How does this interaction affect me? Interpretable attribution for feature interactions

Michael Tsang, Sirisha Rambhatla, Yan Liu
Department of Computer Science
University of Southern California
{tsangm,sirishar,yanliu.cs}@usc.edu

Abstract

Machine learning transparency calls for interpretable explanations of how inputs relate to predictions. Feature attribution is a way to analyze the impact of features on predictions. Feature *interactions* are the contextual dependence between features that jointly impact predictions. There are a number of methods that extract feature interactions in prediction models; however, the methods that assign attributions to interactions are either uninterpretable, model-specific, or non-axiomatic. We propose an interaction attribution and detection framework called `Archipelago` which addresses these problems and is also scalable in real-world settings. Our experiments on standard annotation labels indicate our approach provides significantly more interpretable explanations than comparable methods, which is important for analyzing the impact of interactions on predictions. We also provide accompanying visualizations of our approach that give new insights into deep neural networks.

1 Introduction

The success of state-of-the-art prediction models such as neural networks is driven by their capability to learn complex feature interactions. When such models are used to make predictions for users, we may want to know how they personalize to us. Such model behaviors can be explained via *interaction detection* and *attribution*, i.e. if features influence each other and how these interactions contribute to predictions, respectively. Interaction explanations are useful for applications such as sentiment analysis [35], image classification [47], and recommendation tasks [21, 47].

Relevant methods for attributing predictions to feature interactions are black-box explanation methods based on axioms (or principles), but these methods lack interpretability. One of the core issues is that an interaction's importance is not the same as its attribution. Techniques like Shapley Taylor Interaction Index (STI) [14] and Integrated Hessians (IH) [25] combine these concepts in order to be axiomatic. Specifically, they base an interaction's attribution on non-additivity, i.e. the degree that features non-additively affect an outcome. While non-additivity can be used for interaction detection, it is not interpretable as an attribution measure as we see in Fig. 1. In addition, neither STI nor IH is tractable for higher-order feature interactions [14, 45]. Hence, there is a need for interpretable, axiomatic, and scalable methods for interaction attribution and corresponding interaction detection.

To this end, we propose a novel framework called `Archipelago`, which consists of an interaction attribution method, `ArchAttribute`, and a corresponding interaction detector, `ArchDetect`, to address the challenges of being interpretable, axiomatic, and scalable. `Archipelago` is named after its ability to provide explanations by isolating feature interactions, or feature "islands". The inputs to `Archipelago` are a black-box model f and data instance $\mathbf{x}^\star$, and its outputs are a set of interactions and individual features $\{\mathcal{I}\}$ as well as an attribution score $\phi(\mathcal{I})$ for each of the feature sets $\mathcal{I}$.

`ArchAttribute` satisfies attribution axioms by making relatively mild assumptions: a) disjointness of interaction sets, which is easily obtainable, and b) the availability of a generalized additive

Preprint. Under review.

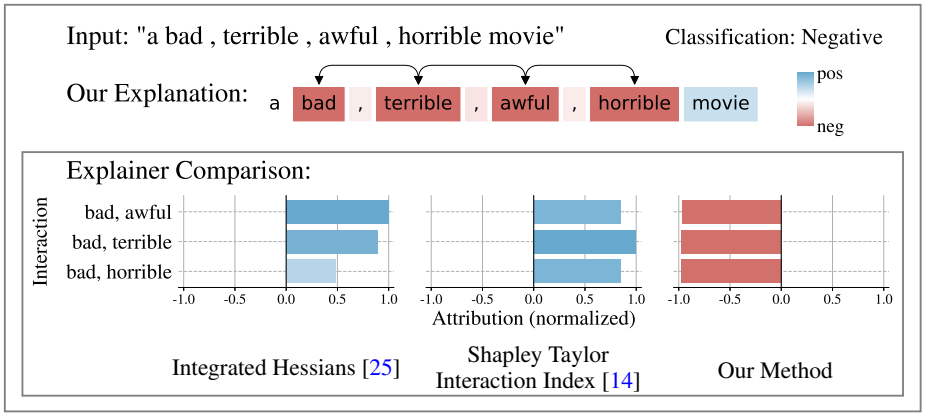


Figure 1: Our explanation for the sentiment analysis example of [25]. Colors indicate sentiment, and arrows indicate interactions. Compared to other axiomatic interaction explainers, only our work corroborates our intuition by showing negative attribution among top-ranked interactions.

function which is a good approximator to any function, as is leveraged in earlier works [48–50]. On the other hand, `ArchDetect` circumvents intractability issues of higher-order interaction detection by removing certain uninterpretable higher-order interactions and leveraging a property of feature interactions that allows pairwise interactions to merge for disjoint arbitrary-order interaction detection. In practice, where any assumptions may not hold in real-world settings, `Archipelago` still performs well. In particular, `Archipelago` effectively detects relevant interactions and is more interpretable than state-of-the-art methods [14, 20, 25, 26, 46, 50] when evaluated on annotation labels in sentiment analysis and image classification. We visualize `Archipelago` explanations on sentiment analysis, COVID-19 prediction on chest X-rays, and ad-recommendation.

Our main contributions are summarized below.

- **Interaction Attribution:** We propose `ArchAttribute`, a feature attribution measure that leverages feature interactions. It has advantages of being model-agnostic, interpretable, and runtime-efficient as compared to other state-of-the-art interaction attribution methods.
- **Principled Attribution:** `ArchAttribute` obeys standard attribution axioms [46] that are generalized to work for feature sets, and we also propose a new axiom for interaction attribution to respect the additive structure of a function.
- **Interaction Detection:** We propose a complementary feature interaction detector, `ArchDetect`, that is also model-agnostic and $\mathcal{O}(p^2)$-efficient for pairwise and disjoint arbitrary-order interaction detection (p is number of features).

Our empirical studies on `ArchDetect` and `ArchAttribute` demonstrate their superior properties as compared to state-of-the-art methods.

2 Notations and Background

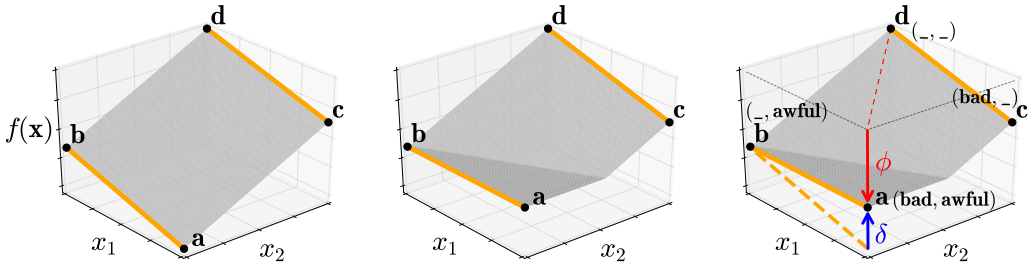
We first introduce preliminaries that serve as a basis for our discussions.

Notations: We use boldface lowercase symbols, such as $\mathbf{x}$, to represent vectors. The i-th entry of a vector $\mathbf{x}$ is denoted by x_i. For a set $\mathcal{S}$, its cardinality is denoted by $|\mathcal{S}|$, and the operation $\setminus\mathcal{S}$ means all except $\mathcal{S}$. For p features in a dataset, let $\mathcal{I}$ be a subset of feature indices: $\mathcal{I} \subseteq \{1, 2, \ldots, p\}$. For a vector $\mathbf{x} \in \mathbb{R}^p$, let $\mathbf{x}_\mathcal{I} \in \mathbb{R}^p$ be defined element-wise in (1). In our discussions, a *context* means $\mathbf{x}_{\setminus\mathcal{I}}$.

$$(\mathbf{x}_\mathcal{I})_i = \begin{cases} x_i, & \text{if } i \in \mathcal{I} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Problem Setup: Let f denote a black-box model with scalar output. For multi-class classification, f is assumed to be a class logit. We use a target vector $\mathbf{x}^\star \in \mathbb{R}^p$ to denote the data instance where we wish to explain f, and $\mathbf{x}' \in \mathbb{R}^p$ to denote a *neutral baseline*. Here, the baseline is a reference vector for $\mathbf{x}^\star$ and conveys an "absence of signal" as per [46]. These vectors form the space of $\mathcal{X} \subset \mathbb{R}^p$, where each element comes from either $x_i^\star$ or x_i', i.e. $\mathcal{X} = \{(x_1, \ldots, x_p) \mid x_i \in \{x_i^\star, x_i'\}, \forall i = 1, \ldots, p\}$.

Feature Interaction: The definition of the feature interaction of interest is formalized as follows.



(a) Additive (linear) function (b) Non-additive (ReLU) function (c) δ vs. ϕ on a text example (Fig. 1)

Figure 2: Non-additive interaction for $p = 2$ features: The corner points are used to determine if x_1 and x_2 interact based on their non-additivity on f, i.e. they interact if $\delta \propto (f(\mathbf{a}) - f(\mathbf{b})) - (f(\mathbf{c}) - f(\mathbf{d})) \neq 0$ (§4.1). In (c), the attribution of (bad, awful) should be negative via ϕ (2), but Shapley Taylor Interaction Index uses the positive δ. Note that ϕ depends on $\mathbf{a}$ and $\mathbf{d}$ whereas δ depends on $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$. Also, Integrated Hessians is not relevant here since it does not apply to ReLU functions.

Definition 1 (Statistical Non-Additive Interaction)**.** *A function f contains a statistical non-additive interaction of multiple features indexed in set $\mathcal{I}$ if and only if f cannot be decomposed into a sum of $|\mathcal{I}|$ subfunctions f_i , each excluding the i-th interaction variable: $f(\mathbf{x}) \neq \sum_{i \in \mathcal{I}} f_i(\mathbf{x}_{\setminus\{i\}})$.*

Def. 1 identifies a non-additive effect among all features $\mathcal{I}$ on the output of function f [18, 45, 48]. For example, this means that the function ReLU$(x_1 + x_2)$ creates a feature interaction because it cannot be represented as an addition of univariate functions, i.e., ReLU$(x_1 + x_2) \neq f_1(x_2) + f_2(x_1)$ (Fig. 2b). We refer to individual feature effects which do not interact with other features as *main effect*. Higher-order feature interactions are captured by $|\mathcal{I}| > 2$, i.e. interactions larger than pairs. Additionally, if a higher-order interaction exists, all of its subsets also exist as interactions [45, 48].

3 Archipelago Interaction Attribution

We begin by presenting our feature attribution measure. Our feature attribution analyzes and assigns scores to detected feature interactions. Our corresponding interaction detector is presented in §4.

3.1 ArchAttribute

Let $\mathcal{I}$ be the set of feature indices that correspond to a desired attribution score. Our proposed attribution measure, called ArchAttribute, is given by

$$\phi(\mathcal{I}) = f(\mathbf{x}^\star_\mathcal{I} + \mathbf{x}'_{\setminus\mathcal{I}}) - f(\mathbf{x}'). \tag{2}$$

ArchAttribute essentially isolates the attribution of $\mathbf{x}^\star_\mathcal{I}$ from the surrounding baseline context while also satisfying axioms (§3.2). We call this isolation an "island effect", where the target features $\{x^\star_i\}_{i \in \mathcal{I}}$ do not specifically interact with the baseline features $\{x'_j\}_{j \in \setminus\mathcal{I}}$. For example, consider sentiment analysis on a phrase $\mathbf{x}^\star =$ "not very bad" with a baseline $\mathbf{x}' =$ "_ _ _" . Suppose that we want to examine the attribution of an interaction $\mathcal{I}$ that corresponds to {very, bad} in isolation. In this case, the contextual word "not" also interacts with $\mathcal{I}$, which becomes apparent when small perturbations to the word "not" causes large changes to prediction probabilities. However, as we move further away from the word "not" towards the empty-word "_" in the word-embedding space, small perturbations no longer result in large prediction changes, meaning that "_" does not specifically interact with {very, bad}. This intuition motivates our use of the baseline context $\mathbf{x}'_{\setminus\mathcal{I}}$ in (2).

3.2 Axioms

We now show how ArchAttribute obeys standard feature attribution axioms [46]. Since ArchAttribute operates on feature sets, we generalize the notion of standard axioms to feature sets. To this end, we also propose a new axiom, Set Attribution, which allows us to work with feature sets.

Let $\mathcal{S} = \{\mathcal{I}_i\}_{i=1}^k$ be all k feature interactions and main effects of f in the space $\mathcal{X}$ (defined in §2), where we take the union of overlapping sets in $\mathcal{S}$. Later in §4, we explain how to obtain $\mathcal{S}$.

Completeness: We consider a generalization of the completeness axiom for which the sum of all attributions equals $f(\mathbf{x}^\star) - f(\mathbf{x}')$. The axiom tells us how much feature(s) impact a prediction.

Lemma 2 (Completeness on $\mathcal{S}$). *The sum of all attributions by* `ArchAttribute` *for the disjoint sets in $\mathcal{S}$ equals the difference of f between $\mathbf{x}^\star$ and the baseline $\mathbf{x}'$: $f(\mathbf{x}^\star) - f(\mathbf{x}')$.*

The proof is in Appendix C. We can easily see `ArchAttribute` satisfying this axiom in the limiting case where $k = 1, \mathcal{I}_1 = \{i\}_{i=1}^p$ because (2) directly becomes $f(\mathbf{x}^\star) - f(\mathbf{x}')$. Existing interaction / group attribution methods: Sampling Contextual Decomposition (SCD) [26], its variant (CD) [35,42], Sampling Occlusion (SOC) [26], and Shapley Interaction Index (SI) [20] do not satisfy completeness, whereas Integrated Hessians (IH) [25] and Shapley Taylor Interaction Index (STI) [14] do.

Set Attribution: We propose an axiom for interaction attribution called **Set Attribution** to work with feature sets as opposed to individual features and follow the additive structure of a function.

Axiom 3 (Set Attribution). *If $f : \mathbb{R}^p \to \mathbb{R}$ is a function in the form of $f(\mathbf{x}) = \sum_{i=1}^k \varphi_i(\mathbf{x}_{\mathcal{I}_i})$ where $\{\mathcal{I}_i\}_{i=1}^k$ are disjoint and functions $\{\varphi_i(\cdot)\}_{i=1}^k$ have roots, then an interaction attribution method admits an attribution for feature set $\mathcal{I}_i$ as $\varphi_i(\mathbf{x}_{\mathcal{I}_i})$ $\forall i = 1, \ldots, k$.*

For example, if we consider a function $y = x_1x_2 + x_3$; it makes sense for the attribution of the x_1x_2 interaction to be the value of x_1x_2 and the attribution for the x_3 main effect to be the value of x_3.

Lemma 4 (Set Attribution on $\mathcal{S}$). *For $\mathbf{x} = \mathbf{x}^\star$ and a baseline $\mathbf{x}'$ such that $\varphi_i(\mathbf{x}'_{\mathcal{I}_i}) = 0$ $\forall i = 1, \ldots, k$,* `ArchAttribute` *satisfies the Set Attribution axiom and provides attribution $\varphi_i(\mathbf{x}_{\mathcal{I}_i})$ for set $\mathcal{I}_i$ $\forall i$.*

The proof is in Appendix E, which follows from Lemma 2. Neither SCD, CD, SOC, SI, IH, nor STI satisfy Set Attribution (shown in Appendix E.1). We can enable Integrated Gradients (IG) [46] to satisfy our axiom by summing its attributions within each feature set of $\mathcal{S}$. `ArchAttribute` differs from IG by its "island effect" (§3.1) and model-agnostic properties.

Other Axioms: `ArchAttribute` also satisfies the remaining axioms: Sensitivity, Implementation Invariance, Linearity, and Symmetry-Preserving, which we show via Lemmas 7-11 in Appendix F.

