CEBaB: Estimating the Causal Effects of Real-World Concepts on NLP Model Behavior

Anonymous Author(s)
Affiliation
Address
email

Abstract

The increasing size and complexity of modern ML systems has improved their predictive capabilities but made their behavior harder to explain. Many techniques for model explanation have been developed in response, but we lack clear criteria for assessing these techniques. In this paper, we cast model explanation as the causal inference problem of estimating causal effects of real-world concepts on the output behavior of ML models given actual input data. We introduce CEBaB, a new benchmark dataset for assessing concept-based explanation methods in Natural Language Processing (NLP). CEBaB consists of short restaurant reviews with human-generated counterfactual reviews in which an aspect (food, noise, ambiance, service) of the dining experience was modified. Original and counterfactual reviews are annotated with multiply-validated sentiment ratings at the aspect-level and review-level. The rich structure of CEBaB allows us to go beyond input features to study the effects of abstract, real-world concepts on model behavior. We use CEBaB to compare the quality of a range of concept-based explanation methods covering different assumptions and conceptions of the problem, and we seek to establish natural metrics for comparative assessments of these methods.[1]

1 Introduction

Explaining model behavior has emerged as a central goal within ML. In NLP, models have grown in size and complexity, and while they have become increasingly successful, they have also become more opaque [28,36], raising concerns about trust [18,23], safety [1,34], and fairness [16,19]. These concerns will persist if these models remain "black-boxes".

Seeking to open the black-box, researchers have developed methods that try to explain model behavior [2,11,13,30,41]. However, there is currently no consensus about how to evaluate such methods to allow robust comparisons. This is not surprising, since such evaluations require very rich empirical data. Intuitively, we would like to (1) intervene on model inputs, to modify specific concepts without changing other correlated information, (2) observe the effects this has on model predictions, and finally (3) assess explanation methods for their ability to accurately predict these observations.

The absence of interventional data, or even an agreed-upon non-interventional benchmark, has created an environment in which explanation methods are often evaluated individually, and without comparison to alternatives. Attempts have been made to conduct comparative evaluations [11,17,37], but only with synthetic, simplified datasets. Furthermore, these attempts do not define a unified evaluation approach, nor do they seek to contribute benchmark datasets that support such evaluations.

In this paper, we seek to overcome this obstacle by introducing **CEBaB** (**C**ausal **E**stimation-**Ba**sed **B**enchmark). Table 1 summarizes the structure of CEBaB with a toy example: beginning with a

[1] Our code and data are included in our supplementary materials.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Table 1: Toy examples illustrating the structure of CEBaB (actual corpus examples are longer and more complex; a sample is given in Appendix B). Beginning from an OpenTable review, we give crowdworkers an actual restaurant review and they generate counterfactual restaurant reviews that would have been written if some aspect of the dining experience were changed and all else were held constant. Five different crowdworkers labeled each of the actual and counterfactual texts according to their aspect-level sentiment and overall sentiment. Aspect level sentiment labels are three way: '+' (positive sentiment), '–' (negative), and 'unk' (the aspect's value is not expressed in the text). Overall sentiment labels are 1 (worst) to 5 (best). Edited aspect labels are shown in blue.

			food	ambiance	service	noise	overall
Original text:		Excellent lobster and decor, but rude waiter.	+	+	–	unk	4
Goal							
food edit:	–	Terrible lobster, excellent decor, but rude waiter.	–	+	–	unk	2
food edit:	unk	Excellent decor, but rude waiter.	unk	+	–	unk	3
ambiance edit:	–	Excellent lobster, but lousy decor and rude waiter.	+	–	–	unk	3
ambiance edit:	unk	Excellent lobster, but rude waiter.	+	unk	–	unk	3
service edit:	+	Excellent lobster and decor, and friendly waiter.	+	+	+	unk	5
service edit:	unk	Excellent lobster and decor.	+	+	unk	unk	5
noise edit:	+	Excellent lobster, decor, and music, but rude waiter.	+	+	–	+	4
noise edit:	–	Excellent lobster and decor, but rude waiter, and noisy.	+	+	–	–	3

review text from the OpenTable website, we crowdsourced edits of the original text that are designed to meet a specific goal, such as changing the food rating in the original text to negative or unknown. All of the resulting edits were validated by five crowdworkers and each full text was evaluated by five crowdworkers for its overall sentiment. CEBaB is grounded in 2,299 original reviews, which were expanded via this editing procedure to a total of 15,089 texts, targeting four different aspect-level concepts (food, service, ambiance, noise) with three potential labels (positive, negative, and unknown, i.e., not expressed in the review), and each full text was labeled on a five-star scale.

We focus on using CEBaB to compare concept-based explanation methods. This allows us to go beyond the effect of individual words to study how more abstract concepts (in our case, the aspect-level evaluations) contribute to model predictions (about the overall sentiment of the text). Our proposed metrics center around assessing concept-based explanation methods for their ability to accurately estimate *causal concept effects* [17], allowing us to isolate the effect of individual concepts.

