Explaining A Black-box By Using A Deep Variational Information Bottleneck Approach

Seojin Bang[1*], **Pengtao Xie**[2,3], **Heewook Lee**[4], **Wei Wu**[1], **Eric Xing**[1,2]
[1]Carnegie Mellon University, PA, USA
[2]Petuum Inc., PA, USA
[3]University of California San Diego, CA, USA
[4]Arizona State University, AZ, USA

Abstract

Interpretable machine learning has gained much attention recently. Briefness and comprehensiveness are necessary in order to provide a large amount of information concisely when explaining a black-box decision system. However, existing interpretable machine learning methods fail to consider briefness and comprehensiveness simultaneously, leading to redundant explanations. We propose the variational information bottleneck for interpretation, VIBI, a system-agnostic interpretable method that provides a brief but comprehensive explanation. VIBI adopts an information theoretic principle, *information bottleneck principle*, as a criterion for finding such explanations. For each instance, VIBI selects key features that are maximally compressed about an input (briefness), and informative about a decision made by a black-box system on that input (comprehensive). We evaluate VIBI on three datasets and compare with state-of-the-art interpretable machine learning methods in terms of both interpretability and fidelity evaluated by human and quantitative metrics.

Introduction

Interpretability is crucial in building and deploying black-box decision systems such as deep learning models. Interpretation of a black-box system helps decide whether or not to follow its decisions, or understand the logic behind the system. In recent years, the extensive use of deep learning black-box systems has given rise to interpretable machine learning approaches (Lipton 2016; Doshi-Velez and Kim 2017), which aim to explain how black-box systems work or why they reach certain decisions. In order to provide sufficient information while avoiding redundancy when explaining a black-box decision, we need to consider both *briefness* and *comprehensiveness*. However, existing approaches lack in-depth consideration for and fail to find both brief but comprehensive explanation.

In order to obtain brief but comprehensive explanation, we adopt the *information bottleneck principle* (Tishby, Pereira, and Bialek 2000). This principle provides an appealing information theoretic perspective for learning supervised models by defining what we mean by a 'good' representation. The principle says that the optimal model transmits as much information as possible from its input to its output through a compressed representation called the information bottleneck. Then, the information bottleneck will maximally compress the mutual information (MI) with an input while preserving as much as possible MI with the output. Recently, it has been shown that the principle also applies to deep neural networks and each layer of a deep neural network can work as an information bottleneck (Tishby and Zaslavsky 2015; Shwartz-Ziv and Tishby 2017). Using this idea of information bottleneck principle, we define a brief but comprehensive explanation as maximally informative about the black-box decision while compressive about a given input.

In this paper, we introduce the variational information bottleneck for interpretation (VIBI), a system-agnostic information bottleneck model that provides a brief but comprehensive explanation for every single decision made by a black-box model. VIBI is composed of two parts: explainer and approximator, each of which is modeled by a deep neural network. The explainer returns a probability whether a chunk of features such as a word, phrase, sentence or a group of pixels will be selected as an explanation or not for each instance, and an approximator mimics behaviour of a black-box model. Using the information bottleneck principle, we learn an explainer that favors brief explanations while enforcing that the explanations alone suffice for accurate approximations to a black-box model.

Our main contribution is to provide a convincing application of the information bottleneck principle that systematically defines and generates a 'good' explanation. Based on this principle, we develop VIBI that favors a brief but comprehensive explanation. In order to make the objective function of VIBI tractable, we derive a variational approximation to the objective. The beneficial characteristics of our method are as follows. 1) System-agnostic: VIBI is learned in a post-hoc manner. A model making black-box predictions, denoted by black-box (decision) system or black-box model, is learned or given as a separate process of learning VIBI. VIBI then learns the explanations by only using input and output of the black-box system. Because of this, VIBI can explain any types of black-box decision systems (i.e., agnostic to black-box systems that should be explained), hence there is no trade-off between task accuracy of a black-box system and interpretability of an explainer. 2) Cognitive chunk: Cognitive chunk is defined as a group

[*]seojinb@cs.cmu.edu; Part of this work was done while SB was at Petuum Inc.

of raw features whose identity is understandable to human. VIBI groups non-cognitive raw features such as a pixel and letter into a cognitive chunk (e.g. a group of pixels, a word, a phrase, a sentence) and selects each unit as an explanation. 3) Separate explainer and approximator: The explainer and approximator are designed for separated tasks so that we do not need to limit the approximator to have a simple structure, which may reduce the fidelity (the ability to imitate the behaviour of a black-box) of approximator.

Related Work

Existing methods are categorized into system-specific and system-agnostic methods. System-specific methods only explain certain black-box decision systems (e.g. using back-propagation algorithm), while system-agnostic methods explain any black-box decision systems.

System-specific methods. To measure change of output with respect to change of input is an intuitive way of obtaining feature attribution for the output. Using this idea, Zeiler and Fergus (2014), and Zintgraf et al. (2017) observe the change of output by making perturbation to each instance. Baehrens et al. (2010); Simonyan, Vedaldi, and Zisserman (2013), and Smilkov et al. (2017) use computationally more efficient approaches; they measure the change of output by propagating contribution of each feature through layers of a deep neural network towards an input. However, these approaches fail to detect the changes of output when the prediction function is flattened at the instance (Shrikumar, Greenside, and Kundaje 2017), which leads to interpretations focusing on irrelevant features. In order to solve this problem, the layer-wise relevance propagation (Bach et al. 2015; Binder et al. 2016), DeepLIFT (Shrikumar, Greenside, and Kundaje 2017), and Integrated Gradients (Sundararajan, Taly, and Yan 2017) compare the changes of output to its reference output. While these methods are used in a post-hoc manner, other methods such as (Yang et al. 2016; Mullenbach et al. 2018; Lei, Barzilay, and Jaakkola 2016) simultaneously learns the interpretation with a black-box model. However, such methods may result in accuracy loss due to the trade-off between the accuracy for the task and the interpretability to human.

