Learning Dynamics of Attention: Human Prior for Interpretable Machine Reasoning

Wonjae Kim
Kakao Corporation
Pangyo, Republic of Korea
dandelin.kim@kakaocorp.com

Yoonho Lee
Kakao Corporation
Pangyo, Republic of Korea
eddy.l@kakaocorp.com

Abstract

Without relevant human priors, neural networks may learn uninterpretable features. We propose **D**ynamics of **A**ttention for **F**ocus **T**ransition (DAFT) as a human prior for machine reasoning. DAFT is a novel method that regularizes attention-based reasoning by modelling it as a continuous dynamical system using neural ordinary differential equations. As a proof of concept, we augment a state-of-the-art visual reasoning model with DAFT. Our experiments reveal that applying DAFT yields similar performance to the original model while using fewer reasoning steps, showing that it implicitly learns to skip unnecessary steps. We also propose a new metric, **T**otal **L**ength of **T**ransition (TLT), which represents the effective reasoning step size by quantifying how much a given model's focus drifts while reasoning about a question. We show that adding DAFT results in lower TLT, demonstrating that our method indeed obeys the human prior towards shorter reasoning paths in addition to producing more interpretable attention maps.

1 Introduction

Humans reason by continually updating their mental representations, meaning they progressively align internal knowledge with external stimuli [Gentner, 2010]. Such alignment forms a tree-like hierarchy with first-order relations as leaves and higher-order relation as intermediate nodes. An example of a high-order relation is *"The number of big cyan things is greater than that of small green cubes"*, which combines numerical and visuospatial reasoning. Vendetti and Bunge [2014] found empirical evidence that within the brain, the lateral frontoparietal network (LFPN) was highly active while performing relational reasoning. The activation distributed over LFPN changes continuously to focus on appropriate mental representations to perform reasoning.

To model this human reasoning prior of continuous focus transition into a machine learning model that aims to perform relational reasoning, we propose **Dynamics of Attention for Focus Transition** (DAFT). DAFT directly models the infinitesimal change of focus (i.e., attention) at each point in time. By solving the initial value problem (IVP) defined by a learned DAFT model, we can acquire a continuous function over time that returns attention for a given time. This IVP solution can replace the discrete attention mechanisms (which most machine reasoning models currently use) with a continuous attention map. While DAFT is applicable for all machine reasoning models that use attention and memory, we applied it to the MAC network [Hudson and Manning, 2018], the state-of-the-art visual reasoning model, to show how this human prior acts in a holistic model.

In addition to DAFT, we define **Total Length of Transition** (TLT), a metric for quantifying the transition of focus. TLT measures the length of focus transition by summing up shifts between adjacent attention maps. TLT can be interpreted as how well the model follows the law of parsimony, also known as *Occam's razor*, since the model with lower TLT plans a more simpler transition of focus (i.e., simpler alignment of representations). Feldman [2016] proposed the *simplicity principle*, a

Preprint. Under review.

contemporary interpretation of Occam's razor which states that our minds seek the simplest possible interpretation of observations. Following the principle, we argue that the interpretation a machine reasoning model yields should be as simple as possible, and therefore that a model with lower TLT is more interpretable. By using the TLT as a quantitative measure of interpretability, we can move away from the traditional qualitative-only assessment which was done by showing attention maps.

Throughout the paper, we establish a link between human reasoning prior and the simplicity principle with extensive experiments. Specifically, we reveal their connection by showing that application of DAFT dramatically lowers the model's TLT and enhances the model's interpretability. Meanwhile, the TLT of original model keep increase and lose its interpretability as the step size is increased.

2 Background

Our work encompasses multiple disciplines of machine learning including machine reasoning, interpretable machine learning, and neural ordinary differential equations. In this section, we summarize each and explain how they are related to our work.

2.1 Machine Reasoning

Machine reasoning tasks were proposed to test whether algorithms can demonstrate high-level reasoning capabilities once believed to be only possible for humans [Bottou, 2014]. Given knowledge base $\mathbf{K}$ and task description $\mathbf{Q}$, the machine should perform progressive alignment of $\mathbf{K}$ conditioned by $\mathbf{Q}$ to produce the answer. There are a variety of such tasks such as casual to social reasoning, but we focus on visual reasoning in this work. Visual Question Answering (VQA) Agrawal et al. [2015], which observes what questions a model can answer about an image, is the most well-known test for visual reasoning capability.

Approaches for solving VQA vary widely on which supervisory signals are given. The usual supervisory signals in VQA comprise images, questions, answers, programs, and object masks. Following Mao et al. [2018], we denote the former three signals as *natural supervision* and the latter two signals as *additional supervision*. The natural supervision signals are only signals that all VQA datasets have in common [Agrawal et al., 2015, Krishna et al., 2017, Goyal et al., 2017, Hudson and Manning, 2019], mainly because acquiring additional supervision is costly. Without additional supervisions, models need to fuse natural supervisions into mental representations and perform progressive alignment on these representations to answer the question.[1] Such mental representations are usually modeled via memory and the progressive alignment of memory with supervisory signals is modeled with attention. Xiong et al. [2016], Hudson and Manning [2018] have proposed such **memory and attention** models.

Among models that adopt the concept of the memory and attention, we applied DAFT to the Memory, Attention, and Composition (MAC) network [Hudson and Manning, 2018], a state-of-the-art machine reasoning model which uses only natural supervision. The MAC network has two memory vectors: one for the mental representation of $\mathbf{K}$ and the other for $\mathbf{Q}$, and Hudson and Manning [2018] called the latter the control vector. We review the MAC network in more detail in section 3.

2.2 Human Prior and Interpretability

With the growing demands on interpretable machine learning, attention-based machine reasoning models demonstrated their interpretability by showing their attention map visualizations. However, Ilyas et al. [2019] claimed that without a human prior, neural networks eventually learn *useful but non-robust features* which are highly predictive for the model but not useful for humans. Concurrently, Poursabzi-Sangdeh et al. [2018] and Lage et al. [2018] empirically show how human prior affects the interpretability of the model. This claim has been shown to hold in the field of interpretable machine reasoning: for example, [Hudson and Manning, 2018] observed that increasing reasoning step length leaves the model's performance intact (*useful*) but their attention maps became uninterpretable (*non-robust*). To solve this problem, we propose DAFT in section 4 to embed the human reasoning prior of continuous focus transition in attention-based machine reasoning models.

[1] For other approaches using additional supervision, refer to A in the appendix for more details.

Another problem is that there exists no quantitative measure of interpretability. This is because the interpretability is fundamentally qualitative, and by principle, it can only be measured via a user study. However, user studies cannot scale to large datasets. In section 5.4, we propose TLT as a quantitative, and therefore scalable, measure of interpretability. Since TLT is built upon the simplicity principle of human perception, we argue that TLT can be used as a proxy to interpretability.

2.3 Neural Ordinary Differential Equations

Recent work on residual networks [Lu et al., 2017, Haber and Ruthotto, 2017, Ruthotto and Haber, 2018] interpret residual connections as an Euler discretization of a continuous transformation through time. Motivated by this interpretation, Chen et al. [2018] generalized residual networks by using more sophisticated black-box ODE solvers such as `dopri5` [Dormand and Prince, 1980]. This enables the models to learn the parametric dynamics in a finer discretization of time ($\Delta t < 1$) than the standard residual connections ($\Delta t = 1$). This generalization of residual networks yields a new family of neural networks called neural ordinary differential equations (neural ODEs) [Chen et al., 2018]. We employed this innovative idea to model continuous transitions of focus.