Discussion: Several axioms required disjoint interaction and main effect sets in $\mathcal{S}$. Though interactions are not necessarily disjoint by definition (Def. 1), it is reasonable to merge overlapping interactions to obtain compact visualizations, as shown in Fig. 1 and later experiments (§5.3). The disjoint sets also allow `ArchAttribute` to yield identifiable non-additive attributions in the sense that it can identify the attribution given a feature set in $\mathcal{S}$. This contrasts with Model-Agnostic Hierarchical Explanations (MAHE) [50], which yields unidentifiable attributions [56].

4 `Archipelago` Interaction Detection

Our axiomatic analysis of `ArchAttribute` relied on $\mathcal{S}$, which contains interaction sets of f on the space $\mathcal{X}$ (defined in §2). To develop an interaction detection method that works in tandem with `ArchAttribute`, we draw inspiration from the discrete interpretation of mixed partial derivatives.

4.1 Discrete Interpretation of Mixed Partial Derivatives

Consider the plots in Fig. 2, which consist of points $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ that each contain two features. From a top-down view of each plot, the points form the corners of a rectangle, whose side lengths are $h_1 = |a_1 - b_1| = |c_1 - d_1|$ and $h_2 = |a_2 - c_2| = |b_2 - d_2|$. When h_1 and h_2 are small, the mixed partial derivative w.r.t variables x_1 and x_2 is computed as follows. First, $\frac{\partial f(\mathbf{a})}{\partial x_1} \approx \frac{1}{h_1}\left(f(\mathbf{a}) - f(\mathbf{b})\right)$ and $\frac{\partial f(\mathbf{c})}{\partial x_1} \approx \frac{1}{h_1}\left(f(\mathbf{c}) - f(\mathbf{d})\right)$. Similarly, the mixed partial derivative is approximated as:

$$\frac{\partial^2 f}{\partial x_1 x_2} \approx \frac{1}{h_2}\left(\frac{\partial f(\mathbf{a})}{\partial x_1} - \frac{\partial f(\mathbf{c})}{\partial x_1}\right) \approx \frac{1}{h_1 h_2}\left(\left(f(\mathbf{a}) - f(\mathbf{b})\right) - \left(f(\mathbf{c}) - f(\mathbf{d})\right)\right). \tag{3}$$

When h_1 and h_2 become large, (3) tells us if a plane can fit through all four points $\mathbf{a}$,$\mathbf{b}$,$\mathbf{c}$, $\mathbf{d}$ (Fig. 2a), which occurs when (3) is zero. In this domain where x_1 and x_2 only take two possible values each, a plane in the linear form $f(\mathbf{x}) = w_1x_1 + w_2x_2 + b$ is functionally equivalent to all functions of the form $f(\mathbf{x}) = f_1(x_1) + f_2(x_2) + b$, so any deviation from the plane, e.g. Fig. 2b, becomes non-additive. Consequently, a *non-zero* value of (3) identifies a non-additive interaction by the definition of statistical interaction (Def. 1). What's more, the magnitude of (3) tells us the degree of deviation from the plane, or the degree of non-additivity. (Additional details in Appendix G)

4.2 `ArchDetect`

Leveraging these insights about mixed partial derivatives, we now discuss the two components of our proposed interaction detection technique – `ArchDetect`.

4.2.1 Handling Context: As defined in §3.2 and §4, our problem is how to identify interactions of p features in $\mathcal{X}$ for our target data instance $\mathbf{x}^\star$ and baseline $\mathbf{x}'$. If $p = 2$, then we can almost directly use (3), where $\mathbf{a} = (x_1^\star, x_2^\star)$, $\mathbf{b} = (x_1', x_2^\star)$, $\mathbf{c} = (x_1^\star, x_2')$, and $\mathbf{d} = (x_1', x_2')$. However if $p > 2$, all possible combinations of features in $\mathcal{X}$ would need to be examined to thoroughly identify just one pairwise interaction. To see this, we first rewrite (3) to accommodate p features, and square the result to measure interaction strength and be consistent with previous interaction detectors [18, 19]. The interaction strength between features i and j for a context $\mathbf{x}_{\setminus\{i,j\}}$ is then defined as

$$\omega_{i,j}(\mathbf{x}) = \left(\tfrac{1}{h_i h_j}\left(f(\mathbf{x}^\star_{\{i,j\}} + \mathbf{x}_{\setminus\{i,j\}}) - f(\mathbf{x}'_{\{i\}} + \mathbf{x}^\star_{\{j\}} + \mathbf{x}_{\setminus\{i,j\}}) - f(\mathbf{x}^\star_{\{i\}} + \mathbf{x}'_{\{j\}} + \mathbf{x}_{\setminus\{i,j\}}) + f(\mathbf{x}'_{\{i,j\}} + \mathbf{x}_{\setminus\{i,j\}})\right)\right)^2, \tag{4}$$

where $h_i = |x_i^\star - x_i'|$ and $h_j = |x_j^\star - x_j'|$. The thorough way to identify the $\{i, j\}$ feature interaction is given by $\bar{\omega}_{i,j} = \mathbb{E}_{\mathbf{x}\in\mathcal{X}}[\omega_{i,j}(\mathbf{x})]$, where each element of $\mathbf{x}_{\setminus\{i,j\}}$ is Bernoulli (0.5). This expectation is intractable because $\mathcal{X}$ has an exponential search space, so we propose the first component of `ArchDetect` for efficient pairwise interaction detection:

$$\bar{\omega}_{i,j} = \frac{1}{2}\left(\omega_{i,j}(\mathbf{x}^\star) + \omega_{i,j}(\mathbf{x}')\right). \tag{5}$$

Here, we estimate the expectation by leveraging the physical meaning of the interactions and `ArchAttribute`'s axioms via the different contexts of $\mathbf{x}$ in (5) as follows:

- **Context of $\mathbf{x}^\star$:** An important interaction is one due to multiple $\mathbf{x}^\star$ features. As a concrete example, consider an image representation of a cat which acts as our target data instance. The following higher-order interaction, $if\ x_{ear} = x^\star_{ear}\ and\ x_{nose} = x^\star_{nose}\ and\ x_{fur} = x^\star_{fur}\ then\ f(\mathbf{x}) = high\ cat\ probability$, is responsible for classifying "cat". We can detect any pairwise subset $\{i, j\}$ of this interaction by setting the context as $\mathbf{x}^\star_{\setminus\{i,j\}}$ using $\omega_{i,j}(\mathbf{x}^\star)$.
- **Context of $\mathbf{x}'$:** Next, we consider $\mathbf{x}'_{\setminus\{i,j\}}$ to detect interactions via $\omega_{i,j}(\mathbf{x}')$, which helps us establish `ArchAttribute`'s completeness (Lemma 2). This also separates out effects of any higher-order baseline interactions from $f(\mathbf{x}')$ in (8) (Appendix C) and recombine their effects in (11). From an interpretability standpoint, the $\mathbf{x}'_{\setminus\{i,j\}}$ context ranks pairwise interactions w.r.t. a standard baseline. This context is also used by `ArchAttribute` (2).
- **Other Contexts:** The first two contexts accounted for any-order interactions created by either target or baseline features and a few interactions created by a mix of baseline and target features. The remaining interactions specifically require a mix of > 3 target and baseline features. This case is unlikely and is excluded, as we discuss next.

The following assumption formalizes our intuition for the *Other Contexts* setting where there is a mix of higher-order (> 3) target and baseline feature interactions.

Assumption 5 (Higher-Order Mixed-Interaction)**.** *For any feature set $\mathcal{I}$ where $|\mathcal{I}| > 3$ and any pair of non-empty disjoint sets $\mathcal{A}$ and $\mathcal{B}$ where $\mathcal{A} \cup \mathcal{B} = \mathcal{I}$, the instances $\mathbf{x} \in \mathcal{X}$ such that $x_i = x_i^\star\ \forall i \in \mathcal{A}$ and $x_j = x_j'\ \forall j \in \mathcal{B}$ do not cause a higher-order interaction of all features $\{x_k\}_{k\in\mathcal{I}}$ via f.*

Assumption 5 has a similar intuition as `ArchAttribute` in §3.1 that target features do not specifically interact with baseline features. To understand this assumption, consider the original sentiment analysis example in Fig. 1 simplified as $\mathbf{x}^\star =$ "bad terrible awful horrible movie" where $\mathbf{x}' =$ "_ _ _ _ _". It is reasonable to assume that there is no special interaction created by token sets such as {bad, terrible, _ , horrible} or {_ , _ , _ , horrible} due to the meaningless nature of the "_" token.

Efficiency: In (5), `ArchDetect` attains interaction detection over all pairs $\{i, j\}$ in $\mathcal{O}(p^2)$ calls of f. Note that in (4), most function calls are reusable during pairwise interaction detection.

4.2.2 Detecting Disjoint Interaction Sets: In this section, the aim here is to recover arbitrary size and disjoint non-additive feature sets $\mathcal{S} = \{\mathcal{I}_i\}$ (not just pairs). `ArchDetect` looks at the union of overlapping pairwise interactions to obtain disjoint feature sets. Merging these pairwise interactions captures any existing higher-order interactions automatically since the existence of a higher-order interaction automatically means all its subset interactions exist (§2). In addition, `ArchDetect` merges these overlapped pairwise interactions with all individual feature effects to account for all features. The time complexity of this merging process is also $\mathcal{O}(p^2)$.

Table 1: Comparison of interaction detectors (b) on synthetic ground truth in (a).

(a) Functions with Ground Truth Interactions

$F_1(\mathbf{x}) =$	$\sum_{i=1}^{10}\sum_{j=1}^{10} x_i x_j + \sum_{i=11}^{20}\sum_{j=21}^{30} x_i x_j + \sum_{k=1}^{40} x_k$
$F_2(\mathbf{x}) =$	$\bigwedge(\mathbf{x}; \{x_i^\star\}_{i=1}^{20}) + \bigwedge(\mathbf{x}; \{x_i^\star\}_{i=11}^{30}) + \sum_{j=1}^{40} x_j$
$F_3(\mathbf{x}) =$	$\bigwedge(\mathbf{x}; \{x_i'\}_{i=1}^{20}) + \bigwedge(\mathbf{x}; \{x_i^\star\}_{i=11}^{30}) + \sum_{j=1}^{40} x_j$
$F_4(\mathbf{x}) =$	$\bigwedge(\mathbf{x}; \{x_1^\star, x_2^\star\} \cup \{x_3'\}) + \bigwedge(\mathbf{x}; \{x_i^\star\}_{i=11}^{30}) + \sum_{j=1}^{40} x_j$

(b) Pairwise Interaction Ranking AUC. The baseline methods fail to detect interactions suited for the desired contexts in §4.2.1.

Method	F_1	F_2	F_3	F_4
Two-way ANOVA	1.0	0.51	0.51	0.55
Integrated Hessians	1.0	N/A	N/A	N/A
Neural Interaction Detection	0.94	0.52	0.48	0.56
Shapley Interaction Index	1.0	0.50	0.50	0.51
Shapley Taylor Interaction Index	1.0	0.50	0.53	0.51
`ArchDetect` (this work)	1.0	1.0	1.0	1.0

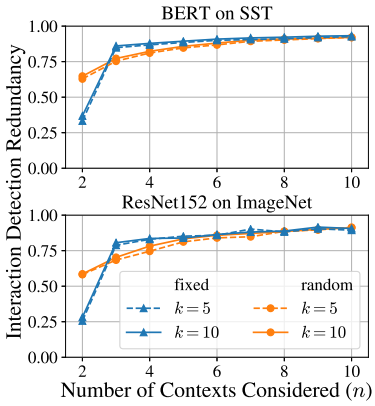


Figure 3: Interaction detection overlap (redundancy) with added contexts to (5). "fixed" at $n = 2$ (`ArchDetect`) already shows good stability.

5 Experiments

5.1 Setup

We conduct experiments first on `ArchDetect` in §5.2 then on `ArchAttribute` in §5.3. We then visualize their combined form as `Archipelago` in §5.3. Throughout our experiments, we commonly study BERT [13, 55] on text-based sentiment analysis and ResNet152 [24] on image classification. BERT was fine-tuned on the SST dataset [43], and ResNet152 was pretrained on ImageNet [12].

For sentiment analysis, we set the baseline vector $\mathbf{x}'$ to be the tokens "_", in place of each word-token from $\mathbf{x}^\star$. For image classification, we set $\mathbf{x}'$ to be an all-zero image, and use the Quickshift superpixel segmenter [52] as per the need for input dimensionality reduction [47] (details in Appendix B). We set $h_1 = h_2 = 1$ for both domains. Several methods we compare to are common across experiments, in particular IG, IH, (disjoint) MAHE, SI, STI, and Difference, defined as $\phi_d(\mathcal{I}) = f(\mathbf{x}^\star) - f(\mathbf{x}'_{\mathcal{I}} + \mathbf{x}^\star_{\setminus\mathcal{I}})$.

5.2 `ArchDetect`

We validate `ArchDetect`'s performance via synthetic ground truth and redundancy experiments.

Synthetic Validation: We set $\mathbf{x}^\star = [1, 1, \ldots, 1] \in \mathbb{R}^{40}$ and $\mathbf{x}' = [-1, -1, \ldots, -1] \in \mathbb{R}^{40}$. Let $z[\cdot]$ be a key-value pair function such that $z[i] = x_i$ for key $i \in z.keys$ and value x_i, so we can define

$$\bigwedge(\mathbf{x}; z) := \begin{cases} 1, & \text{if } x_i = z[i]\ \forall i \in z.keys \\ -1 & \text{for all other cases.} \end{cases}$$

Table 1a shows functions with ground truth interactions suited for the desired contexts in §4.2.1. Table 1b shows interaction detection AUC on these functions by `ArchDetect`, IH, SI, STI, Two-way ANOVA [16] and the state-of-the-art Neural Interaction Detection [48]. On F_2, F_3, & F_4, the baseline methods fail because they are not designed to detect the interactions of our desired contexts (§4.2.1).

Interaction Redundancy: The purpose of the next experiments is to see if `ArchDetect` can omit certain higher-order interactions. We study the form of (5) by examining the redundancy of interactions as new contexts are added to (5), which we now write as $\bar{\omega}_{i,j}(C) = \frac{1}{C}\sum_{c=1}^{C} \omega_{i,j}(\mathbf{x}_c)$. Let n be the number of contexts considered, and k be the number of top pairwise interactions selected after running pairwise interaction detection via $\bar{\omega}_{i,j}$ for all $\{i, j\}$ pairs. Interaction redundancy is the overlap ratio of two sets of top-k pairwise interactions, one generated via $\bar{\omega}_{i,j}(n)$ and the other one via $\bar{\omega}_{i,j}(n-1)$ for some integer $n \geq 2$. We generally expect the redundancy to increase as n increases, which we initially observe in Fig. 3. Here, "fixed" and "random" correspond to different context sequences $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. The "random" sequence uses random samples from $\mathcal{X}$ for all $\{\mathbf{x}_i\}_{i=1}^N$, whereas the "fixed" sequence is fixed in the sense that $\mathbf{x}_1 = \mathbf{x}^\star$, $\mathbf{x}_2 = \mathbf{x}'$, and the remaining $\{\mathbf{x}_i\}_{i=3}^N$ are random samples. Experiments are done on the SST test set for BERT and 100 random test images in ImageNet for ResNet152. Notably, the "fixed" setting has very low redundancy at $n = 2$ (`ArchDetect`) versus "random". As soon as $n = 3$, the redundancy jumps and stabilizes quickly. These experiments support Assumption 5 and (5) to omit specified higher-order interactions.

Table 2: Comparison of attribution methods on BERT for sentiment analysis and ResNet152 for image classification. Performance is measured by the correlation (ρ) or AUC of the top and bottom 10% of attributions for each method with respect to reference scores defined in §5.3.