System-agnostic methods. The great advantage of system-agnostic interpretable machine learning methods over system-specific methods is that their usage is not restricted to a specific black-box system. One of the most well-known system-agnostic methods is LIME (Ribeiro, Singh, and Guestrin 2016). It explains the decision of an instance by locally approximating the black-box decision boundary around the instance with an inherently interpretable model such as sparse linear or decision trees. The approximator is learned by samples generated by perturbing a given instance. Lundberg and Lee (2017) proposes SHAP, a unified measure defined over the additive feature attribution scores in order to achieve local accuracy, missingness, and consistency. As SHAP does, Dabkowski and Gal (2017); Fong and Vedaldi (2017), and Petsiuk, Das, and Saenko (2018) use sample perturbation but they rather learn or estimate desired perturbation masks than using perturbed samples to learn an approximator. Guidotti et al. (2019) generate exemplar images in the latent feature space and use the generated images as explanation. L2X (Chen et al. 2018) learns a stochastic map that selects instance-wise features that are most informative for black-box decisions. Unlike LIME and SHAP, which approximate local behaviors of a black-box system with a simple (linear) model, L2X does not put a limit on the structure of the approximator helping avoid losing fidelity of the approximator. Our method VIBI is system-agnostic. Our comparison experiments will be focused on comparison with existing system-agnostic methods.

Method

Perspective From Information Bottleneck Principle

The information bottleneck principle (Tishby, Pereira, and Bialek 2000) provides an appealing information theoretic view for learning a supervised model by defining what we mean by a 'good' representation. The principle says that the optimal model transmits as much information as possible from the input $\mathbf{x}$ to the output $\mathbf{y}$ through a compressed representation $\mathbf{t}$ (called the information bottleneck). The representation $\mathbf{t}$ is stochastically defined and the optimal stochastic mapping $p(\mathbf{t}|\mathbf{x})$ is obtained by optimizing the following problem with a Markov chain assumption $\mathbf{y} \to \mathbf{x} \to \mathbf{t}$:

$$p(\mathbf{t}|\mathbf{x}) = \underset{p(\mathbf{t}|\mathbf{x}),p(\mathbf{y}|\mathbf{t}),p(\mathbf{t})}{\arg\max} \quad \mathrm{I}(\mathbf{t},\mathbf{y}) - \beta\, \mathrm{I}(\mathbf{x},\mathbf{t}) \qquad (1)$$

where $\mathrm{I}(\cdot,\cdot)$ is the MI and β is a Lagrange multiplier representing the trade-off between the compressiveness $-\mathrm{I}(\mathbf{x},\mathbf{t})$ and informativeness $\mathrm{I}(\mathbf{t},\mathbf{y})$ of the representation $\mathbf{t}$.

We adopt the information bottleneck principle as a criterion for finding brief but comprehensive explanations. Our aim is to learn an explainer generating explanations that are maximally informative about the black-box decision while compressive about a given input.

Proposed Approach

We introduce VIBI, a system-agnostic interpretation approach that provides brief but comprehensive explanations for decisions made by black-box decision system. In order to achieve this, we optimize the following information bottleneck objective.

$$p(\mathbf{z}|\mathbf{x}) = \underset{p(\mathbf{z}|\mathbf{x}),p(\mathbf{y}|\mathbf{t})}{\arg\max} \quad \mathrm{I}(\mathbf{t},\mathbf{y}) - \beta\, \mathrm{I}(\mathbf{x},\mathbf{t}) \qquad (2)$$

where $\mathrm{I}(\mathbf{t},\mathbf{y})$ represents the sufficiency of information retained for explaining the black-box output $\mathbf{y}$, $-\mathrm{I}(\mathbf{x},\mathbf{t})$ represents the briefness of the explanation $\mathbf{t}$, and β is a Lagrange multiplier representing a trade-off between the two. The primary difference between our information bottleneck objective (2) and the one in (Tishby, Pereira, and Bialek 2000) is as follows: the latter aims to identify a stochastic map of the representation $\mathbf{t}$ that itself works as an information bottleneck, whereas our objective aims to identify a stochastic map of $\mathbf{z}$ performing instance-wise selection of cognitive chunks and define information bottleneck as a function of $\mathbf{z}$ and the input $\mathbf{x}$.

As illustrated in Figure 1A, VIBI is composed of two parts: the explainer and the approximator, each of which is

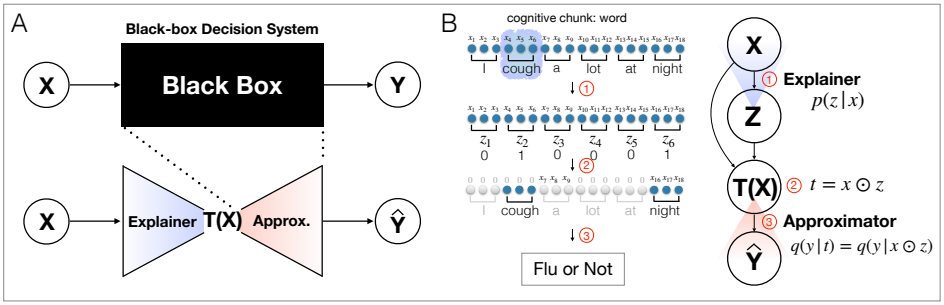

Figure 1: Illustration of VIBI. (A) VIBI is composed of two parts: explainer and approximator. The explainer selects a group of k key cognitive chunks given an instance while the approximator mimics the behaviour of a black-box system using the selected keys as inputs. (B) We set each word as a cognitive chunk and $k = 2$. ① The explainer takes an input $\mathbf{x}$ and returns a stochastic k-hot random vector $\mathbf{z}$ which indicates whether each cognitive chunk will be selected as an explanation or not. ② $\mathbf{t}(\mathbf{x})$ provides instance-specific explanation. ③ The approximator takes $\mathbf{t}(\mathbf{x})$ as an input and approximates the black-box output.