Dupont et al. [2019] stated that the homeomorphism of neural ODEs greatly restricts the representation power of the dynamics and show a number of functions which cannot be represented by the family of neural ODEs. They showed that by augmenting the feature space by adding empty dimensions, the dynamics of neural ODEs can be simplified. To show its efficacy, they measured the number of function evaluation (NFE) along with the training, since complex dynamics requires exponentially many function evaluations while solving IVP. They showed augmented neural ODEs yields a gradually growing NFE during training while their non-augmented counterpart has an NFE that grows exponentially. In section 4, we will show the link between DAFT and augmented neural ODEs.

3 The MAC Network

Algorithm 1 Memory Update Procedure of MAC

Input : current time t_0, next time t_1, current memory $\mathbf{m}_{t_0}$, contextualized question $\mathbf{cw} \in \mathbb{R}^{L \times d}$, atomic question $\mathbf{q} = [\overleftarrow{\mathbf{cw}_1}, \overrightarrow{\mathbf{cw}_L}]$, knowledge base $\mathbf{K} \in \mathbb{R}^{S \times d}$

Output : next memory $\mathbf{m}_{t_1}$

1: $\mathbf{a}_{t_1} = \mathbf{W}^{1\times d}(\mathbf{W}^{d\times d}_{t_1}\mathbf{q} \odot \mathbf{cw})$ ▷ get *attention logit* on $\mathbf{cw}$

2: $\mathbf{c}_{t_1} = \sum_{i=0}^{L} \text{softmax}(\mathbf{a}_{t_1})(i) \odot \mathbf{cw}(i)$ ▷ get *control* vector

3: $\mathbf{rq}_{t_1} = \mathbf{W}^{1\times d}(\mathbf{W}^{d\times 2d}[\mathbf{W}^{d\times d}\mathbf{K} \odot \mathbf{W}^{d\times d}\mathbf{m}_{t_1}, \mathbf{K}] \odot \mathbf{c}_{t_1})$ ▷ get *attention logit* on $\mathbf{K}$

4: $\mathbf{r}_{t_1} = \sum_{i=0}^{S} \text{softmax}(\mathbf{rq}_{t_1})(i) \odot \mathbf{K}(i)$ ▷ get *information* vector

5: $\mathbf{m}_{t_1} = \mathbf{W}^{d\times 2d}[\mathbf{r}_{t_1}, \mathbf{m}_{t_0}]$ ▷ get *memory* vector

We briefly review the MAC network [Hudson and Manning, 2018]. It consists of three subunits (control, read, and write) which rely on each other to perform visual reasoning. Given a task description (in VQA, a question) $\mathbf{Q}$ and image $\mathbf{I}$ to reason on, the model first encodes each modality into its neural representation $\mathbf{cw}$ and $\mathbf{K}$. These features are extracted from a bi-directional LSTM and the `conv4` layer of a ResNet-101 model, respectively. Algorithm 1 describes how the MAC network updates its memory vector given its inputs.

Given initial memory vector $\mathbf{m}_0$, T-step of iterative memory updates produce the final memory vector $\mathbf{m}_T$. MAC infers answer logits by processing the concatenation of $\mathbf{q}$ and $\mathbf{m}_T$ through a 2-layer classifier : $\mathbf{W}^{1\times d}(\mathbf{W}^{d\times 2d}[\mathbf{q}, \mathbf{m}_T])$. Note that bias and nonlinearities are omitted for brevity. The original work optionally considers additional structures inside the write unit. Unlike the description in the original paper, previous control $\mathbf{c}_{t-1}$ is not used when computing the current control $\mathbf{c}_t$ in the author's impelementation[2].

[2]https://github.com/stanfordnlp/mac-network/blob/c7121362df/configs/args.txt

4 Dynamics of Attention for Focus Transition

We now introduce Dynamics of Attention for Focus Transition (DAFT) and its application to MAC. We call this augmented MAC model DAFT MAC.

Algorithm 2 Memory Update Procedure of DAFT MAC

Input : current time t_0, next time t_1, current memory $\mathbf{m}_{t_0}$, contextualized question $\mathbf{cw} \in \mathbb{R}^{L \times d}$, atomic question $\mathbf{q} = [\overleftarrow{\mathbf{cw}_1}, \overrightarrow{\mathbf{cw}_L}]$, knowledge base $\mathbf{K} \in \mathbb{R}^{S \times d}$, current attention logit $\mathbf{a}_{t_0}$

Output : next memory $\mathbf{m}_{t_1}$, next attention logit $\mathbf{a}_{t_1}$

1: **def** f($\mathbf{a}_t$, t): ▷ **Define DAFT**
2: **return** $\mathbf{W}^{1\times(d+1)}[\mathbf{W}^{d\times(d+1)}[t, \mathbf{q}] \odot \mathbf{cw}, \mathbf{a}_t]$ ▷ compute $\frac{d\mathbf{a}_t}{dt}$
3: $\mathbf{a}_{t_1} = \mathbf{a}_{t_0} + \int_{t_0}^{t_1} f(\mathbf{a}_t, t)dt = \text{ODESolve}(\mathbf{a}_t, f, t_0, t_1)$ ▷ Solve IVP using DAFT
4: $\mathbf{c}_{t_1} = \sum_{i=0}^{L} \text{softmax}(\mathbf{a}_{t_1})(i) \odot \mathbf{cw}(i)$
5: $\mathbf{rq}_{t_1} = \mathbf{W}^{1\times d}(\mathbf{W}^{d\times 2d}[\mathbf{W}^{d\times d}\mathbf{K} \odot \mathbf{W}^{d\times d}\mathbf{m}_{t_0}, \mathbf{K}] \odot \mathbf{c}_{t_1})$
6: $\mathbf{r}_{t_1} = \sum_{i=0}^{S} \text{softmax}(\mathbf{rq}_{t_1})(i) \odot \mathbf{K}(i)$
7: $\mathbf{m}_{t_1} = \mathbf{W}^{d\times 2d}[\mathbf{r}_{t_1}, \mathbf{m}_{t_0}]$

Algorithm 2 shows the memory update procedure of DAFT MAC and the definition of DAFT in full detail. Lines 4 to 7 are identical to the lines 2 to 5 in algorithm 1. Since only the highlighted lines in algorithms 1 and 2 were modified to apply DAFT, we point out that DAFT can be just as easily applied to memory-augmented models other than MAC.

Unlike MAC, the memory update procedure of DAFT MAC requires the incorporation of the previous attention logit, meaning we need to define the initial attention logit. We use a zero vector as the initial attention logit $\mathbf{a}_0$ to produce uniformly distributed attention weight, assuming the model's focus distributed evenly at the start of reasoning.

