International Journal of Digital Humanities (2023) 5:431–449
https://doi.org/10.1007/s42803-023-00075-w

ORIGINAL PAPER

On reading and interpreting black box deep neural networks

James E. Dobson[1]

Received: 1 March 2023 / Accepted: 27 October 2023 / Published online: 20 November 2023

Abstract
The deep neural networks used in computer vision and in recent large language models are widely recognized as black boxes, a term that describes their complicated architectures and opaque decision-making mechanisms. This essay outlines several different strategies through which humanist researchers and critics of machine learning might better understand and interpret the class of deep learning methods known as Transformers. These strategies expose different aspects of what might be "learned" as Transformers are trained and used in the analysis of language and can help critics at least partially open the black box of machine learning. They are also especially useful for digital humanists using these models as part of a research program informed by tool criticism in which the use of computational tools is conceived of as a metainterpretive act.

Keywords Hermeneutics · Interpretability · Explainability · Deep learning · Large language models

1 Introduction

Almost since the invention, or discovery, of machine learning and artificial intelligence, researchers and the public have had pressing questions about access to underlying parameters and decision-making logics and a desire for greater interpretability of the resulting outputs. Some of these initial questions were prompted by the occasionally extravagant claims and prognostications made by earlier observers of machine learning, most especially those made by journalists viewing staged demonstrations of learning systems. Other questions about the opacity of artificial intelligence methods came from within the nascent field of computer science. In John McCarthy's 1974 review of the "Lighthill Report," James Lighthill's assessment of what he understood

✉ James E. Dobson
James.E.Dobson@Dartmouth.EDU

[1] Department of English and Creative Writing, Dartmouth College, HB 6032, Hanover 03755, NH, USA

to be the dismal delivery on the promises of artificial intelligence, McCarthy commented on a strain of a "disease" in artificial intelligence research, the presentation of impressive results without any explanation of how these results were obtained. Using the example of posted chess tournament scores from programs using AI methods, McCarthy explains that for these results to be of any use for computer science, "[we] need to know why one program missed the right move in a position-what was it thinking about all that time? We also need an analysis of what class of positions the particular one belonged to and how a future program might recognize this class and play better" (McCarthy, 1974). In the contemporary moment, users of machine learning-enabled technologies, from those inserting images into Microsoft Office and interpreting the automatically generated alt-text captions to users investigating the possibilities and limitations of OpenAI's ChatGPT, are asking similar questions to those posed by McCarthy: what was the algorithm thinking at the time? What sort of training data might have produced this response? What image features and criteria are responsible for the classification of this image? These questions are not likely to disappear as machine learning continues its movement from the margins of computing to cultural ubiquity. These everyday playful explorations share with academic critical scholarship an interest in understanding the conditions of knowledge production involved in machine learning and artificial intelligence.

This essay investigates theoretical concerns and practical aspects of probing and understanding the operation of deep learning Transformer models. Like many deep learning architectures, Transformers are large neural networks with multiple layers. While the degree of depth is relative (networks of twelve to ninety-six layers are common at present), understanding how these models work becomes more complex as layers are added. Deep learning models power many of the most frequently used contemporary machine learning applications, including computer vision tasks and large language models (LLMs). They are a class of algorithms that, as Nick Seaver notes in his account of critical algorithm studies, "are often so complex that they pose interpretive challenges even to their makers" (Seaver, 2019). The large scale and depth of Transformers and the complexity of their inputs have enabled an increasingly wide array of applications-contemporary multi-modal models can classify thousands of types of objects found in digital images and translate into numerous languages-but has also added greatly to the difficulty in understanding the behavior of these models. Another important implementation feature of these recent architectures, the regular separation of training into multiple and distinct stages of pre-training and fine tuning, has also added to the complexity of understanding the origin of a model's outputs, which is to say the relative weight and influence of these stages on the model's behavior. These complexities are not entirely insurmountable. A range of critical methods both novel and well-proven, including hermeneutic strategies developed during earlier phases in the critique of machine learning, can be deployed by humanists seeking to interpret black-box machine learning models.

In calling machine learning methods black boxes, researchers are responding to both the complexity of understanding the models as well as the lack of access to the models, frequently because they require sophisticated software or computing hardware (GPUs and other accelerators specialized for matrix manipulation) or simply because

access has been denied as laboratories and corporations consider the models trade secrets. As these two situations make clear, the creators of a model or system might be invested in opacity and even make use of more complex architectures and transformations in service of added obscurity to protect their products. The notion of a black box also serves corporate interests by disavowing agency in the construction and training of models and limiting access to those attempting to interrogate the models. Even in an "open" and transparent environment, model creators may position themselves as working under black box conditions because of architectural complexity and privileged access to non-public training sources. This framing also participates, although to a lesser degree than those in closed and corporate environments, in structures of disavowal. It also seems quite likely that black box discourse is historically contingent: as methods become better understood and strategies for visualization, examination, and evaluation of complex data structures and model architectures become available, the frame of the black box disappears (although the obsolescence of these methods might happen before they are exposed and out in the open, as it were). Humanists tend to focus their attention on the public-facing use of technologies, including recent projects featuring generative LLMs, and it is essential work to account for the operation of these projects and applications and to interpret their outputs and the ways they make meaning.