Method	BERT Sentiment Analysis		ResNet152 Image Classification
	Word ρ	Phrase ρ †	Segment AUC †
Difference	0.427	0.639	0.705
Integrated Gradients (IG)	0.568	0.737	0.786
Integrated Hessians (IH)	N/A	0.128	N/A
Model-Agnostic Hierarchical Explanations (MAHE)	0.673	0.702	0.712
Shapley Interaction Index (SI)	0.168	−0.018	0.530
Shapley Taylor Interaction Index (STI)	0.754	0.286	0.626
*Sampling Contextual Decomposition (SCD)	0.709	0.742	N/A
*Sampling Occlusion (SOC)	0.768	0.794	N/A
`ArchAttribute` (this work)	**0.809**	**0.836**	**0.919**

† Methods that cannot tractably run for arbitrary feature set sizes are only run for pairwise feature sets.
* SCD and SOC are specifically for sequence models and contiguous words.

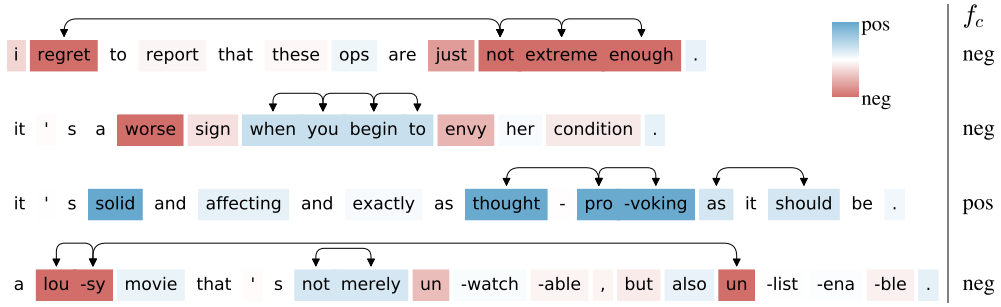


Figure 4: Our BERT visualizations on random test sentences from SST under BERT tokenization. Arrows indicate interactions, and colors indicate attribution strength. f_c is the sentiment classification. The interactions point to salient and sometimes long-range sets of words, and the colors are sensible.

5.3 `ArchAttribute` & `Archipelago`

We study the interpetability of `ArchAttribute` by comparing its attribution scores to ground truth annotation labels on subsets of features. For fair comparison, we look at extreme attributions (top and bottom 10%) for each baseline method. We then visualize the combined `Archipelago` framework. Additional comparisons on attributions, runtime, and visualizations are shown in Appendices I, J, K.

Sentiment Analysis: For this task, we compare `ArchAttribute` to other explanation methods on two metrics: phrase correlation (Phrase ρ) and word correlation (Word ρ) on the SST test set (metrics are from [26]). Phrase ρ is the Pearson correlation between estimated phrase attributions and SST phrase labels (excluding prediction labels) on a 5-point sentiment scale. Word ρ is unlike our label-based evaluations by computing the Pearson correlation between estimated word attributions and the corresponding coefficients of a global bag-of-words linear model, which is also trained on the SST dataset. In addition to the aforementioned baseline methods in §5.1, we include the state-of-the-art SCD and SOC methods for sequence models [26] in our evaluation. In Table 2, `ArchAttribute` compares favorably to all methods where we consider the top and bottom 10% of the attribution scores for each method. We obtain similar performance across all other percentiles in Appendix I.

We visualize `Archipelago` explanations on $\mathcal{S}$ generated by top-3 pairwise interactions (§4.2.2) in Fig. 4. The sentence examples are randomly selected from the SST test set. The visualizations show interactions and individual feature effects which all have reasonable polarity and intensity. Interestingly, some of the interactions, e.g. between "lou-sy" and "un", are long range.

Image Classification: On image classification, we compare `ArchAttribute` to relevant baseline methods on a "Segment AUC" metric, which computes the agreement between the estimated attribution of an image segment and that segment's label. We obtain segment labels from the MS COCO dataset [29] and match them to the label space of ImageNet. All explanation attributions are computed relative to ResNet152's top-classification in the joint label space. The segment label thus becomes whether or not the segment belongs to the same class as the top-classification. Evaluation is

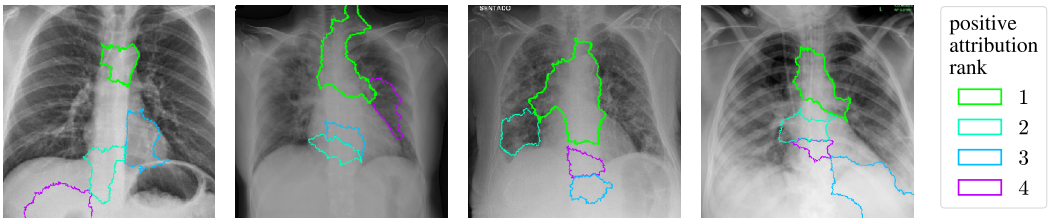


Figure 5: Our explanations of a COVID-19 classifier (COVID-Net) [53] on randomly selected test X-rays [9, 10] classified as COVID positive. COVID-Net accurately distinguishes COVID from pneumonia and normal X-rays. Colored outlines indicate detected feature sets with positive attribution. The explanations tend to detect on the "great vessels" outlined in green, which are mostly interactions.

conducted on all segments with valid labels in the MS COCO dev set. `ArchAttribute` performs especially well on extreme attributions in Table 2, as well as all attributions (in Appendix I).

Fig. 5 visualizes `Archipelago` on an accurate COVID-19 classifier for chest X-rays [53], where $\mathcal{S}$ is generated by top-5 pairwise interactions (§4.2.2). Shown is a random selection of test X-rays [9, 10] that are classified COVID-positive. The explanations tend to detect the "great vessels" near the heart.

Recommendation Task: Fig. 6 shows `Archipelago`'s result for this task using a state-of-the-art AutoInt model [44] for ad-recommendation. Here, our approach finds a positive interaction between"device_id" and "banner_pos" in the Avazu dataset [1], meaning that the online advertisement model decides the banner position based on user device_id. Note that for this task, there are no ground truth annotations.

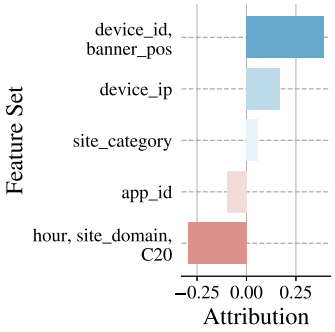


Figure 6: Online ad-targeting: "banner_pos" is used to target ads to a user per their "device_id".

6 Related Works

Attribution: Individual feature attribution methods distill any interactions of a data instance as attribution scores for each feature. Many methods require the scores to sum to equal the output [7, 32, 38, 40, 46], such as LIME and SHAP, which train surrogate linear explainer models on feature perturbations, and IG which invokes the fundamental theorem of calculus. Other methods compute attributions from an information theoretic perspective [8] or strictly from model gradients [4, 39, 41]. These methods interpret feature importance but not feature interactions.

Feature Interaction: Feature interaction explanation methods tend to either perform interaction detection [2,6,16,18,19,45,48] or combined interaction detection and attribution [14,25,30,31,37,50]. Relevant black-box interaction explainers are STI [14] which uses random feature orderings to identify contexts for a variant of (4) so that interaction scores satisfy completeness, IH [25] which extends IG with path integration for hessian computations, and MAHE [50], which trains surrogate explainer models for interaction detection and attribution. STI and IH are axiomatic and satisfy completeness but their attributions are uninterpretable (Table 2) and inefficient. MAHE's attributions are unidentifiable by training additive attribution models on overlapping feature sets. Several methods compute attributions on feature sequences or sets, such as SOC [26], SCD [26], and CD [35, 42], but they do not obey basic axioms. Finally, many methods are not model-agnostic, such as SCD, CD, IG, IH, GA2M [30], and Tree-SHAP [31]. Additional earlier works are discussed in Appendix H.

7 Discussion

Understandable and accessible explanations are cornerstones of interpretability which informed our isolation and disjoint designs of `ArchAttribute` and `ArchDetect`, respectively. Here, we develop an interpretable, model-agnostic, axiomatic, and efficient interaction explainer which achieves state-of-the-art results on multiple attribution tasks. In addition, we introduce a new axiom and generalize existing axioms to higher-order interaction settings. This provides guidance on how to design interaction attribution methods. To be able to solve the transparency issue, we need to understand feature attribution better. This work proposes interpretable and axiomatic feature interaction explanations to motivate future explorations in this area.

Broader Impact

The purpose of this work is to provide new insights into existing and future prediction models. The explanations from `Archipelago` can be used by both machine learning practitioners and audiences without background expertise. The societal risk of this work is any overdependence on `Archipelago`. Users of this explanation method should consider the merits of not only this method but also other explanation methods for their use cases. For example, users may want fine-grained pixel-level explanations of image classifications whereas our explanations may require superpixel segmentation. Nevertheless, we believe this work can help reveal biases in prediction models, assist in scientific discovery, and stimulate discussions on how to debug models based on feature interactions.

References

[1] Avazu click-through-rate prediction. https://www.kaggle.com/c/avazu-ctr-prediction. Accessed: 2020-04-14.

[2] Chunrong Ai and Edward C Norton. Interaction terms in logit and probit models. *Economics letters*, 80(1):123–129, 2003.

[3] Leona S Aiken, Stephen G West, and Raymond R Reno. *Multiple regression: Testing and interpreting interactions*. Sage, 1991.

[4] Marco Ancona, Enea Ceolini, Cengiz Oztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *6th International Conference on Learning Representations*, 2018.

[5] William A Belson. Matching and prediction on the principle of biological classification. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 8(2):65–75, 1959.

[6] Jacob Bien, Jonathan Taylor, and Robert Tibshirani. A lasso for hierarchical interactions. *Annals of statistics*, 41(3):1111, 2013.

[7] 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.

[8] Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning*, pages 883–892, 2018.

[9] Muhammad EH Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar R Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al-Emadi, et al. Can ai help in screening viral and covid-19 pneumonia? *arXiv preprint arXiv:2003.13145*, 2020.

[10] Joseph Paul Cohen, Paul Morrison, and Lan Dao. Covid-19 image data collection. *arXiv 2003.11597*, 2020.

[11] Angela Dean, Max Morris, John Stufken, and Derek Bingham. *Handbook of design and analysis of experiments*, volume 7. CRC Press, 2015.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[13] 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, 2019.

[14] Kedar Dhamdhere, Ashish Agarwal, and Mukund Sundararajan. The shapley taylor interaction index. *arXiv preprint arXiv:1902.05622*, 2019.

[15] Ronald A Fisher. On the'probable error'of a coefficient of correlation deduced from a small sample. *Metron*, 1:1–32, 1921.

[16] Ronald Aylmer Fisher. *Statistical methods for research workers*. Genesis Publishing Pvt Ltd, 1925.

[17] Ronald Aylmer Fisher et al. 048: The arrangement of field experiments. 1926.

[18] Jerome H Friedman, Bogdan E Popescu, et al. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

[19] Muriel Gevrey, Ioannis Dimopoulos, and Sovan Lek. Two-way interaction of input variables in the sensitivity analysis of neural network models. *Ecological modelling*, 195(1-2):43–50, 2006.

[20] Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of game theory*, 28(4):547–565, 1999.

[21] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 1725–1731. AAAI Press, 2017.

[22] Michael Hamada and CF Jeff Wu. Analysis of designed experiments with complex aliasing. *Journal of Quality Technology*, 24(3):130–137, 1992.

[23] Ning Hao and Hao Helen Zhang. Interaction screening for ultrahigh-dimensional data. *Journal of the American Statistical Association*, 109(507):1285–1301, 2014.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[25] Joseph D Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *arXiv preprint arXiv:2002.04138*, 2020.

[26] Xisen Jin, Junyi Du, Zhongyu Wei, Xiangyang Xue, and Xiang Ren. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. *arXiv preprint arXiv:1911.06194*, 2019.

[27] Gordon V Kass. An exploratory technique for investigating large quantities of categorical data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 29(2):119–127, 1980.

[28] Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*, 2016.

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[30] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631. ACM, 2013.

[31] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018.

[32] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pages 4765–4774, 2017.

[33] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[34] James N Morgan and John A Sonquist. Problems in the analysis of survey data, and a proposal. *Journal of the American statistical association*, 58(302):415–434, 1963.

[35] W James Murdoch, Peter J Liu, and Bin Yu. Beyond word importance: Contextual decomposition to extract interactions from lstms. *International Conference on Learning Representations*, 2018.

[36] JA Nelder. A reformulation of linear models. *Journal of the Royal Statistical Society: Series A (General)*, 140(1):48–63, 1977.

[37] Sanjay Purushotham, Martin Renqiang Min, C-C Jay Kuo, and Rachel Ostroff. Factorized sparse learning models with interpretable high order feature interactions. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 552–561, 2014.

[38] 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. ACM, 2016.

[39] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[40] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3145–3153. JMLR. org, 2017.

[41] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[42] Chandan Singh, W James Murdoch, and Bin Yu. Hierarchical interpretations for neural network predictions. *International Conference on Learning Representations*, 2019.

[43] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[44] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. Autoint: Automatic feature interaction learning via self-attentive neural networks. *arXiv preprint arXiv:1810.11921*, 2018.

[45] Daria Sorokina, Rich Caruana, Mirek Riedewald, and Daniel Fink. Detecting statistical interactions with additive groves of trees. In *Proceedings of the 25th international conference on Machine learning*, pages 1000–1007. ACM, 2008.

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

[47] Michael Tsang, Dehua Cheng, Hanpeng Liu, Xue Feng, Eric Zhou, and Yan Liu. Feature interaction interpretability: A case for explaining ad-recommendation systems via neural interaction detection. In *International Conference on Learning Representations*, 2020.

[48] Michael Tsang, Dehua Cheng, and Yan Liu. Detecting statistical interactions from neural network weights. *International Conference on Learning Representations*, 2018.

[49] Michael Tsang, Hanpeng Liu, Sanjay Purushotham, Pavankumar Murali, and Yan Liu. Neural interaction transparency (nit): Disentangling learned interactions for improved interpretability. In *Advances in Neural Information Processing Systems*, pages 5804–5813, 2018.

[50] Michael Tsang, Youbang Sun, Dongxu Ren, and Yan Liu. Can i trust you more? model-agnostic hierarchical explanations. *arXiv preprint arXiv:1812.04801*, 2018.

[51] John W Tukey. One degree of freedom for non-additivity. *Biometrics*, 5(3):232–242, 1949.

[52] Andrea Vedaldi and Stefano Soatto. Quick shift and kernel methods for mode seeking. In *European Conference on Computer Vision*, pages 705–718. Springer, 2008.

[53] Linda Wang and Alexander Wong. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest radiography images. *arXiv preprint arXiv:2003.09871*, 2020.

[54] Martin B Wilk. The randomization analysis of a generalized randomized block design. *Biometrika*, 42(1/2):70–79, 1955.

[55] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.

[56] Simon N Wood. *Generalized additive models: an introduction with R*. CRC press, 2017.

[57] Frank Yates. Sir ronald fisher and the design of experiments. *Biometrics*, 20(2):307–321, 1964.