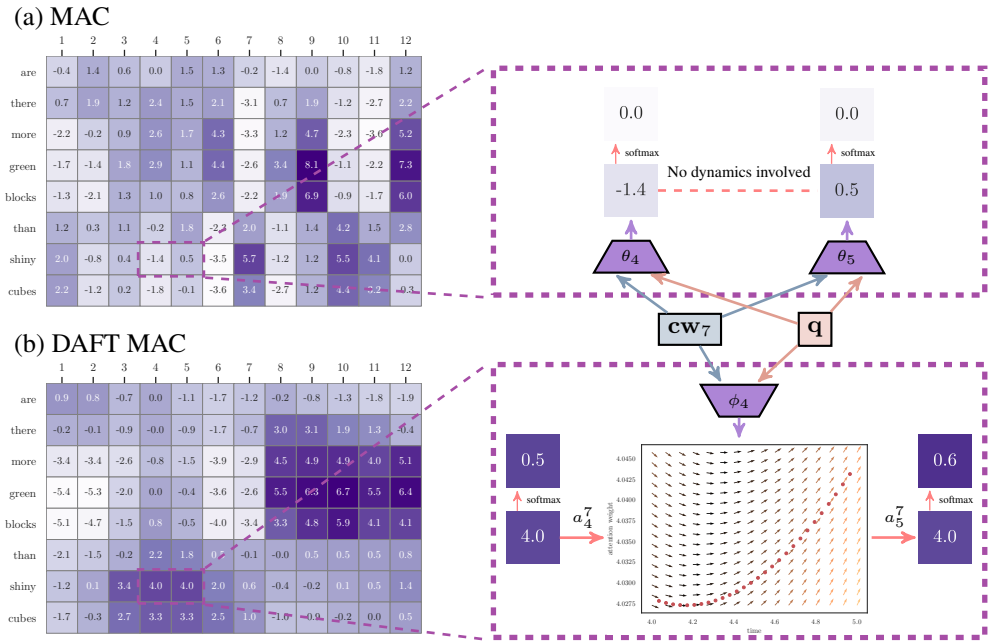

Figure 1: a graphical description of how attention logits change in MAC and DAFT MAC. the given question is *"are there more green blocks than shiny cubes?"*. Attention logits maps of 12-step (a) MAC and (b) DAFT MAC are shown. The right side shows a magnified view of a single step of attention shift on the word *shiny*.

Figure 1 shows the difference between MAC and DAFT MAC graphically. While MAC has no explicit connection between adjacent logits, DAFT MAC computes the next attention logit by solving

the IVP starting from the current attention logit. Note that the actual attention weight is the softmax-ed value of attention logits. Since softmax computes the size of a logit relative to other logits, small changes can result in a large difference in the attention weight (See figure 3 for the attention weight visualization).

Connection to Augmented Neural ODEs As shown in figure 1, every token and its question (in figure, $\mathbf{cw}_7$ and $\mathbf{q}$) acts as a condition on the dynamics. Empirically, we found that the conditionally generated ODE dynamics do not suffer from NFE explosion while solving IVP until the end of training (see figure 9 in the appendix for more details). It is remarkable since the VQA is incomparably more complex than the toy problems treated in previous works. Thus, we argue that these conditional ODE dynamics is another form of augmenting the neural ODEs as it differs from the previous unconditioned neural ODEs [Chen et al., 2018, Dupont et al., 2019].

5 Experiments

We used the CLEVR[3] dataset [Johnson et al., 2017a] for all experiments. To evaluate the efficacy of DAFT, we conducted experiments on two different criteria: performance and interpretability. We propose a novel metric that quantifies interpretability by measuring the total length of transition throughout reasoning. To control all variables which can affect the experiments of DAFT, we used the same hyperparamters as the original MAC network. The only modified part of MAC is the acquisition of attention logit for computing the control vector (highlighted lines in algorithms 1 and 2). Implementation details can be found in appendix B.

5.1 Dataset

Table 1: CLEVR accuracies for baselines with various additional annotation types (**P** for program and **M** for object mask annotation) and our model. D denotes the number of nodes (modules) in the inferred program semantics tree. $\triangle$ means that additional annotation is implicitly provided through the pretrained object detector such as Mask R-CNN.

Model	Anno. **P**	Anno. **M**	# Step	Avg.	Count	Exist	Cmp. Num.	Query Attr.	Cmp. Attr.
Human [Johnson et al., 2017a]	–	–	–	92.6	86.7	96.6	86.5	95.0	96.0
NMN [Andreas et al., 2016]	○	×	D	72.1	52.5	79.3	72.7	79.0	78.0
N2NMN [Hu et al., 2017]	○	×	D	88.8	68.5	85.7	84.9	90.0	88.8
IEP [Johnson et al., 2017b]	○	×	D	96.9	92.7	97.1	98.7	98.1	98.9
DDRprog [Suarez et al., 2018]	○	×	D	98.3	96.5	98.8	98.4	99.1	99.0
TbD [Mascharka et al., 2018]	○	×	D	99.1	97.6	99.2	99.4	99.5	99.6
NS-VQA [Yi et al., 2018]	○	○	D	99.8	99.7	99.9	99.9	99.8	99.8
NS-CL [Mao et al., 2018]	×	△	D	98.9	98.2	99.0	98.8	99.3	99.1
RN [Santoro et al., 2017]	×	×	1	95.5	90.1	97.8	93.6	97.1	97.9
FiLM [Perez et al., 2018]	×	×	4	97.6	94.5	99.2	93.8	99.2	99.0
MAC [Hudson and Manning, 2018]	×	×	12	98.9	97.2	99.5	99.4	99.3	99.5
DAFT MAC (Ours)	×	×	**4**	98.9	97.2	99.5	98.3	99.6	99.3

The CLEVR dataset was proposed to evaluate the visual reasoning capabilities of a model. CLEVR includes four supervisory signals: images, questions, answers, and programs. Images in CLEVR are synthetic scenes containing objects with various attributes (size, material, color, shape), and each image has multiple questions with corresponding answers to test relational and non-relational reasoning abilities. Programs in CLEVR describe a semantics tree, a collection of functions which are used to generate questions. For example, the program for the question "What color is the cube to the right of the yellow sphere?" is as follows: (1) `Filter` out non-yellow and non-sphere objects from the scene, (2) `Relate` the objects in the right of (1), (3) `Filter` out non-cube objects from (2), and (4) `Query` the color of the remaining object. The root of the semantics tree denotes the type of question, which will be used to analyze the result by question type.

[3]https://cs.stanford.edu/people/jcjohns/clevr/

We provide a survey of previous models for CLEVR in table 1, showing the accuracy by question type in addition to what additional supervision is given to the model. In total, CLEVR has 700K questions for training and 150K questions for validation and test split. All accuracies and TLT measured in the following sections were evaluated on the 150K validation set.

5.2 Performance

We re-implemented MAC along with DAFT MAC. We trained each pair of (method, step number) five times using different parameter initialization for thorough verification. As shown in figure 2, the accuracy of DAFT MAC outperforms that of the original MAC for fewer reasoning steps ($2 \sim 6$), and the two methods are roughly tied for larger reasoning steps. Hudson and Manning [2018] reported that MAC achieves its best accuracy (98.9%) at step size 12. DAFT MAC reaches this performance with step size 4. In our experiments, MAC and DAFT MAC both reach 99.0% accuracy at step size 8. Increasing step size beyond 8 results in practically the same performance while using more computation; in our experiments, 12-step requires $\sim$28% more than 8-step.

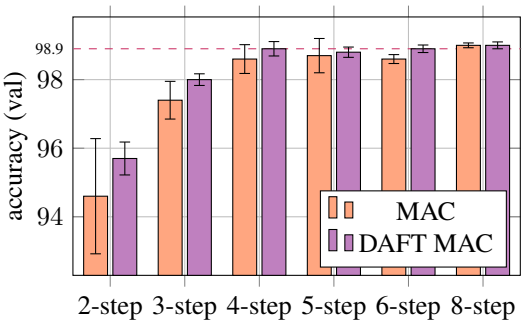

Figure 2: Comparison of overall CLEVR mean accuracy and its 95% confidence interval ($N = 5$) between MAC and DAFT MAC with varying reasoning steps.

The fact that the accuracy of DAFT MAC does not increase when increasing the reasoning step beyond four suggests that four reasoning steps are sufficient for CLEVR dataset. We provide more justification for this claim in section 5.4 by quantifying the effective number of reasoning steps in each model.

5.3 Interpretability

Many attention-based machine reasoning models have put emphasis on the interpretability of the attention map [Lu et al., 2016, Kim et al., 2018, Hudson and Manning, 2018]. Indeed, the attention map is a great source of interpretation since it points to specific temporal and spatial points helping our mind to interpret the observation, following the simplicity principle.