More specifically, we use CEBaB to measure the causal effects of particular variables in a causal graph, and we cast each explanation method as a causal estimator of these measurements. For example, suppose our causal graph of the data says that all four of our aspect-level categories will affect a reviewer's overall rating. To estimate the effect of positive food quality on the predicted overall rating from a classifier, we need to compare examples with high food quality to those with low quality, holding all other aspects constant. Such pairs of examples are normally not observed, but this is precisely what CEBaB provides. With CEBaB, we can directly compare the actual change in model predictions with the change that a concept-based explanation method predicts.

In our experiments, we evaluate five leading concept-based explanation methods: CONEXP [17], TCAV [26], ConceptSHAP [56], INLP [40], and CausaLM [11]. We also propose an estimation technique inspired by *S-Learner* [27]. These methods make a wide range of different assumptions about how much access we have to the model's internal structure, and they also diverge in the degree to which they account for the causal nature of the concept effect estimation problem. CEBaB reveals substantive differences between them. However, our primary contribution lies in providing data and metrics that can support very rich comparisons of this sort.

2 Previous Work

Explanation Methods and Causality Probing is a relatively new technique for understanding what model internal representations encode. In probing, a small supervised [6,51] or unsupervised [5,32,44] model is used to estimate whether specific concepts are encoded at specific places in a network. While probes have helped illuminate what models (especially pretrained ones) have learned from data, Geiger et al. [15] show with simple analytic examples that probes cannot reliably provide causal explanations for model behavior.

Feature importance methods can also be seen as explanation methods [33]. Many methods in this space are restricted to input features, but gradient-based methods can often quantify the relative importance of hidden states as well [3,46,48,57]. The Integrated Gradients method of Sundararajan et al. [50] has a natural causal interpretation stemming from its exploration of baseline (counterfactual) inputs [15]. However, even where these methods can focus on internal states, it remains difficult to connect their analyses with real-world concepts that do not reduce to simple properties of inputs.

Intervention-based methods involve modifying inputs or internal representations and studying the effects that this has on model behavior [30, 41]. Recent methods perturb input or hidden representations to create counterfactual states that can then be used to estimate causal effects [9, 12, 47, 52]. However, these methods are prone to generating implausible inputs or network states unless the interventions are carefully controlled [14].

Generating counterfactual texts automatically remains challenging and is still a work-in-progress [4]. To overcome this problem, another class of approaches proposes to manipulate the representation of the text with respect to some concept, rather than the text itself [9,11,40]. These methods fall into the category of concept-based explanations and we discuss two of them extensively in §3.

In this work, we focus on concept-based explanation methods [26]. This also allows us to go beyond the effect of individual words to potentially model the effects of high-level concepts that are expressed in complex ways in language. For instance, in CEBaB, changing the service evaluation from unknown to positive in a given review text may involve a wide range of changes to the text that would not be easy to model as token-level manipulations. The concept-based methods we assess here offer the promise of being able to estimate the causal effects of such complex manipulations.

Benchmarks for Explanation Methods Benchmark datasets have propelled ML forward by creating shared metrics that predictive models can be evaluated on [22,25,53,54]. Unfortunately, benchmarks that are suitable for assessing the quality of model explanations are still uncommon [10, 21]. Previous work on comparing explanation methods has generally only correlated the performance of a given explainability method with others, without ground-truth comparisons [8,20,21,43]. Existing benchmarks that do compare to some ground-truth [11,56] either use synthetic data or are tailored for a specific explanation method. Our benchmark is naturalistic and general purpose.

3 Estimating Concept Effects with CEBaB

We now define the core metrics that we use to evaluate different explanation methods. Figure 1 provides a high-level view of the causal process we are envisioning. The process begins with an exogenous variable U representing a state of the world. For CEBaB, we can imagine that the value of U is a state of affairs u of a person evaluating a restaurant in a particular way. u contributes to a review variable X, with the value x of X mediated by u and by mediating concepts $C_1, \ldots C_k$, which correspond to the four aspect-level categories in CEBaB (food, service, ambiance, and noise), each of which can have values $c \in \{\text{positive}, \text{negative}, \text{unknown}\}$. The review x is processed by a model that outputs a vector of scores over classes (sentiment labels in CEBaB).

Core Metric Our central goal is to use CEBaB to evaluate explanation methods themselves. CEBaB supports many approaches to such evaluation. In this paper, we adopt an approach based on individual-level rather than average effects. This makes very rich use of the counterfactual text and associated labels provided by CEBaB. The starting point for this metric is the Individual Causal Concept Effect:

Definition 1 (Individual Causal Concept Effect; ICaCE)**.** *For a neural network $\mathcal{N}$ and feature function ϕ, the individual causal concept effect of changing the value of concept C from c to c' for*

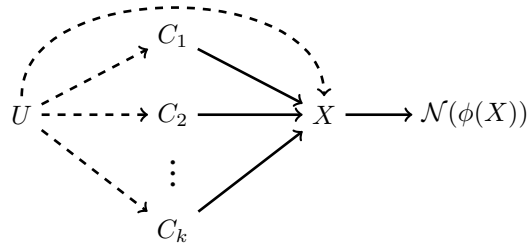

Figure 1: A causal graph describing a data generating process with an exogenous variable U representing the state of the world, mediating concepts $C_1, C_2 \ldots, C_k$, and data X that is featurized with ϕ. $\phi(X)$ is input to a classifier $\mathcal{N}$, which outputs a vector of scores over m output classes.

state of affairs u in an underlying data generation process $\mathcal{G}$ is

$$\textit{ICaCE}_{\mathcal{N}_\phi}(\mathcal{G}, x_u^{C=c}, c') = \mathbb{E}_{x\sim\mathcal{G}}\left[\mathcal{N}\big(\phi(x)\big) \mid do\begin{pmatrix} C = c' \\ U = u \end{pmatrix}\right] - \mathcal{N}\big(\phi(x_u^{C=c})\big). \tag{1}$$

ICaCE is a theoretical quantity. In practice, we use the Empirical Individual Causal Concept Effect.