Appendix

A Acronyms

Table 3: Acronym Definitions

Acronym	Meaning
pos	positive
neg	negative
IG	Integrated Gradients [46]
IH	Integrated Hessians [25]
MAHE	Model-Agnostic Hierarchical Explanations [50]
SI	Shapley Interaction Index [20]
STI	Shapley Taylor Interaction Index [14]
SCD	Sampling Contextual Decomposition [26]
SOC	Sampling Occlusion [26]
ANOVA	Analysis of Variance [16]
LIME	Locally Interpretable Model-Agnostic Explanations [38]
SHAP	Shapley Additive Explanations [32]
GA2M	Generalized Additive Model with Pairwise Interactions [30]
MS COCO	Microsoft Common Objects in Context [29]
SST	Stanford Sentiment Treebank [43]
BERT	Bidirectional Encoder Representations from Transformers [13]
COVID	Coronavirus Disease

B Input Dimensionality Reduction

For a black-box model $f : \mathbb{R}^{p'} \to \mathbb{R}$ which takes as input a vector with p' dimensions (e.g. an image, input embedding, etc.) and maps it to a scalar output (e.g. a class logit), we can make `ArchDetect` more efficient by operating on a lower dimensional input encoding $\mathbf{x} \in \mathbb{R}^p$ with p dimensions. To match the dimensionality p' of the input argument of f, we define a transformation function $\xi : \mathbb{R}^p \to \mathbb{R}^{p'}$ which takes the input encoding $\mathbf{x}$ in the lower dimensional space p and brings it back to the input space of f with dimensionality p'. In other words, (4) becomes

$$\omega_{i,j}(\mathbf{x}) = \left(\frac{1}{h_i h_j}\left(f'(\mathbf{x}^{\star}_{\{i,j\}} + \mathbf{x}_{\setminus\{i,j\}}) - f'(\mathbf{x}'_{\{i\}} + \mathbf{x}^{\star}_{\{j\}} + \mathbf{x}_{\setminus\{i,j\}}) - f'(\mathbf{x}^{\star}_{\{i\}} + \mathbf{x}'_{\{j\}} + \mathbf{x}_{\setminus\{i,j\}}) + f'(\mathbf{x}'_{\{i,j\}} + \mathbf{x}_{\setminus\{i,j\}})\right)\right)^2,$$

where $f' = f \circ \xi$. Correspondingly, `ArchAttribute` (2) becomes

$$\phi(\mathcal{I}) = f'(\mathbf{x}^{\star}_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}) - f'(\mathbf{x}').$$

Examples of input encodings are discussed for the following data types:

- For an image, we use a superpixel segmenter, which selects regions on the image. The selection is covered by the vector $\mathbf{x} \in \{0, 1\}^p$, which encodes which image segments have been selected. Note that wherever $\mathbf{x}$ is 0 corresponds to a baseline feature value (e.g. zeroed image pixels).
- For text, we use the natural correspondence between an input embedding and a word token. The selection of input embedding vectors is also covered by the vector $\mathbf{x} \in \{0, 1\}^p$.
- For recommendation data, we use the same type of correspondence between an input embedding and a feature field.

Similar notions of input encodings have also been used in [38, 47].

C Completeness Axiom

Lemma 2 (Completeness on $\mathcal{S}$)**.** *The sum of all attributions by* `ArchAttribute` *for the disjoint sets in $\mathcal{S}$ equals the difference of f between $\mathbf{x}^\star$ and the baseline $\mathbf{x}'$: $f(\mathbf{x}^\star) - f(\mathbf{x}')$.*

Proof. Based on the definition of non-additive statistical interaction (Def. 1), a function f can be represented as a generalized additive function [48–50], here on the domain of $\mathcal{X}$:

$$f(\mathbf{x}) = \sum_{i=1}^{\eta} q_i(\mathbf{x}_{\mathcal{I}_i^u}) + \sum_{j=1}^{p} q'_j(x_j) + b, \tag{6}$$

where $q_i(\mathbf{x}_{\mathcal{I}_i^u})$ is a function of each interaction $\mathcal{I}_i^u$ on $\mathcal{X}$ $\forall i = 1, \ldots, \eta$ interactions, $q'_j(x_j)$ is a function for each feature $\forall j = 1, \ldots, p$, and b is a bias. The u in $\mathcal{I}^u$ stands for "unmerged".

The disjoint sets of $\mathcal{S} = \{\mathcal{I}_i\}_{i=1}^k$ are the result of merging overlapping interaction sets and main effect sets, so we can merge the subfunctions $q(\cdot)$ and $q'(\cdot)$ of (6) whose input sets overlap to write $f(\mathbf{x})$ as a sum of new functions $g_i(\mathbf{x}_{\mathcal{I}_i})$ $\forall i = 1, \ldots, k$:

$$f(\mathbf{x}) = \sum_{i=1}^{k} g_i(\mathbf{x}_{\mathcal{I}_i}) + b. \tag{7}$$

For some $\{g_i\}_{i=1}^k$ of the form of (7), we rewrite (2) by separating out the effect of index i:

$$\begin{aligned}
\phi(\mathcal{I}_i) &= f(\mathbf{x}^\star_{\mathcal{I}_i} + \mathbf{x}'_{\setminus\mathcal{I}_i}) - f(\mathbf{x}') \quad \forall i = 1, \ldots, k \\
&= \left(g_i(\mathbf{x}^\star_{\mathcal{I}_i}) + \sum_{\substack{j=1 \\ j\neq i}}^{k} g_j(\mathbf{x}'_{\mathcal{I}_j}) + b \right) - \left(g_i(\mathbf{x}'_{\mathcal{I}_i}) + \sum_{\substack{j=1 \\ j\neq i}}^{k} g_j(\mathbf{x}'_{\mathcal{I}_j}) + b \right) && (8) \\
&= g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i}). && (9)
\end{aligned}$$

Since all $\mathcal{I} \in \mathcal{S}$ are disjoint, $g_j(\mathbf{x}'_{\mathcal{I}_j})$ can be canceled in (8) $\forall j$, leading to (9). The result at (9) can also be obtained with an alternative attribution approach, as shown in Corollary 6.

Next, we compute the sum of attributions:

$$\begin{aligned}
\sum_{i=1}^{k} \phi(\mathcal{I}_i) &= \sum_{i=1}^{k} \left(g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i}) \right) && (10) \\
&= \sum_{i=1}^{k} g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - \sum_{i=1}^{k} g_i(\mathbf{x}'_{\mathcal{I}_i}) && (11) \\
&= f(\mathbf{x}^\star) - f(\mathbf{x}')
\end{aligned}$$

□

D Completeness of a Complementary Attribution Method

Corollary 6 (Completeness of a Complement)**.** *An attribution approach: $\phi(\mathcal{I}) = f(\mathbf{x}^\star) - f(\mathbf{x}'_{\mathcal{I}} + \mathbf{x}^\star_{\setminus\mathcal{I}})$, similar to what is mentioned in [26, 28], also satisfies the completeness axiom.*

Proof. Based on Eqs. 7 - 9 of Lemma 2:

$$\begin{aligned}
\phi(\mathcal{I}_i) &= f(\mathbf{x}^\star) - f(\mathbf{x}'_{\mathcal{I}_i} + \mathbf{x}^\star_{\setminus\mathcal{I}_i}) \\
&= \left(g_i(\mathbf{x}^\star_{\mathcal{I}_i}) + \sum_{\substack{j=1 \\ j\neq i}}^{k} g_j(\mathbf{x}^\star_{\mathcal{I}_j}) + b \right) - \left(g_i(\mathbf{x}'_{\mathcal{I}_i}) + \sum_{\substack{j=1 \\ j\neq i}}^{k} g_j(\mathbf{x}^\star_{\mathcal{I}_j}) + b \right) \\
&= g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i})
\end{aligned}$$

We can then resume with (10) of Lemma 2. □

E Set Attribution Axiom

Axiom 3 (Set Attribution). *If $f : \mathbb{R}^p \to \mathbb{R}$ is a function in the form of $f(\mathbf{x}) = \sum_{i=1}^{k} \varphi_i(\mathbf{x}_{\mathcal{I}_i})$ where $\{\mathcal{I}_i\}_{i=1}^{k}$ are disjoint and functions $\{\varphi_i(\cdot)\}_{i=1}^{k}$ have roots, then an interaction attribution method admits an attribution for feature set $\mathcal{I}_i$ as $\varphi_i(\mathbf{x}_{\mathcal{I}_i})$ $\forall i = 1, \ldots, k$.*

Lemma 4 (Set Attribution on $\mathcal{S}$). *For $\mathbf{x} = \mathbf{x}^\star$ and a baseline $\mathbf{x}'$ such that $\varphi_i(\mathbf{x}'_{\mathcal{I}_i}) = 0$ $\forall i = 1, \ldots, k$,* `ArchAttribute` *satisfies the Set Attribution axiom and provides attribution $\varphi_i(\mathbf{x}_{\mathcal{I}_i})$ for set $\mathcal{I}_i$ $\forall i$.*

Proof. From (9) in Lemma 2, `ArchAttribute` can be written as

$$\phi(\mathcal{I}_i) = g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i}) \quad \forall i = 1, \ldots, k,$$

where $f(\mathbf{x}) = \sum_{i=1}^{k} g_i(\mathbf{x}_{\mathcal{I}_i}) + b$. Since $\mathcal{S} = \{\mathcal{I}_i\}_{i=1}^{k}$ are disjoint feature sets for the same function f in Axiom 3, $g_i(\cdot)$ and $\varphi_i(\cdot)$ are related by a constant bias b_i:

$$\varphi_i(\mathbf{x}) = g_i(\mathbf{x}) + b_i$$

Each $\varphi_i(\cdot)$ has roots, so $g_i(\mathbf{x}) + b_i$ has roots. $\mathbf{x}'$ is set such that $\varphi_i(\mathbf{x}'_{\mathcal{I}_i}) = g_i(\mathbf{x}'_{\mathcal{I}_i}) + b_i = 0$. Rearranging,

$$-g_i(\mathbf{x}'_{\mathcal{I}_i}) = b_i.$$

Adding $g_i(\mathbf{x}^\star_{\mathcal{I}_i})$ to both sides,

$$g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i}) = g_i(\mathbf{x}^\star_{\mathcal{I}_i}) + b_i,$$

which becomes

$$\phi(\mathcal{I}_i) = \varphi_i(\mathbf{x}^\star_{\mathcal{I}_i}) \quad \forall i = 1, \ldots, k.$$

□

E.1 Set Attribution Counterexamples

We now provide counterexamples to identify situations in which the related methods do not satisfy the Set Attribution axiom.

Let

$$f(\mathbf{x}) = \text{ReLU}(x_1 + x_3 + 1) + \text{ReLU}(x_2) + 1.$$

$f(\mathbf{x})$ can be written as $f(\mathbf{x}) = \varphi_1(\mathbf{x}_{\{1,3\}}) + \varphi_2(\mathbf{x}_{\{2\}})$ where $\varphi_1(\mathbf{x}) = \text{ReLU}(x_1 + x_3 + 1)$, and $\varphi_2(\mathbf{x}) = \text{ReLU}(x_2) + 1$. According to the Set Attribution axiom, an interaction attribution method admits attributions as

- $\text{ReLU}(x_1 + x_3 + 1)$ for features $\mathcal{I}_1 = \{1, 3\}$
- $\text{ReLU}(x_2) + 1$ for feature $\mathcal{I}_2 = \{2\}$.

The above setting serves as counterexamples to the related methods as follows:

- CD always assigns $\alpha + \frac{\alpha}{\alpha+\beta}$ to $\mathcal{I}_1$ and $\beta + \frac{\beta}{\alpha+\beta}$ to $\mathcal{I}_2$, where $\alpha = \text{ReLU}(x_1 + x_3 + 1)$ and $\beta = \text{ReLU}(x_2)$.
- SCD uses an expectation over an activation decomposition, which does not guarantee admission of $\text{ReLU}(x_1 + x_3 + 1)$ for $\mathcal{I}_1$ and $\text{ReLU}(x_2)$ for $\mathcal{I}_2$ through their respective decompositions. In the ideal case SCD becomes CD, which still does not satisfy Set Attribution from above.
- IH always assigns a zero attribution to $\mathcal{I}_2$ from hessian computations. IH also does not assign attributions to general sets of features.
- SOC does not assign attributions to general feature sets, only contiguous feature sequences.
- Both SI and STI assign the following attribution score to $\mathcal{I}_1$:

  $$\text{ReLU}(x_1 + x_3 + 1) - \text{ReLU}(x_1 + x'_3 + 1) - \text{ReLU}(x'_1 + x_3 + 1) + \text{ReLU}(x'_1 + x'_3 + 1). \tag{12}$$

  There do not exist a selection of x'_1 and x'_3 such that this attribution becomes $\text{ReLU}(x_1 + x_3 + 1)$ for all values of x_1 and x_3.

Proof. We prove via case-by-case contradiction. Only the $\text{ReLU}(x_1 + x_3 + 1)$ term can create an interaction between x_1 and x_3, and this term is also the target result, so any nonzero deviation from this term via independent x_1 or x_3 effects in (12) must be countered. These independent effects manifest as the $\text{ReLU}(x_1 + x_3' + 1)$ or $\text{ReLU}(x_1' + x_3 + 1)$ terms respectively. Since ReLU is always non-negative, the only way either of these terms is nonzero is if it is positive, which implies that $\text{ReLU}(x_1 + x_3' + 1) = x_1 + x_3' + 1$ or $\text{ReLU}(x_1' + x_3 + 1) = x_1' + x_3 + 1$. If both terms are positive, their substitution into (12) yields $\text{ReLU}(x_1 + x_3 + 1) - x_1 - x_3' - 1 - x_1' - x_3 - 1 + \text{ReLU}(x_1' + x_3' + 1)$. Even if $\text{ReLU}(x_1' + x_3' + 1)$ is positive, we obtain $\text{ReLU}(x_1 + x_3 + 1) - x_1 - x_3' - 1 - x_1' - x_3 - 1 + x_1' + x_3' + 1 = \text{ReLU}(x_1 + x_3 + 1) - x_1 - x_3 - 1$. Asserting $-x_1 - x_3 - 1 = 0$ is a contradiction. If only one of the independent effects was positive, we also cannot assert 0 through similar simplifications.

Now consider the remaining case where $\text{ReLU}(x_1 + x_3' + 1) = \text{ReLU}(x_1' + x_3 + 1) = \text{ReLU}(x_1' + x_3' + 1) = 0$. For any real-valued x_1' or x_3' , there can also be a negative real-valued x_3 or x_1 respectively. From either terms $\text{ReLU}(x_1 + x_3' + 1)$ or $\text{ReLU}(x_1' + x_3 + 1)$, we obtain $\text{ReLU}(1) = 0$, which is a contradiction. □

F Other Axioms

F.1 Sensitivity Axiom

Lemma 7 (Sensitivity (a))**.** *If $\mathbf{x}^\star$ and $\mathbf{x}'$ only differ at features indexed in $\mathcal{I}$ and $f(\mathbf{x}^\star) \neq f(\mathbf{x}')$, then $\phi(\mathcal{I})$ (2) yields a nonzero attribution.*

Proof. Since $\mathbf{x}^\star$ and $\mathbf{x}'$ only differ at $\mathcal{I}$, the following is true: $\mathbf{x}^\star_{\setminus\mathcal{I}} = \mathbf{x}'_{\setminus\mathcal{I}}$. We can therefore write $\mathbf{x}^\star$ as

$$\begin{aligned}\mathbf{x}^\star &= \mathbf{x}^\star_{\mathcal{I}} + \mathbf{x}^\star_{\setminus\mathcal{I}} \\ &= \mathbf{x}^\star_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}\end{aligned}$$

Substituting this equivalence in (2), we have

$$\begin{aligned}\phi(\mathcal{I}) &= f(\mathbf{x}^\star_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}) - f(\mathbf{x}') \\ &= f(\mathbf{x}^\star) - f(\mathbf{x}').\end{aligned}$$

Since $f(\mathbf{x}^\star) - f(\mathbf{x}') \neq 0$, we directly obtain $\phi(\mathcal{I}) \neq 0$.

