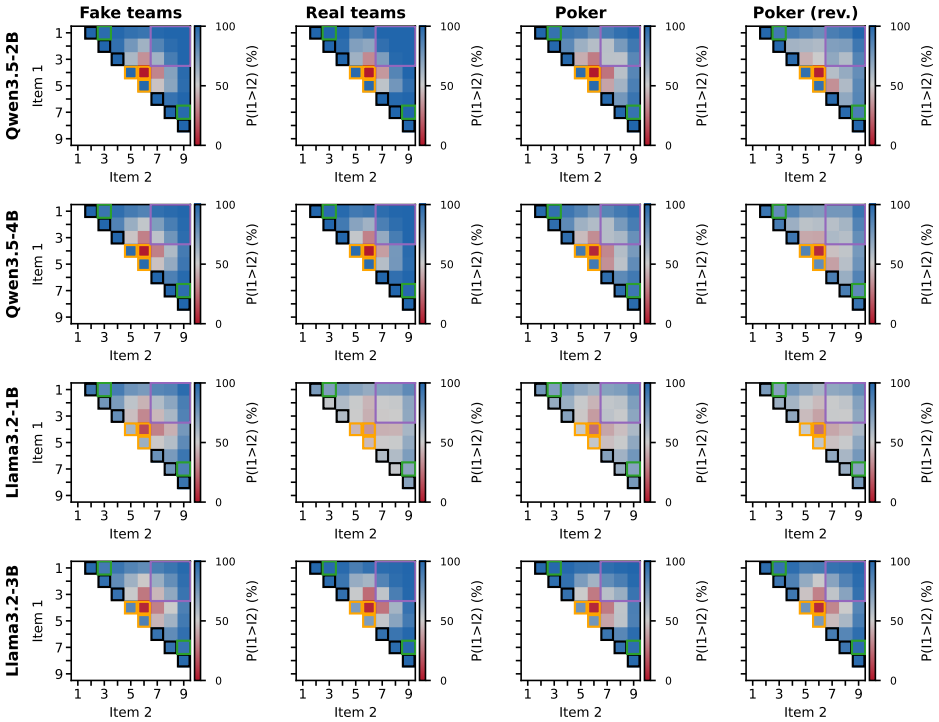

Figure S14. Model predictions for all tasks and models for linear probing ($c = 10^4$).

understood to regulate the amount of changes induced by the network). This highlights that a different theoretical paradigm may be necessary to help us more closely understand the behavior of LoRA on relational tasks. However, we note that our theory nevertheless captured the coarse-grained structure of behavior in LoRA.

D. An Example of an Intransitive Relation: Winningness of Poker Hands

To provide an illustration of an intransitive relation, we consider Texas Hold'em poker. Specifically, we consider the relation where for two hands, $x \succ y$ iff x is more likely to win against y in a heads-up match. We computed the equity of all match-ups between hands using `eval7` and then sampled sets of hands that adhered to the structure of TI with exceptions ($n = 9, p = 6, q = 4$). We then computed the frequency with which $x \succ y$ for any pair of hands (Fig. S20). We found that in most cases, within-section and cross-section generalization would result in the appropriate generalization in this case.

E. Proof of the main theorem

We first state the full version of the theorem:

Theorem 2 (Theorem 1 (detailed).). *On TI with exceptions with parameters n, p, q and the training dataset $\mathcal{D}$, the predictions of a kernel model with an exchangeable representation are given by*

$$f(x_{j,k}) = \begin{cases} r_j(\alpha, \tilde{c}) - r_k(\alpha, \tilde{c}) & \text{for } x \notin \mathcal{D}, \\ m(\alpha, \tilde{c}) y_{i,j} + (1 - m(\alpha, \tilde{c}))(r_i(\alpha, \tilde{c}) - r_j(\alpha, \tilde{c})) & \text{for } x \in \mathcal{D}, \end{cases} \tag{S18}$$

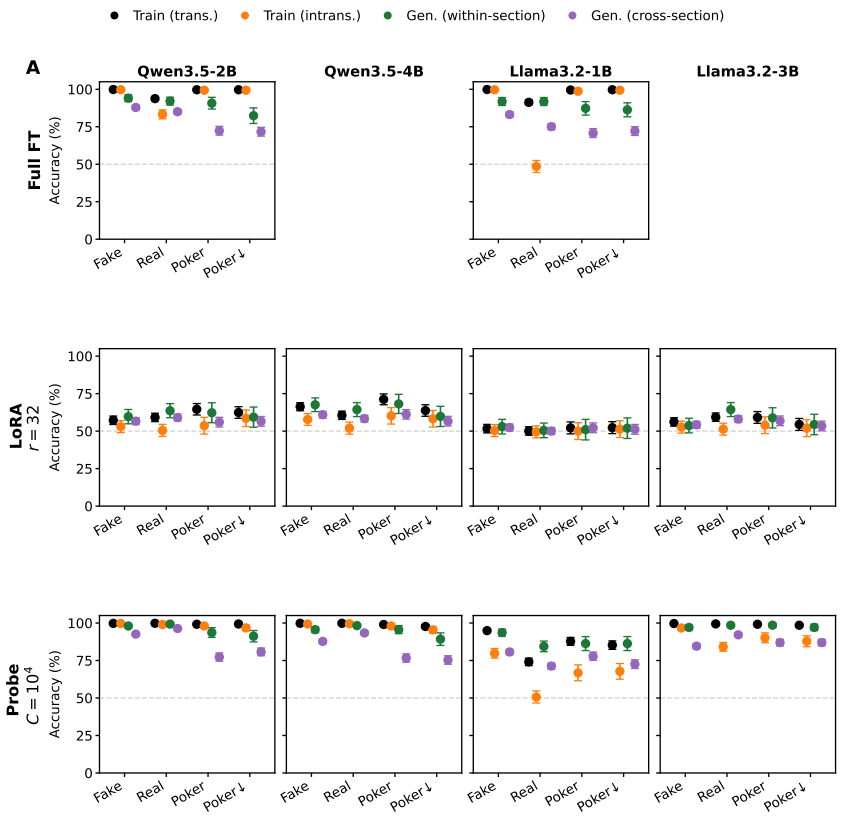

Figure S15. Accuracy on the different training and generalization splits, across tasks, models, and finetuning methods. Line ranges represent 95% binomial confidence intervals (may be too small to see).

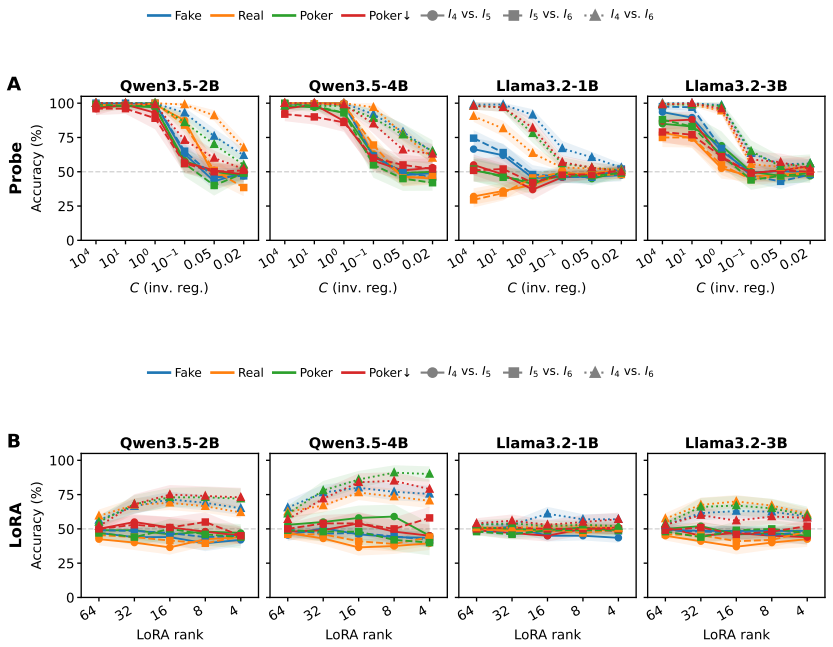

Figure S16. Same plot as Fig. 5E across different task modalities, models, and finetuning methods. Specifically, we vary c (for the linear probe) and we vary the rank (for LoRA). We then compute accuracy for the three distinct intransitive pairs. Shaded regions represent 95% binomial confidence intervals (may be too small to see).

