Bericht Bachelorprojekt Transitive Inferenz

David Borukhson, 4732412
borukhd@cs.uni-freiburg.de
Technische Fakultät
Universität Freiburg

Wintersemester 2019/2020

Contents

1 Übersicht

Das Ziel des Projektes war die Untersuchung von Modellen des transitiven Schließens. Dazu wurden die Modelle in einer Version von CCOBRA implementiert und auf unterschiedlichen Datensätzen ausgewertet. Spezies, welche in die Auswertung eingingen, waren Tauben, Wespen, Makaken und Menschen. Die Ergebnisse erlauben Schlüsse auf Vorhersageeigenschaften einzelner Modelle bezüglich Tierarten.

1 Overview

The goal of this project was to investigate the functioning of models for transitive reasoning. For this purpose, models were implemented in a version of CCOBRA and evaluated on a number of different data sets. Species included in the evaluation are pigeons, wasps, macaque monkeys and humans. The result allow to draw conclusions on prediction qualities of particular models with respect to species.

2 Model Implementation

Nine versions of associative reasoning models and three cognitive reasoning models for Transitive Inference (TI) reasoning were implemented in Python and evaluated using the Cognitive Computation for Behavioral Reasoning Analysis (CCOBRA) Framework[1]. In general, initial associative values of stimuli are regarded as 0.

2.1 RL-ELO

Kumaran, Banino, Blundell, Hassabis, and Dayan (2016): This is an associative model that is based on a rating scheme for stimuli as developed for ranking schemes in competitions between individuals such as chess. Stimuli are assigned associative stimuli that effect the probability of selecting this stimuli over another with a different associative value. The change in associative value of a stimulus depends of the value of the stimulus it is paired with in a training trial and which of the two stimuli is reinforced in the pairing.

Predicting item X from test pair $\{X, Y\}$ happens with probability p as given by the following function, β representing a free parameter to avoid local minima:

$$p(X|\{X,Y\}) = \frac{1}{1+e^{-\beta(V[X]-V[Y])}} = 1 - p(Y|\{X,Y\}). \tag{1}$$

Associative values of stimuli $V[X]$, $V[Y]$ are recursively calculated for each trial

[1] `https://github.com/CognitiveComputationLab/CCOBRA`

considering learning rate α:

$$V[X]' = \begin{cases} V[X] + \alpha * (1 - p(X|\{X,Y\}), & \text{if } X \text{ reinforced;} \\ V[X] + \alpha * (p(X|\{X,Y\} - 1), & \text{if } X \text{ inhibited.} \end{cases} \tag{2}$$

2.2 RL-ELO$_\mathbf{F}$

Kumaran et al. (2016): A variant of the model above additionally uses Gaussian distributed noise to simulate forgetting or inaccurate retrieval of associative values. Using a free parameter σ that forms the standard deviation of random variable $R \sim N(0, \sigma)$, a randomly generated value is added to associative values of stimuli, thus creating increasing distortion over time:

$$V[X]' = \begin{cases} V[X] + \alpha * (1 - p(X|\{X,Y\}) + R, & \text{if } X \text{ reinforced;} \\ V[X] + \alpha * (p(X|\{X,Y\} - 1) + R, & \text{if } X \text{ inhibited.} \end{cases} \tag{3}$$

The probability of choosing a particular stimulus is identically calculated as in RL-ELO.

2.3 Value Transfer Theory

VTT, von Fersen, Wynne, Delius, and Staddon (1991); Kumaran et al. (2016): This model is an associative model that uses recursive associative value reassignments involving a direct and an indirect component. The following equation illustrates assignment of a new value of stimulus X, $V[X]'$ given $V[X]$, the current value of X as well as free variables for learning rate α, and transfer factor θ on presentation of some training pair $\{X, Y\}$.

$$V[X]' = \begin{cases} \alpha * (1 - V[X]), & \text{if } X \text{ reinforced;} \\ \alpha * (-1 - V[X]) + \theta * V[Y], & \text{if } X \text{ inhibited.} \end{cases}$$

The probability of predicting item X from test pair $\{X, Y\}$ is calculated in the same manner as for RL-ELO and RL-ELO$_\mathrm{F}$.

