A Neural Network Subgrid Model of the Early Stages of Planet Formation

Thomas Pfeil[*,1,2] **Miles Cranmer**[3,4] **Shirley Ho**[3,4,5,6] **Philip J. Armitage**[4,7]
Tilman Birnstiel[1,8] **Hubert Klahr**[2]

[*] tpfeil@usm.lmu.de
[1] University Observatory, Ludwig-Maximilians-Universität München, Munich, Germany
[2] Max Planck Institute for Astronomy, Heidelberg, Germany
[3] Department for Astrophysical Sciences, Princeton University, Princeton, USA
[4] Center for Computational Astrophysics, Flatiron Institute, New York, USA
[5] Center for Cosmology and Particle Physics, New York University, New York, USA
[6] Department of Physics, Carnegie Mellon University, Pittsburgh, USA
[7] Department of Physics and Astronomy, Stony Brook University, Stony Brook, USA
[8] Exzellenzcluster ORIGINS, Boltzmannstr. 2, D-85748 Garching, Germany

Abstract

Planet formation is a multi-scale process in which the coagulation of $\mu\mathrm{m}$-sized dust grains in protoplanetary disks is strongly influenced by the hydrodynamic processes on scales of astronomical units ($\approx 1.5 \times 10^8\,\mathrm{km}$). Studies are therefore dependent on subgrid models to emulate the micro physics of dust coagulation on top of a large scale hydrodynamic simulation. Numerical simulations which include the relevant physical effects are complex and computationally expensive. Here, we present a fast and accurate learned effective model for dust coagulation, trained on data from high resolution numerical coagulation simulations. Our model captures details of the dust coagulation process that were so far not tractable with other dust coagulation prescriptions with similar computational efficiency.

1 Introduction to Dust Coagulation - The First Stage of Planet Formation

After the formation of a protostar, remaining material of its parent molecular cloud core forms a so-called protoplanetary disk around it. About $1\,\%$ of the mass of this disk consists of solids in the form of initially $\mu\mathrm{m}$-sized carbonaceous silicate grains and ices. All solid objects, including the rocky planets, the rocky cores of gas giant planets, comets, and asteroids form out of this material. Subsequent collisions between the grains are caused by gas turbulence and differential aerodynamic drag and lead to the formation of larger aggregates via sticking due to van der Waals forces. Since relative velocities between the grains increase with their sizes, growth is halted at some point, when collisions become too violent for sticking and instead lead to fragmentation (break-up). At this so-called fragmentation barrier, an equilibrium size distribution is reached. Its form is determined by the interior composition of the grains and their size-dependent relative velocities.

Theoretically, these processes are described by the Smoluchowski equation [1]—an integro-differential equation that gives the mass exchange rates between grains on a continuous spectrum of sizes. Only a few analytically solvable cases exist, which is why most numerical models of dust coagulation rely on solution techniques for the discretized Smoluchowski equation, which is derived by exchanging the continuum of grain sizes by a discreet grid of sizes. Solving the resulting system of ODEs is an elaborate numerically task that requires the size grid to have >100 bins to lead to meaningful results [2, 3].

Machine Learning and the Physical Sciences workshop, NeurIPS 2022.

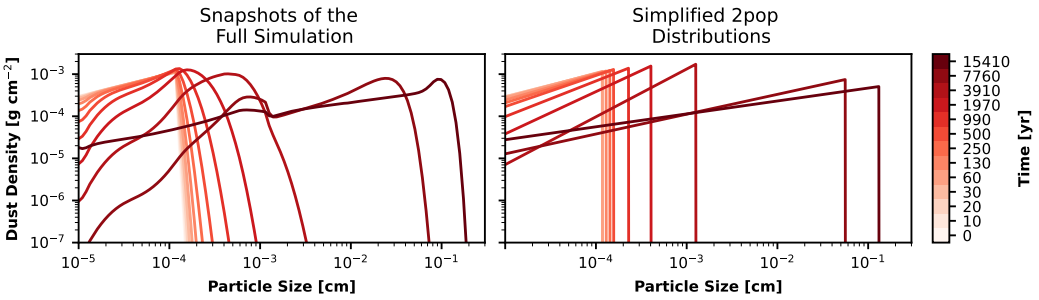

Figure 1: Output of a numerical simulation of dust coagulation in a protoplanetary disk (left side). Initially µm-sized grains grow until they reach the fragmentation barrier. On the right-hand side we show the equivalent power law size distributions derived from the actual simulation results on the left. The simplified time series data is the training data for our machine learning model.