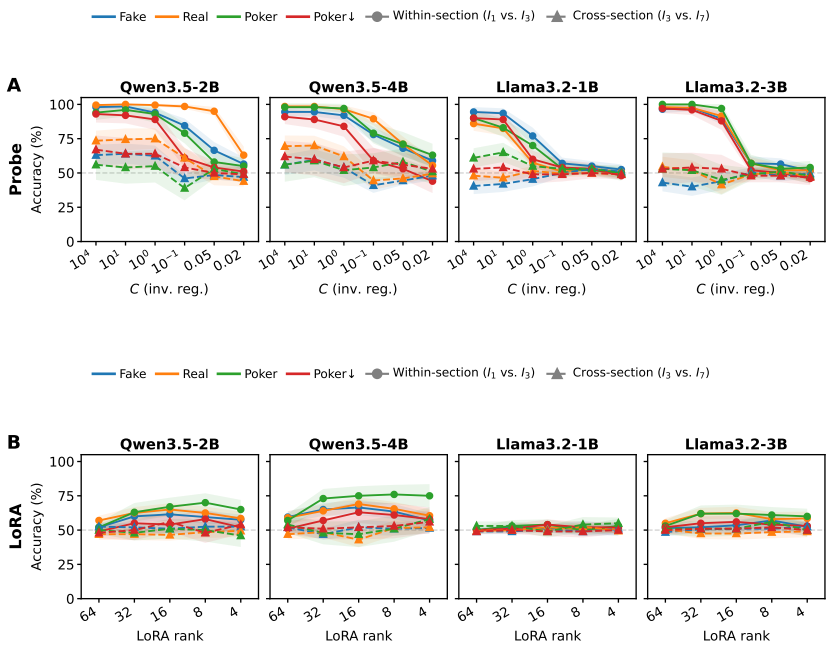

Figure S17. Same plot as Fig. 5F across different task modalities, models, and finetuning methods. Specifically, we vary c (for the linear probe) and we vary the rank (for LoRA). We then compute accuracy for the within-section generalization pair and the most challenging cross-section generalization pair. Shaded regions represent 95% binomial confidence intervals (may be too small to see).

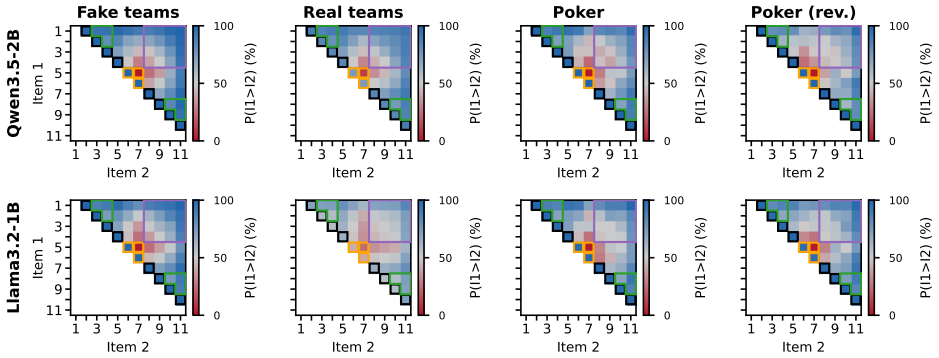

Figure S18. Accuracy on full finetuning for more transitive items.

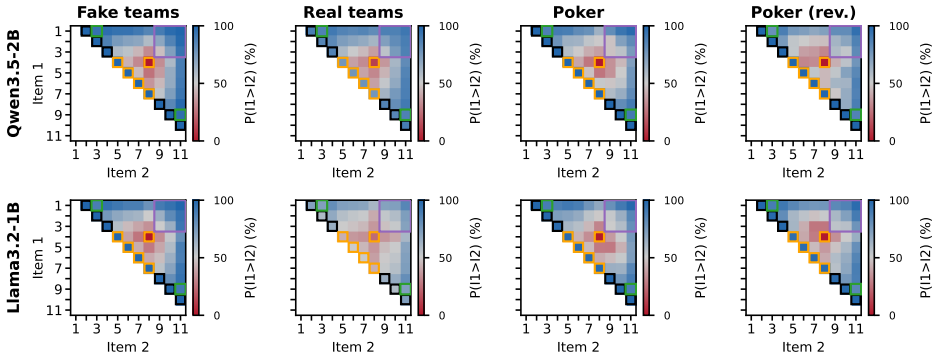

Figure S19. Accuracy on full finetuning for more intransitive items.

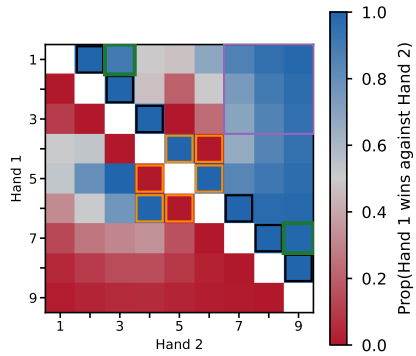

Figure S20. Illustration of TI with exceptions in a real-world example. We sampled sets of hands in poker such that the winningness relation (i.e. for two hands, $x \succ y$ iff x is more likely to win against y) adhered to the training set in TI with exceptions. Under those constraints, we determined the proportion with which hand 1 (represented along the row) is more likely to win against hand 2. This analysis highlights that both cross-section generalization and within-section generalization would, on average result in the correct generalization.

where

$$m(\alpha, \tilde{c}) := \tfrac{\alpha}{\alpha+\tilde{c}^{-1}}, \quad r_j(\alpha, \tilde{c}) := r_j^{\mathrm{TI}}(\alpha, \tilde{c}) + r_j^{\mathrm{pert}}(\alpha, \tilde{c}), \tag{S19}$$

$$r_j^{\mathrm{TI}}(\alpha, \tilde{c}) := \frac{\sinh((\frac{n+1}{2} - j)\lambda)}{\sinh(\frac{n+1}{2}\lambda) - \sinh(\frac{n-1}{2}\lambda)}, \quad \lambda := \operatorname{arccosh}\left(\frac{1+\tilde{c}^{-1}}{1-\alpha}\right), \tag{S20}$$

$$r_j^{\mathrm{pert}}(\alpha, \tilde{c}) := \frac{\left(1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}}\right)\left(\tilde{D}_{jp} - \tilde{D}_{jq}\right)}{\sinh(\lambda)\sinh(n\lambda) - 2\sinh\left(\frac{q-p}{2}\lambda\right) G}, \tag{S21}$$

$$G := \cosh\left(\left(q - \tfrac{1}{2}\right)\lambda\right)\sinh\left(\left(n - \tfrac{p+q-1}{2}\right)\lambda\right) + \cosh\left(\left(n - p + \tfrac{1}{2}\right)\lambda\right)\sinh\left(\tfrac{p+q-1}{2}\lambda\right), \tag{S22}$$

$$\tilde{D}_{ij} := \cosh\big((\min(i,j) - \tfrac{1}{2})\lambda\big)\cosh\big((n - \max(i,j) + \tfrac{1}{2})\lambda\big). \tag{S23}$$

