RTify: Aligning Deep Neural Networks with Human Behavioral Decisions

Yu-Ang Cheng[*1], **Ivan Felipe Rodriguez**[*1], **Sixuan Chen**[1],
Kohitij Kar[2], **Takeo Watanabe**[1], **Thomas Serre**[1]
[1] Brown University [2] York University
{yuang_cheng,ivan_felipe_rodriguez
sixuan_chen1,takeo_watanabe,thomas_serre}@brown.edu
k0h1t1j@yorku.ca

Abstract

Current neural network models of primate vision focus on replicating overall levels of behavioral accuracy, often neglecting perceptual decisions' rich, dynamic nature. Here, we introduce a novel computational framework to model the dynamics of human behavioral choices by learning to align the temporal dynamics of a recurrent neural network (RNN) to human reaction times (RTs). We describe an approximation that allows us to constrain the number of time steps an RNN takes to solve a task with human RTs. The approach is extensively evaluated against various psychophysics experiments. We also show that the approximation can be used to optimize an "ideal-observer" RNN model to achieve an optimal tradeoff between speed and accuracy without human data. The resulting model is found to account well for human RT data. Finally, we use the approximation to train a deep learning implementation of the popular Wong-Wang decision-making model. The model is integrated with a convolutional neural network (CNN) model of visual processing and evaluated using both artificial and natural image stimuli. Overall, we present a novel framework that helps align current vision models with human behavior, bringing us closer to an integrated model of human vision.

1 Introduction

Categorizing visual stimuli is crucial for survival, and it requires an organism to make informed decisions in dynamic and noisy environments. This critical aspect of visual perception has driven the development of computational models to understand and replicate these processes. Traditionally, the field has followed two distinct paths.

On the one hand, image-computable vision models are used to predict behavioral decisions during (rapid) visual categorization tasks ranging from models of early- [1–3], mid- [4–6] and high-level vision [7–9] (see [10] for a review). More recently, these earlier models were superseded by deep convolutional neural networks (CNNs), which have become the de-facto choice for modeling behavioral decision [11–14]. Models are typically evaluated by estimating confidence scores computed for individual images, which are then correlated with similar scores derived for human observers(such as the proportion of correct human responses for each image). Such metrics ignore human reaction times (RTs); hence, current vision models only partially account for human decisions.

On the other hand, decision-making models have been used to explain how visual information gets integrated over time – predicting behavioral choices and RTs jointly. Notably, mathematical models, exemplified by evidence accumulation models such as the drift-diffusion [15–17] and linear ballistic

[*]These authors contributed equally to this work.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-4147

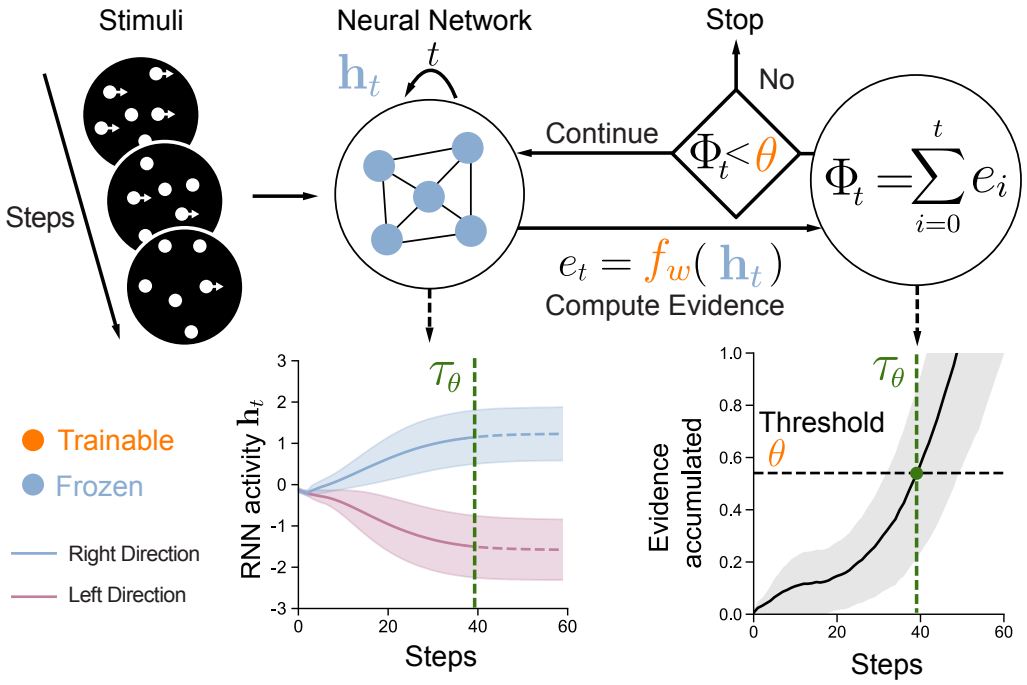


Figure 1: **Illustration of our RTify method.** The input is a visual stimulus represented by random moving dots, but the model can also accommodate color images and video sequences. We take a pretrained task-optimized RNN and use a trainable function f_w to transform the activity of the network into a real-valued evidence measure, e_t, that will be integrated over time by an evidence accumulator, Φ_t. When the evidence accumulator reaches the threshold θ, processing stops, and a decision is taken. The time step at which the accumulated evidence passes this threshold τ_θ is taken as the model RT for this stimulus.

accumulators [18, 19], have been quite successful in modeling an array of behavioral data (see [20] for a review). In addition, mechanistic models, including the Wong-Wang (WW) model, have provided insights into the underlying neural mechanisms [21, 22]. However, these efforts have primarily relied on traditional psychophysics tasks using simple, artificial stimuli, such as Gabor patterns [23] and random moving dots [24]. Beyond these easily parameterizable stimuli, these models have not been extended to deal with more complex, natural stimuli.

While both vision and decision-making models have contributed distinctively to our understanding of visual processes, a complete understanding of human vision will require their integration to explain the whole dynamics of visual decision-making. Recent neuroscience studies have leveraged recurrent neural network (RNN) models as the starting point for this integration [25] (see [26] for a review). More generally, recurrent processing has been shown to be necessary to account for both behavioral and neural recordings in object recognition tasks [27, 28].

Two promising approaches have been described that leverage RNNs to bridge the gap between decision-making and vision models [29, 30]. In [29], an RNN was trained to solve a visual classification task using the backpropagation-through-time (BPTT) learning algorithm. The entropy of the RNN outputs is computed at each timestep and used as a proxy for the network confidence. A decision is taken when the entropy reaches the threshold, halting further recurrent computation. In this approach, the recurrence steps are not differentiable, which prevents the use of gradient methods and inherently limits the complexity of the corresponding decision function. The resulting model may have difficulty predicting the entire distribution of RTs. Besides, the method is only applicable when human RTs are available because it requires an extensive search for the correct threshold value to fit human RT data.

An alternative approach was described in [30]. In [30], a convolutional RNN was trained on a visual classification task using contractor recurrent back-propagation (C-RBP). Besides, instead of searching for an optimal threshold to fit human RT data, a surrogate time-evolving 'uncertainty'

metric was estimated with evidential deep learning [31]. In this framework, model outputs are treated as parameters of a Dirichlet distribution, with the width of the distribution reflecting uncertainty. Remarkably, the resulting approach fits a range of experimental data well, without any supervision from human data. It is true that uncertainty and RTs are tightly coupled and are both affected by task difficulty. However, uncertainty and RTs are conceptually different. Experimental results show that uncertainty and RTs can be positively or negatively correlated [32], and even double dissociated [33] under different experimental conditions. Therefore, a good model of uncertainty is not guaranteed to be a good model of RTs.

Here, based on extensive cognitive neuroscience research [34–37], we introduce a novel trainable module called RTify to allow an RNN to dynamically and nonlinearly accumulate evidence. First, this module can be trained to learn to make human-like decisions using direct human RT supervision. Our results suggest that incorporating a dynamic evidence accumulation process, compared to the entropy heuristics used in [29], can help better capture human RTs. Second, we show how the same general approach can also be used to train an RNN to learn to solve a task with an optimal number of time steps via self-penalty. Our results show that human-like RTs naturally emerge from such ideal-observer models without explicit supervision from human RT data. Hence, our framework is general enough to allow the fitting of human RT data as done in [29] and/or the training of a neural architecture that can optimally trade speed and accuracy via self-penalty as done in [30].

Contributions: Overall, our work makes the following contributions: **(i)** We present RTify, a novel computational approach to optimize the recurrence steps of RNNs to account for human RTs. This enables dynamic nonlinear evidence accumulation learned through back-propagation (a) to fit human data or (b) optimally balance speed and accuracy. **(ii)** We comprehensively demonstrate the effectiveness of our framework for modeling human RTs across a diverse range of psychophysics tasks and stimuli. Our method consistently outperforms alternatives with and without explicit training on human data. **(iii)** As an illustrative example of the framework's potential, we extend the WW decision-making model [22] to create a biologically plausible, multi-class compatible, and fully differentiable RNN module. We show that the enhanced neural circuit can be used as a drop-in module for CNNs to fit human RTs.

