Interactive publications! Python example

New paper

(in this case, a preprint)

+

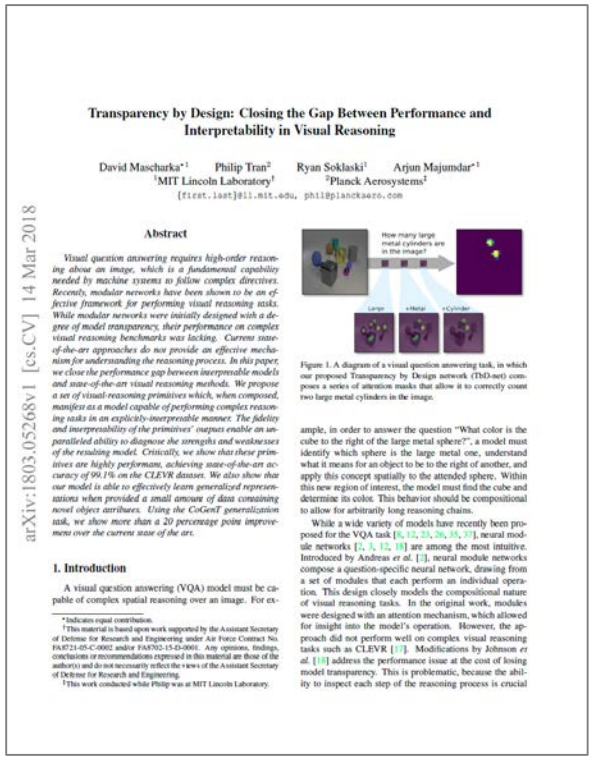
Website explaining paper

(has code to replicate experiments + plots and explicit software versions)

+

Interactive Jupyter notebook with code to apply their method to your own images

(with all required files and software ready to run in your browser with no installation required, thanks to Binder)



Transparency by Design: Closing the Gap Between Performance and Interpretability in Visual Reasoning

David Mascharka*[1] Philip Tran[2] Ryan Soklaski[1] Arjun Majumdar*[1]

[1]MIT Lincoln Laboratory[1] [2]Planck Aerosystems[2]

[first.last]@ll.mit.edu, phil@planckaero.com

Abstract

Visual question answering requires high-order reasoning about an image, which is a fundamental capability needed by machine systems to follow complex directives. Recently, modular networks have been shown to be an effective framework for performing visual reasoning tasks. While modular networks were initially designed with a degree of model transparency, their performance on complex visual reasoning benchmarks was lacking. Current state-of-the-art approaches do not provide an effective mechanism for understanding the reasoning process. In this paper, we close the performance gap between interpretable models and state-of-the-art visual reasoning methods. We propose a set of visual-reasoning primitives which, when composed, manifest as a model capable of performing complex reasoning tasks in an explicitly-interpretable manner. The fidelity and interpretability of the primitives' outputs enable an unparalleled ability to diagnose the strengths and weaknesses of the resulting model. Critically, we show that these primitives are highly performant, achieving state-of-the-art accuracy of 99.1% on the CLEVR dataset. We also show that our model is able to effectively learn generalized representations when provided a small amount of data containing novel object attributes. Using the CoGenT generalization task, we show more than a 20 percentage point improvement over the current state of the art.

1. Introduction

A visual question answering (VQA) model must be capable of complex spatial reasoning over an image. For example, in order to answer the question "What color is the cube to the right of the large metal sphere?", a model must identify which sphere is the large metal one, understand what it means for an object to be to the right of another, and apply this concept spatially to the attended sphere. Within this new region of interest, the model must find the cube and determine its color. This behavior should be compositional to allow for arbitrarily long reasoning chains.

While a wide variety of models have recently been proposed for the VQA task [8, 12, 23, 26, 35, 37], neural module networks [2, 3, 12, 18] are among the most intuitive. Introduced by Andreas *et al.* [3], neural module networks compose a question-specific neural network, drawing from a set of modules that each perform an individual operation. This design closely models the compositional nature of visual reasoning tasks. In the original work, modules were designed with an attention mechanism, which allowed for insight into the model's operation. However, the approach did not perform well on complex visual reasoning tasks such as CLEVR [17]. Modifications by Johnson *et al.* [18] address the performance issue at the cost of losing model transparency. This is problematic, because the ability to inspect each step of the reasoning process is crucial

Figure 1. A diagram of a visual question answering task, in which our proposed Transparency by Design network (TbD-net) composes a series of attention masks that allow it to correctly count two large metal cylinders in the image.