My account of the humanities in this essay follows my earlier work in centering interpretation in critical digital studies (Dobson, 2019). I understand hermeneutics to be one of the core activities and methods of humanists and what makes computation compatible with humanistic research. There are other accounts, especially those that foreground the objects of study rather than methods as key to understanding the humanities. While some might define the humanities by the objects of study (for example, the literary archive) and say that all forms of computational literary studies are humanistic by virtue of their objects of analysis, I am more concerned with assumptions about the ontological status of the object as such. In taking up interpretation as a core activity, we resolve a conflict into the method/object distinction introduced by the digital humanities. This conflict concerns the status of the model as the object of interpretation. When a scholar like John Guillory points out that the presence of scientific methods in humanistic approaches does not remove these analyses from the humanities because they are still focused on what he terms the "given object," he assumes an isomorphic relation between the humanistic object (painting, text, etc) and the object under analysis (Guillory, 2016). The distinction between the constructed and given object in Guillory's account assumes that when scientific methods are used to generate knowledge in the humanities, they are generating knowledge about the given object. Computational modeling erodes this distinction as statistical models are interpretations of constructed objects that, using Guillory's terms, gesture toward the given object. Following Don Ihde's notion of an expanded hermeneutics, a reading strategy that can be used to critique the interpretive function of scientific instruments, I propose that we can understand Transformers, as interpretive instruments (Ihde, 1998). If one considers computational instruments such as Transformers as hermeneutic in nature, then the way in which these methods re-present humanities objects cannot be ignored or assumed to be irrelevant to the question of interpreting the output alongside the modeled object. Computation has also introduced recursion into the object/method

schema: the separation of objects from methods becomes increasingly hard to justify as methods produce objects, typically numerical models, that are then interpreted by other methods. I regard the interpretation of computational interpretations as a necessary component of any computational work within the humanities and thus privilege those methods that provide access to input data, parameters, and the nature of the transformations applied.

Explainable AI (XAI) is a major topic in computer science and engineering and researchers are developing a range of their own techniques to explain deep learning (Guidotti et al., 2019). As Transformers are the enabling technology powering many of the most widely used large language models, understanding and interpreting the behavior of this architecture and the meaning of its output is of special interest to scholars located in the humanities. This is not only because these models are trained on massive textual archives and are now routinely used to produce new texts, but also because humanists using such tools in their research and scholarship have a special responsibility to demystify and critique these technologies (Dobson, 2021). An important framework for such a practice known as "tool criticism" (van Es, 2023) urges the opening of black boxes found in computational systems as much as possible, proposing that "critically engaging with these tools on a theoretical level can be accomplished by understanding the logics and principles of their functioning" (van Es et al., 2018). In using black box models instead of more readily interpretable open and/or transparent models, the cost-benefit analysis for humanists should take into consideration both relative increases in task performance and the loss of interpretability and potential reproducibility. While explainability and interpretability in the machine learning context might seem like cognate terms-both relate to the understanding of a model's operation and function-these tasks take on different objects and are directed toward distinct goals (Miller, 2019). While an explainable model might produce confidence measures for a prediction or decision (i.e., the display of 80% confidence that an image represents an object belonging to a particular class of objects), this does not mean that the model or its outputs are interpretable. Explanation is concerned with the production of an account of how something works or how a decision was made; interpretation offers an account of the possible meaning of ambiguities discovered during the operation of the model. David Berry argues that explainability in the sciences is descriptive and based on a "formal, technical and causal model" (Berry, 2023) and that efforts to understand the operation of machine learning in automated decision making should be informed by a humanist sense of understanding. Foregrounding interpretation rather than explanation enables understanding by exposing the ways in which machine learning itself makes use of hermeneutical operations.

2 Complications to interpretability

The remainder of this essay will focus on the possibilities of interpreting the behavior and outputs of Transformers. These multi-layered networks provide state-of-the-art performance on numerous benchmarks for text-based language tasks and have also been applied to other data objects, most recently including images for use in computer vision applications. The Transformer architecture is used by Google's BERT

(Bidirectional Encoder Representations from Transformers) network, OpenAI's GPT (Generative Pre-trained Transformer) series of large language models, Meta's LLaMA (Large Language Model Meta AI), and an increasingly large number of other widely available pre-trained models. Transformers have been widely used for numerous natural language processing tasks, trained for use as classifiers, and most visibly in recent years, as generative models to produce new text segments. While much of what is known about the operation of Transformers is the result of investigations of smaller relatively open models like BERT and GPT-2, some of this knowledge can be applied to larger closed models of a similar architecture.

The Transformer architecture was introduced in a 2017 conference paper titled "Attention is All You Need" (Vaswani et al., 2017). Building on earlier sequential embedding models, the major innovation of the Transformer was a shift to only use the self-attention mechanism to model the different possible positions in sequences of inputs. Positional encodings enable Transformers to learn from these different sequences. This is to say that Transformers learn distinct representations for the same set of words when they appear in different order. The invention of Transformers enabled a shift in the text modeling practices of humanists from static to contextual embeddings. Simultaneously encoding and learning text segments from word and subword tokens (typical via WordPiece, Byte-Pair Encoding, or other similar tokenization schemes) to sentence-level or larger segments of text, Transformers address limitations in prior neural language models (such as word2vec and fasttext) that made it difficult for these networks to encode the multiple contextual meanings of individual tokens.

The learning in Transformer models takes place in multiple locations and especially within the network component called attention heads that were gestured to in the title of the conference paper in which they were introduced. This complexity adds to the difficulty of tracking how these networks encode and model text. Multiple attention heads read the supplied sequence of input text and learn relations between tokens. Depending upon the model, these self-attention heads read the sequence of tokens in one direction or both backward and forward, learning from the different contextual samples and updating the network. Output from the multiple heads in a single layer are combined and forwarded to higher layers. A generalized pre-training task, like masked language modeling (token prediction) or next sentence prediction forms the basic training of many Transformer models. While this is called "self-supervised" learning in the sense that samples are not labeled, the choice and implementation of the model architecture, model parameterization, and the curation of the datasets are additional sites of supervision. The information learned from these pre-training tasks on provided training datasets are what enables it to perform well on other tasks. Exactly how well such models generalize from these samples remains an open research question. This paradigm enables a division of labor, time and increasingly expense, in that a model creator can provided a pre-trained large language model and an implementor or researcher can modify ("fine-tune") the model for another specialized task that can take advantage of the already-learned general language model. Increasingly these general models are being called foundation models as they serve as the basis for building applications. While I will focus on Transformer models, some of the problems and hermeneutic strategies offered, in terms of techniques for opening these boxes, are applicable to other existing and future deep neural network-based architectures.

