FGAI4H-E-025-A1
Geneva, 30 May - 1 June 2019

Robustness - Safety and Reliability in AI4H
FGAI4H-E

Wojciech Samek, Vignesh Srinivasan, **Luis Oala**

Fraunhofer HHI

luis.oala@hhi.fraunhofer.de

May 30, 2019

Context and Motivation

Big Goal: Safe and reliable AI systems

Observation: Decades of AI research has produced a plethora of tools and methods that deal with safety and reliability

Small Goal: Organize existing tools into meaningful action areas and map them along the life cycle of an AI4H system

Why Robustness?

Huber: "Insensitivity to small deviations from the assumptions" [Huber, 1981]

Dietterich: Known unknowns and unknown unknowns [Dietterich, 2017]

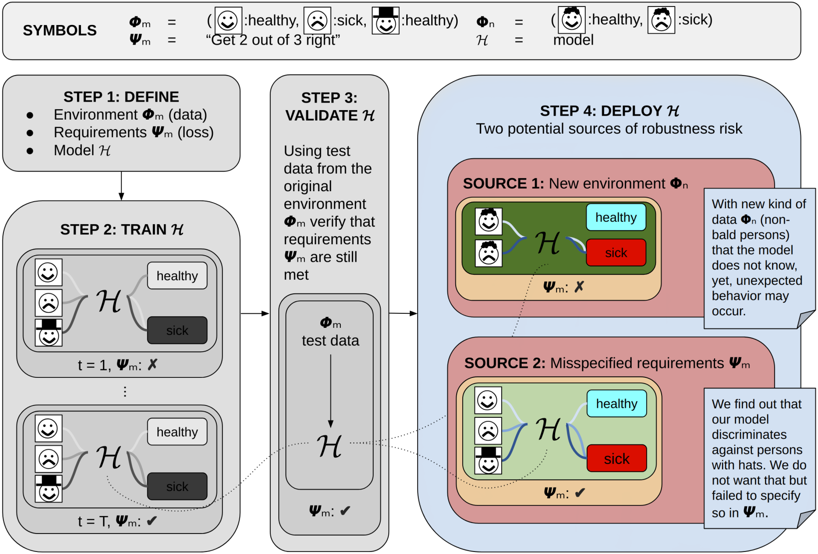
Russell: Validity [Russell et al., 2015]

Safe AI: Adversarial robustness, FAT, misspecification problems

---

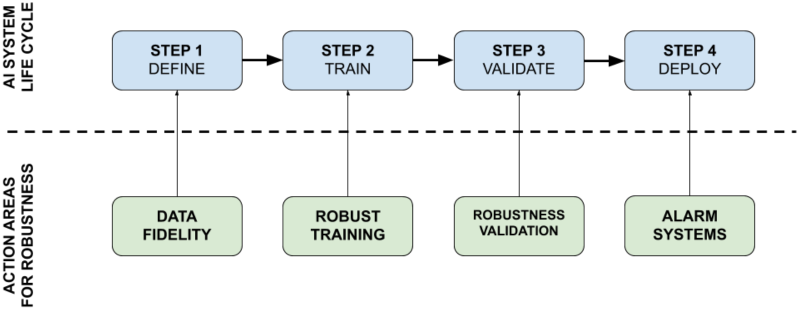
Working def.: No gross, unexpected errors under slight changes of the operating environment; benign error handling

The AI System Life Cycle and Robustness Risks



SYMBOLS
Φ_m = (:healthy, :sick, :healthy)
Ψ_m = "Get 2 out of 3 right"
Φ_n = (:healthy, :sick)
$\mathcal{H}$ = model
STEP 1: DEFINE
Environment Φ_m (data)
Requirements Ψ_m (loss)
Model $\mathcal{H}$
STEP 2: TRAIN $\mathcal{H}$
healthy
sick
t = 1, Ψ_m: ✗
t = T, Ψ_m: ✔
STEP 3: VALIDATE $\mathcal{H}$
Using test data from the original environment Φ_m verify that requirements Ψ_m are still met
Φ_m test data
Ψ_m: ✔
STEP 4: DEPLOY $\mathcal{H}$
Two potential sources of robustness risk
SOURCE 1: New environment Φ_n
Ψ_m: ✗
With new kind of data Φ_n (non-bald persons) that the model does not know, yet, unexpected behavior may occur.
SOURCE 2: Misspecified requirements Ψ_m
Ψ_m: ✔
We find out that our model discriminates against persons with hats. We do not want that but failed to specify so in Ψ_m.

