

## E Dataset-related supplementary material

### E.1 Code and data release

**Access to the code** We attach the code at the time of submission as part of the supplementary material. The code with further updates to the benchmark and the environment will be released, under MIT license, at <https://github.com/ethz-spylab/agentgym>. The only exceptions for the license are explicitly marked portions of code that come from previous work and are licensed under a different license.

### E.2 Hosting, licensing, and maintenance plan

- Hosting plan: the code is hosted and easily accessible on GitHub, and the gym environment will be installable via the `pip install agentgym` command.
- Licensing plan: We are not planning to change the license.
- Maintenance plan: the authors are committed to fix potentially existing bugs in the benchmark’s code, and to update the benchmark content as models, and prompt injection attacks and defenses evolve in time.

### E.3 Reproducibility

We release with the code on GitHub all models outputs and conversations as JSON files, and a Jupyter Notebook that can be use to reproduce all figures and tables in the paper. We cannot include the model outputs as part of the supplementary material because of space constraints, but they can be found on Google Drive ([https://drive.google.com/file/d/16nhDqSTRVac\\_GbcC3-9WMjVaJZfmcwL0/view?usp=share\\_link](https://drive.google.com/file/d/16nhDqSTRVac_GbcC3-9WMjVaJZfmcwL0/view?usp=share_link)). In order to use the data to run the notebook, the file in the Google Drive folder should be unzipped in the “runs” directory in the attached code.

Further, in the README file of the code, we also provide extensive documentation on how to use our framework (including how to run the existing benchmark, create new tools, task, etc.) and we additionally include some Jupyter Notebooks that show how to use our framework. We also include a `requirements.txt` file that can be used to install the exact dependencies we used for the experimental results in the paper.

### E.4 Code and data license

MIT License

Copyright (c) 2024 Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## E.5 Statement of responsibility

The authors confirm that they bear all responsibility in case of violation of rights and confirm that the data is released under MIT license unless otherwise stated in some portions of the code.

## E.6 DOI and Croissant metadata

**DOI** We will create a DOI with Zenodo once we will have made the code public.

**Croissant metadata link** As the set of tasks, tools, and environment data we create to pre-populate the benchmark is a mix of data and code, it is not possible to generate Croissant [1] metadata for it.

## F Data card

We report information about the dataset following the guidelines of Pushkarna, Zaldivar, and Kjartansson [43].

### F.1 Summary

- Dataset name: AgentGym
- Dataset link: <https://github.com/ethz-spylab/agentgym>
- Dacard author: Edoardo Debenedetti, ETH Zurich

### F.2 Authorship

#### F.2.1 Publishers

- Publishing organizations: ETH Zurich, Invariant Labs
- Industry types: Academic - Tech, Corporate - Tech
- Contact details:
  - Publishing POC: Edoardo Debenedetti
  - Affiliation: ETH Zurich
  - Contact: [edoardo.debenedetti@inf.ethz.ch](mailto:edoardo.debenedetti@inf.ethz.ch)

#### F.2.2 Dataset Owners

- Contact details:
  - Dataset Owner: Edoardo Debenedetti
  - Affiliation: ETH Zurich
  - Contact: [edoardo.debenedetti@inf.ethz.ch](mailto:edoardo.debenedetti@inf.ethz.ch)
- Authors:
  - Edoardo Debenedetti, ETH Zurich
  - Jie Zhang, ETH Zurich
  - Mislav Balunovic, ETH Zurich and Invariant Labs
  - Luca Beurer-Kellner, ETH Zurich and Invariant Labs
  - Marc Fischer, ETH Zurich and Invariant Labs
  - Florian Tramèr, ETH Zurich

#### F.2.3 Funding Sources

No institution provided explicit funding for the creation of this benchmark. However, Edoardo Debenedetti is supported by armasuisse Science and Technology with a CYD Fellowship.

### F.3 Dataset overview

- Data subjects: Synthetically generated data, Data about places and objects
- Dataset snapshot:
  - Total samples: 124 tasks, 70 tools
  - Total environment data size: 136 KB
- Content description: The dataset comprises of a set of tasks that a user could potentially delegate to a tool-calling LLM, a set of tools that such LLM could employ, and a pre-populated state that the LLM can access.