There are several complications to the understanding of Transformer models. As I have mentioned above, these models are considered black boxes for reasons beyond their technical complexity or the feasibility of understanding their operation. In terms of issues related to technical complexity, one might point to the numerous newly developed and not well understood architectural features and rapidly increasing number of layers in recent models. These models, especially when used to create LLMs, have been critiqued for introducing several risks connected with their size and their use of problematic web-based training data. Emily Bender et al. outline some of the risks associated with these models, most importantly the lack of diversity in training data and the presence of encoding bias, in their widely cited paper "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" (Bender et al., 2021). This paper anticipated the discoveries made by many casual users of LLMs in the form of chatbots and other generative applications and echoed the experiences of those encountering racist and sexist outputs when using earlier neural language models such as the ones used by Google Translate.

This problem of size, the incredible number of parameters and dimensions of data at every level in contemporary deep neural networks, also contributes to making visualization impossible without some distortion of the underlying data and their relations. As one cannot easily visualize data beyond three dimensions, dimensionality reduction methods such as t-SNE, UMAP, or PCA, to mention three popular methods, are regularly used to extract what these methods understand to be the most meaningful aspects of the data by mapping relations among embeddings (the hidden state embeddings have 768 dimensions for BERT base and 4,096 for Mistral 7B-v0.1) into a lower dimensional space for two or three-dimensional visualization. But what features to select for visualization? The clustering of data introduces problems in defining centers, and distance metrics in general, present a choice among possible measures and hard-to-interpret scales of difference that are at best highly relative.

Another difficulty relates to the typical division of training activities with Transformer models. While descriptions of released pre-trained models might explain the presence of biases that might influence them, even after fine tuning, it is not clear how to separate the two stages of this training procedure. As Hugging Face, the developers of the Transformers package, note in their description of racially biased outputs in the pretrained GPT-2 model, "[this] bias will also affect all fine-tuned versions of this model" (Hugging Face, 2022). This is to say that it is very difficult to detect changes to a model, whether these were maliciously introduced or performed as part of the normal fine-tuning procedures. In withholding and limiting access to GPT-3 (and later models), unlike their open release of prior models, OpenAI provided a rationale rooted in precisely this concern. OpenAI claims that these large models are too powerful and open to misuse. In their blog post announcing the last release of GPT-2 (their 1.5B parameter model), OpenAI was especially concerned with the manipulation of their language model through fine-tuning:

> GPT-2 can be fine-tuned for misuse. Our partners at the Middlebury Institute of International Studies' Center on Terrorism, Extremism, and Counterterrorism (CTEC) found that extremist groups can use GPT-2 for misuse, specifically by fine-tuning GPT-2 models on four ideological positions: white supremacy,

> Marxism, jihadist Islamism, and anarchism. CTEC demonstrated that it's possible to create models that can generate synthetic propaganda for these ideologies. They also show that, despite having low detection accuracy on synthetic outputs, ML-based detection methods can give experts reasonable suspicion that an actor is generating synthetic text. OpenAI (2019).

As their announcement makes clear, OpenAI believes that interpretability and the ability to determine what changes to a model were introduced during fine-tuning, are crucial but tamper-resistant models or signed models are not generally available. The creation of tools to measure the effects of fine-tuning on language models will be necessary for humanities researchers to use these models on corpus or archive-specific tasks and to provide assurances that the models are functioning as expected.

3 Ways of opening black boxes

To some degree, efforts to expose interfaces, performance, and the learning involved in machine learning have been present since its invention. In the late 1950s, Frank Rosenblatt and his colleagues borrowed visualization strategies and metrics from their own psychological studies as well as other behavioral studies to visualize the function of their mechanical and simulated neural architectures (Rosenblatt, 1958). It should be recognized that the basic model evaluation tools within machine learning-the confusion matrix and associated recall, precision, and accuracy scores-do provide both some low-level explanations (e.g., in many class models, in which classes are most commonly confused with each other) and pointers to launch investigations into the training and testing datasets through the identification of boundary classes and objects within the model. These simple metrics, however, do not enable researchers and critics to understand how these "correct" and "incorrect" predictions were obtained and the features that were considered meaningful in determining these predictions. As machine learning developed, the ability to look at the individual units of the networks and to examine the weights of features was in tension with the drive to produce more complicated networks and methods that would deliver state of the art performance on industry and research field benchmarks.

Like data and code, complex deep learning models, to some degree, can be interrogated and read. Meredith Broussard, a data journalist, describes the generation of summary statistics from a dataset as "interviewing the data" (Broussard, 2018). The model's vocabulary, which is to say its list of tokens, can be extracted and much about these models can be learned from what can and can't be tokenized and thus represented by the model (Shoemaker, 2023). Many models provide some very basic information that can be "interviewed," as it were, for some understanding of the model's architecture, provenance, and construction. While this information, in its present configuration and as stored within the model itself, is rather limited, the "config" property exposed by many common implementations of Transformers reveals some parameters. These might include the number of layers and attention heads, the embedding vector dimensions, token vocabulary size, and some parameters used in pre-training. Building on

the earlier argument for producing "datasheets for datasets," Margaret Mitchell and colleagues propose "Model Cards for Model Reporting," a reference form to be used by researchers and model constructors to share contact information, state intended use, provide detail about the construction and training of the model, and declare ethical considerations and limitations of the model (Mitchell et al., 2019). This scheme has had some success, most notably with many of the models and datasets distributed by Hugging Face, and even some third parties have begun creating cards for undescribed models (generally hosted in code repositories as markdown documents).

