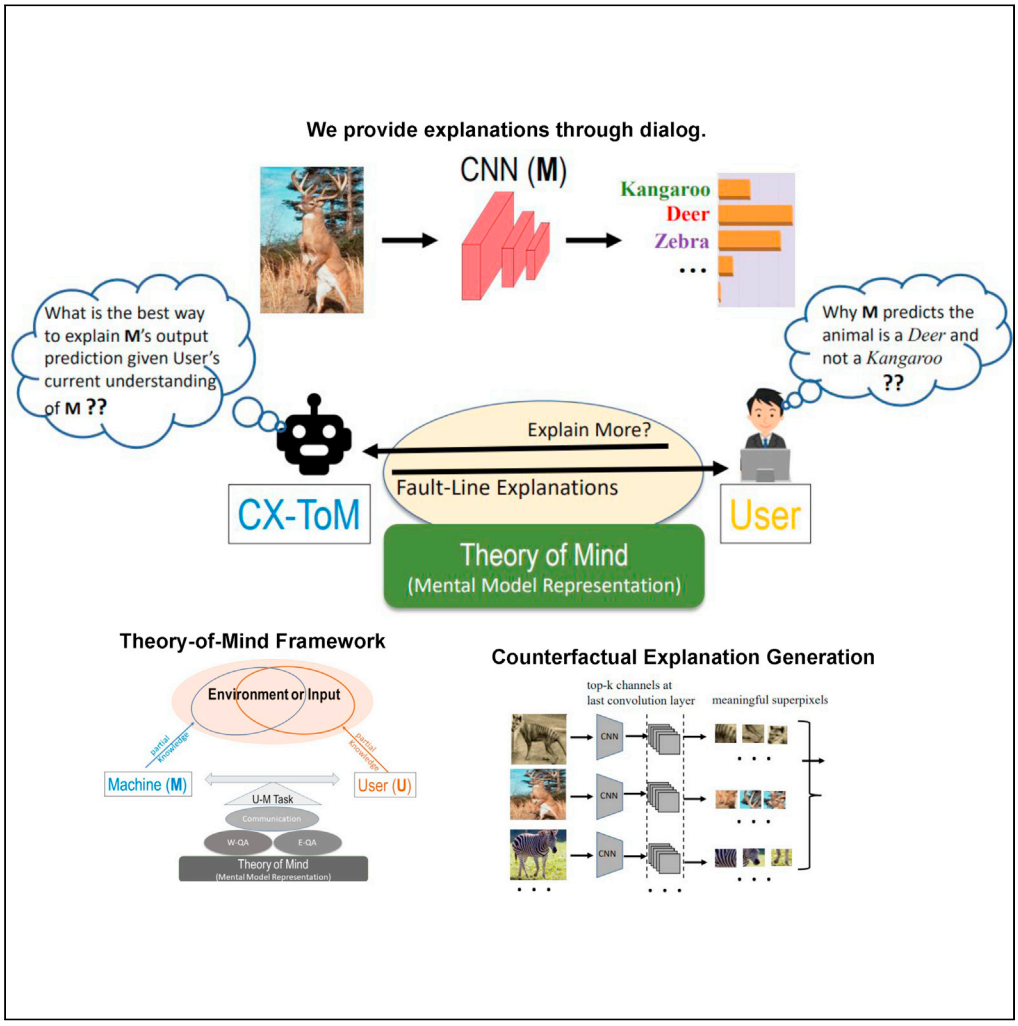
iScience

CellPress
OPEN ACCESS

Article

CX-ToM: Counterfactual explanations with theory-of-mind for enhancing human trust in image recognition models



Arjun R. Akula, Keze Wang, Changsong Liu, ..., Sinisa Todorovic, Joyce Chai, Song-Chun Zhu

aakula@ucla.edu (A.R.A.)
kezewang@gmail.com (K.W.)

Highlights

Attention is not a Good Explanation

Explanation is an Interactive Communication Process

We introduce a new XAI framework based on Theory-of-Mind and counterfactual explanations.

Akula et al., iScience 25, 103581
January 21, 2022
https://doi.org/10.1016/j.isci.2021.103581

iScience

CellPress
OPEN ACCESS

Article

CX-ToM: Counterfactual explanations with theory-of-mind for enhancing human trust in image recognition models

Arjun R. Akula,[1,5,*] Keze Wang,[1,*] Changsong Liu,[1] Sari Saba-Sadiya,[2] Hongjing Lu,[1] Sinisa Todorovic,[3] Joyce Chai,[2] and Song-Chun Zhu[4]

SUMMARY

We propose *CX-ToM*, short for counterfactual explanations with theory-of-mind, a new explainable AI (XAI) framework for explaining decisions made by a deep convolutional neural network (CNN). In contrast to the current methods in XAI that generate explanations as a single shot response, we pose explanation as an iterative communication process, i.e., dialogue between the machine and human user. More concretely, our CX-ToM framework generates a sequence of explanations in a dialogue by mediating the differences between the minds of the machine and human user. To do this, we use Theory of Mind (ToM) which helps us in explicitly modeling the human's intention, the machine's mind as inferred by the human, as well as human's mind as inferred by the machine. Moreover, most state-of-the-art XAI frameworks provide attention (or heat map) based explanations. In our work, we show that these attention-based explanations are not sufficient for increasing human trust in the underlying CNN model. In CX-ToM, we instead use counterfactual explanations called *fault-lines* which we define as follows: given an input image *I* for which a CNN classification model *M* predicts class c_{pred}, a fault-line identifies the minimal semantic-level features (e.g., *stripes* on zebra), referred to as explainable concepts, that need to be added to or deleted from *I* to alter the classification category of *I* by *M* to another specified class c_{alt}. Extensive experiments verify our hypotheses, demonstrating that our CX-ToM significantly outperforms the state-of-the-art XAI models.

INTRODUCTION

Intelligence (AI) systems are becoming increasingly ubiquitous from low risk environments such as movie recommendation systems and chatbots to high-risk environments such as medical-diagnosis and treatment, self-driving cars, drones and military applications (Chancey et al., 2015; Gulshan et al., 2016; Lyons et al., 2017; Mnih et al., 2013; Gupta et al., 2012; Pulijala et al., 2013; Dasgupta et al., 2014; Agarwal et al., 2017; Palakurthi et al., 2015; Akula et al., 2021a, 2021b, 2021c, 2021d). In particular, AI systems built using black box machine learning (ML) models – such as deep neural networks and large ensembles (Lipton, 2016; Ribeiro et al., 2016; Miller, 2018; Yang et al., 2018; Sundararajan et al., 2017; Ramprasaath et al., 2016; Zeiler and Fergus, 2014; Smilkov et al., 2017; Kim et al., 2014; Akula, 2015; Akula et al., 2013, 2020a) – perform remarkably well on a broad range of tasks and are gaining widespread adoption. However, understanding and developing human trust in these systems remains a significant challenge as they cannot explain why they reached a specific recommendation or a decision. This is especially problematic in high-risk environments such as banking, healthcare, and insurance, where AI decisions can have significant consequences.

In light of aforementioned issues, eXplainable Artificial Intelligence (XAI) has become an active area of interest in the research community and industry. XAI models, through explanations, aim at making the underlying inference mechanism of AI systems transparent and interpretable to expert users (system developers) and nonexpert users (end-users) (Lipton, 2016; Ribeiro et al., 2016; Hoffman, 2017). In this work, we focus mainly on increasing justified human trust (JT) in a deep convolutional neural network (CNN), through explanations (Hoffman et al., 2018; Akula et al., 2019a, 2019b). Justified trust is computed based on human

[1]Department of Statistics, UCLA, Los Angeles, CA 90024, USA

[2]Department of Computer Science, University of Michigan, Ann Arbor, MI 48109, USA

[3]Department of Computer Science, Oregon State University, Corvallis, OR 97331, USA

[4]Beijing Institute for General AI (BIGAI), Tsinghua University, Peking University, Beijing 100871, China

[5]Lead contact

*Correspondence: aakula@ucla.edu (A.R.A.), kezewang@gmail.com (K.W.)

https://doi.org/10.1016/j.isci.2021.103581

judgments of CNN model's prediction (more details on this are described in how human trust is measured in CX-ToM?). Despite an increasing amount of work on XAI (Smilkov et al., 2017; Sundararajan et al., 2017; Zeiler and Fergus, 2014; Kim et al., 2014; Zhang et al., 2018a; Akula et al., 2019c), providing explanations that can increase justified human trust remains an important research problem (Jain and Wallace, 2019).

Our work is motivated by the following two key observations:

1. **Attention is not a Good Explanation**: Previous studies have shown that trust is closely and positively correlated to the level of how much human users understand the AI system — *understandability* — and how accurately they can predict the system's performance on a given task — *predictability* (Hoffman, 2017; Lipton, 2016; Hoffman et al., 2018; Miller, 2018). Hence, there has been a growing interest in developing explainable AI systems (XAI) aimed at increasing understandability and predictability by providing explanations about the system's predictions to human users (Lipton, 2016; Ribeiro et al., 2016; Miller, 2018; Yang et al., 2018). Current works on XAI generate explanations about their performance in terms of, e.g., feature visualization and attention maps (Sundararajan et al., 2017; Ramprasaath et al., 2016; Zeiler and Fergus, 2014; Smilkov et al., 2017; Kim et al., 2014; Zhang et al., 2018a). However, solely generating explanations, regardless of their type (visualization or attention maps) and utility, *is not sufficient* for increasing understandability and predictability (Jain and Wallace, 2019). We verify this in our experiments.
2. **Explanation is an Interactive Communication Process**: We believe that an effective explanation cannot be one shot and involves an iterative process of communication between the human and the machine. The context of such interaction plays an important role in determining the utility of the follow-up explanations (Clark and Schaefer, 1989). As humans can easily be overwhelmed with too many or too detailed explanations, interactive communication process helps in understanding the user and identify user-specific content for explanation. Moreover, cognitive studies (Miller, 2018) have shown an explanation can only be optimal if it is generated by taking the user's perception and belief into account.

Based on the above two key observations, we introduce an interactive explanation framework, **CX-ToM**. Unlike current XAI methods that model the explanation as a single shot response, in CX-ToM, we pose the explanation generation as an iterative process of communication between the human and the machine. Central to our approach is the use of Theory-of-Mind (ToM) (Devin and Alami, 2016; Goldman, 2012; Premack and Woodruff, 1978; Bara et al., 2021) in driving the iterative dialogue by taking into account three important aspects at each dialogue turn: (a) human's intention (or curiosity), (b) human's understanding of the machine, and (c) machine's understanding of the human user. Specifically, in our framework, the machine and the user are positioned to solve a collaborative task, but the machine's mind (*M*) and the human user's mind (*U*) only have a partial knowledge of the environment (see Figure 1). Hence, the machine and user need to communicate with each other, using their partial knowledge, otherwise they would not be able to optimally solve the collaborative task. The communication consists of two different types of question-answer (QA) exchanges — namely, a) Factoid question-answers about the environment (W-QA), where the user asks "WH"-questions that begin with what, which, where, and how; and b) Explanation seeking question-answers (E-QA), where the user asks questions that begin with why about the machine's inference.