In figure 3, we compared the qualitative visualization of attention maps for MAC and DAFT MAC. One can see DAFT's given prior, the continuity of focus transition, brings several benefits in terms of its interpretation.

Chunking Compared to MAC, DAFT MAC produces more clustered and chunky attention maps. The question *"Are there more green blocks than shiny cubes?"* contains two noun phrases (NP), *more green blocks* and *shiny cubes*, when parsed to `(S Are there (NP (ADJP (ADVP more) green) blocks) (PP than (NP shiny cubes)))`. In this simple case, an ideal solver would only see each NP once to solve the problem. In figure 3, MAC distributes its attention to multiple temporally distant position to retrieve information while DAFT MAC distributes its attention to the chunks which are the same number as the question's NPs.

Consistency The attention maps produced by DAFT MAC presents a consistent progression of focus. We observed that DAFT MACs initialized with different seeds shares the order of transition. While the learned attention map of MAC varies greatly across different initializations, DAFT MAC consistently attends to *shiny cube* first and then afterwards to *more green blocks* (see figure 6 and 7 in the appendix for the clear distinction).

Interpolation Since the solution of IVP can yield an attention map for given timestamp, we can easily interpolate the attention maps in-between two adjacent steps. See figure 8 in the appendix for the visualization of these interpolated maps. Note that although we visualized the interpolation with the sampling rate of 20 due to limited space, this rate can go infinitely high since DAFT is

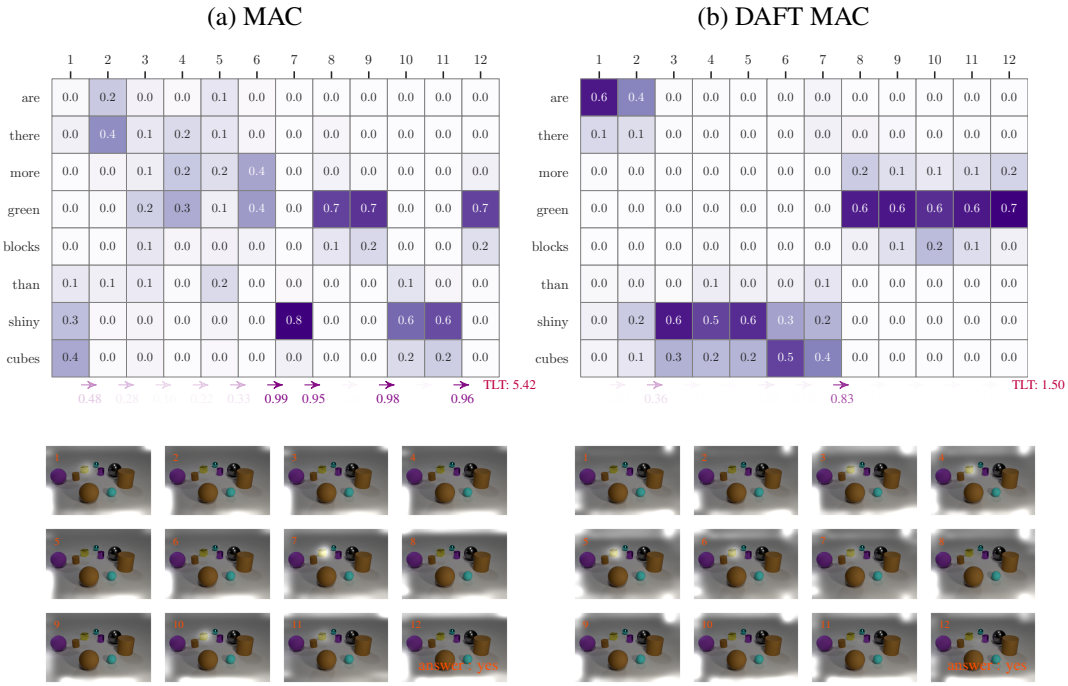

Figure 3: Qualitative exhibition of the question *"Are there more green blocks than shiny cubes?"* and its accompanying image, the same data used to show attention logit map in figure 1. (a) and (b) shows the actual softmax-ed textual and visual attention map which used to acquire the control vector and the information vector in MAC and DAFT MAC, respectively.

continuous in time. Also note that this interpolation differs from simple linear interpolation since DAFT is non-linear dynamics.

5.4 Measuring Length of Transition

Recall that the attention map is a categorical distribution over input tokens. We use the Jensen-Shannon divergence [Lin, 1991] to measure the amount of shift between attention maps throughout reasoning. We chose the Jensen-Shannon divergence because it is bounded (JSD$(P||Q) \in [0, 1]$).

Definition 1 *Length of Transition (LT)*
Let $\mathbf{p}_t \in \mathbb{R}^S$ *be the attention probability for time* $t = 1, \ldots, T$*. The Length of Transition (LT) at time* t *is defined as:*

$$LT(t) = JSD(\mathbf{p}_t||\mathbf{p}_{t+1}) = \frac{1}{2}\sum_{s=1}^{S} p_t^s \cdot \log_2 \frac{2 \cdot p_t^s}{p_t^s + p_{t+1}^s} + p_{t+1}^s \cdot \log_2 \frac{2 \cdot p_{t+1}^s}{p_t^s + p_{t+1}^s} \tag{1}$$

where p_t^s *is the* s*-th element of* $\mathbf{p}_t$*.*

We further define total length of transition (TLT) as TLT $= \sum_{i=1}^{T} LT(i)$. In default, TLT is bounded by $T-1$, and if TLT considers $LT(0)$, it is bounded by T. One can concatenate uniformly distributed attention to $\mathbf{a}$ as a starting attention $\mathbf{a}_0$ to get $LT(0)$. We do not use $LT(0)$ when calculating TLT throughout this paper, making it bounded by $T-1$.

Figure 3 shows LTs and TLT for MAC and DAFT MAC. LT and TLT can be seen as a measure of interpretability since LT peaks when the model's attention shifts. Furthermore, a model with low TLT is more likely to produce consistent attention maps across different initializations since TLT imposes an upper bound on the amount the model's attention can change.

Figure 4 shows the TLT of MAC and DAFT MAC. When the number of reasoning steps increases, the TLT of DAFT MAC is relatively unchanged while that of MAC increases with the step number. This

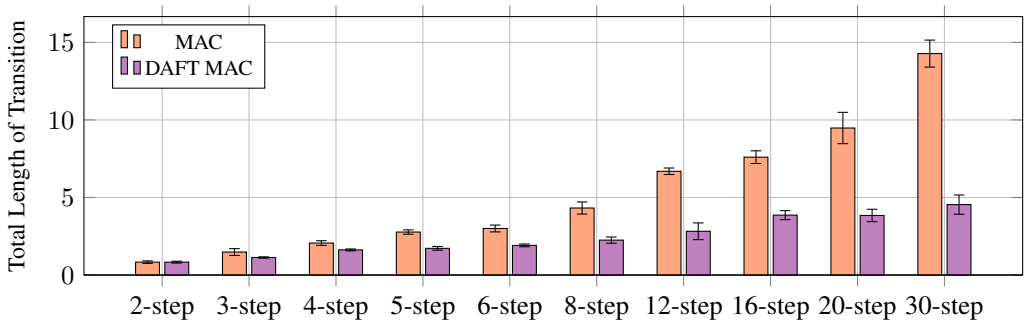

Figure 4: Comparison of TLT mean accuracy and its 95% confidence interval ($N = 5$) between MAC and DAFT MAC with varying reasoning steps.

result supports the qualitative result shown before and demonstrates that DAFT MAC consistently results in simplified reasoning paths across the whole dataset, rather than only in a few cherry-picked examples. In section 5.2, we have argued that the 4-step is enough for solving CLEVR. In figure 4, one can see that step-wise growth reaches its maximum in 4-step (for clear view, see figure 10 in the appendix). It tells that the model requires more space to navigate its focus when the step size is smaller than four.

