Towards Axiomatic, Hierarchical, and Symbolic Explanation for Deep Models

Anonymous authors
Paper under double-blind review

Abstract

This paper proposes a hierarchical and symbolic And-Or graph (AOG) to objectively explain the internal logic encoded by a well-trained deep model for inference. We first define the objectiveness of an explainer model in game theory, and we develop a rigorous representation of the And-Or logic encoded by the deep model. The objectiveness and trustworthiness of the AOG explainer are both theoretically guaranteed and experimentally verified. Furthermore, we propose several techniques to boost the conciseness of the explanation.

1 Introduction

Deep models have shown promise in various tasks in recent years, but the black-box nature of the deep model makes it difficult for people to understand its internal behavior. Many post-hoc explanation methods have been proposed either to explain the deep model visually (Yosinski et al., 2015; Simonyan et al., 2013; Zeiler & Fergus, 2014), or to estimate the attribution/importance of input variables (Ribeiro et al., 2016; Lundberg & Lee, 2017; Selvaraju et al., 2017).

Beyond the above post-hoc explanations at the pixel-wise and unit-wise levels, in this study, we aim to explain the feature representation of the deep model using a hierarchical and symbolic graph, which can be better aligned with human cognition. To this end, previous studies have proposed some hierarchical and symbolic explainer models for deep models, for example, using the decision tree (Frosst & Hinton, 2017; Wu et al., 2018), the explanatory graph (Zhang et al., 2018), and the additive model (Vaughan et al., 2018; Tan et al., 2018) to explain deep models.

Unfortunately, most of the above hierarchical and symbolic explainer models cannot guarantee that the explanation/explainer model objectively reflects the internal logic of the deep model, and lack solid theoretical foundations for the trustworthiness of explanations. For example, some explainer models are learned to mimic the output of the pre-trained deep model via knowledge distillation (Hinton et al., 2015). However, the distilled explainer model may just fit the output of the pre-trained deep model, instead of accurately illustrating all internal logic of the deep model, which is investigated in Section 3.2. More crucially, there is no theoretical design to ensure that the explainer model reflects the exact inference logic used by the deep model.

Objectiveness. In this paper, we define the objectiveness of an explainer model in game theory, and based on this definition, we develop a hierarchical and symbolic explainer to objectively explain the deep model. The objectiveness of an explainer model is defined as follows. *If an explainer model is considered objective, then it is supposed to generate exactly the same output as the explained deep model given any arbitrary input.* More crucially, we can derive the following deduction from this definition. *An objective explainer model should not only generate the same output as the deep model, but also use the same underlying logic/reasons for inference.*

Fortunately, our findings in game theory enable us to evaluate the objectiveness of an explainer model, *i.e.* we find that we can accurately decompose the inference logic of a deep model into utilities of interactions between input variables in game theory. Given an input sample with n variables (*e.g.* a sentence with n words) $N = \{1, 2, \ldots, n\}$, input variables usually interact with each other for inference, instead of working individually. For each subset of variables $S \subseteq N$, we measure the numerical utility of the interaction between variables in S using the Harsanyi dividend (Harsanyi, 1963) $I(S)$. The subset S with a considerable utility is termed an *interaction pattern*. It is proved that the model output can be decomposed as the sum of utilities of all potential interactions patterns,
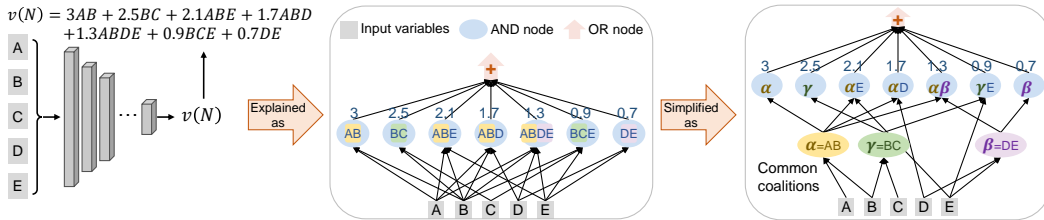
Figure 1: We transform the logic of a model into an three-layer And-Or graph (AOG), and extract common coalitions of variables to build a deeper AOG, so as to further simplify the explanation.

i.e. model output $= \sum_{S \subseteq N} I(S)$. For example, in Figure 1, let us consider the pattern $S = \{A, B, E\}$ in the input $N = \{A, B, C, D, E\}$. Only if the A, B, E appear together, the deep model will generate an additional interaction utility $I(S = \{A, B, E\}) = 2.1$ to the output. The absence of any variable in $S = \{A, B, E\}$ will remove the interaction utility. Then, we obtain the following conclusions.

• **We can use the interaction utility to evaluate the objectiveness of the explainer model.** For each interaction pattern $S \subseteq N$, if its interaction utility $I_g(S)$ encoded by the explainer model does not equal to its utility $I_v(S)$ encoded by the deep model, then the explainer model does not satisfy the objectiveness mentioned above.

• **We can directly consider interaction patterns decomposed from the model output as a set of underlying reasons to explain the inference of the deep model.**

Conciseness. Beyond the objectiveness, the explanation is supposed to be concise enough for people to understand. The above explanation consists of 2^n interaction patterns, which is quite complex. Therefore, we develop the following three techniques to further simplify the explanation.

(1) *Boosting sparsity.* We find that the 2^n interaction patterns are actually sparse. Only a few patterns have significant effects on the model output with large values of $|I(S)|$, namely *salient patterns*. Other patterns have little effects with small values of $|I(S)|$, namely *noisy patterns*. Therefore, we can just choose salient patterns with large $|I(S)|$ to approximately explain the model output, and ignore noisy ones. Furthermore, we can boost the sparsity of salient patterns by learning baseline values, so as to further simplify the explanation. Here, baseline values are used to represent the absence states of input variables, which are used in the computation of $I(S)$.

(2) *Learning hierarchy.* We transform the above interaction-based explanation into an And-Or graph (AOG) with a deep hierarchy. The AOG combines variables, which frequently co-appear in different interaction patterns, to construct a single node in the AOG, thereby simplifying the explanation.

(3) *Adversarial training.* Moreover, we discover that adversarial training (Madry et al., 2018) makes the deep model encode more sparse interaction patterns than standard training.

Discussion about the trustworthiness of the AOG explainer. First, the interaction utility based on the Harsanyi dividend is proved to be compositional elements in other classical metrics for attributions and interactions (Shapley, 1953; Grabisch & Roubens, 1999; Sundararajan et al., 2020). Second, our interaction utility is proved to satisfy seven axioms. Third, strictly speaking, the fully objective explanation consists of 2^n interaction patterns. Thus, there is a trade-off between the explanation conciseness and the explanation accuracy. To achieve scientific rigour, we can quantify the limitation of the AOG explainer, *i.e.* measuring the ratio of components of the model output that cannot be explained by the AOG.

Contributions of this paper can be summarized as follows: (1) We define two types of objectiveness of an explainer model in game theory. (2) We develop an axiomatic, hierarchical, and symbolic AOG to objectively explain the internal logic of a deep model. Furthermore, we propose three techniques to simplify the explanation. (3) We prove the objectiveness and trustworthiness of the AOG explainer in both theory and practice. In particular, the interaction utility used in this paper is proved to have strong connections with other classical game-theoretic attribution/interaction metrics.

2 RELATED WORKS

Explanations for deep models. Many explanation methods have been proposed to explain the knowledge learned by deep models. The typical explanation methods include the visualization of

features learned by the DNN (Simonyan et al., 2013; Zeiler & Fergus, 2014; Yosinski et al., 2015; Dosovitskiy & Brox, 2016), and the estimation of the pixel-wise attribution/saliency/importance of input samples (Ribeiro et al., 2016; Lundberg & Lee, 2017; Fong & Vedaldi, 2017; Zhou et al., 2015; 2016; Selvaraju et al., 2017).

Beyond the above pixel-wise visualization and attribution, some studies explain the logic encoded by a deep model by transforming its feature representations into an interpretable symbolic model. Vaughan et al. (2018); Tan et al. (2018) learned an explainable additive model from a pre-trained DNN via knowledge distillation. Frosst & Hinton (2017); Che et al. (2016); Wu et al. (2018) distilled feature representations of a DNN into tree structures. Zhang et al. (2018) built an explanatory graph to explain the And-Or grammar (Song et al., 2013) between visual concepts learned by a DNN. In fact, the And-Or grammar was also used to learn an interpretable AOGNet (Li et al., 2019).

However, the above previous explainer models were mainly learned to fit the model output, but there is no principle to make the explainer objectively illustrate the actual logic of the deep model. This may raise the risk of incorrect explanations in unusual samples. In this study, we first define the objectiveness of an explainer model, and further theoretically prove the objectiveness of the AOG explainer.

Interactions. Many studies focus on interactions between input variables learned by deep models in recent years (Sorokina et al., 2008; Murdoch et al., 2018; Singh et al., 2018; Jin et al., 2019; Janizek et al., 2020). In game theory, Grabisch & Roubens (1999) first proposed the Shapley interaction index, and this index was used by Lundberg et al. (2018) to analyze tree ensembles. Sundararajan et al. (2020) defined the Shapley-Taylor interaction index based on the Shapley value (Shapley, 1953). In this paper, we use the Harsanyi dividend (Harsanyi, 1963) to measure the utility of interactions between input variables. This interaction metric has a strong connection to (Grabisch & Roubens, 1999), but satisfies the efficiency axiom. The efficiency axiom enables our interaction metric to be used to decompose the model output.

3 METHOD

3.1 DECOMPOSING THE OUTPUT OF A DEEP MODEL

As the theoretical foundation of the objectiveness metric for explanations, let us first introduce how to decompose the output score of a deep model into the sum of utilities of interaction patterns. Given an input sample x with n input variables (*e.g.*, a sentence with n words) $N = \{1, 2, \ldots, n\}$ and a pre-trained deep model $v(\cdot)$, $v(N)$ denotes the output score of the model *w.r.t.* the entire input N. In a deep model, input variables do not contribute to the output score individually. Instead, they interact with each other. Let $I(S)$ denote the numerical utility of the interaction between variables in the subset $S \subseteq N$. Let us take a model learned for sentiment classification and the input sentence "*Hi, sit down and take it*" for example. If we add the word "*easy*" and obtain the sentence "*Hi, sit down and take it easy,*" then the co-appearing of "*take it easy*" will significantly increase the output score of the "positive" sentiment, and the score increase is referred to as the interaction utility $I(S = \{take, it, easy\})$. Therefore, we consider the interaction between words in the subset S has a positive interaction utility $I(S) > 0$ on the output score, and we call S a positive interaction pattern. The absence of any word in S will destroy the interaction between words in S, thereby removing its utility $I(S)$. Similarly, if $I(S) < 0$, S is referred to as a negative interaction pattern. If $I(S)$ approximates 0, it means that variables in S have no interactions.

Fortunately, we discover that we can decompose the output score $v(N)$ of a deep model into the sum of interaction utilities $I(S)$ (as shown in Figure 2 (left)), when we use the Harsanyi dividend (Harsanyi, 1963) to quantify the interaction utility.

$$v(N) = \sum\nolimits_{S \subseteq N} I(S), \quad s.t. \quad I(S) \stackrel{\text{def}}{=} \sum\nolimits_{L \subseteq S} (-1)^{|S|-|L|} \cdot v(L) \tag{1}$$

where $v(S)$ denotes the output score of the model when only variables in S are given. In particular, $I(\emptyset) = v(\emptyset)$. In the computation of $v(S)$, variables out of S are usually replaced by their baseline values to represent their absence. Computational details will be introduced in Section 3.3.

In this way, we can understand the Harsanyi dividend to measure the marginal utility of the interaction between input variables in S, when we remove interaction utilities of all smaller sub-
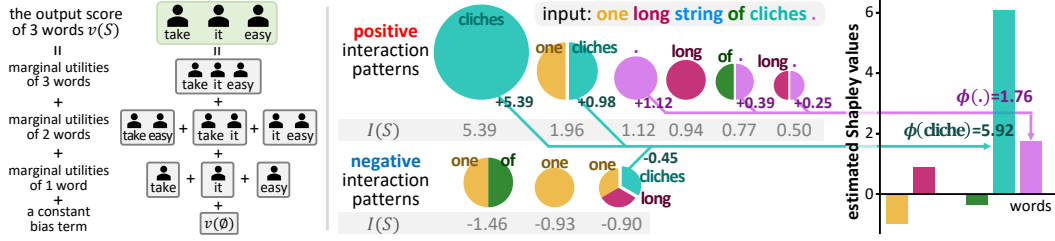
Figure 2: (left) The output score of the model $v(S)$ can be decomposed into the sum of marginal utilities of interactions between input variables in S, $\{I(L)|L \subseteq S\}$. (right) The interaction utility $I(S)$, $|S| = m$, is fairly assigned to the m variables in S in the computation of Shapley values.

sets of variables in S, $\{I(L)|L \subsetneq S\}$. In the aforementioned sentence "*Hi, sit down and take it easy,*" the interaction utility of the pattern $S = \{take, it, easy\}$ is defined as $I(\{take, it, easy\}) = v(\{take, it, easy\}) - I(\emptyset) - I(\{take\}) - I(\{it\}) - I(\{easy\}) - I(\{take it\}) - I(\{take, easy\}) - I(\{it, easy\})$.

Axioms and theorems. In this paper, we further prove that the interaction utility based on the Harsanyi dividend satisfies the following desirable axioms, which enhance the trustworthiness of this metric. Except for the efficiency axiom, detailed proofs of other axioms are shown in Appendix A.

(1) *Efficiency axiom* (proved by Harsanyi (1963)). The output score of a model can be decomposed into interaction utilities of different patterns, *i.e.* $v(N) = \sum_{S \subseteq N} I(S)$.

(2) *Linearity axiom.* If we merge output scores of two models w and v as the output of model u, *i.e.* $\forall S \subseteq N, u(S) = w(S) + v(S)$, then their interaction utilities $I_v(S)$ and $I_w(S)$ can also be merged as $\forall S \subseteq N, I_u(S) = I_v(S) + I_w(S)$.

(3) *Dummy axiom.* If a variable $i \in N$ is a dummy variable, *i.e.* $\forall S \subseteq N\backslash\{i\}, v(S \cup \{i\}) = v(S) + v(\{i\})$, then it has no interaction with other variables, $\forall S \subseteq N\backslash\{i\}, I(S \cup \{i\}) = 0$.

(4) *Symmetry axiom.* If input variables $i, j \in N$ cooperate with other variables in the same way, $\forall S \subseteq N\backslash\{i, j\}, v(S \cup \{i\}) = v(S \cup \{j\})$, then they have same interaction utilities with other variables, $\forall S \subseteq N\backslash\{i, j\}, I(S \cup \{i\}) = I(S \cup \{j\})$.

(5) *Anonymity axiom.* For any permutations π on N, we have $\forall S \subseteq N, I_v(S) = I_{\pi v}(\pi S)$, where $\pi S \triangleq \{\pi(i)|i \in S\}$, and the new model πv is defined by $(\pi v)(\pi S) = v(S)$. This indicates that interaction utilities are not changed by permutation.

(6) *Recursive axiom.* The interaction utilities can be computed recursively. For $i \in N$ and $S \subseteq N\backslash\{i\}$, the interaction utility of the pattern $S \cup \{i\}$ is equal to the interaction utility of S with the presence of i minus the interaction utility of S with the absence of i, *i.e.* $\forall S \subseteq N \backslash \{i\}, I(S \cup \{i\}) = I(S|i \text{ is always present}) - I(S)$. $I(S|i \text{ is always present})$ denotes the interaction utility when the variable i is always present as a constant context, *i.e.* $I(S|i \text{ is always present}) = \sum_{L \subseteq S}(-1)^{|S|-|L|} \cdot v(L \cup \{i\})$.

(7) *Interaction distribution axiom.* This axiom characterizes how interactions are distributed for "interaction functions" (Sundararajan et al., 2020). An interaction function v_T parameterized by a context T satisfies $\forall S \subseteq N$, if $T \subseteq S$, $v_T(S) = c$; otherwise, $v_T(S) = 0$. Then, we have $I(T) = c$, and $\forall S \neq T$, $I(S) = 0$.