In addition, we propose novel counterfactual explanations called *fault-lines* and show that they are superior to attention based explanations. Fault-lines are the high-level semantic aspects of reality that humans zoom in on when they imagine an alternative to it. More concretely, given an input image *I* for which a CNN model *M* predicts class c_{pred}, our fault-line based explanation identifies a *minimal* set of semantic features, referred to as *explainable concepts* (xconcepts), that need to be added to or deleted from *I* to alter the classification category of *I* by *M* to another specified class c_{alt} (Byrne, 2002, 2017; Kahneman and Tversky, 1981; Akula et al., 2020b; Hoffman et al., 2017; Ruth, 2007). For example, let us consider a training dataset for an image classification task shown in Figure 2 containing the classes Dog, Thylacine, Frog, Toad, Goat and Sheep, and a CNN based classification model *M* which is trained on this dataset. To alter the model's prediction of input image I_1 from Dog to Thylacine, the fault-line ($\Psi^{+}_{I_1,c_{pred},c_{alt}}$) suggests adding *stripes* to the Dog. We call this a positive fault-line (PFT) as it involves adding a new xconcept, i.e., *stripedness*, to the input image. Similarly, to change the model prediction of I_2 from Toad to Frog, the fault-line ($\Psi^{-}_{I_2,c_{pred},c_{alt}}$) suggests removing *bumps* from the Toad. We call this a negative fault-line (NFT) as it involves subtracting xconcept, i.e., *bumpedness*, from the input image.

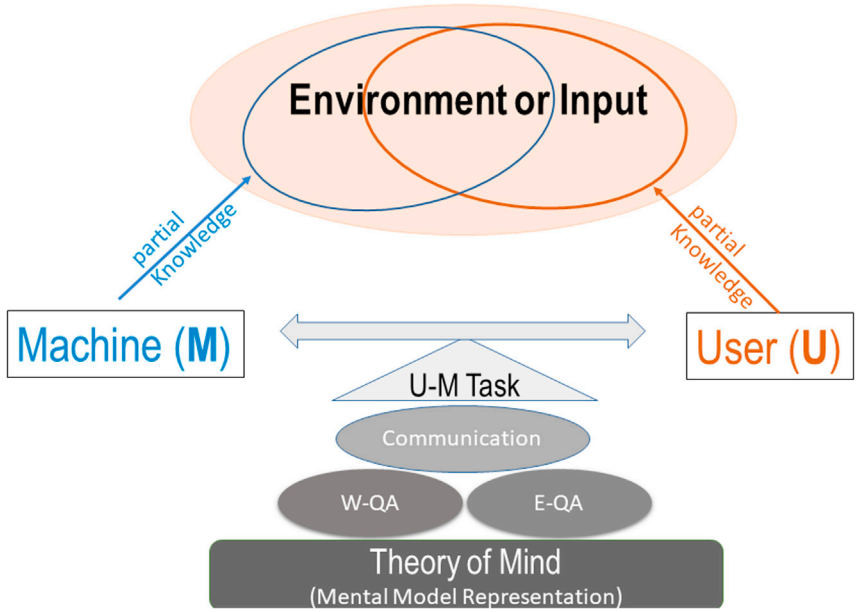


Figure 1. CX-ToM: Our interactive and collaborative XAI framework based on the Theory of Mind
The interaction is conducted through a dialogue where the user poses questions about facts in the environment (W-QA) and explanation seeking questions (E-QA).

In most cases, both PFT and NFT are needed to successfully alter the model prediction. For example, in Figure 2, to change the model prediction of I_3 from Goat to Sheep, we need to add a xconcept *wool* (PFT) to I_3 and also remove xconcepts *beard* and *horns* (NFT) from I_3. As we can see, these fault-lines can be directly used to make the internal decision making criteria of deep neural networks transparent to both expert and nonexpert users. For instance, we answer the question *"Why does the model classify the image I_3 as* Goat *instead of* Sheep*?"* by using PFT $\Psi^{+}_{I_3,c_{pred},c_{alt}}$ and NFT $\Psi^{-}_{I_3,c_{pred},c_{alt}}$ as follows: "Model thinks the input image is Goat and not Sheep mainly because Sheep's feature *woolly* is absent in I_3 and Goat's features beard and horns are present in I_3". It may be noted that there could be several other features of Sheep and Goat that might have influenced the model's prediction. However, fault-lines only capture the most critical (minimal) features that highly influenced the model's prediction.

Note that fault-lines are **counter-factual** in nature, i.e., they provide a *minimal* amount of information capable of altering a decision. This makes them easily digestible and practically useful for understanding the reasons for a model's decision (Wachter et al., 2017). For example, consider the fault-line explanation for image I_3 in Figure 2. The explanation provides only the most critical changes (i.e., adding wool and removing beard and horns) required to alter the model's prediction from Goat to Sheep, though several other changes may be necessary. Although there are recent works on generating pixel-level counter-factual and contrastive explanations (Hendricks et al., 2018; Dhurandhar et al., 2018; Goyal et al., 2019), to the best of our knowledge, this is the first work to propose a method for generating explanations that are iterative, counter-factual as well as conceptual.

It may be noted that there exists multiple fault-lines that could be used to explain the model's decisions. In this work, we pick the most optimal fault-line, i.e., the one that is most influential and suitable given the user's current understanding of CNN model, by using Theory-of-Mind (ToM) (Yoshida et al., 2008; Rabinowitz et al., 2018; Pearce et al., 2014; Raileanu et al., 2018; Ramırez and Geffner, 2011; Edmonds et al., 2019; Zhang and Zhu, 2018).

Example of a ToM based fault-line selection process

Given an input image and two output categories, fault-lines show the most important features or attributes that influence the model's decision in classifying the image as one among the two output categories. In most cases, there exist several thousands of output categories and it is impossible for the human user to

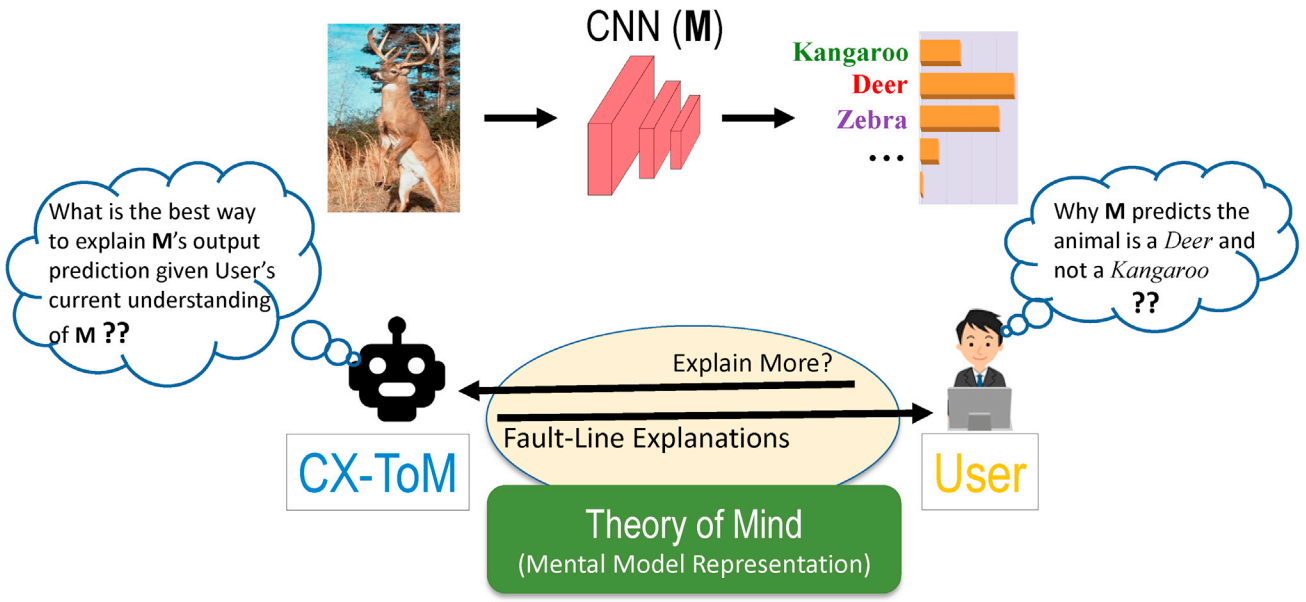


Figure 2. Example of a ToM based Fault-Line Selection Process: The interaction is conducted through a dialogue where the user seeks explanations about CNN output predictions

CX-ToM picks an optimal fault-line as an explanation based on the user's (estimated) current understanding of the model.

verify the model's reasoning and behavior by constructing a fault-line between all the possible pairs of output categories. Therefore, it is important for the model to automatically select the most important pair for constructing fault-line explanation that helps human users to quickly understand the model's strengths or weaknesses. CX-ToM addresses this by incorporating Theory-of-Mind framework which helps in explicitly tracking human user's beliefs. More concretely, at each turn in the dialogue, we estimate the human's understanding of the CNN model and generate a most suitable fault-line explanation aimed at increasing human understanding (and therefore trust) of the model. It may be noted that we are not trying to estimate or build a rich and dynamic true state of a human mind using ToM - a grand challenge for AI. Instead, similar to prior works on ToM (Yoshida et al., 2008; Rabinowitz et al., 2018; Pearce et al., 2014; Raileanu et al., 2018; Ramırez and Geffner, 2011; Zhu et al., 2020), we cast ToM framework as a simple learning problem that enable us to better understand user preferences that improve the utility of the explanations.

For example, consider an input image shown in Figure 3, where the CNN model classifies the image as a Woman. The possible output categories are Woman, Man, and Deer. Generating a most suitable fault-line explanation to help users understand the model's reasoning process requires understanding the human user's current understanding of the model. If the user knows that the model performs well at identifying Person but not very certain in its ability to correctly classify between Man and Woman, then the fault-line for the class pair <Woman, Man> is a most appropriate explanation for the user. However, if the user is not certain about the model's ability in correctly classifying Person, then the fault-line for the class pair <Woman, Deer> is the most appropriate explanation.

In summary, CX-ToM constructs explanations in the dialogue using fault-lines and picks an optimal explanation based on ToM. We perform extensive human study experiments to demonstrate the effectiveness of our approach in improving human understanding of the underlying classification model. Through our ablations and human studies, we show that our CX-ToM explanations significantly outperform the baselines (i.e., attribution techniques and pixel-level counterfactual explanations) in terms of qualitative and quantitative metrics such as Trust and Explanation Satisfaction (Hoffman et al., 2018).

How is human trust measured in CX-ToM?