Figure 5 shows how much TLT each question type yields. Since TLT grows with the size of the reasoning step, we employed a relative value of TLT to normalize this value across different numbers of training steps. Relative TLT is defined as ${}^{TLT_t(question\_type)}/_{TLT_t}$, where t ranges over steps in figure 4. The fact that each question type's relative TLT has the same order within both MAC and DAFT MAC substantiates TLT's ability to measure reasoning complexity regardless of the specific architecture.

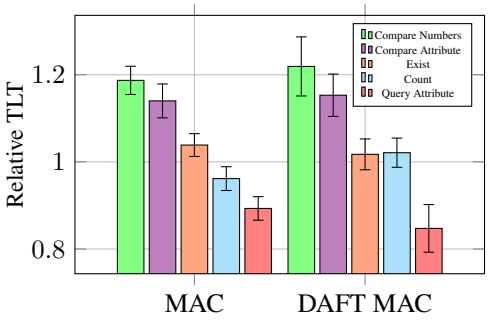

Figure 5: Comparison of relative TLT mean accuracy and its 95% confidence interval ($N = 50$) with varying question type.

Question type *Compare Numbers* and *Compare Attribute* had higher TLT than other question types. This is expected since such comparative questions require more NP chunks than other question types. When we shrank the step size from four to two, the accuracy of *Query Attribute* question type was pretty much unharmed (99.6 → 99.3 in DAFT MAC and 99.6 → 97.5 in MAC) while that of other question types significantly dropped. This is supported by the fact that *Query Attribute* question type had lowest TLT, meaning the question type is solvable using a small number of steps.

6 Conclusion

This paper introduces a method that embeds the human prior of continuous focus transition called the Dynamics of Attention for Focus Transition (DAFT). In contrast to previous approaches, DAFT models the dynamics in-between reasoning steps, yielding more interpretable attention maps. When applied to MAC, the state of the art among models that only use natural supervision, DAFT achieves the same performance while using $\frac{1}{3}$ the number of steps. In addition, we proposed a novel metric called Total Length Transition (TLT). Following the simplicity principle, TLT measures how good the model is on planning effective, short reasoning path, which is directly related to the interpretability of the model.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering. *arXiv preprint arXiv:1505.00468*, 2015.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

Léon Bottou. From machine learning to machine reasoning. *Machine learning*, 94(2):133–149, 2014.

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, pages 6571–6583, 2018.

John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. *arXiv preprint arXiv:1904.01681*, 2019.

Jacob Feldman. The simplicity principle in perception and cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, 7(5):330–340, 2016.

Dedre Gentner. Bootstrapping the mind: Analogical processes and symbol systems. *Cognitive Science*, 34(5):752–775, 2010.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017.

Eldad Haber and Lars Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34 (1):014004, 2017.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 804–813, 2017.

Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067*, 2018.

Drew A Hudson and Christopher D Manning. Gqa: a new dataset for compositional question answering over real-world images. *arXiv preprint arXiv:1902.09506*, 2019.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017a.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2989–2998, 2017b.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1564–1574, 2018.

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

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

Isaac Lage, Andrew Ross, Samuel J Gershman, Been Kim, and Finale Doshi-Velez. Human-in-the-loop interpretability prior. In *Advances in Neural Information Processing Systems*, pages 10159–10168, 2018.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

Yiping Lu, Aoxiao Zhong, Quanzheng Li, and Bin Dong. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. *arXiv preprint arXiv:1710.10121*, 2017.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. 2018.

David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4942–4950, 2018.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. Manipulating and measuring model interpretability. *arXiv preprint arXiv:1802.07810*, 2018.

Lars Ruthotto and Eldad Haber. Deep neural networks motivated by partial differential equations. *arXiv preprint arXiv:1804.04272*, 2018.

Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

Joseph Suarez, Justin Johnson, and Fei-Fei Li. Ddrprog: A clevr differentiable dynamic reasoning programmer. *arXiv preprint arXiv:1803.11361*, 2018.

Michael S Vendetti and Silvia A Bunge. Evolutionary and developmental changes in the lateral frontoparietal network: a little goes a long way for higher-level cognition. *Neuron*, 84(5):906–917, 2014.

Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. In *International conference on machine learning*, pages 2397–2406, 2016.

Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Advances in Neural Information Processing Systems*, pages 1031–1042, 2018.

A Reasoning with Additional Supervision

We taxonomize previous research on visual reasoning according to the additional supervision signals used.

Program CLEVR, along with the recently proposed GQA dataset [Hudson and Manning, 2019] provide program supervision in addition to the natural supervisions (image, question, answer). Program supervision enables neural models to learn to generate a program from the question. Such a generated program can then be used in a logical inference engine [Yi et al., 2018] or to build a layout for neural modules to answer the question [Andreas et al., 2016, Hu et al., 2017, Mascharka et al., 2018]. While models learned with program supervision tend to perform better, the process of acquiring such program semantics tree data is costly.

Object Mask Yi et al. [2018] generated object masks for the CLEVR dataset using its generation code, and used it as a supervisory signal to solve VQA task. Object masks and their corresponding inference module such as Mask R-CNN [He et al., 2017] are enough to build an exact scene graph. Therefore, such object mask data enable us to fully disentangle the model's reasoning from the neural representations of input image and question. While such models achieve state-of-the-art performance on CLEVR, it is infeasible for most non-synthetic dataset since acquiring object masks is very costly.

B Implementation Details

To solve ODE initial value problem, we used `torchdiffeq` [Chen et al., 2018]. For designing and accelerating the computation graph of the model, we used `pytorch` 1.0.1 [Paszke et al., 2017] and CUDA 9.2 on an Nvidia V100 GPU. Every experiment was performed with five different initial seeds by fixing the inital seed with `manual_seed()` for `python`, `pytorch`, and `numpy`. We used the `Adam` optimizer [Kingma and Ba, 2014] with learning rate 1e-4 for all experiments, and halved the learning rate whenever the validation accuracy stopped improving for more than one epoch. We trained using batches of 64 training data. The size of all hidden dimensions was fixed to 512 except for the word embedding layer, which was 300. All weights for the affine transformation were initialized with `xavier` initialization [Glorot and Bengio, 2010], and word embeddings were initialized to random vectors using a uniform distribution following the settings of MAC.

