Opportunities and Challenges of Model Multiplicity in Interactive Software Systems

Kris Luyten[0000−0002−4194−1101], Gilles Eerlings, Jori Liesenborgs[0000−0002−3648−8031], Gustavo Rovelo Ruiz[0000−0001−7580−8950], Sebe Vanbrabant[0009−0001−7996−6048], and Davy Vanacken[0000−0001−8436−5119]

Hasselt University - Flanders Make, Expertise Centre for Digital Media
firstname.lastname@uhasselt.be

Abstract. The proliferation of artificial intelligence (AI) in interactive systems has led to significant challenges in model integration, but also end-user-related aspects such as over- and undertrust. This paper presents an initial exploration on how multiple AI models with the same performance and behavior but different internal workings–a phenomenon called model multiplicity–affect system integration and user interaction. We discuss the implications of model multiplicity for transparency, trust, and operational effectiveness in interactive software systems.

1 Introduction

Integrating artificial intelligence (AI) models into interactive software systems is increasingly common in sectors ranging from healthcare to customer service. In most cases, there is one single AI model being used in an interactive system, which requires careful design to make it accessible and usable by its end users. However, relying on a single model might quickly lead to a miscalibration in trust, as there is a limited basis for comparison and often a low degree of transparency due to the lack of explanations [16]. Trust and transparency are, however, important aspects in human-AI collaboration [4, 8, 9]. We envision an evolution toward a '*many simultaneous expert advisors*' approach, where multiple AI models are used simultaneously by an interactive system to offer users more balanced and substantiated advice. This might mean using multiple black box and white box models at the same time. In our work, we will first focus on closely related and very similar models, introduced in section 2.

The *many simultaneous expert advisors* approach is especially interesting for critical decision support systems, such as those used in medical diagnostics, financial forecasting, and urban planning. In the healthcare sector, for example, using multiple AI models can enable a more comprehensive analysis of patient data by cross-verifying diagnoses to reduce the risk of error. This not only mitigates the risks associated with over-reliance on a single model, but also provides a platform for continuous learning and improvement of AI systems. Discrepancies in the outputs of these multiple models can be studied to refine the algorithms and enhance their accuracy and reliability, and reveal noteworthy patterns in the

training dataset. Furthermore, our strategy promotes transparency and trust among users by demonstrating that decisions are well-considered and vetted through multiple expert advisors (being the multiple models).

This paper discusses the 'many experts' approach as the 'model multiplicity' phenomenon and its relation to the Rashomon effect. We provide an initial framework using a model multiplicity approach and discuss how model multiplicity can be integrated in interactive systems. Lastly, we address some challenges in integrating multiple AI models into a single system.

2 Model Multiplicity

There are often multiple variations of AI models, trained for a very similar or even the same purpose, that achieve a similar performance during the validation. This phenomenon, also known as *model multiplicity* (MM) [5], can be of great value to correct *over- and undertrust* in an AI system. Having a limited understanding of AI can be frustrating, causing users to lose trust [3, 13]. More trust, however, is not always better, since users might trust the system even when it is not behaving as intended [19]. Our aim is to move away from a single AI model, thus one single output, and evolve toward multiple AI models that provide multiple possible outputs and the context on which their output is based.

Integrating model multiplicity into interactive systems not only facilitates enhanced reliability and more nuanced interactions with AI, but also significantly increases the complexity of engineering those interactive systems. The model multiplicity introduces a need for careful orchestration to ensure that the advice from various models is integrated in a coherent, transparent and user-friendly manner. Engineers must design systems capable of handling potentially contradictory advice from different models, deciding which advice to prioritize or how to merge the advice effectively. This process involves developing sophisticated decision-making algorithms or voting systems that can assess and integrate diverse outputs transparently, and the user interface must be designed to effectively communicate the rationale behind not just one but a set of AI-driven decisions.

Furthermore, using several AI models simultaneously can lead to a substantial overhead in terms of resources and computational power. This might affect the system's performance, and thus its responsiveness toward the user, but also its scalability, maintenance, and carbon footprint. Therefore, while model multiplicity brings substantial benefits to the robustness and trustworthiness of AI, it also introduces significant engineering challenges that must be addressed to ensure a successful and sustainable implementation and deployment of such systems. For a sustainable and responsible usage of these models, we need to define metrics (e.g. Pareto optimality [15, Section 3.3]) to decide the benefit of using more models versus the increase in accuracy and trustworthiness.