2 RTify: Overview of the method

First, we explain how our RTify module is applied to a pre-trained RNN. Then, we will explain how to tune a deep-learning RNN-based implementation of the WW model to RTify feedforward networks.

We start with a task-optimized RNN with hidden state $\mathbf{h}_t$, which remains frozen. We then train a learnable mapping function $f_w : \mathbf{R}^k \to \mathbf{R}$ that summarizes the state of the neural population at each time step t by mapping the RNN hidden state $\mathbf{h}_t$ to some "evidence": $e_t = f_w(\mathbf{h}_t)$. At every time step, the evidence is integrated via an "evidence accumulator" $\Phi_t = \sum_{i=1}^{t} e_i$, and when the accumulated evidence passes a learnable threshold θ, the model is read out, and a decision is made. The time step at which the accumulated evidence first passes this threshold is given by $\tau_\theta(\Phi) = \min\{t : \Phi_t > \theta\}$, and is treated as model RTs. In summary, $\tau_\theta(\Phi)$ is directly influenced by the threshold θ and by w through Φ_t.

To align the model RTs with human RTs, or to penalize the model for excessive time steps, we need to optimize a loss function over $\tau_\theta(\Phi)$. In the most general case, we first consider $F(\tau_\theta(\Phi))$ as our loss function to illustrate how we approximate its gradient. Since our goal is to minimize F, we will need to calculate the gradient $\frac{\partial F(\tau_\theta(\Phi))}{\partial \theta}$ and $\frac{\partial F(\tau_\theta(\Phi))}{\partial w}$. Following the chain rule, we get

$$\frac{\partial F(\tau_\theta(\Phi))}{\partial w} = \frac{\partial F(\tau_\theta(\Phi))}{\partial \tau_\theta(\Phi)} \cdot \frac{\partial \tau_\theta(\Phi)}{\partial w}, \frac{\partial F(\tau_\theta(\Phi))}{\partial \theta} = \frac{\partial F(\tau_\theta(\Phi))}{\partial \tau_\theta(\Phi)} \cdot \frac{\partial \tau_\theta(\Phi)}{\partial \theta}. \tag{1}$$

This means a crucial step involves estimating the gradient of $\tau_\theta(\Phi)$ over the trainable parameters w and θ of the RTify.

The primary challenge that arises when extracting the gradient of $\tau_\theta(\Phi)$ over the trainable parameters w and θ is the non-differentiability of $\tau_\theta(\Phi)$, which prevents the direct use of the backpropagation algorithm. This is because $\tau_\theta(\Phi)$ lies in the integer space and requires non-differentiable operations

 https://doi.org/10.52202/079017-4147

such as the minimum function and the inequality. We will relax the original formulation to circumvent this issue, enabling us to approximate the gradient.

Assume that $\Phi(t)$ is a continuous function on the closed interval $[1, N]$, representing the accumulation of evidence over time. Define $\tau_\theta(\Phi) = \min\{t \in [1, N] : \Phi(t) > \theta\}$, which is the earliest time when $\Phi(t)$ exceeds the threshold θ.We can use a first-order Taylor expansion to find the following approximation:

$$\frac{\partial \tau_\theta}{\partial w} \approx \frac{\partial \tau_\theta^*}{\partial w} \approx \frac{1}{\Phi_t - \Phi_{t-1}} \cdot (-\frac{\partial \Phi_t}{\partial w}), \frac{\partial \tau_\theta}{\partial \theta} \approx \frac{\partial \tau_\theta^*}{\partial \theta} \approx \frac{1}{\Phi_t - \Phi_{t-1}} \tag{2}$$

Fig. S1 provides a visual intuition for the approximation, and the full derivation can be found in the SI A.1. This leads to the following gradients for our trainable parameters:

$$\frac{\partial F(\tau_\theta(\Phi))}{\partial w} \approx \frac{\partial F(\tau_\theta(\Phi))}{\partial \tau_\theta(\Phi)} \cdot \frac{1}{\Phi_t - \Phi_t} \cdot (-\frac{\partial \Phi_t}{\partial w}), \frac{\partial F(\tau_\theta(\Phi))}{\partial \theta} \approx \frac{\partial F(\tau_\theta(\Phi))}{\partial \tau_\theta(\Phi)} \cdot \frac{1}{\Phi_t - \Phi_{t-1}}. \tag{3}$$

Since F and Φ are both differentiable by nature, the gradients in Eqs. 2 and 3 are all computable after this approximation.

Under this framework, we consider two different scenarios: Training with human data directly ("supervised"; see Section 2.1) or training with "self-penalty", which involves no explicit human data but uses a penalty term that spontaneously leads to decision times similar to human RTs (e.g., achieving an optimal speed-accuracy trade-off; see Section 2.2).

2.1 Predicting human decisions with direct "supervision"

Human behavioral decisions in the random dot motion (RDM) tasks used in decision-making studies [16, 24, 38] are typically summarized as histograms similar to those shown in Fig. 2. Here, histograms are computed for RTs for correct and incorrect trials corresponding to individual experimental conditions (such as coherence levels shown here; see section 3 for details). Moreover, RTs for incorrect trials are turned into negative RTs. Combined with correct RTs (which stay positive), one single histogram is used for capturing both accuracy (the proportion of positive values) and RTs. To measure the goodness of fit between human RTs and model RTs, we use a mean squared error loss (MSE) between histograms of model RTs and human RTs.

In the object recognition task [39], only RTs averaged across all participants were available. We can match human data on a stimulus-by-stimulus basis using the negative correlation loss between model and human RTs.

2.2 Predicting human decisions with "self-penalty"

Our framework allows us to develop an "ideal-observer" RNN model explicitly trained to balance the computational time required for solving a particular classification task and its own accuracy for the task, i.e., a speed-accuracy trade-off.

To achieve this, we add a regularizer to the cross-entropy loss to encourage the RNN to jointly maximize task accuracy while minimizing the computational time needed to solve the task. With l as the output logits of the network, $\hat{y}$ as the output probabilities of the network, and y as the ground truth, we can write our penalty term for a single sample as:

$$L_{\text{self-penalized}} = L_{CCE}(y, \hat{y}) + \lambda \left(\mathbf{l_y} \cdot \tau_\theta\right) \tag{4}$$

where λ is a hyperparameter for controlling the strength of the penalty, $\mathbf{l_y}$ refers to the logit value of the correct label, τ_θ is the model decision time. This penalty means that the model will be penalized for using too much time, especially for higher confidence (higher $\mathbf{l_y}$).

2.3 RTifying feedforward networks

To integrate temporal dynamics into feedforward neural networks (e.g., CNNs), we describe an RNN module that approximates the WW neural circuit model [22]. The original WW model is a biophysically-realistic neural circuit model of two-alternative forced choices via the temporal

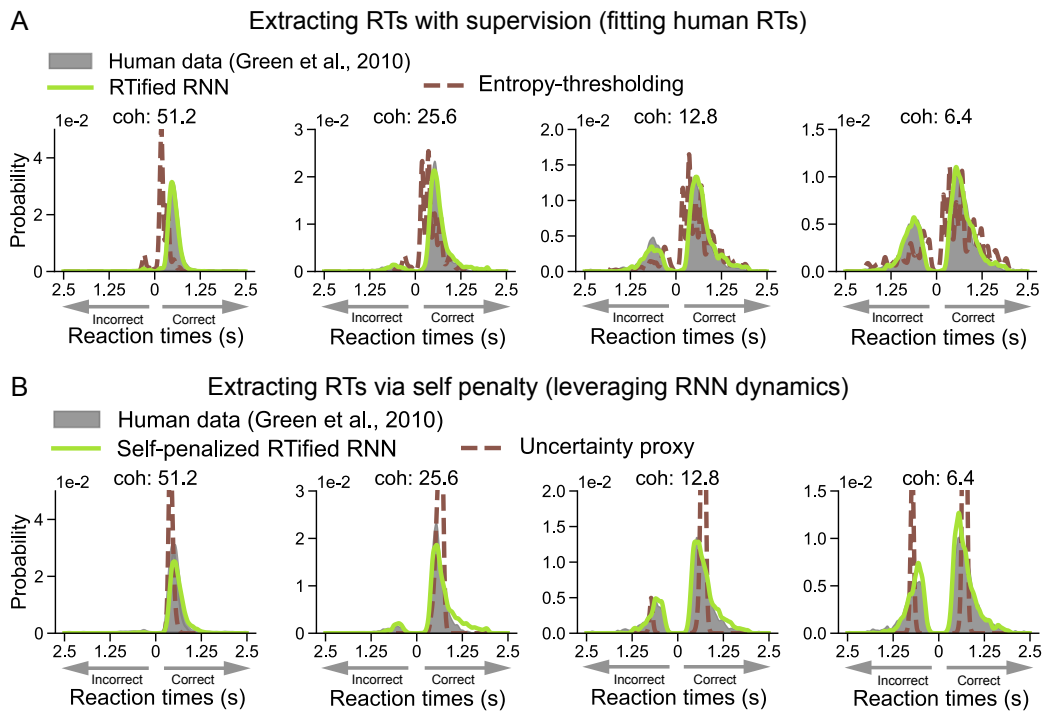


Figure 2: **RTified model evaluation on a RDM task [24]**. Human data are shown as a gray shaded area, and model fits are shown for **(A)** the "supervised" setting where human behavioral responses are used to train the models and **(B)** the "self-penalized" setting where no human data is used. Our approach (green) outperforms the two alternative approaches (brown), i.e., entropy-thresholding [29] for the "supervised" and uncertainty proxy [30] for the "self-penalized" settings (see Fig. 4 for MSE comparisons and Fig. S3 for all coherences).

accumulation of sensory evidence in two distinct neural populations. It takes a constant scalar input (representing a stimulus parameter such as the degree of coherence for randomly moving dot stimuli). It outputs an RT when the activity of either population reaches a decision threshold.

