Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

AI For People Workshop, 2020

9th August, 2020

Who am I ?

- Research Engineer at FAIR.
- Interested in Lifelong Learning and Reinforcement Learning.

Who am I ?

- Research Engineer at FAIR.
- Interested in Lifelong Learning and Reinforcement Learning.
- This talk is based on a recently published report [1].
- Worked on this report while I was a graduate student at Mila, University of Montreal.

[1]: https://www.towardtrustworthyai.com/

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development:
Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims

Highlights of the report

- Initiated with an interdisciplinary expert workshop in San Francisco in April 2019.

Highlights of the report

- Initiated with an interdisciplinary expert workshop in San Francisco in April 2019.

- Joint effort of multiple-stakeholders, 50+ authors from 25+ labs across industry and academia.

Highlights of the report

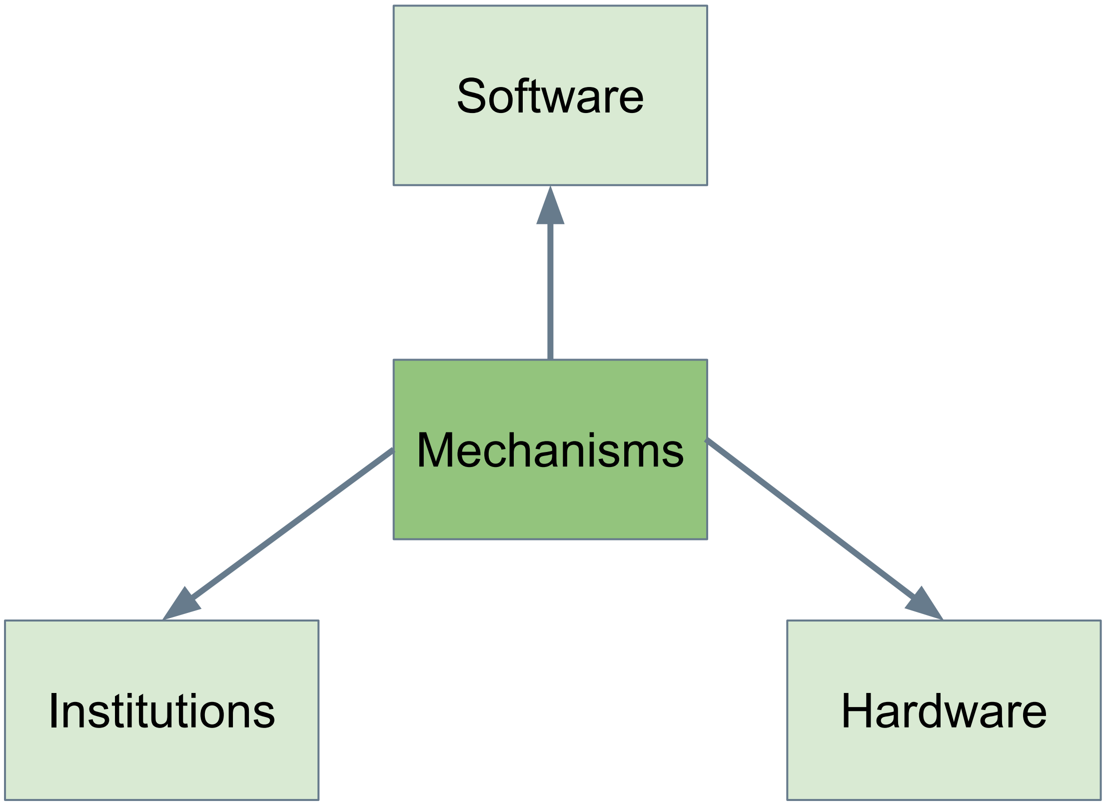
- Initiated with an interdisciplinary expert workshop in San Francisco in April 2019.
- Joint effort of multiple-stakeholders, 50+ authors from 25+ labs across industry and academia.
- Propose ten concrete mechanisms to move toward trustworthy AI development.

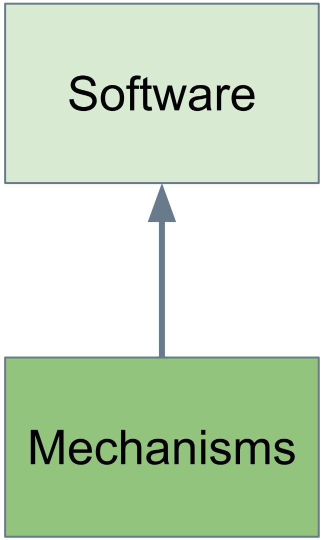
Highlights of the report

- Initiated with an interdisciplinary expert workshop in San Francisco in April 2019.
- Joint effort of multiple-stakeholders, 50+ authors from 25+ labs across industry and academia.
- Propose ten concrete mechanisms to move toward trustworthy AI development.
- https://www.towardtrustworthyai.com/