Definition 2 (Empirical Individual Causal Concept Effect; $\widehat{\text{ICaCE}}$)**.** *For a neural network $\mathcal{N}$ and feature function ϕ, the empirical individual causal concept effect of changing the value of concept C from c to c' for state of affairs u is*

$$\widehat{\textit{ICaCE}}_{\mathcal{N}_\phi}(x_u^{C=c}, x_u^{C=c'}) = \mathcal{N}\big(\phi(x_u^{C=c'})\big) - \mathcal{N}\big(\phi(x_u^{C=c})\big), \tag{2}$$

where $(x_u^{C=c}, x_u^{C=c'})$ is a tuple of inputs originating from u with the concept C set to the values c and c', respectively.

The $\widehat{\text{ICaCE}}_{\mathcal{N}_\phi}$ for a pair of examples $(x_u^{C=c}, x_u^{C=c'})$ is simply the difference between the output score vectors for the two cases. With CEBaB, we can easily calculate these values because we have clusters of examples that are tied to the same reviewing situation u and express different concept values.

For assessing an explanation method $\mathcal{E}$, we compare ICaCE values with those returned by $\mathcal{E}$. Our core metric is the ICaCE-Error:

Definition 3 (ICaCE-Error)**.** *For a neural network $\mathcal{N}$ and feature function ϕ, the ICaCE-Error of an explanation method $\mathcal{E}$ for changing the value of concept C from c to c' is:*

$$\textit{ICaCE-Error}^{\mathcal{D}}_{\mathcal{N}_\phi}(\mathcal{E}) = \frac{1}{|\mathcal{D}|}\sum_{(x_u^{C=c}, x_u^{C=c'})\in\mathcal{D}} \mathsf{Dist}\big(\widehat{\textit{ICaCE}}_{\mathcal{N}_\phi}(x_u^{C=c}, x_u^{C=c'}), \mathcal{E}_{\mathcal{N}_\phi}(x_u^{C=c}, c')\big) \tag{3}$$

where Dist *is a distance metric.*

We use cosine distance as the distance metric, evaluating the ability of the explanation methods to predict the direction of an intervention's causal effect on model behavior. This distance metric is invariant to the scale of the effects. In Appendix D we report results with a metric sensitive to scale as well.

In our analyses, instead of averaging over the entire dataset, we often report the ICaCE-Error for subsets $\mathcal{D}_C^{c\to c'}$, the subsets of interventions where the value of concept C is changed from c to c'.

Aggregating Individual Causal Concept Effect It is often useful to also have a direct estimate of a model's ability to capture concept-level causal effects. For this, we employ an aggregating version of $\widehat{\text{ICaCE}}$, the Empirical Causal Concept Effect:

Definition 4 (Empirical Causal Concept Effect; $\widehat{\text{CaCE}}$)**.** *For a neural network $\mathcal{N}$ and feature function ϕ, the empirical causal concept effect of changing the value of concept C from c to c' in dataset $\mathcal{D}$ is*

$$\widehat{\textit{CaCE}}^{\mathcal{D}}_{\mathcal{N}_\phi}(C, c, c') = \frac{1}{|\mathcal{D}_C^{c\to c'}|}\sum_{(x_u^{C=c}, x_u^{C=c'})\in\mathcal{D}_C^{c\to c'}} \widehat{\textit{ICaCE}}_{\mathcal{N}_\phi}(x_u^{C=c}, x_u^{C=c'}). \tag{4}$$

Table 2: The evaluated explanation methods and their attributes. **Explainer Method** denotes the complexity of the models used by each explanation method. **Access to Explained Model** denotes the degree of access an explainer method needs to the explained model. **Concept Labels Needed** indicates whether a method estimating the effect for an input $x_u^{C=c}$ needs the actual input label c and/or the intervened value c' at test time. Models with a **Counterfactual Representation** approximate $\phi(x_u^{C=c'})$ to estimate the effect. Finally, only CausaLM and S-Learner have **Confounder Control** to minimize the impact of confounding concepts. *We predict these labels with a classifier.

Explanation method	Explainer Method	Access to Explained Model	Concept Labels Needed (test time)	Counterfactual Representation	Confounder Control
CONEXP [17]	None	None	c and c'	✗	✗
S-Learner [27]	Linear	None	All concepts and their labels*	✗	✓
TCAV [26]	Linear	Weights	None	✗	✗
ConceptSHAP [56]	Linear	Weights	None	✗	✗
INLP [40]	Linear	Weights	None	✓	✗
CausaLM [11]	Explained Model	Training Regime	None	✓	✓

This is an empirical estimator of the Causal Concept Effect (CaCE) of Goyal et al. [17]. It estimates, in general, how the classifier predictions change for a given concept and intervention direction.