Furthermore, we have also proven that the interaction utility $I(S)$ can explain the elementary mechanism of existing game-theoretic attributions/interactions, and $I(S)$ is equivalent to an existing interaction metric in some cases.

Theorem 3.1 (Connection to the Shapley value, proved by Harsanyi (1963)). *Let $\phi(i)$ denote the Shapley value (Shapley, 1953) of an input variable i. Then, its Shapley value can be represented as the weighted sum of interaction utilities,* i.e. $\phi(i) = \sum_{S \subseteq N\backslash\{i\}} \frac{1}{|S|+1} I(S \cup \{i\})$. *In other words, the utility of an interaction pattern with m variables should be equally assigned to the m variables in the computation of Shapley values.*

This theorem also proves that the Shapley value is a fair assignment of attributions from the perspective of Harsanyi dividend, as shown in Figure 2 (right). We also conducted experiments to compute the cosine similarity between the accurate Shapley value $\phi(i)$ and the estimated Shapley value $\tilde{\phi}(i) = \sum_{S \subseteq N\backslash\{i\}} \frac{1}{|S|+1} I(S \cup \{i\})$ in Figure 8, when we used different numbers of patterns S. The result in Figure 8 shows that the Harsanyi dividend can accurately approximate the Shapley value.
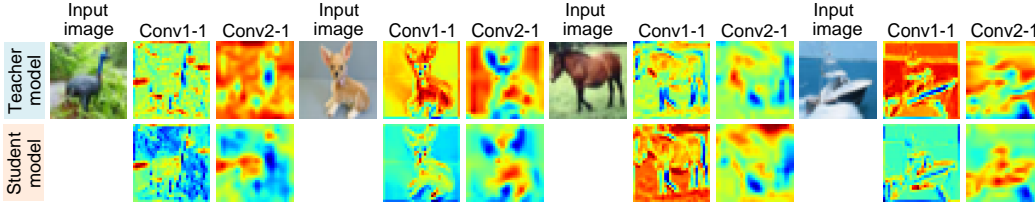
Figure 3: Knowledge distillation cannot ensure the objectiveness of the student model, because the student model and the trained teacher model use different image regions to compute features. The first row shows the Grad-CAM attention (Selvaraju et al., 2017) of the teacher model. The second row shows the Grad-CAM attention of the student model.

Theorem 3.2 (Connection to the marginal benefit, proved in Appendix A.2). *$\Delta v_T(S) = \sum_{L \subseteq T}(-1)^{|T|-|L|}v(L \cup S)$ denotes the marginal benefit (Grabisch & Roubens, 1999) of variables in $T \subseteq N \setminus S$ given the environment S. We have proven that $\Delta v_T(S)$ can be decomposed into the sum of interaction utilities inside T and sub-environments of S, i.e. $\Delta v_T(S) = \sum_{S' \subseteq S} I(T \cup S')$.*

Theorem 3.3 (Connection to the Shapley interaction index, proved in Appendix A.2). *Given a subset of input variables $T \subseteq N$, $I^{Shapley}(T) = \sum_{S \subseteq N \setminus T} \frac{|S|!(|N|-|S|-|T|)!}{(|N|-|T|+1)!}\Delta v_T(S)$ denotes the Shapley interaction index (Grabisch & Roubens, 1999) of T. We have proven that the Shapley interaction index can be represented as the weighted sum of utilities of interaction patterns, $I^{Shapley}(T) = \sum_{S \subseteq N \setminus T} \frac{1}{|S|+1} I(S \cup T)$. This metric treats the coalition of variables in T as a single variable, and thus uniformly allocates the attribution $I(S \cup T)$ to all variables including T.*

Theorem 3.4 (Connection to the Shapley Taylor interaction index, proved in Appendix A.2). *Given a subset of input variables $T \subseteq N$, let $I^{Shapley\text{-}Taylor}(T)$ denote the Shapley Taylor interaction index (Sundararajan et al., 2020) of order k for T. We have proven that the Shapley taylor interaction index can be represented as the weighted sum of interaction utilities, i.e. $I^{Shapley\text{-}Taylor}(T) = I(T)$ if $|T| < k$; $I^{Shapley\text{-}Taylor}(T) = \sum_{S \subseteq N \setminus T} \binom{|S|+k}{k}^{-1} I(S \cup T)$ if $|T| = k$; and $I^{Shapley\text{-}Taylor}(T) = 0$ if $|T| > k$.*

3.2 OBJECTIVENESS OF EXPLANATIONS

Lack of objectiveness of previous explainer models. Previous studies have proposed some symbolic explainer models to explain black-box models, especially explaining DNNs (Krishnan et al., 1999; Che et al., 2016; Wu et al., 2018). Most explainer models are lack of theoretical support for their objectiveness, thereby hurting the trustworthiness of such explainer models. For example, some methods used knowledge distillation (Hinton et al., 2015) to transform a pre-trained DNN into another explainer model (Che et al., 2016; Frosst & Hinton, 2017; Tan et al., 2018). However, as Figure 3 shows, knowledge distillation cannot ensure that the student model exhibits the same logic as the teacher model. We distilled the output before the softmax of a pre-trained VGG-11 (Simonyan & Zisserman, 2014) into another VGG-11 on the CIFAR-10 dataset (Krizhevsky et al., 2009). The result shows that although the student DNN had similar outputs with the teacher DNN, their attention computed by Grad-CAM (Selvaraju et al., 2017) was significantly different. Therefore, knowledge distillation cannot ensure the student model represents the internal logic of the teacher model, making the student model a non-objective explainer of the teacher model. Just like the explainer models learned based on knowledge distillation, explainer models based on clustering (Zhang et al., 2018) also cannot guarantee the objectiveness.

Definition of objectiveness. Therefore, we define the following two types of objectiveness, in order to ensure that the explainer model objectively illustrates the true logic used by the deep model for inference, and provides a trustworthy explanation to people.

Definition 1 (Objectiveness over arbitrary samples). *$\forall x \in \mathcal{R}^n$, $g(N|x) = v(N|x)$.*

This definition guarantees the explainer model $g(\cdot)$ always generates the same output as the pre-trained model $v(\cdot)$ for any input x, including both normal samples and out-of-distribution (OOD) samples. $g(N|x)$ and $v(N|x)$ denote output scores of the explainer model and the deep model given all input variables in the sample x, respectively. From this perspective, the distillation-based explanation can only ensure the explainer model to have the same outputs as the deep model on training samples, but cannot mimic the deep model in all arbitrary samples, including OOD samples.

Understanding patterns as symbolic reasons. Based on Definition 1, we further define and derive the sample-wise objectiveness of explanations from the perspective of symbolic reasoning. *I.e.* the explainer model should use the same reasons for inference as the deep model. Before this, let us first introduce how to understand interaction patterns as symbolic reasons for inference. For example, given an input sentence "*sit down and take it easy*," the model output can be decomposed as $v(\{sit, down, and, take, it, easy\}) = I(\{sit\}) + I(\{sit, down\}) + I(\{sit, take\}) + I(\{and\}) + I(\{take, it\}) + I(\{take, it, easy\}) + \ldots$. The model considers the sentiment of this sentence positive because of the large positive utility of the pattern $I\{take, it, easy\}$, which represents the AND relationship between words. The absence of any word in the pattern will deactivate this interaction. Therefore, the pattern $\{take, it, easy\}$ is considered as a symbolic explanation for the underlying reason of the deep model's inference.

Definition 2 (Sample-wise objectiveness of symbolic reasoning). Given a certain input sample $x \in \mathcal{R}^n$, $\forall S \subseteq N$, $g(S|x) = v(S|x)$, where $g(S|x)$ and $v(S|x)$ denote output scores of the explainer model and the deep model when only partial variables of the sample x in the subset S are input to the model. Variables out of S are masked by their baseline values to represent their absence, which will be introduced in Section 3.3.

Corollary 1 (proved in Appendix B). If an explainer model g satisfies Definition 2, then given an input sample $x \in \mathcal{R}^n$, $\forall S \subseteq N$, $I_g(S|x) = I_v(S|x)$. $I_g(S|x)$ and $I_v(S|x)$ denote interaction utilities of S in the explainer model and the explained model, respectively.

Corollary 1 is derived from Definition 2, and Definition 2 is actually a necessary condition of Definition 1 (proofs in Appendix B). **Definition 2 mainly focuses on the objectiveness of the explanation on a single input sample. Corollary 1 further shows that the explainer model is supposed not only to generate the same output as the deep model, but also to use the same underlying reasons.** According to Equation (1) $v(N|x) = \sum_{S \subseteq N} I(S|x)$, we can consider these interaction utilities $I(S|x)$ as underlying reasons for the inference of the deep model. Thus, Corollary 1 ensures that the reasons used by the explainer model are the same as the reasons used by the deep model.

3.3 CONSTRUCTING THE HIERARCHICAL AND SYMBOLIC AND-OR GRAPH

And-Sum relationship between input variables. The decomposition of the model output in Equation (1), *i.e.* $v(N) = \sum_{S \subseteq N} I(S)$, explains the deep model as an And-Sum representation. Each interaction utility $I(S)$ represents an AND relationship between variables in S. For example, in Figure 1, for the pattern $S = \{A, B, E\}$, when all variables A, B, E appear together in the input, the utility $I(S)$ will be added to the model output. Otherwise, the absence of any variable $i \in S$ will remove the interaction utility $I(S)$ from the model output. The SUM relationship refers to that all utilities $I(S)$ of all interaction patterns add up to the model output $v(N)$.

Learning baseline values to obtain concise explanations. The above And-Sum representation is the sum of $2^n - 1$ interaction patterns and a constant bias $I(\emptyset)$, which is too complex for people to understand. Among all $2^n - 1$ patterns, most of them have little influence on the model output with small absolute values $|I(S)|$, namely *noisy patterns*. A few patterns have large impacts on the model output with large absolute values $|I(S)|$, namely *salient patterns*. The sparsity of salient patterns enables us to only use salient patterns to approximate the model output as $v(N) \approx \sum_{S \in \Omega^{\text{salient}}} I(S)$, which ensures the conciseness of the explanation. Ω^{salient} denotes the set of salient patterns.

In fact, the sparsity of salient patterns does not only depend on the deep model itself, but it is also determined by the choice of baseline values. In this way, we can further boost the sparsity of salient patterns by learning baseline values, in order to enhance the conciseness of the explanation. Specifically, according to Equation (1), $I(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} \cdot v(L)$, where $v(L)$ is the model output when only variables in L are given. In this case, input variables out of L are masked by their baseline values (or called reference values) to represent the absence of these variables, as follows.

$$v(L) = model(x^{\text{mask}}), \quad x_i^{\text{mask}} = \begin{cases} x_i, & i \in L \\ r_i, & i \in N \setminus L \end{cases} \tag{2}$$

where r_i denotes the baseline value of the i-th input variable. $\boldsymbol{r} = [r_1, r_2, \ldots, r_n]$. Therefore, we minimize the following loss to learn baseline values that maximize the sparsity of salient patterns.

$$Loss(\boldsymbol{r}) = \sum_{S \in \Omega^{\text{all}} \setminus \Omega^{\text{salient}}} |I(S)| \tag{3}$$

where $\Omega^{\text{all}} = 2^N = \{S|S \subseteq N\}$ denotes the set of all interaction patterns.

Ratio of the explained utilities R_k. Considering the sparsity of salient patterns, we only use salient patterns with the top-k largest absolute values $|I(S)|$, denoted by $\Omega^{\text{salient}}_{\text{top-}k}$, to approximately explain the output of the deep model. In this way, the ratio of the explained interaction utilities to the overall model output can be quantified R_k as follows.

$$R_k = \frac{\sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}} |I(S)|}{\sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}} |I(S)| + |\Delta|} \tag{4}$$

Instead of computationally expensive enumeration of all interaction patterns (*e.g.* the metric in Appendix H), we compute $\Delta = v(N) - v(\emptyset) - \sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}} I(S)$ to denote utilities of the unexplained patterns to save the computational cost. Appendix H shows another metric for the explained utilities.

Transforming the And-Sum representation into a hierarchical And-Or Graph (AOG). In fact, the above And-Sum representation can be further transformed into a deep hierarchical AOG to make the explanation more concise. Actually, we can consider the And-Sum representation as a three-layer AOG. Then, we summarize common coalitions of input variables shared by different AND nodes to construct a deeper AOG. Figure 1 (left) shows an example of the three-layer And-Sum representation (AOG). At the bottom layer, there are n leaf nodes representing n variables of the input sample. The second layer of the AOG has multiple AND nodes, each representing the AND relationship between its child nodes. The node ABE indicates the AND relationship between its child nodes A, B, and E, *i.e.* the interaction pattern $S = \{A, B, E\}$. Its interaction utility $I(\{A, B, E\}) = 2.1$ is labeled on the top of the node. The root node is a *noisy OR* node (as discussed in (Li et al., 2019)), which sums up utilities of all its child AND nodes to mimic the model output, *i.e.* $output = \sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}} I(S)$.

Furthermore, we extract common coalitions shared by different interaction patterns as single nodes to simplify the AOG. In Figure 1, input variables A, B frequently co-appear in different salient interaction patterns. Thus, we consider A, B as a coalition and add a single AND node $\alpha = \{A, B\}$ to represent their co-appearing. Therefore, the pattern $\{A, B, E\}$ is simplified as $\{\alpha, E\}$.

The extraction of common coalitions of input variables is guided by the minimum description length (MDL) principle (Hansen & Yu, 2001). Given an AOG g and input variables N, let $M = N \cup \Omega^{\text{coalition}}$ denote the set of all terminal nodes and AND nodes in the bottom two layers (*e.g.* $M = N \cup \Omega^{\text{coalition}} = \{A, B, C, D, E\} \cup \{\alpha, \beta, \gamma\}$ in Figure 1 (right)). The MDL principle can be formulated as follows.

$$L(g, M) = \underbrace{L(M)}_{\text{length of describing the set of nodes } M} + \underbrace{L_M(g)}_{\text{length of using nodes in } M \text{ to describe interaction patterns in } g} \tag{5}$$

The first term $L(M) = -\lambda \sum_{m \in M} p(m) \log p(m)$ represents the description length of the set of nodes M, where $\forall m \in M$, $p(m) = count(m)/\sum_{m' \in M} count(m')$ s.t. $count(m) = \sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}: S \ni m} |I(S)|$. $p(m)$ measures the probability of the node m appearing in salient interaction patterns. $\Omega^{\text{salient}}_{\text{top-}k}$ denotes the set of top-k salient patterns which are represented using all nodes in M. $\Omega^{\text{salient}}_{\text{top-}k}$ needs to be updated when we insert a new AND node into M. $\lambda = 10/Z$ is a scalar weight, where $Z = \sum_{S \in \Omega^{\text{salient}}_{\text{top-}k}} |I(S)|$. The second term $L_M(g) = -\mathbb{E}_{S \sim p(S|g)} \sum_{m \in S} \log p(m)$ represents the length of describing salient interaction patterns using nodes in M. The appearing probability of the interaction pattern S in the AOG g is sampled as $p(S|g) \propto |I(S)|$. The loss $L(g, M)$ can be minimized via the greedy strategy to extract common coalitions of input variables by following (Hansen & Yu, 2001). Please see Appendix C for more discussions.