2.4 Rescorla-Wagner

Rescorla, Wagner, et al. (1972) Is treated by Kumaran et al. (2016) as an instance of Value Transfer Theory with $\theta = 0$, yielding associative value updates on X from testing pair $\{X, Y\}$:

$$V[X]' = \begin{cases} \alpha * (1 - V[X]), & \text{if } X \text{ reinforced;} \\ \alpha * (-1 - V[X]), & \text{if } X \text{ inhibited.} \end{cases}$$

However, (Wynne, 1995) interprets the models probability function for choosing X from test pair $\{X, Y\}$:

$$p(X|\{X,Y\}) = 1 - p(Y|\{X,Y\}) = \frac{1}{1 + e^{\beta\left(1 - 2*\left(\frac{V[X]}{V[X]+V[Y]}\right)\right)}}$$

Note, that in this context, omitted the similarity variable was omitted $V[Z]$ – as in the TRI-Toolbox implentation by Goodman and Lazareva[2], because the stimuli dealt with are non-verbal and supposed to bear no inherent meaning to test subjects. The associative value updates with respect to free learning parameter α are thus:

$$V[X]' = \begin{cases} \alpha * (1 - V[X]), & \text{if } X \text{ reinforced;} \\ \alpha * (-V[X]), & \text{if } X \text{ inhibited.} \end{cases}$$

Both variations of the Rescorla-Wagner model were implemented.

2.5 Configural Cues

Wynne (1995): This model is an extension of Rescorla-Wagner using configural associative values in addition to elemental values:

$$V[X|\{X,Y\}]' = V[X|\{X,Y\}] + \beta * (1 - V[X|\{X,Y\}])$$

on reward of X and

$$V[X|\{X,Y\}]' = V[X|\{X,Y\}] - \beta V[X|\{X,Y\}]$$

on non-reward of X. These configural values are part of the computation of prediction probability, weighted by free parameter $\gamma > 0$:

$$p(X|\{X,Y\}) = 1 - p(Y|\{X,Y\}) =$$

$$\left(1 + e^{\beta\left(1 - 2*\left(\frac{V[X] + \gamma * V[X|\{X,Y\}]}{V[X] + V[Y] + \gamma * V[X|\{X,Y\}] + \gamma * V[Y|\{X,Y\}]}\right)\right)}\right)^{-1}$$

2.6 Bush-Mosteller

Bush and Mosteller (1955); Wynne (1995): This model again bears some similarity with the interpretation of the Rescorla-Wagner model by Wynne as to associative value calculation, but uses two distinct free learning parameters for reward α_r and non-reward α_n calculation:

$$V[X]' = \begin{cases} \alpha_r * (1 - V[X]), & \text{if } X \text{ reinforced;} \\ \alpha_n * (-V[X]), & \text{if } X \text{ inhibited.} \end{cases}$$

However, prediction probability calculation differs. Given $r := \frac{V[X]}{V[X]+V[Y]}$:

$$p(X|\{X,Y\}) = \begin{cases} 0.5 + 0.883 * (2 * r - 1)^{0.75}, & r \geq 0.5; \\ 0.5 + 0.883 * (1 - 2 * r)^{0.75}, & r < 0.5. \end{cases}$$

[2] `http://www.copal-lab.com/tri-toolbox.html`

2.7 Siemann-Delius

Siemann and Delius (1998): This model, elaborated on in (Guez & Audley, 2013), bears some resemblance to the Configural Cues model by Wynne.

$$V[X]' = \begin{cases} V[X] * (1 + \alpha_r * p(X|\{X,Y\}) * \epsilon), & \text{if } X \text{ reinforced;} \\ V[X] * (1 + \alpha_n * p(X|\{X,Y\}) * \epsilon), & \text{if } X \text{ inhibited.} \end{cases}$$

