Link Uncertainty, Implementation, and ML Opacity

A Reply to Tamir and Shech

Sullivan, E. Link Uncertainty, Implementation, and ML Opacity: A Reply to Tamir and Shech.
In *Scientific Understanding and Representation* Eds. Lawler, Khalifa, Shech (pp. 341-345).
Routledge.

Emily Sullivan
Eindhoven University of Technology
Eindhoven Artificial Intelligence Systems Institute

This chapter responds to Michael Tamir and Elay Shech's chapter "Understanding from Deep Learning Models in Context" in this volume.

Scientific modeling is an avenue for understanding. Machine learning (ML) models are no exception. However, there are notable differences between ML methods and more "traditional" modeling methods. The ML models are data-driven instead of theory- or hypothesis-driven. They are complex and often opaque. Do these differences have implications for how models enable understanding? In a previous work, I argued that at least ML complexity and opacity does not get in the way of understanding phenomena so long as the link between the model and the target phenomena does not have a high degree of *link uncertainty* (LU) (Sullivan 2022a).

In "Understanding from Deep Learning Models in Context," Tamir and Shech (2022) seek to disambiguate aspects of my work on implementation irrelevance and LU. I generally welcome Tamir and Shech's thought-provoking distinctions; however, I will touch on three areas of disagreement: (1) the limits of implementation irrelevance, (2) empirical versus representational LU, and (3) the target of understanding with ML models.

1 The Limits of Implementation Opacity and Irrelevance

In a previous work, I highlighted that ML models are opaque due to *implementation* opacity (i.e., how ML algorithms and trained models implement functions), and that such opacity is not, in principle, a barrier to understanding phenomena with ML models. Tamir and Shech agree that implementation opacity can be irrelevant and does not matter "any more than drawing a map with red or blue ink matters" (2022, 326). But ultimately, they argue that there is an important distinction between implementation irrelevance and what they call *functionally approximate irrelevance* (FAI). Unlike the colors of a map, other features, like topological facts, cannot be ruled out as mere implementation details. FAI is proposed to mark the difference between mere color choices and the more important topological-like features. The distinction seems to be that there are just some features of models that are *in principle* irrelevant, and other features that are intrinsic difference-makers where the "varied details matter to the studied target" (2022, 330). However, here too, instead of claiming that functional approximation opacity undermines understanding, they do a nice job showing that functional approximation can also be irrelevant for understanding in various contexts.

First, I want to clarify implementation irrelevance and how it relates to the problem of ML opacity, before discussing FAI. Consider a computational system that draws a subway map. Computational systems execute algorithms. These algorithms can be discussed at varying levels of abstraction. We can talk about the algorithm at the high(est) level of abstraction by referring to the draw_subway_map() function, or the various sub-steps of the algorithm, or the sub-steps of its sub-steps. Undoubtedly, some of these sub-steps involve computing various topological features. I am not suggesting that the topological features of the map are irrelevant for adequately representing, say, NYC's subway system. On the contrary, for the draw_subway_map() algorithm to be successful, the topological features of the map output must share the relevant topological features of NYC's subway system. How these topological features are expressed (or computed)—either by a visual depiction of nodes and edges or in mathematical notation—does not affect the general goal of representing NYC's subway system.

We must make a distinction between the *goal* or task of the algorithm and *how* the algorithm achieves this goal or task. My claim in Sullivan (2022a) was that the way algorithms achieve their goals can be opaque—suffer from implementation opacity—and that this opacity is often irrelevant for assessing whether a model can enable understanding of the phenomenon it bears on. However, the general goal of the algorithm must be known for understanding to be possible. That said, this distinction is not so simple. Due to the layers of abstraction in computational systems, lower-level algorithmic goals, such as some topological calculation, become subsumed under a higher-level algorithm. The lower-level algorithms *become* the way of implementing higher-level algorithms. Thus, lower-level algorithm goals are also a matter of implementation.

In my view, higher-level goals of algorithms are necessarily important for evaluating LU and the scope of possible understanding. Concerning ML models, the trained model executes an algorithm that has a specific classification or predictive task / goal. How the algorithm achieves the goal is an implementation question of the various steps that the learned model takes to reach an output, such as a set of learned weights. Which specific learned weights are necessary to know—and their level of description—for understanding phenomena with ML models is *the* question. I have argued that the answer to this question partly depends on the epistemic risk facing the model and is largely an external question regarding LU.

The umbrella term "implementation irrelevance" does not make a distinction in kind between a model implementing color choices, coding language, or feature importance. I sympathize with the inclination for drawing distinctions here. However, I am not sure as to whether there really is a notable difference in kind when discussing understanding. Understanding requires a specific target. This could be the model itself, some phenomena it bears on, or something else.

Distinguishing between the features of a model that are *in principle* irrelevant and those that are intrinsic difference-makers—from a view from nowhere—is untenable. Relevance is related to a target. Perhaps someone is interested in the differences that coding language makes to building and running ML models in real-world scenarios. Some coding languages provide guarantees against certain runtime problems that other coding languages do not. Maybe this makes a difference to the way an ML model functions in a deployed scenario. In this case, the LU that would need to be reduced to provide understanding is the LU between the coding language, the model, and the deployed scenario. Coding language becomes centrally relevant.

In the cases I am generally interested in—understanding real-world phenomena through modeling—the link between the phenomenon and the model regarding color is certainly irrelevant. The link regarding the phenomenon and the coding language is certainly irrelevant.