However, the original WW model has limitations: its parameters must be manually tuned to fit observed data, and it is restricted to binary classification tasks with simple artificial parametric stimuli (e.g., Gabor patterns). To overcome these limitations, we extend the WW model. First, we replace the scalar input with a feedforward neural network (e.g., a CNN), enabling the model to process complex stimuli such as natural images. Second, we generalize the model from two populations to M populations, effectively increasing the number of neural populations to handle multi-class classification problems. Third, we RTify the model to make all parameters trainable via backpropagation. This allows the model to automatically learn the optimal parameters to fit human RTs (see Fig. 3 for an illustration of the multi-population case and SI Fig. A.2 for an illustration of the two-population case).

3 Experiments

In this section, we validate our RTify framework on two psychophysics datasets: the RDM dataset [24, 38] and a natural image categorization dataset [39]. As a side note, all models were trained on single Nvidia RTX GPUs (Titan/3090/A6000) with 24/24/48GB of memory each. All training can be completed in approximately 48 hours. Code and data are available at `https://github.com/Yu-AngCheng/RTify`.

https://doi.org/10.52202/079017-4147

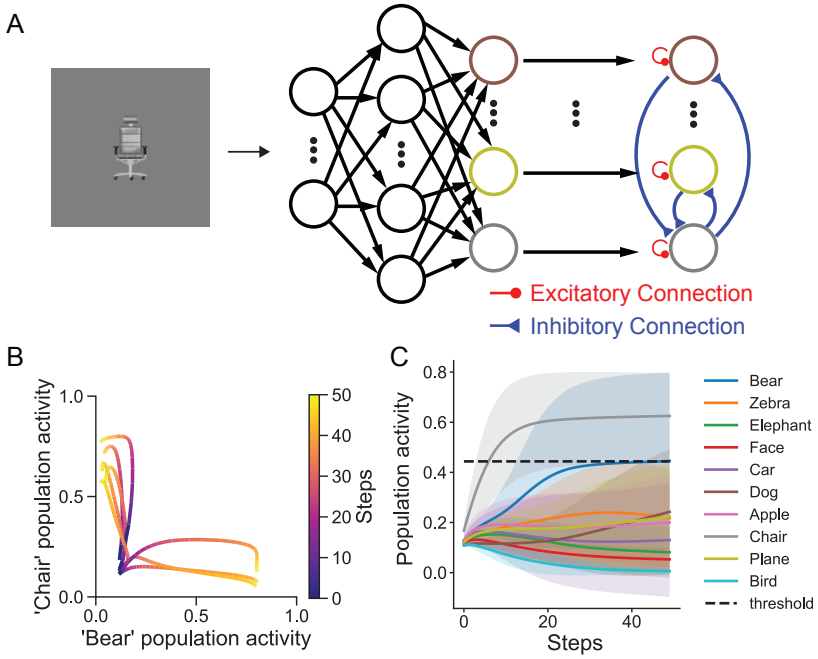


Figure 3: **Illustration of RTifying feedforward neural networks**. We develop a multi-class compatible and fully differentiable RNN module based on the WW model [21, 22]. This module is implemented as an attractor-based RNN, and is stacked on top of a feedforward neural network. The feedforward neural network first takes an image as the input. Outputs from classification units of the network are then sent to RTified WW (**A**). Information is accumulated by multiple populations of neurons in RTified WW while they compete with each other (**B**). A decision is made and the process stops when one of the populations reaches a threshold. The number of time steps needed for the RTified WW to reach the threshold is used to predict human RT (**C**).

3.1 Random dot motion task

The RDM task is a classic experimental paradigm used to test temporal integration that has been extensively used in psychophysics [40], human imaging [41], and electrophysiology studies [42]. The stimuli in this task consist of dots moving on a screen toward a predefined direction vs. randomly. For each time step, each dot only has a specific probability (coherence) ($0.8\%, 1.6\%, 3.2\%, 6.4\%, 12.8\%, 25.6\%$, or 51.2%) to move towards the pre-defined direction, making the task non-trivial. The participants must integrate motion information across time and report it when they are sufficiently confident. The original experimental data are from [24, 38], where 21 young adult participants performed around 40,000 trials in total over 4 consecutive days.

First, we trained an RNN consisting of 5 convolutional blocks (Convolution, BatchNorm, ReLU, Max pooling) and a 4096-unit LSTM with BPTT. In the original experiment, the stimuli were shown on a 75 Hz CRT monitor for up to 2 seconds, and therefore, we also trained our RNN for the RDM stimuli for 150 frames. The RNN was trained for 100 epochs using the Adam optimizer with a learning rate of 1e-4 at full coherence (c = 99.9%) for the first 10 epochs as a warm-up and 1e-5 at all coherence levels for the remaining 90 epochs. Next, we trained our two different RTify modules. For fitting human RTs, it was trained for 10,000 epochs, and for self-penalty, it was trained for 20,000 epochs. In both cases, the Adam optimizer were used, and the weights of the task-optimized RNN were frozen while training the RTify modules.

We trained the first RTify module to predict human RTs by fitting human RT distributions. Here, positive RTs refer to RT with correct choices, and negative RTs refer to RTs with incorrect choices. Therefore, one distribution incorporates both speed and accuracy information from behavioral choices. Results are shown in Fig. 2. Our RTify model can predict the full RT distribution across all coherence levels. In comparison, the entropy-thresholding approach by [29] fails to capture the full distribution (see Fig. 2 for coherence = 51.2% to 6.4% and Fig. S3 for all coherences). Importantly, our method

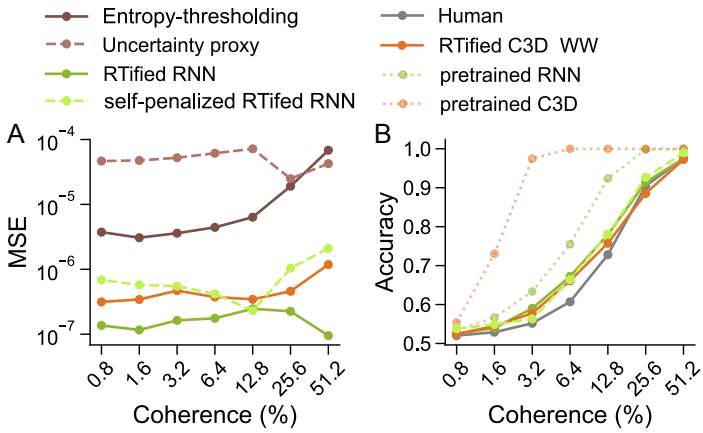


Figure 4: **(A) MSE comparisons for the RDM task [24] for all coherence levels.** The RTified model trained in the "supervised" setting (i.e., with human behavioral responses; green solid line) performs better (lower MSE) than entropy-thresholding [29] (brown solid line) under all coherence levels. Similarly, the RTified model trained in the "self-penalized" setting (i.e., without human data; green dash line) performs better than uncertainty proxy [30] (brown dash line). With the help of our RTified WW module (orange solid line), a convolution neural network (C3D) can also fit the data better than entropy-thresholding [29]. **(B) Classification accuracy comparisons between pretrained and RTified models for the RDM task [24].** The RTified model trained with human RTs data in the "supervised" setting (green solid line) and in the "self-penalized" setting (green dash line) achieve human-like classification accuracy under all coherence levels compared with the pretrained model without RTify (green dotted line). With the help of our RTified WW module (orange solid line), a CNN (C3D) matches human accuracy better than the pretrained model without RTify (orange dotted line).

surpasses entropy-thresholding approach [29] (two-sided Wilcoxon signed-rank test, $p < .05$; for MSE comparisons, see Fig. 4).

It is well known that there is a trade-off between RTs and accuracy in cognitive tasks. To investigate this relationship further, we extended our analysis to examine whether the model's accuracy would approximate human accuracy when it was fitted solely on human RTs without using human accuracy data. Given that conventional RT distributions encompass both RTs and accuracy information, we restricted our fitting procedure to the positive part of the distribution. That is, RTs corresponding to correct responses to prevent inadvertently incorporating human accuracy into the model. Remarkably, by fitting the model on human RTs only, the model naturally reached a classification accuracy comparable to human performance (see Fig. 4).

We also trained a self-penalized RTify module without any human data. This ideal-observer RNN model was trained to minimize the time steps needed for solving the RDM task (see section 2.2 for details). Human-like RTs emerge naturally in this neural network (See Fig. 2). Our model predicts RT data much better than previous approaches, which use a measure of uncertainty computed over the RNN as a surrogate metric. As can be seen, the resulting model tends to overfit the modes of the distribution.(see Fig. 2 for coherence = 51.2% to 6.4% and Fig. S3 for all coherences). Our method surpasses uncertainty proxy approach [30] (two-sided Wilcoxon signed-rank test, $p < .05$, for MSE comparison, see Fig. 4). We also checked the model classification accuracy before and after self-penalized RTify. Interestingly, we also found the self-penalized RTified RNN demonstrated a human-like classification accuracy (see Fig. 4).