modeled by a deep neural network. The explainer selects a group of k key cognitive chunks given an instance while the approximator mimics the behaviour of the black-box system using the selected keys as the input. k controls the level of sparsity in $\mathbf{z}$. In detail, the explainer $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_e)$ is a map from an input $\mathbf{x}$ to its attribution scores $p_j(\mathbf{x}) = p(\mathrm{z}_j|\mathbf{x})$ where j is for the j-th cognitive chunk and z_j is a binary indicator whether the chunk will be selected or not. The attribution score indicates the probability that each cognitive chunk to be selected. In order to select top k cognitive chunks as an explanation, a k-hot vector $\mathbf{z}$ is sampled from a categorical distribution with class probabilities $p_j(\mathbf{x}) = p(\mathrm{z}_j|\mathbf{x})$ and the j-th cognitive chunk is selected if $\mathrm{z}_j = 1$. More specifically, the explanation $\mathbf{t}$ is defined as follows:

$$\mathrm{t}_i = (\mathbf{x} \odot \mathbf{z})_i = \mathrm{x}_i \times \mathrm{z}_j,$$

where j indicates a cognitive chunk, each of which corresponds to multiple row features i. The approximator is modeled by another deep neural network $p(\mathbf{y}|\mathbf{t};\boldsymbol{\theta}_a)$, which mimics the black-box decision system. It takes $\mathbf{t}$ as an input and returns an output approximating the black-box output for the instance $\mathbf{x}$. $\boldsymbol{\theta}_a$ and $\boldsymbol{\theta}_e$ represent the weight parameters of neural networks. The explainer and approximator are trained jointly by minimizing a cost function that favors concise explanations while enforcing that the explanations alone suffice for accurate prediction.

To achieve compressiveness, in addition to encouraging small MI between explanations and inputs, we also encourage the number of selected cognitive chunks to be small, i.e., encouraging $\mathbf{z}$ to be sparse. Note that MI and sparsity are two complementary approaches for achieving compression. MI aims at reducing semantic redundancy in explanations. Sparsity cannot achieve such a goal. For example, consider a movie review where "great" occurs a lot and two explanations in judging the sentiment of the review: "great, great" and "great, thought-provoking". They have the same level of sparsity ($k = 2$), but the former has semantic redundancy. In this case, MI helps to choose a better explanation. The first explanation has a larger MI with the input document. The second explanation has smaller MI and hence is more brief and preferable.

The variational bound. The current form of information bottleneck objective is intractable due to the MIs $\mathrm{I}(\mathbf{t},\mathbf{y})$ and $\mathrm{I}(\mathbf{x},\mathbf{t})$. We address this problem by using a variational approximation of our information bottleneck objective. In this section, we summarize the results and refer to Supplementary Materials S1 for details.

Variational bound for $\mathrm{I}(\mathbf{x},\mathbf{t})$: We first show that $\mathrm{I}(\mathbf{x},\mathbf{t}) \leq \mathrm{I}(\mathbf{x},\mathbf{z}) + C$ where C is constant and use the lower bound for $-\mathrm{I}(\mathbf{x},\mathbf{z}) - C$ as a lower bound for $-\mathrm{I}(\mathbf{x},\mathbf{t})$. As a result, we obtain:

$$\begin{aligned}\mathrm{I}(\mathbf{x},\mathbf{t}) \leq \mathrm{I}(\mathbf{x},\mathbf{z}) + C &\leq \mathbb{E}_{(\mathbf{x},\mathbf{z})\sim p(\mathbf{x},\mathbf{z})}\left[\log \frac{p(\mathbf{z}|\mathbf{x})}{r(\mathbf{z})}\right] + C \\ &= \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})} D_{\mathrm{KL}}(p(\mathbf{z}|\mathbf{x}), r(\mathbf{z})) + C \end{aligned} \quad (3)$$

Note that with proper choices of $r(\mathbf{z})$ and $p(\mathbf{z}|\mathbf{x})$, we can assume that the Kullback-Leibler divergence $D_{\mathrm{KL}}(p(\mathbf{z}|\mathbf{x}), r(\mathbf{z}))$ has an analytical form.

Variational bound for $\mathrm{I}(\mathbf{t},\mathbf{y})$: We obtain the lower bound for $\mathrm{I}(\mathbf{t},\mathbf{y})$ by using $q(\mathbf{y}|\mathbf{t})$ to approximate $p(\mathbf{y}|\mathbf{t})$, which works as an approximator to the black-box system. As a result, we obtain:

$$\begin{aligned}\mathrm{I}(\mathbf{t},\mathbf{y}) &\geq \mathbb{E}_{(\mathbf{t},\mathbf{y})\sim p(\mathbf{t},\mathbf{y})}\left[\log q(\mathbf{y}|\mathbf{t})\right] \\ &= \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\mathbb{E}_{\mathbf{y}|\mathbf{x}\sim p(\mathbf{y}|\mathbf{x})}\mathbb{E}_{\mathbf{t}|\mathbf{x}\sim p(\mathbf{t}|\mathbf{x})}\left[\log q(\mathbf{y}|\mathbf{t})\right] \end{aligned} \quad (4)$$

where $p(\mathbf{t}|\mathbf{x},\mathbf{y}) = p(\mathbf{t}|\mathbf{x})$ by the Markov chain assumption $\mathbf{y} \leftrightarrow \mathbf{x} \leftrightarrow \mathbf{t}$. Combining Equations (3) and (4), we obtain the following variational bound:

$$\begin{aligned}&\mathrm{I}(\mathbf{t},\mathbf{y}) - \beta\, \mathrm{I}(\mathbf{x},\mathbf{t}) \\ &\quad \geq \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\mathbb{E}_{\mathbf{y}|\mathbf{x}\sim p(\mathbf{y}|\mathbf{x})}\mathbb{E}_{\mathbf{t}|\mathbf{x}\sim p(\mathbf{t}|\mathbf{x})}\left[\log q(\mathbf{y}|\mathbf{t})\right] \\ &\quad\quad - \beta\, \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})} D_{\mathrm{KL}}(p(\mathbf{z}|\mathbf{x}), r(\mathbf{z})) + C^*. \end{aligned} \quad (5)$$

where $C^* = -C\beta$ can be ignored since it is independent of the optimization procedure. We use the empirical data distribution to approximate $p(\mathbf{x},\mathbf{y}) = p(\mathbf{x})p(\mathbf{y}|\mathbf{x})$ and $p(\mathbf{x})$.