3 The Rashomon effect and Occam's Razor for AI

Model multiplicity [5, 12] refers to the existence of several models that can provide equivalent outputs with similar accuracy, but do so based on different internal mechanisms. This phenomenon can result from different training processes, initialization parameters, or architectural choices. The concept is derived from the 'Rashomon effect' [14], where multiple explanations exist for the same phenomenon, each equally valid. The Rashomon effect signifies that within a set of functions, there exists a multitude of different elements capable of producing the same minimum error rate [6]. This means that when you have a collection of functions fitted on the same dataset, a certain amount of functions will have the same accuracy while having different parameters. The set of functions exhibiting this behavior is often referred to as the 'Rashomon set' [10].

While model multiplicity offers interesting opportunities to enhance robustness through diversity, the outputs from multiple models only benefit the user if they are integrated into interactive systems in a coherent, transparent and user-friendly manner. One possible approach to cope with multiple models is to apply Occam's razor principle, often interpreted as favoring simpler solutions. This aligns with the benefit that simplicity in AI is often equated with interpretability [6]. For instance, models employing inherently interpretable algorithms, such as decision trees or linear models, are considered simple white-box approaches, whereas (deep) neural networks are considered complex black-box approaches. From a user perspective, when both a decision tree and a neural network achieve the same accuracy, the decision tree should be favored due to its simplicity and interpretability.

Occam's razor principle falls short, however, when multiple models have the same complexity. Consequently, when multiple models exist in a Rashomon set, there is no single 'best' model if only accuracy is being accounted for. This poses a significant challenge regarding the integration of model multiplicity, as exploring the outputs of all members of a Rashomon set can be very daunting, even for experts [18]. Exploring the outputs of multiple AI models is thus a challenge in itself. Only a few visualization tools, such as TimberTrek [17], facilitate the exploration of multiple models in an end-user-accessible manner. Such tools provide an accessible dashboard to explore multiple model results, and their rich visualizations are mostly used as stand-alone systems, with their main purpose being the deep exploration of results through visual explanations. Other approaches use more traditional ways of presenting multiple models with respect to each other, e.g., using graph plots and bar charts presenting the metrics of the models side by side [11]. Moreover, most of these initiatives rely on tree-based AI approaches, which are much more accessible to explain than black-box models. Since white-box models are typically less accurate than black-box models [7], they might not even be part of a problem's Rashomon set. Therefore, more work is needed to address model multiplicity for black-box models.

4 Challenges of Integrating Model Multiplicity

4.1 Engineering Interactive Systems incorporating Model Multiplicity

Building upon the frameworks provided by Amershi et al. for Software Engineering for Machine Learning [1], we designed an initial framework for combining these while using a model multiplicity approach. This process framework is depicted in figure 1. For the 'usage' stage, new ways to deal with multiple outcomes need to be designed for end-users to be able to process the multiple outcomes origination from the AI subsystem. In the future, we will explore explainable AI components to improve appropriate trust and understanding of model multiplicity outcomes (e.g., Spinner et al. [16]). Running multiple models simultaneously

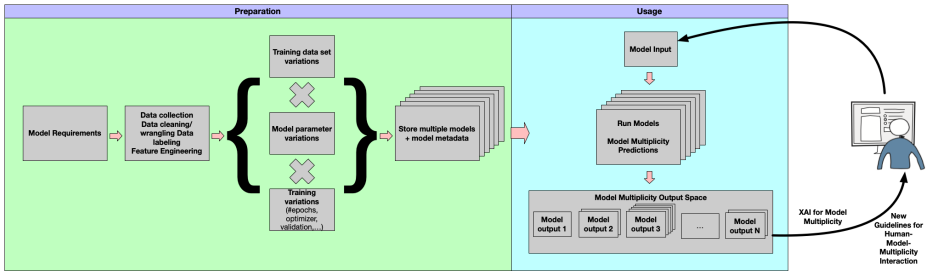

Fig. 1. An initial process framework covering training for model multiplicity and putting these multiple models to use.

results in multiple outcomes that need to be aggregated. Especially for classification models (like neural nets), where the output can be one of many potential classes, getting a visual overview of what models predict which classes can be cumbersome. Visualizing the solution space in itself can be much simpler for predictive models that perform binary classification. However, adding explanations on how models came to a specific conclusion can make the visualizations much more complex, depending on the model being used (e.g., neural net versus decision tree).

4.2 Integration of Model Multiplicity in Interactive Systems

The "Guidelines for Human-AI Interaction" by Amershi, Weld, et al. [2] provide a set of guidelines on integrating AI in interactive systems. We use these guidelines as a reference framework to describe some of the challenges of integrating the outcomes of multiple models in (the user interface of) an interactive software system. We refer to [2] for a full description of these guidelines.