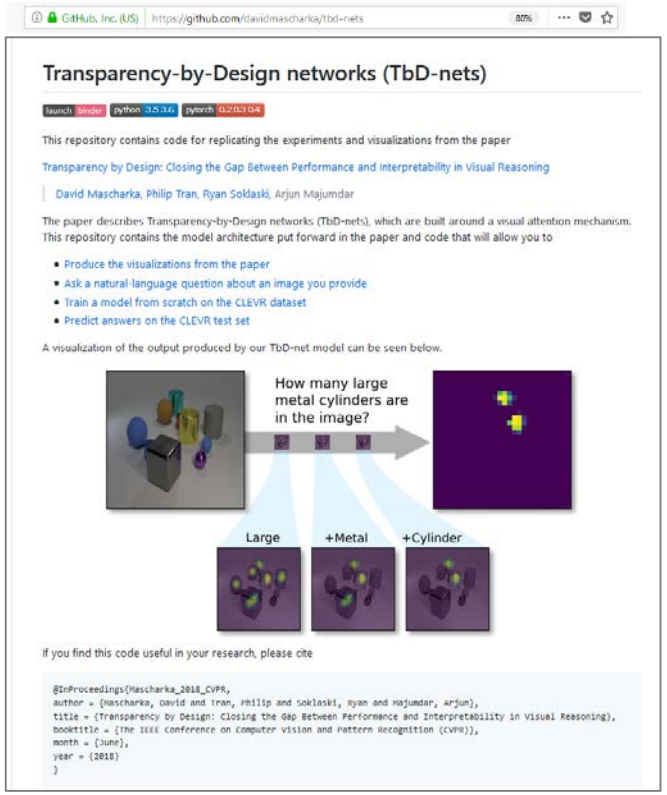
*Indicates equal contribution.

†This material is based upon work supported by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract No. FA8721-05-C-0002 and/or FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Assistant Secretary of Defense for Research and Engineering.

‡This work conducted while Philip was at MIT Lincoln Laboratory.

arXiv:1803.05268v1 [cs.CV] 14 Mar 2018

https://arxiv.org/abs/1803.05268



GitHub, Inc. (US) https://github.com/davidmascharka/tbd-nets

Transparency-by-Design networks (TbD-nets)

launch binder python 3.5 3.6 pytorch 0.2 0.3 0.4

This repository contains code for replicating the experiments and visualizations from the paper

Transparency by Design: Closing the Gap Between Performance and Interpretability in Visual Reasoning

David Mascharka, Philip Tran, Ryan Soklaski, Arjun Majumdar

The paper describes Transparency-by-Design networks (TbD-nets), which are built around a visual attention mechanism. This repository contains the model architecture put forward in the paper and code that will allow you to

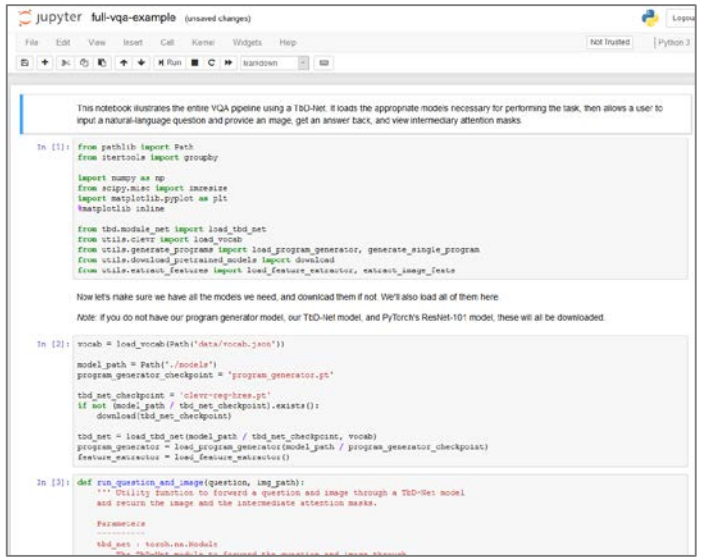
- Produce the visualizations from the paper
- Ask a natural-language question about an image you provide
- Train a model from scratch on the CLEVR dataset
- Predict answers on the CLEVR test set

A visualization of the output produced by our TbD-net model can be seen below.