4 EXPERIMENTS

Datasets and models. We focused on both tasks of natural language processing and the classification/regression task based on tabular datasets. For the natural language processing, we explained LSTMs (Hochreiter & Schmidhuber, 1997) and CNNs used in (Rakhlin, 2016). Each model was trained for the sentiment prediction on the SST-2 dataset (Socher et al., 2013) and trained for the linguistic acceptability classification on the CoLA dataset (Warstadt et al., 2019), respectively. The tabular datasets included the UCI census income dataset (Dua & Graff, 2017), the UCI bike sharing dataset (Fanaee-T & Gama, 2013), and the UCI TV news channel commercial detection dataset (Dua
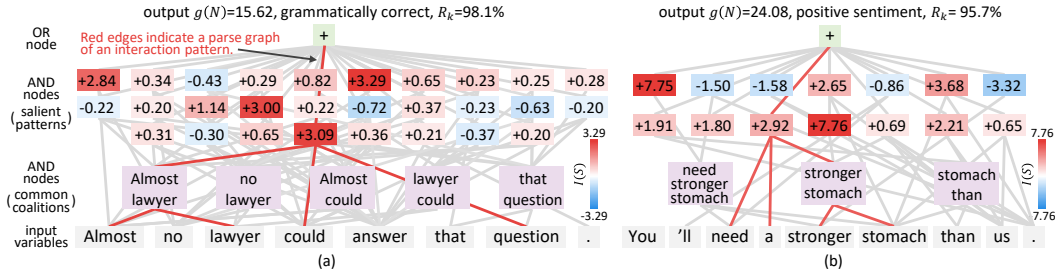
Figure 4: Examples of the AOG extracted from the CNN trained on (a) the CoLA dataset and (b) the SST-2 dataset, respectively. The red color of nodes in the second layer indicates interaction patterns with positive utilities, while the blue color represents patterns with negative utilities. Red edges indicate the parse graph of an interaction pattern.

Table 1: IoU on the synthesized datasets. Our method correctly extracted salient interaction patterns.

Dataset	Model	Average IoU
Addition-Multiplication dataset (Zhang et al., 2021)	functions in the dataset	1.0000
Dataset in (Ren et al., 2021)		1.0000
Manually labeled And-Or dataset	MLP-5	0.9761
	ResMLP-5	0.9787
Re-labeled TV news dataset (Dua & Graff, 2017)	MLP-5	0.8180
	ResMLP-5	0.8189

Table 2: Jaccard similarity between two models. Two adversarially trained models were more similar than two normally trained models.

		TV news	census	bike
MLP-2	normal	0.5965	0.4899	-
	adversarial	**0.6109**	**0.6292**	-
MLP-5	normal	0.3664	0.2482	0.3816
	adversarial	**0.6304**	**0.4971**	**0.4741**
ResMLP-5	normal	0.3480	0.2764	0.3992
	adversarial	**0.5731**	**0.4489**	**0.4491**

& Graff, 2017). These datasets were termed *census, bike,* and *TV news* for simplicity. Each tabular dataset was trained on MLPs, LightGBM (Ke et al., 2017), and XGBoost (Chen & Guestrin, 2016), and we explained these models. For MLPs, we used two-layer MLPs (namely *MLP-2*) and five-layer MLPs (namely *MLP-5*), where each layer contained 100 neurons. Besides, we added a skip-connection to each layer of the MLP-5 to build the residual-MLP (namely *ResMLP-5*). Please see Appendix D.1 for more discussions. Figure 4 shows examples of the AOG generated by our method on CNNs trained on the SST-2 dataset and the CoLA dataset. Please see Appendix G.1 for the visualization of more AOGs.

Intersection over union (IoU) between the extracted salient patterns and ground-truth patterns. In this experiment, we evaluated whether the extracted salient interaction patterns correctly reflected the interaction in the model. However, for most datasets, people could not annotate the ground-truth patterns learned by deep models, as discussed in (Zhang et al., 2021). To this end, we used the following functions and datasets with ground-truth interaction patterns for evaluation. The first two datasets were the Addition-Multiplication dataset (Zhang et al., 2021) and the dataset proposed in (Ren et al., 2021). The third dataset was extended from the TV news dataset (Dua & Graff, 2017). In order to construct ground-truth interaction patterns, we re-labeled samples in this dataset following a set of pre-defined And-Or functions, namely the *re-labeled TV news dataset*. Besides, we also labeled samples from the Gaussian distribution based on the same set of And-Or functions, namely the *manually labeled And-Or dataset*. Please see Appendix D.2 for more details. Then, we learned MLP-5 and ResMLP-5 based on the two And-Or datasets.

Given a model and an input sample, let k denote the number of ground-truth patterns Ω^{truth}, which has been labeled in the dataset. Then, for a fair comparison, we also extracted the top-k salient patterns $\Omega^{\text{salient}}_{\text{top-}k}$ from the model. We measured the IoU between Ω^{truth} and $\Omega^{\text{salient}}_{\text{top-}k}$ as $IoU = |\Omega^{\text{salient}}_{\text{top-}k} \cap \Omega^{\text{truth}}|/|\Omega^{\text{salient}}_{\text{top-}k} \cup \Omega^{\text{truth}}|$ to evaluate the accuracy of the extracted patterns. Results in Table 1 show that our interaction metric based on the Harsanyi dividend successfully extracted most salient interaction patterns. Note that the IoU based on the re-labeled TV news dataset was about 81.8%, instead of approaching 100%. It was because there was no principle to ensure the model to learn ground-truth interaction patterns, rather than the problem of our method.

Objectiveness of the AOG explainer. We also proposed a metric to evaluate the objectiveness of explainer models according to the definition in Section 3.2. We compared the AOG explainer with seven baseline explanation methods, including distillation-based explainer models and attribution-based explanations. The AOG explainer exhibited much stronger objectiveness than baseline explanation methods. Please see Appendix F for detailed metrics and experiments.
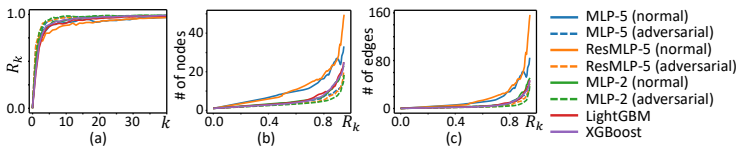
Figure 5: (a) The change of R_k (the ratio of the explained utilities) along with the number of salient patterns k in the AOG. (b, c) The change of the node number and the edge number in the AOG along with R_k. AOGs corresponding to adversarially trained models were less complex than AOGs corresponding to normally trained models.
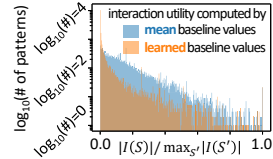
Figure 6: The histogram of the re-scaled interaction strength. The learned baseline values boosted the sparsity of salient interaction patterns.

Ratio of the explained utilities R_k**.** As discussed in Section 3.3, we just used salient interaction patterns with top-k absolute values $|I(S)|$ to approximate the model output. Figure 5 (a) shows the relationship between k and the ratio of the explained utilities R_k in different models based on the TV news dataset. When we used a few interaction patterns, we could explain most utilities of interaction patterns to the model output. Figure 5 (b)(c) shows that the node number and the edge number of the AOG increased along with the increase of R_k. Please see Appendix G.2 for results on other datasets.

Effects of baseline values on the conciseness of explanations. In this experiment, we explored whether the learning of baseline values in Section 3.3 could boost the sparsity of salient patterns. To this end, we followed (Dabkowski & Gal, 2017) to initialize baseline values of input variables as their mean values over different samples. Then, we learned baseline values via Equation (3). Figure 6 shows the histogram of the interaction strength $\frac{|I(S)|}{\max_{S' \subseteq N} |I(S')|}$, which was re-scaled to the range of $[0, 1]$. Compared with mean baseline values, the learned baseline values usually generated fewer salient interaction patterns with more significant strength. Thus, the learned baseline values boosted the sparsity of salient interaction patterns and enhanced the conciseness of explanations. In this experiment, we used MLP-5 and computed the re-scaled strengths of interaction patterns with training samples in the TV news dataset. Please see Appendix I for more experimental results.

Effects of adversarial training. In this experiment, we learned MLP-2, MLP-5, and ResMLP-5 on the TV news dataset via adversarial training (Madry et al., 2018). Figure 5(a) shows that compared with the normally trained model, we could use less salient patterns (smaller k) to explain the same ratio of interaction utilities R_k in the adversarially trained model. Moreover, Figure 5 (b,c) also shows that AOGs corresponding to adversarially trained models were less complex than AOGs corresponding to normally trained models. This indicated that *adversarial training made models encode more sparse interaction patterns than normal training*.

Besides, *adversarial training also made different models encode common patterns*. To this end, we trained different pairs of MLP-2, MLP-5, and ResMLP-5 models with the same architecture but using different initial parameters. Given the same input, we measured the Jaccard similarity coefficient between interaction utilities of each pair of models, in order to examine whether the two models encoded similar interaction patterns. Let $I(S)$ and $I'(S)$ denote the interaction utilities of these two models. The Jaccard similarity coefficient was computed as $J = \frac{\sum_{S \subseteq N} \min(|I(S)|, |I'(S)|)}{\sum_{S \subseteq N} \max(|I(S)|, |I'(S)|)}$. Table 2 shows that the Jaccard similarity between two adversarially trained models was significantly higher than the Jaccard similarity between two normally trained models. This indicated that adversarial training made different models encode common interaction patterns for inference.

5 CONCLUSION

In this paper, we define the objectiveness of an explainer model in game theory, and develop a hierarchical and symbolic And-Or graph (AOG) to explain the deep model. We theoretically prove and experimentally verify the objectiveness and trustworthiness of the AOG explainer. Furthermore, we propose several techniques to boost the conciseness of the explanation. Besides, we discover that adversarial training usually makes different models learn simple and similar interactions.

Reproducibility Statement. Most work in this research is to theoretically derive the AOG representation for the deep model. Therefore, there is no problem with the reproducibility. Appendix A and Appendix B provide proofs for all theoretical results in the paper. Besides, the implementation of MDL is just an engineering module. We have discussed all experimental details about datasets, models, and the implementation of MDL in Section 4 and Appendix D, which ensure the reproducibility. Furthermore, we will release the code of building the AOG when the paper is accepted.

REFERENCES

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Interpretable deep models for icu outcome prediction. In *AMIA annual symposium proceedings*, volume 2016, pp. 371. American Medical Informatics Association, 2016.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Ian Covert and Su-In Lee. Improving kernelshap: Practical shapley value estimation using linear regression. In *International Conference on Artificial Intelligence and Statistics*, pp. 3457–3465. PMLR, 2021.

Ian Covert, Scott M Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33, 2020.

Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *arXiv preprint arXiv:1705.07857*, 2017.

Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4829–4837, 2016.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Hadi Fanaee-T and Joao Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, pp. 1–15, 2013. ISSN 2192-6352. doi: 10.1007/s13748-013-0040-3. URL [WebLink].

Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, pp. 3429–3437, 2017.

Nicholas Frosst and Geoffrey Hinton. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.

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.

Mark H Hansen and Bin Yu. Model selection and the principle of minimum description length. *Journal of the American Statistical Association*, 96(454):746–774, 2001.

John C Harsanyi. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, 4(2):194–220, 1963.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

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

Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *International Conference on Learning Representations*, 2019.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30:3146–3154, 2017.

R. Krishnan, G. Sivakumar, and P. Bhattacharya. Extracting decision trees from trained neural networks. *Pattern Recognition*, 32(12):1999–2009, 1999. ISSN 0031-3203. doi: https://doi.org/10.1016/S0031-3203(98)00181-2. URL https://www.sciencedirect.com/science/article/pii/S0031320398001812.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Xilai Li, Xi Song, and Tianfu Wu. Aognets: Compositional grammatical architectures for deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6220–6230, 2019.

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*, pp. 4768–4777, 2017.

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

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

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

A Rakhlin. Convolutional neural networks for sentence classification. *GitHub*, 2016.

Jie Ren, Zhanpeng Zhou, Qirui Chen, and Quanshi Zhang. Learning baseline values for shapley values. *arXiv preprint arXiv:2105.10719*, 2021.

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*, pp. 1135–1144, 2016.

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*, pp. 618–626, 2017.

Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

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

Chandan Singh, W James Murdoch, and Bin Yu. Hierarchical interpretations for neural network predictions. In *International Conference on Learning Representations*, 2018.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y 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*, pp. 1631–1642, 2013.

Xi Song, Tianfu Wu, Yunde Jia, and Song-Chun Zhu. Discriminatively trained and-or tree models for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3278–3285, 2013.

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*, pp. 1000–1007, 2008.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *International Conference on Machine Learning*, pp. 9259–9268. PMLR, 2020.

Sarah Tan, Rich Caruana, Giles Hooker, Paul Koch, and Albert Gordo. Learning global additive explanations for neural nets using model distillation. *arXiv preprint arXiv:1801.08640*, 2018.

Joel Vaughan, Agus Sudjianto, Erind Brahimi, Jie Chen, and Vijayan N Nair. Explainable neural networks based on additive index models. *arXiv preprint arXiv:1806.01933*, 2018.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

Mike Wu, Michael C Hughes, Sonali Parbhoo, Maurizio Zazzi, Volker Roth, and Finale Doshi-Velez. Beyond sparsity: Tree regularization of deep models for interpretability. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Hao Zhang, Yichen Xie, Longjie Zheng, Die Zhang, and Quanshi Zhang. Interpreting multivariate shapley interactions in dnns. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10877–10886, 2021.

Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *In ICLR*, 2015.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

A   PROOFS OF AXIOMS AND THEOREMS OF THE INTERACTION UTILITY

A.1   PROOFS OF AXIOMS

In this section, we prove that the interaction utility $I(S)$ satisfies the *efficiency, linearity, dummy, symmetry, anonymity, recursive,* and *interaction distribution* axioms.

(1) Efficiency axiom. The output score of a model can be decomposed into interaction utilities inside different patterns, *i.e.* $v(N) = \sum_{S \subseteq N} I(S)$.

● *Proof:* According to the definition of the interaction utility, we have

$$\sum_{S \subseteq N} I(S) = \sum_{S \subseteq N} \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L)$$

$$= \sum_{L \subseteq N} \sum_{S \subseteq N: S \supseteq L} (-1)^{|S|-|L|} v(L)$$

$$= \sum_{L \subseteq N} \sum_{s=|L|}^{n} \sum_{\substack{S \subseteq N: S \supseteq L \\ |S|=s}} (-1)^{s-|L|} v(L)$$

$$= \sum_{L \subseteq N} v(L) \sum_{m=0}^{n-|L|} \binom{n-|L|}{m} (-1)^{m} = v(N)$$

(2) Linearity axiom. If we merge output scores of two models w and v as the output of model u, $\forall S \subseteq N, i.e.$ $u(S) = w(S) + v(S)$, then their interaction utilities $I_v(S)$ and $I_w(S)$ can also be merged as $\forall S \subseteq N, I_u(S) = I_v(S) + I_w(S)$.

● *Proof:* According to the definition of the interaction utility, we have

$$I_u(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} u(S)$$

$$= \sum_{L \subseteq S} (-1)^{|S|-|L|} [v(S) + w(S)]$$

$$= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(S) + \sum_{L \subseteq S} (-1)^{|S|-|L|} w(S)$$

$$= I_v(S) + I_w(S).$$

(3) Dummy axiom. If a variable $i \in N$ is a dummy variable, *i.e.* $\forall S \subseteq N \backslash \{i\}, v(S \cup \{i\}) = v(S) + v(\{i\})$, then it has no interactions with other variables, $\forall S \subseteq N \backslash \{i\}, I(S \cup \{i\}) = 0$.

● *Proof:* According to the definition of the interaction utility, we have

$$I(S \cup \{i\}) = \sum_{L \subseteq S \cup \{i\}} (-1)^{|S|+1-|L|} v(L)$$

$$= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\})$$

$$= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} [v(S) + v(\{i\})]$$

$$= \left[\sum_{L \subseteq S} (-1)^{|S|-|L|} \right] \cdot v(\{i\})$$

$$= 0.$$

(4) Symmetry axiom. If input variables $i, j \in N$ cooperate with other variables in the same way, $\forall S \subseteq N \backslash \{i, j\}, v(S \cup \{i\}) = v(S \cup \{j\})$, then they have same interaction utilities with other variables, $\forall S \subseteq N \backslash \{i, j\}, I(S \cup \{i\}) = I(S \cup \{j\})$.

• *Proof:* According to the definition of the interaction utility, we have

$$
\begin{aligned}
I(S \cup \{i\}) &= \sum_{L \subseteq S \cup \{i\}} (-1)^{|S|+1-|L|} v(L) \\
&= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\}) \\
&= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{j\}) \\
&= \sum_{L \subseteq S \cup \{j\}} (-1)^{|S|+1-|L|} v(L) \\
&= I(S \cup \{j\}).
\end{aligned}
$$

(5) Anonymity axiom. For any permutations π on N, we have $\forall S \subseteq N, I_v(S) = I_{\pi v}(\pi S)$, where $\pi S \triangleq \{\pi(i)|i \in S\}$, and the new model πv is defined by $(\pi v)(\pi S) = v(S)$. This indicates that interaction utilities are not changed by permutation.