AI Design Guidelines	**Impact of Model Multiplicity on Guideline**
(G1) Make clear what the system can do.	Limited impact, MM uses multiple models for the same task
(G2) Make clear how well the system can do what it can do.	Limited impact, MM uses multiple models for the same task
(G3) Time services based on context. Time when to act or interrupt based on the user's current task and environment.	High impact. MM requires additional computation, thus lower responsiveness.
(G4) Show contextually relevant information.	No difference w.r.t. single model solutions
(G5) Match relevant social norms. Ensure the experience is delivered in a way that users would expect, given their social and cultural context.	Limited impact, conveying multiple inconsistent outcomes might not be accepted in some socio-cultural settings or contexts.
(G6) Mitigate social biases. Ensure the AI system's language and behaviours do not reinforce undesirable and unfair stereotypes and biases.	High impact. MM is meant to provide more balanced results and make it easier to detect bias from certain models.
(G7) Support efficient invocation. Make it easy to invoke or request the AI system's services when needed.	No difference w.r.t. single model solutions.
(G8) Support efficient dismissal. Make it easy to dismiss or ignore undesired AI system services.	No difference w.r.t. single model solutions.
(G9) Support efficient correction.	Medium impact. Since MM might provide the user with a number of possibilities, editing all models involved is not possible. With an MM system, users can explore suggestions or predictions by refining their goal criteria and context.
(G10) Scope services when in doubt. Engage in disambiguation or gracefully degrade the AI system's services when uncertain about a user's goals.	Medium impact. With an MM system, users can explore suggestions or predictions by refining their goal criteria and context.
(G11) Make clear why the system did what it did. Enable the user to access an explanation of why the AI system behaved as it did.	Medium to High impact. For MM using black box AI approaches (e.g., deep neural nets), explaining the behaviour is cumbersome since there is no white box equivalent. For MM using glass box approaches, appropriate visual dashboards are required (e.g., [17]).
(G12) Remember recent interactions. Maintain short term memory and allow the user to make efficient references to that memory.	No difference w.r.t. single model solutions.

(G13) Learn from user behaviour. Personalise the user's experience by learning from their actions over time.	No difference w.r.t. single model solutions.
(G14) Update and adapt cautiously.	High impact. Online, direct updates for MM systems are undesirable, since each of the models involved needs to be updated and validated and each model might be impacted differently.
(G15) Encourage granular feedback. Enable the user to provide feedback indicating their preferences during regular interaction with the AI system.	High impact. Online, direct updates for MM systems are undesirable, since each of the models involved needs to be updated and validated and each model might be impacted differently.
(G16) Convey the consequences of user actions. Immediately update or convey how user actions will impact future behaviours of the AI system.	High impact. Online, direct updates for MM systems are undesirable, since each of the models involved needs to be updated and validated and each model might be impacted differently.
(G17) Provide global controls. Allow the user to globally customise what the AI system monitors and how it behaves.	No difference w.r.t. single model solutions.
(G18) Notify users about changes. Inform the user when the AI system adds or updates its capabilities.	No difference w.r.t. single model solutions.

Interactive systems, particularly those used in sensitive applications, require consistent and reliable outputs. Integrating multiple AI models with differing decision-making processes into a single system increases the complexity of maintaining this consistency, potentially leading to conflicting results that can confuse users and erode trust. Trust is, however, foundational to the use and adoption of AI systems [4]. Model multiplicity can undermine trust if users receive inconsistent responses or cannot understand the basis of decisions due to the opaque nature of some AI models. This is particularly problematic in systems that affect life-altering decisions, such as healthcare diagnostics or judicial sentencing. Furthermore, deploying multiple models simultaneously can lead to inefficiencies in system performance, including increased computational costs and complexities in system maintenance and updates. This may result in slower response times and higher operational costs.

5 Conclusion

Model multiplicity presents unique challenges in the integration of AI into interactive systems. To address these issues, mainly related to trust and transparency, we have proposed a shift toward a "many simultaneous expert advisors" approach. This approach can enhance the accuracy and reliability of AI systems

compared to using a single model. Current approaches mainly offer a way to explore the outputs of multiple (typically white-box and tree-based) models in a manner accessible to end users. There are still numerous avenues for further research and development in this field. Future work could focus on expanding the exploration of model multiplicity toward black-box models and addressing the challenges regarding increased computational costs and limited responsiveness to user interaction. To facilitate this process, we designed an initial framework that uses a model multiplicity approach for explainable software engineering.

Acknowledgments. This work was funded by the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme, R-13509, and by the Special Research Fund (BOF) of Hasselt University, BOF23OWB31.