ϵ being a free parameter for the weighting of elemental associative values. The probability of choosing a stimulus is considered on calculation of associative values $V[X|\{X,Y\}]$ of respective stimuli and stimuli pairs:

$$p(X|\{X,Y\}) = \frac{V[X] * V[X|\{X,Y\}]}{V[X] * V[X|\{X,Y\}] + V[Y] * V[Y|\{X,Y\}]},$$

defining configural associative values $V[X|\{X,Y\}]$ with learning parameter α and weighting parameter κ:

$$V[X]' = \begin{cases} V[X|\{X,Y\}] * (1 + \alpha_r * p(X|\{X,Y\}) * \kappa), & \text{if } X \text{ reinforced;} \\ V[X|\{X,Y\}] * (1 + \alpha_n * p(X|\{X,Y\}) * \kappa), & \text{if } X \text{ inhibited.} \end{cases}$$

2.8 Internal Linear Array

Riley and Trabasso (1974): A cognitive model based on the assumption that a linear series is produced in the test subject, allowing to retrieve predictions about stimulus pairs by comparing their order in the series; end items of an internal list are easier to retrieve and compare than those in the middle of a series. A correct answer to a stimulus pair requiring transitive inference can only be guaranteed, if internal arrays have been merged — such as AB and BC to ABC for answering testing pair {A,C}. Notably, multiple conclusions may valid while constructing the array; given only the two pairs "$A > B$" and "$D > E$", the following linear representation may occur with equal logical justification:

```
ABDE   ADBE   ADEB   DAEB   DEAB
```

But also, using multiple arrays maintained in parallel:

```
A B    A B    A B     AB    AB
 CD    CD     C D    CD      CD
```

Multiple implementations are therefore conceivable; here, multiple arrays are not employed, but the array is merged at random, i. e. any of the above examples may occur. However, if such a representation results in incorrect prediction of training stimuli, the positions of the respective stimulus pair within a continuous linear array are swapped, one taking the place of the other respectively.

There is also ambiguity as to whether the model can make predictions that are objectively correct, once the representation has been constructed (Lazareva & Wasserman, 2012), or not (Riley & Trabasso, 1974). In this project, the linear array was implemented in such a way that stimuli are always inserted successfully, however retrieval is hampered by a parametric function for stimuli closer to the middle of the internal array:

$$p(X|\{X,Y\}) = \begin{cases} \frac{|arrayLength - positionInArray|}{arrayLength * h}, & X \textit{ in higher array position}; \\ 1 - \frac{|arrayLength - positionInArray|}{arrayLength * h}, & Y \textit{ in higher array position}; \\ 0.5, & X \textit{ not in internal array.} \end{cases}$$

h being a free parameter of retrieval difficulty.

2.9 Spatial Paralogic Theory

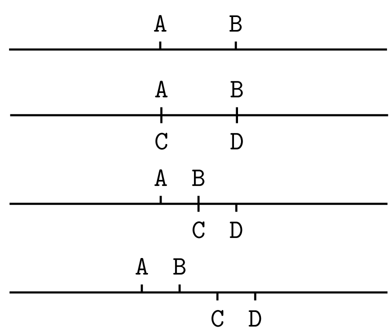
De Soto, London, and Handel (1965): Spatial Paralogic essentially means that during training, stimuli are arranged on a continuous axis in a cognitive representation of space. Here, when new stimuli are inserted to the axis that have no definite relation to items already positioned, they are first inserted in the middle of the axis and then shifted outward in opposite directions, if the position results in inconsistencies:

Trial$_1$ $A < B$:

Trial$_2$ $C < D$:

Trial$_3$ $B < C$:

Trial$_4$ $B < C$:



A stimulus is always correctly selected, if entailed by the internal representation:

$$p(X|\{X,Y\}) = \begin{cases} 1, & X \textit{ in higher array position than } Y; \\ 0, & Y \textit{ in higher array position than } X; \\ 0.5, & X \textit{ not contained in internal array.} \end{cases}$$