Estimating Real-World Causal Effect of Aspect Sentiment on Overall Sentiment We can also estimate ground truth causal effects in CEBaB by simply using its labels directly. There are again a variety of ways that this could be done. We opt for the one that makes the richest use of the structures afforded by CEBaB. For perspicuity, in parallel to the neural network-based $\widehat{\text{ICaCE}}$ (Definition 2), we define the Empirical Individual Treatment Effect for our dataset:

Definition 5 (Empirical Individual Treatment Effects in CEBaB; $\widehat{\text{ITE}}$)**.** *The empirical individual treatment effect of changing the value of concept C from c to c' in CEBaB is*

$$\widehat{ITE}^{CEBaB}(x_u^{C=c}, x_u^{C=c'}) = f(x_u^{C=c'}) - f(x_u^{C=c}) \tag{5}$$

where f is a simple look-up procedure that retrieves the overall sentiment labels for CEBaB examples.

We aggregate over these values by taking their average, in parallel to what we do for network predictions (Definition 4). This yields the Empirical Average Treatment Effect ($\widehat{\text{ATE}}$) for CEBaB.

Alternative Metrics In Appendix A in our supplementary materials, we consider alternative formulations of the core metrics with *causal concept effects* and *absolute causal concept effects*, relating them to the different questions they engage with. We opt for the individual causal concept effect in our central metric (Definition 3), taking the central question to be what caused an ML model to produce an output for an *actual* input created from a real-world process.

4 Evaluated Explanation Methods

We compare several model explanation methods that share three main characteristics. First, they are all suitable for NLP models and have been used in the literature for generating model explanations in the form of estimated effects on model predictions. Second, they all provide concept-level explanations, for a pre-defined list of human-interpretable concepts (e.g., how sensitive a restaurant review rating classifier is to language related to food quality). This approach is also forward-looking, allowing more researchers to construct new hypotheses (i.e., concepts we have not collected labels for) and estimate their effect on the predictor. Third, all of the tested methods are model-agnostic, meaning that they separate the explanation from the model. At the same time, these methods differ in five important ways, as summarized Table 2.

We now turn to reviewing the explanation methods that we later compare on CEBaB (§6). In our mathematical formulas, we employ a unified notation for all methods, to make the definitions more accessible and easier to integrate into our experimental set-up. Assume we have a classifier $\mathcal{N}$ (which outputs a probability vector) and feature function ϕ, and we want to compute the effect on $\mathcal{N}_\phi(x_u^{C=c})$ of changing the value of concept C from c to c' using an unseen test set $(\mathcal{D}, Y)$.

Conditional Expectation (CONEXP) Goyal et al. [17] propose a baseline where the effect of a concept C is the average difference in predictions on examples with different values of C.

$$\text{CONEXP}^{\mathcal{D}}_{\mathcal{N}_\phi}(C, c, c') = \frac{1}{|\mathcal{D}^{C=c'}|} \sum_{x \in \mathcal{D}^{C=c'}} \mathcal{N}(\phi(x)) - \frac{1}{|\mathcal{D}^{C=c}|} \sum_{x \in \mathcal{D}^{C=c}} \mathcal{N}(\phi(x)) \tag{6}$$

where $\mathcal{D}^{C=c}$ and $\mathcal{D}^{C=c'}$ are subsets of $\mathcal{D}$ where C takes values c and c', respectively.

To predict an effect, this method only relies on C, c, and c', resulting in an estimate that does not depend on the specific input text itself.

Conditional Expectation Learner (S-Learner) We adapt *S-Learner*, a popular method for estimating the Conditional Average Treatment Effect (CATE) proposed by Künzel et al. [27]. To estimate causal concept effects, our *S-Learner* trains a logistic regression model $\mathcal{E}_x$ to predict $\mathcal{N}(\phi(x))$ using the values of all the labeled concepts of example x, denoted by x'.[2] Then, during inference, we compute an individual effect for example pair $(x_u^{C=c}, x_u^{C=c'})$ by comparing the output of the model $\mathcal{E}_x$ on this pair:

$$\textit{S-Learner}(C, c, c'; x) = \mathcal{E}_x({x'}_u^{C=c'}) - \mathcal{E}_x({x'}_u^{C=c}) \tag{7}$$

At inference time, S-Learner assumes access to all aspect-level labels x', which might not always be available. To alleviate this issue, we instead *predict* the aspect-level labels x' from the original text x using models described in Appendix C.

TCAV Kim et al. [26] use *Concept Activation Vectors* (CAVs), which are semantically meaningful directions in the embedding space of ϕ. Our adapted version of Testing with CAVs (TCAV) outputs a vector measuring the sensitivity of each output class k to changes towards the direction of a concept v_C at the point of the embedded input. It is computed as:

$$\text{TCAV}_{\mathcal{N}_\phi}(C; x) = \big(\nabla \mathcal{N}_k(\phi(x)) \cdot v_C\big)_{k=1}^{K} \tag{8}$$

where K is the number of classes and v_C is a linear separator learned to separate concept C in the embedding space of ϕ. A key limitation of TCAV is that, when the classifier $\mathcal{N}$ is linear, it yields the same score for each concept, regardless of the input.

