Recurrent Neural Physics Simulator

Andrew Nam, Joshua Ryu, Jinxiao Zhang
Department of Psychology

Background

Humans build probabilistic models of the world
- Humans receive uncertain sensory information and neural processes have inherent noise [8,10]
- Humans implicitly learn physical laws of motion and form intuitive physics engines [1,2]
- Humans conduct probabilistic mental simulations when reasoning about the world [3,8,9]

Neural Physics Engines
- Previous neural physics engine models only output a single deterministic prediction for each timestep [4-7]
- We allow the network to learn distributions (e.g., Gaussian ~ $N(\mu,\Sigma)$) for predicted states
- Predicted states could be either samples from predicted distributions or the distribution μ

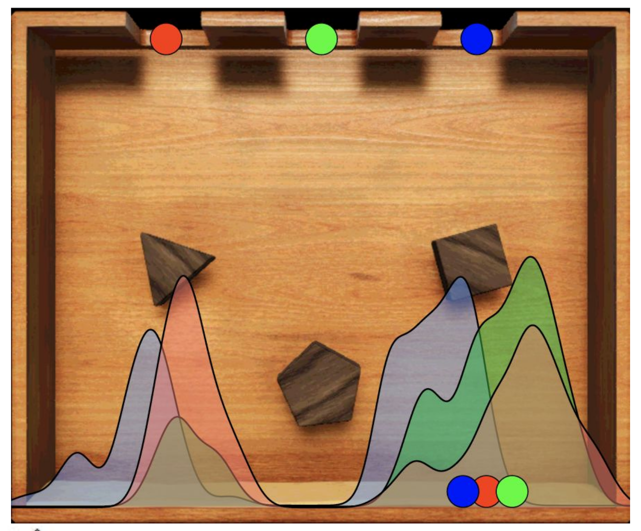
Task and Models

Main Task
- Plinko task: Shown the initial state of the plinko environment, predict the path of ball dropping
- Simulation: $\{x_1, x_2, ..., x_T\}|x_0$

Inputs
- Environment: (x, y, r) for each obstacle
- State at t: position (px, py) and velocity (vx,vy) of ball at time t

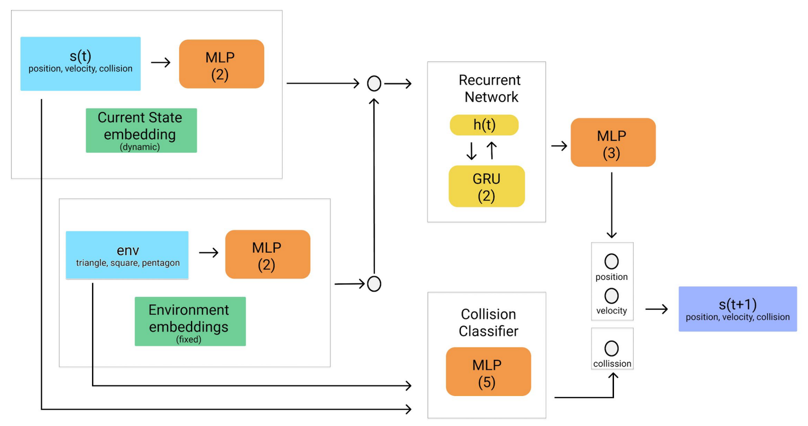
Outputs: position and velocity of ball at time t+1

Loss function:
- mean squared error: $||\text{predicted - target}||^2$
- Cross entropy for collision classifier: $CE(y,\hat{y}) = -\sum_{k=1}^{K} y_k \log \hat{y}_k$

