Abstract

Artificial intelligence and machine learning (AI/ML) are being used in an increasing number of applications. Anyone with basic coding knowledge can download datasets or models from code-sharing sites like GitHub and use them in their own projects. However, reusing AI/ML components out of their original contexts can raise a whole range of ethical issues especially when the original creators have failed to critically assess the provenance, performance, and proper uses of their own work. Our ability to maintain effective documentation has not kept up with the recent explosion of AI/ML. This project looks at a number of existing proposals for what ethics documentation should look like. While these proposals offer a robust set of standards for what questions data scientists should be asking, they pay much less attention to how to get more people to adopt documentation in their projects. EthicsHub is a Google Chrome browser extension that modifies the GitHub user interface allow users to document and discuss potential ethical issues. It features an interactive documentation editor that allows users to select from a number of interoperable templates based on existing proposals, customize them for domain-specific needs or workflows, and commit documentation files directly to their repositories in a number of file formats. The hope is that this extension demonstrates the need for software platforms to make space for ethical concerns that is on par with existing open source standards, such as README files, contributing guidelines, and license agreements.

Thomas Zembowicz
Yale University
Department of Computer Science
Professor Elisa Celis, Advisor
Professor Nisheeth Vishnoi, Faculty Supervisor

github.com/tzembo/ethics-hub
1 Introduction

Artificial intelligence and machine learning (AI/ML) are no longer exclusively the domain of researchers in universities or large technology companies. Over the past decade, a plethora of AI/ML libraries, frameworks, websites, and competitions have *democratized* these fields in the sense that anyone with basic coding skills can leverage AI/ML for their own projects or businesses. The internet abounds with datasets and trained models, ready to be downloaded and incorporated as part of other systems. However, our ability to create and maintain effective documentation has not kept pace with many of these trends—even as we’re increasingly aware of the ethical issues that AI/ML systems can raise when they deal with human-centered data or become involved in important social applications such as education, health care, employment, or sentencing.

This lack of effective documentation becomes obvious when browsing GitHub, a popular web-based hosting service for managing code with version control. GitHub has more than 39 million users as of April 2019 according to its user search tool. The website is home to the world’s largest collection of source code, including many projects that have to do with artificial intelligence and machine learning. In addition to hosting source code, GitHub offers features for distributed collaboration such as pull requests, issues management, and project wikis. GitHub also encourages repository maintainers to include certain kinds of documentation that makes it easier for the public to understand and contribute to their projects. These include README files explaining how to approach the repository, contributing guidelines with rules for how to get involved, open source licenses governing how the project may be reused, and issue templates providing a common structure for collaboration and communication. However, there is a lack of tools (whether created by GitHub or by third parties contributing to GitHub’s large application ecosystem) that give ethical questions the same real estate in the GitHub user interface as other community concerns. GitHub is therefore home to a staggering amount of datasets and models lacking information about their provenance, performance, and proper use.
At the same time, there’s a growing understanding that artificial intelligence and machine learning pose many ethical risks that must be carefully considered and discussed, especially when datasets or models are shared outside of the contexts in which they were created. As this report will describe, data science projects should require clear documentation in order to prevent improper uses, adverse performance effects that propagate through a system, and privacy threats to individuals. These problems are difficult enough to manage within companies and other organizations that participate in the entire data pipeline, from collecting and cleaning to the final publication. They’re exacerbated when people without specialized training can access publicly hosted data or models and use them as part of their own projects.

EthicsHub is an effort to synthesize existing work promoting standards for documenting potential ethical issues. At the same time, this project tries to imagine ways that some of these proposals can be thoughtfully embedded in the workflows of software engineers and data scientists in ways that promote their use. I propose a Google Chrome extension that modifies GitHub’s user interface to provide dedicated space for creating ethics documentation files and engaging in conversations about the ethical ramifications of AI/ML projects. The hope is that this extension pushes platforms like GitHub toward incorporating ethical thinking into the tools that data scientists and programmers use in their daily work.