Continuous relaxation and reparameterization. Current form of the bound (5) is still intractable because we need to sum over the $\binom{d}{k}$ combinations of feature subsets. This is because we sample top k out of d cognitive chunks

where each chunk is assumed drawn from a categorical distribution with class probabilities $p_j(\mathbf{x}) = p(\mathrm{z}_j|\mathbf{x})$. In order to avoid this, we use the generalized Gumbel-softmax trick (Jang, Gu, and Poole 2017; Chen et al. 2018). This is a well-known technique that is used to approximate a non-differentiable categorical subset sampling with differentiable Gumbel-softmax samples. The steps are as follows.

First, we independently sample a cognitive chunk for k times. For each time, a random perturbation e_j is added to the log probability of each cognitive chunk $\log p_j(\mathbf{x})$. From this, Concrete random vector $\mathbf{c} = (\mathrm{c}_1, \cdots, \mathrm{c}_d)$ working as a continuous, differentiable approximation to argmax is defined:

$$\mathrm{g}_j = -\log(-\log \mathrm{e}_j) \quad \text{where } \mathrm{e}_j \sim U(0,1)$$
$$\mathrm{c}_j = \frac{\exp\left(\left(\mathrm{g}_j + \log p_j(\mathbf{x})\right)/\tau\right)}{\sum_{j=1}^{d} \exp\left(\left(\mathrm{g}_j + \log p_j(\mathbf{x})\right)/\tau\right)},$$

where τ is a tuning parameter for the temperature of Gumbel-Softmax distribution. Next, we define a continuous-relaxed random vector $\mathbf{z}^* = [\mathrm{z}_1^*, \cdots, \mathrm{z}_d^*]^\top$ as the element-wise maximum of the independently sampled Concrete vectors $\mathbf{c}^{(l)}$ where $l = 1, \cdots, k$:

$$\mathrm{z}_j^* = \max_l \mathrm{c}_j^{(l)} \text{ for } l = 1, \cdots, k$$

With this sampling scheme, we approximate the k-hot random vector and have the continuous approximation to the variational bound (5). This trick allows using standard back-propagation to compute the gradients of the parameters via reparameterization. By putting everything together, we obtain:

$$\frac{1}{NL}\sum_{n}^{N}\sum_{l}^{L}\Big[\log q(\boldsymbol{y}_{(n)}|\mathbf{x}_{(n)} \odot f(\mathbf{e}_{(n)}^{(l)}, \mathbf{x}_{(n)})) - \beta\, D_{\mathrm{KL}}(p(\mathbf{z}_{(n)}^*|\mathbf{x}_{(n)}), r(\mathbf{z}_{(n)}^*))\Big]$$

where N is the number of samples, n indicate the n-th sample, $f(\mathbf{e}_{(n)}^{(l)}, \mathbf{x}_{(n)}) = \mathbf{z}_{(n)}^*$, $q(\boldsymbol{y}_{(n)}|\mathbf{x}_{(n)} \odot \mathbf{z}_{(n)}^*)$ is the approximator to the black-box system and $-D_{\mathrm{KL}}(p(\mathbf{z}^*|\mathbf{x}_{(n)}), r(\mathbf{z}^*))$ represents the compactness of the explanation. Once we learn the model, the attribution score $p_j(\mathbf{x})$ for each cognitive chunk is used to select top k key cognitive chunks that are maximally compressive about the input $\mathbf{x}$ and informative about the black-box decision $\mathbf{y}$ on that input.

Experiments

We evaluated VIBI on three datasets and compared with state-of-the-art interpretable machine learning methods. The evaluation is performed from two perspectives: *interpretability* and *fidelity*. The interpretability indicates the ability to explain a black-box model with human understandable terms. The fidelity implies how accurately our approximator approximates the black-box model. Based on these criteria, we compared VIBI with three state-of-the-art system-agnostic methods (LIME (Ribeiro, Singh, and Guestrin 2016), SHAP (Lundberg and Lee 2017) and L2X (Chen et al. 2018)), and a commonly used model-specific method called Saliency Map (Simonyan, Vedaldi, and Zisserman 2013). For fair comparison, we compare with methods that learned in a post-hoc manner, so that we use the same black-box systems that should be explained. For Saliency Map, we used the smooth gradient technique (Smilkov et al. 2017) to get visually sharp gradient-based sensitivity maps over the basic gradient saliency map. See Supplementary Material S2 for further experimental details.

We examined how VIBI performs across different experimental settings varying the number of selected chunks k (amount or number of explanation), size of chunk (unit of explanation), and trade-off parameter β (trade-off between the compressiveness of explanation and information preserved about the output). The settings of hyper-parameter tuning include (bold indicate the choice for our final model): the temperature for Gumbel-softmax approximation τ – $\{0.1, 0.2, 0.5, \mathbf{0.7}, 1\}$, learning rate – $5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}, \mathbf{10^{-4}}, 5 \times 10^{-5}\}$ and β – $\{0, 0.001, 0.01, \mathbf{0.1}, 1, 10, 100\}$. We use Adam algorithm (Kingma and Ba 2014) with batch size 100 for MNIST and 50 for IMDB, the coefficients used for computing running averages of gradient and its square $(\beta_1, \beta_2) = (0.5, 0.999)$, and $\epsilon = 10^{-8}$. We tuned the hyperparameters via grid search and picked up the hyperparameters that yield the best fidelity score on the validation set. The code is publicly available on GitHub: github.com/SeojinBang/VIBI.