We note that r_j^{TI} is the ranking system for TI without an exception and hence r_j^{pert} represents the perturbation on top of it. The above definition is only valid for $\alpha < 1$. For $\alpha = 1$,

$$r_j(\alpha, \tilde{c}) = 0. \tag{S24}$$

E.1. Prerequisites

Theorem 3 (Hu & O'Connell (Hu & O'Connell, 1996))**.** *Let $a, b \in \mathbb{R}$ such that $\frac{a}{b} \leq -2$ and*

$$B \in \mathbb{R}^{m\times m}, \quad B_{ij} = \begin{cases} a & \text{if } i = j, \\ b & \text{if } |i-j| = 1, \\ 0 & \text{else.} \end{cases} \tag{S25}$$

Then

$$B_{ij}^{-1} = -\frac{\cosh\left((m+1-|j-i|)\lambda\right) - \cosh\left((m+1-i-j)\lambda\right)}{2b\sinh(\lambda)\sinh\left((m+1)\lambda\right)}, \tag{S26}$$

where

$$\lambda := \operatorname{arccosh}\left(-\frac{a}{2b}\right). \tag{S27}$$

E.2. Setup

We consider $n \in \mathbb{N}$ items $(x_j)_{i=1,\dots,n}$, $x^{(i)} \in \mathbb{R}^d$. The task presents two of those items together and we denote this by

$$x_{(j,k)} = \begin{pmatrix} x_j \\ x_k \end{pmatrix} \in \mathbb{R}^{2d}. \tag{S28}$$

The corresponding label is given by

$$y_{j,k} = \begin{cases} +1 & \text{if } (j,k) = (p,q), \\ -1 & \text{if } (j,k) = (q,p), \\ +1 & \text{else, if } j < k, \\ -1 & \text{else, if } j > k, \end{cases} \tag{S29}$$

where (p, q) with $1 \leq q < p \leq n$ and $p - q \geq 2$ denotes the designated *exception pair*. Since $q < p$, the transitive rule would assign $y_{p,q} = -1$ and $y_{q,p} = +1$; the exception flips both, making this pair inconsistent with the underlying ranking.

The training dataset consists of all adjacent items together with the exception pair, i.e.

$$\mathcal{D} = \underbrace{\big\{ \big(x_{j,j+1}, 1\big), \big(x_{j+1,j}, -1\big) \,\big|\, j = 1, \ldots, n-1 \big\}}_{\text{adjacent pairs}} \cup \underbrace{\big\{ \big(x_{p,q}, +1\big), \big(x_{q,p}, -1\big) \big\}}_{\text{exception pair}}. \tag{S30}$$

In matrix form, we denote this dataset as

$$X = \begin{pmatrix} x_{1,2} \\ \vdots \\ x_{n-1,n} \\ \hline x_{2,1} \\ \vdots \\ x_{n,n-1} \\ \hline x_{p,q} \\ x_{q,p} \end{pmatrix} \in \mathbb{R}^{(2n)\times 2d}, \quad y = \begin{pmatrix} +1 \\ \vdots \\ +1 \\ \hline -1 \\ \vdots \\ -1 \\ \hline +1 \\ -1 \end{pmatrix} \in \mathbb{R}^{2n}. \tag{S31}$$

We consider a set of features

$$\phi : \mathbb{R}^{2d} \to \mathbb{R}^h, \quad h \in \mathbb{N}, \tag{S32}$$

and a linear model of those latent features that predicts the output, i.e.

$$\hat{y}_{i,j} = \langle w, \phi(x_{i,j}) \rangle. \tag{S33}$$

E.3. General intuition

Many standard statistical learning models (in particular the ones we consider below) only depend on their input representation through its trial-by-trial similarity (or "induced kernel")

$$K(x, x') = \langle \phi(x), \phi(x') \rangle. \tag{S34}$$

In particular, the ridge regression solution (2) can be described in terms of the "dual coefficients"

$$a = (K + \tfrac{1}{c} I)^{-1} y, \tag{S35}$$

where K is the trial-by-trial similarity of the training dataset:

$$K = \phi(X)^T \phi(X) \in \mathbb{R}^{2n \times 2n}. \tag{S36}$$

Each dual coefficient corresponds to a particular training data point (j, k) and we denote it by $a_{(j,k)}$. The dataset now contains $2n$ training points: the $2(n-1)$ adjacent pairs plus the two exception pairs (p, q) and (q, p). The model behavior on a new data point $x_{j,k}$ is then given by

$$f(x_{j,k}) = \langle k^{(\text{test})}_{(j,k)}, a \rangle, \quad k^{(\text{test})}_{(j,k)} := \phi(X)^T \phi(x_{j,k}). \tag{S37}$$

Importantly, the similarity between test data points and training data points can only take on two possible values: κ_o and κ_d. The training points whose similarity to (j, k) equals κ_o are exactly those that share an item with (j, k). Among the adjacent pairs, these are $a_{(j,j+1)}$, $a_{(j,j-1)}$, $a_{(k-1,k)}$, and $a_{(k+1,k)}$. The exception pairs $a_{(p,q)}$ and $a_{(q,p)}$ additionally contribute with

similarity κ_o whenever j or k coincides with p or q. (If $j, k \in \{1, n\}$, some adjacent-pair coefficients will not exist; we set them to zero. For example, if $i = 0$, $a_{(i,i-1)} = a_{(1,0)}$ corresponds to a non-existent data point and is set to zero.) All other training points have similarity κ_d. Taken together, we can express model behavior as

$$\begin{aligned} f(x_{j,k}) = \kappa_d \sum a &+ (\kappa_o - \kappa_d)\big(a_{(j,j+1)} + a_{(j,j-1)} + a_{(k-1,k)} + a_{(k+1,k)}\big) \\ &+ (\kappa_o - \kappa_d)\big((\delta_{j=p} + \delta_{k=q})a_{(p,q)} + \delta_{j=q} + \delta_{k=p})a_{(q,p)}\big). \end{aligned} \tag{S38}$$

The first line is identical to the no-exception case: a sum of values dependent on i and values dependent on j, which already implements a ranking system. The second line is the exception correction: it activates only when the test pair shares an item with the exception pair, and it too decomposes additively into an j-dependent and k-dependent contribution. We can therefore define a generalized ranking system

$$r_1(j) = (\kappa_o - \kappa_d)\big(a_{(j,j+1)} + a_{(j,j-1)}\big) + (\kappa_o - \kappa_d)\,\delta_{j=p}a_{(p,q)} + \delta_{j=q}a_{(q,p)}, \tag{S39}$$

$$r_2(k) = -(\kappa_o - \kappa_d)\big(a_{(k-1,k)} + a_{(k+1,k)}\big) - (\kappa_o - \kappa_d)\,\delta_{k=q}a_{(p,q)} + \delta_{k=p}a_{(q,p)}, \tag{S40}$$

yielding

$$f(x_{j,k}) = r_1(j) - r_2(k). \tag{S41}$$

To see whether the model generalizes, we need to determine whether the ranks are ordered correctly. To do so, we analytically solve the inverse problem Eq. (S35). This is described in detail in the next section. As a brief summary, we rely on previous results on inverses for banded diagonal matrices (Hu & O'Connell, 1996) to compute the inverse matrix. This inverse is expressed in terms of hyperbolic functions. We then use hyperbolic identities in a few different ways to compute a and, finally, r_1 and r_2 from this. This reveals that $r_1(i) = r_2(i)$ and further gives rise to the analytical expression provided in the main text.

E.4. Expression of the dual solution.