□

Lemma 8 (Sensitivity (b))**.** *If f does not functionally depend on $\mathcal{I}$, then $\phi(\mathcal{I})$ is always zero.*

Proof. Since f does not functionally depend on $\mathcal{I}$,

$$\begin{aligned}f(\mathbf{x}^\star_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}) &= f(\mathbf{x}'_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}) \\ &= f(\mathbf{x}')\end{aligned}$$

Therefore,

$$\phi(\mathcal{I}) = f(\mathbf{x}^\star_{\mathcal{I}} + \mathbf{x}'_{\setminus\mathcal{I}}) - f(\mathbf{x}') = 0.$$

□

F.2 Implementation Invariance

Lemma 9 (Implementation Invariance)**.** *For functionally equivalent models (with the same input-output mapping), $\phi(\cdot)$ are the same.*

The definition of (2) only relies on function calls to f, which implies Implementation Invariance.

F.3 Linearity

Lemma 10 (Linearity on $\mathcal{S}$)**.** *If two models f_1, f_2 have the same disjoint feature sets $\mathcal{S}$ and $f = c_1 f_1 + c_2 f_2$ where c_1, c_2 are constants, then $\phi(\mathcal{I}) = c_1\phi_1(\mathcal{I}) + c_2\phi_2(\mathcal{I})\ \forall \mathcal{I} \in \mathcal{S}$.*

Proof. Since f_1 and f_2 have the same $\mathcal{S} = \{\mathcal{I}_i\}_{i=1}^k$, we can write f_1 and f_2 as follows via (7) in Lemma 2:

$$f_1(\mathbf{x}) = \sum_{i=1}^{k} g_i^{(1)}(\mathbf{x}_{\mathcal{I}_i}) + b^{(1)},$$
$$f_2(\mathbf{x}) = \sum_{i=1}^{k} g_i^{(2)}(\mathbf{x}_{\mathcal{I}_i}) + b^{(2)}.$$

Since $f = c_1 f_1 + c_2 f_2$,

$$\begin{aligned} f(\mathbf{x}) &= c_1 f_1(\mathbf{x}) + c_2 f_2(\mathbf{x}) \\ &= \left(\sum_{i=1}^{k} c_1 \times g_i^{(1)}(\mathbf{x}_{\mathcal{I}_i}) + c_1 \times b^{(1)}\right) + \left(\sum_{i=1}^{k} c_2 \times g_i^{(2)}(\mathbf{x}_{\mathcal{I}_i}) + c_2 \times b^{(2)}\right) \\ &= \sum_{i=1}^{k} \left(c_1 \times g_i^{(1)}(\mathbf{x}_{\mathcal{I}_i}) + c_2 \times g_i^{(2)}(\mathbf{x}_{\mathcal{I}_i})\right) + c_1 b^{(1)} + c_2 b^{(2)}. \end{aligned} \tag{13}$$

By grouping terms as $g_i(\mathbf{x}_{\mathcal{I}_i}) = c_1 \times g_i^{(1)}(\mathbf{x}_{\mathcal{I}_i}) + c_2 \times g_i^{(2)}(\mathbf{x}_{\mathcal{I}_i})$ and $b = c_1 b^{(1)} + c_2 b^{(2)}$, we write (13) as

$$f(\mathbf{x}) = \sum_{i=1}^{k} g_i(\mathbf{x}_{\mathcal{I}_i}) + b. \tag{14}$$

From the form of (14), we can invoke (9): $\phi(\mathcal{I}_i) = g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i})$ via Lemma 2. This equation is rewritten as

$$\begin{aligned} \phi(\mathcal{I}_i) &= g_i(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i(\mathbf{x}'_{\mathcal{I}_i}) \\ &= \left(c_1 \times g_i^{(1)}(\mathbf{x}^\star_{\mathcal{I}_i}) + c_2 \times g_i^{(2)}(\mathbf{x}^\star_{\mathcal{I}_i})\right) - \left(c_1 \times g_i^{(1)}(\mathbf{x}'_{\mathcal{I}_i}) + c_2 \times g_i^{(2)}(\mathbf{x}'_{\mathcal{I}_i})\right) \\ &= c_1 \left(g_i^{(1)}(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i^{(1)}(\mathbf{x}'_{\mathcal{I}_i})\right) + c_2 \left(g_i^{(2)}(\mathbf{x}^\star_{\mathcal{I}_i}) - g_i^{(2)}(\mathbf{x}'_{\mathcal{I}_i})\right) \\ &= c_1\phi_1(\mathcal{I}_i) + c_2\phi_2(\mathcal{I}_i). \end{aligned}$$

By noting that $\mathcal{S} = \{\mathcal{I}_i\}_{i=1}^k$, this concludes the proof.

□

F.4 Symmetry-Preserving

We first define *symmetric feature sets* as a generalization of "symmetric variables" from [46]. Feature index sets $\mathcal{I}_1$ and $\mathcal{I}_2$ are symmetric with respect to function f if swapping features in $\mathcal{I}_1$ with the features in $\mathcal{I}_2$ does not change the function, This implies that for symmetric $\mathcal{I}_1$ and $\mathcal{I}_2$, their cardinalities are the same $|\mathcal{I}_1| = |\mathcal{I}_2|$, and they are disjoint sets in order to swap the features to any valid set index.

Lemma 11 (Symmetry-Preserving)**.** *For $\mathbf{x}^\star$ and $\mathbf{x}'$ that each have identical feature values between symmetric feature sets with respect to f, the symmetric feature sets receive identical attributions $\phi(\cdot)$.*

Proof. Since $\mathbf{x}^\star$ and $\mathbf{x}'$ each have identical feature values between the symmetric feature sets,

$$\{x_i^\star\}_{i\in\mathcal{I}_1} = \{x_j^\star\}_{j\in\mathcal{I}_2},$$
$$\{x'_i\}_{i\in\mathcal{I}_1} = \{x'_j\}_{j\in\mathcal{I}_2}.$$

Therefore, the symmetry implies the following for any $\mathbf{x}$ in the domain of f.

$$f\left(\mathbf{x}^{\star}_{\mathcal{I}_1} + \mathbf{x}'_{\mathcal{I}_2} + \mathbf{x}_{\setminus(\mathcal{I}_1 \cup \mathcal{I}_2)}\right) = f\left(\mathbf{x}'_{\mathcal{I}_1} + \mathbf{x}^{\star}_{\mathcal{I}_2} + \mathbf{x}_{\setminus(\mathcal{I}_1 \cup \mathcal{I}_2)}\right) \tag{15}$$

Setting $\mathbf{x} = \mathbf{x}'$, we rewrite (15) as

$$\begin{aligned}
&f\left(\mathbf{x}^{\star}_{\mathcal{I}_1} + \mathbf{x}'_{\mathcal{I}_2} + \mathbf{x}'_{\setminus(\mathcal{I}_1 \cup \mathcal{I}_2)}\right) - f\left(\mathbf{x}'_{\mathcal{I}_1} + \mathbf{x}^{\star}_{\mathcal{I}_2} + \mathbf{x}'_{\setminus(\mathcal{I}_1 \cup \mathcal{I}_2)}\right) = 0 \\
&= f(\mathbf{x}^{\star}_{\mathcal{I}_1} + \mathbf{x}'_{\setminus_{\mathcal{I}_1}}) - f(\mathbf{x}^{\star}_{\mathcal{I}_2} + \mathbf{x}'_{\setminus_{\mathcal{I}_2}}) \\
&= \left(f(\mathbf{x}^{\star}_{\mathcal{I}_1} + \mathbf{x}'_{\setminus_{\mathcal{I}_1}}) - f(\mathbf{x}')\right) - \left(f(\mathbf{x}^{\star}_{\mathcal{I}_2} + \mathbf{x}'_{\setminus_{\mathcal{I}_2}}) - f(\mathbf{x}')\right) \\
&= \phi(\mathcal{I}_1) - \phi(\mathcal{I}_2)
\end{aligned}$$

Therefore, $\phi(\mathcal{I}_1) = \phi(\mathcal{I}_2)$.

□

G Discrete Mixed Partial Derivatives Detect Non-Additive Statistical Interactions

A generalized additive model f_g is given by

$$f_g(\mathbf{x}) = \sum_{i=1}^{p} g_i(x_i) + b, \tag{16}$$

where $g_i(\cdot)$ can be any function of individual features x_i and b is a bias. Since each x_i of $\mathbf{x} \in \mathcal{X}$ only takes on two values, a line can connect all valid points in each feature. Therefore, (16) is equivalent to

$$f_\ell(\mathbf{x}) = \sum_{i=1}^{p} w_i x_i + b, \tag{17}$$

for weights $w_i \in \mathbb{R}$ and the function domain being $\mathcal{X}$.

For the case where $p = 2$, the discrete mixed partial derivative is given by (3) or

$$\frac{\partial^2 f}{\partial x_1 \partial x_2} = \frac{1}{h_1 h_2}\left(f([x_1^{\star}, x_2^{\star}]) - f([x_1^{\star}, x_2']) - f([x_1', x_2^{\star}]) + f([x_1', x_2'])\right),$$

where $h_1 = |x_1^{\star} - x_1'|$ and $h_2 = |x_2^{\star} - x_2'|$. Since any three points (not on the same line) define a plane of the form (17) ($p = 2$), we can write the fourth point as having a function value with deviation δ from the plane.

$$\begin{aligned}
\frac{\partial^2 f}{\partial x_1 \partial x_2} &= \frac{1}{h_1 h_2}\left(f([x_1^{\star}, x_2^{\star}]) - f([x_1^{\star}, x_2']) - f([x_1', x_2^{\star}]) + f([x_1', x_2'])\right) \\
&= \frac{1}{h_1 h_2}\left((w_1 x_1^{\star} + w_2 x_2^{\star} + b + \delta) - (w_1 x_1^{\star} + w_2 x_2' + b) - (w_1 x_1' + w_2 x_2^{\star} + b)\right. \\
&\qquad\qquad \left. + (w_1 x_1' + w_2 x_2' + b)\right) \\
&= \frac{\delta}{h_1 h_2}.
\end{aligned} \tag{18}$$

If (18) is 0, then $\delta = 0$, which implies that f can be written as (17). $\delta \neq 0$ implies the opposite, that f cannot be written in linear form (by definition). Since (17) is equivalent to (16) in the domain of $\mathcal{X}$, this implies that $\delta \neq 0$ if and only if $f(\mathbf{x}) \neq g_1(x_1) + g_2(x_2) + b$.

Based on Def. 1, we can conclude that a nonzero discrete mixed partial derivative w.r.t. x_1 and x_2 in the space $\mathcal{X}$ at $p = 2$ detects a non-additive statistical interaction between the two features.

For the case where $p > 2$, Def. 1 states that a pairwise interaction $\{i, j\}$ exists in f if and only if $f(\mathbf{x}) \neq f_i(\mathbf{x}_{\setminus\{i\}}) + f_j(\mathbf{x}_{\setminus\{j\}})$ for functions $f_i(\cdot)$ and $f_j(\cdot)$. This means that $\{i, j\}$ is declared to be an interaction if a local $\{i, j\}$ interaction occurs at any $\mathbf{x}_{\setminus\{i,j\}}, \mathbf{x} \in \mathcal{X}$.

Therefore, we can detect non-additive statistical interactions $\{i, j\}$ for general $p \geq 2$ via

$$\mathbb{E}_{\mathbf{x}} \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right]^2 > 0,$$

which mirrors the definition of pairwise interaction for real-valued $\mathbf{x}$ in [18].

H Early Works on Feature Interaction Interpretation

We discuss early works on feature interaction interpretation and provide a timeline for this research history in Table 4. We also discuss mixed partial derivatives on dichotomous variables in H.3.

H.1 Origins

The notion of a feature interaction has been studied at least since the 19th century when John Lawes and Joseph Gilbert used factorial designs in agricultural research at the Rothamsted Experimental Station [11]. A factorial design is an experiment that includes observations at all combinations of categories of each factor or feature. However, the "advantages [of factorial design] had never been clearly recognised, and many research workers believed that the best course was the conceptually simple one of investigating one question at a time" [57]. In the early 20th century, Fisher et al. (1926) [17] emphasized the importance of factorial designs as being the only way to obtain information about feature interactions. Near the same time, Fisher (1921) [15] also developed one of the foundations of statistical analysis called Analysis of Variance (ANOVA) including two-way ANOVA [16], which is a factorial method to detect pairwise feature interactions based on differences among group means in a dataset. Tukey (1949) [51] extended two-way ANOVA to test if two categorical features are non-additively related to the expected value of a outcome variable. This work set a precedent for later research on detecting feature interactions based on their non-additive definition. Soon after, experimental designs were generalized to study feature interactions, in particular the generalized randomized block design [54], which assigns test subjects to different categories (or blocks) between features in a way where cross-categories between features serve as interaction terms in linear regression.

There was a surge of interest in improving the analysis of feature interactions after the mid 20th century. Belsion (1959) [5] and Morgan & Sonquist (1963) [34] proposed Automatic Interaction Detection (AID) originally under a different name. AID detects interactions by subdividing data into disjoint exhaustive subsets to model an outcome based on categorical features. Based on AID, Kass (1980) [27] developed Chi-square Automatic Interaction Detection (CHAID), which determines how categorical features best combine in decision trees via a chi-square test. AID and CHAID were precursors to modern decision tree prediction models. Concurrently, Nelder (1977) [36] introduced the "Principle of Marginality" arguing that a feature interaction and its marginal variables should not be considered separately, for example in linear regression. Hamada & Wu (1992) [22] provided a contrasting view that an interaction is only important if one or both of its marginal variables are important. Around the same time, an influential book on interpreting feature interactions was published on how to test, plot, and understand interactions of two or three continuous or categorical features [3].

H.2 Early 21st Century Works

At the start of the 21st century, efforts began to focus on interpreting interactions in accurate prediction models. Ai & Norton (2003) [2] proposed extracting interactions from logit and probit models via mixed partial derivatives. Gevrey (2006) [19] followed up by proposing mixed partial derivatives to extract interactions from multilayer perceptrons with sigmoid activations when at the time, only shallow neural networks were studied. Friedman & Popescu (2008) [18] proposed using hybrid models to capture interactions with decision trees and univariate effects with linear regression. Sorokina et al. (2008) [45] proposed to use high-performance additive trees to detect feature interactions based on their non-additive definition. At the turn of the decade, we saw Bien et al. [6] capture interactions with different heredity conditions using a hierarchical lasso on linear regression models. Then, Hao & Zhang (2014) [23] drew attention towards interaction screening in high dimensional data. This summarizes feature interaction research before 2015.

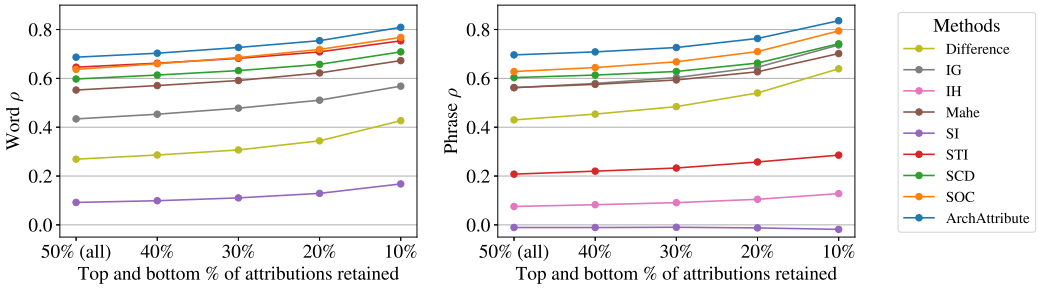
H.3 Note on Mixed Partial Derivatives on Dichotomous Variables

To our knowledge, the usage of mixed partial derivatives for interaction detection on dichotomous variables (features that only take two possible values) originated at the turn of the 21st century [2, 20], but existing methods rely on single contexts [2] or random contexts [14, 20]. Furthermore, these methods do not consider the union of overlapping pairwise interactions for disjoint higher-order interaction detection. Our choice of contexts and our disjoint interaction detection are both important to the `Archipelago` framework, as we discussed in §4.2 and showed through axiomatic analysis (§3.2) and experiments (§5.2).