3.2 Object recognition task

Unlike previous visual decision-making models, we want to show that our method can also be applied to natural images and multi-class datasets. Specifically, we consider an object recognition task, a classic paradigm used extensively in computer vision [43, 44] and cognitive neuroscience studies [45, 46]. The original data is from [39], where 88 participants perform the task through Mturk. The stimuli in this task belong to 10 categories, and for each category, there are 20 natural images

https://doi.org/10.52202/079017-4147

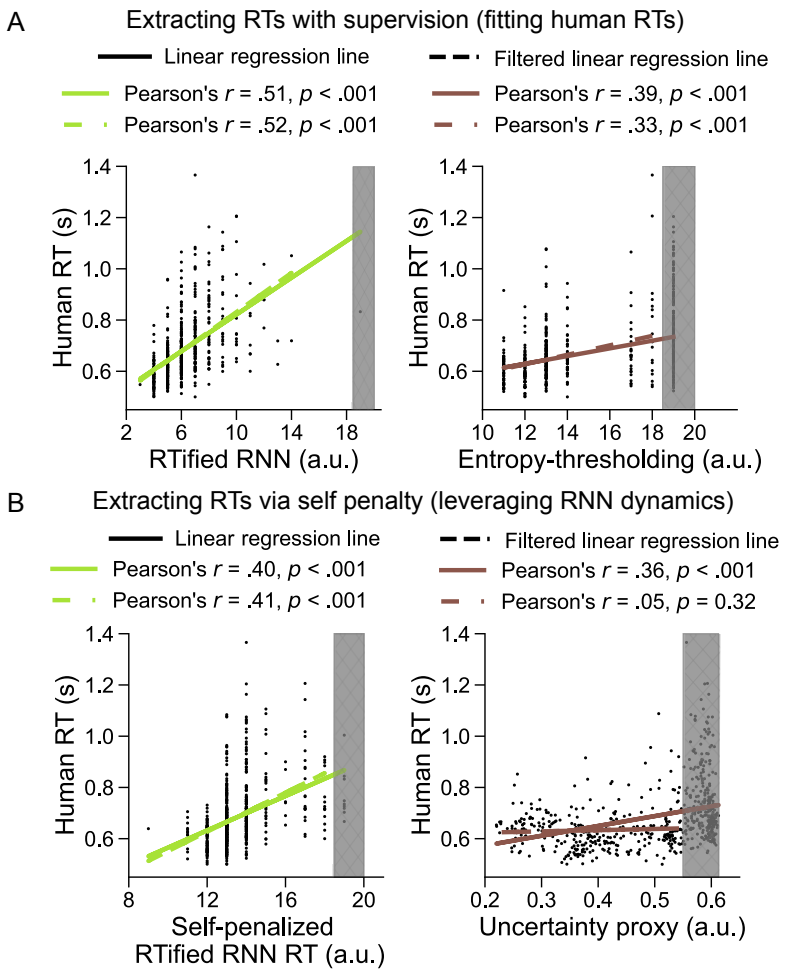


Figure 5: **RTified model evaluation on an object categorization task [39].** Model vs. human RT predictions for our RTified model (green) vs. alternative approaches (brown) **(A)** in the "supervised" setting where human behavioral responses are used to train the model and **(B)** the "self-penalized" setting where no human data is used. Solid lines are linear regression fits between model and human RTs. Crossed-shaded areas and the dashed lines are controls to show the fits after removing the highest model RTs. Our approach outperforms the two alternative approaches, i.e., entropy-thresholding [29] for the "supervised" setting and uncertainty proxy [30] for the "self-penalized" setting.

taken from the COCO dataset [47] and 112 synthetically generated images with different backgrounds and object positions.

We first train our RNN with BPTT to perform a 10-way classification task. In the original study, participants performed a binary classification task. However, since the individual binary pairs were not saved, we trained the model in a 10-class classification task. We used Cornet-S [48] pretrained on the Imagenet dataset [43] because it was used in the original study and it achieves a relatively high brainscore in terms of explaining neural activities [49]. We trained the network for 100 epochs, using the Adam optimizer with a learning rate of 1e-4 and a learning scheduler (StepLR) with a step size of 2,000. We used 20 timesteps (instead of 2 in the original model) for the IT layer in the Cornet to achieve high temporal resolution. Results show that our RNN achieves 75.2% on a held-out test set. Similarly, we trained our two different RTify modules. For fitting human RTs, it was trained for 100,000 epochs, while for self-penalty, it was trained for 10,000 epochs with a learning scheduler (StepLR) with a step size of 2,000 and a gamma of 0.3. In both cases, Adam optimizers were used, and the weights of the task-optimized RNN were frozen while training the RTify modules.

We then extracted RTs from our RNN to fit human RTs. Notably, model RTs significantly correlate with the human RT observed in the psychophysics experiment ($r = .51, p < .001$). Besides, our

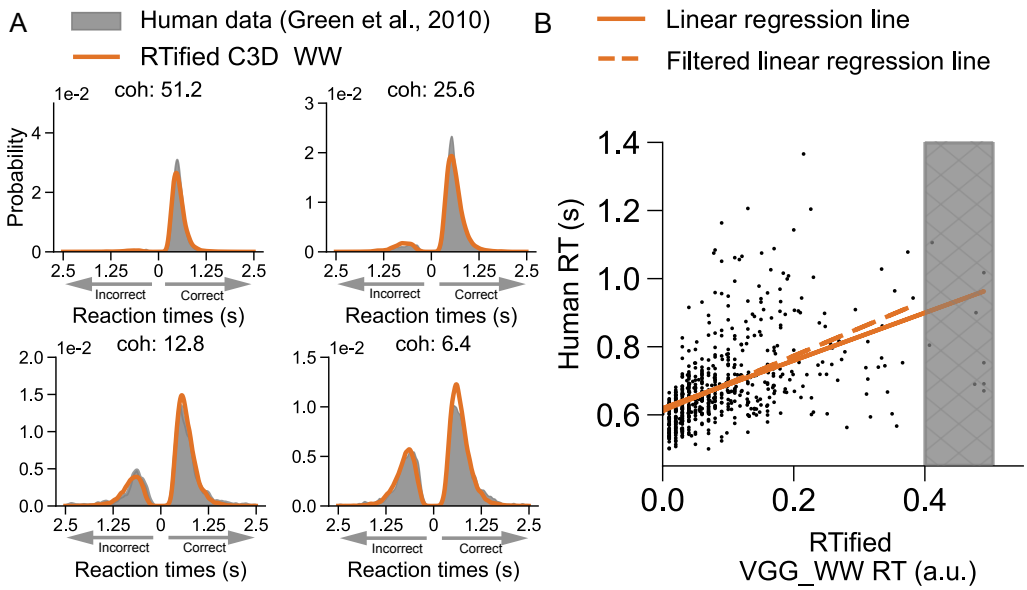


Figure 6: **RTified WW model evaluation.** We combine our RTified WW module with **(A)** a 3D CNN to fit human RTs collected in an RDM task [24] (see Fig. 4 for MSE comparisons with other methods) and **(B)** a VGG to fit human RTs in a rapid object categorization task [39] (Crossed-shaded areas and the dashed lines are controls to show the fits after removing the highest model RTs).

method also surpasses entropy-thresholding [29] (bootstrapping shows that our method is superior to theirs with a probability of 99.9%). We also extract RTs from an ideal-observer RNN model trained with a time self-penalty (see section 2.2 for details). We show here that model RTs are significantly correlated with human RTs (see Fig. 5, $r = .40, p < .001$). This failed to be captured by uncertainty proxy [30]. We argue this in two ways. First, bootstrapping shows that our method is superior to theirs with a probability of 87.1%. Second, and most importantly, although the uncertainty seems to correlate with human RTs in the dataset ($r = .36, p < .001$), it only applies to high-uncertainty cases. When trials with high model uncertainty are excluded, uncertainty shows no significant correlation with human RTs ($r = .05, p = 0.32$). However, using our method, the correlation remains strong and significant ($r = .41, p < .001$).

3.3 RTifying feedforward neural networks

Given the prevalence of feedforward networks (e.g. CNNs) and their incredible performance in visual tasks, a natural question is how to align such networks in the temporal domain of decision-making. We thus developed a biologically plausible, multi-class, differentiable RNN module based on the WW recurrent circuit model [21, 22]. This module can be stacked on top of any neural network, even if not recurrent, and can be used to align model RTs with human RTs.