In this work, we focus mainly on measuring and increasing **Justified Positive Trust** (JPT) and **Justified Negative Trust** (JNT) (Hoffman et al., 2018) in image classification models. We measure JPT and JNT by evaluating the human's understanding of the machine's (M) decision-making process. For example, if the image

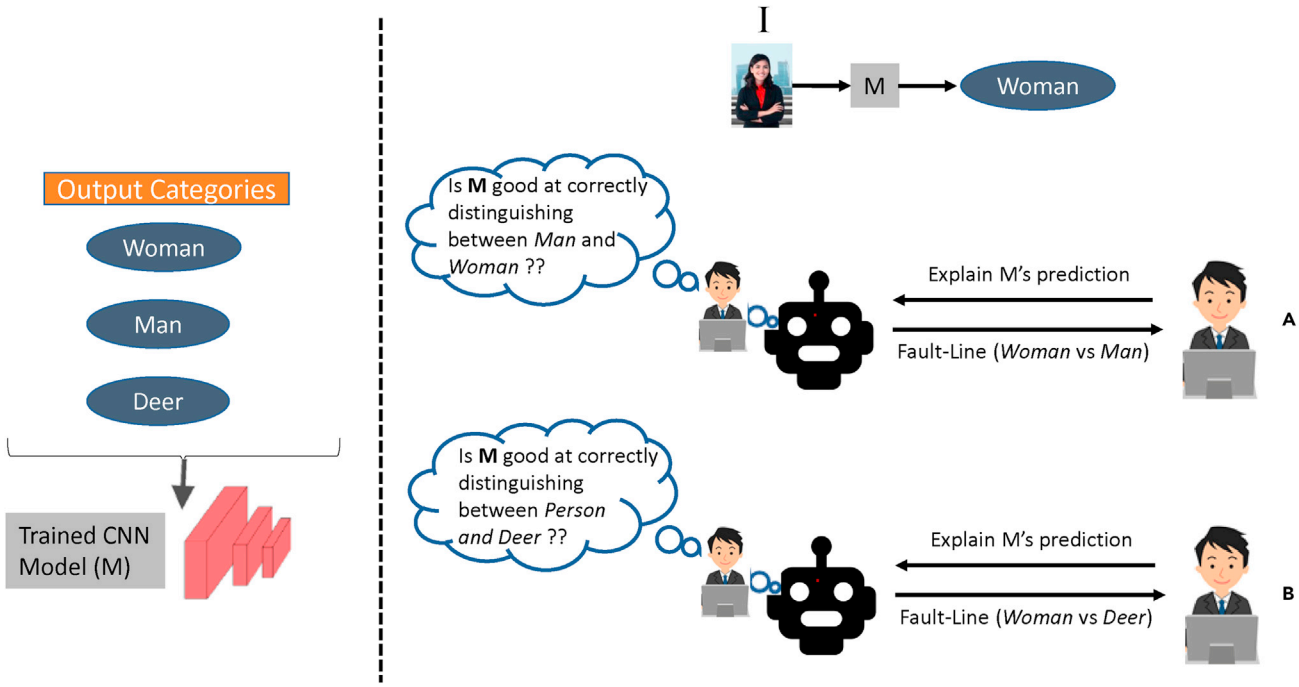


Figure 3. We select a fault-line explanation by estimating human user's current understanding of the model

(A) For example, consider the first scenario (A), where CX-ToM estimates that the user is not confident in the model's ability to correctly classify between Woman and Man. Therefore, CX-ToM generates a fault-line explanation using the output categories Woman and Man.

(B) Whereas in the second scenario (B), CX-ToM thinks that users do not trust the model's ability in correctly classifying Person, and therefore shows a fault-line explanation using categories Woman and Deer.

classification model *M* predicts images in the set *C* correctly and makes incorrect decisions on the images in the set *W*. Intuitively, JPT will be computed as the percentage of images in *C* that the human subject felt *M* would correctly predict. Similarly, JNT (also called as mistrust), will be computed as the percentage of images in *W* that the human subject felt *M* would fail to predict correctly. In other words, given an image, justified trust evaluates whether the users could reliably predict the model's output decision. Note that this definition of justified trust is domain generic and can be easily adapted to any task. For example, in an AI-driven clinical world, our definitions of JPT and JNT can effectively measure how much doctors and patients understand the AI systems that assist in clinical decisions.

Our contributions are summarized below:

- We introduce a new XAI framework based on Theory-of-Mind and counterfactual explanations.
- We present a ToM based approach to automatically select the most important pair of output categories for constructing fault-line explanation.
- We show that the CX-ToM XAI framework qualitatively and quantitatively outperforms baselines in improving human understanding of the classification model.

The remainder of this paper is organized as follows. Related work reviews the previous work done in explaining image classification models. CX-ToM framework introduces our CX-ToM explanation framework. In experiments, we present our experimental results, draw conclusions, and point to future directions for research.

RELATED WORK

The importance of generating explanations or justifications of decisions made by an AI system has been emphasized and widely explored in numerous works over the past decades (Alang, 2017; Bornstein, 2016; Champlin et al., 2017; Bach et al., 2015; Shrikumar et al., 2017; Zhou et al., 2016; Berry and Broadbent,

1987; Biran and Cotton, 2017; Darlington, 2013; Doshi-Velez and Kim, 2017a, 2017b; Goodman and Flaxman, 2017; Hoffman, 2017; Hoffman and Klein, 2017; Keil, 2006; Kulesza et al., 2010, 2011; Moore and Swartout, 1990; Walton, 2004; Douglas, 2007; Walton, 2011; Sheh, 2017; Sheh and Monteath, 2018; Tapaswi et al., 2016; Williams et al., 2016; Agarwal et al., 2018; Akula et al., 2018, 2019a, 2019b, 2019c, 2021d; Akula and Zhu, 2019; Gupta et al., 2016; Bivens et al., 2017; Zhang et al., 2019a, 2020a, 2020b). Most prior work in explaining CNN's predictions has focused on generating explanations using feature visualization and attribution.

Feature visualization techniques typically identify qualitative interpretations of features used for making predictions or decisions. For example, gradient ascent optimization is used in the image space to visualize the hidden feature layers of unsupervised deep architectures (Erhan et al., 2009). In addition, convolutional layers are visualized by reconstructing the input of each layer from its output (Zeiler and Fergus, 2014). Recent visual explanation models seek to jointly classify the image and explain why the predicted class label is appropriate for the image (Hendricks et al., 2016). Other related work includes a visualization-based explanation framework for Naive Bayes classifiers (Szafron et al., 2003), an interpretable character-level language models for analyzing the predictions in RNNs (Karpathy et al., 2015), and an interactive visualization for facilitating analysis of RNN hidden states (Strobelt et al., 2016).

Attribution is a set of techniques that highlight pixels of the input image (saliency maps) that most caused the output classification. Gradient-based visualization methods (Zhou et al., 2016; Selvaraju et al., 2017a) have been proposed to extract image regions responsible for the network output. The LIME method proposed by (Ribeiro et al., 2016) explains predictions of any classifier by approximating it locally with an interpretable model. SHAP (Lundberg and Lee, 2017), another common attribution technique, uses shapley values to explain output predictions of a model for given input by computing the contribution of each feature to the prediction.

There are few recent works in the XAI literature that go beyond the pixel-level explanations. For example, the TCAV technique proposed by (Kim et al., 2018) aims to generate explanations based on high-level user defined concepts. Contrastive explanations are proposed by (Dhurandhar et al., 2018) to identify minimal and sufficient features to justify the classification result (Goyal et al., 2019). proposed counterfactual visual explanations that identify how the input could change such that the underlying vision system would make a different decision. More recently, few methods have been developed for building models which are intrinsically interpretable (Zhang et al., 2018a). In addition, there are several works (Miller, 2018; Hilton, 1990; Lombrozo, 2006) on the goodness measures of explanations which aim to assess the underlying characteristics of explanations.

We further categorize above works on feature visualization and attribution as follows:

Intrinsic vs post-hoc explanations

Explanations that are derived (or understood) directly from the model's internal representation or the output parse structure are called Intrinsic Explanations (Doshi-Velez and Kim, 2017b; Zhang et al., 2018a; Stone et al., 2017). For example, the reasoning behind the predictions made by linear regression models, decision trees, and And-Or Graphs (Li et al., 2013; Zhang et al., 2017) is easier to understand without using any external XAI models and hence are considered as intrinsically explainable. These models, because of their simple structure, typically do not fare well in terms of performance compared to black-box models such as deep neural nets. Majority of the work in XAI is focused on generating post-hoc (Lei et al., 2016; Ribeiro et al., 2016; Kim et al., 2014, 2015, 2018; Wang et al., 2016) explanations where an external XAI model is employed to explain the model. More recently, there are efforts in making the complex deep neural networks intrinsically explainable (Zhang et al., 2018a; 2018b; 2019b). For example (Zhang et al., 2019b), proposed a decision tree to encode decision modes in fully-connected layers and thereby quantitatively explain the logic for each CNN prediction.

Model-agnostic vs model-specific explanations

Explainable AI models that do not require CNN model specific details (for example, weights of CNN) for generating explanations are called model-agnostic models (Ribeiro et al., 2018). In other words, they simply analyze the dependencies of input features against the output predictions to explain the model's

decision. It may be noted that intrinsic explanations are typically model-specific whereas post-hoc XAI models are model-agnostic. Several XAI works belong to this category, to name a few:

1. *Local Interpretable Model-Agnostic Explanation (LIME)* (Ribeiro et al., 2016). LIME produces an attention map as an explanation, generated through super-pixel based perturbation. Though LIME is a post-hoc model-agnostic model, it generates explanations by approximating the model (locally) with an intrinsic model-specific XAI model.
2. *Contrastive Explanation Methods (CEM)* (Dhurandhar et al., 2018). CEM provides contrastive explanations by identifying pertinent positives and pertinent negatives in the input image.
3. *Counterfactual Visual Explanations (CVE)* (Goyal et al., 2019). CVE provides counterfactual explanation describing what changes to the situation would have resulted in arriving at the alternative decision.

Human interpretable explanations (concept activation vectors)

Most XAI models represent the explanations using attention maps (saliency). However, these explanations are difficult for humans to understand. For example, authors in (Jain and Wallace, 2019) considered NLP tasks (text classification, natural language inference (NLI), and question answering) to show that attention mechanisms are not useful for humans. Therefore, there is a dire need to represent and generate human-friendly explanations. Recent work by (Kim et al., 2018) presents a first step toward this goal. They propose a technique called TCAV that takes the user defined concept (X) represented using a set of example images and maps it to the activation space of any given layer l in the network. It then constructs a vector representation of each concept, called CAV (denoted as v_X), by using a direction normal to a linear classifier trained to distinguish between the concept activations from the random activations. The sensitivity of network predictions toward a concept is gauged by computing directional derivatives ($S_{c,X}$) to produce estimates of how important the concept X was for a CNN's prediction of a target class c, e.g., how important is the concept stripedness for predicting the zebra class.

$$S_{c,X} = \nabla g_c(f(l)) \cdot v_X \quad \text{(Equation 1)}$$

where g_c denotes the classifier component of CNN that takes output of f and predicts log-probability of output class c. Because TCAV provides explanations using high-level concepts, it is expected to achieve higher human trust and reliance values compared to the attention based explanations (Selvaraju et al., 2017a; Ribeiro et al., 2016).

Proxy or surrogate models

A Proxy or surrogate model is a simpler interpretable model that approximates the behavior of the complex model (Ribeiro et al., 2016; Alvarez-Melis and Jaakkola, 2018; Sato and Tsukimoto, 2001; Augasta and Kathirvalavakumar, 2012). It reduces the complexity of the original model but produces similar output estimates. Most surrogate XAI models are model-agnostic. A surrogate model that is trained to explain individual instances is referred to as local surrogate model. For example, LIME (Ribeiro et al., 2016) approximates a model with a local linear model that serves as a surrogate for the model in the neighborhood of the input. Similarly, neural networks are locally approximated by using decision trees (Sato and Tsukimoto, 2001; Zhang et al., 2017). This notion of using proxy models is also referred to as Knowledge Distillation (Hinton et al., 2015; Hernández-García and König, 2018; Polino et al., 2018) and Rule Extraction (Zilke et al., 2016).

Perturbation analysis

Perturbation analysis helps in measuring the feature importance for the predictions made by model (Fisher et al., 2018; Moosavi-Dezfooli et al., 2017). The assumption here is that the model's confidence in the prediction will be low if an important feature has been removed (or masked) after perturbing the input features. Adversarial analysis (Goodfellow et al., 2014) and Probing techniques (Clark et al., 2019) are few popular techniques for perturbation analysis.

