Preamble

Fairness is a complex concept and deeply contextual. Keep the following points in mind:

- There is no single definition of fairness that will apply equally well to different applications of AI.
- Given the many complex sources of unfairness, it is not possible to fully “debias” a system or to guarantee fairness; the goal is to detect and to mitigate fairness-related harms as much as possible.
- Prioritizing fairness in AI systems often means making tradeoffs based on competing priorities. It is therefore important to be explicit and transparent about priorities and assumptions.
- There are seldom clear-cut answers. It is therefore important to document your processes and considerations (including priorities and tradeoffs), and to seek help when needed.
- Detecting and mitigating fairness-related harms requires continual attention and refinement.
- If you do not feel you can detect or mitigate fairness-related harms sufficiently, seek help.

Prioritizing fairness in AI systems is a *sociotechnical* challenge. AI systems can behave unfairly for a variety of reasons, some social, some technical, and some a combination of both social and technical.

- AI systems can behave unfairly because of societal biases reflected in the datasets used to trained them.
- AI systems can behave unfairly because of societal biases that are either explicitly or implicitly reflected in the decisions made by teams during the AI development and deployment lifecycle.
- AI systems can possess characteristics that, while not necessarily reflective of societal biases, can still result in unfair behavior when these systems interact with particular stakeholders after deployment.

AI systems can cause a variety of fairness-related harms, including harms involving people’s individual experiences with AI systems or the ways that AI systems represent the groups to which they belong.

- AI systems can unfairly allocate opportunities, resources, or information.
- AI systems can fail to provide the same quality of service to some people as they do to others.
- AI systems can reinforce existing societal stereotypes.
- AI systems can denigrate people by being actively derogatory or offensive.
- AI systems can over- or underrepresent groups of people, or even treat them as if they don’t exist.

These types of harm are not mutually exclusive; a single AI system can exhibit more than one type.

Fairness-related harms can have varying severities. However, the cumulative impact of even comparatively “non-severe” harms can be extremely burdensome or make people feel singled out or undervalued.

Identifying who is at risk of experiencing fairness-related harms involves considering both the people who will use the system and the people who will be directly or indirectly affected by the system, either by choice or not. Although fairness is often discussed with respect to groups of people who are protected by anti-discrimination laws, such as groups defined in terms of race, gender, age, or disability status, the most relevant groups are often context-specific. Moreover, such groups may be difficult to identify. It can therefore be useful to consider the system’s purpose and expected deployment contexts; different stakeholders, including the people who are responsible for, will use, or will be affected by the system, as well as the different demographic groups represented by these stakeholders; and any relevant standards, regulations, guidelines, or policies. Finally, people often belong to overlapping groups—different combinations of race, gender, and age, for example—and specific intersectional groups may be at greatest risk of experiencing fairness-related harms and at risk of experiencing different types of harm. Considering each group separately from the others may obscure these harms.

For more information about this checklist, please see M. Madaio, L. Stark, J. W. Vaughan, and H. Wallach. 2020. *Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI.* In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI 2020).

AI Fairness Checklist

The items in this checklist are intended to be used as a starting point for teams to customize. Not all items will be applicable to all AI systems, and teams will likely need to add, revise, or remove, items to better fit their specific circumstances. Undertaking the items in this checklist will not guarantee fairness. The items are intended to prompt discussion and reflection. Most items can be undertaken in multiple different ways and to varying degrees.

Envision

Consider doing the following items in moments like:

- Envisioning meetings
- Pre-mortem screenings
- Product greenlighting meetings

1.1 Envision system and scrutinize system vision

1.1.a Envision system and its role in society, considering:

- System purpose, including key objectives and intended uses or applications
  - Consider whether the system should exist and, if so, whether the system should use AI
- Sensitive, premature, dual, or adversarial uses or applications
  - Consider whether the system will impact human rights
  - Consider whether these uses or applications should be prohibited
- Expected deployment contexts (e.g., geographic regions, time periods)
- Expected stakeholders (e.g., people who will make decisions about system adoption, people who will use the system, people who will be directly or indirectly affected by the system, society), including demographic groups (e.g., by race, gender, age, disability status, skin tone, and their intersections)
- Expected benefits for each stakeholder group, including demographic groups
- Relevant regulations, standards, guidelines, policies, etc.

1.1.b Scrutinize resulting system vision for potential fairness-related harms to stakeholder groups, considering:

- Types of harm (e.g., allocation, quality of service, stereotyping, denigration, over- or underrepresentation)
- Tradeoffs between expected benefits and potential harms for different stakeholder groups
  - Consider who the system will give power to and who it will take power from
  - Consider which expected benefits you are willing to sacrifice to mitigate potential harms

1.1.c Revise system vision to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

1.2 Solicit input and concerns on system vision

1.2.a Solicit input on system vision and potential fairness-related harms from diverse perspectives, including:

- Members of stakeholder groups, including demographic groups
  - Consider whether any stakeholder groups would prefer that the system not exist or not be deployed in all contexts, what alternatives they would prefer, and why
- Domain or subject-matter experts
- Team members and other employees

1.2.b Revise system vision to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

1.3 Escalate potential harms involving sensitive, premature, dual, or adversarial uses or applications to leadership

Define

Consider doing the following items in moments like:

- Spec reviews
- Game plan reviews
- Design reviews

For more information about this checklist, please see M. Madaio, L. Stark, J. W. Vaughan, and H. Wallach. 2020. *Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI.* In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI 2020).