Here we assume that the weights are learned through ridge regression. Specifically, our loss function is given by the mean squared error over the training set:

$$\begin{aligned} L(w) &= \sum_{(x,y)\in\mathcal{D}} \ell\big(y, \langle w, \phi(x)\rangle\big) \\ &= \sum_{j=1}^{n-1} \ell\big(1, \langle w, \phi(x_{j,j+1})\rangle\big) + \sum_{j=1}^{n-1} \ell\big(-1, \langle w, \phi(x_{j+1,j})\rangle\big) \\ &\quad + \ell\big(1, \langle w, \phi(x_{p,q})\rangle\big) + \ell\big(-1, \langle w, \phi(x_{q,p})\rangle\big)\,. \end{aligned} \tag{S42}$$

where $\ell(y, \hat{y}) = (y - \hat{y})^2$. The weights w are then determined by minimizing the optimization problem

$$L(w) + \tfrac{1}{c}\|w\|_2^2, \tag{S43}$$

where c is the inverse strength of the regularization. The corresponding dual problem is given by

$$a := \big(K + \tfrac{1}{c}I\big)^{-1} y, \tag{S44}$$

where $K = \phi(X)\phi(X)^T$ is the cross-sample similarity. Writing a as

$$a = \begin{pmatrix} b^{(1)} \\ b^{(2)} \\ h^{(1)} \\ h^{(2)} \end{pmatrix}, \quad b^{(1)} \in \mathbb{R}^{n-1},\ b^{(2)} \in \mathbb{R}^{n-1}, h^{(1)} \in \mathbb{R}, h^{(2)} \in \mathbb{R} \tag{S45}$$

we can write $(K + \frac{1}{c}I)^{-1}a = y$ as

$$\begin{aligned}
(\delta_s + \tfrac{1}{c})b_i^{(1)} + \delta_o\big(b_{i-1}^{(2)} + b_{i+1}^{(2)}\big) + \delta_o\big(h^{(1)}(\delta_{(i=p)} + \delta_{(i+1=q)})\big) + \delta_o\big(h^{(2)}(\delta_{(i=q)} + \delta_{(i+1=p)})\big) + \kappa_d\langle a\rangle &= 1,\\
(\delta_s + \tfrac{1}{c})b_i^{(2)} + \delta_o\big(b_{i-1}^{(1)} + b_{i+1}^{(1)}\big) + \delta_o\big(h^{(2)}(\delta_{(i=p)} + \delta_{(i+1=q)})\big) + \delta_o\big(h^{(1)}(\delta_{(i=q)} + \delta_{(i+1=p)})\big) + \kappa_d\langle a\rangle &= -1,\\
(\delta_s + \tfrac{1}{c})h^{(1)} + \delta_o\big(b_{q-1}^{(1)} + b_p^{(1)}\big) + \delta_o\big(b_q^{(2)} + b_{p-1}^{(2)}\big) + \kappa_d\langle a\rangle &= 1,\\
(\delta_s + \tfrac{1}{c})h^{(2)} + \delta_o\big(b_{q-1}^{(2)} + b_p^{(2)}\big) + \delta_o\big(b_q^{(1)} + b_{p-1}^{(1)}\big) + \kappa_d\langle a\rangle &= -1,
\end{aligned} \tag{S46}$$

where we define $\delta_s = \kappa_s - \kappa_d$, $\delta_o = \kappa_o - \kappa_d$, $\langle a\rangle := \sum_{i=1}^{2n} a_i$, and for notational simplicity, $b_0^{(j)} = b_n^{(j)} = 0$. This means that if $b^{(1)}, b^{(2)}, h^{(1)}, h^{(2)}$ are a solution to the equation, then so are

$$\tilde{b}^{(1)} := -b^{(2)}, \quad \tilde{b}^{(2)} := -b^{(1)}, \qquad \tilde{h}^{(2)} := -h^{(1)}, \quad \tilde{h}^{(2)} := -h^{(1)}. \tag{S47}$$

Since (S44) is well-defined, the solution b must be unique and we can infer

$$b^{(1)} = -b^{(2)}, \qquad h^{(1)} = -h^{(2)}. \tag{S48}$$

In particular, this means that

$$\langle a\rangle = 0. \tag{S49}$$

We thus define

$$\bar{b} := b^{(1)} = -b^{(2)}, \qquad \bar{h} := h^{(1)} = -h^{(2)}, \tag{S50}$$

Setting

$$\tilde{c} := c\delta_s, \tag{S51}$$

(S46) yields

$$\begin{aligned}
\delta_s(1 + \tilde{c}^{-1})\bar{b}_i - \delta_o\big(\bar{b}_{i-1} + \bar{b}_{i+1}\big) + \delta_o\big(\bar{h}(\delta_{(i=p)} + \delta_{(i+1=q)})\big) - \delta_o\big(\bar{h}(\delta_{(i=q)} + \delta_{(i+1=p)})\big) &= 1\\
\delta_s(1 + \tilde{c}^{-1})\bar{h} + \delta_o\big(\bar{b}_{q-1} + \bar{b}_p\big) - \delta_o\big(\bar{b}_q + \bar{b}_{p-1}\big) &= 1,
\end{aligned} \tag{S52}$$

as $\langle a\rangle = \langle b^{(1)}\rangle + \langle b^{(2)}\rangle + h^{(1)} + h^{(2)} = 0$ and therefore the last summand vanishes.

As $\frac{\delta_o}{\delta_s} = \frac{1-\alpha}{2}$, setting $\bar{b}$ in matrix form results in the following. Note we denote e_j as the n-dimensional one hot vector with a non-zero entry in the j-th dimension. We define

$$\tilde{K} \in \mathbb{R}^{(n-1)\times(n-1)}, \qquad \tilde{K}_{ij} = \begin{cases} 1, & \text{if } i = j,\\ -\dfrac{1-\alpha}{2}, & \text{if } |i-j| = 1,\\ 0, & \text{else,}\end{cases} \quad B := \tilde{K} + \tilde{c}^{-1}I. \tag{S53}$$

Notably $(\delta_s B)^{-1}\mathbf{1}$ is the solution to the dual problem for TI without an exception. We therefore define $\bar{b}^{\mathrm{TI}} := (\delta_s B)^{-1}\mathbf{1}$. We then observe

$$\begin{aligned}
&\delta_s(\tilde{K} + \tilde{c}^{-1}I)\bar{b} + \delta_o\bar{h}(e_p + e_{q-1} - e_q - e_{p-1})] = 1\\
&\implies \delta_s B\bar{b} = 1 - \delta_o\bar{h}(e_p + e_{q-1} - e_q - e_{p-1}))\\
&\implies \bar{b} = (\delta_s B)^{-1}(1 - \delta_o\bar{h}(e_p + e_{q-1} - e_q - e_{p-1}))\\
&\implies \bar{b} = \bar{b}^{\mathrm{TI}} - \alpha'\bar{h}B^{-1}(e_p + e_{q-1} - e_q - e_{p-1}))\\
&\implies \bar{b} = \bar{b}^{\mathrm{TI}} - \alpha'\bar{h}(B_{ip}^{-1} + B_{i(q-1)}^{-1} - B_{iq}^{-1} - B_{i(p-1)}^{-1})
\end{aligned} \tag{S54}$$

The matrix B is tridiagonal with diagonal entries $1+\tilde{c}$ and off–diagonal entries $-\frac{1-\alpha}{2}$. The ratio between the diagonal and off–diagonal terms is therefore

$$-2\,\frac{1+\tilde{c}^{-1}}{1-\alpha} \;\leq\; -2. \tag{S55}$$

As this ratio is smaller than -2, we can apply Theorem 3 to infer

$$B^{-1}_{ij} = \frac{\cosh\big((n-|j-i|)\,\lambda\big) - \cosh\big((n-i-j)\,\lambda\big)}{(1-\alpha)\,\sinh(\lambda)\,\sinh(n\lambda)}, \qquad \lambda := \operatorname{arccosh}\left(\frac{1+\tilde{c}^{-1}}{1-\alpha}\right). \tag{S56}$$