2 Motivations

Artificial intelligence and machine learning raise many ethical issues, especially when they involve human-centered data and are embedded in social systems like hiring or health care that effect people in their most vulnerable states. These ethical issues are too numerous to cover here, and have been extensively documented in both scientific literature and the media. Instead, I want to focus on the special challenges presented when AI/ML projects become publicly accessible on sites like GitHub without proper context. Documentation should exist
not only to ensure that project participants are thinking about the ethical ramifications of their work, but also to give the broader public information about how a project should be interpreted or reused in order to avoid some of the pitfalls that I’ve sketched out in three broad categories below.

2.1 Proper Use

As Gitelman explains, we tend to see "raw data" as naturally-occurring when they are really instead created, organized, and used in specific contexts [8]. This context is often missing when models or data become publicly available online without effective documentation describing their provenance and original purpose, creating the potential for misuse. For example, a recent study by researchers from Boston University and Microsoft Research showed that word embeddings trained on Google News articles encoded biases that associated men with fields such as programming while women were more associated with homemaking, which can raise serious concerns when used in contexts such as employment [2]. Datasets or models made for one purpose like studying trends in online news coverage may cause preventable adverse effects when used in a very different context. General-purpose machine learning platforms also create the potential for misuse. Google released its internal platform for accelerating artificial intelligence development called TensorFlow in 2018, hoping to make AI/ML research more accessible. However, one of the biggest results of this move has been the proliferation of deepfakes, or realistic-looking face swap videos that can insert the faces of politicians or celebrities into pornographic videos, created using tools over which Google has lost control [7]. Clear documentation that discloses intended uses and speculates about uses that should not be permitted might not stop all bad actors from abusing powerful machine learning tools, but can help cut down on unintentional misuse.
2.2 Performance

The fact that a model is made available on GitHub does not mean that it performs well, nor that it does not make decisions based on unfair criteria. As machine learning models get incorporated in a growing number of sensitive applications, like employment, health care, and education, it’s important that developers are aware of these biases before they propagate adverse effects in larger systems. For example, one recent study of datasets used to test the performance of commercial gender classification systems showed that they significantly overrepresent lighter skin tones, and that these classification systems therefore consequently perform far worse for darker-skinned females, which had error rates of up to 34% on a balanced testing dataset [3]. Another study of recidivism prediction instruments such as COMPAS used to assess the likelihood that a defendant will re-offend in the future showed that black defendants were much more likely to be incorrectly judged to pose a higher risk than white defendants, a result of a failure to satisfy the error rate balance criterion [4].

There already exist a number of performance measures that capture how trained models may perform across a range of sensitive groups, and it’s important that individuals have access to these measures when considering whether they want to use existing work as part of their own projects.

2.3 Privacy

Making datasets and models publicly available raises questions about how machine learning projects might encroach on the privacy and other rights of human beings. This includes whether people implicated in human-centered projects gave informed consent at various points during the project, from the data collection stage to the final publication in academic journals or on public platforms such as GitHub. Although personally identifiable information such as names, social security numbers, or addresses are usually removed from datasets, it’s often surprisingly easy to reconstruct some of that information through other means. For example, researchers at the University of Texas at Austin were able to uncover the identity
of certain users in a Netflix movie rankings dataset released as part of a machine learning
competition by cross-referencing rankings and timestamps with publicly available data from
the Internet Movie Database, or IMDb [11]. In 2006, AOL released anonymized search logs
for a large number of its users to the general public for research purposes that nonetheless
revealed personally identifiable information in some of the search queries [1]. When data
science increasingly occurs in the public arena, it’s important that we pay special attention
to the rights of the individuals whose data is used in AI/ML systems, and documentation
can better describe the steps taken to respect individuals and their privacy.