TABLE 4 Timeline of research on feature interaction interpretation (Pre-2015)

Lawes & Gilbert - factorial design in agricultural research at the Rothamsted Experimental Station	1843	
Fisher - two-way Analysis of Variance (ANOVA)	1925	
	1949	*Tukey* - Tukey's test of additivity
	1955	*Wilk* - generalized random block design
Belson - Automatic Interaction Detection by subdividing data	1959	
Nelder - Principle of Marginality	1977	
	1980	*Kass* - Chi-square Automatic Interaction Detection by combining features in decision trees via chi-square tests
	1991	*Aiken & West* - book on interpreting interaction effects
Hamada & Wu - heredity conditions	1992	
Ai & Norton - interactions in logit and probit models	2003	
	2006	*Gevry et al.* - interactions in sigmoid neural networks
Friedman & Popescu - RuleFit to detect interactions by mixing linear regression and trees	2008	*Sorokina et al.* - Additive Groves to detect non-additive interactions
Bien et al. - Hierarchical Lasso	2013	
Hao & Zhang - interaction screening in high dimensional data	2014	

I Attributions Compared to Annotation Labels



(a) Word ρ (b) Phrase ρ

Figure 7: Text explanation metrics ((a) Word ρ and (b) Phrase ρ) versus top and bottom $\%$ of attributions retained for different attribution methods on BERT over the SST test set. These plots expand the analysis of Table 2.

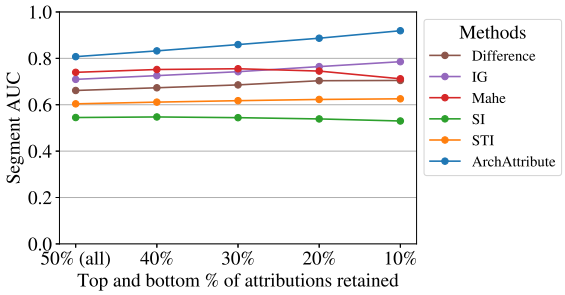
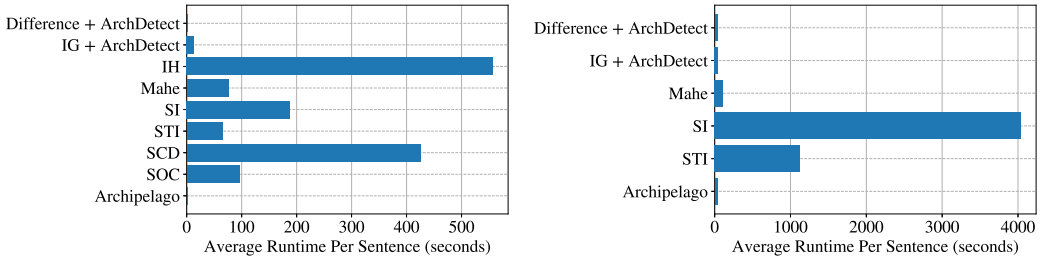


Figure 8: Image explanation metric (segment AUC) versus top and bottom $\%$ of attributions retained for different attribution methods on ResNet152 over the MS COCO test set. These plots expand the analysis of Table 2.

J Runtime



(a) Sentiment Analysis on SST (b) Image Classification on ImageNet

Figure 9: Serial runtime comparison of relevant explainer methods for (a) BERT sentiment analysis on SST and (b) ResNet152 image classification on ImageNet. Runtimes are averaged across 100 random data samples from respective test sets. These experiments were done on a server with 32 Intel Xeon E5-2640 v2 CPUs @ 2.00GHz and 2 Nvidia 1080 Ti GPUs.

K Visualization Comparisons

K.1 Sentiment Analysis

Visualization comparisons of different attribution methods on BERT are shown in Figs. 11-15 for random test sentences from SST. The visualization format is the same as Fig. 4. Note that all *individual* feature attributions that correspond to stop words (from [33]) are omitted in these comparisons and Figs. 1, 4.

K.2 Image Classification

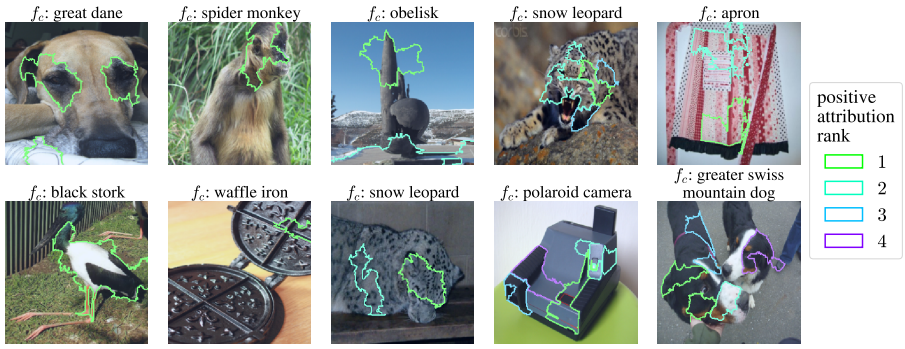


Figure 10: Our ResNet152 visualizations on random test images from ImageNet. Colored outlines indicate interactions with positive attribution. f_c is the image classification result. To our knowledge, only this work shows interactions that support the image classification via interaction attribution.

In Fig. 10, we visualize `Archipelago` explanations on $\mathcal{S}$ via top-5 pairwise interactions (§4.2.2), where positive attribution interactions are shown for clarity. The images are randomly selected from the ImageNet test set. It is interesting to see which image parts interact, such as the eyes of the "great dane" image.

Visualization comparisons of different attribution methods on ResNet152 are shown in Figs. 16-20 for the same random test images from ImageNet.

L `ArchDetect` Ablation Visualizations

We run an ablation study removing the $\mathbf{x}'_{\setminus\{i,j\}}$ baseline context from (5) for disjoint interaction detection and examine its effect on visualizations. The visualizations are shown in Fig. 21 for sentiment analysis and Figs. 22 and 23 for image classification. Top-3 and top-5 pairwise interactions are used in sentiment analysis and image classification respectively before merging the interactions.

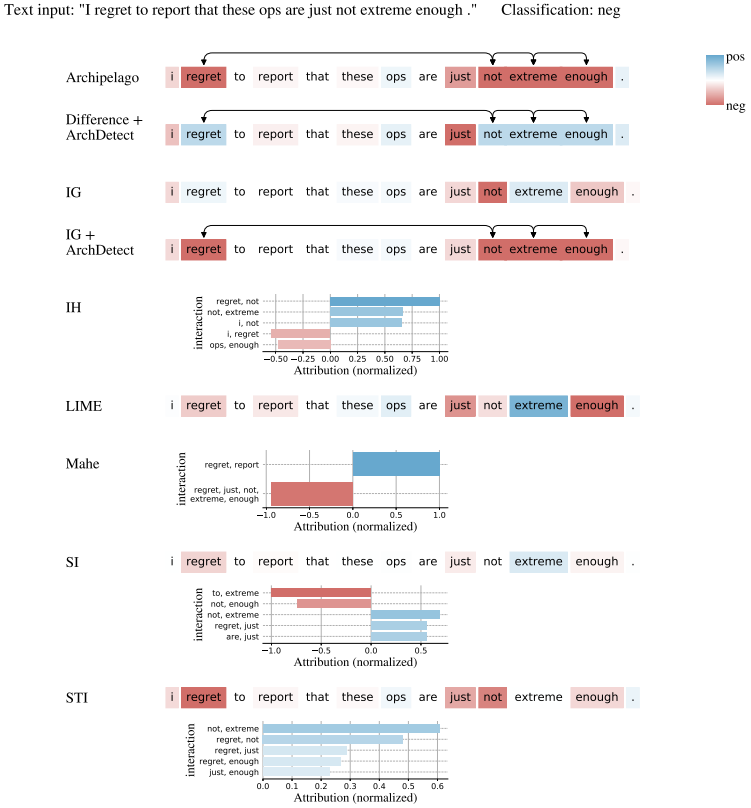


Figure 11: Text Viz. Comparison A. In the first text example, “regret, not extreme enough” is a meaningful and strongly negative interaction. In the second example, “when you begin to” interacts to diminish its overall attribution magnitude.

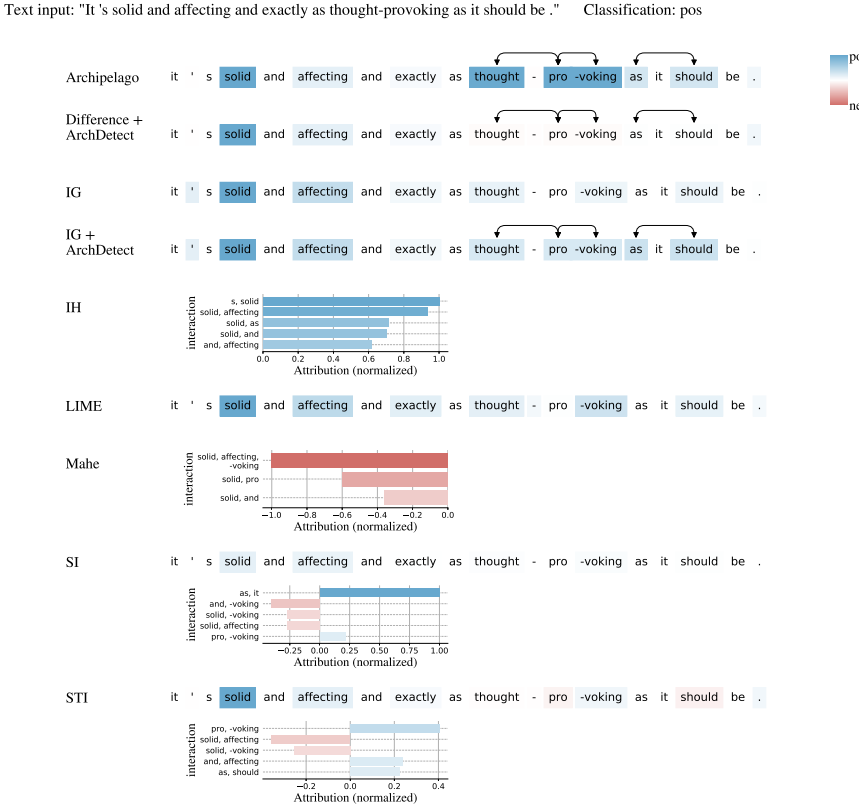


Figure 12: Text Viz. Comparison B. In the first text example, “thought provoking” is a meaningful and strongly positive interaction. In the second example, the “lousy, un” interaction factors in a large context to make a negative text classification.

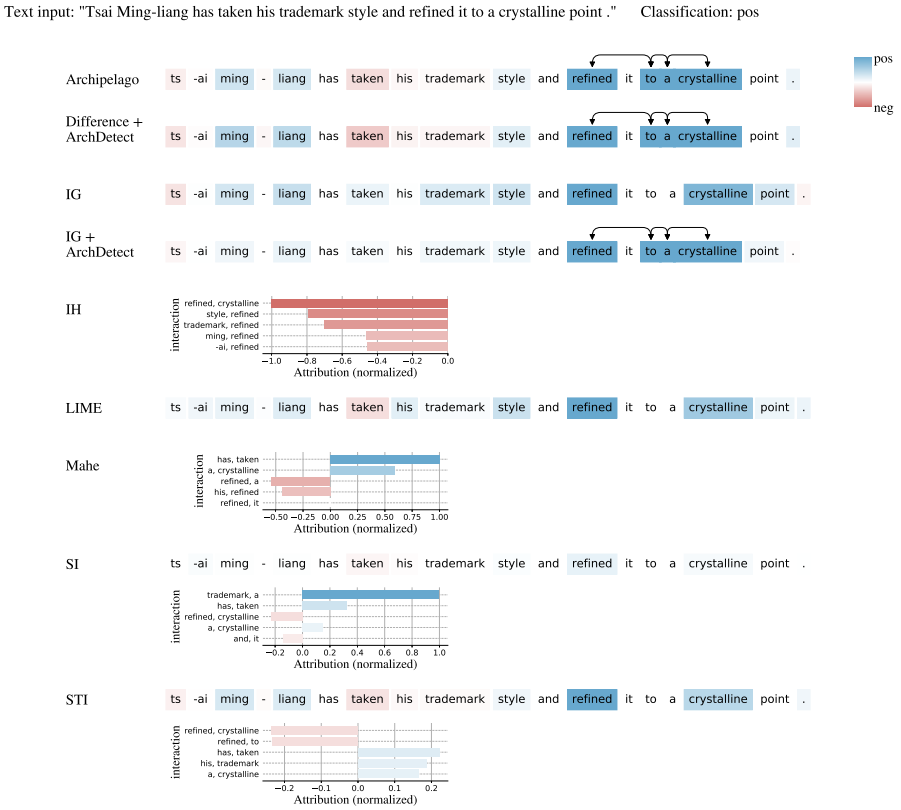


Figure 13: Text Viz. Comparison C. In the first text example, “refined, to a crystalline” is a meaningful and strongly positive interaction. In the second example, “is aptly named” is also a meaningful and strongly positive interaction.

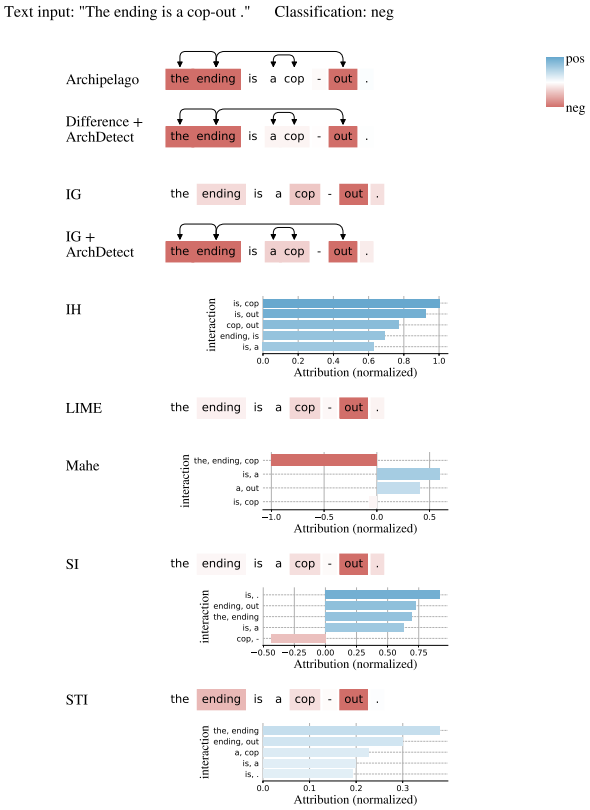


Figure 14: Text Viz. Comparison D. In the first text example, “the ending, out” is a meaningful and negative interaction. In the second example, “a feel good, best” is a meaningful and strongly positive interaction.