• *Proof:* According to the definition of the interaction utility, we have

$$
\begin{aligned}
I_{\pi v}(\pi S) &= \sum_{L \subseteq S} (-1)^{|S|-|L|} (\pi v)(\pi L) \\
&= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= I_v(S).
\end{aligned}
$$

(6) Recursive axiom. The interaction utilities can be computed recursively. For $i \in N$ and $S \subseteq N \setminus \{i\}$, the interaction utility of the pattern $S \cup \{i\}$ is equal to the interaction utility of S with the presence of i minus the the interaction utility of S with the absence of i, *i.e.* $I(S \cup \{i\}) = I(S|i \text{ is always present}) - I(S)$ (*s.t.* $S \subseteq N \setminus \{i\}$). $I(S|i \text{ is always present})$ denotes the interaction utility when the variable i is always present as a constant context, *i.e.* $I(S|i \text{ is always present}) = \sum_{L \subseteq S} (-1)^{|S|-|L|} \cdot v(L \cup \{i\})$.

• *Proof:* According to the definition of the interaction utility, we have

$$
\begin{aligned}
I(S \cup \{i\}) &= \sum_{L \subseteq S \cup \{i\}} (-1)^{|S|+1-|L|} v(L) \\
&= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\}) \\
&= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\}) - \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= I(S|i \text{ is always present}) - I(S).
\end{aligned}
$$

(7) Interaction distribution axiom. This axiom characterizes how interactions are distributed for "interaction functions" (Sundararajan et al., 2020). An interaction function v_T parameterized by a context T satisfies $\forall S : T \subseteq S \subseteq N, v_T(S) = c$, and $v_T(S) = 0$ if $T \subsetneq S$. Then, we have $I(T) = c$, and $\forall S \neq T, I(S) = 0$.

• *Proof:* If $S \subsetneq T$, we have

$$
I(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} \cdot \underbrace{v(L)}_{\forall L \subsetneq S \subsetneq T, v(L)=0} = 0.
$$

If $S = T$, we have

$$I(S) = I(T) = \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L)$$

$$= v(T) + \sum_{L \subsetneq T} (-1)^{|T|-|L|} \underbrace{v(L)}_{=0} = c.$$

If $S \subsetneq T$, we have

$$I(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L)$$

$$= c \cdot \sum_{\substack{L \subseteq S \\ L \supseteq T}} (-1)^{|S|-|L|}$$

$$= c \cdot \sum_{m=0}^{|S|-|T|} \binom{|S|-|T|}{m} (-1)^m = 0.$$

A.2 PROOFS OF THEOREMS

In this section, we prove connections between the interaction utility $I(S)$ and several game-theoretic attributions/interactions. We first prove Theorem 3.2, which can be seen as the foundation for proofs of Theorem 3.1, 3.3, 3.4, and 3.5.

Theorem 3.2 (Connection to the marginal benefit) $\Delta v_T(S) = \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L \cup S)$ denotes the marginal benefit (Grabisch & Roubens, 1999) of variables in $T \subseteq N \setminus S$ given the environment S. We have proven that $\Delta v_T(S)$ can be decomposed into the sum of interaction utilities inside T and sub-environments of S, *i.e.* $\Delta v_T(S) = \sum_{S' \subseteq S} I(T \cup S')$.

• *Proof:* By the definition of the marginal benefit, we have

$$\Delta v_T(S) = \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L \cup S)$$

$$= \sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{K \subseteq L \cup S} I(K) \quad \text{// by Equation (1,left)}$$

$$= \sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{L' \subseteq L} \sum_{S' \subseteq S} I(L' \cup S') \quad \text{// since } L \cap S = \emptyset$$

$$= \sum_{S' \subseteq S} \left[\sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{L' \subseteq L} I(L' \cup S') \right]$$

$$= \sum_{S' \subseteq S} \left[\sum_{L' \subseteq T} \sum_{\substack{L \subseteq T \\ L \supseteq L'}} (-1)^{|T|-|L|} I(L' \cup S') \right]$$

$$= \sum_{S' \subseteq S} \left[\underbrace{I(S' \cup T)}_{L'=T} + \underbrace{\sum_{L' \subsetneq T} \left(\sum_{l=|L'|}^{|T|} \binom{|T|-|L'|}{l-|L'|} (-1)^{|T|-|L|} I(L' \cup S') \right)}_{L' \subsetneq T} \right]$$

$$= \sum_{S' \subseteq S} \left[I(S' \cup T) + \sum_{L' \subsetneq T} \left(I(L' \cup S') \cdot \underbrace{\sum_{l=|L'|}^{|T|} \binom{|T|-|L'|}{l-|L'|} (-1)^{|T|-|L|}}_{=0} \right) \right]$$

$$= \sum_{S' \subseteq S} I(S' \cup T) \quad \square$$

In particular, if T is a singleton set, *i.e.* $T = \{i\}$, we can obtain the conclusion in (Ren et al., 2021) that $\Delta v_{\{i\}}(S) = \sum_{L \subseteq S} I(L \cup \{i\})$.

The proof for Theorem 3.1, 3.3, and 3.4 are similar. Since the Shapley interaction index is a generalization of the Shapley value and the bi-variate interaction utility, we first prove Theorem 3.4, and then prove Theorem 3.1 and 3.3.

Theorem 3.1 (Connection to the Shapley value) Let $\phi(i)$ denote the Shapley value (Shapley, 1953) of an input variable i. Then, its Shapley value can be represented as the weighted sum of interaction utilities, $\phi(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{1}{|S|+1} I(S \cup \{i\})$. In other words, the utility of an interaction pattern with m variables should be equally assigned to the m variables in the computation of Shapley values.

• *Proof:* By the definition of the Shapley value, we have

$$\phi(i) = \mathbb{E}_{m} \mathbb{E}_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} [v(S \cup \{i\}) - v(S)]$$

$$= \frac{1}{|N|} \sum_{m=0}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \sum_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \left[v(S \cup \{i\}) - v(S) \right]$$

$$= \frac{1}{|N|} \sum_{m=0}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \sum_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \Delta v_{\{i\}}(S)$$

$$= \frac{1}{|N|} \sum_{m=0}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \sum_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \left[\sum_{L \subseteq S} I(L \cup \{i\}) \right] \quad \text{// by Theorem 3.2}$$

$$= \frac{1}{|N|} \sum_{L \subseteq N \setminus \{i\}} \sum_{m=0}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \sum_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m \\ S \supseteq L}} I(L \cup \{i\})$$

$$= \frac{1}{|N|} \sum_{L \subseteq N \setminus \{i\}} \sum_{m=|L|}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \sum_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m \\ S \supseteq L}} I(L \cup \{i\}) \quad \text{// since } S \supseteq L, |S| = m \geq |L|.$$

$$= \frac{1}{|N|} \sum_{L \subseteq N \setminus \{i\}} \sum_{m=|L|}^{|N|-1} \frac{1}{\binom{|N|-1}{m}} \cdot \binom{|N|-|L|-1}{m-|L|} I(L \cup \{i\})$$

$$= \frac{1}{|N|} \sum_{L \subseteq N \setminus \{i\}} I(L \cup \{i\}) \underbrace{\sum_{k=0}^{|N|-|L|-1} \frac{1}{\binom{|N|-1}{|L|+k}} \cdot \binom{|N|-|L|-1}{k}}_{w_L}$$

Then, we leverage the following properties of combinatorial numbers and the Beta function to simplify the term $w_L = \sum_{k=0}^{|N|-|L|-1} \frac{1}{\binom{|N|-1}{|L|+k}} \cdot \binom{|N|-|L|-1}{k}$.

(i) A property of combinitorial numbers. $m \cdot \binom{n}{m} = n \cdot \binom{n-1}{m-1}$.

(ii) The definition of the Beta function. For $p, q > 0$, the Beta function is defined as $B(p,q) = \int_0^1 x^{p-1}(1-x)^{1-q} dx$.

(iii) Connections between combinitorial numbers and the Beta function.

○ When $p, q \in \mathbb{Z}^+$, we have $B(p,q) = \frac{1}{q \cdot \binom{p+q-1}{p-1}}$.

○ For $m, n \in \mathbb{Z}^+$ and $n > m$, we have $\binom{n}{m} = \frac{1}{m \cdot B(n-m+1, m)}$.

16

$$w_L = \sum_{k=0}^{|N|-|L|-1} \frac{1}{\binom{|N|-1}{|L|+k}} \cdot \binom{|N|-|L|-1}{k}$$

$$= \sum_{k=0}^{|N|-|L|-1} \binom{|N|-|L|-1}{k} \cdot (|L|+k) \cdot B(|N|-|L|-k, |L|+k)$$

$$= \sum_{k=0}^{|N|-|L|-1} |L| \cdot \binom{|N|-|L|-1}{k} \cdot B(|N|-|L|-k, |L|+k) \qquad \cdots ①$$

$$+ \sum_{k=0}^{|N|-|L|-1} k \cdot \binom{|N|-|L|-1}{k} \cdot B(|N|-|L|-k, |L|+k) \qquad \cdots ②$$

Then, we solve ① and ② respectively. For ①, we have

$$① = \int_0^1 |L| \sum_{k=0}^{|N|-|L|-1} \binom{|N|-|L|-1}{k} \cdot x^{|N|-|L|-k-1} \cdot (1-x)^{|L|+k-1} \, dx$$

$$= \int_0^1 |L| \cdot \underbrace{\left[\sum_{k=0}^{|N|-|L|-1} \binom{|N|-|L|-1}{k} \cdot x^{|N|-|L|-k-1} \cdot (1-x)^k \right]}_{=1} \cdot (1-x)^{|L|-1} \, dx$$

$$= \int_0^1 |L|(1-x)^{|L|-1} \, dx = 1$$

For ②, we have

$$② = \sum_{k=1}^{|N|-|L|-1} (|N|-|L|-1) \cdot \binom{|N|-|L|-2}{k-1} \cdot B(|N|-|L|-k, |L|+k)$$

$$= (|N|-|L|-1) \sum_{k'=0}^{|N|-|L|-2} \binom{|N|-|L|-2}{k'} \cdot B(|N|-|L|-k'-1, |L|+k'+1)$$

$$= (|N|-|L|-1) \int_0^1 \sum_{k'=0}^{|N|-|L|-2} \binom{|N|-|L|-2}{k'} \cdot x^{|N|-|L|-k'-2} \cdot (1-x)^{|L|+k'} \, dx$$

$$= (|N|-|L|-1) \int_0^1 \underbrace{\left[\sum_{k'=0}^{|N|-|L|-2} \binom{|N|-|L|-2}{k'} \cdot x^{|N|-|L|-k'-2} \cdot (1-x)^{k'} \right]}_{=1} \cdot (1-x)^{|L|} \, dx$$

$$= (|N|-|L|-1) \int_0^1 (1-x)^{|L|} \, dx = \frac{|N|-|L|-1}{|L|+1}$$

Hence, we have

$$w_L = ① + ② = 1 + \frac{|N|-|L|-1}{|L|+1} = \frac{|N|}{|L|+1}$$

Therefore, we proved $\phi(i) = \frac{1}{|N|} \sum_{S \subseteq N \setminus \{i\}} w_L \cdot I(L \cup \{i\}) = \sum_{S \subseteq N \setminus \{i\}} \frac{1}{|S|+1} I(S \cup \{i\})$. $\qquad \square$

Theorem 3.3 (Connection to the Shapley interaction index) Given a subset of input variables $T \subseteq N$, $I^{\text{Shapley}}(T) = \sum_{S \subseteq N \setminus T} \frac{|S|!(|N|-|S|-|T|)!}{(|N|-|T|+1)!} \Delta v_T(S)$ denotes the Shapley interaction index (Grabisch & Roubens, 1999) of T. We have proven that the Shapley interaction index can be represented as the weighted sum of utilities of interaction patterns, $I^{\text{Shapley}}(T) = \sum_{S \subseteq N \setminus T} \frac{1}{|S|+1} I(S \cup T)$.

- *Proof:*

$$I^{\text{Shapley}}(T) = \sum_{S \subseteq N \setminus T} \frac{|S|!(|N| - |S| - |T|)!}{(|N| - |T| + 1)!} \Delta v_T(S)$$

$$= \frac{1}{|N| - |T| + 1} \sum_{m=0}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \Delta v_T(S)$$

$$= \frac{1}{|N| - |T| + 1} \sum_{m=0}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \left[\sum_{L \subseteq S} I(L \cup T) \right]$$

$$= \frac{1}{|N| - |T| + 1} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m \\ S \supseteq L}} I(L \cup T)$$

$$= \frac{1}{|N| - |T| + 1} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \binom{|N| - |L| - |T|}{m - |L|} I(L \cup T)$$

$$= \frac{1}{|N| - |T| + 1} \sum_{L \subseteq N \setminus T} I(L \cup T) \underbrace{\sum_{k=0}^{|N|-|L|-|T|} \frac{1}{\binom{|N|-|T|}{|L|+k}} \binom{|N| - |L| - |T|}{k}}_{w_L}$$

Just like the proof of Theorem 3.1, we leverage the properties of combinitorial numbers and the Beta function to simplify w_L.

$$w_L = \sum_{k=0}^{|N|-|L|-|T|} \frac{1}{\binom{|N|-|T|}{|L|+k}} \binom{|N| - |L| - |T|}{k}$$

$$= \sum_{k=0}^{|N|-|L|-|T|} \binom{|N| - |L| - |T|}{k} \cdot \left(|L| + k \right) \cdot B\left(|N| - |L| - |T| - k + 1, |L| + k \right)$$

$$= \sum_{k=0}^{|N|-|L|-|T|} |L| \cdot \binom{|N| - |L| - |T|}{k} \cdot B\left(|N| - |L| - |T| - k + 1, |L| + k \right) \quad \cdots \text{①}$$

$$+ \sum_{k=0}^{|N|-|L|-|T|} k \cdot \binom{|N| - |L| - |T|}{k} \cdot B\left(|N| - |L| - |T| - k + 1, |L| + k \right) \quad \cdots \text{②}$$

Then, we solve ① and ② respectively. For ①, we have

$$\text{①} = \int_0^1 |L| \sum_{k=0}^{|N|-|L|-|T|} \binom{|N| - |L| - |T|}{k} \cdot x^{|N|-|L|-|T|-k} \cdot (1 - x)^{|L|+k-1} \, dx$$

$$= \int_0^1 |L| \cdot \underbrace{\left[\sum_{k=0}^{|N|-|L|-|T|} \binom{|N| - |L| - |T|}{k} \cdot x^{|N|-|L|-|T|-k} \cdot (1 - x)^k \right]}_{=1} \cdot (1 - x)^{|L|-1} \, dx$$

$$= \int_0^1 |L| \cdot (1 - x)^{|L|-1} \, dx = 1$$

For ②, we have

$$② = \sum_{k=1}^{|N|-|L|-|T|} (|N|-|L|-|T|) \binom{|N|-|L|-|T|-1}{k-1} \cdot B\Big(|N|-|L|-|T|-k+1, |L|+k\Big)$$

$$=(|N|-|L|-|T|) \sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot B\Big(|N|-|L|-|T|-k', |L|+k'+1\Big)$$

$$=(|N|-|L|-|T|) \int_0^1 \sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot x^{|N|-|L|-|T|-k'-1} \cdot (1-x)^{|L|+k'} \, dx$$

$$=(|N|-|L|-|T|) \int_0^1 \underbrace{\left[\sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot x^{|N|-|L|-|T|-k'-1} \cdot (1-x)^{k'} \right]}_{=1} \cdot (1-x)^{|L|} \, dx$$

$$=(|N|-|L|-|T|) \int_0^1 (1-x)^{|L|} \, dx = \frac{|N|-|L|-|T|}{|L|+1}$$

Hence, we have

$$w_L = ① + ② = 1 + \frac{|N|-|L|-|T|}{|L|+1} = \frac{|N|-|T|+1}{|L|+1}$$

Therefore, we proved that $I^{\text{Shapley}}(T) = \frac{1}{|N|-|T|+1} \sum_{L \subseteq N \setminus T} w_L \cdot I(L \cup T) = \sum_{L \subseteq N \setminus T} \frac{1}{|L|+1} I(L \cup T)$.

