Humans and Computers Working Together to Measure Machine Learning Interpretability

Jordan Boyd-Graber
University of Maryland

Machine learning is ubiquitous: detecting spam e-mails, flagging fraudulent purchases, and providing the next movie in a Netflix binge. But few users at the mercy of machine learning outputs know what's happening behind the curtain. My research goal is to demystify the black box for non-experts by creating algorithms that can inform, collaborate with, and compete with in real-world settings.

This is at odds with mainstream machine learning--take topic models. Topic models are sold as a tool for understanding large data collections: lawyers scouring Enron e-mails for a smoking gun, journalists making sense of Wikileaks, or humanists characterizing the oeuvre of Lope de Vega. But topic models' proponents never asked what those lawyers, journalists, or humanists needed. Instead, they optimized held-out likelihood. When my colleagues and I developed the interpretability measure to assess whether topic models' users understood their outputs, we found that interpretability and held-out likelihood were negatively correlated [Chang et al., 2009]! The machine learning community (including me) had fetishized complexity at the expense of usability.

This is not only a technical improvement but also an improvement to the social process of machine learning adoption. A program manager who used topic models to characterize NIH investments uncovered interesting synergies and trends, but the

results were unpresentable because of a fatal flaw: one of the 700 clusters lumped urology together with the nervous system, anathema to NIH insiders [Talley et al., 2011]. Algorithms that prevent non-experts from fixing such obvious problems (obvious to a human, that is), will never overcome the social barriers that often hamper adoption.

These problems are also evident in supervised machine learning. Ribeiro et al. [2016] provide an example of a classifier to distinguish wolves from dogs that only detects whether the background is snow. More specifically for deep learning, Karpathy et al. [2015] look at the activations of individual cells to highlight their focus.

However, these first steps at interpretability fall short because they ignore utility. At the risk of exaggeration, engineers can only optimize what they can measure. How can we actually measure what machine learning algorithms are supposed to be doing?

To answer that question, we take a brief detour through question answering. Completely open-domain question answering is considered AI-complete [Yampolskiy, 2013]. Question answering is difficult because it has all the nuance and ambiguity associated with natural language processing (NLP) tasks and it requires deep, expert-level world knowledge.

Figure 1: An example quiz bowl question. The question begins with obscure information and gradually uses more well-known clues as it progresses. In our exhibition match, Ken Jennings answered (*) this question before the computer could (""), showing he had deeper knowledge on this topic.

This man ordered Thomas Larkin to buy him seventy square miles of land, leading him to acquire his Mariposa gold mine. He married Jessie, the daughter of Thomas Hart Benton, and, during the Civil War, he controversially confiscated slave-holder property while acting as the leader of Missouri. Kit Carson served as the guide for the first two of his expeditions to survey the American West. For 10 points, name this explorer known as "the Pathfinder" "" who was also the first presidential candidate of the Republican Party.

A: John C. Fremont

We can make short-answer QA more interactive and more discriminative by giving up the assumptions of batch QA to allow questions to be interrupted so that earlier answers reward deeper knowledge. Figure 1 shows an example of a question written to reward deeper knowledge and the positions where our system and Ken Jennings answered the question. A moderator reads the question word by word, and as soon as either player knows the answer, they use a signaling device to "buzz in". If the player has the correct answer, they earn points; if not, the moderator reads the rest of the question to the opponent. Because the question begins with obscure clues and moves to more well-known information, the player who can buzz first presumably has more knowledge.

Fortunately, there is a ready-made source of questions written with these properties from a competition known as quiz bowl. Thousands of questions are written every year for competitions between middle schoolers up to grizzled

veterans on the "open circuit". These questions represent decades of iterative refinement of how to best discriminate which humans are most knowledgeable (in contrast, Jeopardy's format has not changed since its debut half a century ago, it is thus not considered as "pure" a competition among trivia enthusiasts).

Interpretability cannot be divorced from the task a machine learning algorithm is attempting to solve. Here, the existence of quiz bowl as a popular recreational activity is again a benefit: we have thousands of trivia enthusiast forming teams to compete in quiz bowl tournaments. Thus far, our algorithm has not been a good team player; it's only played by itself. Can it also be a good team player? And can it learn from its teammates? Answering these questions can also reveal how useful it is at conveying its intentions.

We have good evidence that quiz bowl serves as a good setting for conveying how computers think. Our trivia-playing robot [Boyd-Graber et al., 2012, Iyyer et al., 2014, 2015] faced off against four former Jeopardy champions in front of 600 high school students.[1] The computer claimed an early lead, but we foolishly projected the computer's thought process for all to see. The humans learned to read the algorithm's ranked dot products and schemed to answer just before the computer. In five years of teaching machine learning, I've never had students catch on so quickly to how linear classifiers work. The probing questions from high school students in the audience showed they caught on too. (Later, when we played again

[1] https://www.youtube.com/watch?v=LqsUaprYMOw

against Ken Jennings,[2] he sat in front of the dot products and our system did much better.)

A growing trend in competitive chess is "centaur chess" [Thompson, 2013]. The best chess players are neither a human nor a computer but a computer and a human playing together. The language of chess is relatively simple; given a single board configuration, only a handful of moves that are worthwhile. Unlike chess, quiz bowl is grounded in language, which makes the task of explaining hypotheses, features, and probabilities more complicated than chess.