3 Existing Projects

In light of these ethical challenges, researchers and organizations are paying more attention
to how more robust documentation can lead to more reliable and socially responsible AI/ML
systems. The questions that they think documentation accompanying AI/ML projects should
ask illustrate a concerted effort to grapple with some of the problems described in the previous
section. However, these proposals all struggle more with the question of how documentation
should fit into existing workloads and what form documentation should take to be useful.
3.1 Datasheets for Datasets

Datasheets for Datasets is a proposal from researchers at Google, Microsoft Research, Georgia Tech, and the University of Maryland that calls for documents to accompany datasets inspired by industry practices in other high-stakes fields, like electronics [6]. Individual electronic components come with datasheets describing their operating characteristics, benchmark testing results, and recommended uses among other important information. Components of machine learning systems, the authors urge, should come with similar information so that a faulty component does not propagate biases throughout an AI/ML system. The proposal lists several categories of questions that this kind of datasheet should include: Motivation, Composition, Collection Process, Preprocessing, Uses, Distribution, and Maintenance. Researchers developed this robust set of questions was developed with feedback from major...
stakeholders in academia, technology firms, and government. However, these categories and the questions that fall under them are not meant to be exhaustive, and the authors welcome modifications that make them better suited to domain-specific needs or existing organizational infrastructure.

While the questions posed in this proposal would succeed in helping users avoid the ethical pitfalls described in the previous section, the paper doesn’t pay as much attention to the critical question of how to get more people to document their datasets. The authors don’t offer many suggestions about the design of these datasheets or what tools data scientists should use to create them. The sample datasheet included in the paper, while extremely thorough, is ten pages long. If these proposed datasheets are to be used by all sorts of AI/ML projects, big or small, there have to be carefully designed tools that manage some of the informational complexity and encourage users to answer as many questions as they can without becoming overwhelmed by the scope of the task.
3.2 Model Cards for Model Reporting

Model Card - Smiling Detection in Images

Model Details
- Developed by researchers at Google and the University of Toronto, 2018, v1.
- Convolutional Neural Net.
- Pretrained for face recognition then fine-tuned with cross-entropy loss for binary
  smiling classification.

Intended Use
- Intended to be used for fun applications, such as creating cartoon smiles on real
  images; augmentative applications, such as providing details for people who are
  blind; or assisting applications such as automatically finding smiling photos.
- Particularly intended for younger audiences.
- Not suitable for emotion detection or determining affect; smiles were annotated
  based on physical appearance, and not underlying emotions.

Factors
- Based on known problems with computer vision face technology, potential rele-
  vant factors include groups for gender, age, race, and Fitzpatrick skin type;
  hardware factors of camera type and lens type; and environmental factors of
  lighting and humidity.
- Evaluation factors are gender and age group, as annotated in the publicly available
dataset CelebA [36]. Further possible factors not currently available in a public
smiling dataset. Gender and age determined by third-party annotators based
on visual presentation, following a set of examples of male/female gender and
young/old age. Further details available in [36].

Metrics
- Evaluation metrics include False Positive Rate and False Negative Rate to
  measure disproportionate model performance errors across subgroups. False

Figure 2: Model Cards for Model Reporting

Model cards is a framework from researchers at Google and the University of Toronto for clarifying the intended use cases of trained machine learning models [10]. They suggest that models should be accompanied by short documents that evaluate their performance in a number of conditions, including across sensitive categories such as gender, race, or skin type, in addition to supplying more information about the context in which they were created. This proposal is very similar to the datasheet concept, except that it’s applied to trained models. The paper proposes the following set of modules: Model Details, Intended Use, Factors, Metrics, Evaluation Data, Training Data, Quantitative Analyses, Ethical Considerations, and Caveats and Recommendations. While this project explicitly focuses on human-centered models for computer vision or natural language processing tasks, the authors note that many of these features could be applied to a larger categories of models.
Although researchers focused on developing questions that would help describe a model’s intended uses and performance characteristics in order to mitigate some of the issues identified earlier, the proposal once again offers little guidance about how these model cards should be designed for readability and ease-of-use. The prototype included in the paper is divided into modules just like Datasheets for Datasets. However, the information under each module lacks any headings or other labels that describe what specific concerns a module is trying to address. While the sample model cards also include charts of quantitative analysis that help illustrate key performance characteristics across sensitive groups, the proposal does not suggest any tools or graphical conventions for creating these figures. There is an overall lack of structure or consistency in informational organization.