Miles Brundage[1†], Shahar Avin[3,2†], Jasmine Wang[4,29†‡], Haydn Belfield[3,2†], Gretchen Krueger[1†], Gillian Hadfield[1,5,30], Heidy Khlaaf[6], Jingying Yang[7], Helen Toner[8], Ruth Fong[9], Tegan Maharaj[4,28], Pang Wei Koh[10], Sara Hooker[11], Jade Leung[12], Andrew Trask[9], Emma Bluemke[9], Jonathan Lebensold[4,29], Cullen O'Keefe[1], Mark Koren[13], Théo Ryffel[14], JB Rubinovitz[15], Tamay Besiroglu[16], Federica Carugati[17], Jack Clark[1], Peter Eckersley[7], Sarah de Haas[18], Maritza Johnson[18], Ben Laurie[18], Alex Ingerman[18], Igor Krawczuk[19], Amanda Askell[1], Rosario Cammarota[20], Andrew Lohn[21], David Krueger[4,27], Charlotte Stix[22], Peter Henderson[10], Logan Graham[9], Carina Prunkl[12], Bianca Martin[1], Elizabeth Seger[16], Noa Zilberman[9], Seán Ó hÉigeartaigh[2,3], Frens Kroeger[23], Girish Sastry[1], Rebecca Kagan[8], Adrian Weller[16,24], Brian Tse[12,7], Elizabeth Barnes[1], Allan Dafoe[12,9], Paul Scharre[25], Ariel Herbert-Voss[1], Martijn Rasser[25], Shagun Sodhani[4,27], Carrick Flynn[8], Thomas Krendl Gilbert[26], Lisa Dyer[7], Saif Khan[8], Yoshua Bengio[4,27], Markus Anderljung[12]

[1]OpenAI, [2]Leverhulme Centre for the Future of Intelligence, [3]Centre for the Study of Existential Risk, [4]Mila, [5]University of Toronto, [6]Adelard, [7]Partnership on AI, [8]Center for Security and Emerging Technology, [9]University of Oxford, [10]Stanford University, [11]Google Brain, [12]Future of Humanity Institute, [13]Stanford Centre for AI Safety, [14]École Normale Supérieure (Paris), [15]Remedy.AI, [16]University of Cambridge, [17]Center for Advanced Study in the Behavioral Sciences,[18]Google Research, [19]École Polytechnique Fédérale de Lausanne, [20]Intel, [21]RAND Corporation, [22]Eindhoven University of Technology, [23]Coventry University, [24]Alan Turing Institute, [25]Center for a New American Security, [26]University of California, Berkeley, [27]University of Montreal, [28]Montreal Polytechnic, [29]McGill University, [30]Schwartz Reisman Institute for Technology and Society


Software
Mechanisms
Institutions
Hardware


Software
Mechanisms

Software Mechanisms

- Mechanisms to enable greater understanding AI systems.

Software Mechanisms

- Mechanisms to enable greater understanding AI systems.
- Can support claims such as:
  - *“This system is robust to distributional shifts”*
  - *“This system provides repeatable or reproducible results.”*

Software Mechanisms

Reproducibility

Software Mechanisms

Reproducibility

Formal Verification

Software Mechanisms

Reproducibility

Formal Verification

Validation of ML by ML

Software Mechanisms

- Reproducibility
- Formal Verification
- Validation of ML by ML
- Practical Verification

Software Mechanisms

- Reproducibility
- Formal Verification
- Validation of ML by ML
- Practical Verification

Reproducibility vs Replicability

Reproducibility vs Replicability

- Replicability/Repeatability
  - Discrete technical results being reproducible, given the same initial conditions.

Reproducibility vs Replicability

- Replicability/Repeatability
  - Discrete technical results being reproducible, given the same initial conditions.

- Reproducibility
  - Reported performance gains carrying over to different contexts and implementations.

Reproducibility

- Publication of models, and code enable others to verify results.

Reproducibility

- Publication of models, and code enable others to verify results.
- Reproducibility increases confidence in the robustness of the method.

Reproducibility

- Publication of models, and code enable others to verify results.
- Reproducibility increases confidence in the robustness of the method.
- Incentivize reproducibility of reported results.
  - https://www.acm.org/publications/policies/artifact-review-badging
  - https://reproindex.com/event/reprosml2020
  - http://cknowledge.org/request.html
  - https://reproducibility-challenge.github.io/neurips2019/

Software Mechanisms

- Reproducibility
- Formal Verification
- Validation of ML by ML
- Practical Verification

Formal verification

- Use formal methods of mathematics to verify that the system satisfies some conditions.