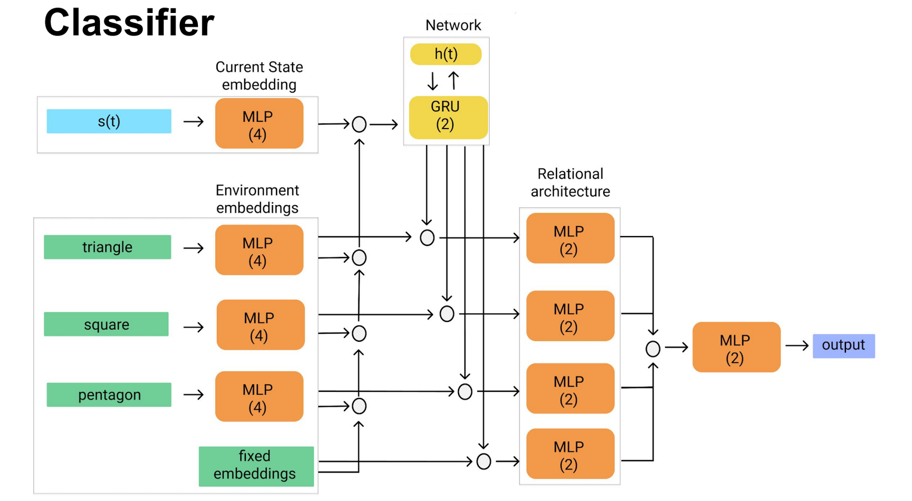


Model Architectures

- **Model 1: Gated Recurrent Unit + Collision Classifier (cGRU)**
  - *Inputs embedder*: 2-layer MLP (ReLU activations)
  - *Recurrent network*: 2 hidden layers GRU
  - *GRU outputs layer*: 3-layer MLP (ReLU activations)
  - Collision classifier: 5-layer MLP (eLU activations)
- **Model 2: Relational GRU (rGRU)**
  - *Inputs embedder*: 4-layer MLP (eLU activations, transferred)
  - *Recurrent network*: 2 hidden layers GRU
  - *Relational layer*: 2-layer MLP (eLU activations)
  - *Outputs layer*: 2-layer MLP (eLU activations)
- **Model 3: rGRU with collision Module (rcGRU)**
  - *Inputs embedder*: 4-layer MLP (eLU activations, transferred)
  - *Collision detector*: 3-layer MLP (eLU activations, transferred)
  - *Recurrent network*: 2 hidden layers GRU
  - *Relational layer*: 2-layer MLP (eLU activations)
  - *Reweighting layer*: relational layer outputs weighted by collision prob
  - *Outputs layer*: 2-layer MLP (eLU activations)