Counterfactual explanations

Counterfactual (and Contrastive) explanations provide a *minimal* amount of information capable of altering a model's decision. In other words, they aim at describing the causal situations such as "What would be the

output of the model if X had not occurred?". This makes them easily digestible and practically useful for understanding the reasons for a model's decision (Pedreschi et al., 2018; Wachter et al., 2017; Goyal et al., 2019; Van Looveren and Klaise, 2019).

For example (Fong and Vedaldi, 2017), propose a counterfactual reasoning framework to find the part of an image most responsible for a classifier decision. This saliency based explanation framework helps in understanding where the model looks by discovering which parts of an image most affect its output score when perturbed (Goyal et al., 2019) and proposes a counterfactual explanation framework to identify how the input image could be changed such that the model would output a different specified class. To do this, they select a distractor image that the model predicts as class c_1 and identify spatial regions such that replacing the identified region in input image with the regions from the distractor image would push the model toward classifying I as c_2. Contrastive explanations are proposed by (Dhurandhar et al., 2018) to identify minimal and sufficient features to justify the classification result. Unlike these prior counterfactual explanation frameworks which mainly focus on pixel-level explanations (viz. saliency maps), our proposed ToM based counterfactual explanations, i.e., fault-lines, are **concept-level** explanations. Pixel-level explanations are not effective at human scale, whereas concept level explanations are effective, less ambiguous, and more natural for both expert and nonexpert users in building a mental model of a vision system (Kim et al., 2018). Moreover, with conceptual explanations, humans can easily generalize their understanding to new unseen instances/tasks.

Partial dependence plots

Partial dependence plots (PD) is a model-agnostic XAI technique that helps in understanding the relationships between one or more input variables as well as marginal effect of a given variable on a model's decision (Friedman, 2001; Hastie et al., 2001; Molnar, 2019).

Class Activation Mapping (CAM)

CAM produces an attention map as an explanation, i.e., it highlights the important regions in the image for predicting a target output. Gradient-weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al., 2017a) uses the gradients of the target class flowing into the final convolutional layer to produce an attention map as explanation. Layer-wise Relevance Propagation (LRP) (Bach et al., 2015) generates attention map by propagating classification probability backward through the network and then calculates relevance scores for all pixels. SmoothGrad (Smilkov et al., 2017) produces an attention map as an explanation by adding Gaussian noise to the original image and then calculating gradients multiple times and averaging the results.

CX-TOM FRAMEWORK

In this section, we first demonstrate the importance of ToM based explanations by designing a collaborative task-solving game for visual recognition (importance of ToM). We next present the fault-lines as an alternative to attention based explanations (fault-lines as an alternative to attention based explanations). Finally, we detail our CX-ToM model which integrates both ToM and fault-lines into one single explanation framework (CX-ToM framework).

Importance of ToM

We test the importance of ToM for providing effective explanations by designing a collaborative task-solving game for visual recognition. In this game, the machine is given an original image and is supposed to detect and localize objects and parts of interest or a human activity appearing in the image. The user is given a blurred version of the original image, and the user seeks the machine's help essentially through the explanations generated by the machine to recognize objects/parts in the blurred image. This provides a unique collaborative setting where the system is motivated to provide a human-understandable explanation for its visual recognition and the user is motivated to seek the system's recognition and explanation to help his/her own understanding. To facilitate this collaborative interaction, we use ToM to explicitly model mental states of visual understanding ("minds") of the machine and user using parse graphs (*pg*) in the form of And-Or Graph (AOG) (Zhu and Mumford, 2007). In a *pg*, nodes represent objects and parts detected in the image, and edges represent spatial relationships identified between the objects. As shown in Figure 4, we have three main components as part of this interaction:

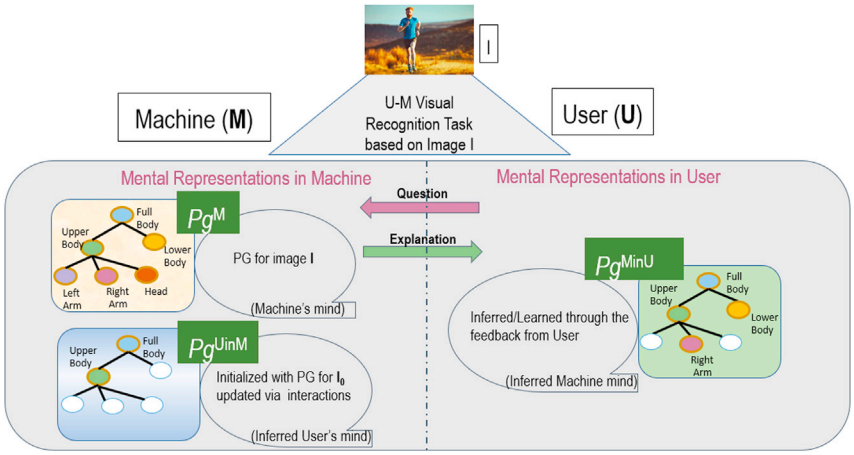


Figure 4. Our collaborative ToM based interaction framework for a visual recognition task consists of three distinct parse graphs (*pg*'s): pg^{M} representing the machine's interpretation of the image, pg^{UinM} — the human's mind as inferred by the machine, and pg^{MinU} — the machine's mind as inferred by the human

Nodes of a parse graph represent objects and parts appearing in the image, and edges represent spatial relationships of the objects. We use ToM to optimize explanations so as to reduce a difference among the three parse graphs.

- A **Performer** that generates image interpretations (i.e., machine's mind represented as pg^{M}) using a set of computer vision algorithms;
- An **Explainer** that generates maximum utility explanations in a dialogue with the user by accounting for pg^{M} and pg^{UinM} using reinforcement learning;
- An **Evaluator** that quantitatively evaluates the effect of explanations on the human's understanding of the machine's behaviors (i.e., pg^{MinU}) and measures human trust by comparing pg^{MinU} and pg^{M}.

The game consists of two phases. In the first phase, the user is shown a blurred image and given a task to recognize what the image shows. The machine has access to the original (unblurred) image and the machine's (i.e., **Performer's**) inference result pg^{M}. The user is allowed to ask questions regarding objects and parts in the image that the user finds relevant for his/her own recognition task. Using the detected objects and parts in pg^{M}, **Explainer** provides visual explanations to the user, as shown in Figure 5. This process allows the machine to infer what the user sees and iteratively update pg^{UinM}, and thus select an optimal explanation at every turn of the game. Optimal explanations generated by the **Explainer** are the key to maximize the human trust in the machine.

The second phase is specifically designed for evaluating whether the explanation provided in the first phase helps the user understand the system behaviors. The **Evaluator** shows a set of original (unblurred) images to the user that are similar to (but different from) the ones used in the first phase of the game (i.e., the set of images shows the same class of objects or human activity). The user is then given a task to predict in each image the locations of objects and parts that would be detected by the machine (i.e., in pg^{M}) according to his/her understanding of the machine's behaviors. Based on the human predictions, the **Evaluator** estimates pg^{MinU} and quantifies human trust in the machine by comparing pg^{MinU} and pg^{M}.

The three minds pg^{M}, pg^{MinU}, and pg^{UinM} are subgraphs of an And-Or Graph (AOG) defining all objects, parts, and their relationships and attributes of the visual domain considered. The AOG uses AND nodes to represent decompositions of human body parts into subparts and OR nodes for alternative decompositions. Each node is characterized by attributes that pertain to the corresponding human body part, including the pose and action of the entire body. In addition, edges in the AOG capture hierarchical and contextual relationships of the human body parts. Our AOG-based performer uses three inference processes α, β, and γ at each node. Figure 5 shows an example part of the AOG relevant for human body pose estimation (Park et al., 2018). Theα process detects nodes (i.e., human body parts) of the AOG directly based on image features, without taking advantage of the surrounding context. The β process infers nodes of the AOG by binding the previously detected children nodes in a bottom-up fashion,

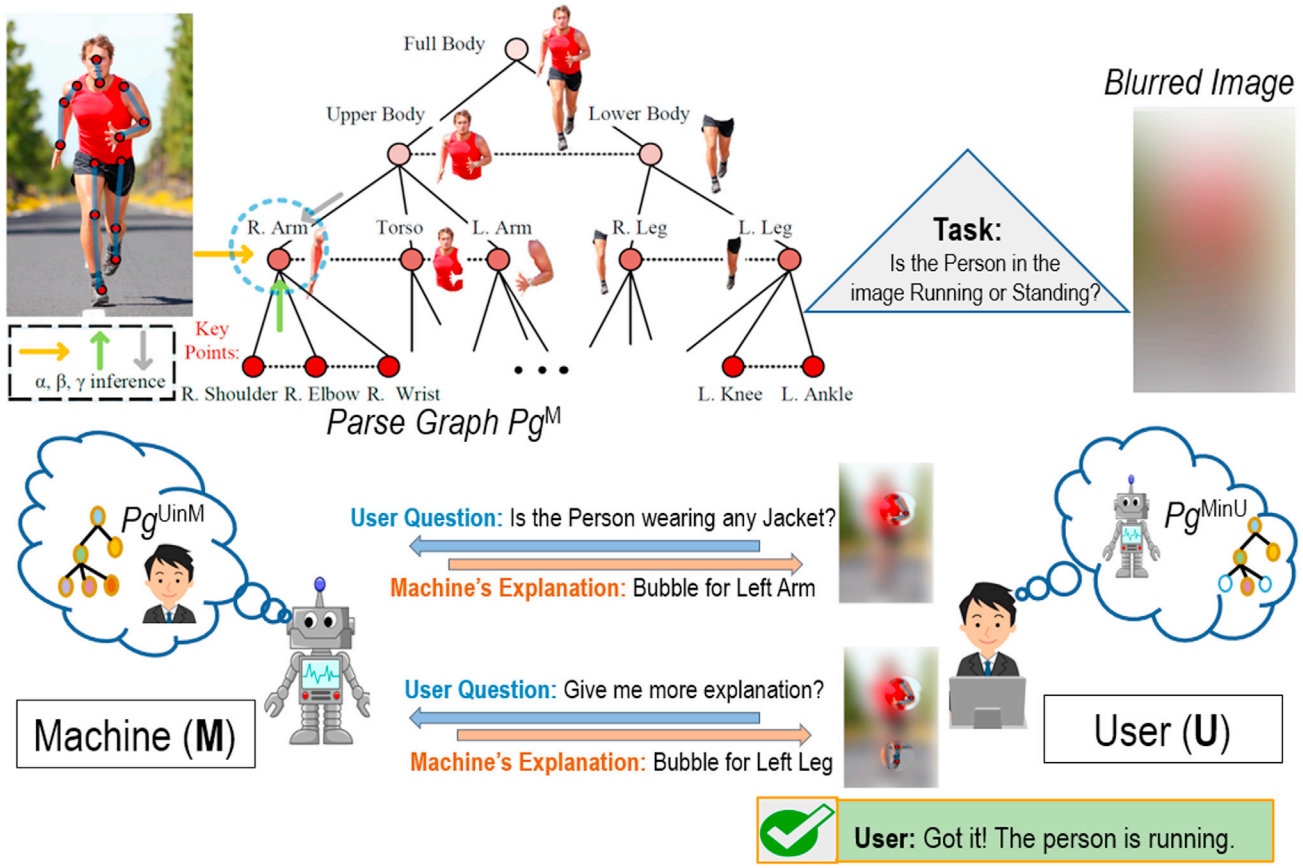


Figure 5. An example of the first phase of our ToM based collaborative game aimed at estimating pg^{UinM}: The user is shown a blurred image and given a task to recognize if the person in the image is running or walking

The machine has access to the original (unblurred) image and pg^{M}. The user then asks questions regarding objects and parts in the image. Using the detections in pg^{M}, the machine provides visual explanations as "bubbles" that reveal the corresponding image parts in the blurred image. The generated explanations are used to update pg^{UinM}.

where the children nodes have been detected by the α process (e.g., detecting human's upper body from the detected right arm, torso, and left arm). Note that the β process is robust to partial object occlusions as it can infer an object from its detected parts. The γ process infers a node of the AOG top-down from its previously detected parent nodes, where the parents have been detected by the α process (e.g., detecting human's right leg from the detected outline of the lower body). The parent node passes contextual information so that the performer can detect the presence of an object or part from its surroundings. Note that the γ process is robust to variations in scale at which objects appear in images.