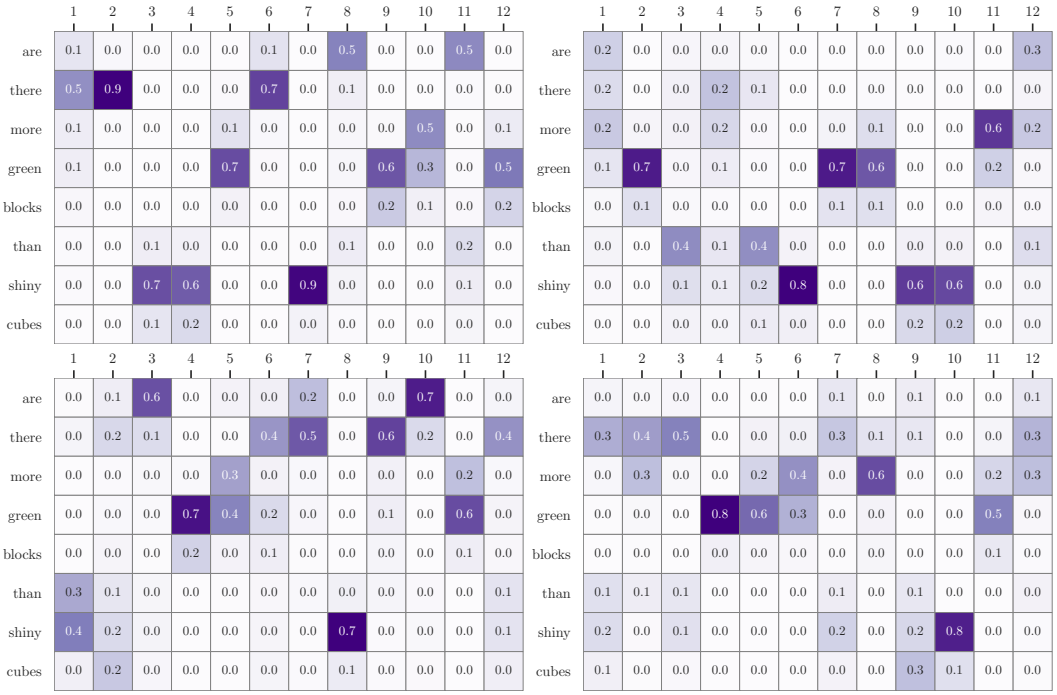

Figure 6: Attention maps from the other four 12-step MACs initialized with different seeds, distributed over question *"Are there more green blocks than shiny cubes?"*. All of them perform similiarly to the model used in figure 3 in terms of CLEVR validation accuracy.

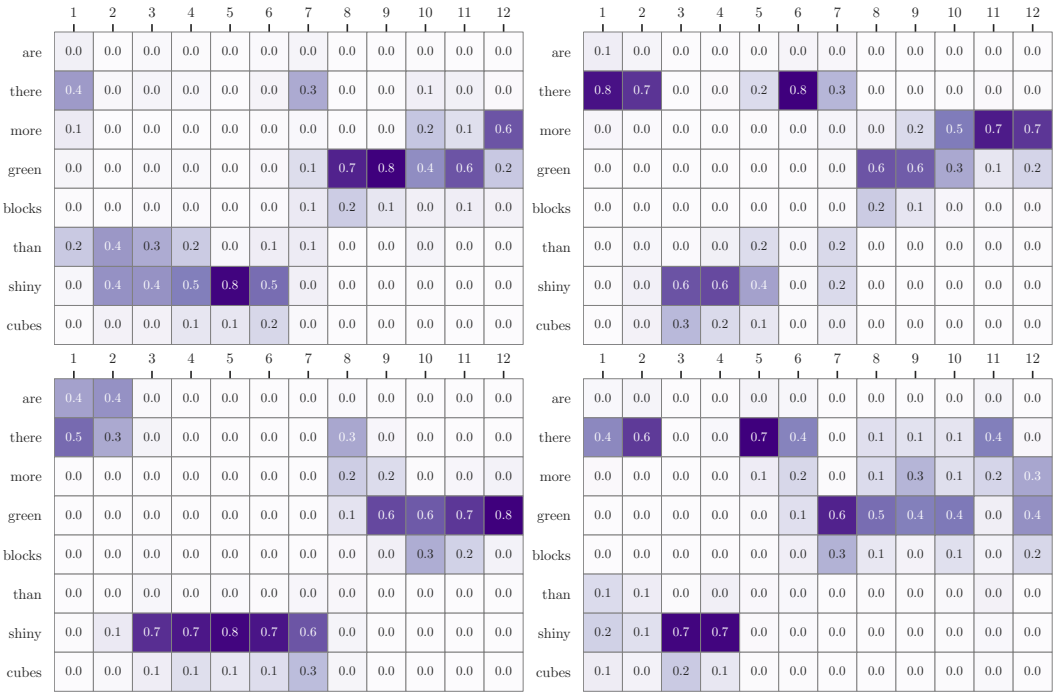

Figure 7: Attention maps from the other four 12-step DAFT MACs initialized with different seeds, distributed over question *"Are there more green blocks than shiny cubes?"*. All of them perform similiarly to the model used in figure 3 in terms of CLEVR validation accuracy.

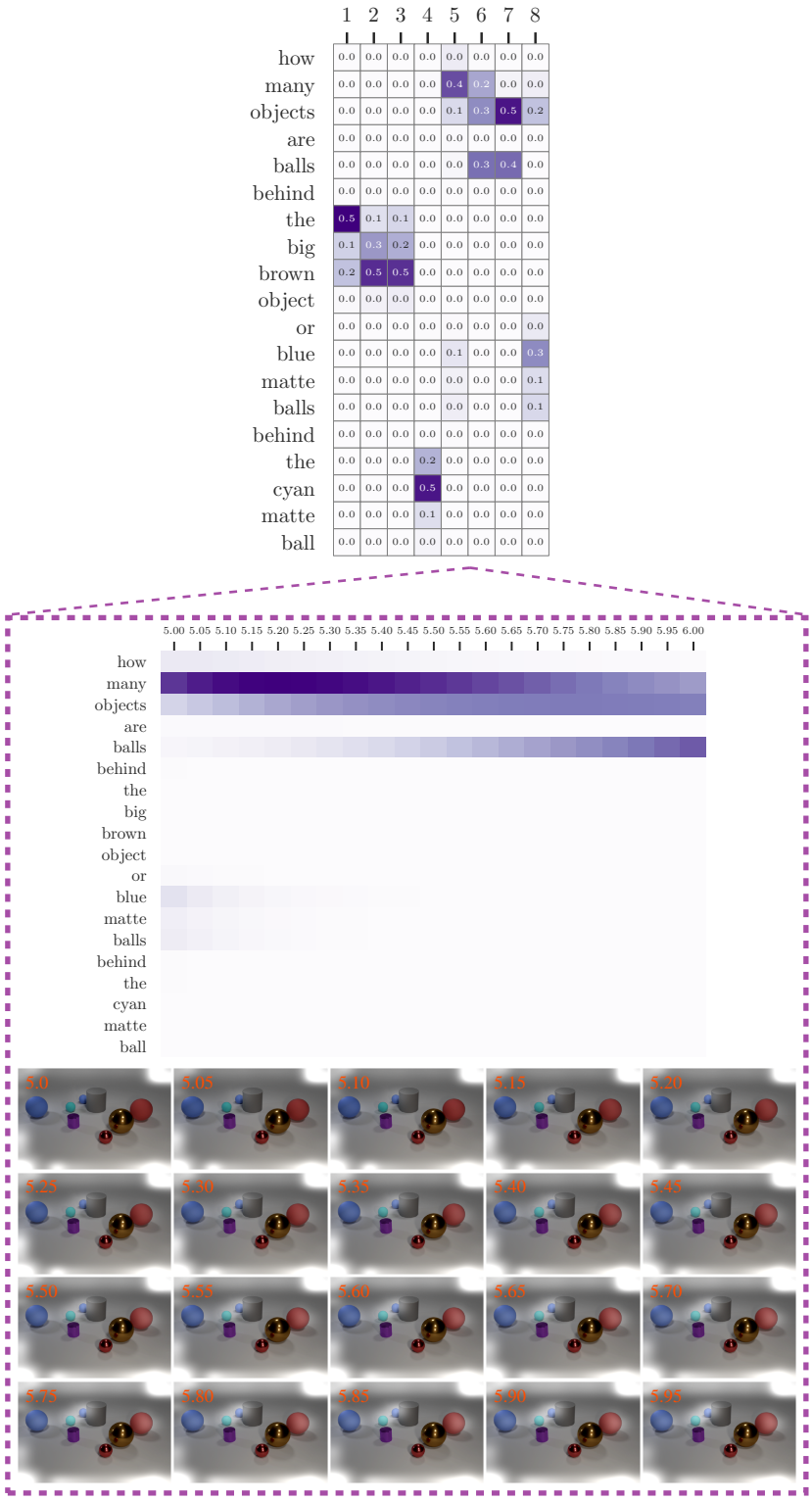

Figure 8: Interpolation in-between steps. Since the solution of IVP is a continuous function of time, we can get a attention map for any given intermediate time value. This fact enables infinitely fine-grained interpolation. Also note that this is not a linear interpolation, see how the attention on *many* reaches a maximum around 5.2 instead of on either end.