For notational convenience we define $\alpha' := \delta_o/\delta_s = \frac{1-\alpha}{2}$. Thus the dual coefficients are:

$$\begin{aligned}\bar{b} &= \bar{b}^{\mathrm{TI}}_i - \alpha'\bar{h}(B^{-1}_{ip} + B^{-1}_{i(q-1)} - B^{-1}_{iq} - B^{-1}_{i(p-1)}))\\ \bar{h} &= \tfrac{1}{\delta_s(1+\tilde{c}^{-1})}\big(1-\delta_o(\bar{b}_p + \bar{b}_{q-1} - \bar{b}_q - \bar{b}_{p-1})\big).\end{aligned} \tag{S57}$$

The predictions of the fitted model can be expressed in terms of the dual coefficients by

$$\begin{aligned} f\Big(x^{(j,k)}\Big) &= \sum_{(x,y)\in\mathcal{D}} a_x K(x, x^{(j,k)})\\ &= \sum_{i=1}^{n-1} K(i,i+1;j,k)\,b^{(1)}_i + \sum_{i=1}^{n-1} K(i+1,i;j,k)\,b^{(2)}_i + K(p,q;j,k)h^{(1)} + K(q,p;j,k)h^{(2)}.\end{aligned} \tag{S58}$$

We can express this in terms of $\bar{b}$ and $\bar{h}$ as

$$f(x_{j,k}) = \sum_{i=1}^{n-1} K(i,i+1;j,k)\,\bar{b}_i - \sum_{i=1}^{n-1} K(i+1,i;j,k)\,\bar{b}_i + K(p,q;j,k)\bar{h} - K(q,p;j,k)\bar{h}. \tag{S59}$$

There are now two different cases: either $x_{j,k}$ is in the training data or not.

Case 1: Test data. If $x_{j,k}$ is not in the training data, all data points are either overlapping or distinct. In the first data point, a data point is overlapping if $i=j$ or $i=k-1$ and therefore

$$\sum_{i=1}^{n-1} K(i,i+1;j,k)\,\bar{b}_i + K(p,q;j,k)\bar{h} = \kappa_d\langle\bar{b}\rangle + \delta_o(\bar{b}_j + \bar{b}_{k-1}) + \delta_o\bar{h}(\delta_{(j=p)} + \delta_{(k=q)}) \tag{S60}$$

(Note that one of these dual coefficients, as in the previous section, may not exist, if $i\in\{1,n\}$. To leave the expression simple, we define $\bar{b}_0 = \bar{b}_n = 0$, covering these cases as well.) Analogously, the second sum amounts to

$$\sum_{i=1}^{n-1} K(i+1,i;j,k)\bar{b}_i + K(q,p;j,k)\bar{h} = \kappa_d\langle\bar{b}\rangle + \delta_o(\bar{b}_{(j-1)} + \bar{b}_k) + \delta_o\bar{h}(\delta_{(j=q)} + \delta_{(k=p)}) \tag{S61}$$

and overall, the model behavior is expressed as

$$f(x_{j,k}) = \delta_o\big(\bar{b}_j + \bar{b}_{k-1} - \bar{b}_{(j-1)} - \bar{b}_k\big) + \delta_o\bar{h}\big(\delta_{(j=p)} + \delta_{(k=q)} - \delta_{(j=q)} - \delta_{(k=p)}\big). \tag{S62}$$

Case 2: Training data. If $x_{j,k}$ is in the training data, then there are four cases. Either $k=j+1$, $k=j-1$, or $(j,k)\in\{(p,q),(q,p)\}$. In the first case, the first sum has as one of its data points $(j,k)=(j,j+1)$ itself. The only overlapping data points are potentially with the exception training pairs.

$$\sum_{i=1}^{n-1} K(i,i+1;j,k)\,\bar{b}_i + K(p,q;j,j+1)\bar{h} = \kappa_d\langle a\rangle + \delta_s(\bar{b}_j) + \delta_o\bar{h}(\delta_{(j=p)} + \delta_{(k=q)}) \tag{S63}$$

Analogously the second sum is:

$$\sum_{i=1}^{n-1} K(i+1,i;j,k)\bar{b} + K(q,p;j,j,k=j+1)\bar{h} = \kappa_d\langle\bar{a}\rangle + \delta_o(\bar{b}_{j-1}+\bar{b}_k) + \delta_o\bar{h}(\delta_{(j=q)}+\delta_{(k=p)}) \tag{S64}$$

and overall, the model behavior is expressed as

$$f(x_{j,k}) = \delta_s\bar{b}_j - \delta_o(\bar{b}_{j-1}+\bar{b}_k) + \delta_o\bar{h}(\delta_{(j=p)}+\delta_{(k=q)}-\delta_{(j=q)}-\delta_{(k=p)}) \tag{S65}$$

The second case is analogous:

$$f(x_{j,k}) = -\delta_s\bar{b}_k + \delta_o(\bar{b}_j+\bar{b}_{k-1}) + \delta_o\bar{h}(\delta_{(j=p)}+\delta_{(k=q)}-\delta_{(j=q)}-\delta_{(k=p)}) \tag{S66}$$

The third case is analogous to the first case. Note that the similar data point is the exception training pair.

$$f(x_{j,k}) = \delta_s\bar{h} - \delta_o(\bar{b}_p+\bar{b}_{q-1}-\bar{b}_{p-1}+\bar{b}_q) \tag{S67}$$

The last case is analogous to the fourth case.

$$f(x_{j,k}) = -\delta_s\bar{h} + \delta_o(\bar{b}_p+\bar{b}_{q-1}-\bar{b}_{p-1}+\bar{b}_q) \tag{S68}$$

Taken together, the prediction made by $\bar{a}$ are expressed as

$$f(x_{j,k}) = \begin{cases} \delta_s\bar{b}_j - \delta_o(\bar{b}_{j-1}+\bar{b}_k) + \delta_o\bar{h}(\delta_{(j=p)}+\delta_{(k=q)}-\delta_{(j=q)}-\delta_{(k=p)}), & \text{if } k=j+1, \\ -\delta_s\bar{b}_k + \delta_o(\bar{b}_j+\bar{b}_{k-1}) + \delta_o\bar{h}(\delta_{(j=p)}+\delta_{(k=q)}-\delta_{(j=q)}-\delta_{(k=p)}), & \text{if } k=j-1, \\ \delta_s\bar{h} - \delta_o(\bar{b}_p+\bar{b}_{q-1}-\bar{b}_{p-1}-\bar{b}_q), & \text{if } j=p \text{ and } k=q, \\ -\delta_s\bar{h} + \delta_o(\bar{b}_p+\bar{b}_{q-1}-\bar{b}_{p-1}-\bar{b}_q), & \text{if } j=q \text{ and } k=p, \\ \delta_o\big(\bar{b}_j+\bar{b}_{k-1}-\bar{b}_{(j-1)}-\bar{b}_k\big) + \delta_o\bar{h}\big(\delta_{(j=p)}+\delta_{(k=q)}-\delta_{(j=q)}-\delta_{(k=p)}\big), & \text{else} \end{cases}$$

This makes apparent the emergent rank representation for test cases:

$$r_j = \delta_o\big(\bar{b}_j - \bar{b}_{(j-1)}\big) + \delta_o\bar{h}\big(\delta_{(j=p)}-\delta_{(j=q)}\big), \tag{S69}$$

i.e.

$$f(x_j,x_k) = r_j - r_k. \tag{S70}$$

E.5. Reformulating $\bar{h}$

Using the rank, can reformulate $\bar{h}$ as the following:

$$\bar{h} = \frac{1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}}}{\delta_s(1+\tilde{c}^{-1}) - \delta_o(D_{pq}+D_{qp}-D_{pp}-D_{qq})}, \tag{S71}$$

where

$$D_{ij} := \delta_o\Big(B^{-1}_{(i-1)j} + B^{-1}_{i(j-1)} - B^{-1}_{ij} - B^{-1}_{(i-1)(j-1)}\Big), \tag{S72}$$

$$r_j^{\mathrm{TI}} := \delta_o(\bar{b}_j^{\mathrm{TI}} - \bar{b}_{j-1}^{\mathrm{TI}}). \tag{S73}$$

We note that we can use Lemma S1.4 (Lippl et al., 2024) to express

$$r_j^{\mathrm{TI}} = \frac{\sinh((\frac{n+1}{2}-j)\lambda)}{\sinh(\frac{n+1}{2}\lambda) - \sinh(\frac{n-1}{2}\lambda)}. \tag{S74}$$

We show this below.