For the RDM task, we take a 3D CNN with 6 convolutional blocks (convolution, BatchNorm, ReLU, Max pooling) and an MLP. We train the network for 100 epochs using the Adam optimizer with a learning rate of 1e-4 at full coherence (c = 99.9%) for the first 10 epochs as a warm-up and 1e-5 at all coherence levels for the remaining 90 epochs. Since this model is not an RNN, it has no temporal dynamics. Therefore, we drop in our WW module and further train the WW module for 5,000 epochs using the Adam optimizer with a learning rate of 1e-4, a StepLR scheduler with a step size of 1,000 and gamma of 0.3, and a grad clip at 1e-5. Interestingly, when we RTify the C3D model using the WW module, it is able to capture the distribution of human RTs across all coherence levels, see Fig. 6A for coherence = 51.2% to 6.4% and Fig. S4 for all coherences, for MSE comparison see Fig. 4). Furthermore, we also observed a human-like classification accuracy for the model when it is solely trained to fit human RTs but not human accuracy (see Fig. 4).

Similarly, we take a VGG-19 pre-trained on Imagenet for the object recognition task and fine-tune it on the dataset provided by Kar et al. [39] in a 10-class classification way for the abovementioned reasons. We train the model for 100 epochs using a batch size of 32. The optimizer was AdamW,

https://doi.org/10.52202/079017-4147

with a learning rate of 1e-5. We use a OneCycleLR scheduler adjusted after 10 epochs of warm-up. Results show that our RNN achieves 81.6% on a held-out test set. We further train the WW module for 100,000 epochs using the Adam optimizer with a learning rate of 1e-4 and a grad clip at 0.0001. By using the multi-class version of the WW model, we show that the RTify VGG also exhibited a significant correlation with human data ($r = .49, p < .001$, see Fig. 6B).

4 Conclusion

We have described a computational framework to train RNNs to learn to dynamically accumulate evidence nonlinearly so that decisions can be made based on a variable number of time steps to approximate human behavioral choices, including both decisions and RTs. We showed that such optimization can be used to fit an RNN directly to human behavioral responses. We also showed that such a framework can be extended to an ideal-observer model whereby the RNN is trained without human data but with self-penalty that encourages the network to make a decision as quickly as possible. Under this setting, human-like behavioral responses naturally emerge from the RNN – consistent with the hypothesis that humans achieve a speed-accuracy trade-off. Finally, we provided an RNN implementation of a popular neural circuit decision-making model, the WW model, as a trainable deep learning module that can be combined with any vision architecture to fit human behavioral responses. Our computational framework provides a way forward to integrating image-computable models with decision-making models, advancing toward a more comprehensive understanding of the brain mechanisms underlying dynamic vision.

Limitations Certain limitations will need to be addressed in future work. Most of the human data used in our study remains relatively small-scale and is limited primarily to synthetic images because more naturalistic benchmarks only include behavioral choices [44, 50] and lack RT data. To properly evaluate our approach and that of others, larger-scale psychophysics datasets using more realistic visual stimuli will be needed. There is already evidence that large-scale psychophysics data can be used to effectively align AI models with humans [51, 52]. We hope this work will encourage researchers to collect novel internet-scale benchmarks that include both behavioral choices and RTs.

Broader Impacts As AI vision models become more prevalent in our daily lives, ensuring their trustworthy behavior is increasingly important [53, 54]. Our framework contributes to this effort by exploring how to align certain aspects of models' behavior with human responses in specific contexts. While our approach is limited to predicting RT distributions, it constitutes a first step toward more human-aligned AI models.

Acknowledgments and Disclosure of Funding

This work was supported by NSF (IIS-2402875), ONR (N00014-24-1-2026) and the ANR-3IA Artificial and Natural Intelligence Toulouse Institute (ANR-19-PI3A-0004) to T.S and National Institutes of Health (NIH R01EY019466 and R01EY027841) to T.W. Computing hardware was supported by NIH Office of the Director grant (S10OD025181) via Brown's Center for Computation and Visualization (CCV).

References

[1] Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision **42**(3) (2001) 145–175

[2] Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology **160**(1) (1962) 106

[3] Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on pattern analysis and machine intelligence **20**(11) (1998) 1254–1259

[4] Jagadeesh, A.V., Gardner, J.L.: Texture-like representation of objects in human visual cortex. Proceedings of the National Academy of Sciences **119**(17) (2022) e2115302119

[5] Doshi, F.R., Konkle, T., Alvarez, G.A.: A feedforward mechanism for human-like contour integration. bioRxiv (2024) 2024–06

[6] Parthasarathy, N., Simoncelli, E.P.: Learning a texture model for representing cortical area v2. In: Computational and Systems Neuroscience (CoSyNe), Denver, CO (Feb 2020)

[7] Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. Nat Neurosci **2**(11) (Nov 1999) 1019–1025

[8] Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(3) (2007) 411–426

[9] Biederman, I.: Recognition-by-components: a theory of human image understanding. Psychological review **94**(2) (1987) 115

[10] Crouzet, S.M., Serre, T.: What are the visual features underlying rapid object recognition? Frontiers in Psychology **2** (2011)

[11] Kheradpisheh, S.R., Ghodrati, M., Ganjtabesh, M., Masquelier, T.: Deep networks can resemble human feed-forward vision in invariant object recognition. Scientific Reports **6**(1) (September 2016)

[12] Eberhardt, S., Cader, J.G., Serre, T.: How deep is the feature analysis underlying rapid visual categorization? In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., eds.: Advances in Neural Information Processing Systems. Volume 29., Curran Associates, Inc. (2016)

[13] Rajalingham, R., Issa, E.B., Bashivan, P., Kar, K., Schmidt, K., DiCarlo, J.J.: Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. J Neurosci **38**(33) (Aug 2018) 7255–7269

[14] Wichmann, F.A., Geirhos, R.: Are deep neural networks adequate behavioural models of human visual perception? (2023)

[15] Ratcliff, R.: A theory of memory retrieval. Psychological review **85**(2) (1978) 59

[16] Ratcliff, R., Smith, P.L., McKoon, G.: Modeling regularities in response time and accuracy data with the diffusion model. Curr Dir Psychol Sci **24**(6) (Dec 2015) 458–470

[17] Wiecki, T.V., Sofer, I., Frank, M.J.: Hddm: Hierarchical bayesian estimation of the drift-diffusion model in python. Frontiers in neuroinformatics **7** (2013) 55610

[18] Brown, S.D., Heathcote, A.: The simplest complete model of choice response time: Linear ballistic accumulation. Cognitive psychology **57**(3) (2008) 153–178

[19] Trueblood, J.S., Brown, S.D., Heathcote, A.: The multiattribute linear ballistic accumulator model of context effects in multialternative choice. Psychological review **121**(2) (2014) 179

[20] De Boeck, P., Jeon, M.: An overview of models for response times and processes in cognitive tests. Front Psychol **10** (2019) 102

[21] Wang, X.J.: Probabilistic decision making by slow reverberation in cortical circuits. Neuron **36**(5) (Dec 2002) 955–968

[22] Wong, K.F., Wang, X.J.: A recurrent network mechanism of time integration in perceptual decisions. J. Neurosci. **26**(4) (January 2006) 1314–1328

[23] Petrov, A.A., Van Horn, N.M., Ratcliff, R.: Dissociable perceptual-learning mechanisms revealed by diffusion-model analysis. Psychonomic bulletin & review **18** (2011) 490–497

[24] Green, C.S., Pouget, A., Bavelier, D.: Improved probabilistic inference as a general learning mechanism with action video games. Curr Biol **20**(17) (Sep 2010) 1573–1579

[25] Spoerer, C.J., McClure, P., Kriegeskorte, N.: Recurrent convolutional neural networks: A better model of biological object recognition. Front. Psychol. **8** (September 2017) 1551

[26] Kreiman, G., Serre, T.: Beyond the feedforward sweep: feedback computations in the visual cortex. Ann. N. Y. Acad. Sci. **1464**(1) (March 2020) 222–241

[27] Kar, K., Kubilius, J., Schmidt, K., Issa, E.B., DiCarlo, J.J.: Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. Nat. Neurosci. **22**(6) (June 2019) 974–983

[28] Van Kerkoerle, T., Self, M.W., Dagnino, B., Gariel-Mathis, M.A., Poort, J., Van Der Togt, C., Roelfsema, P.R.: Alpha and gamma oscillations characterize feedback and feedforward processing in monkey visual cortex. Proceedings of the National Academy of Sciences **111**(40) (2014) 14332–14341

[29] Spoerer, C.J., Kietzmann, T.C., Mehrer, J., Charest, I., Kriegeskorte, N.: Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. PLOS Computational Biology **16**(10) (10 2020) 1–27

[30] Goetschalckx, L., Govindarajan, L.N., Karkada Ashok, A., Ahuja, A., Sheinberg, D., Serre, T.: Computing a human-like reaction time metric from stable recurrent vision models. Advances in Neural Information Processing Systems **36** (2024)

 https://doi.org/10.52202/079017-4147

[31] Sensoy, M., Kaplan, L., Kandemir, M.: Evidential deep learning to quantify classification uncertainty. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., eds.: Advances in Neural Information Processing Systems. Volume 31., Curran Associates, Inc. (2018)

[32] Vickers, D., Packer, J.: Effects of alternating set for speed or accuracy on response time, accuracy and confidence in a unidimensional discrimination task. Acta psychologica **50**(2) (1982) 179–197

[33] Rahnev, D.: A robust confidence–accuracy dissociation via criterion attraction. Neuroscience of Consciousness **2021**(1) (2021) niab039