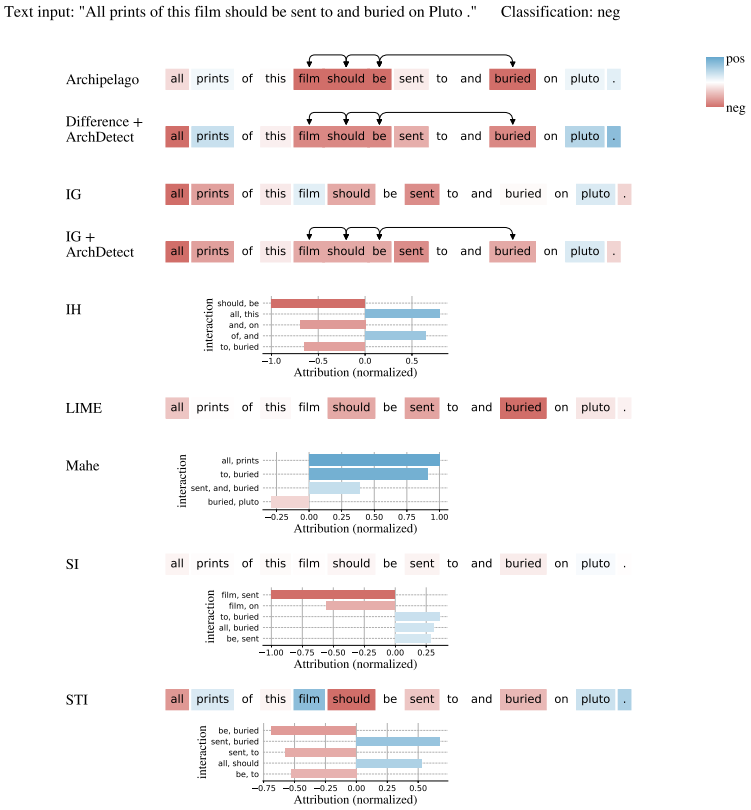


Figure 15: Text Viz. Comparison E. In the first text example, “film should be, buried” is a meaningful and strongly negative interaction. In the second example, “-oherent” belongs to a negative word “incohorent”.

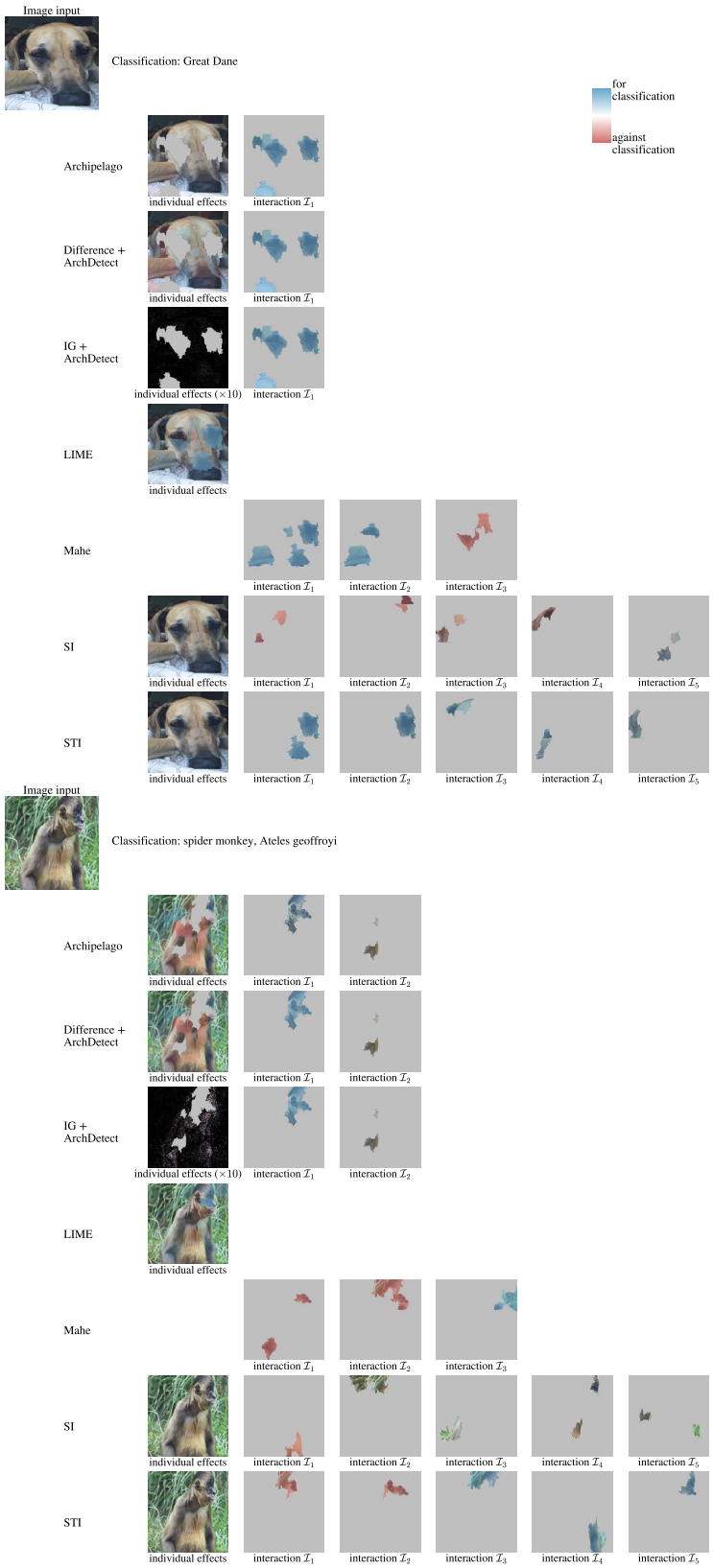


Figure 16: Image Viz. Comparison A. In the first image example, the dog's eyes are a meaningful interaction supporting the classification. In the second example, the monkey's head is also a positive interaction.

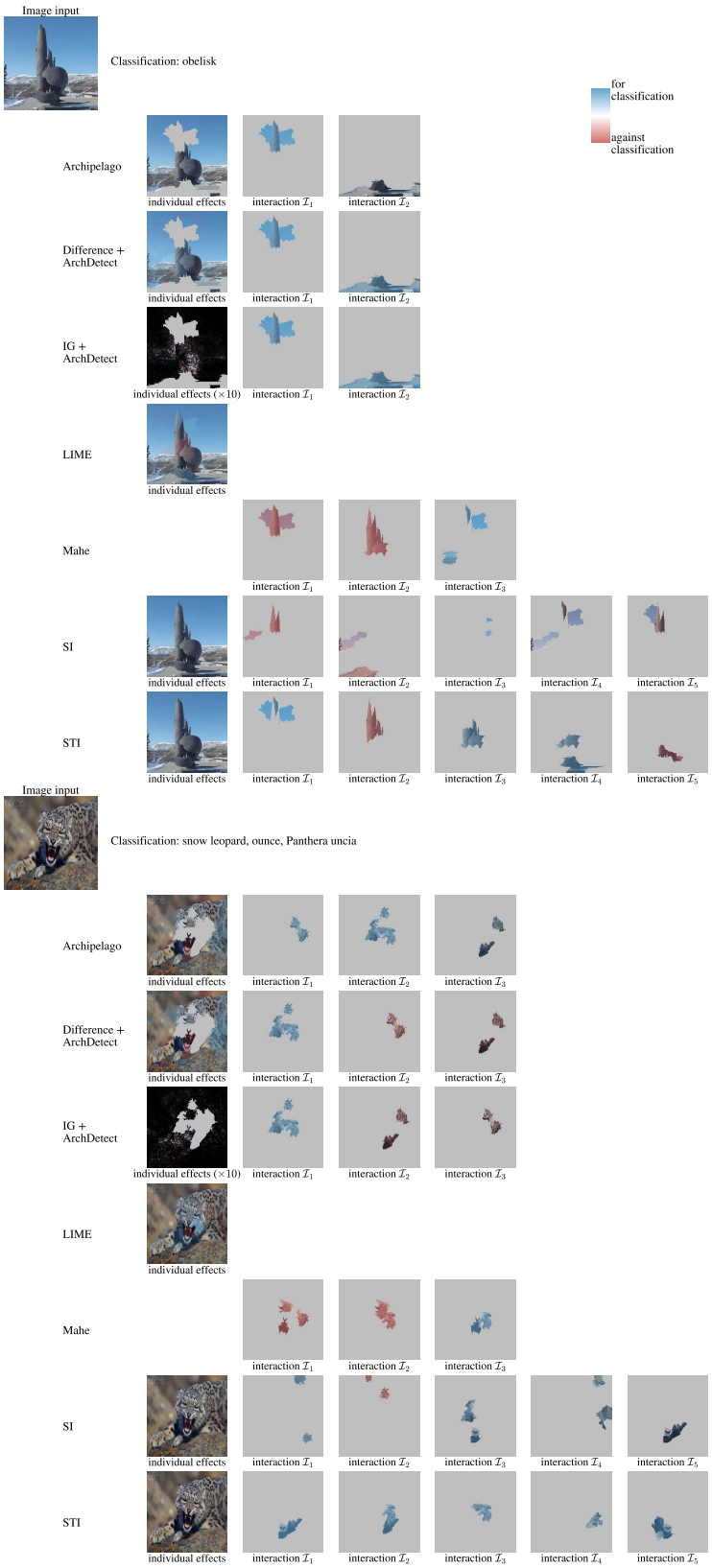


Figure 17: Image Viz. Comparison B. In the first image example, the obelisk tip is a meaningful interaction supporting the classification. In the second example, the leopard's face is also a positive interaction.

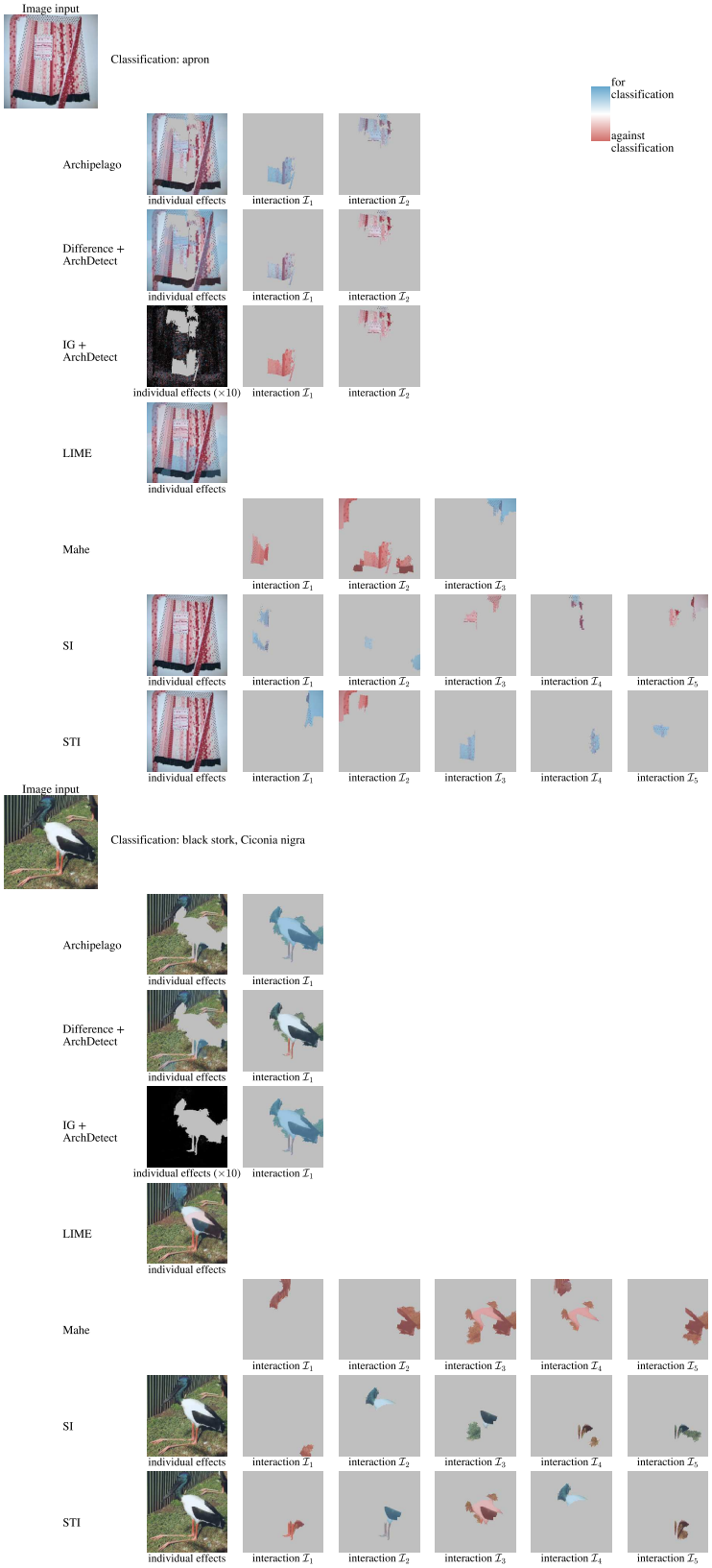


Figure 18: Image Viz. Comparison C. In the first image example, different patches of the apron are interactions supporting the classification. In the second example, the stork's body is an interaction that strongly supports the classification.

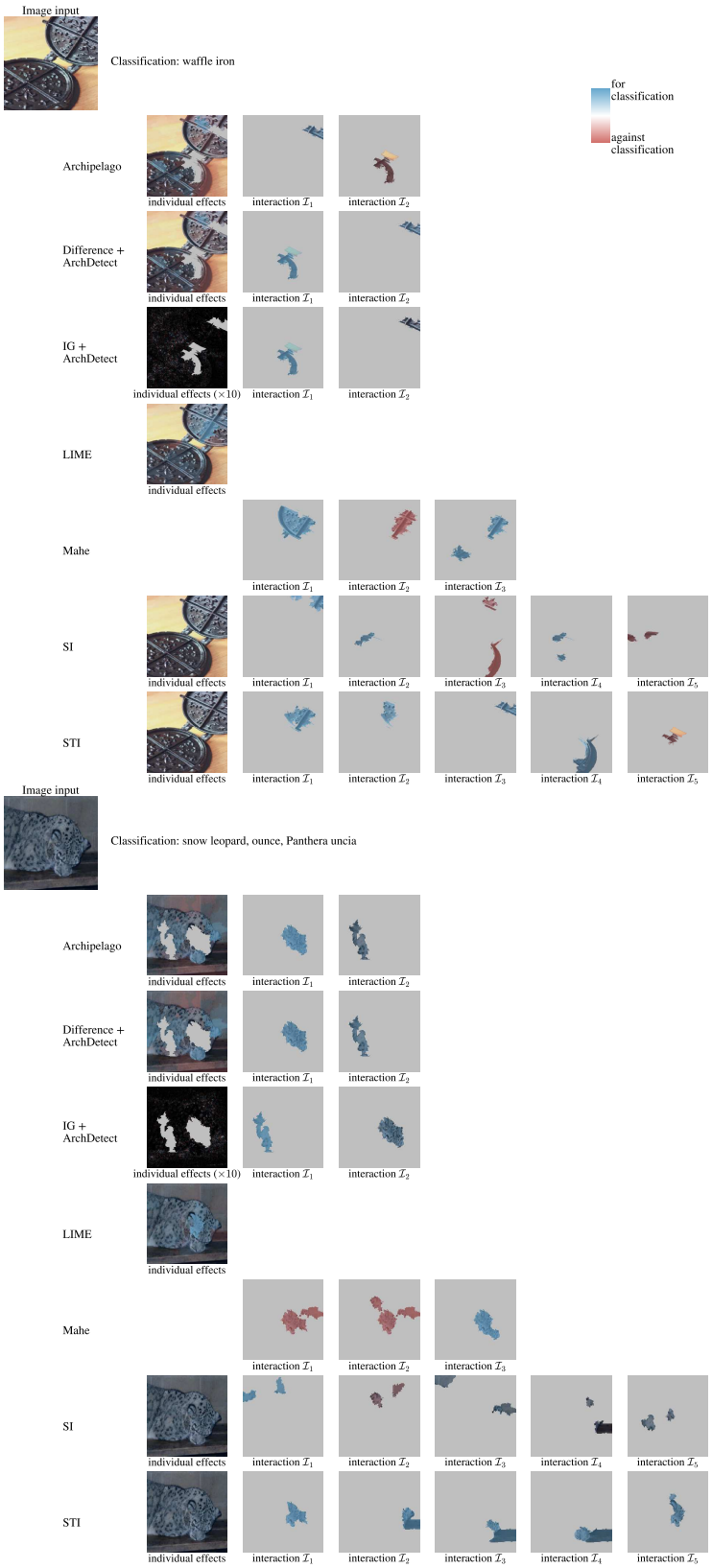


Figure 19: Image Viz. Comparison D. In the first image example, certain small patches of the waffle iron interact, one of which supports the classification. In the second example, the leopard's face is the primary positive interaction.