2.10 Stimulus Control Topography

SCT, McIlvane and Dube (2003): This model assumes that subjects sometimes take their decision only by selecting the reinforced stimulus from a training pair or rejecting the reinforced one instead of comparing their values as a pair:

$$\text{Select mode: } V_{SCT}[X]' = \begin{cases} V_{SCT}[X] + 1, & X \textit{ reinforced}; \\ V_{SCT}[X], & X \textit{ inhibited.} \end{cases}$$

$$\text{Reject mode: } V_{SCT}[X]' = \begin{cases} V_{SCT}[X], & X \textit{ reinforced}; \\ V_{SCT}[X] - 1, & X \textit{ inhibited.} \end{cases}$$

$$\text{Select and Reject mode: } V_{SCT}[X]' = \begin{cases} V_{SCT}[X]+1, & X \textit{ reinforced}; \\ V_{SCT}[X]-1, & X \textit{ inhibited}. \end{cases}$$

On testing, the stimulus is assigned the higher value is utteered as a prediction:

$$p(X|\{X,Y\}) = \begin{cases} 1, & V_{SCT}[X] > V_{SCT}[Y]; \\ 0, & V_{SCT}[X] < V_{SCT}[Y]; \\ 0.5, & V_{SCT}[X] = V_{SCT}[Y]. \end{cases}$$

Two versions are implemented: One, where the model randomly chooses to select the reinforced stimulus, inhibit the rejected one or do both actions — and one, where select and reject decisions are remembered per stimulus in a pair and repeated in the following pairs, if applicable. The latter version is dubbed "SCTinterpr" in the implementation.

3 Model Evaluation

3.1 General Procedure

The models are designed to be treated the same way as test subjects in respective experiment settings, that is, for each subject of a dataset, the same training and test queries were conducted in unchanged order. Depending on whether a reward was given after a trial, the models adapt accordingly, using no more information than the information available to test subjects. Testing of implemented models takes place using an extended version of the CCOBRA framework.

3.2 Evaluation Structure

CCOBRA implements three different modes of interacting with implementing models: Prediction, Adaptation and Pre-Training. All three functions can be called independently, however the last one is normally used in advance of the evaluation of model performance for a subject in an experiment.

3.2.1 Prediction

In this context, a model prediction means that a certain stimulus is uttered by the model instance given a pair of two stimuli and considering previously presented stimuli, trying to emulate the behaviour of an individual subject. For example, given a trial with stimuli $\{X, Y\}$, a model may select either X, Y or even another stimulus Z. If the stimulus selected by the model is identical with the stimulus the subject chose in the trial, the prediction is successful. Usually, prediction of a stimulus employ a random component or a Bernoulli distribution. The probability that a stimulus response is predicted successfully for a particular model instance, subject and trial is represented by the value of function $cP : \textit{Trials} \times \textit{Models} \times \textit{Parameters} \times \textit{Subjects} \to \mathbb{R}_{\geq 0}$ that depends on the model's own prediction function $p : \textit{Stimulus} \times \textit{Stimulus} \to \mathbb{R}_{\geq 0}$.

In cognitive models, calling the prediction function also may influence thee model's representation, i. e. including a stimulus not mentioned previously. In most cases however, the model is left unchanged after a call for prediction, as no additional information about stimulus orderings is gained by a call for prediction.

3.2.2 Adaptation

A model changes, if it receives information from the outside. This change, adaptation, can be updating associative values of stimuli using equations provided in the implementation or modifying cognitive representation by specified rules that are different but clearly determined respectively for each reasoning model, as specified in the Model Implementation section.

In the present case, models are treated the same way as it is assumed participants were treated in the experiments. Therefore, adaptation only occurs when the correct response to a pair of stimuli would be shown to a subject — that is, after presentation of a test trial and after each task in the experiment of Kao, Jensen, Michaelcheck, Ferrera, and Terrace (2019).

3.2.3 Pre-Training

Pre-training a model for some subject means to use sample data from other subjects to adapt a model already before an experiment takes place. This method is especially useful, if other information is given about the participant, i. e. if they completed other tasks indicative of cognitive capabilities beforehand. As no information usable for such adjustment was given for subjects in the available data sets, the only sensible kind of pre-training would be parameter optimization on other subjects; however, as in this implementation parameters are optimized independently for each subject and none of the implemented models involve a particular statement that can be formulated as a pre-training condition, this method was not employed.