The explainer, in the first phase of the game, makes the underlying α, β, and γ inference process of the performer more transparent to the human through a collaborative dialogue. At one end, the explainer is provided access to an image and the performer's inference result pg^{M} on that image. At the other end, the human is presented with a blurred version of the same image, and asked to recognize a body part, or pose, or human action depicted (e.g., whether the person is running or walking). To solve the task, the human may ask the explainer various "what", "where," and "how" questions (e.g., "Where is the left arm in the image"). We make the assumption that the human will always ask questions that are related to the task at hand so as to solve it efficiently. As visual explanations, we use "bubbles" (Gosselin and Schyns, 2001), where each bubble reveals a circular part of the blurred image to the human. The bubbles coincide with relevant image parts for answering the question from the human, as inferred by the performer in pg^{M}. For example, a bubble may unblur the person's left leg in the blurred image, because that image part has been estimated in pg^{M} as relevant for recognizing the human action "running" occurring in the image.

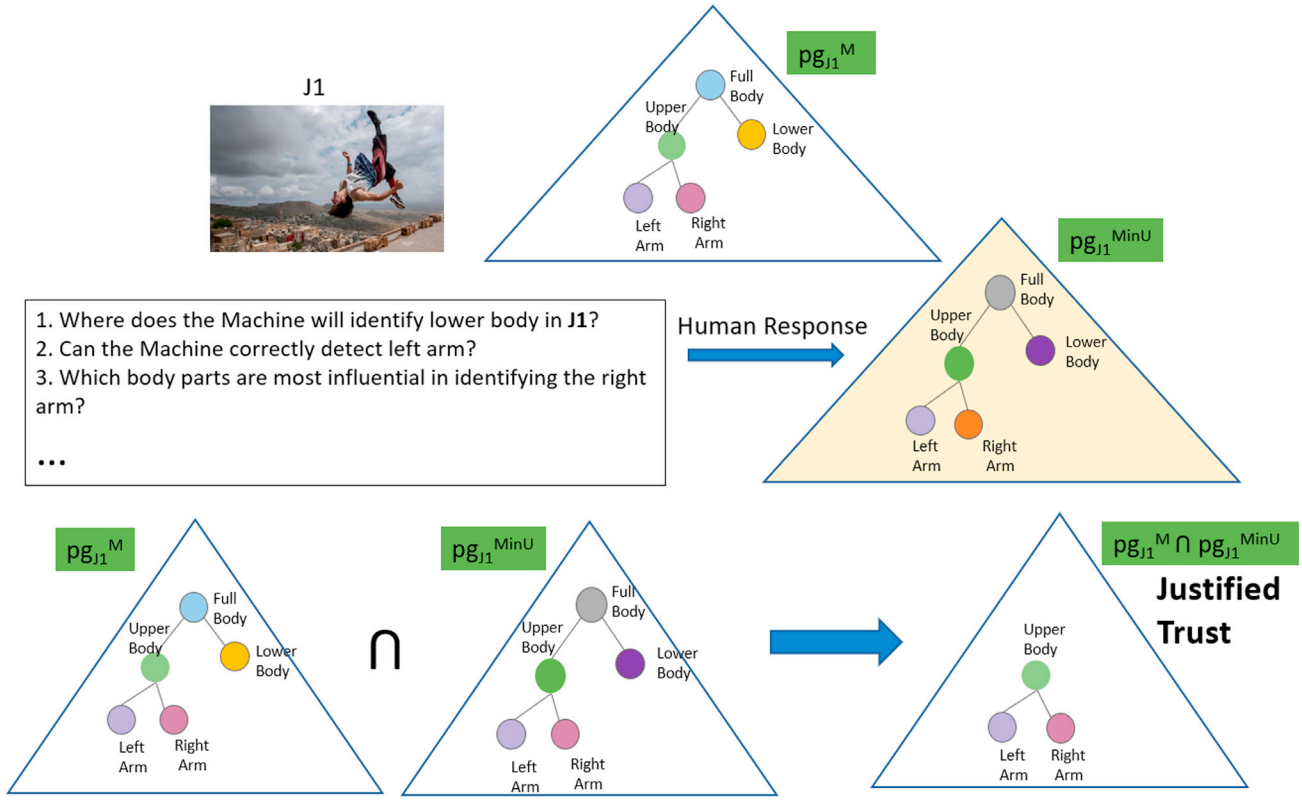


Figure 6. An example of second phase of ToM game where we estimate pg^{MinU} and also quantitatively compute justified trust

The second phase of the X-ToM game serves to assess the effect of the explainer on the human's understanding of the performer. This assessment is conducted by the evaluator. The human is presented with a set of (unblurred) images that are different from those used in the first phase. For every image, the evaluator asks the human to predict the performer's output. The evaluator poses multiple-choice questions and the user clicks on one or more answers. As shown in Figure 6, we design these questions to capture different aspects of human's understanding of α, β, and γ inference processes in the performer. Based on responses from the human, the evaluator estimates pg^{MinU}. By comparing pg^{MinU} with the actual machine's mind pg^{M} (generated by the performer), we have defined the following metrics to quantitatively assess human trust (Hoffman, 2017; Hoffman et al., 2010, 2018; Miller, 2018) in the performer:

(1) *Justified Positive and Negative Trust:* It is possible for humans to feel positive trust with respect to certain tasks, while feeling negative trust (i.e., mistrust) on some other tasks. The positive and negative trust can be a mixture of justified and unjustified trust (Hoffman, 2017; Hoffman et al., 2018). We compute justified positive trust (JPT) and negative trust (JNT) as follows:

$$\mathrm{JPT} = \frac{1}{N}\sum_{i}\sum_{z=\alpha,\beta,\gamma}\Delta\mathrm{JPT}(i,z),$$

$$\Delta\mathrm{JPT}(i,z) = \frac{\left|\left|pg_{i,z,+}^{MinU} \cap pg_{i,+}^{M}\right|\right|}{\left|\left|pg_{i,+}^{M}\right|\right|},$$

$$\mathrm{JNT} = \frac{1}{N}\sum_{i}\sum_{z=\alpha,\beta,\gamma}\Delta\mathrm{JNT}(i,z),$$

$$\Delta\mathrm{JNT}(i,z) = \frac{\|pg_{i,z,-}^{MinU} \cap pg_{i,-}^{M}\|}{\left|\left|pg_{i,-}^{M}\right|\right|},$$

where N is the total number of games played. z is the type of inference process. $\Delta JPT(i,z)$, $\Delta JNT(i,z)$ denote the justified positive and negative trust gained in the i-th turn of a game on the z inference process, respectively. $pg_{i,z,+}^{MinU}$ denotes the nodes in pg_i^{MinU} for which the user thinks the performer is able to accurately detect in the image using the z inference process. Similarly, $pg_{i,z,-}^{MinU}$ denotes nodes in pg_i^{MinU} for which the user thinks the performer would fail to detect in the image using the z inference process. $||pg||$ is the size of pg. Symbol $\cap$ denote the graph intersection of all nodes and edges from two pg's.

(2) *Reliance:* Reliance (Rc) captures the extent to which a human can accurately predict the performer's inference results without over- or under-estimation. In other words, Reliance is proportional to the sum of JPT and JNT.

$$Rc = \frac{1}{N}\sum_{i}\sum_{z=\alpha,\beta,\gamma}\Delta Rc(i,z),$$

$$\Delta Rc(i,z) = \frac{\left|\left|pg_{i,z}^{MinU} \cap pg_{i,z}^{M}\right|\right|}{||pg_i^M||}.$$

We deployed the ToM game on the Amazon Mechanical Turk (AMT) and trained the Explainer through the interactions with turkers. All the turkers have a bachelor's degree or higher. We used three visual recognition tasks to our experiments, namely, human body parts identification, pose estimation, and action identification. We used 1000 images randomly selected from Extended Leeds Sports (LSP) dataset (Johnson and Everingham, 2010). Each image is used in all the three tasks. During training, each trial consists of one ToM game where a turker solves a given task on a given image. We restrict Turkers from solving a task on an image more than once. In total, about 2400 unique workers contributed in our experiments. We performed off-policy updates after every 200 trials, using Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.001 and gradients were clipped at [-5.0, 5.0] to avoid explosion. We used ϵ-greedy policy, which was annealed from 0.6 to 0.0. We stopped the training once the model converged. Using our Evaluator module, we conduct human subject experiments to assess the effectiveness of the ToM Explainer, that is trained on AMT, in increasing human trust through explanations. We recruited 120 human subjects from our institution's Psychology subject pool (these experiments were reviewed and approved by our institution's IRB). We applied between-subject design and randomly assigned each subject into one of the three groups. One group used ToM Explainer, and two groups used the following two baselines, respectively:

- Ω_{QA}: we measure the gains in human trust only by revealing the answers for the tasks without providing any explanations to the human.
- $\Omega_{Salience}$: in addition to the answers, we also provide saliency maps generated using attribution techniques to the human as explanations (Zhou et al., 2016; Selvaraju et al., 2017b).

Within each group, each subject will first go through an introduction phase where we introduce the tasks to the subjects. Next, they will go through a familiarization phase where the subjects become familiar with the machine's underlying inference process (Performer), followed by a testing phase where we apply our trust metrics and assess their trust in the underlying Performer.

Figure 7 compares the justified positive trust (JPT), justified negative trust (JPT), and Reliance (Rc) of the ToM Explainer with the baselines. As we can see, JPT, JNT, and Rc values of ToM based framework are significantly higher than Ω_{QA} and $\Omega_{Salience}$ ($p < 0.01$). *In addition, it should be noted that attribution techniques ($\Omega_{Salience}$) did not perform any better than the Ω_{QA} baseline where no explanations are provided to the user*. This could be attributed to the fact that, though saliency maps help human subjects in localizing the region in the image based on which the performer made a decision, they do not necessarily reflect the underlying inference mechanism. In contrast, ToM Explainer makes the underlying inference processes (α, β, andγ) more explicit and transparent and also provides explanations tailored for individual user's perception and understanding. Therefore, ToM explanations lead to the significantly higher values of JPT, JNT, and Rc, confirming our hypothesis that ToM helps in providing effective explanations to the user.