### 3.3 Data Nutrition Project

![Provenance](image)

<table>
<thead>
<tr>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>U.S. Centers for Medicare &amp; Medicaid Services</td>
</tr>
<tr>
<td>UTL</td>
<td><a href="https://www.cms.gov/OpenPayments/">https://www.cms.gov/OpenPayments/</a></td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:openpayments@cms.hhs.gov">openpayments@cms.hhs.gov</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Author</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Propublica</td>
</tr>
<tr>
<td>UTL</td>
<td><a href="https://www.propublica.org/datastore/">https://www.propublica.org/datastore/</a></td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:data.store@propublica.org">data.store@propublica.org</a></td>
</tr>
</tbody>
</table>

**Variables**

<table>
<thead>
<tr>
<th>Id</th>
<th>A unique ID number for this payment &amp; product combination. This is assigned by ProPublica for internal use</th>
</tr>
</thead>
</table>

| Applicable_manufacturer | ID of the applicable manufacturer or, if applicable, one of the following OCP codes |

Figure 3: Dataset Nutrition Label

The Data Nutrition Project is an initiative to create tools and practices for responsible AI development, spearheaded by researchers and technologists at MIT’s Media Lab and Harvards
Berkman Klein Center for Internet and Society [9]. Their proposed Dataset Nutrition Label is based on the FDA-mandated nutrition labels for food and beverage products. Like other proposals, the labels feature a modular design for easier navigation and customizability, with the following sections enabled by default: Metadata, Provenance, Variables, Statistics, Pair Plot, Probabilistic Model, and Ground Truth Correlations. The latter sections feature interactive tables graphics that the proposed label editor automatically generates with the help of a backend. This makes the Nutrition Labels much more focused on quantitative characteristics of the dataset than Datasheets, which pay more attention to the normative questions surrounding a dataset’s creation, use, and publications.

The Dataset Nutrition Labels take design and readability seriously, concerns that are looked over in other documentation standard proposals. While the skeumorphic design meant to mimic FDA nutrition labels is a little heavy-handed, it succeeds in creating a familiar metaphor for users who may not be aware that they should be documenting their projects. Since these labels are HTML documents rendered by a special , they enable click-based navigation and interactive graphics that the word processor-produced Datasheets and Model Cards can’t match. However, many of these interactive features and auto-generated graphics rely on a special-purpose backend that carries out statistical analyses as well as generates and hosts the Nutrition Labels. This centralized architecture raises a new concern. If Nutrition Labels take off in the AI/ML should it be the responsibility of one party maintain the infrastructure that supports documentation?
3.4 Deon: An Ethics Checklist for Data Scientists

Data Science Ethics Checklist

A. Data Collection

☐ A.1 Informed consent: If there are human subjects, have they given informed consent, where subjects affirmatively opt-in and have a clear understanding of the data uses to which they consent?

☐ A.2 Collection bias: Have we considered sources of bias that could be introduced during data collection and survey design and taken steps to mitigate those?

☐ A.3 Limit PII exposure: Have we considered ways to minimize exposure of personally identifiable information (PII) for example through anonymization or not collecting information that isn’t relevant for analysis?

B. Data Storage

☐ B.1 Data security: Do we have a plan to protect and secure data (e.g., encryption at rest and in transit, access controls on internal users and third parties, access logs, and up-to-date software)?

☐ B.2 Right to be forgotten: Do we have a mechanism through which an individual can request their personal information be removed?

Figure 4: Deon

Deon is an open source command line tool that allows data scientists to add an ethics checklist to their projects [5]. Deon works on slightly different level than the previous proposals, offering checklists that assess ethical concerns for the entire project and not just individual components. Deon offers a default checklist that allows changes through community consensus in the Issues panel of Deon’s GitHub repository. The checklist currently categorizes questions into five sections that trace the stages of any AI/ML project: Data Collection, Data Storage, Analysis, Modeling, and Deployment. Users may also modify the checklist on their own by specifying a YAML file with the same structure and labels as the default checklist file.

The command line tool is simple to use and creates checklists that have a consistent informational organization. Deon allows users to append the checklist to existing files (such as R code) or create new standalone files in a number of formats (including Markdown and HTML). However, Deon checklists offer no space for users to provide information beyond toggling a checkbox, which makes them easier to fill out but prevents them from providing meaningful context to users discovering a checklist on a site like GitHub. It would be useful...
to allow users to describe how their project meets the criteria for each checked question, or why unchecked questions might not be applicable to the project at hand.