In this implementation, evaluation of model prediction performance occurs by simulating an experiment as closely as possible on the basis of given data sets, the implemented algorithmic models trying to emulate the behaviour of the test subject. Thus, for each model, the adaptation function was called on every trial part of a training phase or in Kao et al.'s dataset. The prediction function was called on every trial, as subjects too were required to choose one stimulus in testing as well as training pairs. This way, a model instance optimized by the technique given in section Optimization Procedure is iterated through all trials presented to some participant in the same order as done in the experiment setting and denoted in the respective data set. If both methods are applied on the same trial, the prediction method is always executed before adaptation is allowed to begin.

Note that for evaluation of the model, only the prediction successes in testing phase trials are considered.

3.3 Optimization Procedure

The parameters for all CCOBRA model implementations were fitted by constrained optimization by linear approximation as implemented in scipy[3] at a maximum of 20 iteration steps and an accuracy of 0.001. Optimization took place gathering all pairs (not only {B, D}) and for all subjects within one training or both training and testing data set. Pairs were regarded as non-ordered and it was assumed that models adapt during training phase only, except in the data set of (Kao, Jensen, Michaelcheck, Ferrera, & Terrace, 2018), where they adapt after both training and test trials as is meaningful with respect to the experiment setting and data provided. Mathematically, the performance function to maximize can be formulated as follows:

$$\mathit{perf}\colon \mathit{TI\text{-}Models} \times \mathit{Parameters} \times \mathit{Test\text{-}Subjects} \to \mathbb{R}_{\geq 0};$$

$$\mathit{perf}(\mathit{mod}, \mathit{par}, \mathit{subj}) := \frac{1}{|\mathit{Trials}|} \sum_{i=1}^{|\mathit{Trials}|} \mathit{cP}(\mathit{trial}_i, \mathit{mod}, \mathit{par}, \mathit{subj}),$$

with trial_i being a trial record from the dataset, that is an entry that includes a subject, a task of two stimuli and the stimulus that was recorded to be chosen by the test subject; *Trials* represents an ordered set of training — not testing — trials $\{\mathit{trial}_0, ..., \mathit{trial}_{|trials|-1}\}$ in order of presentation to the subject. *cP* indicates the chance of correct prediction

$$\mathit{cP}(\mathit{trial}_i, \mathit{mod}, \mathit{par}, \mathit{subj}) := p_{mod,par,subj}(\mathit{resp}(\mathit{trial}_i)|\{X_i, Y_i\}),$$

where X_i, Y_i are the stimuli utilized in trial_i and $\mathit{resp}(\mathit{trial}_i)$ the reply given by a subject in trial_i (the stimulus selected by the subject) and $p_{mod,par,subj}$ the probability of an instance of model *mod* that includes associative values or representations resulting from preceding trials given parameters *par* and test subject *subj* with respect to the response stimulus *resp* in trial data entry trial_i and test stimuli $\{X_i, Y_i\}$.

For example, if *mod* is an instance of Internal Linear Array representation, *par* is $\{h = 1\}$, *subj* is a particular wasp with id M140, the trial is trial_1 with stimuli $\{C, D\}$ and $\mathit{resp}(\mathit{trial}_i)$ is C, p has the value 0.5, as yielded by the respective probability equation for an Internal Linear Array, no other values inserted in the representation prior to this trial.[4] If trial_1 is a training trial, that is the model adapts afterward, an identical following trial $\mathit{trial}_2 = \mathit{trial}_1$ will have $\mathit{cP} = 1$, as can be verified using the adaptation rule for Internal Linear Array representation.

The optimization is executed before an evaluation for some model and participant takes place, thus finding the optimal parameter values to achieve best possible performance for a participant over all training trials. This is done by iterating sequentially through all corresponding trials in the data set and comparing the resulting value of *perf* for different parameter values. Notably, the

[3] https://www.scipy.org/

[4] In this example, the value of h is not relevant for the calculation of p.

models perform quite well in testing trails on this optimization mode, employing only training trials to optimize parameters, but evaluating on testing trials merely (training trials do not require TI reasoning in most experiment settings).