LSTM Movie Sentiment Prediction Model Using IMDB

The IMDB (Maas et al. 2011) is a large text dataset containing movie reviews labeled by sentiment (positive/negative). We grouped the reviews into training, validation, and test sets, which have 25,000, 12,500, and 12,500 reviews respectively. Then, we trained a hierarchical LSTM for sentiment prediction, which has two LSTM layers where each layer encodes words and sentences respectively. It achieved 87% of test accuracy. In order to explain this LSTM black-box model, we applied VIBI. We parameterized the explainer using a bidirectional LSTM and approximator using a 2D CNN. For the details of the black-box model and VIBI architectures, see Supplementary Material S2.1.

VIBI explains why the LSTM predicts each movie review to be positive/negative and provides instance-wise key words that are the most important attributes to the sentiment prediction. As seen in the top-right and top-left of Figure 2, VIBI shows that the positive (or negative) words pass through the bottleneck and make a correct prediction. The bottom of Figure 2 shows that the LSTM sentiment prediction model makes a wrong prediction for a negative review because the review includes several positive words such as 'enjoyable' and 'exciting'.

CNN Digit Recognition Model Using MNIST

The MNIST (LeCun et al. 1998) is a large dataset contains 28×28 sized images of handwritten digits (0 to 9).

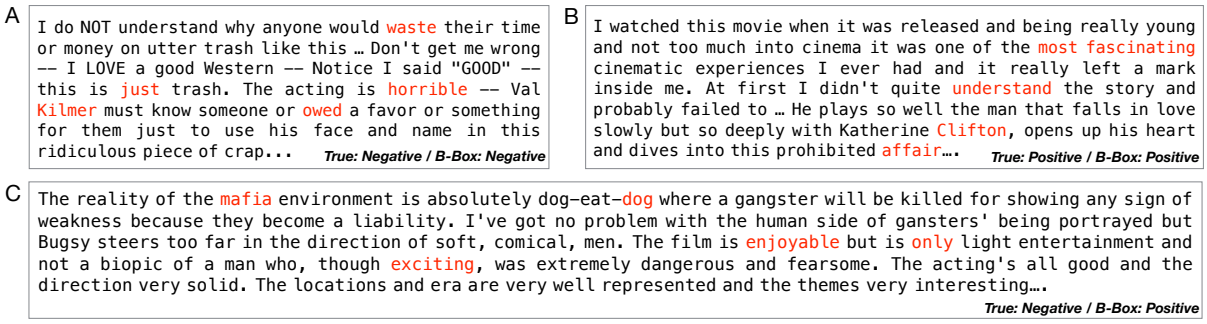

Figure 2: The movie reviews and explanations provided by VIBI were randomly selected from the validation set. The selected words are colored red. Each word is used as a cognitive chunk and $k = 5$ words are provided for each review. (A) VIBI explains why the negative review is correctly predicted as negative by highlighting the negative words (*waste*, *horrible*) and an adjective describing a negative noun (*just*). (B) VIBI explains why the positive review is correctly predicted as positive by highlighting the positive words (*most fascinating*). (C) VIBI explains why the negative review is incorrectly predicted as positive by highlighting positive words (*enjoyable*, *exciting*).

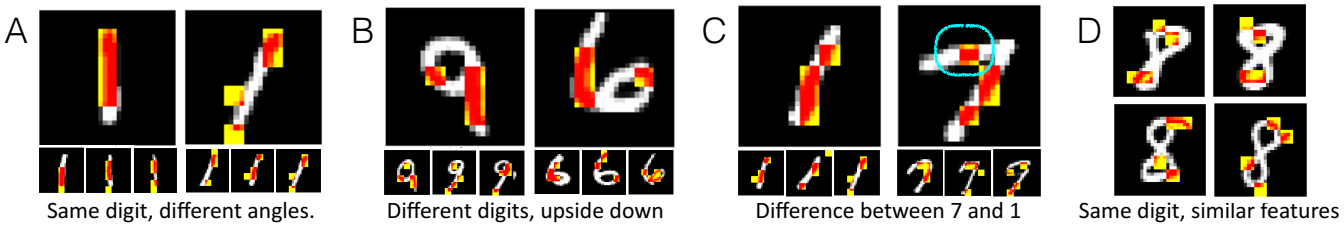

Figure 3: The hand-written digits and explanations provided by VIBI were randomly selected from the validation set. The selected patches are colored red if the pixel is activated (i.e. white) and yellow otherwise (i.e. black). A patch composed of 4×4 pixels is used as a cognitive chunk and $k = 4$ patches are identified for each image.

We grouped the images into training, validation, and test sets, which have 50,000, 10,000, and 10,000 images respectively, and trained a simple 2D CNN for the digit recognition, which achieved 97% of test accuracy. In order to explain this CNN black-box model, we applied VIBI. We parameterized each the explainer and approximator using a 2D CNN. For the details of the black-box model and VIBI architectures, see Supplementary Material S2.2.

VIBI explains how the CNN characterizes a digit and recognizes differences between digits. The first two examples in Figure 3 show that the CNN recognizes digits using both shapes and angles. In the first example, the CNN characterizes '1's by straightly aligned patches along with the activated regions although '1's in the left and right panels are written at different angles. Contrary to the first example, the second example shows that the CNN recognizes the difference between '9' and '6' by their differences in angles. The last two examples in Figure 3 show that the CNN catches a difference of '7's from '1's by patches located on the activated horizontal line on '7' (see the cyan circle) and recognizes '8's by two patches on the top of the digits and another two patches at the bottom circle. More qualitative examples for VIBI and the baselines are shown in Figure S2.