[34] Winkel, J., Keuken, M.C., van Maanen, L., Wagenmakers, E.J., Forstmann, B.U.: Early evidence affects later decisions: Why evidence accumulation is required to explain response time data. Psychonomic bulletin & review **21** (2014) 777–784

[35] Bogacz, R., Brown, E., Moehlis, J., Holmes, P., Cohen, J.D.: The physics of optimal decision making: a formal analysis of models of performance in two-alternative forced-choice tasks. Psychological review **113**(4) (2006) 700

[36] Drugowitsch, J., Moreno-Bote, R., Churchland, A.K., Shadlen, M.N., Pouget, A.: The cost of accumulating evidence in perceptual decision making. Journal of Neuroscience **32**(11) (2012) 3612–3628

[37] Cisek, P., Puskas, G.A., El-Murr, S.: Decisions in changing conditions: the urgency-gating model. Journal of Neuroscience **29**(37) (2009) 11560–11571

[38] Cochrane, A., Sims, C.R., Bejjanki, V.R., Green, C.S., Bavelier, D.: Multiple timescales of learning indicated by changes in evidence-accumulation processes during perceptual decision-making. npj Science of Learning **8**(1) (2023) 19

[39] Kar, K., Kubilius, J., Schmidt, K., Issa, E.B., DiCarlo, J.J.: Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. Nature neuroscience **22**(6) (2019) 974–983

[40] Palmer, J., Huk, A.C., Shadlen, M.N.: The effect of stimulus strength on the speed and accuracy of a perceptual decision. J Vis **5**(5) (May 2005) 376–404

[41] Shibata, K., Chang, L.H., Kim, D., Náñez, Sr, J.E., Kamitani, Y., Watanabe, T., Sasaki, Y.: Decoding reveals plasticity in v3a as a result of motion perceptual learning. PLOS ONE **7**(8) (08 2012) 1–7

[42] Law, C.T., Gold, J.I.: Neural correlates of perceptual learning in a sensory-motor, but not a sensory, cortical area. Nature neuroscience **11**(4) (2008) 505–513

[43] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)

[44] Mehrer, J., Spoerer, C.J., Jones, E.C., Kriegeskorte, N., Kietzmann, T.C.: An ecologically motivated image dataset for deep learning yields better models of human vision. Proceedings of the National Academy of Sciences **118**(8) (2021) e2011417118

[45] DiCarlo, J.J., Zoccolan, D., Rust, N.C.: How does the brain solve visual object recognition? Neuron **73**(3) (2012) 415–434

[46] Serre, T., Oliva, A., Poggio, T.: A feedforward architecture accounts for rapid categorization. Proceedings of the national academy of sciences **104**(15) (2007) 6424–6429

[47] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer (2014) 740–755

[48] Kubilius, J., Schrimpf, M., Kar, K., Rajalingham, R., Hong, H., Majaj, N., Issa, E., Bashivan, P., Prescott-Roy, J., Schmidt, K., et al.: Brain-like object recognition with high-performing shallow recurrent anns. Advances in neural information processing systems **32** (2019)

[49] Schrimpf, M., Kubilius, J., Lee, M.J., Ratan Murty, N.A., Ajemian, R., DiCarlo, J.J.: Integrative benchmarking to advance neurally mechanistic models of human intelligence. Neuron **108**(3) (November 2020) 413–423

[50] Hebart, M.N., Contier, O., Teichmann, L., Rockter, A.H., Zheng, C.Y., Kidder, A., Corriveau, A., Vaziri-Pashkam, M., Baker, C.I.: Things-data, a multimodal collection of large-scale datasets for investigating object representations in human brain and behavior. eLife **12** (feb 2023) e82580

[51] Fel, T., Rodriguez, I.F., Linsley, D., Serre, T.: Harmonizing the object recognition strategies of deep neural networks with humans. Advances in Neural Information Processing Systems (NeurIPS) (2022)

[52] Linsley, D., Rodriguez, I.F., Fel, T., Arcaro, M., Sharma, S., Livingstone, M., Serre, T.: Performance-optimized deep neural networks are evolving into worse models of inferotemporal visual cortex (2023)

[53] Liang, W., Tadesse, G.A., Ho, D., Fei-Fei, L., Zaharia, M., Zhang, C., Zou, J.: Advances, challenges and opportunities in creating data for trustworthy ai. Nature Machine Intelligence **4**(8) (2022) 669–677

[54] Li, B., Qi, P., Liu, B., Di, S., Liu, J., Pei, J., Yi, J., Zhou, B.: Trustworthy ai: From principles to practices. ACM Computing Surveys **55**(9) (2023) 1–46

 https://doi.org/10.52202/079017-4147

A Appendix / Supplemental material

A.1 Complete derivation of the differentiable framework

Proposition 1. *Let us define* $\tau^*_\theta(\Phi_t) = \min\{t \in \mathbb{R}_{[1,N]} : \Phi_t > \theta\}$ *as the time in which* Φ_t *reaches the threshold of activity* θ*. Provided that* Φ_t *is continuously differentiable:*

$$\frac{\partial \tau^*_\theta}{\partial w} \approx \frac{1}{\Phi_t - \Phi_{t-1}} \cdot (-\frac{\partial \Phi_t}{\partial w}) \tag{5}$$

Proof. By definition,

$$\tau^*_\theta(\Phi_t) = \min\{t \in \mathbb{R}_{[1,N]} : \Phi_t > \theta\}.$$

Let us consider a small change $\delta\Phi$. By the Taylor expansion, we can write:

$$\tau^*_\theta(\Phi_t + \delta\Phi) \approx \tau^*_\theta(\Phi_t) + \frac{\partial \tau^*_\theta}{\partial \Phi_t}\delta\Phi.$$

On the other hand, t is considered a continuous value. By our definition, we can induce a small change in the value of Φ_t by introducing a small change in time $\delta\Phi$, which we can write in the following way:

$$\Phi_{t+\delta t} \approx \Phi_t + \frac{\partial \Phi_t}{\partial t}\delta t$$

Without loss of generality, let us take $\delta t = \left(\frac{\partial \Phi_t}{\partial t}\right)^{-1}\delta\Phi$ and then

$$\Phi_{t+\delta t} \approx \Phi_t + \frac{\partial \Phi_t}{\partial t}\left(\frac{\partial \Phi_t}{\partial t}\right)^{-1}\delta\Phi = \Phi_t + \delta\Phi$$

Now,

$$\begin{aligned}\tau^*_\theta(\Phi_t + \delta\Phi) &= \min\{t \in \mathbb{R}_{[1,N]} : \Phi_t + \delta\Phi > \theta\} \\ &\approx \min\{t \in \mathbb{R}_{[1,N]} : \Phi_{t+\delta t} > \theta\} \\ &= \min\{t \in \mathbb{R}_{[1,N]} : \Phi_{t+\left(\frac{\partial \Phi_t}{\partial t}\right)^{-1}\delta\Phi} > \theta\}\end{aligned}$$

Note that, by definition, if the evidence is evolving by $\Phi_{t+\Delta}$. It just means the whole function is moving along the t-axis, and therefore the minimum time to pass the threshold would be given by $\tau^*_\theta - \Delta$, therefore

$$\begin{aligned}\min\{t \in \mathbb{R}_{[1,N]} : \Phi_{t+\left(\frac{\partial \Phi}{\partial t}\right)^{-1}\delta\Phi}\} &= \min\{t \in \mathbb{R}_{[1,N]} : \Phi_t\} - \left(\frac{\partial \Phi_t}{\partial t}\right)^{-1}\delta\Phi \\ &= \tau^*_\theta(\Phi) - \left(\frac{\partial \Phi}{\partial t}\right)^{-1}\delta\Phi\end{aligned}$$

Finally, we can join the two sides and obtain:

$$\tau^*_\theta(\Phi_t + \delta\Phi) \approx \tau^*_\theta(\Phi) + \frac{\partial \tau^*_\theta}{\partial \Phi}\delta\Phi = \tau^*_\theta(\Phi) - \left(\frac{\partial \Phi}{\partial t}\right)^{-1}\delta\Phi$$

From this, we can deduce:

$$\frac{\partial \tau^*_\theta}{\partial \Phi} = -\left(\frac{\partial \Phi}{\partial t}\right)^{-1} \tag{6}$$

$$\frac{\partial \tau^*_\theta}{\partial \Phi_t} \approx -(\frac{\Phi_t - \Phi_{t-1}}{t-(t-1)})^{-1} \tag{7}$$

$$\frac{\partial \tau^*_\theta}{\partial w} \approx -\frac{1}{\Phi_t - \Phi_{t-1}}\frac{\partial \Phi_t}{\partial w} \tag{8}$$

So the first part of Eq 2 in the main text is proved. □

Similarly, we can derive $\frac{\partial \tau^*_\theta}{\partial \theta} \approx \frac{1}{\Phi_t - \Phi_{t-1}}$.