3.4 Extending CCOBRA

In order to enable precise fitting, CCOBRA model implementations were extended by additional predict functions that return probability values usually calculated solely for internal use for response predictions in the models (the cP function as defined in the section Optimization Procedure), instead of just prediction responses in the context of an experiment. At the same time, to allow for generalization on integration of varying experiment settings, the prediction and adaptation mechanisms were augmented by the possibility to model multiple successive runs of training and test trials for a creature, instead of just one training phase followed by a test phase. Effectively, the split between a "training phase" and a "test phase" was replaced a single evaluation phase that nevertheless can distinguish between training and test trials and can react to them differently.

Also, an option was implemented that allows to choose whether or not a model adapts during test trials: This is helpful when dealing with both experiment settings where correct replies where shown to subjects after test trials (e.g., Kao et al., 2018) and including some where they where not shown.

Furthermore, as CCOBRA's current prediction accuracy evaluation mechanism was not capable of handling large data sets as the ones used for this investigation, a new mechanism was written that relies less on computer working memory by presenting the output in separate csv–files for each experiment instead of one file that gathers all data. Result visualization is then produced by a separate script iterating through result files.

4 Results

4.1 Overview of model performance in B-D pairs

Below, visualizations of prediction performance for implemented models with respect to species and learning phase intervals are depicted, on data from five sources:

1. Tibbetts, Agudelo, Pandit, and Riojas, 2019: 40 polistes paper wasps of two species, classical visual 5-term TI-task.

2. Camarena, García-Leal, Burgos, Parrado, and Ávila-Chauvet, 2018: 10 pigeons, 1) classical visual 5-term TI-task and 2) overtraining on pair CD and retesting.

3. Jensen, Alkan, Ferrera, and Terrace, 2019: 4 macaques, classical 7-term TI-task with photographs (varied rewards with list position).

4. Jensen, Muñoz, Alkan, Ferrera, and Terrace, 2015: 1) 19 humans, classical visual 7-term TI-task and 2) 3 macaques, same setting but additionally adjacent-pair training in beginning of every session.

5. Kao et al., 2018 Experiment 2: 77 humans, classical visual 7-term TI-task and other experiment settings involving coping with changed list orderings.

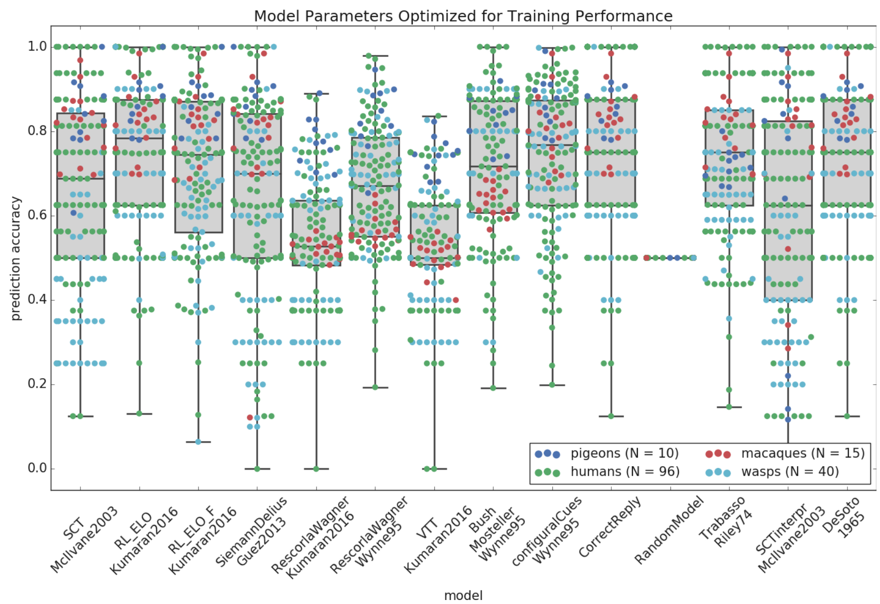


Figure 1: Prediction accuracy for only B-D testing pairs per subject.