Treating the tool, the model, or the algorithm, as withholding knowledge (even if it might not be appropriate to use that term for the information found within a digital object) enables a rethinking of the task of critique. In directing critical inquiry toward the shape and scope of the model, it becomes possible to refocus questions on the production and deployment of that withheld knowledge. One might imagine a model making "slips" and revealing information that may expose the construction of the training data, the training task, fine tuning procedures, and even insertions or modifications of input sequences. So-called hackers, reverse engineers, and the intelligence and reconnaissance community make use of similar strategies that range from examining information leaking from models to explicitly adversarial procedures that seek to probe responses to unexpected inputs. Some of these strategies have been adopted by everyday users of artificial intelligence tools, and especially the hosted generative applications like chatbots and image generation tools that provide limited access to the underlying models.

3.1 Examining training data

For machine learning in the present, and in what would be considered relatively open environments with well-described models built using common open-source machine learning packages (i.e., Scikit-Learn, Pytorch, Transformers, etc), the first level and point of access to knowledge about the models would be the inspection and evaluation of training and testing data. This is especially true with difficult to interpret deep learning methods. Despite being released as open-source projects and distributed with datasets, some of the truly large models recently developed have been trained on undefined data collections, either because of privileged access to private data or because these datasets are drawn from illicitly scraped and therefore unshareable data. The list of these is sprawling and includes the neural language model word2vec that was released with pre-trained embeddings extracted from a private Google News dataset; complete versions of the scraped dataset of amateur-produced books known as BookCorpus that was used, with other datasets, to train Google's BERT and many other models (Bandy & Vincent, 2021); the collection of outbound linked documents from Reddit known as WebText that was used to train OpenAI's GPT models; the set of books found in the Books3 dataset and exposed by Alex Reisner in *The Atlantic* (Reisner, 2023); and the Common Crawl dataset and extractions from that are used to train many tools and networks, especially those making use of image data. Adding to the complexity of parsing datasets, many of these are combined together in training

collections like The Pile, an 800GB dataset composed from twenty-two sub-datasets (Gao et al., 2020) that range from Enron emails to scientific and research articles archived in PubMed Central.

Despite the many cases of private, lost, or otherwise unavailable training datasets, many machine learning models are released alongside their training datasets and these data can be examined on their own or in concert with an examination of the model. Kate Crawford and Trevor Paglen's "Excavating AI: The Politics of Training Sets for Machine Learning" project explores the datasets and labels provided by the ImageNet, a highly organized taxonomy of images widely used in computer vision research (Crawford and Paglen, 2019). ImageNet is not a static collection; it is an ongoing research project that works in concert with developments in the field and the major benchmarks and competitions that shape the priorities of the field. In response to critiques like the one made by Crawford and Paglen and many others, ImageNet has also revised its taxonomy and removed many images, rendering many branches of the tree in "ruins," as Nicolas Malevé has observed (Malevé, 2021). Others have investigated CommonCrawl and its derived datasets, including those coupled to language models (Birhane et al., 2021). Access to training data, importantly, can reveal attributes or features of these data that are not necessarily visible or exposed in the resulting models (Denton et al., 2021). Depending on the nature of the data, an investigation of the data can involve viewing the images or reading the texts. It can also involve reading and interpreting assigned labels and an understanding of any preprocessing that was performed (image normalization, removal of vocabulary, spelling standardization, etc). From these data, in the case of a large number of machine learning models, much more can be learned about the model than would be found in the architecture of the model itself.

3.2 Prompting and generative evaluation

The recent development of a repertoire of interlinked and multi-clausal queries known as prompt engineering or "promptology" might be understood as a form of probing, a method by which knowledge of a model's construction and operation might be extracted or revealed by working within the design constraint of an interface. Likewise, in-context learning (ICL), the name given to the category of learning from training samples provided as input without updating the model, can be used to extract information about a model and its limitations (Chan et al., 2022). Like the algorithm audits of closed-source commercial applications using carefully assembled datasets that have been used to determine exclusions in training datasets (Buolamwini, 2023), generative models can reveal what has been excluded and included. These models appear, for example, to leak the contents of training datasets through a cloze task, the completing of a masked prompt with text-specific tokens (Chang et al., 2023). Carefully crafted prompts can expose pre-existing prompts, that is to say some modification of queries through the manipulation of user-supplied input. These strategies have been especially useful to expose the functioning of hosted black box methods. Fabian Offert and Thao Phan propose that "humanist tactics" can become a metalanguage to reveal the implicit norms in systems like the image generative application DALL·E 2 by interpreting the

results of prompting the application with "A sign that spells" and interpreting the resulting images that show the limited debiasing implemented by OpenAI to be little more than the tacking on of diversity signifiers to prompts (Offert & Phan, 2022).

With access to pre-trained models and deep learning frameworks, many common Transformers can be used to generate text using relatively simple next-token prediction or using more complex pipelines and procedures (i.e., Greedy Search, Beam Search, Top-K sampling, etc). Using text generation, prompts can be designed that reveal relatively stable predictions from the sampled language used to pre-train the model. Such queries can reveal bias within the model, the collection used as training data, the historical moment in which training data were collected and the model generated, and much more. Text generation methods offer a rough sense of the norms encoded by the language model. This straightforward use of Transformers-which is typically the first experience that most people have with large language models through the use of chatbots and other generative applications-provides one way of watching model inference in progress.