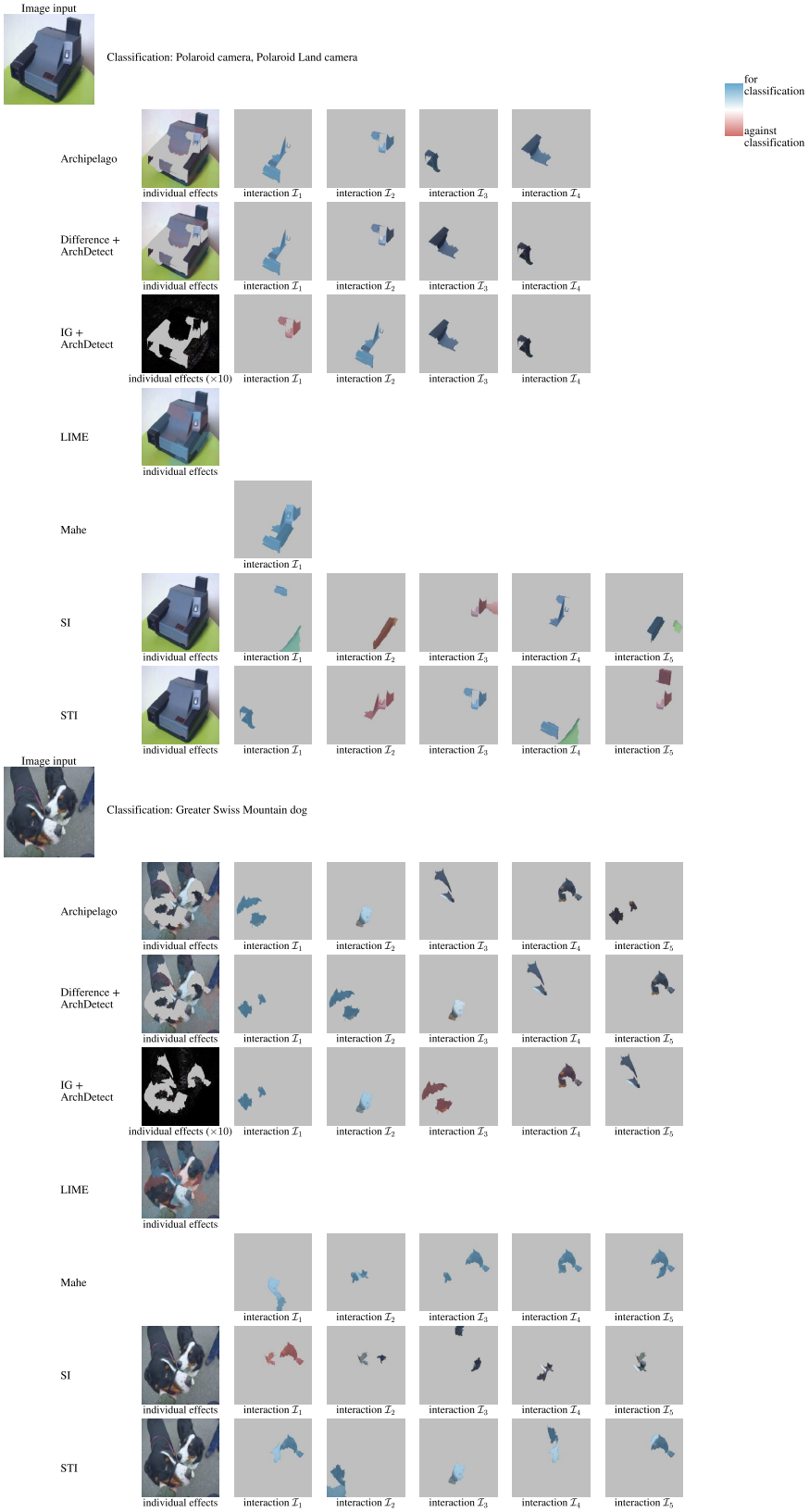


Figure 20: Image Viz. Comparison E. In the first image example, different parts of the polaroid camera are interactions that positively support the classification. In the second example, the dogs' heads and body are also positive interactions.

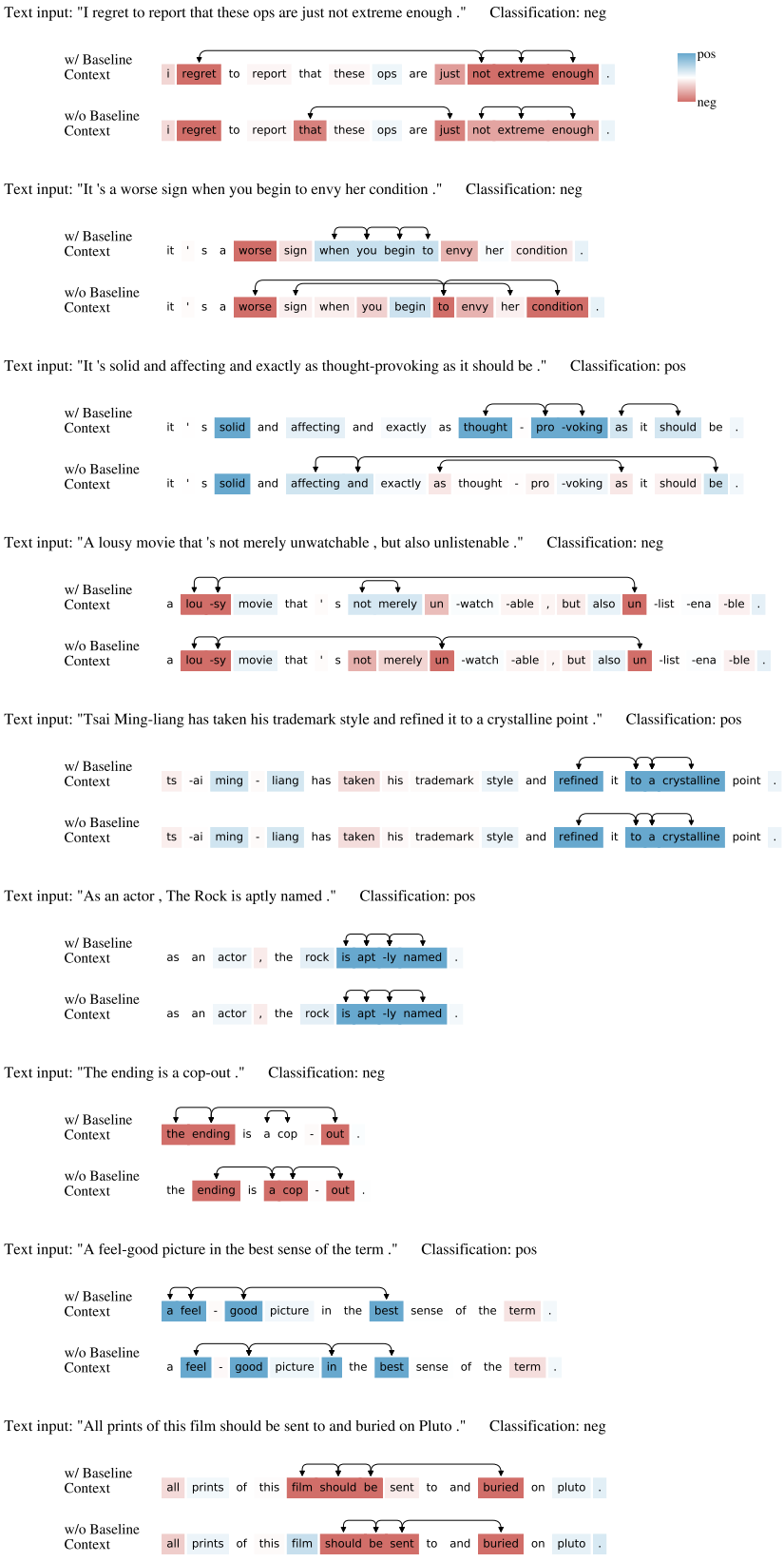


Figure 21: Text Viz. with `ArchDetect` Ablation. The interactions tend to use more salient words when including the baseline context, which is proposed in `ArchDetect`.

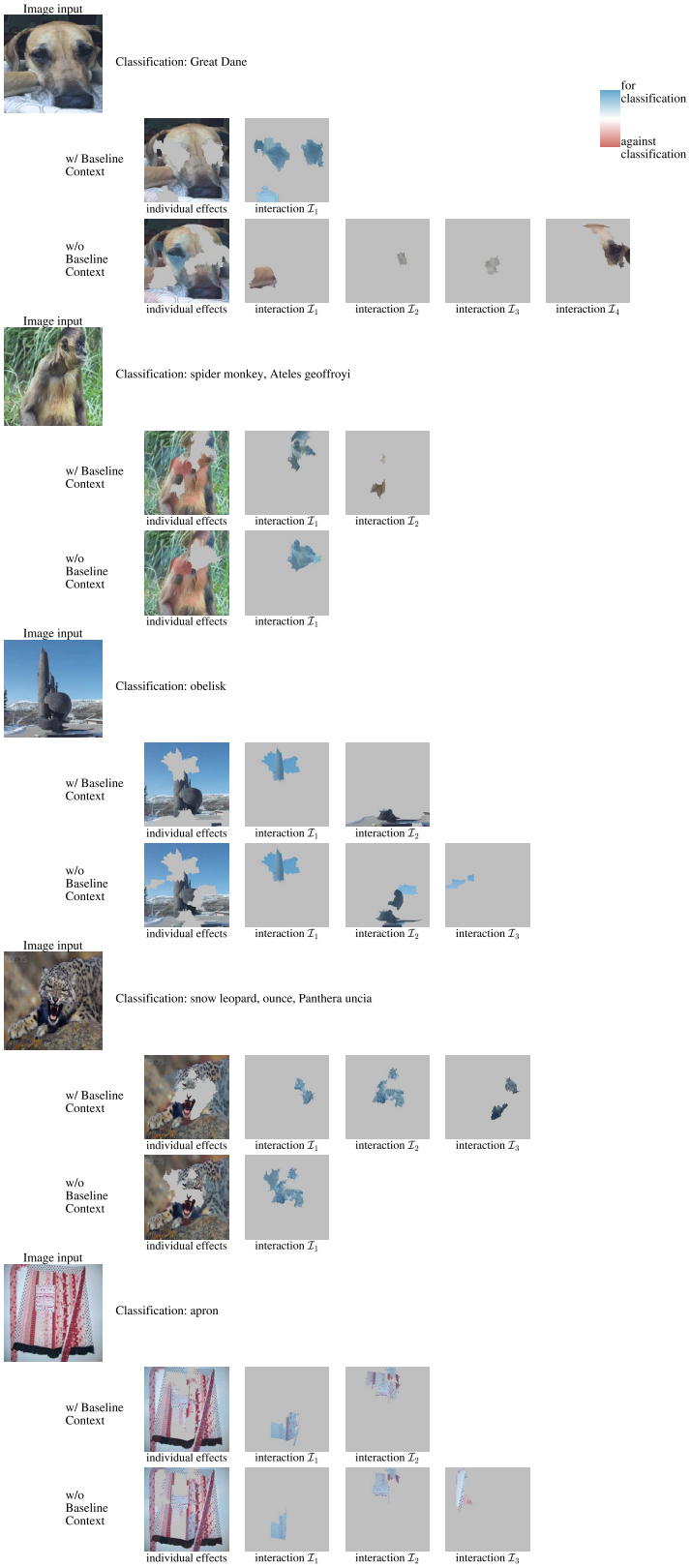


Figure 22: Image Viz. with `ArchDetect` Ablation A. The interactions tend to focus more on salient patches of the images when including the baseline context, which is proposed in `ArchDetect`.

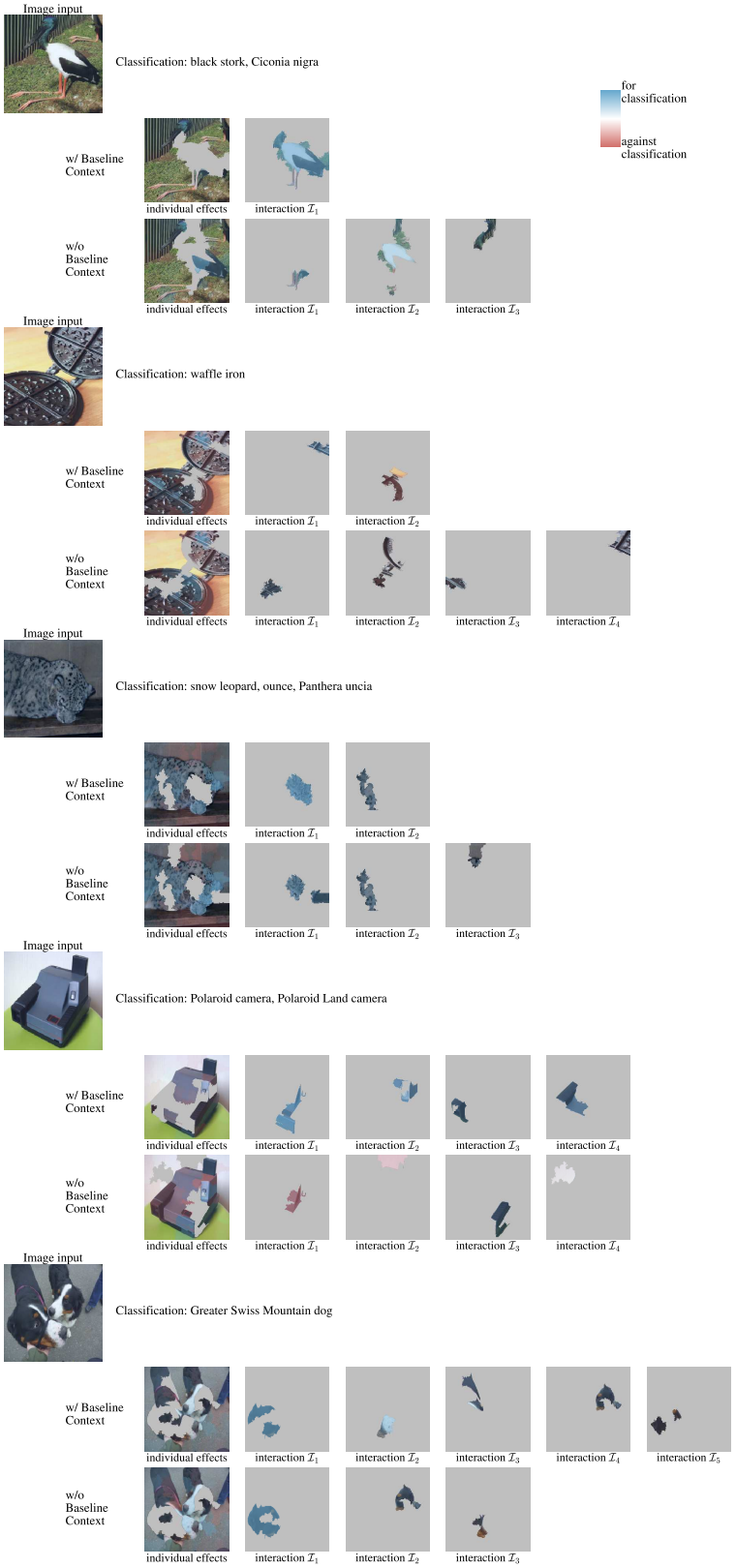


Figure 23: Image Viz. with `ArchDetect` Ablation B. The interactions tend to focus on salient patches of the images when including the baseline context.