B-D trials are chosen as the most indicative ones, because theyare considered to be subject to position effects to a lesser extent than pairs involving exterior stimuliin both 5-item and 7-item list tasks, thus allowing for greatest universal comparability.

4.2 Performance in B-D trials over all data sets

In the table below, the average performance (*perf*) on B-D test trials and ratio of rationally correct responses of models is depicted over all subject, data sets, experiments and species, with parameters fitted per individual subject.

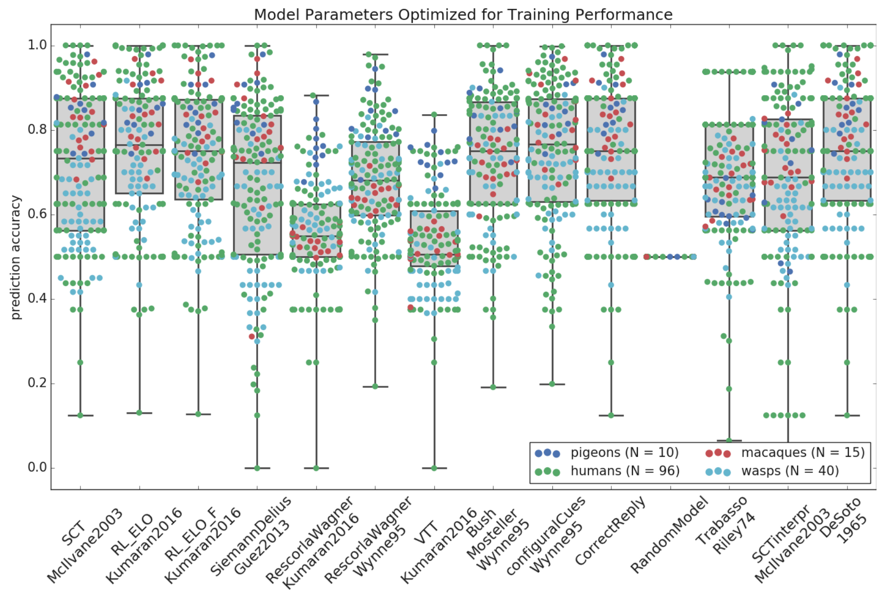


Figure 2: Prediction accuracy for all testing pairs per subject.

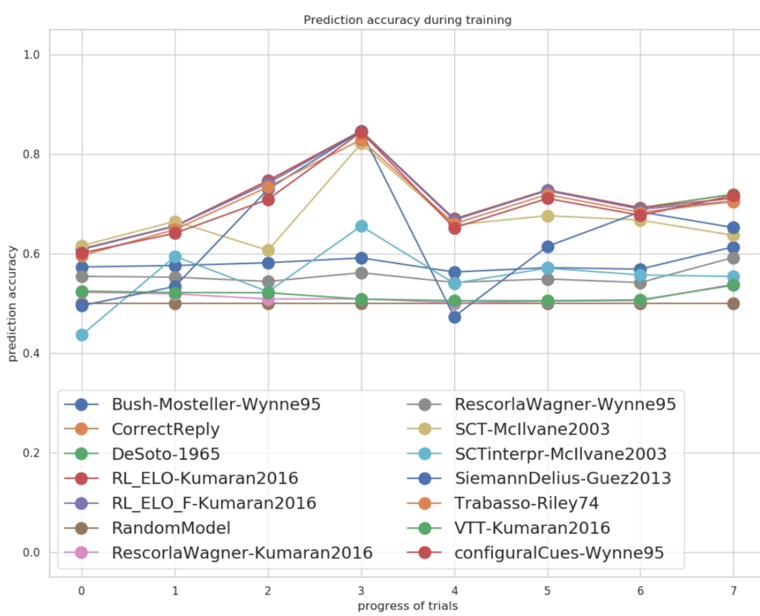


Figure 3: Prediction accuracy for intervals during training on all pairs.