4 EthicsHub: Functionality

This project builds on the standards being established by the projects in the previous section, but tackles a question that they fail to address: how to actually get people to create documentation for their projects. EthicsHub is a Google Chrome extension that modifies GitHub’s user interface to provide easy-to-use and familiar tools allowing repository maintainers to create ethics documentation files and engage in discussions about the ethical aspects of their work and how it should or should not be used. Even though only a small set of all AI/ML projects end up in public GitHub repositories, this project specifically focuses on GitHub for a number of reasons. First of all, GitHub is one of the top destinations for people looking for interesting projects on the internet. GitHub also has an active open source community that demands certain standards from one another, like README files in every root directory to orient new users. Finally, GitHub already has a robust ecosystem of applications, plugins, and APIs that allow developers to modify the user interface and create additional features. An application specifically addressing ethical issues would fill a hole in the existing slate of offerings and bring much-needed visibility to ethics documentation.

EthicsHub is designed to unobtrusively fit in right alongside the GitHub features that repository maintainers already use in their everyday workflows. When a user installs EthicsHub, the extension adds an Ethics tab to GitHub’s navigation bar for the main repository view. This would be familiar to anyone who has used ZenHub, a popular task management GitHub application that adds a ZenHub tab to the same navigation bar. When a user clicks on Ethics, the EthicsHub dashboard loads below the navigation bar. The application mimics the appearance of other repository tools down to the style of buttons and text fields. This dashboard is launching pad for users to create their own documentation files.
EthicsHub's documentation features are built with flexibility and interoperability in mind. Upon creating a new documentation file, users can select a template from a variety of templates based on current research and community projects, including the Deon Default Checklist, Datasheets for Datasets, and Model Cards for Model Reporting. All of these templates are created with the schema that Deon uses for its checklists in order to promote interoperability between documentation standards. The template browser presents these options with a short description and a list of the modules they come with by default. Users may instead choose to import their existing Deon checklists, or any template that follows the same YAML schema outlined below. The application also allows users to start from scratch using a completely blank template, adding modules and questions as they see fit.

The documentation editor itself gives users many ways to edit and ultimately commit documentation to users' repositories. A column on the left side lists modules, reflecting the modular design present in many proposed frameworks. Users may rename modules by clicking the edit button, reorder modules by dragging and dropping them, or create a module by clicking the button at the bottom of the list. Selecting a module loads only that module's questions in the area on the right, minimizing the amount of information overload at any given time. As with the modules themselves, users may edit, reorder, and create individual questions. Each question features a toggle button that allows them to be enabled or disabled, depending on whether the question is relevant to this particular project. This reflects the reality that not all projects will need to address the same questions. When the question...
is enabled, the text box below each question asks how that question has been addressed. When the question is disabled, the text box asks the user to explain why the question is not applicable in this instance. Clicking on the "Edit" button allows users to edit the fields on each question card and provide more information about their work.

Users have two options for exporting their documentation. Clicking the "Export template" button at the top of the page brings up a modal with that allows users to copy a YAML template representing their documentation to their clipboards. This template strips questions of their answers and toggled status, which allows users to reuse them in other projects or make them publicly available. Clicking the "Generate documentation" button, on the other hand, allows users to stitch together the documentation into a number of file types (including Markdown) and provide a commit message and optional extended description. The changes are then committed directly to the user’s repository using GitHub’s APIs.

5 EthicsHub: Implementation

EthicsHub is packaged as an extension for the popular Google Chrome web browser, which commanded a 62% market share as of April 2019. Chrome extensions are small software modules that share a common structure, which includes a manifest file specifying certain background scripts and content scripts to run during its operation. Chrome extensions can draw on a robust API that provides access to current windows and tabs, local storage, events, and other features of the web browser. Developers may publish their packaged extensions to the Chrome Web Store, where users can install them with one click. Alternatively, users may directly load extensions from a local folder.