Theorem 3.4 (Connection to the Shapley Taylor interaction index) Given a subset of input variables $T \subseteq N$, let $I^{\text{Shapley-Taylor}}(T)$ denote the Shapley Taylor interaction index (Sundararajan et al., 2020) of order k for T. We have proven that the Shapley taylor interaction index can be represented as the weighted sum of interaction utilities, *i.e.* $I^{\text{Shapley-Taylor}}(T) = I(T)$ if $|T| < k$; $I^{\text{Shapley-Taylor}}(T) = \sum_{S \subseteq N \setminus T} \binom{|S|+k}{k}^{-1} I(S \cup T)$ if $|T| = k$; and $I^{\text{Shapley-Taylor}}(T) = 0$ if $|T| > k$.

• *Proof:* By the definition of the Shapley Taylor interaction index,

$$I^{\text{Shapley-Taylor}(k)}(T) = \begin{cases} \Delta v_T(\emptyset) & \text{if } |T| < k \\ \frac{k}{|N|} \sum_{S \subseteq N \setminus T} \frac{1}{\binom{|N|-1}{|S|}} \Delta v_T(S) & \text{if } |T| = k \\ 0 & \text{if } |T| > k \end{cases}$$

When $|T| < k$, by the definition of the interaction utility in Equation (1), we have

$$I^{\text{Shapley-Taylor}(k)}(T) = \Delta v_T(\emptyset) = \sum_{L \subseteq T} (-1)^{|T|-|L|} \cdot v(L) = I(T).$$

When $|T| = k$, we have

$$I^{\text{Shapley-Taylor}(k)}(T) = \frac{k}{|N|} \sum_{S \subseteq N \setminus T} \frac{1}{\binom{|N|-1}{|S|}} \cdot \Delta v_T(S)$$

$$= \frac{k}{|N|} \sum_{m=0}^{|N|-k} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \frac{1}{\binom{|N|-1}{|S|}} \cdot \Delta v_T(S)$$

$$= \frac{k}{|N|} \sum_{m=0}^{|N|-k} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \frac{1}{\binom{|N|-1}{|S|}} \left[\sum_{L \subseteq S} I(L \cup T) \right]$$

$$= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-k} \frac{1}{\binom{|N|-1}{|S|}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m \\ S \supseteq L}} I(L \cup T)$$

$$= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-k} \frac{1}{\binom{|N|-1}{|S|}} \binom{|N|-|L|-k}{m-|L|} I(L \cup T)$$

$$= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} I(L \cup T) \underbrace{\sum_{m=0}^{|N|-|L|-k} \frac{1}{\binom{|N|-1}{|L|+m}} \binom{|N|-|L|-k}{m}}_{w_L}$$

Just like the proof of Theorem 3.1, we leverage the properties of combinitorial numbers and the Beta function to simplify w_L.

$$w_L = \sum_{m=0}^{|N|-|L|-k} \frac{1}{\binom{|N|-1}{|L|+m}} \binom{|N|-|L|-k}{m}$$

$$= \sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot \left(|L|+m \right) \cdot B\left(|N|-|L|-m, |L|+m \right)$$

$$= \sum_{m=0}^{|N|-|L|-k} |L| \cdot \binom{|N|-|L|-k}{m} \cdot B\left(|N|-|L|-m, |L|+m \right) \qquad \cdots \text{①}$$

$$+ \sum_{m=0}^{|N|-|L|-k} m \cdot \binom{|N|-|L|-k}{m} \cdot B\left(|N|-|L|-m, |L|+m \right) \qquad \cdots \text{②}$$

Then, we solve ① and ② respectively. For ①, we have

$$\text{①} = \int_0^1 |L| \cdot \sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot x^{|N|-|L|-m-1} \cdot (1-x)^{|L|+m-1} \, dx$$

$$= \int_0^1 |L| \cdot \underbrace{\left[\sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot x^{|N|-|L|-m-k} \cdot (1-x)^m \right]}_{=1} \cdot x^{k-1} \cdot (1-x)^{|L|-1} \, dx$$

$$= \int_0^1 |L| \cdot x^{k-1} \cdot (1-x)^{|L|-1} \, dx = |L| \cdot B(k, |L|) = \frac{1}{\binom{|L|+k-1}{k-1}}$$

For ②, we have

$$\text{②} = \sum_{m=1}^{|N|-|L|-k} (|N| - |L| - k) \cdot \binom{|N| - |L| - k - 1}{m - 1} \cdot B\Big(|N| - |L| - m, |L| + m\Big)$$

$$= \sum_{m'=0}^{|N|-|L|-k-1} (|N| - |L| - k) \cdot \binom{|N| - |L| - k - 1}{m'} \cdot B\Big(|N| - |L| - m' - 1, |L| + m' + 1\Big)$$

$$= \int_0^1 (|N| - |L| - k) \sum_{m'=0}^{|N|-|L|-k-1} \binom{|N| - |L| - k - 1}{m'} \cdot x^{|N|-|L|-m'-2} \cdot (1 - x)^{|L|+m'} \, dx$$

$$= \int_0^1 (|N| - |L| - k) \underbrace{\left[\sum_{m'=0}^{|N|-|L|-k-1} \binom{|N| - |L| - k - 1}{m'} \cdot x^{|N|-|L|-m'-k-1} \cdot (1 - x)^{m'} \right]}_{=1} \cdot x^{k-1} \cdot (1 - x)^{|L|} \, dx$$

$$= \int_0^1 (|N| - |L| - k) \cdot x^{k-1} \cdot (1 - x)^{|L|} \, dx = (|N| - |L| - k) \cdot B(k, |L| + 1)$$

$$= \frac{|N| - |L| - k}{(|L| + 1)\binom{|L|+k}{k-1}}$$

Hence, we have

$$\begin{aligned}
w_L &= \text{①} + \text{②} = \frac{1}{\binom{|L|+k-1}{k-1}} + \frac{|N| - |L| - k}{(|L| + 1)\binom{|L|+k}{k-1}} \\
&= \frac{|L|! \cdot (k - 1)!}{(|L| + k - 1)!} + \frac{|N| - |L| - k}{|L| + 1} \cdot \frac{(|L| + 1)! \cdot (k - 1)!}{(|L| + k)!} \\
&= \frac{|L|! \cdot (k - 1)!}{(|L| + k - 1)!} + \frac{|N| - |L| - k}{|L| + k} \cdot \frac{|L|! \cdot (k - 1)!}{(|L| + k - 1)!} \\
&= \left[1 + \frac{|N| - |L| - k}{|L| + k} \right] \cdot \frac{|L|! \cdot (k - 1)!}{(|L| + k - 1)!} \\
&= \frac{|N|}{|L| + k} \cdot \frac{|L|! \cdot (k - 1)!}{(|L| + k - 1)!} \\
&= \frac{|N|}{k} \cdot \frac{|L|! \cdot k!}{(|L| + k)!} \\
&= \frac{|N|}{k} \cdot \frac{1}{\binom{|L|+k}{k}}
\end{aligned}$$

Therefore, we proved that when $|T| = k$, $I^{\text{Shapley-Taylor}}(T) = \frac{k}{|N|} \sum_{L \subseteq N \setminus T} w_L \cdot I(L \cup T) = \frac{k}{|N|} \sum_{L \subseteq N \setminus T} \frac{|N|}{k} \cdot \frac{1}{\binom{|L|+k}{k}} \cdot I(L \cup T) = \sum_{L \subseteq N \setminus T} \binom{|L|+k}{k}^{-1} I(L \cup T)$.

B   PROOF OF THE RELATIONSHIP BETWEEN TWO TYPES OF OBJECTIVENESS AND COROLLARY 1

Section 3.2 formally defined two types of objectiveness. In this section, we prove the relationship between them, *i.e.* Definition 2 is actually a necessary condition of Definition 1. We also prove the Corollary 1 from Definition 2.

Proof of the relationship between Definition 1 and Definition 2. Since $g(N|x) = v(N|x)$ holds for any input $x \in \mathcal{R}^n$, it also holds for the input x^{mask} when some of input variables in x are masked by their baseline values, *i.e.* $g(N|x^{\text{mask}}) = v(N|x^{\text{mask}})$. Let S be an arbitrary subset of N, and x^{mask} is obtained by replacing variables out of S with their baseline values, as follows.

$$x^{\text{mask}} = \begin{cases} x_i, & i \in S \\ r_i, & \text{otherwise} \end{cases}, \tag{6}$$

where r_i denotes the baseline value of the i-th input variable, as discussed in Section 3.3. In this case, $g(S|x) = g(N|x^{\text{mask}})$ and $v(S|x) = v(N|x^{\text{mask}})$. Therefore, we have $g(S|x) = v(S|x)$. Because S is an arbitrary subset of N, we have proved that given an input sample $x \in \mathcal{R}^n$, $\forall S \subseteq N$, $g(S|x) = v(S|x)$. Hence, Definition 2 is a necessary condition of Definition 1.

Proof of Corollary 1. According to Definition 2, given a certain input sample x, $\forall S \subseteq N, g(S|x) = v(S|x)$. Thus, we have $I_v(S|x) = \sum_{L \subseteq S}(-1)^{|S|-|L|}v(S|x) = \sum_{L \subseteq S}(-1)^{|S|-|L|}g(S|x) = I_g(S|x)$. Thus, we proved Corollary 1.

C SIMPLIFYING THE EXPLANATION USING THE MINIMUM DESCRIPTION LENGTH PRINCIPLE

In this section, we discussed the algorithm of extracting common coalitions to minimize the total description length in Equation (5). Given an AOG g and input variables N, let $M = N \cup \Omega^{\text{coalition}}$ denote the set of all terminal nodes and AND nodes in the bottom two layers (*e.g.* $M = N \cup \Omega^{\text{coalition}} = \{A, B, C, D, E\} \cup \{\alpha, \beta, \gamma\}$ in Figure 1 (right)). The total description length $L(g, M)$ was given in Equation (5).

In order to minimize $L(g, M)$, we used the greedy strategy to iteratively extract common coalitions of input variables. In each iteration, we chose the coalition $\alpha \subseteq N$ which most efficiently decreased the total description length. Then we took this coalition as an AND node, and added it into $\Omega^{\text{coalition}}$ in the third layer of the AOG. The efficiency of a coalition α *w.r.t.* the decrease of the total description length was defined as follows.

$$\delta(\alpha) = \frac{\Delta L}{|\alpha|} = \frac{L(g, M \cup \{\alpha\}) - L(g, M)}{|\alpha|}, \tag{7}$$

where $L(g, M)$ denoted the total description length without using the newly added coalition α, and $L(g, M \cup \{\alpha\})$ denoted the total description when we added the node α to further simplify the description of g. $|\alpha|$ denotes the number of input variables in α. We iteratively extracted the most efficient coalitions α to minimize the total description length. The extracting process stopped when there was no new coalition α could further reduce the total description length (*i.e.* $\forall \alpha \notin M, L(g, M \cup \{\alpha\}) - L(g, M) > 0$), or the most efficient α was not shared by multiple patterns. Alg. 1 shows the pseudo-code of this algorithm.

Algorithm 1: The greedy algorithm to minimize total description length $L(g, M)$

Input: The set of terminal nodes N, the set of top-k salient patterns $\Omega_{\text{top-}k}^{\text{salient}}$, the interaction utilities of these patterns $\{I(S)\}_{S \in \Omega_{\text{top-}k}^{\text{salient}}}$, the maximum iteration times T, the coefficient λ in Equation (5)

Output: The set of nodes in the bottom two layers $M = N \cup \Omega^{\text{coalition}}$

1 Initialize $\Omega^{\text{coalition}} = \emptyset$ and $M = N \cup \Omega^{\text{coalition}}$
2 **for** *iteration 1 to T* **do**
3   **foreach** *possible coalition $\alpha \subseteq N$* **do**
4   Calculate the efficiency $\delta(\alpha)$ according to Equation (7)
5   **end**
6   Select α as the coalition whose $\delta(\alpha)$ is the smallest
7   **if** $\delta(\alpha) > 0$ *or α co-appears in only one pattern* **then**
8   **break**
9   **end**
10   Update AND nodes, $\Omega^{\text{coalition}} \leftarrow \Omega^{\text{coalition}} \cup \{\alpha\}$
11   Rewrite each pattern $S \in \Omega_{\text{top-}k}^{\text{salient}}$ according to $M = N \cup \Omega^{\text{coalition}}$
12 **end**
13 **return** $M = N \cup \Omega^{\text{coalition}}$

D EXPERIMENT DETAILS

D.1 DATASETS AND MODELS

Datasets. We conducted experiments on both tasks of natural language processing and the classification/regression tasks based on tabular datasets. For natural language processing, we used the SST-2 dataset (Socher et al., 2013) for sentiment prediction and the CoLA dataset (Warstadt et al., 2019) for linguistic acceptability. For tabular datasets, we used the UCI census income dataset (census) (Dua & Graff, 2017), the UCI bike sharing dataset (bike) (Fanaee-T & Gama, 2013), and the UCI TV news channel commercial detection dataset (TV news) (Dua & Graff, 2017). We followed (Covert et al., 2020; Covert & Lee, 2021) to conduct data pre-processing for these tabular datasets. We also normalized data in each dataset to zero mean and unit variance.

Models. We trained LSTMs and CNNs based on NLP datasets. The LSTM was unidirectional and had two layers, with a hidden layer of size 100. The architecture of the CNN was the same as the network architecture in (Rakhlin, 2016). Besides, for tabular datasets, we followed (Covert et al., 2020; Covert & Lee, 2021) to train LightGBMs (Ke et al., 2017), XGBoost (Chen & Guestrin, 2016), and two-layer MLPs (namely *MLP-2*). We also trained five-layer MLPs (namely *MLP-5*) and five layer MLPs with skip-connections (namely *ResMLP-5*) on these datasets. For the ResMLP-5, we added a skip connection to each fully connected layer of the MLP-5. Figure 7 shows the architecture of the ResMLP-5. The hidden layers in MLP-5 and ResMLP-5 had the same width of 100. In our experiment, we also learned MLP-2, MLP-5, and ResMLP-5 on each tabular dataset via adversarial training (Madry et al., 2018). During adversarial training, adversarial examples $x^{\mathrm{adv}} = x + \delta$ were generated by the ℓ_∞ PGD attack, where $\|\delta\|_\infty \leq 0.1$. The attack was iterated for 20 steps with the step size of 0.01.
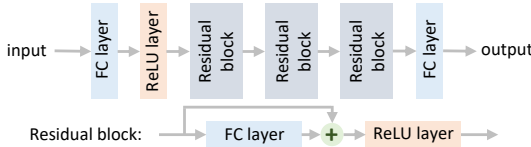

Figure 7: The architecture of the ResMLP-5.

Accuracy of the models. Table 3 reports the classification accuracy of models trained on the commercial dataset, the classification accuracy of models trained on the census dataset, and the mean squared error of models trained on the bike dataset. Table 4 reports the classification accuracy of models trained on the CoLA dataset and the SST-2 dataset.

Table 3: Classification accuracy (on commercial and census dataset) and mean squared error (on bike dataset) of different models.

Dataset	MLP-2		MLP-5		ResMLP-5		XGBoost	LightGBM
	normal	adversarial	normal	adversarial	normal	adversarial		
commercial	83.11%	78.49%	79.86%	80.24%	79.01%	80.13%	84.48%	84.19%
census	79.91%	75.77%	78.96%	77.79%	80.49%	77.99%	87.35%	87.54%
bike	-	-	2161.47	3080.73	2149.43	2708.59	1623.71	-

Table 4: Accuracy of models trained on NLP datasets.

Dataset	LSTM	CNN
CoLA	64.42%	65.79%
SST-2	86.83%	78.19%

D.2 DETAILS OF EXPERIMENTS ON SYNTHESIZED FUNCTIONS AND DATASETS

This section provides details of synthesized functions and datasets used in Section 4.