Thus, I propose "centaur quiz bowl" as a method of evaluating the interpretability of predictions from a machine learning system. A system could be a part of a team with humans if it could communicate its hypotheses to its teammates. At our exhibitions, we have shown ordered lists of predictions while the system is considering answers. This is effective for communicating what the system is thinking, but not why it wants to provide that answer. Thus, a necessary prerequisite for cooperative question answering is creating interpretable explanations for the answers that machine learning systems provide.

Deep learning algorithms have earned a reputation for being uninterpretable and susceptible to tampering to produce the wrong answer [Szegedy et al., 2013]. Instead of making predictions based on explicit features, one of the strengths of

[2] https://www.youtube.com/watch?v=kTXJCEvCDYk

deep learning algorithms is that they embed features in a continuous space. These representations are central to deep learning, but how these representations translate into final results is often difficult--if not impossible--to diagnose. Ribeiro et al. [2016] propose LIME (Local Interpretable Model-agnostic Explanations): linear approximations of a complicated deep learning model around an example.

LIME can, for example, create a story of why a particular word caused an algorithm to answer a question in a particular sentence. A logistic regression (a linear approximation of a more complicated predictor) can explain that seeing the words "poet" and "Leander" in a question would be a good explanation of why "John Keats" would be a reasonable answer. However, it would be even better to highlight the phrase "This poet of On a Picture of Leander" as its explanation.

I propose to extend lime's formula to capture a larger set of features as possible explanations for why a model makes the predictions it does. Individual words are often poor clues for why the algorithm suggests a particular answer.

For example, "And no birds sing" is a well-known quote from the poem La Belle Dame Sans Merci, but explaining the prediction by providing a high weight for the single word "sing" would be a poor predictor. The algorithm should make itself clear by explaining that the whole phrase "no birds sing" is why it thinks La Belle Dame Sans Merci is the answer. While recurrent neural networks can discover these multi-word patterns, they lack a clear mechanism to communicate this clue to a user.

Fortunately, quiz bowl provides the framework we need to measure the collaboration between computers and humans. The goal of team quiz bowl is to take a combination of players and produce a consensus answer. Thus, it is the ideal proxy for seeing how well computers can help humans answer questions if we can separate out how well the computer aids its "teammates".

Just as baseball computes a "runs created" statistic [James, 1985] for players to gauge how much they contribute to a team, quiz bowlers create statistical analyses to determine how effective a player is.[3] A simple version of this analysis is a regression that predicts the number of points a team will win by (negative if it's a loss) when given a set of players on a team.

But there are two independent variables we want to understand: the effect of the algorithm and the effect of visualizations. We thus analyze the effect of a question answering system and a visualization as two distinct "team members". The better a visualization is doing, the better its individual statistics will be. This allows us to measure the contribution of a visualization to overall team performance and thus optimize how well a visualization is communicating what a machine learning algorithm is thinking.

[3] SQBS, http://ai.stanford.edu/~csewell/sqbs/

Combined with the renaissance of reinforcement learning [Thrun and Littman 2000] in machine learning, having a clear metric based on interpretability allows algorithms to adapt their presentations to best aid human collaboration. In other words, the rise of machine learning in our everyday lives becomes a virtuous cycle: with a clear objective that captures human interpretability, machine learning algorithms become less opaque and more understandable every time they are used.

Despite the hyperbole about an impending robot apocalypse surrounding AI killing all humans, I think that a bigger threat is automation disrupting human livelihood. In juxtaposition to the robot apocalypse is a utopia of human-computer cooperation, where machines and people work together using their complementary skill sets to be better than either could be on their own. This is the future that I would like to live in, and if we are to get there as engineers we need to be able to measure our progress toward that goal.

References

Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daum'e III. Besting the quiz master: Crowdsourcing incremental classification games. In Empirical Methods in Natural Language Processing, 2012.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. Reading tea leaves: How humans interpret topic models. In Proceedings of Advances in Neural Information Processing Systems, 2009.

Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daume III. A neural network for factoid question answering over paragraphs. In Proceedings of Empirical Methods in Natural Language Processing, 2014.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daum'e III. Deep unordered composition rivals syntactic methods for text classification. In Association for Computational Linguistics, 2015.

Bill James. The Bill James Historical Baseball Abstract. Villard, 1985. ISBN 0394537130.

Andrej Karpathy, Justin Johnson, and Fei-Fei Li. Visualizing and understanding recurrent networks. CoRR, abs/1506.02078, 2015. URL http://arxiv.org/abs/1506.02078.

Marco T'ulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Knowledge Discovery and Data Mining, 2016.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. CoRR, abs/1312.6199, 2013. URL http://arxiv.org/abs/1312.6199.

Edmund M. Talley, David Newman, David Mimno, Bruce W. Herr, Hanna M.

Wallach, Gully A. P. C. Burns, A. G. Miriam Leenders, and Andrew McCallum. Database of NIH grants using machine-learned categories and graphical clustering. Nature Methods, 8(6):443-444, May 2011. ISSN 1548-7091.

Clive Thompson. Smarter Than You Think: How Technology is Changing Our Minds for the Better. Penguin Group , The, 2013. ISBN 1594204454, 9781594204456.

Sebastian Thrun and Michael L. Littman. A review of reinforcement learning.

AI Magazine, 21(1):103-105, 2000.

Roman V. Yampolskiy. Turing Test as a Defining Feature of AI-Completeness, pages 3-17. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-29694-9. doi: 10.1007/978-3-642-29694-9 1. URL http://dx.doi.org/10.1007/978-3-642-29694-9_1.