An example simulation is shown in the left hand side of Figure 1. The model is initialized with a distribution of µm-sized grains. Collisions first lead to an almost exponential growth phase, which, in this case is halted by fragmentation after $\sim 10^4\,\mathrm{yr}$. The result is a top-heavy equilibrium distribution of up to $2\,\mathrm{mm}$-sized grains.

These multi-bin models are applicable to 0D [local; see 2] or 1D [vertically and azimuthally averaged; see 4] disk models, but due to their high numerical cost, can not be applied in 3D models of protoplanetary disks.

1.1 A Power Law Prescription for Dust Coagulation and the Need for a Machine Learning Approach

We aim to develop an approach in which the dust size distribution is described by a truncated power law, instead of a discretized distribution with hundreds of size bins. Our goal is to make the modeling of dust coagulation on top of large scale hydrodynamic simulations more feasible. For a given total dust column density σ_{tot}, and a minimum particle size $a_{\mathrm{min}} = 10^{-5}\,\mathrm{cm}$, this simplified distribution can be described by only two parameters:

$$\begin{aligned} a_{\mathrm{max}}: &\quad \text{The size of the largest particles (truncation size of the power law)} \\ \sigma_1: &\quad \text{The column density of particles larger than } a_{\mathrm{int}} = \sqrt{a_{\mathrm{max}} a_{\mathrm{min}}}. \end{aligned}$$

It can be shown that the exponent of the power law size distribution $\sigma(a) \propto a^{p+4}$ is then given by $p = \frac{\log(\sigma_1/\sigma_0)}{\log(a_{\mathrm{max}}/a_{\mathrm{int}})} - 4$, where $\sigma_0 = \sigma_{\mathrm{tot}} - \sigma_1$ is the column density of particles smaller than a_{int}. In contrast to other approximate models like `two-pop-py` [5], this approach makes it possible to retain information about the overall shape of the size distribution. It is, however, not trivial to find a mathematical description for the time evolution of the power law distribution without making strongly simplifying assumptions. We therefore propose a machine learning aided power law model, which predicts the time evolution of the simplified distribution.

2 Method

For our method, we trained a Multilayer Perceptron (MLP) on the evolution of power law grain size distributions derived from detailed multi-bin simulations of dust coagulation. The general workflow of our model is laid out in Figure 2. The inputs of our neural network are the size distribution parameters, and the parameters of the protoplanetary disk environment, like gas temperature, gas density, etc. The model's output are the respective time derivatives $\partial_t a_{\mathrm{max}}$ and $\partial_t \sigma_1$, which are then used as source terms in a numerical integration scheme.
Our simple neural network model therefore makes it possible to simulate the temporal evolution of the physical system, similar to other machine learning approaches explored in recent years [6, 7].
Our MLP consists of 3 hidden layers, each with 100 nodes, 14 nodes in the input layer, and two nodes in the output layer. The layers are fully connected with ReLU activation functions.

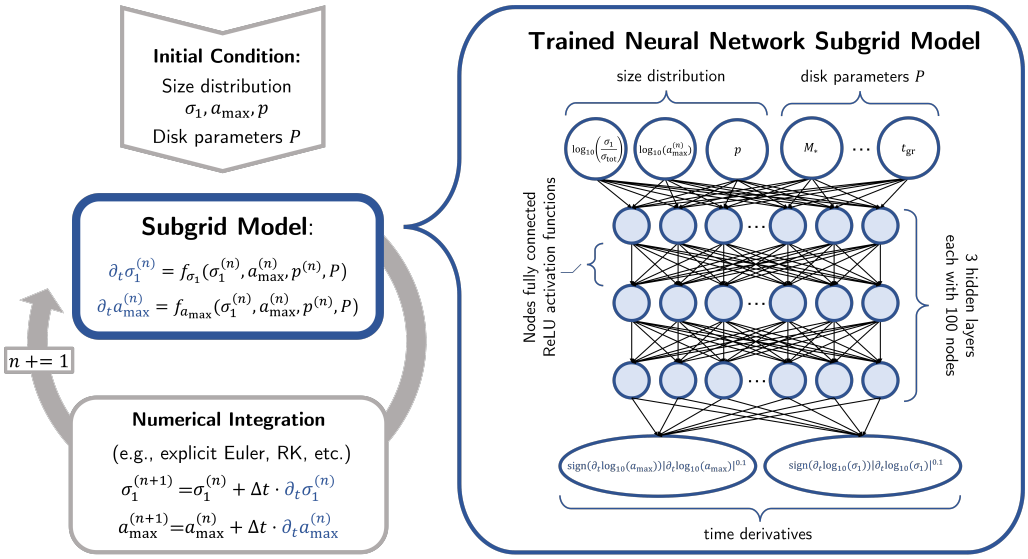

Figure 2: General outline of the trained machine learning subgrid model of dust coagulation. An artificial neural network is trained to predict the time derivatives of the size distribution's power law representation. The resulting source term is used to evolve the distribution in time.