It can be difficult to distinguish between memorization of samples, a condition that researchers link to overfitting, and the norms embedded as highly likely probabilities within training data (Tirumala et al., 2022). One can also generate a list of closest probable predictions by sorting rather than taking the maximum value of the inference returned by the model. While the sorted distribution of generated tokens returned might give one comfort that the model possesses information about numbers and addition, the uncertainty of why these other numbers appear suggests that this information might not be entirely about addition. Iteratively experimenting with text generation exposes something about what has been learned from the samples in the training data, although there are many variables that compound the difficulties of interpreting this machine interpretation of the model. These include the measures and parameters used as selection criteria, such as the Top-K sampling previously mentioned, or the temperature variable used in the generation of returned probabilities. In generative applications, the use of these selection criteria methods is thought to increase linguistic diversity and render the predictions less predictable.

Researchers have experimented with understanding these predictions as a form of "knowledge storage" and developed methods to change or edit the network to introduce counterfactual information and associations (Meng et al., 2023). While these predictions, with increasing frequency as the models improve and increase in parameters, may return what could be considered "correct" responses, they should not be confused with facts or knowledge as such. OpenAI's model card for GPT-2 lays the case out clearly that there are no such things as "facts" in large language models: "Because large-scale language models like GPT-2 do not distinguish fact from fiction, we don't support use-cases that require the generated text to be true" (OpenAI, 2019). While they might not be repositories of facts, large language models have been shown to expose personal identifying information contained within training data. Prompts can easily be constructed to leak information by completing substring retrieval tasks from information available on the public web that was likely scrapped as training data for large language models (Carlini et al., 2021).

3.3 Adversarial testing and probing

While training data might give insight into the imaginative construction of the model (its framing of the task at hand, its expected inputs, its ontology, especially in the computational and information science sense of the term, etc), behavior on new input data might produce more insight into its operation. This sort of knowledge-knowledge that might be contained within predictions of most likely output classes, nearest neighbors data, etc-can be made meaningful through the selection of problematic data, in terms of assumed unrepresentative features of a particular class, or overdetermined data, characterized by the dominance of class-specific features, and the construction of datasets featuring antipodal samples, such as negative and positive examples of a particular class. Jill Walker Rettberg takes what she calls "algorithmic failure" as a key method for humanities researchers to use machine learning "against the grain" to investigate false positives from classification texts (Rettberg, 2022). This allows Rettberg to explore assumptions and ambiguities in the categories used by the classifier (i.e., active vs. passive actions) and within datasets. This strategy of examining black box classification probes results without directly querying what has been learned. Adversarial testing is a variant of probing models with testing data by involving the selection of intentionally disruptive data objects and features as inputs to the models. These can be considered disruptive because they are designed to exploit the limits of learned decision criteria by using boundary objects (Star, 1989), those data that might represent features from multiple classes or perhaps those that contain the minimal possible set of features to invoke a specific response.

3.4 Exploiting architecture-specific features and knowledge

Some of the best strategies for understanding deep neural networks at present focus on the behavior of specific major components of a model's architecture. The exposure of the hidden layers contained within a model as well as the attention heads in Transformer models can provide some measure of model specialization, which is to say the degree to which these models have learned to distinguish among features from supplied language samples. The transition from static embeddings in neural language models to contextual embeddings in Transformers adds considerably to the difficulties in interpreting what has been learned from pre-training as the embeddings are not as meaningful as the pre-trained static embeddings from these earlier models. The initial or input weights might be made available as a property of the model (in GPT-2, for example, as "transformer.wte.weight") and used for learning more about the model. The embedding space of the pre-trained model can be productively probed using cosine similarities of the mean values of various tokens of interest. These can be used to expose the construction of the training data, ideological content, and embedded biases. While these token embeddings, when used as static features, might be of limited value for many tasks, especially as sources of semantic information, they may provide information about the model and its learned language features prior to running contextual samples, which is to say sentences or longer passages of text, through the model. Queries of the embedding matrix of such a model only use information

internal to the model; contextual samples of language run through the model introduce additional information but also provide access to the architectural features (and differences) of these Transformers.

Opening the black box of deep learning involves more than a little speculation and guesswork and there are limited best practices at present to assist with selecting from multiple sources of information contained within the model. The two major architectural features to target for critical readings of Transformer models would be the hidden layers and the attention heads. Research has shown that in the architecture of deep neural networks, the higher layers recognize more complex features (Tenney et al., 2019). This is to say that lower-layers are specialized for lower-level language features:

> In image classification models, lower layers recognize more generic features such as edges while upper layers recognize more class-specific features (Yosinski et al., 2014). Similarly, upper layers of LSTMs trained on NLP tasks learn more task-specific representations (Liu et al., 2019a). Therefore, it follows that upper layers of neural language models learn more context-specific representations, so as to predict the next word for a given context more accurately. Ethayarajh (2019)

The input weights shown above provide some compelling evidence that the embedding space might be specialized for learning lower-level language features. We might productively use nineteenth-century German philosopher Friedrich Schleiermacher's hermeneutics and his distinction between grammatical and technical interpretation to cast the movement up the layers of abstractions in deep neural networks as a transition from grammatical to technical methods for the analysis of discourse (Schleiermacher, 1998). In Transformer models, embeddings can be extracted from multiple hidden layers for individual words and the entire input string. Depending upon the model, some experimenting with extracting embeddings from the final, higher layers might be necessary. Following the above suggestions, comparisons of these context-sensitive embeddings can be made across several of these final hidden layers to understand how meaningful these differences might be or, perhaps, mean embedding values could be calculated from multiple layers to generate more stable embeddings to enable the comparisons of key terms in different contexts.

By examining contextual token embeddings in a pre-trained model, one might bring to light differences among a set of sample text fragments or discourse statements, although potentially problematically filtered through the learned grammatical language features from pre-trained data. Nearest neighbor searching of the embedding space using sample language fragments can function like the 'most similar' feature of the static neural language models, although now showing the multiple contexts in which terms appear. Using the ball tree algorithm, the contextual neighbors of key terms of interest can be discovered. These neighbors, derived from a known text or document archive, can be compared to those found in the embedding matrix to interpret the local significance of these texts. As the semantic neighborhood of a single term in a Transformer model has the potential of being quite wide, some strategies are needed to make them more interpretable. One can visualize that neighborhood by using principal component analysis (PCA) to extract the two most meaningful dimensions of

multi-dimensional data and plotting these as x and y coordinates. This approach can be used to produce embeddings for all contexts of a token, with the distances indicating similarity of the multiple uses of this token throughout the modeled text.