ConceptSHAP Yeh et al. [56] propose this expansion to SHAP [30], to generate concept-based explanation based on Shapley values [45]. Given a *complete* (i.e., such that the accuracy it achieves on a test set is higher than some threshold β) set of m concepts $\{C_1, \ldots, C_m\}$, ConceptSHAP calculates the contribution of each concept to the final prediction. Our adapted version outputs a vector for each $C \in \{C_1, \ldots, C_m\}$ and x according to the following equation:

$$\text{ConceptSHAP}_{\mathcal{N}_\phi}(C; x) = \sum_{S \subseteq \{C_1, \ldots, C_m\} \setminus C} \frac{(m - |S| - 1)!\,|S|!}{m!} \left[\eta(S \cup \{C\}) - \eta(S)\right], \tag{9}$$

where η is a function defined as $\eta_{\mathcal{N}_\phi}(S) = \sup_g \mathcal{N}\big(g\big(V_S\, \phi(x)\big)\big)$, and V_S is a matrix with the learned concept directions as its rows $V_S = \big(v_C^T\big)_{C \in S} \in \mathbb{R}^{|S| \times h}$.

Yeh et al. [56] calculate concept directions v_{C_j} automatically by learning a neural network classifier. To allow for a fair comparison between ConceptSHAP and the other evaluated methods, we use the concept activation vectors $v_{C_1}, \ldots, v_{C_m}$ as the input concepts (similarly to those used in Kim et al. [26]). We justify this modification and provide implementation details in Appendix G.

CausaLM Feder et al. [11] estimate the causal effect of a binary concept C on the model's predictions by adding auxiliary adversarial tasks to the language representation model in order to learn a counterfactual representation $\phi_C^{\text{CF}}(x)$, while keeping essential information about potential confounders (control concepts). Their method outputs the text representation-based individual treatment effect (TReITE), which is computed as:

$$\text{TReITE}_{\mathcal{N}_\phi}(C; x) = \mathcal{N}'\big(\phi_C^{\text{CF}}(x)\big) - \mathcal{N}\big(\phi(x)\big) \tag{10}$$

[2] This training approach, where an explainer model is fit to predict the output of the original model, shares the intuition of LIME, the widely used explanation method [41], but for concept-level effects.

Table 3: Dataset statistics of CEBaB combining train/dev/test splits.

	Positive	Negative	Unknown	no maj.	Total
food	5726 (41%)	5526 (38%)	2605 (15%)	208 (31%)	14065
service	4045 (29%)	4098 (28%)	3877 (22%)	178 (27%)	12198
ambiance	2928 (21%)	2597 (18%)	5121 (29%)	203 (30%)	10849
noise	1365 (10%)	2215 (15%)	5883 (34%)	78 (12%)	9541

(a) Aspect-level labels.

1 star	1870	(12%)
2 star	3056	(20%)
3 star	3517	(23%)
4 star	2035	(13%)
5 star	2732	(18%)
no maj.	1879	(12%)

(b) Review-level ratings.

	{Neg, Pos}	{Neg, Unk}	{Pos, Unk}
food	898	1316	1291
service	851	857	938
ambiance	947	585	472
noise	1145	208	260

(c) Edit pair distribution. Edit pairs are examples that come from the same original source text and differ only in their rating for a particular aspect.

	Neg to Pos	Neg to Unk	Pos to Unk
food	0.77	0.49	−0.41
service	0.25	0.20	−0.16
ambiance	0.14	0.18	−0.14
noise	0.08	0.04	−0.14

(d) Empirical ATE for binary classification. Reversing concept order results in the negation of the value given. See Appendix B for the corresponding value for the 5-way sentiment labels.

where ϕ_C^{CF} denotes the learned counterfactual representation, where the information about concept C is not present, and $\mathcal{N}'$ is a classifier trained on this counterfactual representation. A key feature of CausaLM is its ability to control for confounding concepts (if modeled).[3] An inherent drawback of this technique is that it can only estimate interventions well for $c' =$ Unknown, since the counterfactual representation is only trained to *remove* a concept C.

Iterative Nullspace Projection (INLP) Ravfogel et al. [40] remove a concept from a representation vector by repeatedly training linear classifiers that aim to predict that attribute from the representations and projecting the learned representations on their null-space. Similar to CausaLM, INLP also estimates the TReATE (Equation 10) and can only estimate interventions for $c' =$ Unknown.

5 The CEBaB Dataset

Table 1 provides an intuitive overview of the structure of CEBaB. In the *editing* phase of dataset creation, crowdworkers modified an existing OpenTable review in an effort to achieve a specific aspect-level goal while holding all other properties of the original text constant. Our aspect-level categories are food, ambiance, service, and noise. In the *validation* phrase, crowdworkers labeled each example relative to each aspect as 'Positive', 'Negative', or 'Can't tell' (Unknown). Having five labels per example allows us to infer a majority label or reason in terms of the full label distributions. In the *rating* phase, each full text was labeled using a common five-star scale, again by five crowdworkers.

We began with 2,299 original reviews from OpenTable (related to 1,084 restaurants) and expanded them, via the above editing procedure, into a total of 15,089 texts. The distribution of normalized edit distances has peaks around 0.28 and 0.77, showing that workers made non-trivial changes to the originals, and even often had to make substantial changes to achieve the editing goal. (See Appendix B for the full distribution.)