We remind ourselves of the following.

$$\bar{b} = \bar{b}_i^{\mathrm{TI}} - \alpha' \bar{h}\big(B_{ip}^{-1} + B_{i(q-1)}^{-1} - B_{iq}^{-1} - B_{i(p-1)}^{-1}\big), \tag{S75}$$

$$\bar{h} = \tfrac{1}{\delta_s(1+\tilde{c}^{-1})}\big(1 - \delta_o(\bar{b}_p + \bar{b}_{q-1} - \bar{b}_q - \bar{b}_{p-1})\big), \tag{S76}$$

$$r_j = \delta_o\big(\bar{b}_j - \bar{b}_{(j-1)}\big) + \delta_o \bar{h}\big(\delta_{(j=p)} - \delta_{(j=q)}\big). \tag{S77}$$

Currently, $\bar{b}$ and $\bar{h}$ are interdependent—we will now solve for $\bar{h}$.

We define

$$D_{ij} := \alpha'\big(B_{(i-1)j}^{-1} + B_{i(j-1)}^{-1} - B_{ij}^{-1} - B_{(i-1)(j-1)}^{-1}\big).$$

Then the first term of the rank can be written as

$$\begin{aligned}\delta_o\big(\bar{b}_j - \bar{b}_{j-1}\big) &= \underbrace{\delta_o(\bar{b}_j^{\mathrm{TI}} - \bar{b}_{j-1}^{\mathrm{TI}})}_{=r_j^{\mathrm{TI}}} - \alpha'\delta_o\bar{h}\Big[\big(B_{jp}^{-1} - B_{(j-1)p}^{-1}\big) + \big(B_{j(q-1)}^{-1} - B_{(j-1)(q-1)}^{-1}\big) \\ &\qquad - \big(B_{jq}^{-1} - B_{(j-1)q}^{-1}\big) - \big(B_{j(p-1)}^{-1} - B_{(j-1)(p-1)}^{-1}\big)\Big] \qquad \text{(S78)} \\ &= r_j^{\mathrm{TI}} + \delta_o\bar{h}\big(D_{jp} - D_{jq}\big). \qquad \text{(S79)}\end{aligned}$$

Thus, the rank can be rewritten as

$$r_j = r_j^{\mathrm{TI}} + \delta_o\bar{h}\big(D_{jp} - D_{jq}\big) + \delta_o\bar{h}\big(\delta_{j=p} - \delta_{j=q}\big). \tag{S80}$$

We now rewrite $\bar{h}$. Starting from (S76) and defining $Z := \delta_s(1+\tilde{c}^{-1})$,

$$Z\bar{h} = 1 - \delta_o\Big[(\bar{b}_p - \bar{b}_{p-1}) - (\bar{b}_q - \bar{b}_{q-1})\Big]. \tag{S81}$$

Substituting $\delta_o(\bar{b}_j - \bar{b}_{j-1}) = r_j^{\mathrm{TI}} + \delta_o\bar{h}(D_{jp} - D_{jq})$ at $j = p$ and $j = q$,

$$Z\bar{h} = 1 - \Big[\big(r_p^{\mathrm{TI}} + \delta_o\bar{h}(D_{pp} - D_{pq})\big) - \big(r_q^{\mathrm{TI}} + \delta_o\bar{h}(D_{qp} - D_{qq})\big)\Big]. \tag{S82}$$

$$Z\bar{h} = 1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}} - \delta_o\bar{h}(D_{pp} - D_{pq}) + \delta_o\bar{h}(D_{qp} - D_{qq}). \tag{S83}$$

$$Z\bar{h} + \delta_o\bar{h}(D_{pp} - D_{pq} - D_{qp} + D_{qq}) = 1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}}. \tag{S84}$$

$$Z\bar{h} - \delta_o\bar{h}(-D_{pp} + D_{pq} + D_{qp} - D_{qq}) = 1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}}. \tag{S85}$$

Dividing by the factor on the left establishes (S71).

E.6. Characterizing D_{ij}

We compute D_{ij} in closed form. Recall

$$D_{ij} := \alpha'\big(B_{(i-1)j}^{-1} + B_{i(j-1)}^{-1} - B_{ij}^{-1} - B_{(i-1)(j-1)}^{-1}\big), \tag{S86}$$

and

$$\alpha' B_{ij}^{-1} = \alpha' \frac{\cosh\big((n - |j-i|)\lambda\big) - \cosh\big((n-i-j)\lambda\big)}{(1-\alpha)\sinh(\lambda)\sinh(n\lambda)} = \frac{\sinh\big((n - \max(i,j))\lambda\big)\sinh\big(\min(i,j)\,\lambda\big)}{\sinh(\lambda)\sinh(n\lambda)}. \tag{S87}$$

Applying the identity $\cosh A - \cosh B = 2\sinh\big(\frac{A+B}{2}\big)\sinh\big(\frac{A-B}{2}\big)$ with $A = (n - |j-i|)\lambda$ and $B = (n-i-j)\lambda$, and using $|j-i| + i + j = 2\max(i,j), i + j - |j-i| = 2\min(i,j)$, we find

$$\cosh\big((n - |j-i|)\lambda\big) - \cosh\big((n-i-j)\lambda\big) = 2\sinh\big((n - \max(i,j))\lambda\big)\sinh\big(\min(i,j)\,\lambda\big). \tag{S88}$$

Substituting and using $\alpha' = (1-\alpha)/2$, so that $2\alpha'/(1-\alpha) = 1$, gives the second equality. When $i \le j$, this further simplifies to

$$\alpha' B^{-1}_{ij} = \frac{\sinh\big((n-j)\lambda\big)\sinh(i\lambda)}{\sinh(\lambda)\sinh(n\lambda)}. \tag{S89}$$

We split into three cases and use a hyperbolic trigonometric identities to simplify $D_i j$.

Case 1: $i \le j-1$**.**

$$D_{ij} = \frac{\big(\sinh(i\lambda) - \sinh((i-1)\lambda)\big)\big(\sinh((n+1-j)\lambda) - \sinh((n-j)\lambda)\big)}{\sinh(\lambda)\sinh(n\lambda)} \tag{S90}$$

$$= \frac{4\cosh\big((i-\frac{1}{2})\lambda\big)\sinh(\frac{\lambda}{2})\cosh\big((n-j+\frac{1}{2})\lambda\big)\sinh(\frac{\lambda}{2})}{\sinh(\lambda)\sinh(n\lambda)} \tag{S91}$$

$$= 2\cosh\big((i-\tfrac{1}{2})\lambda\big)\cosh\big((n-j+\tfrac{1}{2})\lambda\big)\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}. \tag{S92}$$

Case 2: $j \le i-1$**.** By the symmetry $D_{ij} = D_{ji}$,

$$D_{ij} = 2\cosh\big((j-\tfrac{1}{2})\lambda\big)\cosh\big((n-i+\tfrac{1}{2})\lambda\big)\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}. \tag{S93}$$