Four Action Areas to Mitigate Robustness Risks


AI SYSTEM
LIFE CYCLE
STEP 1
DEFINE
STEP 2
TRAIN
STEP 3
VALIDATE
STEP 4
DEPLOY
ACTION AREAS
FOR ROBUSTNESS
DATA
FIDELITY
ROBUST
TRAINING
ROBUSTNESS
VALIDATION
ALARM
SYSTEMS

Data Fidelity

Impose desiderata on data that are used as input to an AI system

Examples

- Datasheets for datasets [Gebru et al., 2018]
- Normalization and standardization, e.g.
  - Zero-centering
  - Decorrelation
  - Whitening

Robust Training

Expose an AI system to changes in the data environment during training that would otherwise be likely to induce robustness risks during deployment

Examples

- Adversarial training [Madry et al., 2017]

$$\min_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{y}\sim\mathcal{D}}[\max_{\boldsymbol{\delta}\in\mathcal{S}} J(\mathbf{x}+\boldsymbol{\delta},\mathbf{y})] \tag{1}$$

- Stability training [Zheng et al., 2016]

$$L(\mathbf{x},\mathbf{x}';\boldsymbol{\theta}) = L_0(\mathbf{x};\boldsymbol{\theta}) + \alpha L_{\text{stability}}(\mathbf{x},\mathbf{x}';\boldsymbol{\theta}) \tag{2}$$

Robustness Validation

Verify the robustness of an AI system in a controlled testing environment

Examples

- Hypothesis testing (if amenable)
- Perturbation and adversarial stress tests [Hendrycks and Dietterich, 2019, Madry et al., 2017]
- FAT misspecification testing, e.g. *predictive equality* [Hardt et al., 2016]

$$\mathbb{E}[d(\mathbf{x})|y = 0, g(\mathbf{x})] = \mathbb{E}[d(\mathbf{x})|y = 0]$$

Alarm Systems

Flag unusual behavior of the AI system during deployment

Examples

- Outlier tests via generative modelling [Meng and Chen, 2017]
- Attribution methods (see [Ancona et al., 2017] for overview), e.g.

$$R_j = x_j \frac{\delta y_c}{\delta x_j}$$

- Uncertainty quantification, e.g. [Gal and Ghahramani, 2016]

$$\mathbb{E}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*), \mathbb{V}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*)$$

Recommendations and Outlook

- Integrate established robustness tools into the AI4H system life cycle
- Screen for additional methods that can be included in the action areas

Bibliography I

Ancona, M., Ceolini, E., ztireli, C., and Gross, M. (2017).
Towards better understanding of gradient-based attribution methods for deep neural networks.
arXiv preprint arXiv:1711.06104.

Dietterich, T. G. (2017).
Steps toward robust artificial intelligence.
AI Magazine, 38(3):3–24.

Gal, Y. and Ghahramani, Z. (2016).
Dropout as a bayesian approximation: Representing model uncertainty in deep learning.
In *international conference on machine learning*, pages 1050–1059.

Bibliography II

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H., Daume III, H., and Crawford, K. (2018).
Datasheets for datasets.
arXiv preprint arXiv:1803.09010.

Hardt, M., Price, E., and Srebro, N. (2016).
Equality of opportunity in supervised learning.
In *Advances in neural information processing systems*, pages 3315–3323.

Hendrycks, D. and Dietterich, T. (2019).
Benchmarking neural network robustness to common corruptions and perturbations.
arXiv preprint arXiv:1903.12261.

Huber, P. J. (1981).
Robust statistics.
Wiley, New York.

Bibliography III

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017).
Towards deep learning models resistant to adversarial attacks.
arXiv preprint arXiv:1706.06083.

Meng, D. and Chen, H. (2017).
Magnet: a two-pronged defense against adversarial examples.
In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 135–147. ACM.

Russell, S., Dewey, D., and Tegmark, M. (2015).
Research priorities for robust and beneficial artificial intelligence.
Ai Magazine, 36(4):105–114.

Bibliography IV

Zheng, S., Song, Y., Leung, T., and Goodfellow, I. (2016).
Improving the robustness of deep neural networks via stability training.
In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 4480–4488.