Table 3 summarizes the resulting label distributions, where an example has label y if at least 3 of the 5 labelers chose y, otherwise it is in the 'no majority' category. 99% of aspect-level edits have a majority label that corresponds to the editing goal, and 88% of the texts have a review-level majority label on the five-star scale. Overall, these percentages show that workers were extremely successful in achieving their editing goals and that edits have systematic effects on overall sentiment.

[3] As in Feder et al. [11], we control for the most correlated potential confounder.

The central goal of CEBaB is to create *edit pairs*: pairs of examples that come from the same original text and differ only in their labels for a particular aspect. For example, in Table 1, the first two 'food edit' cases form an edit pair, since they come from the same original text and differ only in their food label. Original texts can also contribute to edit pairs; the original text in Table 1 forms an edit pair with each of the texts it is related to by edits. Table 3c summarizes the distribution of edit pairs, and Table 3d reports the ground-truth $\widehat{\text{ATE}}$ values (§3).

We release the dataset with fixed train/dev/test splits. In creating these splits, we enforce two high-level constraints. The first is our 'grouped' requirement: for each original review t, all texts that are related to t via editing occur in the same split as t. This ensures that models are not evaluated on examples that are related by editing to those they have seen in training. Second, if any text t in a group received a 'no majority' label, then the entire group containing t is put in the train set. This ensures that there is no ambiguity about how to evaluate models on dev and test examples.

Once these high-level conditions were imposed, the examples were sampled randomly to create the splits. This allows that individual workers can contribute edited texts across splits. This minor compromise was necessary to ensure that we could have large dev and test splits. Appendix C in our supplementary materials shows that worker identity has negligible predictive power.

There are two versions of the train set: *inclusive* and *exclusive*. The inclusive train set contains all original and edited non-dev/test texts (11,728 texts). The exclusive version samples exactly one train text from each set of texts that are related by editing (1,755 examples). The rationale is that models trained with an original review as well as its edited counterparts may explicitly learn causal effects trivially by aggregating learning signals across inputs. Our exclusive train split prevents this, which helps facilitate fair comparisons between explanation methods and better resembles a real-world setting.

Our dataset is released publicly in JSON format and is available in the Hugging Face `datasets` library. It includes restaurant metadata, full rating distributions, and anonymized worker ids. Appendix B in our supplementary materials provides additional details on the dataset construction, including the prompts used by the crowdworkers, the number of workers per task, worker compensation, and a sample of examples with ratings to help convey the nature of workers' edits and the overall quality of the resulting texts and labels. In addition, Appendix C reports on a wide range of classifier experiments at the aspect-level and text-level that show that models perform well on CEBaB classification tasks, which bolsters the claim that CEBaB is a reliable tool for assessing explanation methods.

6 Experiments and Results

For each experiment, we fine-tune a pretrained language model to predict the overall binary sentiment of all restaurant reviews from our *exclusive* OpenTable train set. Since the goal of our work is not to achieve state-of-the-art performance, but rather to compare explanation methods and demonstrate the usage of CEBaB, we test the ability to explain commonly used models, trained with the accepted experimental configurations.

In the main text, we report results for `bert-base-uncased` fine-tuned as a binary classifier. Our supplementary materials include results for `GPT-2`, `RoBERTa`, and `T5`, fine-tuned on binary, 3-way and 5-way versions of the sentiment task. All results, including the ground-truth effect that depends on the specific instance of a model, are averaged across 5 seeds.

To evaluate the intrinsic capacity of a model to capture causal effects, we report the $\widehat{\text{CaCE}}$ values, as in Definition 4. The results for `bert-base-uncased` are given in Table 4. They are intuitive and well-aligned with the $\widehat{\text{ATE}}$ estimates in Table 3d.

Recall from Table 2 that our compared explanation methods require different access to manually generated aspect labels at test time (the *concept labels needed (test time)* column of the table). Hence, the following methods are directly comparable: CONEXP and S-Learner (both require test-time access to the intervention label before and after intervention), and TCAV, ConceptSHAP, CausaLM and INLP (they require no test-time access to concept labels).

Our primary assessment of the evaluation methods is given in Table 5, again focusing for now on a binary `bert-base-uncased` model as representative of our results. The best (lowest) scores are in bold. When focusing on cosine as the distance metric, results clearly favor S-Learner and CONEXP,

Table 4: $\widehat{\text{CaCE}}$ (Definition 4) for `bert-base-uncased` fine-tuned as a binary sentiment classifier. We report values for the Positive class (the Negative is the negative of the Positive). Results are averaged over 5 distinct seeds with standard deviation. The $\widehat{\text{CaCE}}$ for the reverse of a given concept change is the negative of the value given.