The bulk of the code for EthicsHub is written in TypeScript, which is a strict superset of JavaScript that adds static typing features typically found in languages like C or Java. TypeScript code compiles to normal JavaScript, which is necessary for Chrome extensions. While TypeScript adds upfront complexity both in terms of setting up a build system and
defining variable types, it forces JavaScript developers to write clearer code and helps with debugging by flagging errors at compile time. The project uses a web application bundler called Parcel that handles TypeScript compilation, bundles assets, and generates a dist folder that can be loaded into the Chrome web browser as an extension.

The background script background.ts contains code that handles OAuth authentication and calls to the GitHub API. It communicates with the content script using message passing enabled by the chrome.extension API, where messages are simple JSON objects. GitHub encourages applications to authenticate using OAuth, an authorization protocol that allows secure access to an application through the use of an authorization token that has information about duration and access scope. The EthicsHub user first declares their intent to gain access, which brings them to a page where they can enter their GitHub username and password and grant EthicsHub access to their data with the necessary scopes before being redirected back to EthicsHub with a code. The user then exchanges this code for an GitHub authorization token that the EthicsHub application stores in Chrome’s local storage so that EthicsHub can make several requests within one session.

The content script main.ts is the part of the extension that modifies the GitHub user interface and renders the EthicsHub application. Its instructed in the manifest file to run only on GitHub websites. The script manipulates the front-end HTML code to insert the Ethics button into the repository navigation bar, and listens to MutationObserver events to ensure that the button is added each time the GitHub server re-renders the UI. The script also listens to changes in the browser’s current location in order to determine if the browser is at a location where the extension should render the EthicsHub application below the repository navigation bar.

The EthicsHub dashboard and editor tools are built using React, a popular front-end framework originally created by Facebook that allows developers to create dynamic and modular web applications much more easily. React applications are split into modular components. The content script renders the main component below the repository navigation
bar when the EthicsHub tab is selected, which determines what other components to render using the `HashRouter` component of the popular `react-router` package based on the path that follows the `#` symbol of the current URL. The documentation editor draws heavily on React’s stateful components, which allows user interface blocks corresponding to modules or questions to trigger events and respond to changes in the underlying data.

Two helper files in `src/lib` help with importing and exporting documentation templates. The first, `parser.ts`, defines the schema that underlies each template and allows easy conversion between JavaScript objects and YAML strings. The second, `formats.ts`, contains classes that take a JavaScript object and generate documentation files in the specified format.

## 6 Conclusion

EthicsHub has plenty of room to grow. There’s an extensive software wish list that remains at the end of this semester, which will become part of the project README in the GitHub repository. One top priority is persisting data (whether in local storage, GitHub repositories, or through a backend service) in order to allow users to save their work before generating documentation files. The other top priority is implementing the Discussion side of EthicsHub to encourage users to add a standardized "ethics" issue label to their projects, as well provide a portal for asking and responding to questions about ethical issues raised by published work. A lot of work remains to make the code more robust, which includes fixing various front-end bugs and introducing more error handling for GitHub API requests, before this is a piece of software that belongs in the GitHub Marketplace, Chrome Web Store, and other distribution points. I welcome collaborators on GitHub to help make EthicsHub ready for more public use.

While this project focused on GitHub for its massive size, active open source community, and robust application ecosystem, there are many other websites that host data and models for AI/ML projects. These include sites that are very similar to GitHub (such as BitBucket,
GitLab, and SourceForge) as well as data science competitions (like Kaggle), all of which host many datasets and trained models and deserve similar attention. Publishing the results of AI/ML projects also comes at the end of what is often a long process involving data collection, pre-processing, and analysis. Documentation should be part of every stage of this process, so it’s worth considering how documentation tools can be better integrated at earlier stages.

The hope is that many of these features eventually make it into the native GitHub user interface and other tools that are key parts of AI/ML workflows and the broader open source community. The design of these tools and platforms play an important and under-recognized role in shaping standards and norms for collaboration. This is evident by how GitHub provides tools for creating README files, contributing guidelines, codes of conduct, license agreements, issue templates, and a number of other kinds of documentation essential to a well-functioning open source community. If developers and data scientists are going to take ethical challenges seriously as artificial intelligence and machine learning become more embedded in socio-technical systems, ethical documentation must have the same real estate reserved in the platforms we use to publish and share AI/ML projects.
References