After training on a classification task, contextual embeddings for key terms thought to be important to learning how to discriminate between classes of text can be extracted. These can then be compared alongside the language fragment in which they appear in the training set. Such approaches might demonstrate the degree to which a selected key term (as a composite of tokens) has informed the decision-making criteria of the classifier. This approach combined with others might iteratively lead to the mapping of the (contextual) semantic space that makes up key elements of classification criteria. As we learn more about specific models, freezing, disabling, or "knocking out" layers, heads, or even specific "neurons" will enable understanding the contribution of those elements and, more importantly for humanists, the features that they appear to be specialized to recognize (Wang et al., 2022). The disabling, or ablation, of individual and clusters of neurons appears to be a useful strategy, especially on smaller models trained on high-quality and narrowly constructed datasets.

3.5 Feature visualization

In studying computer vision applications, critics might more easily have recourse to the metaphor of seeing in conceptualizing and critiquing how deep learning networks "see." Doing so raises important questions about the representation of input data and efforts to yoke these representations to the input objects. As Fabian Offert and Peter Bell argue, these networks are "biased towards a distributed, entangled, deeply non-human way of representing the world" (Offert & Bell, 2021). In their account, feature visualizations of these networks provide metapictures of that which is otherwise non-interpretable. Transformer networks, as applied to text, have fewer problems for the critic desiring to know how the model reads, as the smallest features are words or multi-character subword units. Nearest-neighbor searches of the embedding space return vectors that can be decoded as tokens, reversing the encoding process that converts input text into vector representations. Feature visualizations, however, have become quite useful for comparing some of the architecture-specific features in these networks, namely for understanding the varied responses from attention heads across the layers of the model.

While the majority of the above approaches focus on learned representations of individual tokens across the layers of a Transformer, attention heads are theorized as one of the greatest contributors to the power of this architecture (Rogers et al., 2020). In typical models, the number of attention heads attached to each layer is equivalent to the total number of layers. The twelve-layer GPT-2 model thus has twelve attention heads for each layer. These heads generate weights as representations of supplied language fragments, for all the tokens contained within that context. Recent models used with the Hugging Face transformers package make these available when the model is loaded with the "output_attentions" argument. These attention heads in Transformer models play an important role and yet they are not well understood. As decision criteria used in classification tasks, the weights of the attention heads are seen as important, but it

is not clear how these multiple heads (one hundred and forty-four in the basic GPT-2 model) differently contribute to learned representations. Visualization in the form of heatmaps of the various attention heads can help identify the relative importance of contextualized tokens at multiple layers. A greater degree of attention from one token to another can be visualized with a heatmap to show the degree to which a particular head has learned a relation between specific tokens. Such a highlighting of neighboring contextual tokens may show which terms are particularly sensitive to specific contexts and attention to specific semantic, lexical, or grammatical features suggests that the attention in question has some degree of specialization for recognizing these features (Rogers et al., 2020). After training for a classification task, and with access to attention data, the attention weights from text samples predicted to belong to a class can be visualized to help understand the contribution of individual tokens to decisions and relations among the tokens.

3.6 Evaluation with another model

Transformer models are regularly fine-tuned for supervised classification tasks. Such tasks might include the identification of certain kinds of statements, like hate speech, or subject/topic categories of text samples. While these complex deep neural networks may produce higher accuracy classifications than earlier methods, learning what specific features (i.e.,tokens) within that segment were influential to that classification decision is a little more complicated, even more so if access to model weights is restricted. Depending upon the transformer architecture and implementation and the methods of access provided to the model, feature weights (in present models, when the "output_hidden_states" argument is used in loading the model) for the contextual tokens may be able to be extracted from multiple layers and information gleaned from these weights or embeddings can help determine the relative significance of key terms. Classifications of a set of supplied input data, in the form of extracted token and fragment embeddings, can be turned into training data for a more interpretable and accessible model that while incapable of explaining the behavior of the black box model, can assist a critic in understanding the nature of these classifications.

As LLM technology develops, it is likely that multiple models based on the same foundation will continue to be released. Due to the high cost of training, open-source models are attractive starting points for building task or domain-specific models. Because of its relatively open licensing and large size, Meta's Llama-2 model has had significant uptake in the ML community (Touvron et al., 2023). Comparing the foundation base to a fine-tuned version or using one model to evaluate another can give some insight into the effects of fine tuning and other "downstream" tasks applied to the models. When models fine-tuned for instruction, the training on pairs of question and answers and the category of task commonly used for creating agents or chatbots, are compared to the based model, the degree to which this fine-tuning has influenced the behavior of the model can be examined.

Output from deep neural networks like Transformers in the form of labeled data from the evaluation dataset can be used as criteria for interpretation by creating a statistical model from the learned features within these data. Statistical models derived

from simple token frequencies rather than embedding values, for example, can provide information about the decision criteria and the distribution of tokens within Transformer-classified data. Such statistical models, which would include classic machine learning classifiers like Support Vector Machines (SVM), k-Nearest Neighbors (kNN), Naïve Bayes (NB) all provide easier access to feature weights and have higher levels of interpretability than deep learning networks. While learning the feature weights, for example with SVM, to discriminate class-labeled text segments from a classifier would produce another model but one that is modeling the output of the first model. Familiar statistical tests of significance can also be used to compare multiple series of classifications and reduce uncertainty and the variation produced from stochastic models. These methods are standard ways of understanding the effects of decisions and investigating possible bias in unknown classification criteria.