However, feature importance becomes more salient for determining *how* to reduce LU. It is not a difference in a kind of irrelevance or opacity that is operating here, but rather a difference in the target for understanding. There is no space to say that there are "varied details [that] matter to the studied target," but those details are also irrelevant for understanding the target. Difference-making is entwined with the target. The main disagreement seems to be that Tamir and Shech have a model-centric view of understanding phenomena, whereas in my view models are merely a means—not the target—for understanding phenomena (see §3). Different kinds of opacity or irrelevance are a red herring. The central question is how to reduce LU for various types of targets, and which aspects of how the model works need to be known to reduce LU for specific cases.

Reducing LU may require a comparison between different models, different interpretability methods, or even between a "black-box" model and a "white-box" model. As Tamir and Shech discuss, if applying different interpretability methods to the same ML model reveals that the same set of features are identified as being most important, the less we need to know about how the ML model makes its estimates. I agree. Various robustness checks are undoubtedly important and something I discuss in my (2022a). However, in my (2022a), I did not provide a roadmap for which levels of abstraction are necessary for understanding phenomena. My goals were more modest. I sought to show that the target of understanding is important. And that in actual cases of ML models, we generally know enough about the high-level decision points of a model that the real epistemic problem of opacity becomes an *external* problem, where we look to validating models not by looking inside to what a model is doing, but by looking at external connections or robustness checks with other models.

2 Link Uncertainty and Representation

Tamir and Shech introduce two types of link uncertainty: unintended representational uncertainty and empirical justification uncertainty. The former concerns how the data and the model may inadequately represent targets; the latter concerns whether the ML model has an empirical connection to its targets. I largely welcome this discussion. Answering representational questions is a necessary step for understanding, whether representational mismatches are intentional or not. Moreover, for various targets, issues of representation may go beyond empirical validation, such as the pneumonia risk assessment case I discuss in Sullivan (2022b). Medical researchers developed an ML model exploring the risk of death for patients with pneumonia. The neural network model was highly accurate at the *technical* task of optimizing for risk of death under a certain set of assumptions. However, the model also had a different goal of helping medical researchers predict and devise treatment plans. The model failed at this secondary goal; patients with asthma were classified as low risk. But this low risk was precisely because of existing hospital treatment plans. So, although the model and data uncovered a real empirical link, they did not represent the target of interest.

Does this distinction change the overall relationship between model opacity and understanding? Although I cannot address this question here, I suspect that it does not. There are unresolved disagreements regarding how similar representations need to be of their targets. Issues of data leakage and confounding bias seem like clear candidates for increasing LU. However, other aspects of data manipulation or normalization, like RGB color changes, may not be a representational worry or a case of LU in the way Tamir and Shech seem to indicate. Moreover, if similarity is not a direct aim of representation, then opacity can still be compatible with adequate representation.

3 The Target of ML Models

Tamir and Shech propose the following hypothesis about the target of understanding with ML models:

> *TML Hypothesis*: The target phenomenon of understanding with ML models is the relationship(s) of features represented by the data. (2022, 328)

They say further that:

> If the TML hypothesis is correct…ML models help us understand relationships between features represented by the data, but expecting such models to further provide causal explanations e.g. for why a feature is predictive is inappropriate… (2022, 349; (see also 335-336))

I do not deny that in some contexts the TML hypothesis is true. However, notice that the TML hypothesis is *model* centric: the target consists of understanding the model. A model-centric view of ML overlooks an important way in which scientific modeling can provide understanding of the world. When models enable understanding of real-world phenomena (or possible real-world phenomena), the target of understanding is not the relationships of features represented by the data. Data relationships are used as a means to some further end, or, as I have argued elsewhere with Insa Lawler, models provide understanding by *inducing* explanations of phenomena instead of the model itself being an explanation (Lawler and Sullivan 2021). It is not the ML model alone that provides understanding; the model induces an explanation paired with external links connecting the model to the phenomena that enables understanding. This is why I argued that most ML models merely provide how-possibly explanations. The ML models do not provide causal support themselves, but rather indicate possible (causal) hypotheses that additional research must justify through reducing LU.

The fundamental question that I started with remains: how much detail about the model needs to be known to understand phenomena with ML models? Again, I submit that is largely an external problem of LU. Restricting ML models to the model-centric view of the TML hypothesis goes against the goals that many ML researchers themselves postulate, as well as keeping ML models apart from the rest of model-based science, which unnecessarily constrains our scientific toolbox.

References

Lawler, Insa and Emily Sullivan. 2021. "Model Explanation Versus Model-induced Explanation." *Foundations of Science*, 26(4): 1049–1074. https://doi.org/10.1007/s10699-020-09649-1.

Sullivan, Emily. 2022a. "Understanding from Machine Learning Models." *The British Journal for the Philosophy of Science*, 73(1): 109–133. https://doi.org/10.1093/bjps/axz035.

Sullivan, Emily. 2022b. “How Values Shape the Machine Learning Opacity Problem.” In *Scientific Understanding and Representation* Eds. Lawler, Shech, Khalifa (pp. 306-322). Routledge.

Tamir, Michael and Elay Shech. 2022. “Understanding from Deep Learning Models in Context.” In *Scientific Understanding and Representation* Eds. Lawler, Shech, Khalifa Routledge.