#### F.3.1 Sensitivity of data

- Fields with sensitive data:
  - Intentionally Collected Sensitive Data: None
  - Unintentionally Collected Sensitive Data: None
- Risk types: No known risks

#### F.3.2 Dataset version and maintenance

- Maintenance status: Regularly updated
- Version details:
  - Current version: v1.0
  - Last updated: 06/2024
  - Release date: 06/2024
- Maintenance plan:
  - Versioning: We will use semantic versioning. The addition of new tasks that are in-distribution with the existing tasks will constitute a minor release. We will consider a new dataset with more tasks that are either very different or significantly more difficult than the previous versions to be a major release, hence with an increase in the first number of the version.
  - Updates: We plan to update the dataset as models capabilities improve and the current set of tasks becomes too easy. We further plan to include tasks that add more layers of indirection, e.g., so that the models can't know what tools they need in advance. We also consider adding multi-modal tasks in the future.
  - Errors: we consider errors tasks that can be solved by the model without seeing the prompt injection, tasks are actually not solvable by the model, ground truths and utility checks that are incorrect, tools that are wrongly and/or inappropriately documented.
- Next planned updates: We don't have a timeline yet.
- Expected changes: N/A

### F.4 Example of data points

- Primary data modality: Text Data (prompts and code)
- Sampling of data points: we show some example prompts for user and injection tasks in Table 1.
- Data fields:
  - Tasks: User/adversary prompt (what the user and the adversary want the agent to do), utility/security check functions (code that checks that if the task was correctly executed), ground truth functions (a sequence of function calls that solves the corresponding task)
  - Environment data: We have data from fake calendar, inbox, bank account, restaurants, hotels, car rental companies, Slack workspace, web, cloud drive.
  - Tools: the tools contain a description of the tool and the arguments needed by it, and the code that runs the tool itself.

## **F.5 Motivations and intentions**

### **F.5.1 Motivations**

- Purpose: Research
- Domains of application: Machine Learning, Large Language Models, Agents
- Motivating factors: studying the utility and robustness of tool-calling agents against prompt injection attacks, studying prompt injection attacks and defenses.

### **F.5.2 Intended use**

- Dataset use: Safe for research use
- Suitable use cases: testing the robustness and utility of tool-calling agents against prompt injection attacks, testing the effectiveness of prompt injection attacks and defenses.
- Unsuitable use cases: using this benchmark to evaluate the robustness of agents and defenses by using only the default attacks, without employing an adaptive attack with a thorough security evaluation.
- Citation guidelines: TBD upon acceptance.

## **F.6 Access, retention, & wipeout**

### **F.6.1 Access**

- Access type: External – Open Access
- Documentation link: <https://github.com/ethz-spylab/agentgym>
- Pre-requisites: None
- Policy links: None
- Access Control Lists: None

## **F.7 Provenance**

### **F.7.1 Collection**

- Methods used:
  - Artificially Generated
  - Authors creativity
- Methodology detail:
  - Source: Authors, GPT-4o, Claude Opus
  - Is this source considered sensitive or high-risk? No
  - Dates of Collection: 05/2024
  - Primary modality of collection data: Text Data
  - Update Frequency for collected data: static
  - Additional Links for this collection: <https://chatgpt.com/share/42362e6c-7c37-44d5-8a31-d3c767f70464> (example chat used to generate the data for the Cloud Drive. Other environment data has been generated in the same way).
  - Source descriptions: We used GPT-4o and Claude Opus to generate the data that the models obtain when the models use the tools. We provided the models with the schema that the data should follow, and a few examples. The tasks and the some of the dummy data were created by the authors with their creativity.
  - Collection cadence: Static.
  - Data processing: We manually inspected the LLM-generated data to check for correctness and consistency. We also manually changed some of the data to provide more realistic tasks.

### **F.7.2 Collection criteria**

We included and selected the LLM-generated data that were syntactically correct (the generation should be YAML format), and that were consistent with each other (e.g., calendar events that had realistic invitees, or emails that have reasonable subjects and bodies).

### **F.8 Human and Other Sensitive Attributes**

There are no human or other sensitive attributes.

### **F.9 Extended use**

#### **F.9.1 Use with Other Data**

- Safety level: safe to use with other data
- Known safe/unsafe datasets or data types: N/A

#### **F.9.2 Forking and sampling**

- Safety level: Safe to fork. Sampling not recommended as the dataset is not particularly large in the first place.
- Acceptable sampling methods: N/A

#### **F.9.3 Use in AI and ML systems**

- Dataset use: Validation
- Usage guidelines: the benchmark can be used to assess the quality of models and defenses, as long as the users make their best effort to effectively attack their model and/or defense with a strong adaptive attack.
- Known correlations: N/A

### **F.10 Transformations**

#### **F.10.1 Synopsis**

- Transformations applied: Cleaning Mismatched Values, Fixing YAML syntax errors, manually adding samples that are needed for the user and injection tasks, manual changes to fields such as dates to ensure consistency across the data.
- Fields transformed: the LLM-generated data.
- Libraries and methods used: manual changes.

#### **F.11 Validation types**

- Methods: Data Type Validation, Consistency Validation
- Descriptions: we define a schema for all environment data, and validate all the LLM- and manually-generated data against the schema. Moreover, we ensure that the ground truths and utility/security checks in all user and injection tasks are consistent with each other. We do so by running the ground truth and checking that it successfully passes the utility/security checks.

#### **F.12 Known applications and benchmarks**

- ML Applications: tool-calling agents
- Evaluation results and processes: we show the evaluation results and methodology in the main paper, in Section 4.