The briefness of explanations also depends on the sparsity k. Figure S4 shows how our method works under different sparsity. When we increase k, VIBI tends to select patches that are the same with or nearby previously selected patches and additionally select patches that catch new characteristics of digits.

TCR To Epitope Binding Prediction Model Using VDJdb And IEDB

We next illustrate how VIBI can be used to get insights from a model and ensure the safety of a model in a real world application. Identifying which T-cell receptor (TCR) will bind to a specific epitope (i.e. cancer induced peptide molecules presented by the major histocompatibility complex to T-cells) is important for screening T-cells or genetically engineering T-cells that are effective in recognizing and destroying tumor cells. Therefore, there has been efforts in developing computational methods to predict binding affinity of given TCR-epitope pairs (Jurtz et al. 2018; Jokinen et al. 2019). These approaches rely on known interacting TCR-epitope pairs available from VDJdb (Shugay et al. 2017) and IEDB (Vita et al. 2014), which are the largest databases of several thousand entries. However, the number of unique TCRs harbored in a single individual is estimated to be 10^{10} (Lythe et al. 2016) and a theoretical number of epitopes of length l is 20^l, which are much larger than the number of known interacting TCR-epitope pairs.

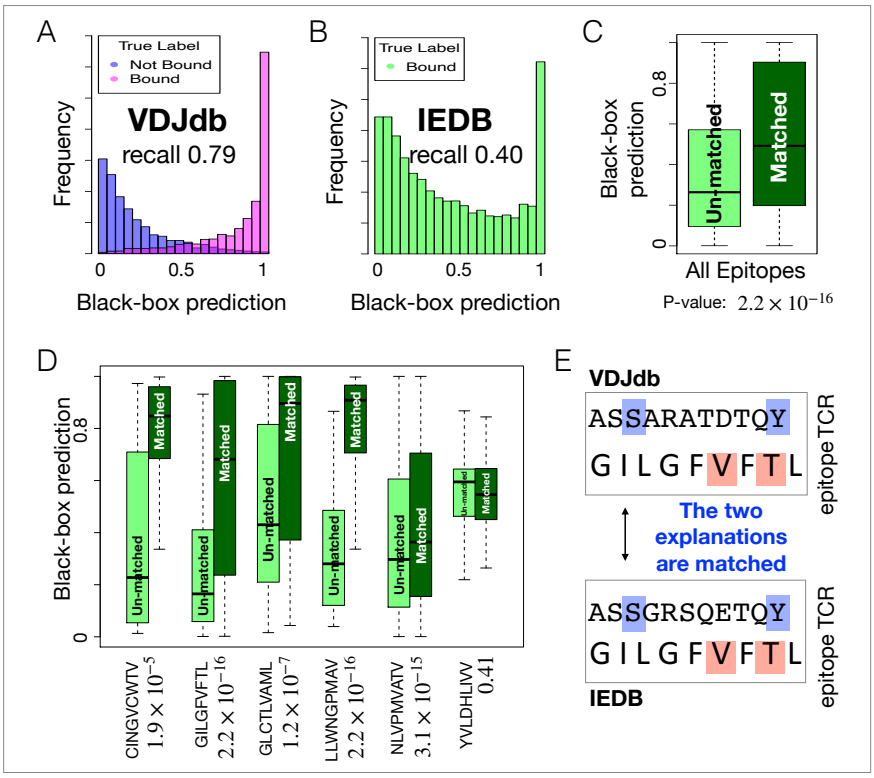

Figure 4: Black-box prediction scores of (A) VDJdb and (B) IEDB. (C) Black-box prediction scores between the matched and unmatched instances from IEDB and (D) those by six epitope sequences. (E) An example of matched explanation (The VIBI selected amino acids are shaded).

One of the main concerns is whether a black-box model trained on such limited dataset can accurately predict TCR-epitope bindings of out-of-samples. This concern becomes pressing in a TCR-epitope binding prediction model trained on VDJdb (For details of the data, black-box model architecture, and parameter tuning, see Supplementary Material S2.3). The model accurately predicted the (in-sample) bindings from VDJdb (recall 0.79, Figure 4A). However, it achieved poor prediction performance when it is used to predict the (out-of-sample) bindings from another dataset, IEDB (recall 0.40, Figure 4B). In an attempt to address this problem, we applied VIBI and determined whether or not to accept a decision made by the black-box model based on VIBI's explanation. As illustrated in Figure 4E, VIBI provided *matched explanations*—the identical amino acids in same positions (`S` and `Y` in this example) are highlighted in different TCR sequences when they are bound to the same epitope (`GILGFVFTL` in this example). Moreover, we found that if two TCR sequences binding to the same epitope, each from IEDB and VDJdb, are assigned with matched explanations by VIBI, then it significantly better predicts the binding than the others with no matching TCRs (Figure 4C-D, p-values are shown). Therefore, if a TCR sequence from IEDB has a matched explanation to a TCR from VDJdb, then we safely follow the positive decision made by the black-box model.

Fidelity

We assessed fidelity of the methods in approximating the black-box output. First, we compared the ability of the approximators to imitate behaviour of the black-box, denoted as *Approximator fidelity*. (See Text S2.4 for details about how each approximator fidelity is evaluated.) As shown in Table 1, VIBI has a better approximator fidelity than Saliency, LIME and SHAP in most cases. VIBI and L2X showed similar levels of approximator fidelity, so we further compared them based on *Rationale fidelity*. The difference between approximator and rationale fidelity is as follows. Approximator fidelity is quantified by prediction performance of the approximators that takes $\mathbf{t}^*$, the continuous relaxation of $\mathbf{t}$, as an input and the black-box output as a targeted label; rationale fidelity is quantified by using $\mathbf{t}$ instead of $\mathbf{t}^*$. Note that $\mathbf{t}$ only takes the top k chunks and sets the others to be zero, while $\mathbf{t}^*$ sets the others to be small, non-zero values. Therefore, rationale fidelity allows to evaluate how much information purely flows through the explanations, not through a narrow crack made during the continuous relaxation procedure. As shown in Table 1, VIBI has a better rationale fidelity than L2X in most cases. Note that L2X can be viewed as a special case of VIBI without the compressiveness term, i.e., $\beta = 0$. The rationale fidelity empirically demonstrates that the compressiveness term can help the information to flow purely through the explanations.