There are several other methods for interpreting black box models that can be organized under this category. These methods have the advantage of working with multiple kinds of models with vastly different architectures, from relatively simple linear classifiers to complex deep learning networks. A method using linear classifier probes has been used with deep learning models prior to the development of Transformers. Guillaume Alain and Yoshua Bengio, who have developed this approach, conceptualize their linear probes as "thermometers used to measure the temperature simultaneously at many different locations" (Alain & Bengio, 2018). Fitting such probes to deep networks enabled them to identify feature separability (in images in their initial work in this area) across multiple network layers. One well-known package called Local Interpretable Model-agnostic Explanations (LIME) can help a critic evaluate and understand a complex model by visualizing data representations from the black box in a way that is more understandable yet aligned with the target model's operation (Ribeiro et al., 2016). Feature visualizations from LIME can be projected or overlayed on images (Dobson, 2023) and similar strategies can be used to identify influential features from text-based models. Other model-agnostic tools have been developed to analyze decisions made by classifiers and these can be applied to decisions made by Transformer networks. Similar architecture-specific visualization tools for Transformers include the Learning Interpretability Tool (LIT), TransformerLens, and BertViz, which despite the name suggesting that it works only with BERT, not only supports several different models but also other deep learning architectures.

3.7 Using programmatic features

Given programmatic access to a Transformer model, even in the absence of pre-training data, one key strategy would involve the appropriation of software debugging techniques to insert hooks or triggers into the operation of the model. While private or protected models with restricted APIs may not expose interfaces required for debugging, they are available for a number of reasonably well-described models that are distributed without training datasets or replicable sources. This method might enable closer inspection by recording changes to the network as it is trained, fine-tuned, and deployed. Such instrumentation of neural networks can be facilitated by already existing hooks available in common machine learning frameworks like Pytorch that can

be activated during selected functions, such as feed forward or backpropagation functions. The use of programmatic hooks to examine the training of deep neural networks would be most useful in the construction of new models and when implementing a network from scratch, which provides some measure of access to in-progress learning during training as well as inference activities. That said, programmatic hooks can also be used to instrument pre-trained models, which is to say add capacities for measurements and control. There are also some sources for data provenance found in deep learning packages, such as the autograd history stored in tensors created with Pytorch, that can be used to extract some information about the operation that created a data object and identify its parent sources.

4 Conclusion

There is an important line of research in the philosophy of science and science technology studies that makes an argument against the dictate for critics to open all black boxes. The demand for transparency in algorithmic decision making, in particular, has been put under some pressure for its limitations and its redirection of critical energy toward technical and away from social and political questions (Ananny & Crawford, 2018). Humanities researchers have also produced compelling alternative modes of analysis that can read aspects of black boxes from a distance (Paßmann and Boersma, 2017. Despite these compelling arguments, giving up on tracing neural networks and classifying their operation as incommensurable with human understanding (Fazi, 2021) too readily cedes ground to positivist accounts of these technologies and unfounded claims to modeling reality. In focusing primarily on causal explanations for their behavior, researchers risk ignoring the interpretative nature of these instruments and the limitations in their ability to model language. This essay demonstrates that state-of-the-art deep neural networks are not beyond interpretation by humanists and that there are multiple methods available now to help us understand and criticize their sources, operations, representations, and outputs.

Acknowledgements Akshay I. Kelshiker and Pulkit Nagpal were incredibly helpful in spending a term researching and playing with Transformer models with me. I thank them for their thoughts on these models and comparisons with neural language models. A conversation with Anders Knospe about his research into Transformer-based classifiers at an opportune time contributed greatly to my thinking about methods for investigating Transformers.

Author Contributions J.D. wrote the manuscript text and is responsible for the conceptualization and execution of this research.

Funding Not applicable.

Availability of data and materials Code used to support this research can be found in the following repository: https://github.com/jeddobson/blackbox-transformers

Declarations

Ethical Approval Not applicable.

Competing interests Not applicable.

References

Alain, G., & Bengio, Y. (2018). Understanding intermediate layers using linear classifier probes. arXiv:1610.01644 [cs, stat]

Ananny, M., & Crawford, K. (2018). Seeing without knowing: limitations of the transparency ideal and its application to algorithmic accountability. *New Media & Society, 20*(3), 973–989. https://doi.org/10.1177/1461444816676645

Bandy, J., & Vincent, N. (2021). Addressing 'documentation debt' in machine learning research: a retrospective datasheet for bookcorpus. arXiv:2105.05241 [cs].

Bender, E.M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, Virtual Event Canada, pp. 610–623. ACM.

Berry, D. (2023). The explainability turn. *DHQ: Digital Humanities Quarterly 17*(2)

Birhane, A., Prabhu, V.U., & Kahembwe, E. (2021). Multimodal datasets: misogyny, pornography, and malignant stereotypes. arXiv:2110.01963 [cs].

Broussard, M. (2018). *Artificial Unintelligence: How Computers Misunderstand the World*. Cambridge, MA: MIT Press.

Buolamwini, J. (2023). *Unmasking AI: My Mission to Protect What is Human in a World of Machines*. New York: Random House.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., Oprea, A., & Raffel, C. (2021). Extracting training data from large language models. arXiv:2012.07805 [cs].

Chan, S.C.Y., Dasgupta, I., Kim, J., Kumaran, D., Lampinen, A.K., & Hill, F. (2022). Transformers generalize differently from information stored in context vs in weights. arXiv:2210.05675 [cs].

Chang, K.K., Cramer, M., Soni, S., & Bamman, D. (2023). Speak, memory: an archaeology of books known to ChatGPT/GPT-4. arXiv:2305.00118 [cs].