Model name	**Average**	**Correctness**
RL-ELO	0.72	0.99
RL-ELO$_{\mathrm{F}}$	0.71	0.97
VTT	0.60	0.60
Rescorla-Wagner (Wynne)	0.62	0.71
Rescorla-Wagner (Kumaran)	0.57	0.62
Configural Cues	0.70	0.96
Bush-Mosteller	0.63	0.79
Siemann-Delius	0.72	0.93
Internal Linear Array	0.72	0.94
Spacial Paralogic	0.73	1
SCT	0.59	0.95
SCTinterpr	0.55	0.79
Correct Response	0.71	1
Subject Response	1	0.81

Note: In Spacial Paralogic, prediction differs from correct response in training trials only, a number of other models also do not differ significantly.

5 Repository

The program code of this implementation is available at the GKI GitLab[5] together with raw result data from the model evaluations (at `benchmarks/relational/results/`).

References

Bush, R. R., & Mosteller, F. (1955). *Stochastic models for learning.* John Wiley & Sons, Inc.

Camarena, H. O., García-Leal, O., Burgos, J. E., Parrado, F., & Ávila-Chauvet, L. (2018). Transitive Inference Remains Despite Overtraining on Premise Pair C+D-. *Frontiers in Psychology*, *9*.

De Soto, C. B., London, M., & Handel, S. (1965). Social reasoning and spatial paralogic. *Journal of Personality and Social Psychology*, *2*(4), 513 – 521.

Guez, D., & Audley, C. (2013). Transitive or Not: A Critical Appraisal of Transitive Inference in Animals. *Ethology*, *119*(9), 703-726.

Jensen, G., Alkan, Y., Ferrera, V. P., & Terrace, H. S. (2019, July). Reward associations do not explain transitive inference performance in monkeys. *Science Advances*, *5*(7), eaaw2089.

Jensen, G., Muñoz, F., Alkan, Y., Ferrera, V. P., & Terrace, H. S. (2015). Implicit Value Updating Explains Transitive Inference Performance: The Betasort Model. *PLOS Computational Biology*, *11*(9), e1004523.

[5] `https://gkigit.informatik.uni-freiburg.de/borukhsd/transitive-inference-models`

Kao, T., Jensen, G., Michaelcheck, C., Ferrera, V. P., & Terrace, H. S. (2018). Absolute and relative knowledge of ordinal position. *PeerJ Preprints*, 6:e26453v1.

Kao, T., Jensen, G., Michaelcheck, C., Ferrera, V. P., & Terrace, H. S. (2019). Absolute and relative knowledge of ordinal position on implied lists. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, Advance online publication.

Kumaran, D., Banino, A., Blundell, C., Hassabis, D., & Dayan, P. (2016, December). Computations Underlying Social Hierarchy Learning: Distinct Neural Mechanisms for Updating and Representing Self-Relevant Information. *Neuron*, *92*(5), 1135-1147.

Lazareva, O. F., & Wasserman, E. A. (2012). Transitive inference in pigeons: Measuring the associative values of Stimuli B and D. *Behavioural Processes*, *89*(3), 244-255.

McIlvane, W. J., & Dube, W. V. (2003). Stimulus control topography coherence theory: Foundations and extensions. *The Behavior Analyst*, *26*(2), 195–213.

Rescorla, R. A., Wagner, A. R., et al. (1972). A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical Conditioning II: Current Research and Theory*, *2*, 64–99.

Riley, C. A., & Trabasso, T. (1974). Comparatives, Logical Structures and Encoding in a Transitive Inference Task. *Journal of Experimental Child Psychology*, *17*(2), 187-203.

Siemann, M., & Delius, J. D. (1998). Algebraic Learning and Neural Network Models for Transitive and Non-transitive Responding. *European Journal of Cognitive Psychology*, *10*(3), 307-334.

Tibbetts, E. A., Agudelo, J., Pandit, S., & Riojas, J. (2019). Transitive inference in Polistes paper wasps. *Biology Letters*, *15*(5), 20190015.

von Fersen, L., Wynne, C. D., Delius, J. D., & Staddon, J. E. (1991). Transitive inference formation in pigeons. *Journal of Experimental Psychology: Animal Behavior Processes*, *17*(3), 334-341.

Wynne, C. D. L. (1995). Reinforcement accounts for transitive inference performance. *Animal Learning & Behavior*, *23*(2), 207-217.