			Approximator Fidelity					Rationale Fidelity	
	chunk size	k	Saliency	LIME	SHAP	L2X	VIBI (Ours)	L2X	VIBI (Ours)
IMDB	sentence	1	38.7 ± 0.9	72.7 ± 0.8	49.5 ± 1.0	$\mathbf{87.6 \pm 0.6}$	$\mathbf{87.7 \pm 0.6}$	$\mathbf{72.7 \pm 0.8}$	$\mathbf{73.1 \pm 0.8}$
	word	5	41.9 ± 0.9	$\mathbf{75.6 \pm 0.8}$	50.1 ± 1.0	73.8 ± 0.8	$\mathbf{74.4 \pm 0.8}$	63.8 ± 0.8	$\mathbf{65.7 \pm 0.8}$
	5 words	1	42.4 ± 0.9	29.0 ± 0.8	49.7 ± 1.0	$\mathbf{75.9 \pm 0.7}$	$\mathbf{76.4 \pm 0.7}$	60.1 ± 0.9	$\mathbf{63.2 \pm 0.8}$
	5 words	3	41.4 ± 0.9	67.9 ± 0.8	49.1 ± 1.0	$\mathbf{83.3 \pm 0.7}$	$\mathbf{83.5 \pm 0.7}$	$\mathbf{69.4 \pm 0.8}$	66.0 ± 0.8
MNIST	2×2	16	91.2 ± 0.6	77.0 ± 0.8	$\mathbf{94.2 \pm 0.5}$	93.4 ± 0.5	$\mathbf{94.8 \pm 0.4}$	73.5 ± 0.9	$\mathbf{77.1 \pm 0.8}$
	2×2	24	93.8 ± 0.5	80.7 ± 0.8	$\mathbf{95.4 \pm 0.4}$	$\mathbf{95.1 \pm 0.4}$	$\mathbf{95.3 \pm 0.4}$	77.6 ± 0.8	$\mathbf{85.6 \pm 0.7}$
	2×2	40	95.7 ± 0.4	85.9 ± 0.7	95.4 ± 0.4	$\mathbf{96.7 \pm 0.4}$	$\mathbf{96.2 \pm 0.4}$	81.1 ± 0.8	$\mathbf{91.5 \pm 0.5}$
	4×4	4	86.3 ± 0.7	60.9 ± 1.0	$\mathbf{94.8 \pm 0.4}$	$\mathbf{95.3 \pm 0.4}$	$\mathbf{94.8 \pm 0.4}$	65.0 ± 0.9	$\mathbf{77.5 \pm 0.8}$
	4×4	6	90.6 ± 0.6	63.7 ± 0.9	93.6 ± 0.5	$\mathbf{95.7 \pm 0.4}$	$\mathbf{95.6 \pm 0.4}$	51.1 ± 1.0	$\mathbf{70.1 \pm 0.9}$
	4×4	10	94.9 ± 0.4	70.5 ± 0.9	95.1 ± 0.4	$\mathbf{96.5 \pm 0.4}$	$\mathbf{96.7 \pm 0.4}$	83.5 ± 0.7	$\mathbf{93.3 \pm 0.5}$

Table 1: Evaluation of approximator and rationale fidelity. $\beta = 0.1$ for VIBI. Accuracy and 0.95 confidence interval is shown. We performed three runs for each method and reported the best results. See more evaluations using F1-score and further results from different parameter settings in Table S4 and S6 for approximator fidelity and Table S3 and S5 for rationale fidelity.

	Saliency	LIME	L2X	VIBI (Ours)
IMDB	34.2%	33.8%	35.6%	**44.7%**
MNIST	3.448	1.369	1.936	**3.526**

Table 2: Evaluation of interpretability. For IMDB, the percentage indicates how well the MTurk worker's answers match the black-box output. For MNIST, the score indicates how well the highlighted chunks catch key characteristics of handwritten digits (0 to 5). The average scores over all samples is shown. See the survey example and detailed results in Supplementary Material S3.

Interpretability Evaluated By Humans

We evaluated interpretability of the methods on the LSTM movie sentiment prediction model and the CNN digit recognition model. For the movie sentiment prediction model, we provided instances that the black-box model had correctly predicted and asked humans to infer the output of the primary sentiment of the movie review (Positive/Negative/Neutral) given five key words selected by each method. Each method was evaluated by the humans on Amazon Mechanical Turk (MTurk, https://www.mturk.com/) who are awarded the Masters Qualification, high-performance workers who have demonstrated excellence across a wide range of tasks. We randomly selected and evaluated 200 instances for VIBI and 100 instances for the others. Five workers were assigned per instance. For the digit recognition model, we asked humans to directly score the explanation on a 0–5 scale. Each method was evaluated by 16 graduate students at Carnegie Mellon University who have taken at least one graduate-level machine learning class. For each method, 100 instances were randomly selected and evaluated. Four cognitive chunks with the size 4×4 were provided as an explanation for each instance ($\beta = 0.1$ for VIBI). On average, 4.26 students were assigned per instance. Further details regarding the experiments can be found in Supplementary Material S3.