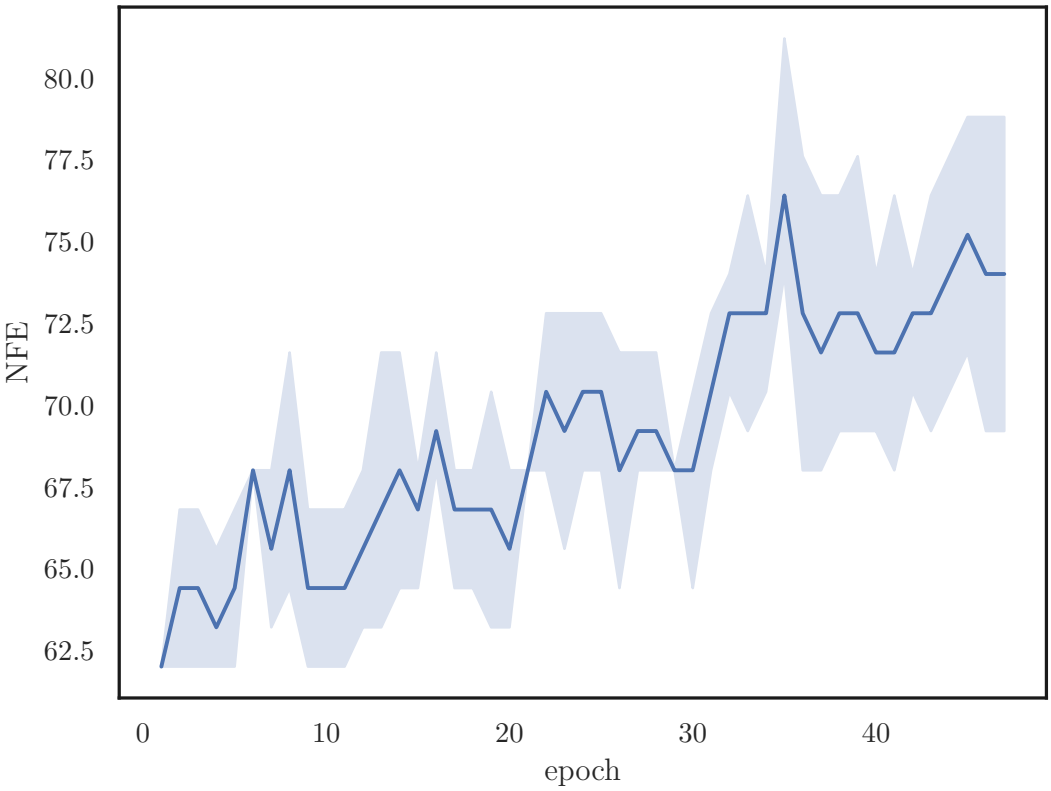

Figure 9: Growth of the Number of Function Evaluation (NFE) for 4-step DAFT MAC as training progresses. Mean value and 95% confidence interval ($N = 5$) are denoted as line and gradation.

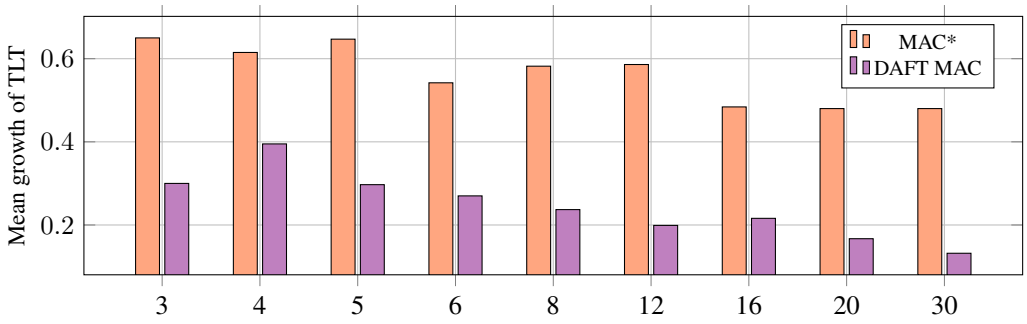

Figure 10: Mean growth of TLT that starts from 2-step. Bars denote arithmetic mean value of given interval. For example, the bar at 12 represents $\frac{TLT_{12}-TLT_2}{10}$. The figure is linked with figure 4.

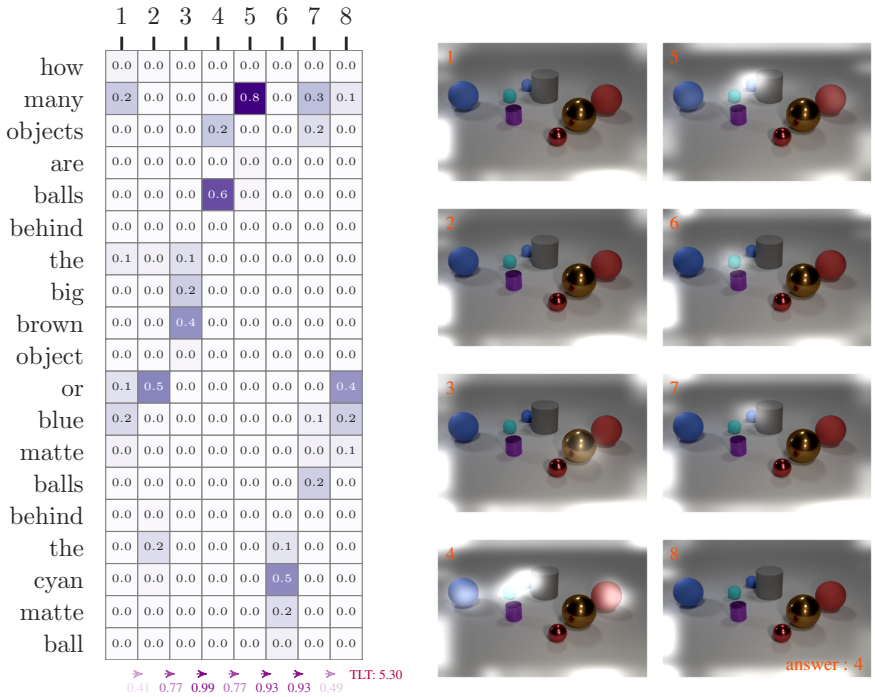

Figure 11: Attention maps of 8-step MAC, distributed over question *"How many objects are balls behind the big brown object or blue matte balls behind the cyan matte ball?"*. This model achieves 99% CLEVR validation accuracy.