The Addition-Multiplication dataset (Zhang et al., 2021). This dataset contained 100 functions, which only consisted of addition and multiplication operations. For example, $v(\mathbf{x}) = x_1 + x_2 x_3 + x_3 x_4 x_5 + x_4 x_6$. Each variable x_i was a binary variable, *i.e.* $x_i \in \{0, 1\}$.

The ground-truth interaction patterns corresponding to these functions can be easily determined. For each term in these functions (*e.g.* the term of $x_1, x_2 x_3, x_3 x_4 x_5, x_4 x_6$ in the function $v(\mathbf{x}) = x_1 + x_2 x_3 + x_3 x_4 x_5 + x_4 x_6$), only when variables contained by this term were all present, $x_1 = 1$, this term would contribute to the output. Therefore, we could consider input variables in each term formed a ground-truth interaction pattern. In the above example function, given the input $\mathbf{x} = (1, 1, 1, 1, 1, 1)$, the ground-truth interaction patterns were $\Omega^{\mathrm{truth}} = \{\{x_1\}, \{x_2, x_3\}, \{x_3, x_4, x_5\}, \{x_4, x_6\}\}$. Given the input $\mathbf{x} = (1, 1, 0, 1, 1, 1)$, the ground-truth interaction patterns were $\Omega^{\mathrm{truth}} = \{\{x_1\}, \{x_4, x_6\}\}$.

In our experiments, we randomly generated 100 Addition-Multiplication functions. Each of them had 10 input variables, and had 10 to 100 terms. Then, we randomly generated 200 binary input samples for each of these functions. For each input sample, let $k = |\Omega^{\text{truth}}|$ denote the number of the labeled ground-truth patterns. For fair comparison, we computed interaction utilities $I(S)$ and extracted the top-k salient patterns $\Omega^{\text{salient}}_{\text{top-}k}$. Then, we averaged the value of IoU $= \frac{|\Omega^{\text{salient}}_{\text{top-}k} \cap \Omega^{\text{truth}}|}{|\Omega^{\text{salient}}_{\text{top-}k} \cup \Omega^{\text{truth}}|}$ over all samples.

The dataset in (Ren et al., 2021). This dataset contained 100 functions, which consisted of addition, subtraction, multiplication, and the sigmoid operations. Just like the Addition-Multiplication dataset, the ground-truth interaction patterns in this dataset could also be easily determined. Let us consider the function $v(\mathbf{x}) = -x_1 x_2 x_3 - \text{sigmoid}(5x_4 x_5 - 5x_6 - 2.5), x_i \in \{0, 1\}$ as an example. The term $x_1 x_2 x_3$ was activated ($= 1$) if and only if $x_1 = x_2 = x_3 = 1$. The term $\text{sigmoid}(5x_4 x_5 - 5x_6 - 2.5)$ was activated (> 0.5) if and only if $x_4 = x_5 = 1$ and $x_6 = 0$. Thus, we could also consider this function contained two ground-truth interaction pattern. In other words, for the above function, given the input $\mathbf{x} = (1, 1, 1, 1, 1, 0)$, the ground-truth interaction patterns were $\Omega^{\text{truth}} = \{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6\}\}$. Given the input $\mathbf{x} = (1, 1, 1, 1, 1, 1)$, the ground-truth interaction patterns were $\Omega^{\text{truth}} = \{\{x_1, x_2, x_3\}\}$.

In our experiments, we followed (Ren et al., 2021) to randomly generated 100 functions. Each of them had 6 to 12 input variables. Then, we randomly generated 200 binary input samples for each of these functions. Just like the Addition-Multiplication dataset, we extracted the top-k ($k = |\Omega^{\text{truth}}|$) salient patterns $\Omega^{\text{salient}}_{\text{top-}k}$, and computed the average IoU between Ω^{truth} and $\Omega^{\text{salient}}_{\text{top-}k}$ over all samples for comparison.

The manually labeled And-Or dataset. This dataset contained 10 functions with AND operations (denoted by $\&$) and OR operations (denoted by $|$). For example, the function $f(\mathbf{x}) = (x_1 > 0)\&(x_2 > 0)|(x_2 > 0)\&(x_3 > 0)\&(x_4 > 0)|(x_3 > 0)\&(x_5 > 0)$. Each input variable is a scalar, *i.e.* $x_i \in \mathbb{R}$, and the output is binary, *i.e.* $f(\mathbf{x}) \in \{0, 1\}$. For each And-Or function, we randomly generated 100,000 Gaussian noises with $n = 8$ variables as input samples, and labeled these samples following functions in the And-Or dataset, namely the *manually labeled And-Or dataset*.

The ground-truth interaction patterns in this dataset could be determined as follows. For the above function, we could consider $\{x_1, x_2\}$, $\{x_2, x_3, x_4\}$, and $\{x_3, x_5\}$ as possible interaction patterns. If any of these patterns was significantly activated, *i.e.* if all input variables in this pattern were greater than a threshold $\tau = 0.5$, then we consider this pattern is significant enough to be a valid ground-truth interaction pattern. *I.e.* for the above function, given the input $\mathbf{x} = (1.0, 2.0, 1.5, 0.9, 0.8)$, the ground-truth interaction patterns were $\Omega^{\text{truth}} = \{\{x_1, x_2\}, \{x_2, x_3, x_4\}, \{x_3, x_5\}\}$. Given the input $\mathbf{x} = (0.8, 1.5, 1.2, 0.1, 0.9)$, the ground-truth interaction patterns were $\Omega^{\text{truth}} = \{\{x_1, x_2\}, \{x_3, x_5\}\}$.

In our experiments, we trained one MLP-5 and one ResMLP-5 networks for binary classification based on the manually labeled dataset generated based on each And-Or function. For each well-trained model, just like the above two datasets, we extracted the top-k salient patterns and computed the average IoU over 1000 training samples for comparison. Note that there was no principle to ensure that the model learned exactly the ground-truth interactions between input variables for inference. Therefore, the average IoU on this dataset was less than 1.

The re-labeled TV news dataset. This dataset was extended from the TV news dataset. Just like the above manually labeled And-Or dataset, we re-labeled samples in the TV news dataset following 10 functions only with AND operations and OR operations.

We also trained MLP-5 and ResMLP-5 networks for binary classification based on the re-labeled TV news dataset generated based on each And-Or function. We used the same method in the manually labeled And-Or dataset to determine the ground-truth interaction patterns for each input sample. Then, we averaged the value of the IoU between Ω^{truth} and $\Omega^{\text{salient}}_{\text{top-}k}$ over the 1,000 training samples, as discussed above. In this dataset, there was also no principle to ensure that the model learned exactly the ground-truth interactions between input variables for inference. Therefore, the average IoU on this dataset was also less than 1.

An extended version of the Addition-Multiplication dataset. In order to evaluate the correctness of the computed interaction utilities, we also extended the Addition-Multiplication dataset to generate functions with not only ground-truth interaction patterns, but also ground-truth interaction utilities for evaluation. The extended Addition-Multiplication dataset also contained 100 functions,

which consisted of addition and multiplication operations. Each variable x_i was a binary variable, *i.e.* $x_i \in \{0, 1\}$. Different from functions in the Addition-Multiplication dataset, there were different coefficients before each term in each function. For example, $v(\mathbf{x}) = 3x_1 - 2x_2 x_3 - x_3 x_4 x_5 + 5x_4 x_6$.

The ground-truth interaction utilities in these functions can be easily determined. Just like the original Addition-Multiplication dataset, each term was a ground-truth pattern. In this case, we could consider the interaction utility of each pattern as the value of its coefficient. For the above function, given the input $\mathbf{x} = (1, 1, 1, 1, 1, 1)$, the ground-truth utilities of interaction patterns were $I(\{x_1\}) = 3, I(\{x_2, x_3\}) = -2, I(\{x_3, x_4, x_5\}) = -1, I(\{x_4, x_6\}) = 5$, and for other $S \subseteq \{x_1, ..., x_6\}$, $I(S) = 0$. Given the input $\mathbf{x} = (1, 1, 0, 1, 1, 1)$, the ground-truth interaction utilities were $I(\{x_1\}) = 3, I(\{x_4, x_6\}) = 5$, and for other $S \subseteq \{x_1, ..., x_6\}$, $I(S) = 0$.

In our experiments, we randomly generated 100 functions. Each of them had 10 input variables, and had 10 to 100 terms. Then, we randomly generated 200 binary input samples for each of these functions. For each input sample, we measured the Jaccard similarity coefficient between ground-truth interaction utilities $I^{\text{truth}}(S)$ and interaction utilities $I(S)$ computed by our method as $J = \frac{\sum_{S \subseteq N} \min(|I^{\text{truth}}(S)|, |I(S)|)}{\sum_{S \subseteq N} \max(|I^{\text{truth}}(S)|, |I(S)|)}$. The average value of J over all samples was 1.00, indicating that our interaction metric based on the Harsanyi dividend correctly extracted the interaction utilities in these functions.

D.3 SETTINGS OF $v(\cdot)$

In the computation of interaction utilities, people usually used different settings of $v(\cdot)$. For example, Lundberg & Lee (2017) directly set $v(S) = p(y^{\text{truth}}|x^{\text{mask}})$ in the classification task, where x^{mask} was defined in Equation (6). $p(y^{\text{truth}}|x^{\text{mask}})$ denoted the output probability of the ground-truth category. Covert et al. (2020) computed $v(S)$ as the cross-entropy loss. In this paper, we used $v(S) = \log \frac{p(y^{\text{truth}}|x^{\text{mask}})}{1 - p(y^{\text{truth}}|x^{\text{mask}})}$ for models learned on classification tasks. For models learned on regression tasks, we directly computed $v(S)$ as the scalar output of the model on x^{mask}.

E VERIFICATION OF THEOREM 3.1

In this section, we conducted an experiment to verify Theorem 3.1, *i.e.* the Shapley value can be accurately approximated by the Harsanyi dividend. Let $\phi(i)$ denote the accurate Shapley value and $\tilde{\phi}(i)$ denote the estimated Shapley value using the Harsanyi dividend $I(S)$, which was mentioned in Section 3.1. In order to estimate the Shapley value via interaction utilities $I(S)$, we first selected the most salient k interaction patterns $\Omega_{\text{top-}k}^{\text{salient}}$. Then, according to Theorem 3.1, we computed the estimated Shapley value $\tilde{\phi}(i)$ as $\tilde{\phi}(i) = \sum_{S \in \Omega_{\text{top-}k}^{\text{salient}}:S \ni i} \frac{1}{|S|} I(S)$. Figure 8 shows the cosine similarity between the accurate Shapley values $\phi = [\phi(1), \phi(2), \ldots, \phi(n)] \in \mathbb{R}^n$ and the estimated Shapley values $\tilde{\phi} = [\tilde{\phi}(1), \tilde{\phi}(2), \ldots, \tilde{\phi}(n)] \in \mathbb{R}^n$, when we used different ratios of salient patterns $\frac{k}{2^n}$ to approximate the Shapley value. It indicated that the interaction utilities based on the Harsanyi dividend could accurately approximate the Shapley value.
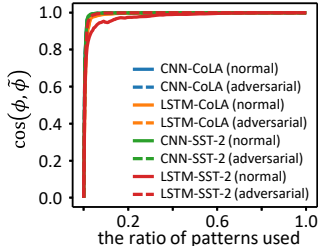

Figure 8: The cosine similarity between the accurate Shapley value ϕ and the estimated Shapley value $\tilde{\phi}$, when we used different ratios of salient patterns for estimation.

Table 5: Non-objectiveness ($\downarrow$) of different explanation methods. The AOG explainer exhibited the strongest objectiveness.

Explanation method		TV news		census		bike	
		MLP-5	ResMLP-5	MLP-5	ResMLP-5	MLP-5	ResMLP-5
Distillation-based explainer models	GBT	0.7661	0.4305	9.3453	2.1947	1.5120	1.9422
	SDT	1.1612	0.8991	25.9458	4.0938	-	-
	distillation	0.6718	0.5573	12.7566	3.8734	1.4025	1.6904
Attribution-based explainer models	Shapley value	0.7897	0.6157	11.0216	2.3912	1.7850	3.4056
	Input × Gradient	2.5373	2.2533	78.2222	7.5902	5.7450	6.9758
	LRP	1.3558	7.9203	49.9572	114.5870	5.7450	152.1785
	Occlusion	2.7904	1.9336	62.8212	9.1191	9.1114	13.3653
Ours		3.58×10^{-7}	1.93×10^{-7}	3.68×10^{-6}	1.16×10^{-6}	1.12×10^{-6}	1.54×10^{-6}

F ANALYSIS ON THE OBJECTIVENESS OF THE AOG EXPLAINER

F.1 THE OBJECTIVENESS OF THE AOG EXPLAINER

In this section, we proposed a metric to evaluate the objectiveness of explainer models, according to definitions in Section 3.2. Given a pre-trained model $v(\cdot)$ and a corresponding explainer model $g(\cdot)$, we defined the metric $\rho^{\text{non-obj}}$ to measure the non-objectiveness of the explainer model g, as follows.

$$\rho^{\text{non-obj}} = \frac{1}{|v(N)|} \mathop{\mathbb{E}}_{S \subseteq N} \Big[\big| v(S) - g(S) \big| \Big] \tag{8}$$

$\rho^{\text{non-obj}}$ measured the components of the deep model output $v(S)$ that was not explained by the output $g(S)$ of the explainer model. For fair comparisons, $|v(N)|$ was used for normalization. A lower value of $\rho^{\text{non-obj}}$ indicated a stronger objectiveness.

We compared the objectiveness of the AOG explainer with both *distillation-based explainer models* and *attribution-based explanations*, including GBT (Che et al., 2016), SDT (Frosst & Hinton, 2017), knowledge distillation (Hinton et al., 2015), the Shapley value (Shapley, 1953), Input×Gradient (Shrikumar et al., 2016), LRP (Bach et al., 2015), and Occlusion (Zeiler & Fergus, 2014). Due to their different formulation for explanations, the computation of $g(S)$ in different explanation methods were different, which were discussed as follows.

• For *distillation-based explainer models*, given the input sample x, let $f_{\text{explainer}}(x)$ denote the output of the explainer model. Then, $g(S)$ was computed as the output of the explainer model when only variables in S were given. In this case, input variables out of S were masked by their baseline values to represent the absence of these variables, as follows.

$$g(S) = f_{\text{explainer}}(x^{\text{mask}}), \quad x^{\text{mask}} = \begin{cases} x_i, & i \in S \\ r_i, & \text{otherwise} \end{cases} \tag{9}$$

where r_i denoted the baseline value of the i-th input variable.

• For *attribution-based explainer models*, given the input sample x, let $\phi_{\text{Shapley}}(i)$, $\phi_{\text{IG}}(i)$, $\phi_{\text{LRP}}(i)$, $\phi_{\text{Occ}}(i)$ denote the attribution of the input variable i computed by the Shapley value, Input × Gradient, LRP, and Occlusion, respectively. For Input × Gradient, LRP, and Occlusion, $g(S)$ was computed as the sum of attributions of input variables in S, as follows.

$$g_{\text{IG}}(S) = \sum_{i \in S} \phi_{\text{IG}}(i), \quad g_{\text{LRP}}(S) = \sum_{i \in S} \phi_{\text{LRP}}(i), \quad g_{\text{Occ}}(S) = \sum_{i \in S} \phi_{\text{Occ}}(i) \tag{10}$$

In particular, due to the efficiency property of the Shapley value (Shapley, 1953), *i.e.* $v(N) = v(\emptyset) + \sum_{i \in N} \phi_{\text{Shapley}}(i)$, $g(S)$ was computed as follows.

$$g_{\text{Shapley}}(S) = v(\emptyset) + \sum_{i \in S} \phi_{\text{Shapley}}(i) \tag{11}$$

In this way, we have $g_{\text{Shapley}}(N) = v(\emptyset) + \sum_{i \in N} \phi_{\text{Shapley}}(i) = v(N)$.