If you find this code useful in your research, please cite

```
@InProceedings{Mascharka_2018_CVPR,
author = {Mascharka, David and Tran, Philip and Soklaski, Ryan and Majumdar, Arjun},
title = {Transparency by Design: Closing the Gap Between Performance and Interpretability in Visual Reasoning},
booktitle = {The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)},
month = {June},
year = {2018}
}
```

https://github.com/davidmascharka/tbd-nets



jupyter full-vqa-example (unsaved changes)

This notebook illustrates the entire VQA pipeline using a TbD-Net. It loads the appropriate models necessary for performing the task, then allows a user to input a natural-language question and provide an image, get an answer back, and view intermediary attention masks.

```
In [1]: from pathlib import Path
        from itertools import groupby

        import numpy as np
        from scipy.misc import imresize
        import matplotlib.pyplot as plt
        %matplotlib inline

        from tbd.module_net import load_tbd_net
        from utils.clevr import load_vocab
        from utils.generate_programs import load_program_generator, generate_single_program
        from utils.download_pretrained_models import download
        from utils.extract_features import load_feature_extractor, extract_image_featu
```

Now let's make sure we have all the models we need, and download them if not. We'll also load all of them here.

Note: if you do not have our program generator model, our TbD-Net model, and PyTorch's ResNet-101 model, these will all be downloaded.

```
In [2]: vocab = load_vocab(Path('data/vocab.json'))

        model_path = Path('./models')
        program_generator_checkpoint = 'program_generator.pt'

        tbd_net_checkpoint = 'clevr-reg-hres.pt'
        if not (model_path / tbd_net_checkpoint).exists():
            download(tbd_net_checkpoint)

        tbd_net = load_tbd_net(model_path / tbd_net_checkpoint, vocab)
        program_generator = load_program_generator(model_path / program_generator_checkpoint)
        feature_extractor = load_feature_extractor()
```

```
In [3]: def run_question_and_image(question, img_path):
            ''' Utility function to forward a question and image through a TbD-Net model
            and return the image and the intermediate attention masks.

            Parameters
            ----------
            tbd_net : torch.nn.Module
```

https://mybinder.org/v2/gh/davidmascharka/tbd-nets/binder?filepath=full-vqa-example.ipynb

Additional reference: Link to tweet by first author publicizing this work

Demo – reproduce figures from paper on the previous slide

1. See the paper: *https://arxiv.org/abs/1803.05268*
   - *Note the PDF includes link to a GitHub repo*

2. Inspect the code
   - *Open the associated GitHub repository containing the Jupyter notebook used to produce the figures for the paper*
     - *Repo:* https://github.com/davidmascharka/tbd-nets
     - *Notebook: https://github.com/davidmascharka/tbd-nets/blob/master/visualize-output.ipynb*

3. Reproduce the visualizations without installing anything!
   - *Click the* launch binder *button on the main page of the repo*

Interactive publications! R example

New paper
(open access)

ECOLOGY LETTERS

Ecology Letters, (2018) doi: 10.1111/ele.13085

REVIEW AND SYNTHESIS

From noise to knowledge: how randomness generates novel phenomena and reveals information

Carl Boettiger*

Dept of Environmental Science, Policy, and Management, University of California Berkeley, Berkeley, CA 94720-3114, USA

*Correspondence: E-mail: cboettig@berkeley.edu

Abstract

Noise, as the term itself suggests, is most often seen a nuisance to ecological insight, a inconvenient reality that must be acknowledged, a haystack that must be stripped away to reveal the processes of interest underneath. Yet despite this well-earned reputation, noise is often interesting in its own right: noise can induce novel phenomena that could not be understood from some underlying deterministic model alone. Nor is all noise the same, and close examination of differences in frequency, colour or magnitude can reveal insights that would otherwise be inaccessible. Yet with each aspect of stochasticity leading to some new or unexpected behaviour, the time is right to move beyond the familiar refrain of "everything is important" (Bjørnstad & Grenfell 2001). Stochastic phenomena can suggest new ways of inferring process from pattern, and thus spark more dialog between theory and empirical perspectives that best advances the field as a whole. I highlight a few compelling examples, while observing that the study of stochastic phenomena are only beginning to make this translation into empirical inference. There are rich opportunities at this interface in the years ahead.

Keywords

Coloured noise, demographic noise, environmental noise, quasi-cycles, stochasticity, tipping points.

Ecology Letters (2018)

INTRODUCTION: NOISE THE NUISANCE

To many, stochasticity, or more simply, noise, is just that – something which obscures patterns we are trying to infer (Knape & de Valpine 2011); and an ever richer batteries of statistical methods are developed largely in an attempt to strip away this undesirable randomness to reveal the patterns beneath (Coulson 2001). Over the past several decades, literature in stochasticity has transitioned from thinking of stochasticity in such terms; where noise is a nuisance that obscures the deterministic skeleton of the underlying mechanisms, to the recognition that stochasticity can itself be a mechanism for driving many interesting phenomena (Coulson *et al.* 2004). Yet this transition from *noise the nuisance* to *noise the creator* of ecological phenomena has had, with a few notable exceptions, relatively little impact in broader thinking about stochasticity. One of the most provocative of those exceptions has turned the classical notion of noise the nuisance on its head: recognising that noise driven phenomena can become a tool to reveal underlying processes: to become *noise the informer*. Here I argue that this third shift in perspective offers an opportunity to better bridge the divide between respective primarily theoretical and primarily empirical communities by seeing noise not as mathematical curiosity or statistical bugbear, but as a source for new opportunities for inference.