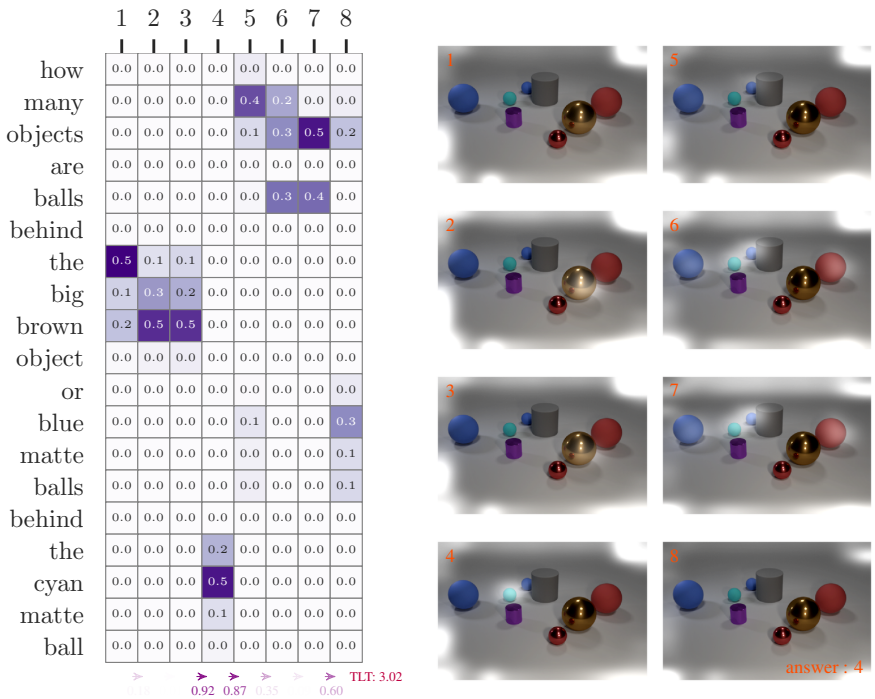

Figure 12: Attention maps of 8-step DAFT MAC, distributed over question *"How many objects are balls behind the big brown object or blue matte balls behind the cyan matte ball?"*. This model achieves 99% CLEVR validation accuracy.

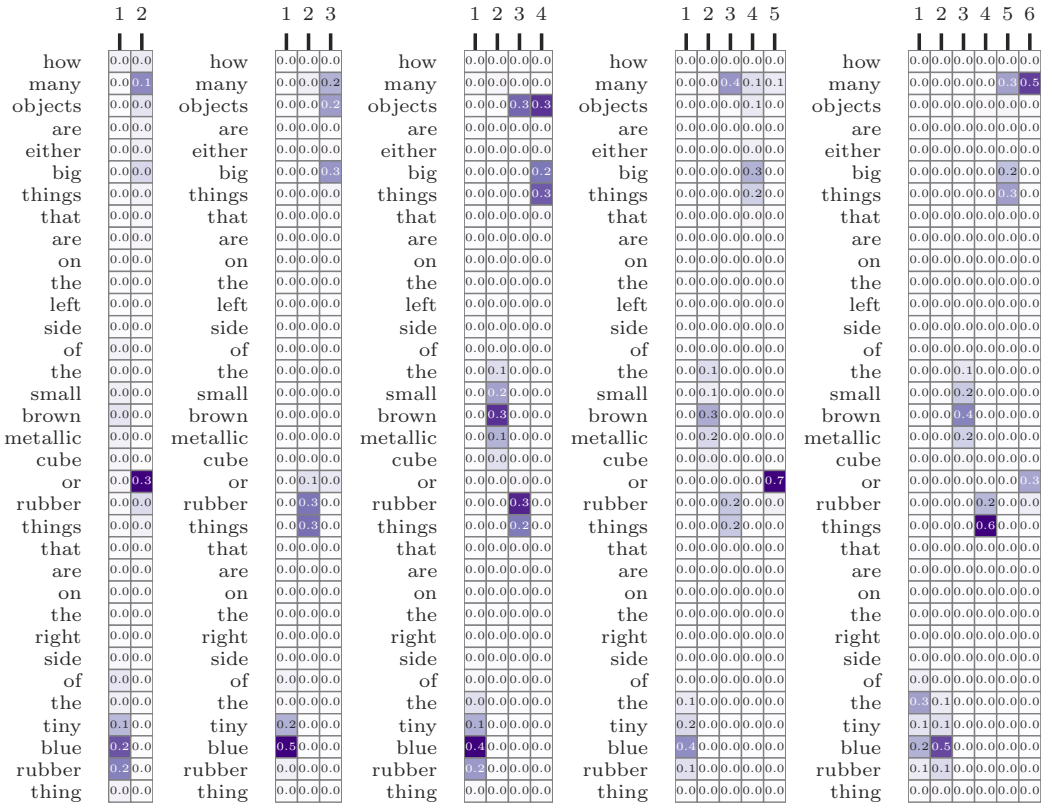

Figure 13: Attention maps of DAFT MAC with 2 to 6 steps, distributed over the very long question *"How many objects are either big things that are on the left side of the small brown metallic cube or rubber things that are on the right side of the tiny blue rubber thing?"*. Note that these five models are seperately initialized and thus have totally *different* parameters. The order of transition is unchanged among these completely separate models with different expressive power.

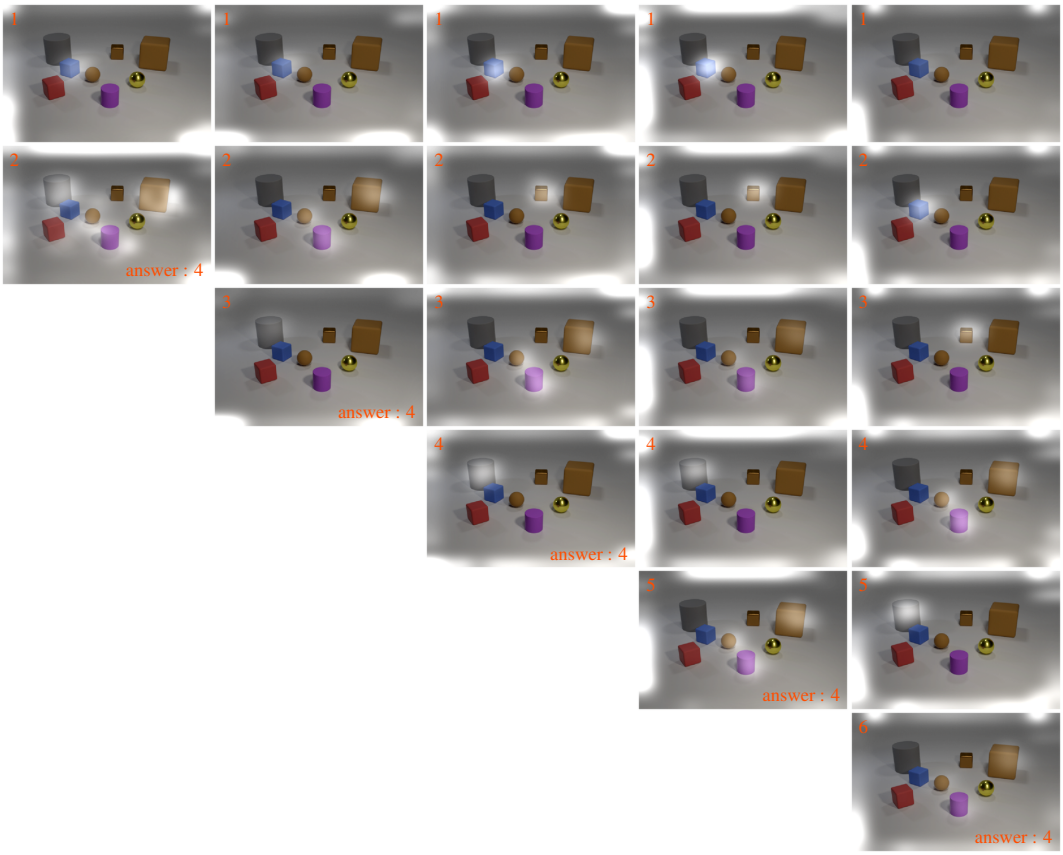

Figure 14: Accompanying image attention maps for figure 13.