Case 3: $i = j$**.** By the symmetry of B^{-1},

$$D_{ii} = \alpha'\big(2B^{-1}_{(i-1)i} - B^{-1}_{ii} - B^{-1}_{(i-1)(i-1)}\big) \tag{S94}$$

$$= \frac{2\sinh((n-i)\lambda)\sinh((i-1)\lambda) - \sinh((n-i)\lambda)\sinh(i\lambda) - \sinh((n-i+1)\lambda)\sinh((i-1)\lambda)}{\sinh(\lambda)\sinh(n\lambda)} \tag{S95}$$

$$= \frac{\sinh((i-1)\lambda)\big[\sinh((n-i)\lambda) - \sinh((n-i+1)\lambda)\big] + \sinh((n-i)\lambda)\big[\sinh((i-1)\lambda) - \sinh(i\lambda)\big]}{\sinh(\lambda)\sinh(n\lambda)} \tag{S96}$$

$$= \frac{-2\sinh(\frac{\lambda}{2})\Big[\sinh((i-1)\lambda)\cosh((n-i+\frac{1}{2})\lambda) + \sinh((n-i)\lambda)\cosh((i-\frac{1}{2})\lambda)\Big]}{\sinh(\lambda)\sinh(n\lambda)}. \tag{S97}$$

Applying $2\sinh A\cosh B = \sinh(A+B) + \sinh(A-B)$,

$$2\sinh((i-1)\lambda)\cosh((n-i+\tfrac{1}{2})\lambda) = \sinh((n-\tfrac{1}{2})\lambda) + \sinh((2i-n-\tfrac{3}{2})\lambda), \tag{S98}$$

$$2\sinh((n-i)\lambda)\cosh((i-\tfrac{1}{2})\lambda) = \sinh((n-\tfrac{1}{2})\lambda) + \sinh((n-2i+\tfrac{1}{2})\lambda). \tag{S99}$$

Adding and using $\sinh x + \sinh y = 2\sinh(\frac{x+y}{2})\cosh(\frac{x-y}{2})$ with $x+y = -\lambda$, $x-y = (4i-2n-2)\lambda$,

$$\begin{aligned}&\sinh((i-1)\lambda)\cosh((n-i+\tfrac{1}{2})\lambda) + \sinh((n-i)\lambda)\cosh((i-\tfrac{1}{2})\lambda) = \\ &\sinh((n-\tfrac{1}{2})\lambda) - \sinh(\tfrac{\lambda}{2})\cosh((2i-n-1)\lambda).\end{aligned} \tag{S100}$$

Therefore

$$D_{ii} = \frac{-2\sinh(\frac{\lambda}{2})\sinh((n-\frac{1}{2})\lambda) + 2\sinh^2(\frac{\lambda}{2})\cosh((2i-n-1)\lambda)}{\sinh(\lambda)\sinh(n\lambda)}. \tag{S101}$$

Using $\sinh((n-\frac{1}{2})\lambda) = \sinh(n\lambda)\cosh(\frac{\lambda}{2}) - \cosh(n\lambda)\sinh(\frac{\lambda}{2})$, $\sinh(\lambda) = 2\sinh(\frac{\lambda}{2})\cosh(\frac{\lambda}{2})$, and $2\sinh^2(\frac{\lambda}{2}) = \cosh(\lambda) - 1$,

$$-2\sinh(\tfrac{\lambda}{2})\sinh((n-\tfrac{1}{2})\lambda) = -\sinh(\lambda)\sinh(n\lambda) + (\cosh(\lambda)-1)\cosh(n\lambda), \tag{S102}$$

$$2\sinh^2(\tfrac{\lambda}{2})\cosh((2i-n-1)\lambda) = (\cosh(\lambda)-1)\cosh((2i-n-1)\lambda). \tag{S103}$$

Combining,

$$D_{ii} = \frac{-\sinh(\lambda)\sinh(n\lambda) + (\cosh(\lambda)-1)\big[\cosh(n\lambda) + \cosh((2i-n-1)\lambda)\big]}{\sinh(\lambda)\sinh(n\lambda)}. \tag{S104}$$

Applying $2\cosh A \cosh B = \cosh(A+B) + \cosh(A-B)$ with $A = (i-\frac{1}{2})\lambda$, $B = (n-i+\frac{1}{2})\lambda$,

$$\cosh(n\lambda) + \cosh((2i-n-1)\lambda) = 2\cosh\big((i-\tfrac{1}{2})\lambda\big)\cosh\big((n-i+\tfrac{1}{2})\lambda\big). \tag{S105}$$

Substituting and using $(\cosh(\lambda)-1)/\sinh(\lambda) = \tanh(\lambda/2)$,

$$D_{ii} = -1 + 2\,\frac{\cosh\big((i-\frac{1}{2})\lambda\big)\cosh\big((n-i+\frac{1}{2})\lambda\big)\tanh(\lambda/2)}{\sinh(n\lambda)}. \tag{S106}$$

Combined form. Writing the off-diagonal cases uniformly in terms of $\min$ and $\max$, we find

$$\boxed{D_{ij} = -\delta_{i=j} + 2\,\frac{\cosh\big((i-\frac{1}{2})\lambda\big)\cosh\big((n-i+\frac{1}{2})\lambda\big)\tanh(\lambda/2)}{\sinh(n\lambda)}.} \tag{S107}$$

It will be useful to define a subpart of this:

$$\tilde{D}_{ij} := \cosh\big((\min(i,j)-\tfrac{1}{2})\lambda\big)\cosh\big((n-\max(i,j)+\tfrac{1}{2})\lambda\big), \tag{S108}$$

Note that

$$D_{ij} = -\delta_{i=j} + 2\,\tilde{D}_{ij}\,\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}. \tag{S109}$$

We start from the generalized expression for the exception scaling factor, $\bar{h}$, which isolates the localized rank correction for the exception pair (p,q).

$$\bar{h} = \frac{1 - r_p^{\text{TI}} + r_q^{\text{TI}}}{Z - \delta_o(2D_{pq} - D_{pp} - D_{qq})} \tag{S110}$$

where the regularization factor is $Z := \delta_s(1+\tilde{c}^{-1})$. To evaluate the denominator, we expand the D_{ij} terms. From (S109), we infer

$$\begin{aligned} 2D_{pq} - D_{pp} - D_{qq} =& 2\left(2\tilde{D}_{pq}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\right) \\ &- \left(-1 + 2\tilde{D}_{pp}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\right) - \left(-1 + 2\tilde{D}_{qq}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\right) \\ =& 2 + 4\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\left(\tilde{D}_{pq} - \frac{1}{2}\tilde{D}_{pp} - \frac{1}{2}\tilde{D}_{qq}\right). \end{aligned} \tag{S111}$$

We apply the hyperbolic difference identity $\cosh A - \cosh B = 2\sinh(\frac{A+B}{2})\sinh(\frac{A-B}{2})$ to the partial distance expressions within the parentheses. This allows us to factor out $\sinh\big(\frac{q-p}{2}\lambda\big)$, consolidating the remaining terms into

$$G := \cosh\left(\left(q - \frac{1}{2}\right)\lambda\right)\sinh\left(\left(n - \frac{p+q-1}{2}\right)\lambda\right) + \cosh\left(\left(n - p + \frac{1}{2}\right)\lambda\right)\sinh\left(\frac{p+q-1}{2}\lambda\right) \tag{S112}$$

Substituting G back into our expanded term yields:

$$2D_{pq} - D_{pp} - D_{qq} = 2 + 4\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\sinh\left(\frac{q-p}{2}\lambda\right)G. \tag{S113}$$

Finally, substituting this back into the denominator of our original equation transforms the regularized scaling factor $\bar{h}$ into a purely analytical form:

$$\bar{h} = \frac{1 - r_p^{\text{TI}} + r_q^{\text{TI}}}{(Z - 2\delta_o) - 4\delta_o\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\sinh\big(\frac{q-p}{2}\lambda\big)G} \tag{S114}$$

We then substitute $\bar{h}$ and the baseline transitive rank $\tilde{r}^{\text{TI}}$ back into the full rank expression (S80):

$$r_i = \tilde{r}_i^{\text{TI}} + \delta_o \bar{h}(D_{ip} - D_{iq}) + \delta_o \bar{h}(\delta_{i=p} - \delta_{i=q}) \tag{S115}$$