In arguing for this shift, it essential to recognise this is a call for a bigger tent, not for the rejection of previous paradigms. What I will characterize as 'noise the nuisance' reflects a predominately statistical approach, in which noise, almost by definition, represents all the processes we are not interested in that create additional variation which might obscure the pattern of interest. By contrast, an extensive literature has long explored how noise itself can create patterns and explain processes from population cycling to coexistence. These broad categories should be seen as a spectrum and not be mistaken for either a sharp dichotomy nor a reference to a strictly empirical-theoretical divide. Each paradigm expands upon rather than rejects the previous notion of noise: the recognition that noise can create novel phenomena does not mean that noise cannot also obscure the signal of some process of interest. Likewise, seeking to use noise as a novel source of information about underlying processes will be informed by both previous paradigms, as our discussion will illustrate.

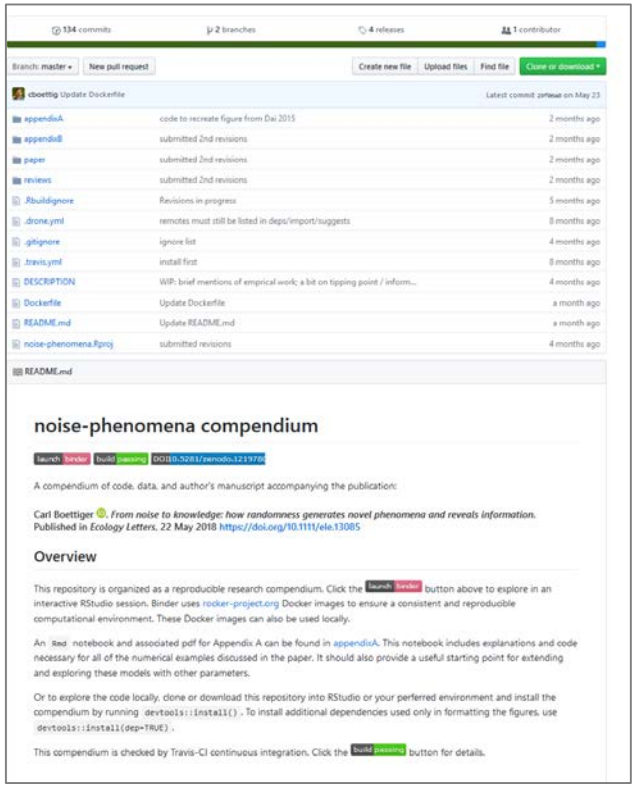
Accompanying this discussion, I provide concise and commented code for simulating each of the models we will discuss as Appendix S1, and more mathematical background and derivations in Appendix S2. Numerical simulations permit poking and prodding investigation unencumbered by either experimental design or mathematical formalism. A copy of these appendices is maintained at https://github.com/cboettig/noise-phenomena.

To emphasise the underlying trend in the changing roles in which we see and understand noisy processes, I will also restrict my focus to relatively simple models primarily from population ecology context. Simplicity not only makes examples (in equations and in code) more tractable but also allows us to focus on aspects that are germane to many contexts rather than unique to particular complexities (Bartlett 1960;

© 2018 John Wiley & Sons Ltd/CNRS

https://onlinelibrary.wiley.com/doi/epdf/10.1111/ele.13085

Corresponding "compendium"
(with text, code, data)



noise-phenomena compendium

A compendium of code, data, and author's manuscript accompanying the publication:

Carl Boettiger. *From noise to knowledge: how randomness generates novel phenomena and reveals information.* Published in *Ecology Letters*, 22 May 2018 https://doi.org/10.1111/ele.13085

Overview

This repository is organized as a reproducible research compendium. Click the launch binder button above to explore in an interactive RStudio session. Binder uses rocker-project.org Docker images to ensure a consistent and reproducible computational environment. These Docker images can also be used locally.

An Rmd notebook and associated pdf for Appendix A can be found in appendixA. This notebook includes explanations and code necessary for all of the numerical examples discussed in the paper. It should also provide a useful starting point for extending and exploring these models with other parameters.