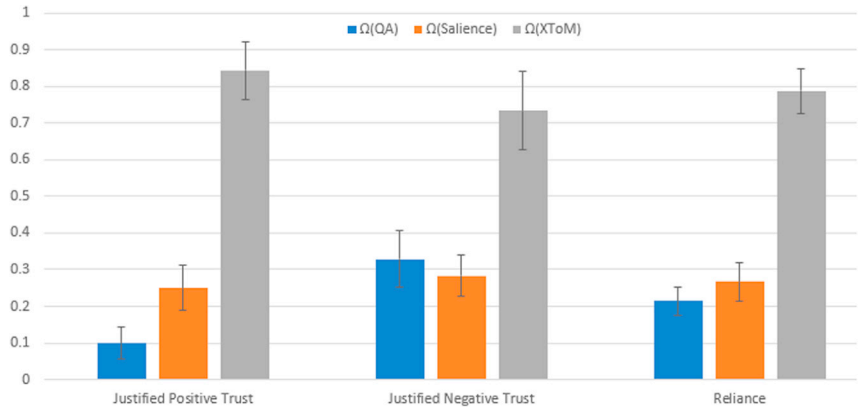


Figure 7. Gain in Justified Positive Trust, Justified Negative Trust and Reliance: our ToM framework (denoted as X-ToM) vs baselines (QA, Saliency Maps)
Error bars denote standard errors of the means.

Fault-lines as an alternative to attention based explanations

In this section, we detail our ideas and methods for generating fault-line explanations. Without loss of generality, we consider a pre-trained CNN (M) for image classification. Given an input image I, the CNN predicts a log-probability output $\log P(\mathbf{Y}|I)$ over the output classes $\mathbf{Y}$. Let χ denote a dataset of training images, where $\chi_c \subset \chi$ represents the subset that belongs to category $c \in \mathbf{Y}$, (c = 1, 2, …, C). We denote the score (logit) for class c (before the softmax) as y^c and the predicted class label as c_{pred}. Our high-level goal is to find a fault-line explanation (Ψ) that alters the CNN prediction from c_{pred} to another specified class c_{alt} using a minimal number of xconcepts. We follow (Kim et al., 2018) in defining the notion of xconcepts where each xconcept is represented using a set of example images. This representation of xconcepts provides great flexibility and portability as it will not be constrained to input features or a training dataset, and one can utilize the generated xconcepts across multiple datasets and tasks.

We represent the quadruple $<I, c_{pred}, c_{alt}>$ as a human's query Q that will be answered by showing a fault-line explanation Ψ. We use Σ to represent all the xconcepts mined from χ. The xconcepts specific to the class c_{pred} and c_{alt} are represented as Σ_{pred} and Σ_{alt}, respectively. Our strategy will be to first identify the xconcepts Σ_{pred} and Σ_{alt} and then generate a fault-line explanation by finding a minimal set of xconcepts from Σ_{pred} and Σ_{alt}. Formally, the objective is to find a fault-line that maximizes the posterior probability:

$$\underset{\Psi}{\operatorname{argmax}} P\left(\Psi, \Sigma_{pred}, \Sigma_{alt}, \Sigma \mid Q\right) \qquad \text{(Equation 2)}$$

Mining xconcepts

We first compute $P(\Sigma \mid \chi, M)$ by identifying a set of semantically meaningful superpixels from every image and then performing clustering such that all the superpixels in a cluster are semantically similar. Each of these clusters represents an xconcept. We then identify class specific xconcepts i.e., $P(\Sigma_{pred} \mid \Sigma, \chi, I, c_{pred}, M)$ and $P(\Sigma_{alt} \mid \Sigma, \chi, I, c_{alt}, M)$.

Finding semantically meaningful super-pixels as xconcepts. Figure 8 shows the overall algorithm for computing $P(\Sigma \mid \chi, M)$. As deeper layers of the CNN capture richer semantic aspects of the image, we construct the xconcepts by making use of feature maps from the last convolution layer. Let f denote the feature extractor component of the CNN and g denote the classifier component of the CNN that takes the output of f and predicts log-probabilities over output classes $\mathbf{Y}$. We denote the m feature maps produced at layer L of the CNN as $A^{m,L} = \{a^L \mid a^L = f(I)\}$ which are of width u and height v. We consider each feature map as an instance of an xconcept and obtain its localization map (i.e., super-pixels of each feature map). To produce the localization map, we use Grad-CAM (Selvaraju et al., 2017a) to compute the gradients of y^c with respect to the feature maps $A^{m,L}$ and are then spatially pooled using Global Average Pooling (GAP) to obtain the importance weights ($\alpha^c_{m,L}$) of a feature map m at layer L for a target class c:

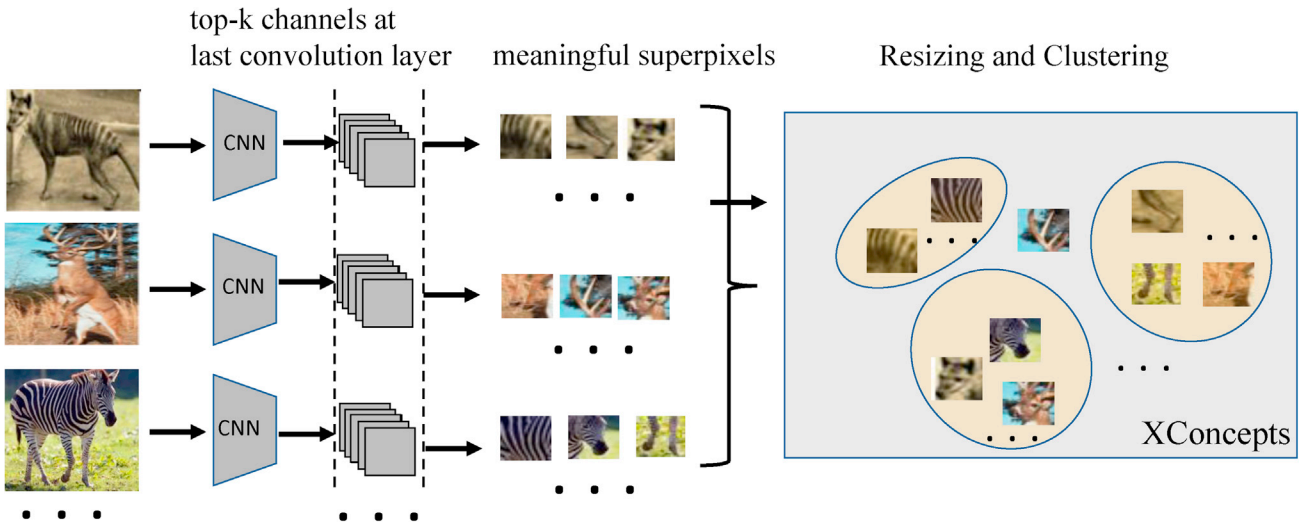


Figure 8. We consider feature maps from the last convolutional layer as instances of xconcepts and obtain their localization maps (i.e., superpixels) by computing the gradients of the output with respect to the feature maps

We select highly influential superpixels and then apply K-means clustering with outlier removal to group these superpixels into clusters where each cluster represents an xconcept.

$$\alpha_{m,L}^{c} = \frac{1}{Z}\sum_{i}\sum_{j}\frac{\partial y^{c}}{\partial A_{ij}^{m,L}} \quad \text{(Equation 3)}$$

Each element in the feature map $A^{m,L}$ is indexed by i, j, and $A_{ij}^{m,L}$ refers to the activation at location (i,j) of the feature map $A^{m,L}$. Z denotes the proportionality constant representing the total number of elements in $A^{m,L}$. Intuitively, Grad-CAM uses the gradient information flowing into the last convolutional layer of the convolution network to assign importance values to each neuron. In other words, the gradients flowing back are global-average-pooled over the width and height dimensions to compute the importance weights $\alpha_{m,L}^{c}$.

Using the importance weights, we select top p super-pixels for each class. Given that there are C output classes in the dataset χ, we get $p*C$ super-pixels from each image in the training dataset. We apply K-means clustering with outlier removal to group these super-pixels into G clusters where each cluster represents an xconcept (as shown in Figure 8). For clustering, we consider the spatial feature maps $f(I)$ instead of the super-pixels (i.e., actual image regions) themselves. We use the silhouette score value of a different range of clusters to determine the value of K.

Identifying class-specific xconcepts. For each output class c, we learn the most common xconcepts that are highly influential in the prediction of that class over the entire training dataset χ. We use the TCAV technique (Kim et al., 2018) to identify these class-specific xconcepts. Specifically, we construct a vector representation of each xconcept, called a CAV (denoted as v_X), by using a direction normal to a linear classifier trained to distinguish between the xconcept activations from the random activations. We then compute directional derivatives ($S_{c,X}$) to produce estimates of how important the concept X was for a CNN's prediction of a target class c, e.g., how important the xconcept stripedness is for predicting the zebra class.

$$S_{c,X} = \nabla g_c(f(I)) \cdot v_X \quad \text{(Equation 4)}$$

where g_c denotes the classifier component of the CNN that takes the output of f and predicts log-probability of output class c. Note that directional derivatives represent the derivative of logit values with respect to activations at the layer of interest, which helps in quantifying the model prediction's sensitivity to a xconcept. We argue that these class-specific xconcepts facilitate in generating meaningful explanations by pruning out incoherent xconcepts. For example, the xconcepts such as wheel and wings are irrelevant in explaining why the network's prediction is a *zebra* and not a *cat*.

Fault-line generation

In this subsection, we describe our approach to generate a fault-line explanation using the class-specific xconcepts. Let us consider that n_{pred} and n_{alt} xconcepts have been identified for output classes c_{pred} and c_{alt}, respectively, i.e., $|\Sigma_{pred}| = n_{pred}$ and $|\Sigma_{alt}| = n_{alt}$. We denote CAVs of the n_{pred} xconcepts belonging to the class c_{pred} as $v_{pred} = \{v^i_{pred},\ i = 1, 2, \ldots, n_{pred}\}$ and CAVs of the n_{alt} xconcepts belonging to the class c_{alt} as $v_{alt} = \{v^i_{alt},\ i = 1, 2, \ldots, n_{alt}\}$. We formulate finding a fault-line explanation as the following optimization problem:

$$\begin{aligned} &\underset{\delta_{pred},\delta_{alt}}{\text{minimize}}\ \alpha D(\delta_{pred}, \delta_{alt}) + \beta||\delta_{pred}||_1 + \lambda||\delta_{alt}||_1; \\ &D(\delta_{pred}, \delta_{alt}) = max\{g^{pred}(I') - g^{alt}(I'), -\tau\}; \\ &I' = A^{m,L} \circ v^{\top}_{pred}\delta_{pred} \circ v^{\top}_{alt}\delta_{alt}; \\ &\delta^i_{pred} \in \{-1, 0\},\ \delta^i_{alt} \in \{0, 1\}\ \forall i \text{ and } \alpha, \beta, \lambda, \tau \geq 0. \end{aligned} \qquad \text{(Equation 5)}$$

We elaborate on the role of each term in Equation 5 as follows. Our goal here is to derive a fault-line explanation that gives us the minimal set of xconcepts from Σ_{pred} and Σ_{alt} that will alter the model prediction from c_{pred} to c_{alt}. Intuitively, we try creating new images (I') by removing xconcepts in Σ_{pred} from I and adding xconcepts in Σ_{alt} to I until the classification result changes from c_{pred} to c_{alt}. To do this, we do not directly perturb the original image but change the activations obtained at the last convolutional layer $A^{m,L}$ instead. It may be noted that our goal is not to produce realistic images I'. We instead pick the most influential xconcepts by directly modifying the activation maps at a convolution layer (it is a very difficult task to produce realistic resulting images for datasets that have a diverse set of target classes).

To perturb the activations, we take the Hadamard product ($\circ$) between the activations ($A^{m,L}$), $v^{\top}_{pred}\delta_{pred}$, and $v^{\top}_{alt}\delta_{alt}$. The difference between the new logit scores for c_{pred} (i.e., $g^{pred}(I')$) and c_{alt} (i.e,. $g^{alt}(I')$) is controlled by the parameter τ.