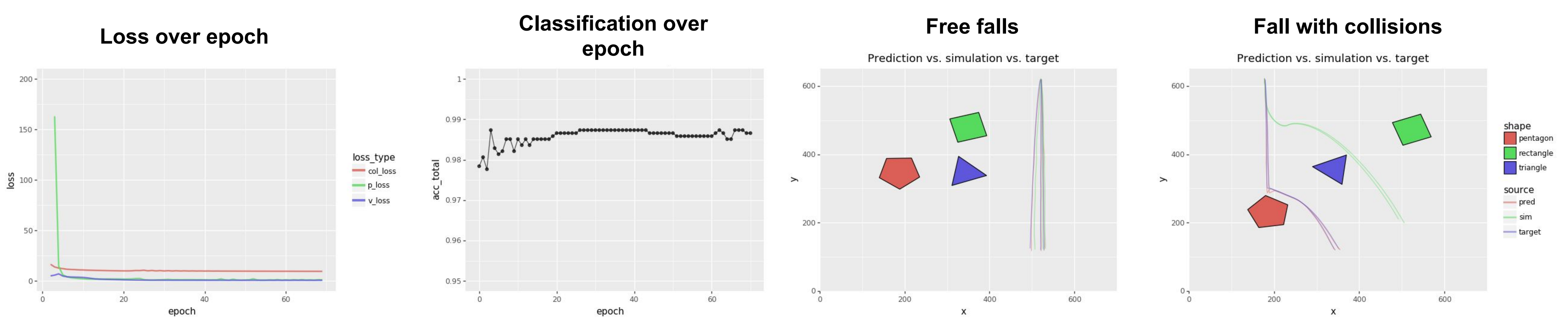


Model 1: cGRU Gated Recurrent Unit + Collision Classifier



Model 2: rGRU relational, recurrent architecture

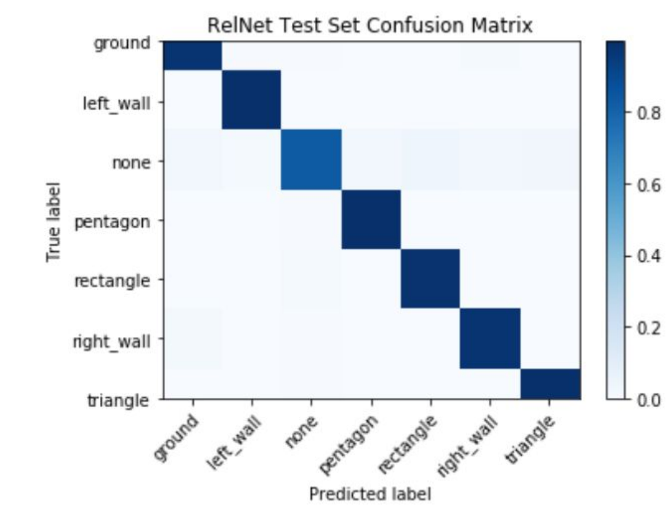
cGRU model results



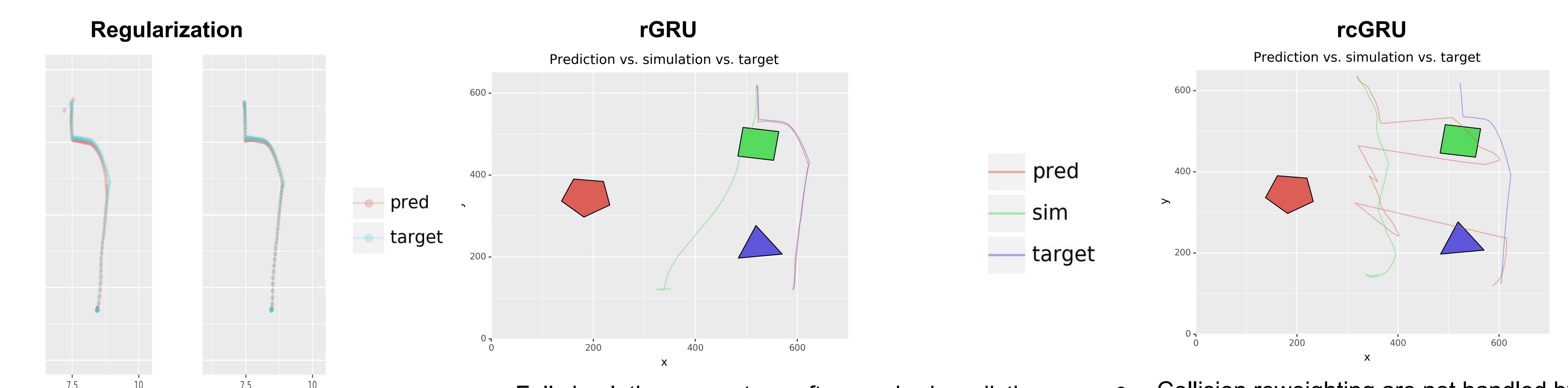
- Model converges quickly: error close to 0 for position and velocity prediction
- Collision classification has high accuracy (98-99%)
  - However, collisions are rare (~ 3%)
- Model prediction of s(t+1)|s(t) works well
- For the whole simulation of s(1), s(2), … s(t) |s(0)
  - Model works well for free fall cases
  - Model fails in collisions cases

Analysis: Collision Handling

- **Question**: Given that the model's predicted variance is relatively constant, is it struggling to detect when collisions occur? (Free falls should have low variance, collisions high)
- **Inputs**: shapes (x, y, r) and ball position and velocity (px, py,vx,vy)
- **Outputs**: 7-way softmax (no collision, left wall, right wall, ground, triangle, square, pentagon)
- **Testing accuracy**: average = **96.5%**; object collisions = **99%**; free fall = **86.1%**
- **Findings**: Need deep architectures for high accuracy, there are still latent variable not accounted for by the model



Relational network (rGRU, rcGRU) results



rGRU, bias regularized

rGRU, bias unregularized

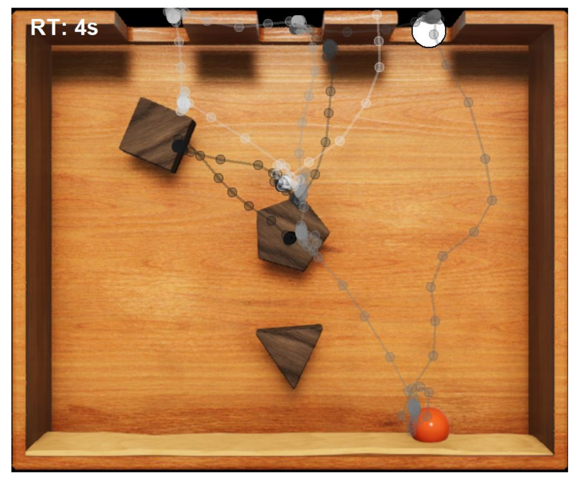
- Full simulations go astray, after one bad prediction
- Collisions are not yet learned (simulation goes through green square)
- The model learns continuity in motion

- Collision reweighting are not handled by the subsequent layers in the given architecture
- At each "collision", the trajectory jumps
- High bias, loses continuity

Discussion

- Physics simulation model learns a notion of continuity, giving smooth trajectories
- The models perform well in free falls but they have **difficulty learning the collisions**
  - This may be due to the more free falls sample in the continuous time series drop
  - This may be overcome guided simulations that simulates collisions more
  - This may be due to discrete sampling of a continuous path.
- Combining various architectural choices may yield better results
  - Feeding outputs from a pre well-trained collision classifier to (r)GRU

Future Directions
- Compare simulation patterns to human eye-tracking data (trajectories, lingering)
- Use reinforcement learning to select informative simulations



Human eye-tracking data

References

[1] Baillargeon et al. (2004). Infants' physical world. *Current directions in psychological science.*
[2] Gerstenberg et al. (2017). Intuitive theories. *Oxford handbook of causal reasoning*.
[3] Gerstenberg et al. (2018). What happened? Reconstructing the past through vision and sound. *CogSci*.
[4] Fragkiadaki et al. (2015). Learning visual predictive models of physics for playing billiards.
[5] Battaglia et al. (2016). Interaction networks for learning about objects, relations and physics. *NIPS.*
[6] Watters et al. (2017). Visual interaction networks: Learning a physics simulator from video. *NIPS.*
[7] Li et al. (2019). Propagation networks for model-based control under partial observation. ICRA.
[8] Smith et al. (2013). Sources of Uncertainty in Intuitive Physics. TiCS.
[9] Battaglia et al (2013). Simulation as an engine of physical understanding. PNAS.
[10] Knill et al. (2004). The Bayesian brain: the role of uncertainty in neural coding and computation.