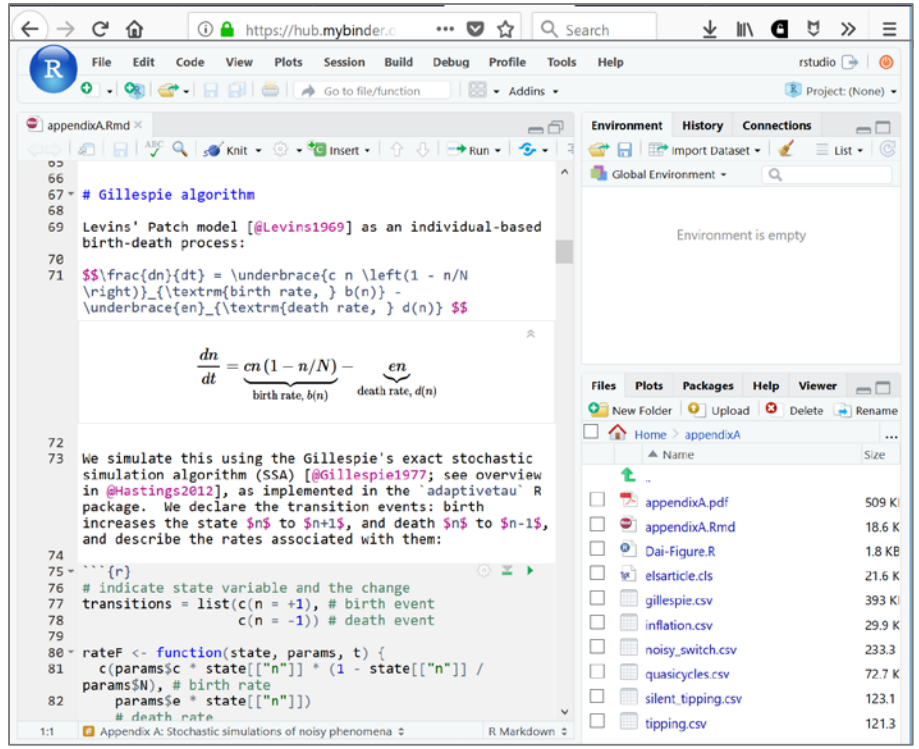
Or to explore the code locally, clone or download this repository into RStudio or your perferred environment and install the compendium by running `devtools::install()`. To install additional dependencies used only in formatting the figures, use `devtools::install(dep=TRUE)`.

This compendium is checked by Travis-CI continuous integration. Click the build passing button for details.

https://github.com/cboettig/noise-phenomena

Interactive R Markdown notebook to explore code

(with all required files and software ready to run in Rstudio in your browser with no installation required, thanks to Binder)



https://hub.mybinder.c

File Edit Code View Plots Session Build Debug Profile Tools Help

appendixA.Rmd

```
# Gillespie algorithm

Levins' Patch model [@Levins1969] as an individual-based
birth-death process:

$$\frac{dn}{dt} = \underbrace{c n \left(1 - n/N
\right)}_{\textrm{birth rate, } b(n)} -
\underbrace{en}_{\textrm{death rate, } d(n)} $$

$$\frac{dn}{dt} = \underbrace{cn(1-n/N)}_{\text{birth rate, } b(n)} - \underbrace{en}_{\text{death rate, } d(n)}$$

We simulate this using the Gillespie's exact stochastic
simulation algorithm (SSA) [@Gillespie1977; see overview
in @Hastings2012], as implemented in the `adaptivetau` R
package.  We declare the transition events: birth
increases the state $n$ to $n+1$, and death $n$ to $n-1$,
and describe the rates associated with them:

```{r}
# indicate state variable and the change
transitions = list(c(n = +1), # birth event
                   c(n = -1)) # death event

rateF <- function(state, params, t) {
  c(params$c * state[["n"]] * (1 - state[["n"]] /
params$N), # birth rate
    params$e * state[["n"]])
  # death rate
```

Environment History Connections

Import Dataset List

Global Environment

Environment is empty

Files Plots Packages Help Viewer

New Folder Upload Delete Rename

Home appendixA

Name	Size
appendixA.pdf	509 KI
appendixA.Rmd	18.6 K
Dai-Figure.R	1.8 KB
elsarticle.cls	21.6 K
gillespie.csv	393 KI
inflation.csv	29.9 K
noisy_switch.csv	233.3
quasicycles.csv	72.7 K
silent_tipping.csv	123.1
tipping.csv	121.3

1:1 Appendix A: Stochastic simulations of noisy phenomena R Markdown

https://github.com/cboettig/noise-phenomena

Additional reference: Link to tweet by author (Carl Boettiger)
```