References

1. Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., Zimmermann, T.: Software engineering for machine learning: A case study. In: 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). pp. 291–300 (2019). https://doi.org/10.1109/ICSE-SEIP.2019.00042
2. Amershi, S., Inkpen, K., Teevan, J., Kikin-Gil, R., Horvitz, E., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P.N.: Guidelines for human-AI interaction. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19. pp. 1–13. ACM Press (2019). https://doi.org/10.1145/3290605.3300233
3. Antifakos, S., Kern, N., Schiele, B., Schwaninger, A.: Towards improving trust in context-aware systems by displaying system confidence. In: Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices & Services. p. 9. ACM Press (2005). https://doi.org/10.1145/1085777.1085780
4. Asan, O., Bayrak, A.E., Choudhury, A.: Artificial intelligence and human trust in healthcare: focus on clinicians. Journal of medical Internet research **22**(6), e15154 (2020)
5. Black, E., Raghavan, M., Barocas, S.: Model multiplicity: Opportunities, concerns, and solutions. In: 2022 ACM Conference on Fairness, Accountability, and Transparency. FAccT '22, ACM (Jun 2022). https://doi.org/10.1145/3531146.3533149
6. Breiman, L.: Statistical modeling: The two cultures (with comments and a rejoinder by the author). Statistical Science **16**(3) (Aug 2001). https://doi.org/10.1214/ss/1009213726, http://dx.doi.org/10.1214/ss/1009213726
7. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., Elhadad, N.: Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1721–1730 (2015)
8. Coppers, S., Van den Bergh, J., Luyten, K., Coninx, K., van der Lek-Ciudin, I., Vanallemeersch, T., Vandeghinste, V.: Intellingo: An Intelligible Translation Environment. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18. pp. 1–13. ACM Press, Montreal QC, Canada (2018). https://doi.org/10.1145/3173574.3174098, http://dl.acm.org/citation.cfm?doid=3173574.3174098

9. Coppers, S., Vanacken, D., Luyten, K.: Fortniot: Intelligible predictions to improve user understanding of smart home behavior. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. **4**(4) (dec 2020). https://doi.org/10.1145/3432225
10. D'Amour, A.: Revisiting rashomon: A comment on "the two cultures". Observational Studies **7**(1), 59–63 (2021). https://doi.org/10.1353/obs.2021.0022, http://dx.doi.org/10.1353/obs.2021.0022
11. Li, Y., Fujiwara, T., Choi, Y.K., Kim, K.K., Ma, K.L.: A visual analytics system for multi-model comparison on clinical data predictions. Visual Informatics **4**(2), 122–131 (2020). https://doi.org/https://doi.org/10.1016/j.visinf.2020.04.005, pacificVis 2020 Workshop on Visualization Meets AI
12. Marx, C., Calmon, F., Ustun, B.: Predictive multiplicity in classification. In: III, H.D., Singh, A. (eds.) Proceedings of the 37th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 119, pp. 6765–6774. PMLR (7 2020), https://proceedings.mlr.press/v119/marx20a.html
13. Muir, B.M.: Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems. Ergonomics **37**(11), 1905–1922 (Nov 1994). https://doi.org/10.1080/00140139408964957, http://www.tandfonline.com/doi/abs/10.1080/00140139408964957
14. ROTH, W.D., MEHTA, J.D.: The rashomon effect: Combining positivist and interpretivist approaches in the analysis of contested events. Sociological Methods & Research **31**(2), 131–173 (2002). https://doi.org/10.1177/0049124102031002002, https://doi.org/10.1177/0049124102031002002
15. Shoham, Y., Leyton-Brown, K.: Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press (2008)
16. Spinner, T., Schlegel, U., Schäfer, H., El-Assady, M.: explainer: A visual analytics framework for interactive and explainable machine learning. IEEE Transactions on Visualization and Computer Graphics **26**(1), 1064–1074 (2020). https://doi.org/10.1109/TVCG.2019.2934629
17. Wang, Z.J., Zhong, C., Xin, R., Takagi, T., Chen, Z., Chau, D.H., Rudin, C., Seltzer, M.: TimberTrek: Exploring and Curating Trustworthy Decision Trees with Interactive Visualization. In: 2022 IEEE Visualization Conference (VIS) (2022)
18. Xin, R., Zhong, C., Chen, Z., Takagi, T., Seltzer, M., Rudin, C.: Exploring the whole rashomon set of sparse decision trees. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 14071–14084. Curran Associates, Inc. (2022)
19. Yang, F., Huang, Z., Scholtz, J., Arendt, D.L.: How do visual explanations foster end users' appropriate trust in machine learning? In: Proceedings of the 25th International Conference on Intelligent User Interfaces. pp. 189–201. IUI '20, Association for Computing Machinery, Cagliari, Italy (Mar 2020). https://doi.org/10.1145/3377325.3377480