2.1 Training Data Generation

We create our training data using the `COALA` dust coagulation routine, which was provided by Til Birnstiel and Sebastian Stammler, and which was already used in a hydrodynamic simulation [8]. `COALA` is a local dust coagulation code, written in FORTRAN that numerically solves the Smoluchowski equation on a mass grid (in our case with 171 bins). 10000 dust coagulation simulations have been created, each with 150 time outputs. The dust distributions are evolved over a time corresponding to 50 dust growth time scales, or maximally 10^6 yr to ensure that an equilibrium is reached at the end of each simulation. The initial conditions are chosen randomly from a parameter space that represents the known typical conditions within protoplanetary disks from simulations and observations.

2.2 Training Data Pre-Processing

As a first step, we derive the two parameters of the power law size distributions from the full size distributions with 171 size bins. We define $a_{\max}$ as the particle size for which $\int_{a_{\min}}^{a_{\max}} \sigma(a)\,\mathrm{d}a/\sigma_{\mathrm{tot}} = 0.99$ holds, i.e. $99\,\%$ of the total mass of the particles has sizes smaller than $a_{\max}$. σ_1 is then derived by summing up the mass of all bins with sizes larger than $a_{\mathrm{int}} = \sqrt{a_{\max} a_{\min}}$. This results in 10000×150 time series data points for both quantities, from which we derive the respective time derivatives. For training, we scale the data to a range from 0 to 1 and divide the dataset into 8000 training data simulations and 2000 test data simulation. We found that even small deviations from the actual equilibrium states can lead to large errors after time integration with the predicted gradients. Our experiments have shown that the best training results are achieved if we use the tenth root of the time derivatives, multiplied by their sign as the training data. In that way, also small scale features around the equilibrium states ($\partial_t = 0$) can be learned, leading to the best results during numerical integration and to the correct equilibrium distribution.

2.3 Training Procedure

We train our neural network model within the Pytorch Lightning framework [9, 10], using the Adam optimization algorithm [11]. The batch size is set to 1000, we apply a learning rate of 3×10^{-4}, and train the model for 1000 epochs. We employ the Mean Absolute Percentage Error [MAPE, 12] as a loss function, which also penalized deviations of small absolute value. To avoid division by

zero when applying the loss function, we offset the normalized training data by +0.1. Training was conducted on a single Nvidia A100-40GB GPU.

3 Results

After training we evaluate the resulting model by using the predicted time derivatives for numerical time integration of the setups from the training data set (see our method in Figure 2 and `https://github.com/ThomasPfeil/2popML`). For the tests performed in this work, we utilize an explicit Euler scheme, as shown in Figure 2. We limit the time step to ensure numerical stability for the given source terms as

$$\Delta t = C \cdot \min\left(\left| \frac{a_{\max}}{\partial_t a_{\max}} \right|, \left| \frac{\sigma_1}{\partial_t \sigma_1} \right| \right), \tag{1}$$

with $C = 0.1$. In Figure 3, we present an example simulation from the test dataset. The average deviation from the actual time series is about $\sim 4\,\%$. We have conducted this procedure with all 2000 parameter combinations from the test data set. On average, one full integration run takes $\approx 73\,\mathrm{ms}$ wall clock time, compared to $791\,\mathrm{ms}$ for the full numerical model on the same machine. 11 integrations failed, reaching either negative dust densities or errors larger than $1000\,\%$, resulting in a $99.45\,\%$ success rate.

For the 1989 successfully finished test simulations, we plot the distribution of the mean relative deviation of each time series to the respective actual time series in Figure 4. On average, the deviation between the integration series conducted with the model prediction and the actual data is $\sim 4\,\%$ for the maximum particle size, and $\sim 0.5\,\%$ for the column density of large particles.