	Negative to Positive	Negative to unknown	Positive to unknown
food	0.75 (± 0.01)	0.27 (± 0.01)	−0.20 (± 0.01)
service	0.29 (± 0.01)	0.10 (± 0.01)	−0.10 (± 0.00)
ambiance	0.24 (± 0.02)	0.09 (± 0.01)	−0.07 (± 0.01)
noise	0.09 (± 0.01)	0.02 (± 0.00)	−0.03 (± 0.01)

Table 5: ICaCE-Error (Definition 3) for `bert-base-uncased` fine-tuned as a binary sentiment classifier, averaged per aspect. Scores are in the $[0, 2]$ range, with 1 being random chance. **Lower is better**. Results averaged over 5 distinct seeds; standard deviations in parentheses. †RandomExplainer takes the difference between two random probability vectors as the predicted effect.

	ambiance	food	noise	service
RandomExplainer†	0.99 (± 0.01)	0.98 (± 0.04)	0.99 (± 0.02)	1.00 (± 0.02)
CONEXP	**0.68** (± 0.05)	**0.45** (± 0.04)	**0.82** (± 0.07)	**0.70** (± 0.04)
S-Learner	0.70 (± 0.05)	0.49 (± 0.03)	**0.82** (± 0.07)	**0.70** (± 0.03)
ConceptShap	1.12 (± 0.03)	1.39 (± 0.05)	1.04 (± 0.01)	1.16 (± 0.01)
TCAV	1.01 (± 0.00)	0.98 (± 0.03)	1.00 (± 0.00)	1.00 (± 0.01)
CausaLM	0.86 (± 0.01)	0.59 (± 0.01)	0.96 (± 0.00)	0.82 (± 0.01)
INLP	0.88 (± 0.01)	0.59 (± 0.01)	0.95 (± 0.00)	0.83 (± 0.00)

indicating that these models are best at estimating the direction of the intervention effect. CausaLM and INLP behave similarly, which is not surprising because both estimate effects by using the presence or absence of concepts. Because of this, these methods cannot estimate all types of interventions and perform worse. TCAV performs on par with a random baseline, and ConceptSHAP performs even worse. A detailed breakdown of the ICaCE-Error, for binary, 3-way and 5-way sentiment analysis, is given in our supplementary materials.

Our cosine-base metric captures whether the direction of the explanation effect is correct but is indifferent to its magnitude. This is useful given that some of our methods inherently return smaller values and so might have unfair advantages when estimating tiny effects, without actually being able to predict their direction. In Appendix D, we report results with distance metrics that are sensitive to the magnitude (e.g., the L2 norm). In these comparisons, S-Learner emerges as strictly better than CONEXP in all aspect categories, and CausaLM emerges as strictly better than INLP. The core difference is that only S-Learner and CausaLM control for confounds (Table 2). This further emphasizes the importance of controlling for confounders.

7 Conclusion

Our main contributions in this paper are twofold. First, we introduced CEBaB, the first benchmark dataset to support comparing different explanation methods against a single ground-truth with human-created counterfactual texts and multiply-validated labels for concepts and overall sentiment. Using this resource, one can isolate the true causal concept effect of any trained classifier. Second, we have provided an in-depth experimental analysis of how well multiple model explanation methods are able to capture the true concept effect. While these methods differ in their assumptions about their access to the model, their computational demands, their access to ground-truth concept labels at inference time, and their overall conception of the explanation problem, they can all be compared using CEBaB. Furthermore, these methods make relatively minimal use of CEBaB's counterfactual train set. In turn, we hope that CEBaB will facilitate the development of explanation methods that can take advantage of the very rich counterfactual structure CEBaB provides across all its splits.

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety. *arXiv:1606.06565 [cs]*, 2016. URL http://arxiv.org/abs/1606.06565.

[2] Jasmijn Bastings, Yonatan Belinkov, Emmanuel Dupoux, Mario Giulianelli, Dieuwke Hupkes, Yuval Pinter, and Hassan Sajjad. Proceedings of the fourth BlackboxNLP workshop on analyzing and interpreting neural networks for NLP. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2021.

[3] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*, pages 63–71. Springer, 2016.

[4] Nitay Calderon, Eyal Ben-David, Amir Feder, and Roi Reichart. Docogen: Domain counterfactual generation for low resource domain adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7727–7746, 2022.

[5] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? An analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy, August 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W19-4828.

[6] Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1198. URL https://www.aclweb.org/anthology/P18-1198.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

[8] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.408. URL https://aclanthology.org/2020.acl-main.408.

[9] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 2021.

[10] Amir Feder, Katherine A Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E Roberts, et al. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *arXiv preprint arXiv:2109.00725*, 2021.

[11] Amir Feder, Nadav Oved, Uri Shalit, and Roi Reichart. CausaLM: Causal model explanation through counterfactual language models. *Computational Linguistics*, 47(2):333–386, June 2021. doi: 10.1162/coli_a_00404. URL https://aclanthology.org/2021.cl-2.13.

[12] Matthew Finlayson, Aaron Mueller, Sebastian Gehrmann, Stuart Shieber, Tal Linzen, and Yonatan Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. In *Proceedings of the 59th Annual Meeting of the Association for*

Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1828–1843, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.144. URL https://aclanthology.org/2021.acl-long.144.

[13] Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. Causal Mediation Analysis for Interpreting Neural NLP: The Case of Gender Bias. *arXiv:2004.12265 [cs]*, November 2020. URL http://arxiv.org/abs/2004.12265. arXiv: 2004.12265.

[14] Atticus Geiger, Kyle Richardson, and Christopher Potts. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.16. URL https://aclanthology.org/2020.blackboxnlp-1.16.