For any given confidence $\tau > 0$, the loss function $D(\delta_{pred}, \delta_{alt})$ is minimized when the new logit scores for c_{pred} i.e., $g^{pred}(I')$ is smaller than logits for c_{alt} i.e,. $g^{alt}(I')$ by at least τ. Similar to (Dhurandhar et al., 2018), the terms $\beta||\delta_{pred}||_1$, $\lambda||\delta_{alt}||_1$ in the optimization are introduced as L_1 regularizers to select sparse features. We apply a projected fast iterative shrinkage-thresholding algorithm (FISTA) (Beck and Teboulle, 2009; Dhurandhar et al., 2018) for solving the above optimization problem. We outline our method in Algorithm 1.

Algorithm 1. Generating Fault-Line Explanations

input image I, classification model M, predicted class label c_{pred}, alternate class label c_{alt}, and training dataset χ

1. Find semantically meaningful superpixels in χ,

$$\alpha^c_{m,L} = \frac{1}{Z}\sum_i\sum_j \frac{\partial y^c}{\partial A^{m,L}_{ij}}$$

2. Apply K-means clustering on superpixels and obtain xconcepts (Σ).
3. Identify class specific xconcepts (Σ_{pred} and Σ_{alt}) using TCAV,

$$S_{c,X} = \nabla g_c(f(I)) \cdot v_X$$

4. Solve Equation 5 to obtain fault-line Ψ,

$$\Psi \leftarrow \min_{\delta_{pred},\delta_{alt}} \alpha D(\delta_{pred}, \delta_{alt}) + \beta||\delta_{pred}||_1 + \lambda||\delta_{alt}||_1$$

return Ψ.

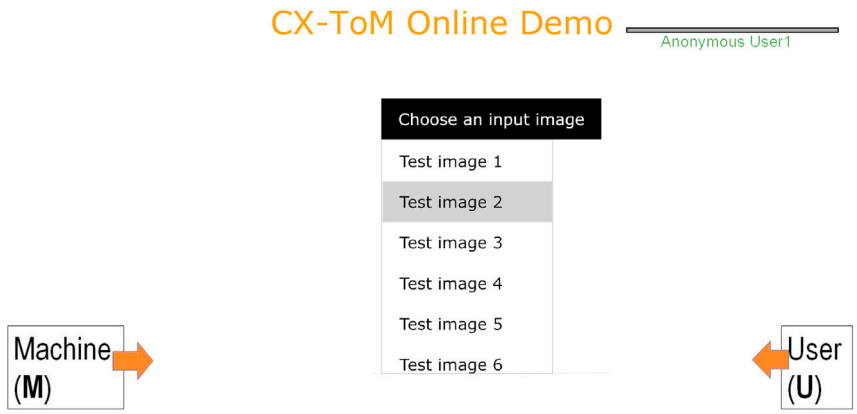


Figure 9. User interaction with CX-ToM in dialogue to learn user preferences/utilities
User is first asked to select an input image.

CX-ToM framework

In CX-ToM, we integrate both the ToM (importance of ToM) and fault-lines (fault-lines as an alternative to attention based explanations) into one single explanation framework. Essentially, CX-ToM performs fault-line selection using ToM. Given an input image and two output categories, fault-lines show the most important features or attributes that influence a model's decision in classifying the image as one among the two output categories. In most cases, there exist several thousands of output categories and it is impossible for the human user to verify the model's reasoning and behavior by constructing a fault-line between all the possible pairs of output categories. Therefore, we learn an optimal policy to automatically select the most important pair for constructing fault-line explanations that helps human users to quickly understand the model's strengths or weaknesses. This eliminates the need for the human user to see a large number of fault-lines before understanding the model's behavior.

We cast this as a reinforcement learning (RL) problem where CX-ToM interacts with several human users in a dialogue to learn user preferences/utilities that help them to understand the model in fewer dialogues (i.e., less number of fault-lines). We express reward in terms of a user feedback and the number of dialogue turns (less the number of dialogues, higher is the reward). Figures 9 and 10 show the user interaction interface. In the interaction, the user is first asked to select an image from a list of randomly drawn images from the training data (we only consider image classes for which we extracted xconcepts). After the input image is selected, the user is then asked to select an alternate class to which the model needs to modify its decision through fault-lines. We show the list of alternate classes through a dropdown list. The entries in this dropdown are dynamically loaded based on the model's current state of the RL policy. CX-ToM shows the optimal fault-line to the user and tracks the sequence of user's preferences through the RL policy. After showing the fault-line, the CX-ToM assesses the user's understanding of the model's important features in classifying the input image. If the user correctly answers the question, the reward is considered positive, otherwise negative. The RL policy is updated after every 15 dialogue interactions.

The RL policy is learned by a standard recurrent neural network, called Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). In this paper we use a 2-layer LSTM parameterized by θ. Thus, the goal of the policy learning is to estimate the LSTM parameters θ. We use actor-critic with experience replay for policy optimization (Wang et al., 2017). The training objective is to find parameterized policy $\pi(a_i|s_i;\theta)$ that maximizes the expected reward $J(\theta)$ over all possible fault-line sequences given a starting state. The state of the RL policy (s) captures whether an image class is already selected in the dialogue to generate a fault-line for the input image. Our goal is to learn the best user preferred alternate image classes for each prediction class. Similarly, the action space (a) constitutes the set of all image classes. The gradient of the objective function has the following form:

$$\nabla_\theta J(\theta) = \mathbb{E}[\nabla_\theta \log \pi_\theta(a_i|s_i;\theta) A(s_i, a_i)] \quad \text{(Equation 6)}$$

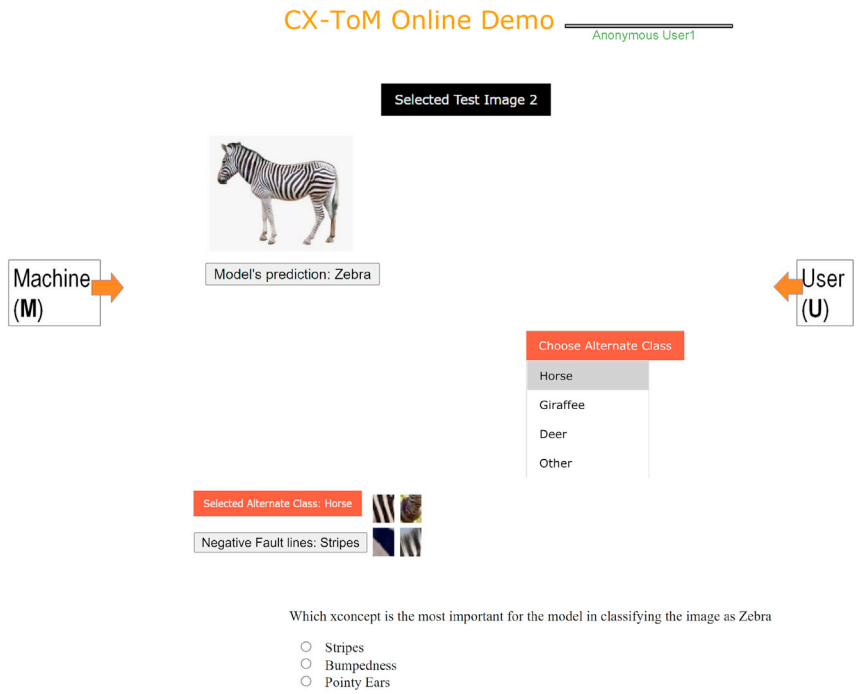


Figure 10. After the input image is selected by the user, user is then asked to select an alternate class to modify the model's decision

User is then shown a question to assess his/her understanding of the model's important features in classifying the input image. If the user correctly answers the question, the reward is considered positive, otherwise negative.

where $A(s_i, a_i) = Q(s_i, a_i) - V(s_i)$ is the advantage function (Sutton et al., 2000). $Q(s_i,a_i)$ is the standard Q-function, and $V(s_i)$ is the value (baseline) function aimed at reducing the variance of the estimated gradient. Intuitively, the above policy optimization can be seen as the task of learning to select the sequence of responses (actions) at each turn which maximizes the long-term objective defined by the reward function. The learning agent uses the value of the value function to update the optimal policy function. The policy function represents the probabilistic distribution of the action space. In other words, the learning agent determines the conditional probability that the agent chooses the action a when in state s. We use the same specifications of $Q(s_i,a_i)$ and $V(s_i)$ as in (Sutton et al., 2000). As in (Sutton et al., 2000), we sample the dialogue experiences randomly from the replay pool for training.

EXPERIMENTS

We conducted extensive human subject experiments to quantitatively and qualitatively assess the effectiveness of the proposed CX-ToM explanations in helping expert human users and nonexpert human users understand the internal workings of the underlying model. We chose an image classification task for our experiments (although the proposed approach is generic and can be applied to any task). We use the following metrics (Hoffman, 2017; Hoffman et al., 2018) to compare our method with the baselines (we empirically observed that the metrics Justified Trust and Explanation Satisfaction are effective in evaluating the core objective of XAI, i.e., to evaluate whether the user's understanding of the model improves with explanations. These metrics are originally defined at a high-level in the work by (Hoffman et al., 2018) and we adapt them for the image classification task.).

1. ***Justified Trust*** (Quantitative Metric). Justified Trust is computed by evaluating the human's understanding of the model's (M) decision-making process. In other words, given an image, it evaluates whether the users could reliably predict the model's output decision. More concretely, let us consider that M predicts images in a set C correctly and makes incorrect decisions on the images in the set W. Justified trust is given as sum of the percentage of images in C that the human subject thinks M would correctly predict and the percentage of images in W that the human subject thinks M would fail to predict correctly.

2. ***Explanation Satisfaction (ES)*** (Qualitative Metric). We measure human subjects' feeling of satisfaction at having achieved an understanding of the machine in terms of usefulness, sufficiency, appropriate detail, confidence, and accuracy (Hoffman, 2017; Hoffman et al., 2018). We ask the subjects to rate each of these metrics on a Likert scale of 0–9.

We used the ILSVRC2012 dataset (Imagenet) (Russakovsky et al., 2015) and considered VGG-16 (Simonyan and Zisserman, 2014) as the underlying network model. We randomly chose 80 classes in the dataset for our experiments and identified 57 xconcepts using our algorithm (we manually removed noisy xconcepts and fault-lines. We couldn't find an automatic approach to filter them. We leave this for future exploration).

We recruited 150 human subjects from our institution's Psychology subject pool (these experiments were reviewed and approved by our institution's IRB). These subjects have no background in computer vision, deep learning or NLP and we considered them as nonexpert users. We recruited an additional 60 human subjects with a background in computer vision. These subjects are experienced in training an image classification model using CNN, and therefore we considered them as expert users.

We applied between-subject design and randomly assigned subjects into eleven groups. We perform this separately with an expert user pool and nonexpert user pool. Each group in the nonexpert pool is assigned 12 subjects and each group in the expert pool is assigned 5 subjects. Within each group, each subject will first go through a familiarization phase where the subjects become familiar with the underlying model through explanations (with 25 training images), followed by a testing phase where we apply our evaluation metrics and assess their understanding (on 8 test images) in the underlying model. We trained our ToM policy through the interactions with 15 subjects. In the testing phase, the human will be given only I and will not see c_{pred}, c_{alt}, and explanations, and we evaluate whether the human can correctly identify c_{pred} based on his/her understanding of the model gained in the familiarization phase. All our data and code will be made publicly available.