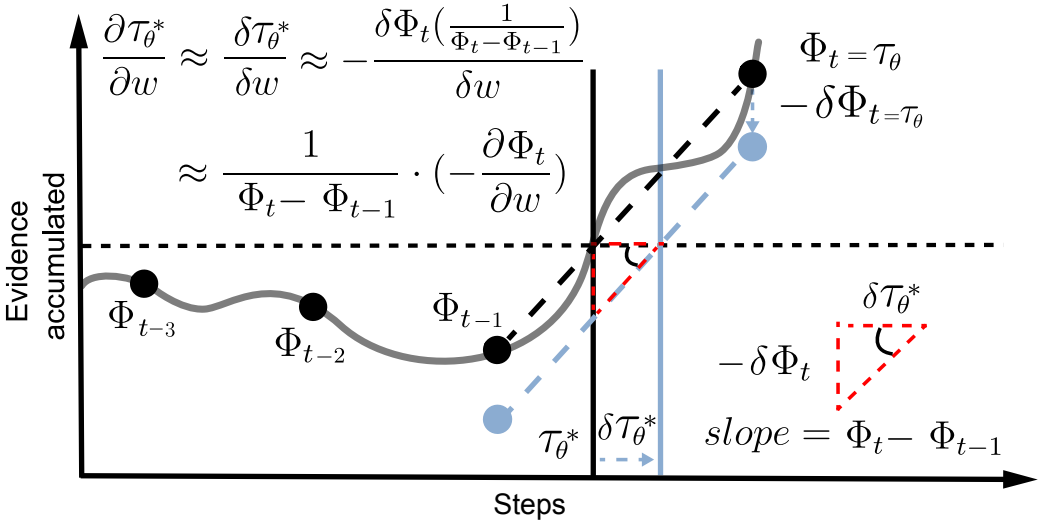
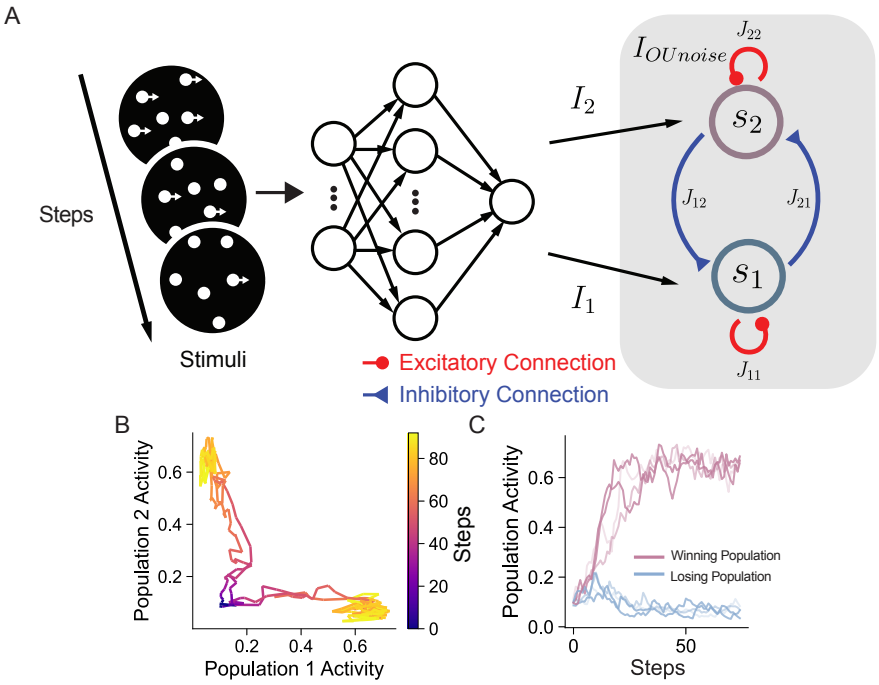


Figure S1: **Illustration of Mathematics Proof** The discrete RNN step τ_θ is not differentiable. Therefore, we introduce a piecewise linear approximation of the accumulated evidence over time Φ_t. Consider the effect of changes in Φ_t on $\tau_\theta(\Phi)$. A small perturbation in Φ_t will produce a proportional change in the time it takes for the accumulated evidence to cross the threshold θ, thereby inducing a shift in time. In simple terms, fine-tuning w to decrease Φ_t will delay the time at which the network crosses the threshold θ thus increasing $\tau_\theta(\Phi)$, while fine-tuning w to increase Φ_t will cause the threshold to be crossed earlier thus decreasing $\tau_\theta(\Phi)$.

https://doi.org/10.52202/079017-4147

A.2 Illustration of the RTified WW circuit



D

Algorithm 1 Forward Function of WW circuit for Random Dot Motion

Input: Image, threshold
Output: decision_times
Initialize $s_1, s_2, decision\_times$
while $s_1 < threshold$ and $s_2 < threshold$ **do**
  $I_1, I_2 \leftarrow f_w(\zeta(Image))$
  Combining excitatory and inhibitory currents
  $H_1 \leftarrow f(J_{11} \cdot s_1 - J_{12} \cdot s_2 + I_0 + I_1 + I_{\text{OU noise1}})$
  $H_2 \leftarrow f(J_{22} \cdot s_2 - J_{21} \cdot s_1 + I_0 + I_2 + I_{\text{OU noise2}})$
  Calculate rate of change
  $\frac{ds_1}{dt} = -\frac{s_1}{\tau_s} + H_1 \cdot (1 - s_1)$
  $\frac{ds_2}{dt} = -\frac{s_2}{\tau_s} + H_2 \cdot (1 - s_2)$
  Update $s_1, s_2, decision\_times$
end while
return decision_times

Figure S2: **Illustration of the RTified WW circuit.** To RTify feedforward neural networks, we used an RNN based on the WW circuit. Here, we consider a binary classification on random moving dots for illustration **(A)**. When receiving a visual input, the two populations sensitive to left/right directions compete with each other **(B)** and accumulate evidence until one of them reaches a threshold **(C)**. The number of time steps needed for the RNN to reach the threshold is defined as the "model RT" and is used to predict human RT. We provide pseudo-code for a more detailed description of the circuit **(D)**. Here, $f(x) = \gamma \cdot max(ax - b, 0)/(1 - exp(-d(ax - b)))$ is a fixed nonlinear function. And the model and equations marked blue are how we extend WW model.

A.3 Extended results for RDM task

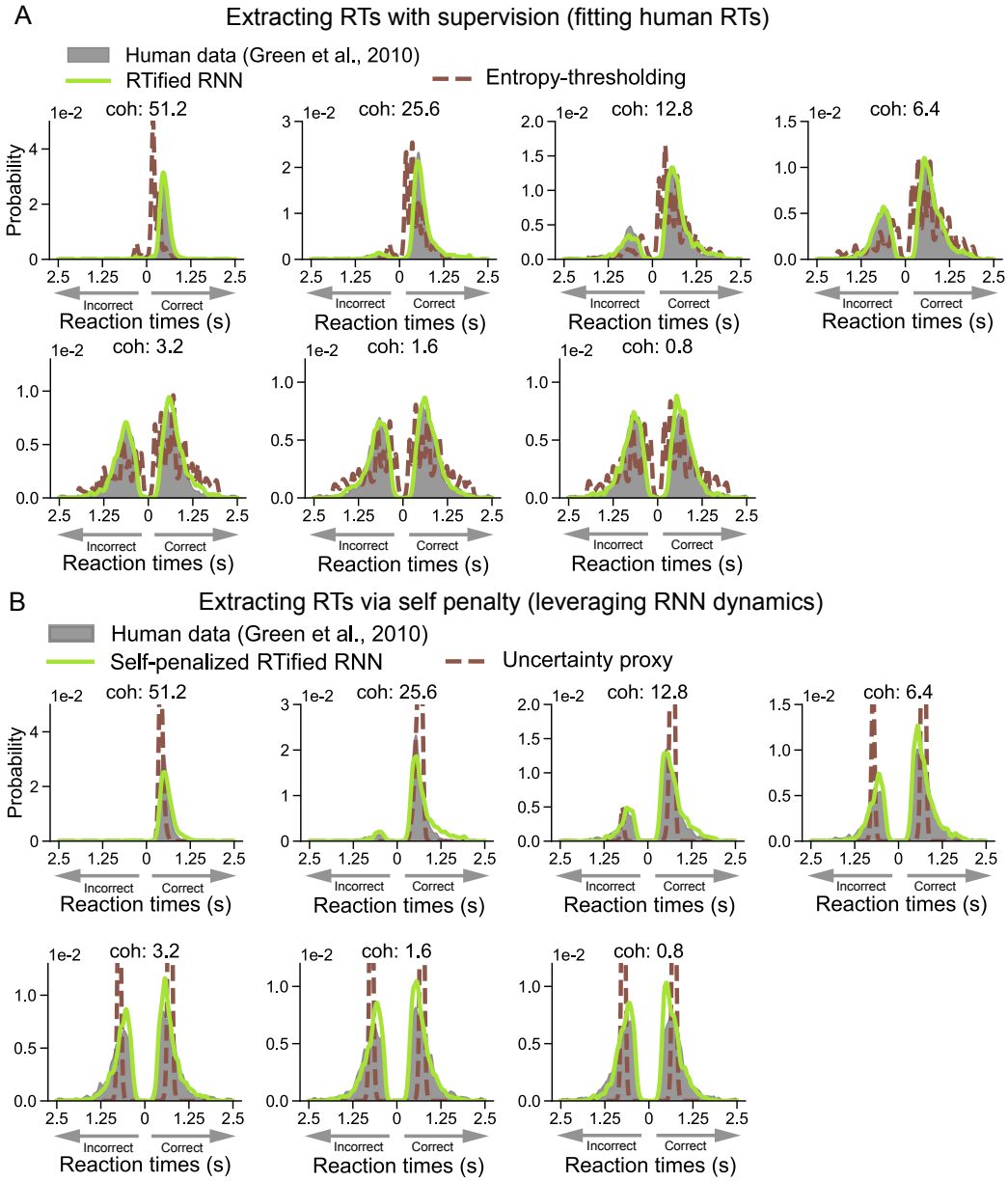


Figure S3: **RTified model evaluation on a RDM task [24] across all coherences.** Human data are shown as a gray shaded area, and model fits are shown for **(A)** the "supervised" setting where human behavioral responses are used to train the models and **(B)** the "self-penalized" setting where no human data is used. Our approach (green) outperforms the two alternative approaches (brown), i.e., entropy-thresholding [29] for the "supervised" and uncertainty proxy [30] for the "self-penalized" settings.