[15] Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

[16] Bryce Goodman and Seth Flaxman. European Union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, October 2017. ISSN 2371-9621, 0738-4602. doi: 10.1609/aimag.v38i3.2741. URL http://arxiv.org/abs/1606.08813. arXiv: 1606.08813.

[17] Yash Goyal, Amir Feder, Uri Shalit, and Been Kim. Explaining Classifiers with Causal Concept Effect (CaCE). *arXiv:1907.07165 [cs, stat]*, February 2020. URL http://arxiv.org/abs/1907.07165. arXiv: 1907.07165.

[18] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

[19] Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. Equality of Opportunity in Supervised Learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/9d2682367c3935defcb1f9e247a97c0d-Paper.pdf.

[20] Peter Hase and Mohit Bansal. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? *arXiv preprint arXiv:2005.01831*, 2020.

[21] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[22] Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR, 2020.

[23] Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.386. URL https://aclanthology.org/2020.acl-main.386.

[24] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

[25] Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts,

and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online, June 2021. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2021.naacl-main.324.

[26] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In *International Conference on Machine Learning*, pages 2668–2677. PMLR, July 2018. URL http://proceedings.mlr.press/v80/kim18d.html. ISSN: 2640-3498.

[27] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences*, 116(10):4156–4165, 2019.

[28] Zachary C. Lipton. The mythos of model interpretability. *Communications of the ACM*, 61(10):36–43, September 2018. ISSN 0001-0782. doi: 10.1145/3233231. URL https://doi.org/10.1145/3233231.

[29] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*, July 2019. URL http://arxiv.org/abs/1907.11692. arXiv: 1907.11692.

[30] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 4768–4777, Red Hook, NY, USA, December 2017. Curran Associates Inc. ISBN 978-1-5108-6096-4.

[31] Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015.

[32] Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054, 2020. ISSN 0027-8424. doi: 10.1073/pnas.1907367117. URL https://www.pnas.org/content/117/48/30046.

[33] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[34] Clemens Otte. Safe and Interpretable Machine Learning: A Methodological Review. In Christian Moewes and Andreas Nürnberger, editors, *Computational Intelligence in Intelligent Data Analysis*, Studies in Computational Intelligence, pages 111–122, Berlin, Heidelberg, 2013. Springer. ISBN 978-3-642-32378-2. doi: 10.1007/978-3-642-32378-2_8.

[35] Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

[36] Judea Pearl. The limitations of opaque learning machines. *Possible minds: twenty-five ways of looking at AI*, pages 13–19, 2019.

[37] Danish Pruthi, Rachit Bansal, Bhuwan Dhingra, Livio Baldini Soares, Michael Collins, Zachary C Lipton, Graham Neubig, and William W Cohen. Evaluating explanations: How much do explanations from the teacher aid students? *Transactions of the Association for Computational Linguistics*, 10:359–375, 2022.

[38] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[39] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

[40] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. *arXiv:2004.07667 [cs]*, April 2020. URL http://arxiv.org/abs/2004.07667. arXiv: 2004.07667.

[41] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "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*, pages 1135–1144, San Francisco California USA, August 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939778. URL https://dl.acm.org/doi/10.1145/2939672.2939778.

[42] D. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, 66:688–701, 1974.

[43] Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J Anders, and Klaus-Robert Müller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021.

[44] Naomi Saphra and Adam Lopez. Understanding learning dynamics of language models with SVCCA. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3257–3267, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1329. URL https://www.aclweb.org/anthology/N19-1329.

[45] Lloyd Shapley. A value for n-person games. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games II*, pages 307–317. Princeton University Press, 1953.

[46] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMLR, 2017.

[47] Paul Soulos, R. Thomas McCoy, Tal Linzen, and Paul Smolensky. Discovering the compositional structure of vector representations with role learning networks. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 238–254, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.23. URL https://aclanthology.org/2020.blackboxnlp-1.23.

[48] Jost Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, 12 2014.

[49] Chi Sun, Luyao Huang, and Xipeng Qiu. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *arXiv preprint arXiv:1903.09588*, 2019.

[50] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org, 2017.

[51] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1452.

[52] Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. Investigating gender bias in language models using causal mediation analysis. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/92650b2e92217715fe312e6fa7b90d82-Abstract.html.

[53] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural*

Networks for NLP, pages 353–355, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL `http://aclweb.org/anthology/W18-5446`.

[54] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

[55] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[56] Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. *Advances in Neural Information Processing Systems*, 33:20554–20565, 2020.

[57] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Limitations that we found most significant are discussed in the supplementary material.
   (c) Did you discuss any potential negative societal impacts of your work? [No] The work does not treat a specific task where we see an immediate risk for negative societal impact. Though the methods described in this paper might be used by practitioners, and we suggest doing so with care since the suggested methods need to be studied further in controlled settings before being used in such domains.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] The paper conforms with the provided guidelines.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code and Data for reproducing all of our results are submitted as part of the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5 and supplementary material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] All results are averaged across 5 runs
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See supplementary material.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 3
   (b) Did you mention the license of the assets? [Yes] See supplementary material.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] All assets we use are allowed for public use for the purposes of scientific research.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] The data we use is taken from publicly available datasets, where all personally identifiable information have been removed.

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

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See Section 5 for a high-level description of the crowdsourced data and the supplementary material for the exact set of instructions given to crowd workers.

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

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] See supplementary material for worker pay and costs breakdown.