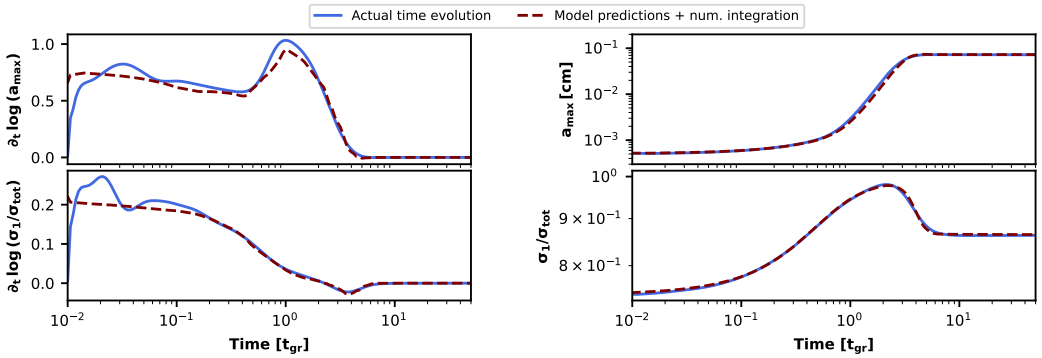

Figure 3: Result of a numerical integration with the neural network predictions for the respective time derivatives.

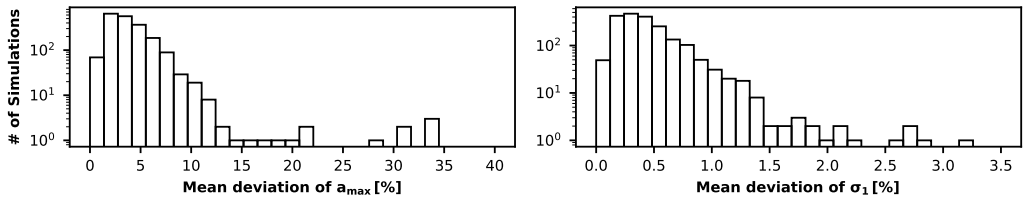

Figure 4: Distribution of deviations from the actual simulation time series for both simulated parameters $a_{\max}$ and σ_1.

4 Conclusions and Outlook

Our results strongly suggest that numerical efforts to study the early phases of planet formation can benefit from the use of machine learning techniques. Our neural network model was capable of predicting gradients with high enough precision to allow for time integration of the vast majority

of the test data set ($99.45\,\%$ of the simulations). Our model could therefore be used as a fast and accurate alternative to commonly used full coagulation simulations. Due to its much shorter runtime, it could, for the first time, make large scale hydrodynamic simulations of protoplanetary disks with dust coagulation feasible.

Since our model is trained on simulation data with various parameter combinations, we expect it to produce accurate results as long as the applied model parameters lie within the ranges used for training. This means, the most important limitation of our model lies in the range of applicable stellar parameters and disk parameters, e.g., stellar mass (varies from 0.01 to $1.4\,\mathrm{M}_\odot$), distances to the central star (varied from 0.1 to $100\,\mathrm{au}$), etc.

Further testing is needed for the use of our model in disks with substructure, e.g., disks with planetary gaps and pressure bumps. It is not clear if our model will produce reliable outputs in these environments, since it was trained on parameter combinations derived from simple power law disks (without substructure).

Testing this requires an implementation of our neural network model as a subgrid model into a hydrodynamics code to simulate gas and dust dynamics in protoplanetary disks. We therefore aim to couple our model to the `PLUTO` code [13]. Once a stable run is achieved, we can test our subgrid model in an evolving environment and under the conditions in substructures.

The structural similarity of our approach (Figure 2) to semi-analytic physical models could also make it possible to interpret the trained neural network in the future and derive insights into the underlying physics, which could make our results interpretable [14–16].

Acknowledgments

T.P. expresses his gratitude to the Simons Foundation for the opportunity to conduct this project as part of the 2022 Flatiron Machine Learning X Science Summer School. Special thanks goes to the summer school mentors S.H., M.C., and P.A. for their advise and many helpful discussions. T.P., H.K., and T.B. acknowledge the support of the German Science Foundation (DFG) priority program SPP 1992 "Exploring the Diversity of Extrasolar Planets" under grant Nos. BI 1816/7-2 and KL 1469/16-1/2. T.B. acknowledges funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 714769 and funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under grants 361140270, 325594231, and Germany's Excellence Strategy - EXC-2094 - 390783311. All computations were conducted on the VERA cluster of the Max Planck Institute for Astronomy, Heidelberg.