https://doi.org/10.52202/079017-4147

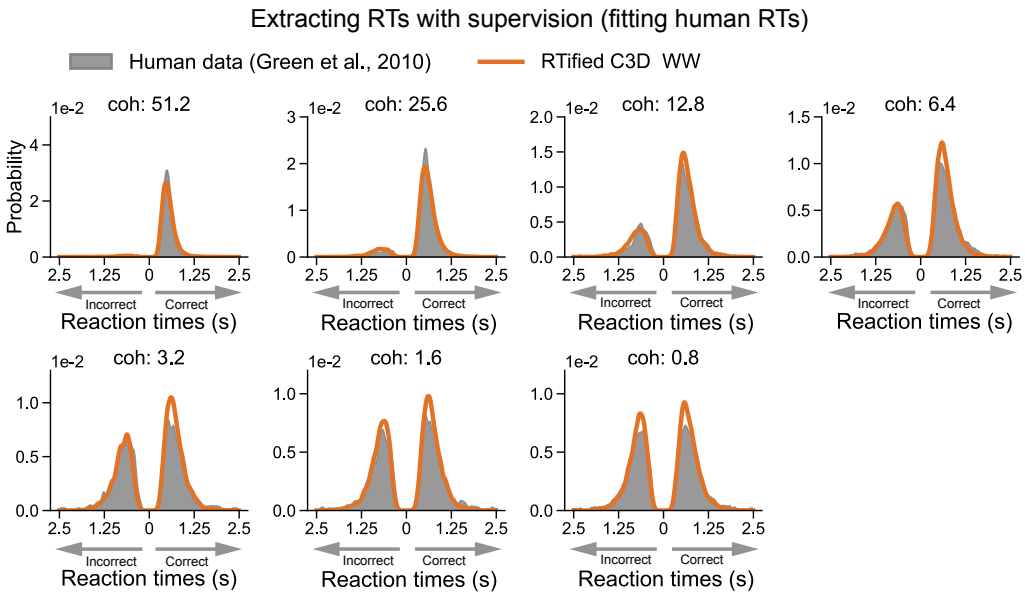


Figure S4: **RTified WW model on a RDM task [24] across all coherences.** We combined our RTified WW module with a 3D CNN to fit human RTs collected in an RDM task [24]. Human data are shown as a gray shaded area and model results are shown as orange solid lines.

A.4 Additional Experiment for RDM task

Recurrent neural networks generate a sequence of outputs across time steps, raising the question of how to convert these multiple outputs into a single final prediction. In the main text, we combine all the network's outputs up until the decision time point. Here, as an additional experiment, we only use the network's output at the decision time point. In both "supervised" and "self-penalized" settings, our approach surpasses two alternative approaches [29, 30] (see Fig. S5).

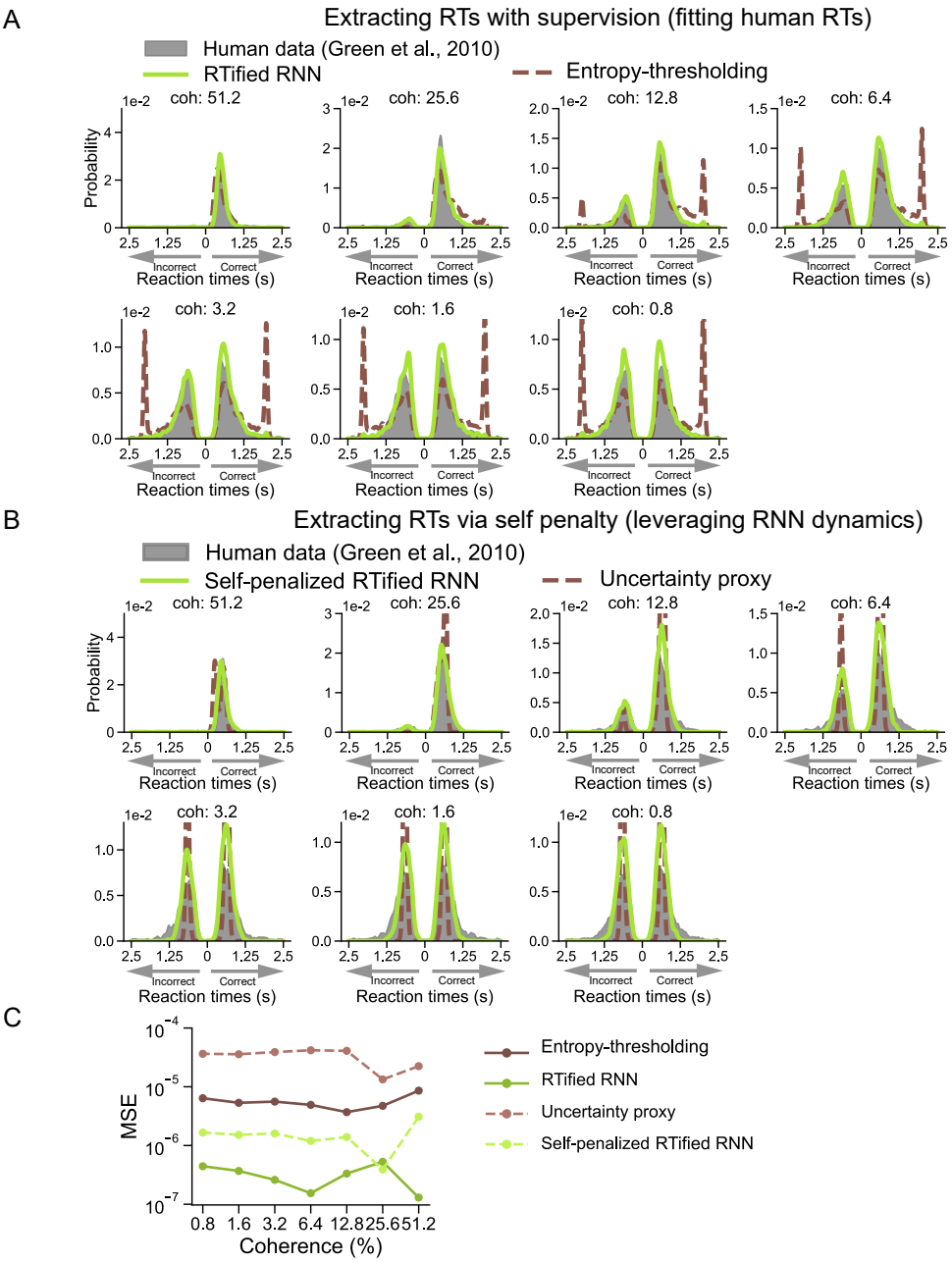


Figure S5: **RTified model evaluation on a RDM task [24] across all coherences.** Human data are shown as a gray shaded area, and model fits are shown for **(A)** the "supervised" setting where human behavioral responses are used to train the models and **(B)** the "self-penalized" setting where no human data is used. Our approach (green) outperforms the two alternative approaches (brown), i.e., entropy-thresholding [29] for the "supervised" and uncertainty proxy [30] for the "self-penalized" settings. MSE comparison is shown in **(C)**.

https://doi.org/10.52202/079017-4147

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main contributions of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations of the proposed method are discussed, the scope and assumptions are reflected.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All theoretical results, along with their assumptions and proofs, are provided in main text and supplementary files.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Detailed descriptions of the experimental setup, including datasets, hyperparameters, and evaluation metrics, are provided. Links to the code and data can be found in our GitHub.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/079017-4147

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer:[Yes]

Justification: The code and datasets used in the experiments are publicly available in our GitHub, with detailed instructions provided.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access raw data, preprocessed data, intermediate data, generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Comprehensive details on the training and test settings, including data splits, hyperparameters, and optimizer types, are provided.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The experimental results are presented with appropriate information about the statistical significance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or Fig.s symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding Fig.s or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper specifies the types of compute resources used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed the broader impacts of the proposed method, including potential applications.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

https://doi.org/10.52202/079017-4147

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not involve the release of high-risk data or models.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All external assets used in the paper are properly credited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: The paper only involves published human data where all relevant experimental information can be found in the original papers.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: The paper only involves published human data where all IRB information can be found in the original papers.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

https://doi.org/10.52202/079017-4147

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.