Formal verification

- Use formal methods of mathematics to verify that the system satisfies some conditions.

- ML systems are generally not subjected to such rigor.

Formal verification

- Use formal methods of mathematics to verify that the system satisfies some conditions.
- ML systems are generally not subjected to such rigor.
- Techniques (for ML systems) are still in infancy.

Challenges to Formal verification

- Need to reconceive and redevelop traditional formal properties.

Challenges to Formal verification

- Need to reconceive and redevelop traditional formal properties.
- Difficulty of modelling ML systems as mathematical objects.

Challenges to Formal verification

- Need to reconceive and redevelop traditional formal properties.
- Difficulty of modelling ML systems as mathematical objects.
- The size of real-world ML models can be more than the limits that existing verification techniques can work with.

Software Mechanisms

Reproducibility

Formal Verification

Validation of ML by ML

Practical Verification

Validation of ML by ML Systems

- Alternative to formal verification - more practical but less robust.

Validation of ML by ML Systems

- Alternative to formal verification - more practical but less robust.
- An example
  - Adaptive Stress Testing (AST) uses RL to find the most likely failure of a system for a given scenario [1]
  - It is used to validate aircraft collision avoidance software [2].

[1]: Mark Koren, Anthony Corso, and Mykel Kochenderfer. "The Adaptive Stress Testing Formulation". In: RSS 2019: Workshop on Safe Autonomy. Freiburg, 2019. URL: https://openrev iew.net/pdf?id=rJgoNK-oaE.
[2] Ritchie Lee et al. "Adaptive stress testing of airborne collision avoidance systems". In: AIAA/IEEE Digital Avionics Systems Conference - Proceedings. Institute of Electrical and Electronics Engineers Inc., Oct. 2015. ISBN: 9781479989409. DOI: 10.1109/DASC.2015.7311613. URL: htt ps://ieeexplore.ieee.org/document/7311613/versions.

Software Mechanisms

Reproducibility

Formal Verification

Validation of ML by ML

Practical Verification

Practical Verification

- Use scientific protocols to characterize a model’s data, assumptions, and performance.

Practical Verification

- Use scientific protocols to characterize a model’s data, assumptions, and performance.
- Training data can be rigorously evaluated for representativeness

Practical Verification

- Use scientific protocols to characterize a model's data, assumptions, and performance.
- Training data can be rigorously evaluated for representativeness
- Assumptions can be characterized by clearly output uncertainties

Practical Verification

- Use scientific protocols to characterize a model's data, assumptions, and performance.
- Training data can be rigorously evaluated for representativeness
- Assumptions can be characterized by clearly output uncertainties
- Performance can be characterized by measuring generalization and performance heterogeneity across data subsets.

Software Mechanisms

Audit Trails

Interpretability

Privacy preserving ML

Software Mechanisms

Audit Trails

Interpretability

Privacy preserving ML

Audit Trails

- Traceable log of steps in system design, testing, and operation.

Audit Trails

- Traceable log of steps in system design, testing, and operation.
- Already used in numerous industries and safety-critical systems.

Audit Trails

- Traceable log of steps in system design, testing, and operation.
- Already used in numerous industries and safety-critical systems.
- Documenting audit trails can help make AI systems auditable.

Audit Trails

- Traceable log of steps in system design, testing, and operation.
- Already used in numerous industries and safety-critical systems.
- Documenting audit trails can help make AI systems auditable.
- For example, code changes, logs of training runs, all outputs of a model, etc.

Audit Trails

- Traceable log of steps in system design, testing, and operation.
- Already used in numerous industries and safety-critical systems.
- Documenting audit trails can help make AI systems auditable.
- For example, code changes, logs of training runs, all outputs of a model, etc.
- It could be useful if standards are defined for audit trails in AI.

Software Mechanisms

Audit Trails

Interpretability

Privacy preserving ML

Interpretability

- Difficult to verify the claims about AI systems if we can not interpret their output.

Interpretability

- Difficult to verify the claims about AI systems if we can not interpret their output.
- Moreover, interpretability is a multi-faceted term.

Interpretability

- Difficult to verify the claims about AI systems if we can not interpret their output.

- Moreover, interpretability is a multi-faceted term.

- Following directions could be useful for supporting verifiable claims:
  - Developing and establishing consensus on the criteria, objectives, and frameworks for interpretability research
  - Constraining models to be interpretable by default, instead of interpret a model post-hoc.

Software Mechanisms

Audit Trails

Interpretability

Privacy preserving ML

Privacy Preserving Machine Learning

- Aims to protect the privacy of data/models during training, evaluation and deployment.