This is almost the final expression—we simply need to make a few further algebraic simplifications. First we note that

$$\begin{aligned} D_{ip} - D_{iq} &= \left(-\delta_{i=p} + 2\tilde{D}_{ip}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\right) - \left(-\delta_{i=q} + 2\tilde{D}_{iq}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\right) \\ &= -(\delta_{i=p} - \delta_{i=q}) + 2\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right). \end{aligned} \tag{S116}$$

Hence,

$$\begin{aligned} r_i &= r_i^{\text{TI}} + \delta_o \bar{h}\left[-(\delta_{i=p} - \delta_{i=q}) + 2\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right)\right] + \delta_o \bar{h}(\delta_{i=p} - \delta_{i=q}) \\ &= r_i^{\text{TI}} + 2\delta_o \bar{h}\frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right). \end{aligned} \tag{S117}$$

Finally, plugging in (S114) yields

$$\begin{aligned} r_i &= r_i^{\text{TI}} + \frac{2\delta_o \frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\left(1 - \tilde{r}_p^{\text{TI}} + \tilde{r}_q^{\text{TI}}\right)\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right)}{(Z - 2\delta_o) - 4\delta_o \frac{\tanh(\lambda/2)}{\sinh(n\lambda)}\sinh\left(\frac{q-p}{2}\lambda\right) G} \\ &= r_i^{\text{TI}} + \frac{\left(1 - r_p^{\text{TI}} + r_q^{\text{TI}}\right)\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right)}{\frac{(Z-2\delta_o)}{2\delta_o \frac{\tanh(\lambda/2)}{\sinh(n\lambda)}} - 2\sinh\left(\frac{q-p}{2}\lambda\right) G} \\ &= r_i^{\text{TI}} + \frac{\left(1 - r_p^{\text{TI}} + r_q^{\text{TI}}\right)\left(\tilde{D}_{ip} - \tilde{D}_{iq}\right)}{\sinh(\lambda)\sinh(n\lambda) - 2\sinh\left(\frac{q-p}{2}\lambda\right) G}, \end{aligned} \tag{S118}$$

where we used

$$Z - 2\delta_o = \delta_s(\alpha + \tilde{c}^{-1}) = 2\delta_o \tanh(\tfrac{\lambda}{2})\sinh\lambda. \tag{S119}$$

E.7. Expression of the training set behavior

We derive the prediction on adjacent training pairs in terms of r_j, which already incorporates the exception correction. Starting from the case-split expression for $k = j + 1$,

$$f(x_j, x_{j+1}) = \delta_s \bar{b}_j - \delta_o(\bar{b}_{j-1} + \bar{b}_{j+1}) + \delta_o \bar{h}\big(\delta_{j=p} + \delta_{j+1=q} - \delta_{j=q} - \delta_{j+1=p}\big), \tag{S120}$$

we observe that the row-j dual equation (S52) reads

$$\delta_s(1 + \tilde{c}^{-1})\bar{b}_j - \delta_o(\bar{b}_{j-1} + \bar{b}_{j+1}) + \delta_o \bar{h}\big(\delta_{j=p} + \delta_{j+1=q} - \delta_{j=q} - \delta_{j+1=p}\big) - 1 = 0. \tag{S121}$$

Subtracting the left-hand side of (S121) from (S120), the off-diagonal $\bar{b}$ terms and the indicator terms cancel, yielding

$$f(x_j, x_{j+1}) = 1 - \delta_s \tilde{c}^{-1}\,\bar{b}_j. \tag{S122}$$

Next, we express $\bar{b}_j$ in terms of the full rank. From the rank definition $r_j = \delta_o(\bar{b}_j - \bar{b}_{j-1}) + \delta_o \bar{h}(\delta_{j=p} - \delta_{j=q})$,

$$r_j - r_{j+1} = \delta_o(2\bar{b}_j - \bar{b}_{j-1} - \bar{b}_{j+1}) + \delta_o \bar{h}\big(\delta_{j=p} - \delta_{j+1=p} - \delta_{j=q} + \delta_{j+1=q}\big). \tag{S123}$$

(S121) implies

$$-\delta_o(\bar{b}_{j-1} + \bar{b}_{j+1}) + \delta_o \bar{h}\big(\delta_{j=p} + \delta_{j+1=q} - \delta_{j=q} - \delta_{j+1=p}\big) = 1 - \delta_s(1 + \tilde{c}^{-1})\bar{b}_j. \tag{S124}$$

This lets us infer

$$r_j - r_{j+1} = 2\delta_o \bar{b}_j - \delta_s(1+\tilde{c}^{-1})\bar{b}_j + 1. \tag{S125}$$

Using $2\delta_o = \delta_s(1-\alpha)$,

$$r_j - r_{j+1} = 1 - \delta_s\big(\alpha + \tilde{c}^{-1}\big)\bar{b}_j \quad \Longrightarrow \quad \delta_s \bar{b}_j = \frac{1-(r_j - r_{j+1})}{\alpha + \tilde{c}^{-1}}. \tag{S126}$$

Substituting (S126) into (S122),

$$f(x_j, x_{j+1}) = 1 - \frac{\tilde{c}^{-1}}{\alpha + \tilde{c}^{-1}}\big[1 - (r_j - r_{j+1})\big] = \frac{\alpha}{\alpha + \tilde{c}^{-1}} + \frac{\tilde{c}^{-1}}{\alpha + \tilde{c}^{-1}}(r_j - r_{j+1}). \tag{S127}$$

Defining $m := \alpha/(\alpha + \tilde{c}^{-1})$, so $1 - m = \tilde{c}^{-1}/(\alpha + \tilde{c}^{-1})$,

$$f(x_j, x_{j+1}) = m \cdot 1 + (1-m)(r_j - r_{j+1}), \qquad m = \frac{\alpha}{\alpha + \tilde{c}^{-1}}. \tag{S128}$$

we have established the expression for the training case.

E.8. Properties of the ranking system

We find that within each transitive section r_j is monotonically decreasing in j. We note that $\lambda > 0$ and so r_j^{TI} is monotonically decreasing in j. We recall

$$r_j^{\mathrm{pert}}(\alpha, \tilde{c}) = \frac{\big(1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}}\big)\left(\tilde{D}_{jp} - \tilde{D}_{jq}\right)}{\sinh(\lambda)\sinh(n\lambda) - 2\sinh\left(\frac{q-p}{2}\lambda\right)G}.$$

Note that $1 - r_p^{\mathrm{TI}} + r_q^{\mathrm{TI}} > 0$. Further, because $\sinh\left(\frac{q-p}{2}\lambda\right) < 0$ and $G > 0$, the denominator is also positive. Hence monotonicity is a function of $\tilde{D}_{jp} - \tilde{D}_{jq}$. Recall

$$\tilde{D}_{ij} = \cosh\big((\min(i,j) - \tfrac{1}{2})\lambda\big)\cosh\big((n - \max(i,j) + \tfrac{1}{2})\lambda\big).$$

Hence, for $j \le q$,

$$\tilde{D}_{jp} - \tilde{D}_{jq} = \cosh((j - \tfrac{1}{2})\lambda)\left(\cosh\big((n - p + \tfrac{1}{2})\lambda\big) - \cosh\big((n - q + \tfrac{1}{2})\lambda\big)\right) \tag{S129}$$

Because $\cosh(x\lambda)$ is monotonically increasing in x,

$$\cosh\big((n - p + \tfrac{1}{2})\lambda\big) - \cosh\big((n - q + \tfrac{1}{2})\lambda\big) < 0. \tag{S130}$$

Because $\cosh((j - \frac{1}{2})\lambda)$ is monotonically increasing, the total expression is monotonically decreasing. The claim is established analogously for $j \ge p$.