2.1 Define and scrutinize system architecture

2.1.a Define system architecture, considering:

- Machine learning models, including their structures, relationships, and interactions
- Objective functions and training algorithms
- Performance metrics (e.g., accuracy, user satisfaction, relevance)
- Functionality for stakeholder feedback (e.g., comments or concerns, third-party audits)
- Functionality for rollback or shutdown in the event of unanticipated fairness-related harms
- Functionality for preventing any prohibited uses or applications
- User interfaces or user experiences
- Other hardware, software, or infrastructure
- Assumptions made when operationalizing system vision via system architecture
    - Consider whether these assumptions are sufficiently well justified

2.1.b Scrutinize resulting definitions for potential fairness-related harms to stakeholder groups, considering:

- Types of harm (e.g., allocation, quality of service, stereotyping, denigration, over- or underrepresentation)
- Tradeoffs between expected benefits and potential harms for different stakeholder groups

2.1.c Revise system architecture definitions to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

2.2 Define and scrutinize datasets

2.2.a Define datasets needed to develop and test the system, considering:

- Desired quantities and characteristics, considering:
    - Relevant stakeholder groups, including demographic groups
        - Consider oversampling smaller stakeholder groups, but be aware of overburdening
    - Expected deployment contexts
- Potential sources of data
    - Consider reviewing all datasets from third-party vendors
- Collection, aggregation, or curation processes, including:
    - Procedures for obtaining meaningful consent from data subjects
    - People involved in collection, aggregation, or curation, including demographic groups
        - Consider whether people involved might introduce societal biases
    - Incentives for data subjects and people involved in collection, aggregation, or curation
        - Consider whether data subjects might feel undue pressure to provide data
    - Software, hardware, or infrastructure involved in collection, aggregation, or curation
- Relevant regulations, standards, guidelines, policies, etc.
- Assumptions made when operationalizing system vision via datasets
    - Consider whether these assumptions are sufficiently well justified

Scrutinize resulting definitions for potential fairness-related harms to stakeholder groups, considering:

Types of harm (e.g., allocation, quality of service, stereotyping, denigration, over- or underrepresentation)

- Tradeoffs between expected benefits and potential harms for different groups

2.2.b Revise dataset definitions to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

2.3 Define and scrutinize fairness criteria

2.3.a Based on potential fairness-related harms identified so far, define fairness criteria, considering:

- How criteria will be assessed (e.g., fairness metrics and benchmark dataset, system walkthroughs with diverse stakeholders or personas) at each subsequent stage of the lifecycle, including
    - People involved in assessment (e.g., judges), including demographic groups
        - Consider whether people involved might introduce societal biases
    - Datasets needed to assess fairness criteria
- Acceptable (levels of) deviation from fairness criteria
- Potential adversarial threats or attacks to fairness criteria (e.g., "brigading")
- Assumptions made when operationalizing system vision via fairness criteria

For more information about this checklist, please see M. Madaio, L. Stark, J. W. Vaughan, and H. Wallach. 2020. *Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI.* In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI 2020).

- Consider whether these assumptions are sufficiently well justified

2.3.b Scrutinize fairness criteria definitions for potential fairness-related harms that may not be covered

2.3.c Revise fairness criteria definitions to cover any not-covered potential harms; if this is not possible, document why, along with contingency plans, etc., and consider aborting development

2.4 Solicit input and concerns on system architecture, dataset, and fairness criteria definitions

2.4.a Solicit input on definitions and potential fairness-related harms from diverse perspectives, including:

- Members of stakeholder groups, including demographic groups
- Domain or subject-matter experts
- Team members and other employees

2.4.b Revise definitions to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

Prototype

Consider doing the following items in moments like:

- Go / no-go discussions
- Code reviews

3.1 Prototype (and scrutinize) datasets

3.1.a Prototype datasets according to dataset definitions; if datasets deviate from definitions during development, revisit checklist items from "define" stage

3.1.b Document dataset characteristics and limitations (e.g., by creating datasheets), considering:

- Potential audiences for documentation, including:
  - Members of stakeholder groups
  - Team members and other employees
  - Regulators and other third parties

3.2 Prototype (and scrutinize) system

3.2.a Prototype system according to system architecture definitions; if system architecture deviates from definitions during development, revisit checklist items from "define" stage

3.2.b Document system characteristics and limitations (e.g., by creating model cards for the models that comprise the system or a transparency note or factsheet for the system itself), considering:

- Potential audiences for documentation, including:
  - Members of stakeholder groups
  - Team members and other employees
  - Regulators and other third parties

3.3 Assess fairness criteria

3.3.a Assess fairness criteria according to fairness criteria definitions, considering:

- Acceptable (levels of) deviation from fairness criteria
- Tradeoffs between different fairness criteria
- Tradeoffs between performance metrics and fairness criteria
- Discrepancies between development environment and expected deployment contexts

3.3.b If system prototype fails to satisfy fairness criteria, revise system accordingly; if this is not possible, document why, along with contingency plans, etc., and consider aborting development

3.4 Undertake user testing

3.4.a Undertake user testing with diverse stakeholders, analyzing results broken down by relevant stakeholder groups. This should be done even if the system satisfies the fairness criteria because the system may exhibit unanticipated fairness-related harms not covered by the fairness criteria. Consider conducting:

- Online experiments
- Ring testing or dogfooding
- Field trials or pilots in deployment contexts

3.4.b Revise production system to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

For more information about this checklist, please see M. Madaio, L. Stark, J. W. Vaughan, and H. Wallach. 2020. *Co-Designing Checklists to Understand Organizational Challenges and Opportunities around Fairness in AI.* In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI 2020).

3.5 Solicit input and concerns on system prototype

3.5.a Solicit input on system prototype from diverse perspectives, including:

- Members of stakeholder groups, including demographic groups
- Domain or subject-matter experts
- Team members and other employees

3.5.b Revise system prototype to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

Build

Consider doing the following items in moments like:

- Go / no-go discussions
- Code reviews
- Ship reviews
- Ship rooms

4.1 Build (and scrutinize) production datasets

4.1.1 Build production datasets according to dataset definitions; if datasets deviate from definitions during development, revisit checklist items from "define" stage

4.1.2 Update dataset documentation

4.2 Build (and scrutinize) production system

4.2.1 Build production system according to system architecture definitions; if system architecture deviates from definitions during development, revisit checklist items from "define" stage

4.2.2 Update system documentation

4.3 Assess fairness criteria

4.3.1 Assess fairness criteria according to fairness criteria definitions, considering

- Acceptable (levels of) deviation from fairness criteria
- Tradeoffs between different fairness criteria
- Tradeoffs between performance metrics and fairness criteria
- Discrepancies between development environment and expected deployment contexts

4.3.2 If production system fails to satisfy fairness criteria, revise system accordingly; if this is not possible, document why, along with contingency plans, etc., and consider aborting development

4.4 Undertake user testing

4.4.1 Undertake user testing with diverse stakeholders, analyzing results broken down by relevant stakeholder groups. This should be done even if the system satisfies the fairness criteria because the system may exhibit unanticipated fairness-related harms not covered by the fairness criteria. Consider conducting:

- Online experiments
- Ring testing or dogfooding
- Field trials or pilots in deployment contexts

4.4.2 Revise production system to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

4.5 Solicit input and concerns on production system

4.5.1 Solicit input on production system from diverse perspectives, including:

- Members of stakeholder groups, including demographic groups
- Domain or subject-matter experts
- Team members and other employees

4.5.2 Revise production system to mitigate any potential harms; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting development

Launch

Consider doing the following items in moments like:

- Ship review before launch
- Code reviews

5.1 Participate in public benchmarks

5.1.a Participate in public benchmarks so that stakeholders can contextualize system performance, considering:

- Competitors' responsible AI principles and development practices
- Alternatives to public benchmarks if relevant public benchmarks don't exist (e.g., distributing and publicizing private benchmark datasets for use by competitors or third parties)

5.1.b Revise system to mitigate any harms revealed by benchmarks; if this is not possible, document why, along with future mitigation or contingency plans, etc., and consider aborting deployment

5.2 Enable functionality for stakeholder feedback

5.2.a Establish processes for responding to or escalating stakeholder feedback, including:

- Stakeholder comments or concerns
    - Consider establishing processes for redress
- Third-party audits

5.3 Enable functionality for rollback or shutdown in the event of unanticipated fairness-related harms

5.3.a Establish processes for deciding when to roll back or shut down

5.4 Enable functionality to prevent prohibited uses or applications

5.4.a Establish processes for deciding whether unanticipated uses or applications should be prohibited

Evolve

Consider doing the following items in moments like:

- Regular product review meetings
- Code reviews

6.1 Monitor deployment contexts

6.1.a Monitor deployment contexts for deviation from expectations, including:

- Unanticipated stakeholder groups, including demographic groups
- Adversarial threats or attacks

6.1.b Revise system (including datasets) to match actual deployment contexts; if this is not possible, document why, along with expected impacts on stakeholders, and consider rollback or shutdown

6.2 Monitor fairness criteria

6.2.a Monitor fairness criteria for deviation from expectations, including:

- Adversarial threats or attacks

6.2.b If system fails to satisfy fairness criteria, revise system accordingly; if this is not possible, document why, along with expected impacts on stakeholders, and consider rollback or shutdown

6.3 Monitor stakeholder feedback

6.3.a Follow processes for responding to or escalating stakeholder feedback

6.3.b Revise system to mitigate any harms revealed by stakeholder feedback; if this is not possible, document why, update system documentation, and consider rollback or shutdown

6.4 Revise system at regular intervals to capture changes in societal norms and expectations

6.4.a Revisit checklist items from previous stages