As shown by the Table 2, VIBI better explains the black-box models. When explaining the movie sentiment prediction model, humans better inferred the (correctly predicted) black-box output given the five keywords when they were provided by VIBI. Therefore, it better captures the most contributing key words to the LSTM decision and better explains why the LSTM predicted each movie review by providing five key words. For explaining the digit recognition model, VIBI also highlighted the most concise chunks for explaining key characteristics of handwritten digit. Thus, it better explains how the CNN model recognized each the handwritten digit.

Conclusion

We employ the information bottleneck principle as a criterion for learning 'good' explanations, providing a convincing application of the principle in explaining a black-box decision system. Instance-wisely selected cognitive chunks work as an information bottleneck, hence, provide concise but comprehensive explanations for each decision made by a black-box system. Information bottleneck uses MI to measure briefness and comprehensiveness, and provides a principled way of balancing them. MI can measure the redundancy between raw data and explanations and measure the relevance between explanations and prediction results at the latent semantics level in a holistic way.

However, the way this information is represented may have a substantial effect on interpretability. VIBI helps to address this issue to some extent by always returning a certain form of output (i.e., a k-hot vector $\mathbf{z}$ assigned to each chunk) and having a certain form of the information bottleneck layer (i.e., a masked input) so that it makes sure that the explanations are easily understandable to humans. In practice, such a chunking strategy leads to a deviation from the strict theory that a 'good' explanation is the most compressed one but helps to achieve better interpretability in practice.

Acknowledgments

This work was supported by the grants P30DA035778 and R01GM140467 from the NIH, and Petuum Inc..

References

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE* 10(7): e0130140.

Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and Müller, K.-R. 2010. How to explain individual classification decisions. *Journal of Machine Learning Research* 11(Jun): 1803–1831.

Binder, A.; Montavon, G.; Lapuschkin, S.; Müller, K.-R.; and Samek, W. 2016. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*, 63–71. Springer.

Chen, J.; Song, L.; Wainwright, M. J.; and Jordan, M. I. 2018. Learning to Explain: An Information-Theoretic Perspective on Model Interpretation. *International Conference on Machine Learning (ICML)* .

Dabkowski, P.; and Gal, Y. 2017. Real time image saliency for black box classifiers. In *Advances in Neural Information Processing Systems*, 6967–6976.

Doshi-Velez, F.; and Kim, B. 2017. Towards a Rigorous Science of Interpretable Machine Learning. *arXiv preprint arXiv:1702.08608* .

Fong, R. C.; and Vedaldi, A. 2017. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, 3429–3437.

Guidotti, R.; Monreale, A.; Matwin, S.; and Pedreschi, D. 2019. Black Box Explanation by Learning Image Exemplars in the Latent Feature Space. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 189–205. Springer.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical reparameterization with gumbel-softmax. *International Conference on Learning Representations* .

Jokinen, E.; Heinonen, M.; Huuhtanen, J.; Mustjoki, S.; and Lähdesmäki, H. 2019. TCRGP: Determining epitope specificity of T cell receptors. *bioRxiv* 542332.

Jurtz, V. I.; Jessen, L. E.; Bentzen, A. K.; Jespersen, M. C.; Mahajan, S.; Vita, R.; Jensen, K. K.; Marcatili, P.; Hadrup, S. R.; Peters, B.; et al. 2018. NetTCR: sequence-based prediction of TCR binding to peptide-MHC complexes using convolutional neural networks. *bioRxiv* 433706.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

Lei, T.; Barzilay, R.; and Jaakkola, T. 2016. Rationalizing neural predictions. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* .

Lipton, Z. C. 2016. The Mythos of Model Interpretability. *arXiv preprint arXiv:1606.03490* .

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.

Lythe, G.; Callard, R. E.; Hoare, R. L.; and Molina-París, C. 2016. How many TCR clonotypes does a body maintain? *Journal of Theoretical Biology* 389: 214–224.

Maas, A. L.; Daly, R. E.; Pham, P. T.; Huang, D.; Ng, A. Y.; and Potts, C. 2011. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 142–150. Portland, Oregon, USA: Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

Mullenbach, J.; Wiegreffe, S.; Duke, J.; Sun, J.; and Eisenstein, J. 2018. Explainable Prediction of Medical Codes from Clinical Text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1101–1111.

Petsiuk, V.; Das, A.; and Saenko, K. 2018. Rise: Randomized input sampling for explanation of black-box models. *The British Machine Vision Conference* .

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. Why should I trust you?: Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. ACM.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 3145–3153. JMLR. org.

Shugay, M.; Bagaev, D. V.; Zvyagin, I. V.; Vroomans, R. M.; Crawford, J. C.; Dolton, G.; Komech, E. A.; Sycheva, A. L.; Koneva, A. E.; Egorov, E. S.; et al. 2017. VDJdb: a curated database of T-cell receptor sequences with known antigen specificity. *Nucleic Acids Research* 46(D1): D419–D427.

Shwartz-Ziv, R.; and Tishby, N. 2017. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810* .

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034* .

Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; and Wattenberg, M. 2017. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825* .

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 3319–3328. JMLR. org.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. *arXiv preprint* .

Tishby, N.; and Zaslavsky, N. 2015. Deep learning and the information bottleneck principle. In *Information Theory Workshop (ITW), 2015 IEEE*, 1–5. IEEE.

Vita, R.; Overton, J. A.; Greenbaum, J. A.; Ponomarenko, J.; Clark, J. D.; Cantrell, J. R.; Wheeler, D. K.; Gabbard, J. L.; Hix, D.; Sette, A.; et al. 2014. The immune epitope database (IEDB) 3.0. *Nucleic Acids Research* 43(D1): D405–D412.

Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; and Hovy, E. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–1489.

Zeiler, M. D.; and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, 818–833. Springer.

Zintgraf, L. M.; Cohen, T. S.; Adel, T.; and Welling, M. 2017. Visualizing deep neural network decisions: Prediction difference analysis. *International Conference on Learning Representations* .