References

[1] M. V. Smoluchowski. Drei Vortrage uber Diffusion, Brownsche Bewegung und Koagulation von Kolloidteilchen. *Zeitschrift fur Physik*, 17:557–585, January 1916.

[2] F. Brauer, C. P. Dullemond, and Th. Henning. Coagulation, fragmentation and radial motion of solid particles in protoplanetary disks. *A&A*, 480(3):859–877, March 2008. doi: 10.1051/0004-6361:20077759.

[3] T. Birnstiel, C. P. Dullemond, and F. Brauer. Dust retention in protoplanetary disks. *A&A*, 503 (1):L5–L8, August 2009. doi: 10.1051/0004-6361/200912452.

[4] Sebastian M. Stammler and Tilman Birnstiel. DustPy: A Python Package for Dust Evolution in Protoplanetary Disks. *ApJ*, 935(1):35, August 2022. doi: 10.3847/1538-4357/ac7d58.

[5] T. Birnstiel, H. Klahr, and B. Ercolano. A simple model for the evolution of the dust population in protoplanetary disks. *A&A*, 539:A148, March 2012. doi: 10.1051/0004-6361/201118136.

[6] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to Simulate Complex Physics with Graph Networks. *arXiv e-prints*, art. arXiv:2002.09405, February 2020.

[7] Patrick Kidger. On Neural Differential Equations. *arXiv e-prints*, art. arXiv:2202.02435, February 2022.

[8] Joanna Drążkowska, Shengtai Li, Til Birnstiel, Sebastian M. Stammler, and Hui Li. Including Dust Coagulation in Hydrodynamic Models of Protoplanetary Disks: Dust Evolution in the

Vicinity of a Jupiter-mass Planet. *ApJ*, 885(1):91, November 2019. doi: 10.3847/1538-4357/ab46b7.

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *arXiv e-prints*, art. arXiv:1912.01703, December 2019.

[10] W.A. Falcon. PyTorch Lightning. *Github, https://github.com/Lightning-AI/lightning*, March 2019. doi: 10.5281/zenodo.3828935.

[11] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, art. arXiv:1412.6980, December 2014.

[12] Arnaud De Myttenaere, Boris Golden, Bénédicte Le Grand, and Fabrice Rossi. Mean Absolute Percentage Error for regression models. *arXiv e-prints*, art. arXiv:1605.02541, May 2016.

[13] A. Mignone, G. Bodo, S. Massaglia, T. Matsakos, O. Tesileanu, C. Zanni, and A. Ferrari. PLUTO: A Numerical Code for Computational Astrophysics. *ApJs*, 170:228–242, May 2007. doi: 10.1086/513316.

[14] Miles Cranmer, Alvaro Sanchez-Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering Symbolic Models from Deep Learning with Inductive Biases. *arXiv e-prints*, art. arXiv:2006.11287, June 2020.

[15] Dmitrii Kochkov, Jamie A. Smith, Ayya Alieva, Qing Wang, Michael P. Brenner, and Stephan Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118, 2021. doi: 10.1073/pnas.2101784118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2101784118.

[16] Kimberly Stachenfeld, Drummond B. Fielding, Dmitrii Kochkov, Miles Cranmer, Tobias Pfaff, Jonathan Godwin, Can Cui, Shirley Ho, Peter Battaglia, and Alvaro Sanchez-Gonzalez. Learned Coarse Models for Efficient Turbulence Simulation. *arXiv e-prints*, art. arXiv:2112.15275, December 2021.

Impact Statement

Our work could be of benefit for future numerical studies of protoplanetary disks and planetesimal formation. To the current date, only one two-dimensional hydrodynamic simulation, including a full dust coagulation model, exists, which was published [8]. The authors also highlight the extensive computational cost, which only allowed them to run a single simulation. Our model reduces the computational cost of these kinds of models significantly. Firstly, it allows to model a dust size distribution with only two additional dust fluids, compared to the 171 dust fluids in full coagulation models. This means the time to compute dust transport is already reduced by a factor of ~ 100. Secondly, the computing time for the coagulation step is significantly reduced. From our results, discussed in section 3, we expect a speed up of a factor of ~ 10. Therefore, larger simulation runs and parameter studies could be conducted, benefiting our understanding of the dynamics of gas and dust in protoplanetary disks, the interpretation of observational data, and our understanding of the formation of planetesimals—the building blocks of planets.

We can not identify any ethical implications of our work.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes]
- Did you include the license to the code and datasets? [Yes]
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]