• For *the AOG explainer*, given the input sample x, $g(S)$ was computed as the sum of interaction utilities of all patterns in S, as follows.

$$g_{\text{AOG}}(S) = \sum_{L \subseteq S} I(L) \tag{12}$$

Table 6: Non-objectiveness ($\downarrow$) of attribution-based explanations when we used the normalized and original attributions, repsectively.

Explanation method		TV news		census		bike	
		MLP-5	ResMLP-5	MLP-5	ResMLP-5	MLP-5	ResMLP-5
Input $\times$ Gradient	original	2.5373	2.2533	78.2222	7.5902	5.7450	6.9758
	normalized	2.2418	3.9514	26.6890	6.2027	4.5947	6.2172
LRP	original	1.3558	7.9203	49.9572	114.5870	5.7450	152.1785
	normalized	2.2627	1.5283	28.2636	5.2428	4.5947	5.9722
Occlusion	original	2.7904	1.9336	63.8212	9.1191	9.1114	11.3653
	normalized	1.8655	0.9323	26.2608	4.9668	5.4882	6.0682
Ours		$\mathbf{3.58 \times 10^{-7}}$	$\mathbf{1.93 \times 10^{-7}}$	$\mathbf{3.68 \times 10^{-6}}$	$\mathbf{1.16 \times 10^{-6}}$	$\mathbf{1.12 \times 10^{-6}}$	$\mathbf{1.54 \times 10^{-6}}$
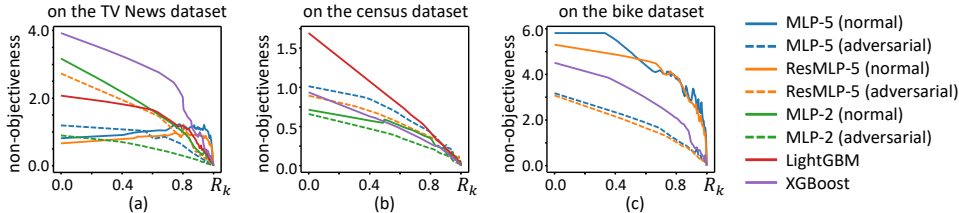
Figure 9: The relationship between R_k and the non-objectiveness $\rho^{\text{non-obj}}$ of the AOG explainer.

Then, we compared the non-objectiveness of the AOG explainer with the above seven baseline explanation methods. Based on each tabular dataset, we computed the average $\rho^{\text{non-obj}}$ over the training samples, *i.e.* $\mathbb{E}_x[\rho^{\text{non-obj}}]_{\text{given } x}$. Table 5 shows that the AOG explainer exhibited significantly stronger objectiveness than other explanation methods.

Besides, for fair comparison, we also normalized attribution values of Input $\times$ Gradient, LRP, and Occlusion, in order to force them to satisfy the efficiency property. *I.e.* $\tilde{\phi}_{\text{IG}}(i) = \frac{\phi_{\text{IG}}(i)}{\sum_{i \in N} \phi_{\text{IG}}(i)}$, $\tilde{\phi}_{\text{LRP}}(i) = \frac{\phi_{\text{LRP}}(i)}{\sum_{i \in N} \phi_{\text{LRP}}(i)}$, and $\tilde{\phi}_{\text{Occ}}(i) = \frac{\phi_{\text{Occ}}(i)}{\sum_{i \in N} \phi_{\text{Occ}}(i)}$. In this way, we have $v(N) = \sum_{i \in N} \tilde{\phi}_{\text{IG}}(i) = 1$, $v(N) = \sum_{i \in N} \tilde{\phi}_{\text{LRP}}(i) = 1$, and $v(N) = \sum_{i \in N} \tilde{\phi}_{\text{Occ}}(i) = 1$. Therefore, in this case $g(S)$ can be calculated as follows.

$$\tilde{g}_{\text{IG}}(S) = \sum_{i \in S} \tilde{\phi}_{\text{IG}}(i), \quad \tilde{g}_{\text{LRP}}(S) = \sum_{i \in S} \tilde{\phi}_{\text{LRP}}(i), \quad \tilde{g}_{\text{Occ}}(S) = \sum_{i \in S} \tilde{\phi}_{\text{Occ}}(i) \tag{13}$$

Table 6 shows that the normalized attributions usually exhibited stronger objectiveness, compared with their original attributions. Nevertheless, the AOG explainer still exhibited much stronger objectiveness compared with the normalized attributions.

F.2 THE RELATIONSHIP BETWEEN RATIO OF THE EXPLAINED UTILITIES AND THE OBJECTIVENESS

We further studied the relationship between the ratio of explained utilities R_k in Equation (4) and the non-objectiveness $\rho^{\text{non-obj}}$ of the explanation. We averaged the R_k-$\rho^{\text{non-obj}}$ curve over the training samples in each tabular dataset. Figure 9 shows that the objectiveness of the AOG explainer increased along with the increase of ratio of the explained utilities R_k.

G MORE EXPERIMENTAL RESULTS

G.1 MORE VISUALIZATION OF AOGS

This section provides the visualization of more AOGs generated by our method on various datasets.

For tabular datasets, Figure 12, Figure 13, Figure 14, Figure 15, and Figure 16 show examples of AOGs generated by our method on different models trained on the census, bike, and commercial dataset, respectively. The up-arrow($\uparrow$)/down-arrow($\downarrow$) labeled in terminal nodes indicated the actual value of the input variable was greater than / smaller than the baseline value.

For NLP datasets, Figure 17 and Figure 18 show examples of AOGs generated by our method on LSTMs and CNNs trained on the SST-2 dataset and the CoLA dataset. We discovered that in
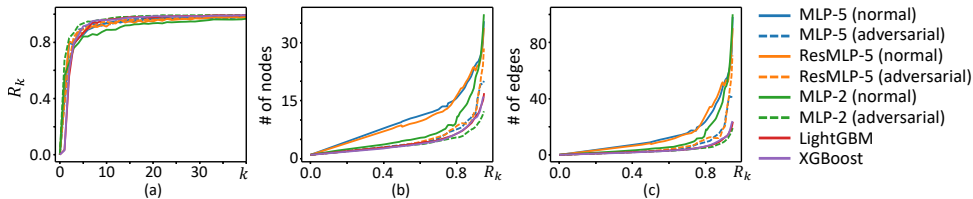
Figure 10: (a) The relationship between the number of salient patterns k in the AOG and the ratio of the explained utilities R_k, based on the census dataset. The relationship between R_k and (b) the number of nodes, and (c) the number of edges in the AOG, based on the census dataset.
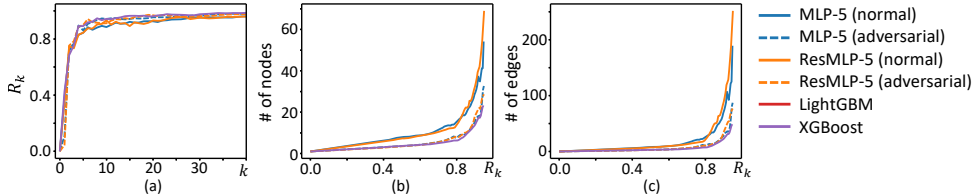

Figure 11: (a) The relationship between the number of salient patterns k in the AOG and the ratio of the explained utilities R_k, based on the bike dataset. The relationship between R_k and (b) the number of nodes, and (c) the number of edges in the AOG, based on the bike dataset.

LSTMs, single words were prone to be assigned a large utility to the model output. In comparison, CNNs were prone to encode salient patterns that consisted of multiple words.

G.2 MORE EXPERIMENTAL RESULTS ON THE RATIO OF THE EXPLAINED UTILITIES R_k

This section provides more experimental results on the relationship between the ratio of explained utilities R_k and the AOG explainer.

Just like the experiment in Paragraph *Ratio of the explained utilities*, Section 4, we used salient interaction patterns with top-k absolute values $|I(S)|$ to approximately explain the model output. Figure 10(a) and Figure 11(a) show the relationship between k and the ratio of explained utilities R_k in different models, based on the census dataset and the bike dataset. We found that when we used a few interaction patterns, we could explain most utilities of interaction patterns to the model output. Figure 10(b,c) and Figure 11(b,c) show that the node number and the edge number increased along with the increase of R_k.

Besides, Figure 10(a) and Figure 11 also show that compared with the normally trained model, we could use less salient patterns (smaller k) to achieve the same ratio of the explained utilities R_k in the adversarially trained model. Moreover, Figure 10(b,c) and 11(b,c) also show that AOGs corresponding to adversarially trained models were less complex than AOGs corresponding to normally trained models. This indicated that *adversarial training made models encode more sparse interaction patterns than normal training*.
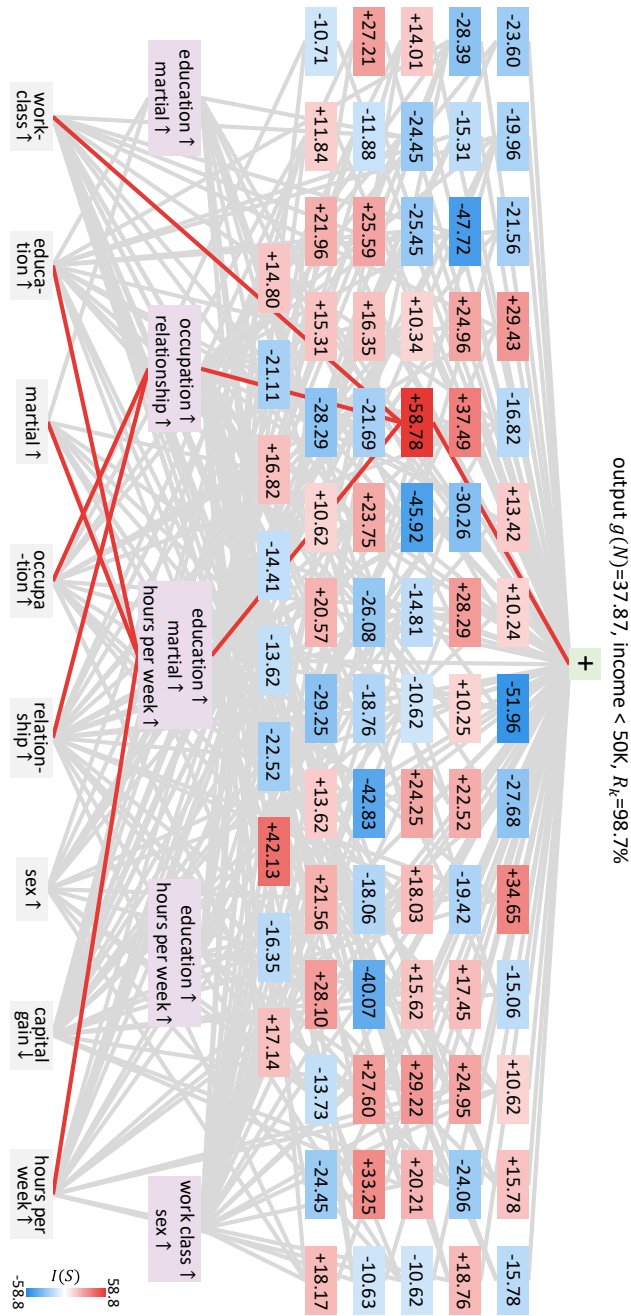
Figure 12: An example of the AOG extracted from the MLP-5 network, trained on the census dataset. Red edges indicate the parse graph of the most salient interaction pattern.
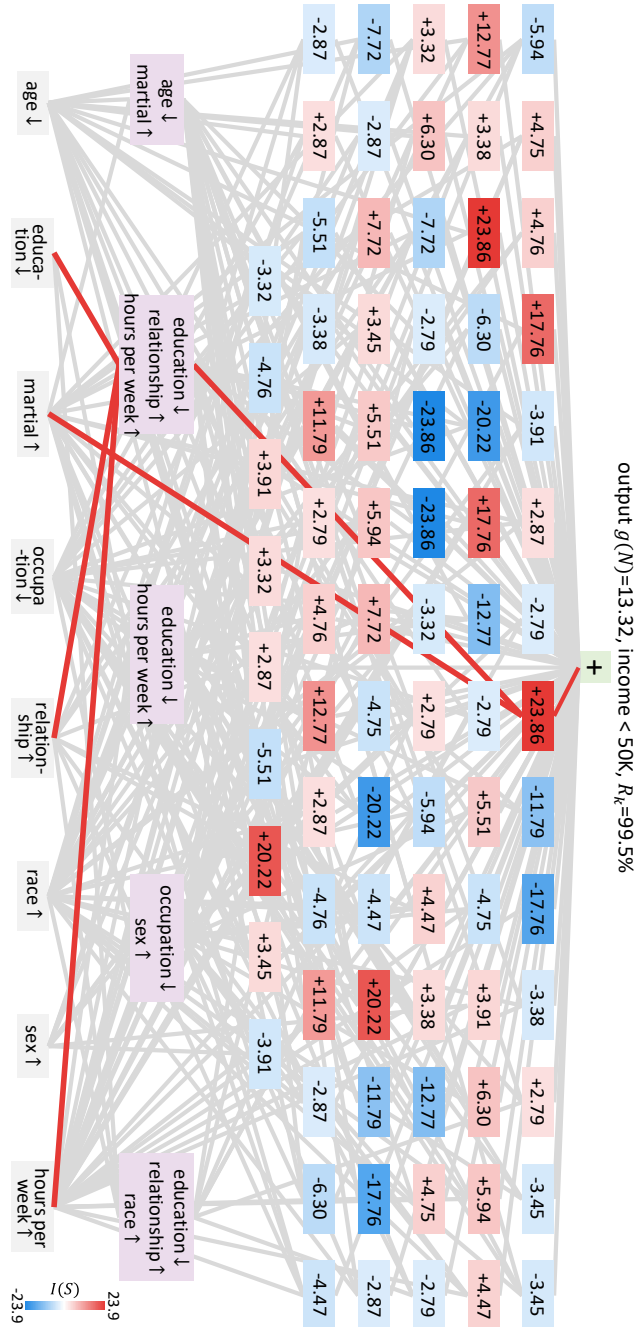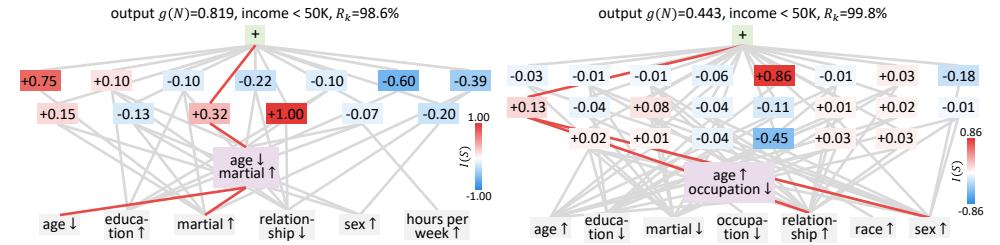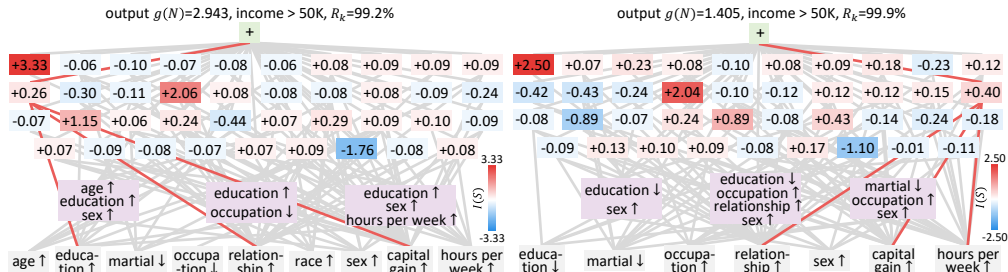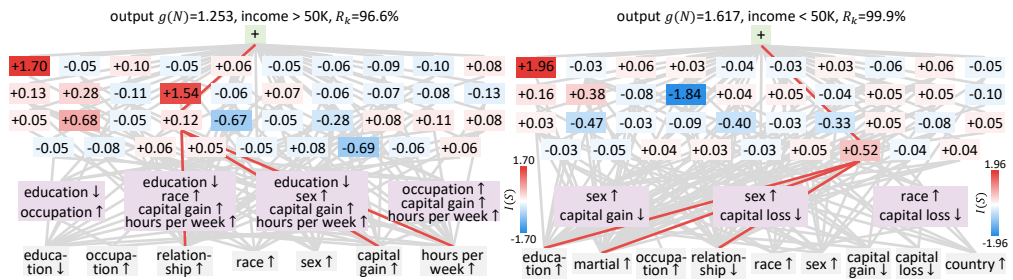
Figure 13: An example of the AOG extracted from the ResMLP-5 network, trained on the census dataset. Red edges indicate the parse graph of the most salient interaction pattern.