Privacy Preserving Machine Learning

- Aims to protect the privacy of data/models during training, evaluation and deployment.

- Federated learning:
    - Many clients users collaboratively train a model without sharing data with each-other.

Privacy Preserving Machine Learning

- Aims to protect the privacy of data/models during training, evaluation and deployment.

- Federated learning:
  - Many clients users collaboratively train a model without sharing data with each-other.
  - Learning model could still memorize some data.

Privacy Preserving Machine Learning

- Aims to protect the privacy of data/models during training, evaluation and deployment.

- Federated learning:
    - Many clients users collaboratively train a model without sharing data with each-other.
    - Learning model could still memorize some data.
    - Can be mitigated using differential privacy techniques

Privacy Preserving Machine Learning

- Differential privacy
  - Add controlled amount of statistical noise to the dataset

Privacy Preserving Machine Learning

- Differential privacy
  - Add controlled amount of statistical noise to the dataset
  - Obscure contribution from individual data points while retraining the group patterns.

Privacy Preserving Machine Learning

- Differential privacy
    - Add controlled amount of statistical noise to the dataset
    - Obscure contribution from individual data points while retraining the group patterns.
    - Works well with federated learning

Privacy Preserving Machine Learning

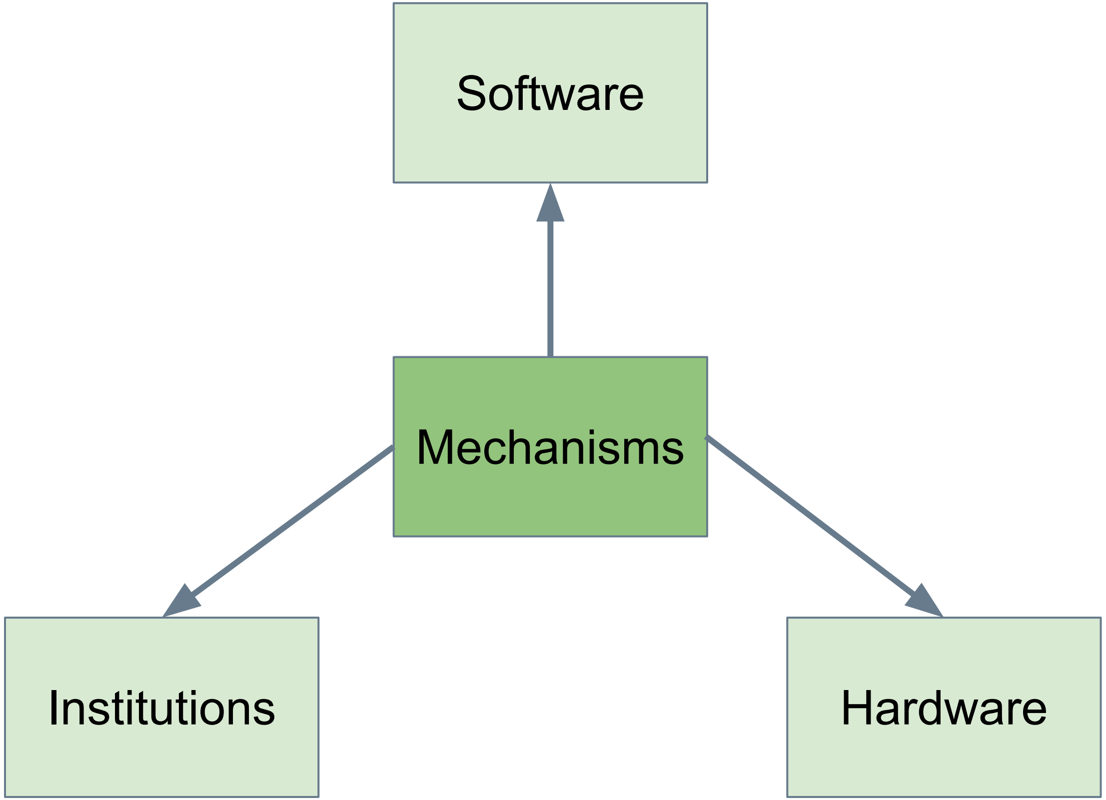
- Encrypted Computation
    - The model is trained and deployed on encrypted data

Privacy Preserving Machine Learning

- Encrypted Computation
  - The model is trained and deployed on encrypted data
  - Eg: homomorphic encryption, secure multi-party computation, and functional encryption

Privacy Preserving Machine Learning

- Encrypted Computation
  - The model is trained and deployed on encrypted data
  - Eg: homomorphic encryption, secure multi-party computation, and functional encryption
  - Such models can be securely shared.


Software
Mechanisms
Institutions
Hardware

Thank you

@shagunsodhani