Crawford, K., & Paglen, T. (2019). Excavating AI: the politics of training sets for machine learning. https://excavating.ai

Denton, E., Hanna, A., Amironesei, R., Smart, A., & Nicole, H. (2021). On the genealogy of machine learning datasets: a critical history of ImageNet. *Big Data & Society, 8*(2), 1–14. https://doi.org/10.1177/20539517211035955

Dobson, J. E. (2019). *Critical Digital Humanities: The Search for a Methodology*. Urbana, IL: University of Illinois Press.

Dobson, J.E. (2021). Interpretable outputs: criteria for machine learning in the humanities. *DHQ 15*(2) .

Dobson, J. E. (2023). Objective vision: confusing the subject of computer vision. *Social Text, 41*(3), 35–55.

Ethayarajh, K. (2019). How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, pp. 55–65. Association for Computational Linguistics.

Fazi, M. B. (2021). Beyond human: deep learning, explainability and representation. *Theory, Culture & Society, 38*(7–8), 55–77. https://doi.org/10.1177/0263276420966386

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., & Leahy, C. (2020). The Pile: An 800GB dataset of diverse text for language modeling. arXiv:2101.00027 [cs].

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. *ACM Computing Surveys, 51*(5), 1–42. https://doi.org/10.1145/3236009

Guillory, J. (2016). Monuments and documents: Panofsky on the object of study in the humanities. *History of Humanities, 1*(1), 9–30. https://doi.org/10.1086/684635

Hugging face. (2022). GPT2. https://huggingface.co/gpt2

Ihde, D. (1998). *Expanding Hermeneutics: Visualism in Science*. Evanston, IL: Northwestern University Press.

Malevé, N. (2021). On the data set's ruins. *AI & Society: Knowledge, Culture and Communication, 36*(4), 1117–1131. https://doi.org/10.1007/s00146-020-01093-w

McCarthy, J. (1974). review of artificial intelligence: a general survey, by James Lighthill. *Artificial Intelligence, 5*, 317–322.

Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2023). Locating and editing factual associations in GPT. arXiv:2202.05262 [cs].

Miller, T. (2019). Explanation in artificial intelligence: insights from the social sciences. *Artificial Intelligence, 267*, 1–38. https://doi.org/10.1016/j.artint.2018.07.007

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., & Gebru, T. (2019). Model cards for model reporting. *Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT* '19*: 220–229. https://doi.org/10.1145/3287560.3287596 .

Offert, F., & Bell, P. (2021). Perceptual bias and technical metapictures: critical machine vision as a humanities challenge. *AI & Society: Knowledge, Culture and Communication, 36*(4), 1133–1144. https://doi.org/10.1007/s00146-020-01058-z

Offert, F., & Phan, T. (2022). A sign that spells: DALL-E 2, invisual images and the racial politics of feature space. arXiv:2211.06323 [cs].

OpenAI. (2019). GPT-2 model card. https://github.com/openai/gpt-2/blob/master/model_card.md

Paßmann, J., & Boersma, A. (2017). Unknowing algorithms on transparency of unopenable black boxes. In M.T. Schäfer, & K. van Es (Eds.) *The Datafied Society* 139–146. Amsterdam: Amsterdam University Press. https://doi.org/10.1515/9789048531011-012.

Reisner, A. (2023). These 183,000 books are fueling the biggest fight in publishing and tech. *The Atlantic*

Rettberg, J. W. (2022). Algorithmic failure as a humanities methodology: machine learning's mispredictions identify rich cases for qualitative analysis. *Big Data & Society, 9*(2), 1–6. https://doi.org/10.1177/20539517221131290

Ribeiro, M.T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA, pp. 1135–1144. ACM.

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A Primer in BERTology: what we know about how BERT works. *Transactions of the Association for Computational Linguistics, 8*, 842–866. https://doi.org/10.1162/tacl_a_00349

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review, 65*(6), 386–408. https://doi.org/10.1037/h0042519

Schleiermacher, F. (1998). *Hermeneutics and Criticism: and Other Writings*. New York: Cambridge University Press.

Seaver, N. (2019). Knowing algorithms. In J. Vertesi & D. Ribes (Eds.), *digitalSTS: A Field Guide for Science & Technology Studies* (pp. 412–422). Princeton, NJ: Princeton University Press.

Shoemaker, T. (2023). Verkettete Textualität Verkettete Textualität [Concatenative Textuality]. In H. Bajohr, & M. Krajewski (Eds.) *Quellcodekritik: Zur Philologie von Algorithmen* Berlin: August Verlag.

Star, S.L. (1989). The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving. In L. Gasser, & M.N. Huhns (Eds.) *Distributed Artificial Intelligence*, 37–54. San Francisco, CA: Morgan Kaufmann. https://doi.org/10.1016/B978-1-55860-092-8.50006-X

Tenney, I., Das, D., & Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 4593–4601. Association for Computational Linguistics.

Tirumala, K., Markosyan, A.H., Zettlemoyer, L., & Aghajanyan, A. (2022). Memorization without overfitting: analyzing the training dynamics of large language models. arXiv:2205.10770 [cs].

Touvron, H., Martin, L., & Stone, K. (2023). Llama 2: Open foundation and fine-tuned chat models. https://ai.meta.com/research/publications/llama-2-open-foundation-and-fine-tuned-chat-models/

van Es, K. (2023). Unpacking tool criticism as practice, in practice. *DHQ: Digital Humanities Quarterly 17*(2) .

van Es, K., Wieringa, M., & Schäfer, M.T. (2018). Tool criticism: from digital methods to digital methodology. In *Proceedings of the 2nd International Conference on Web Studies - WS*.2 2018, Paris, France, pp. 24–27. ACM Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention is all you need*. CA, USA: Long Beach.

Wang, K., Variengien, A., Conmy, A., Shlegeris, B., & Steinhardt, J. (2022). Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. arXiv:2211.00593 [cs].