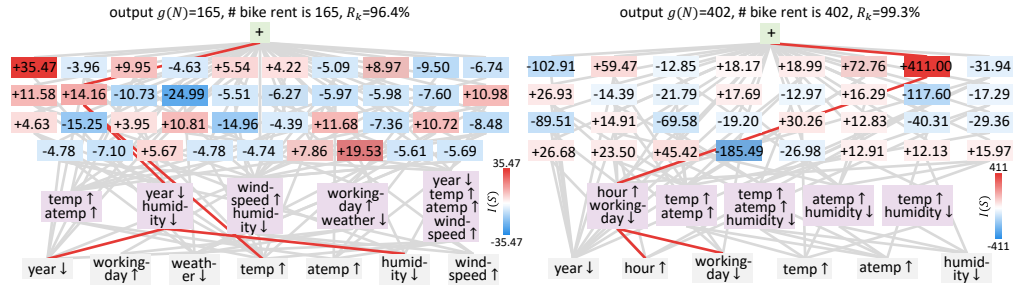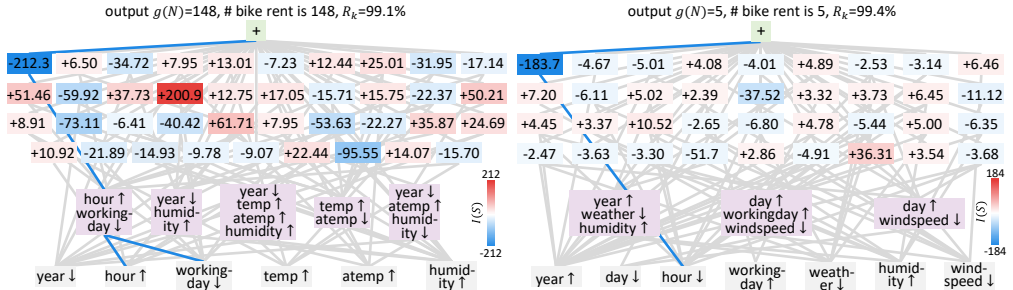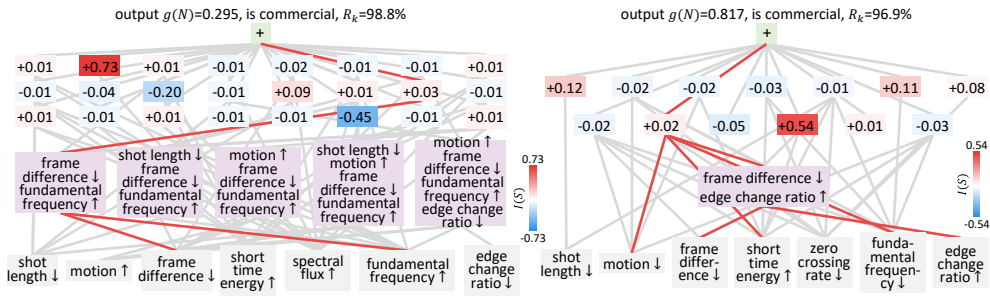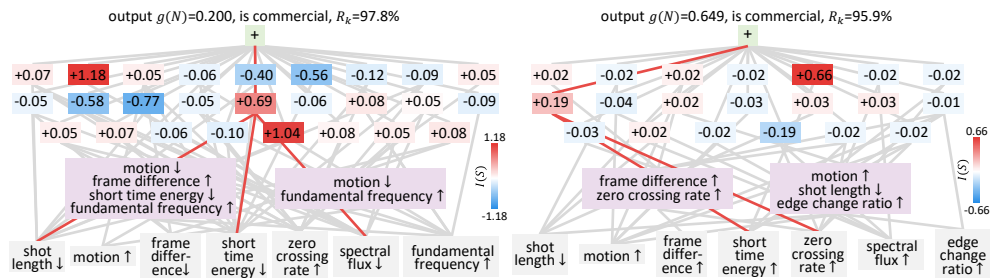(a) Examples of AOGs extracted from the MLP-2 network, adversarially trained on the census dataset.

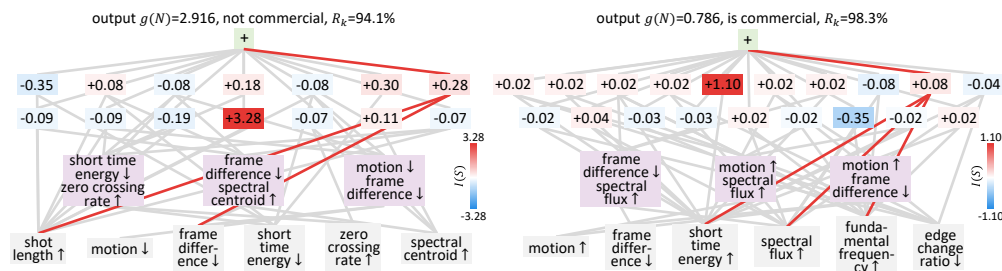(b) Examples of AOGs extracted from the MLP-5 network, adversarially trained on the census dataset.

(c) Examples of AOGs extracted from the ResMLP-5 network, adversarially trained on the census dataset.

Figure 14: Examples of AOGs extracted from models trained on the census dataset. Red edges indicate the parse graph of a specific interaction pattern.

(a) Examples of AOGs extracted from the MLP-5 network, adversarially trained on the bike dataset.

(b) Examples of AOGs extracted from the ResMLP-5 network, adversarially trained on the bike dataset.

Figure 15: Examples of AOGs extracted from models trained on the bike dataset. Red edges indicate the parse graph of a specific interaction pattern.

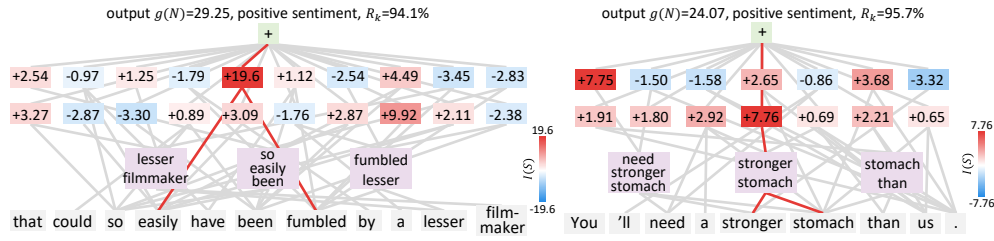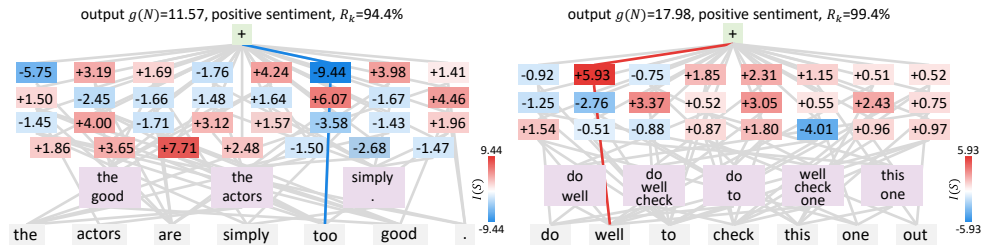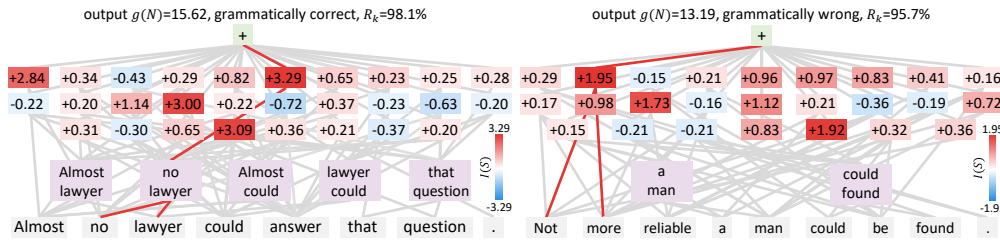(a) Examples of AOGs extracted from the MLP-2 network, adversarially trained on the commercial dataset.

(b) Examples of AOGs extracted from the MLP-5 network, adversarially trained on the commercial dataset.

(c) Examples of AOGs extracted from the ResMLP-5 network, adversarially trained on the commercial dataset.

Figure 16: Examples of AOGs extracted from models trained on the commercial dataset. Red edges indicate the parse graph of a specific interaction pattern.

(a) Examples of AOGs extracted from the CNN network, trained on the SST-2 dataset.
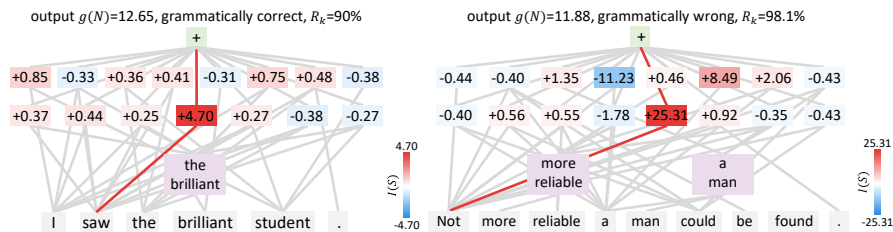

(b) Examples of AOGs extracted from the LSTM network, trained on the SST-2 dataset.

Figure 17: Examples of AOGs extracted from models trained on the SST-2 dataset. Red edges indicate the parse graph of the most salient interaction pattern.

(a) Examples of AOGs extracted from the CNN network, trained on the CoLA dataset.

(b) Examples of AOGs extracted from the LSTM network, trained on the CoLA dataset.

Figure 18: Examples of AOGs extracted from models trained on the CoLA dataset. Red edges indicate the parse graph of the most salient interaction pattern.
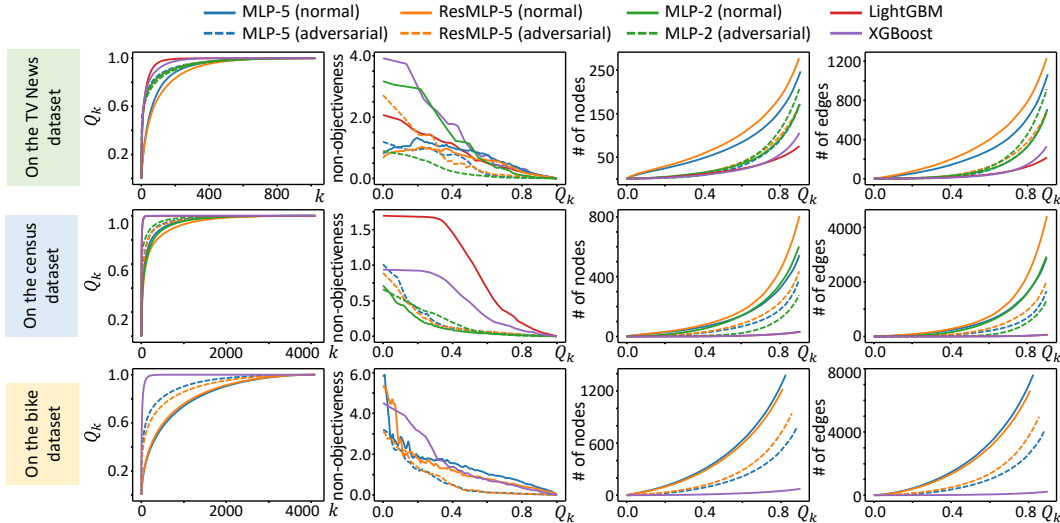
Figure 19: (1) The first column shows the relationship between the number of salient patterns k in the AOG and the ratio of the explained utilities Q_k, based on different datasets. (2) The second column shows the relationship between Q_k and the non-objectiveness $\rho^{\text{non-obj}}$ of the AOG explainer, based on different datasets. (3) The third column and the fourth column show the relationship between Q_k and the node number, and the edge number in the AOG, respectively.

H ANOTHER METRIC FOR THE RATIO OF THE EXPLAINED UTILITIES

In this section, we provide another metric for the ratio of the explained utilities. As is discussed in Section 3.3, we only used salient patterns with the top-k largest absolute values $|I(S)|$, denoted by $\Omega_{\text{top-}k}^{\text{salient}}$, to approximately explain the output of the deep model. The ratio of the explained interaction utilities could also be quantified as follows.

$$Q_k = \frac{\sum_{S \in \Omega_{\text{top-}k}^{\text{salient}}} |I(S)|}{\sum_{S \subseteq N} |I(S)|}, \tag{14}$$

where we used $\sum_{S \in \Omega_{\text{top-}k}^{\text{salient}}} |I(S)|$ to quantify the explained interaction utilities in the AOG explainer, while $\sum_{S \subseteq N} |I(S)|$ represented all interaction utilities encoded by the deep model.

We also studied the relationship between the number of salient patterns k and Q_k, the relationship between Q_k and the non-objectiveness of the AOG explainer, and the relationship between Q_k and the AOG complexity. Figure 19 show that the ratio of explained utilities Q_k increased along with the increase of the number of salient patterns k. Besides, the objectiveness of the AOG explainer was boosted along with the increase of Q_k. Moreover, the number of nodes and the number of edges in the AOG also increased along with the increase of Q_k.

I MORE ANALYSIS ON THE EFFECTIVENESS OF THE LEARNED BASELINE VALUES

This section provides more experimental analysis on the effects of baseline values on the conciseness of explanations. Beyond experiments in the Paragraph *Effects of baseline values on the conciseness of explanations*, Section 4, in this section, we analyzed the effectiveness of the learned baseline values in terms of the AOG complexity from different perspectives. To this end, we first computed interaction utilities using baseline values obtained in different epochs during the learning phase. Then, based on the computed interaction utilities, we measured the number of salient patterns, the node number, and the edge number in the AOG at each learning epoch. For fair comparison, we selected the minimum number k of salient patterns such that the ratio of the explained utilities Q_k exceeded 70%, *i.e.* $k = \arg\min_{k'}\{k' : Q_{k'} > 70\%\}$, to construct the AOG. Figure 20 shows the change of the AOG complexity during the learning process of baseline values, in terms of the number of salient patterns, the number of nodes, and the number of edges in the AOG. We found
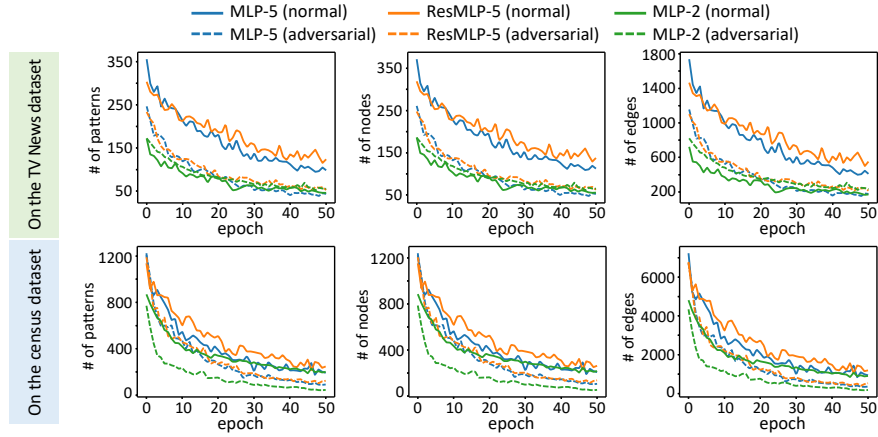
Figure 20: The number of patterns (the first column), nodes (the second column), and edges (the third column) in the AOG, based on baseline values of different learning epochs. The learned baseline value significantly enhanced the conciseness of explanations.

that the learning of baseline values significantly simplified the AOG, thus boosting the conciseness of explanations.