For the first group, called NO-X (short for no-explanation group), we show the model's classification output on all the 25 images in the familiarization phase but we do not provide any explanation for the model's prediction. For the subjects in groups two to nine, in addition to the model's classification output, we also provide explanations in the familiarization phase for the model's prediction generated using the following state-of-the-art XAI models, respectively: CAM (Zhou et al., 2016), Grad-CAM (Selvaraju et al., 2017a), LIME (Ribeiro et al., 2016), LRP (Bach et al., 2015), SmoothGrad (Smilkov et al., 2017), TCAV (Kim et al., 2018), CEM (Dhurandhar et al., 2018), and CVE (Goyal et al., 2019). For the subjects in the 10th group, we show the fault-line explanations without incorporating ToM policy. For the subjects in the 11th group, we show the fault-line explanations selected based on our trained ToM policy. It may be noted that, in the testing phase, the human will be shown only the image I and will not be provided c_{pred}, c_{alt}, and explanations.

RESULTS

Table 1 compares the Justified Trust (JT) and Explanation Satisfaction (ES) of all the groups in expert subject pool and nonexpert subject pool. As we can see, JT and ES values of attention map based explanations such as Grad-CAM, CAM, and SmoothGrad do not differ significantly from the NO-X baseline, i.e., attention based explanations are not effective at increasing human trust and reliance (we did not evaluate ES for NO-X group as these subjects are not shown any explanations). This finding is consistent with the recent study by (Jain and Wallace, 2019) which shows that attention is not an explanation. On the other hand, concept based explanation framework TCAV andcounterfactual explanation frameworks CEM and CVE performed significantly better than the NO-X baseline (in both expert and nonexpert pool). Our CX-ToM based explanations, which are both conceptual and counterfactual, significantly outperformed all the baselines. Note that, fault-lines with ToM policy performs better than randomly selecting a fault-line. Interestingly, expert users preferred LRP (JT = 51.1%) to LIME (JT = 42.1%) and nonexpert users preferred LIME (JT = 46.1%) to LRP (JT = 31.1%).

Furthermore, human subjects in our CX-ToM group, compared to all the other baselines, found that explanations are highly useful, sufficient, understandable, detailed, and are more confident in answering the questions in the testing phase. These findings verify our hypothesis that fault-line explanations with ToM policy are lucid and easy for both expert and nonexpert users to understand (interestingly, we did not find significant differences across all the groups in terms of response time in answering the questions.

Table 1. Quantitative (Justified Trust) and Qualitative (Explanation Satisfaction) comparison of CX-ToM with random guessing baseline, no explanation (NO-X) baseline, and other state-of-the-art XAI frameworks such as CAM, Grad-CAM, LIME, LRP, SmoothGrad, TCAV, CEM, and CVE

		Explanation satisfaction (±std)				
XAI framework	Justified trust (±std)	Confidence	Usefulness	Appropriate detail	Understandability	Sufficiency
Non-expert subject pool						
Random guessing	6.6%	NA	NA	NA	NA	NA
NO-X	21.4 ± 2.7%	NA	NA	NA	NA	NA
CAM (Zhou et al., 2016)	24.0 ± 1.9%	4.2 ± 1.8	3.6 ± 0.8	2.2 ± 1.9	3.2 ± 0.9	2.6 ± 1.3
Grad-CAM (Selvaraju et al., 2017a)	29.2 ± 3.1%	4.1 ± 1.1	3.2 ± 1.9	3.0 ± 1.6	4.2 ± 1.1	3.2 ± 1.0
LIME (Ribeiro et al., 2016)	46.1 ± 1.2%	5.1 ± 1.8	4.2 ± 1.6	3.9 ± 1.1	4.1 ± 2.0	4.3 ± 1.6
SHAP (Lundberg and Lee, 2017)	40.9 ± 2.0%	4.8 ± 3.0	3.9 ± 1.1	3.6 ± 1.9	3.8 ± 1.4	4.0 ± 2.3
LRP (Bach et al., 2015)	31.1 ± 2.5%	1.1 ± 2.2	2.8 ± 1.0	1.6 ± 1.7	2.8 ± 1.0	2.1 ± 1.8
SmoothGrad (Smilkov et al., 2017)	37.6 ± 2.9%	1.4 ± 1.0	2.2 ± 1.8	2.8 ± 1.0	3.1 ± 0.8	2.9 ± 0.8
TCAV (Kim et al., 2018)	49.7 ± 3.3%	3.6 ± 2.1	3.2 ± 1.8	3.3 ± 1.6	3.6 ± 2.1	3.9 ± 1.1
CEM (Dhurandhar et al., 2018)	51.0 ± 2.1%	4.1 ± 1.4	3.4 ± 1.4	3.1 ± 2.1	2.9 ± 0.9	3.3 ± 1.6
CVE (Goyal et al., 2019)	50.9 ± 3.0%	3.8 ± 1.9	3.1 ± 0.9	3.6 ± 2.1	4.1 ± 1.2	4.2 ± 1.2
Fault-lines without ToM	69.1 ± 2.1%	6.2 ± 1.2	6.6 ± 0.7	7.2 ± 0.9	7.1 ± 0.6	6.2 ± 0.8
CX-ToM (fault-lines with ToM)	72.1 ± 1.1%	6.9 ± 0.8	6.5 ± 0.9	7.8 ± 1.2	7.7 ± 0.2	6.9 ± 0.6
Expert subject pool						
NO-X	28.1 ± 4.1%	NA	NA	NA	NA	NA
CAM (Zhou et al., 2016)	37.1 ± 3.9%	3.2 ± 1.8	3.3 ± 1.4	3.1 ± 2.1	3.1 ± 1.8	2.9 ± 1.9
Grad-CAM (Selvaraju et al., 2017a)	39.1 ± 2.1%	3.7 ± 1.2	3.1 ± 2.2	2.7 ± 1.9	3.7 ± 1.1	3.4 ± 1.6
LIME (Ribeiro et al., 2016)	42.1 ± 3.1%	3.1 ± 2.2	3.0 ± 1.2	2.8 ± 1.9	3.1 ± 2.2	2.8 ± 1.7
LRP (Bach et al., 2015)	51.1 ± 3.1%	3.2 ± 4.1	3.5 ± 1.6	4.2 ± 1.5	4.3 ± 1.0	3.9 ± 0.9
SmoothGrad (Smilkov et al., 2017)	40.7 ± 2.1%	3.1 ± 1.0	2.9 ± 1.2	3.8 ± 1.5	3.3 ± 1.1	3.1 ± 1.0
TCAV (Kim et al., 2018)	55.1 ± 3.3%	3.9 ± 2.8	3.6 ± 1.6	4.1 ± 1.3	4.9 ± 1.2	3.9 ± 0.8
CEM (Dhurandhar et al., 2018)	61.1 ± 2.2%	4.8 ± 1.6	3.7 ± 1.6	4.0 ± 1.2	3.7 ± 1.0	4.0 ± 1.1
CVE (Goyal et al., 2019)	64.5 ± 3.7%	4.1 ± 2.3	3.9 ± 1.5	4.6 ± 1.5	4.5 ± 1.4	3.9 ± 1.2
Fault-lines without ToM	70.5 ± 1.3%	5.7 ± 1.1	4.9 ± 0.8	5.8 ± 1.2	6.9 ± 1.1	6.4 ± 1.0
CX-ToM (fault-lines with ToM)	74.5 ± 0.7%	6.1 ± 0.8	5.3 ± 0.4	5.9 ± 1.2	7.1 ± 0.8	6.9 ± 0.7

We did an additional study with four subjects in each of the groups to verify this and again found similar results. We leave this observation for future exploration).

Comparison with SHAP baseline

We conduct an additional study to compare how the proposed method compares with SHAP approach (Lundberg and Lee, 2017). SHAP, using shapley values, explains output predictions of a model for given input by computing the contribution of each feature to the prediction. Specifically, we use GradientExplainer (https://shap.readthedocs.io/en/latest/image_examples.html) implementation to compute SHAP explanations for the image classifier. We experiment with additional 12 human subjects (nonexperts) to measure Justified Trust and Explanation Satisfaction. As shown in Table 1, SHAP underperforms compared to CX-ToM and shows similar performance to LIME. This is expected as both LIME and SHAP are attribution based techniques.

Gain in justified trust over time

We hypothesized that subjects' justified trust in the CNN model might improve over time. This is because it can be harder for humans to fully understand the machine's underlying inference process in one single session. Therefore, we conducted an additional experiment with eight human subjects (nonexperts) for each group where the subjects' reliance was measured after every session. Note that each session consists of a familiarization phase followed by a testing phase. The results are shown in Figure 11. As we can see, the subjects' JT in

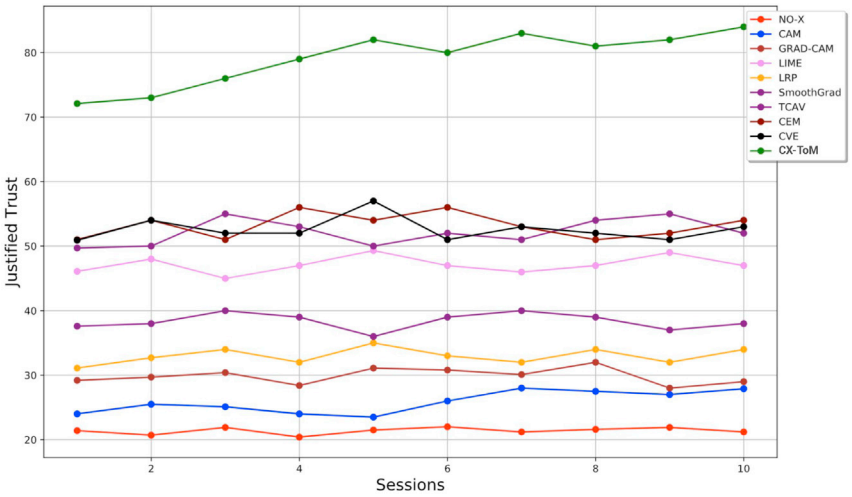


Figure 11. Gain in Justified Trust over time

CX-ToM group increased at a higher-rate compared to other baselines. However, we did not find any significant increase in JT after the fifth session across all the groups. This is consistent with our expectation that it is difficult for humans to focus on a task for longer periods (in the future, we also intend to experiment with subjects by arranging sessions over days or weeks instead of having continuous back to back sessions). It should be noted that the increase in JT with attention map based explanations such as Grad-CAM and CAM is not significant. This finding again demonstrates that attention maps are not effective to improve human trust.

Subjective evaluation of justified trust

In addition to the quantitative evaluation of the justified trust, we also collect subjective trust values (on a Likert scale of 0–9) from the subjects. This helps in understanding to what extent the users think they trust the model. The results are shown in Figure 12. As we can see, these results are consistent with our quantitative trust measures except that qualitative trust in Grad-CAM, CAM, and, SmoothGrad is lower compared to the NO-X group.

Case study

Figure 13 shows examples of the xconcepts (cropped and rescaled for better view) identified using our approach. As we can see, our method successfully extracts semantically coherent xconcepts such as *pointed curves* of deers, *stripedness* of zebras, and *woolliness* of deerhounds from the training dataset. In addition the fault-lines generated by our method correctly identify the most critical xconcepts that can alter the classification result from c_{pred} to c_{alt}. For example, consider the image of deerhound shown in Figure 13. Our fault-line explanation suggests removing *woolliness* and adding *black and white pattern* to alter the model's classification on the image from deerhound to greyhound.

Additional experiments

In this section, we further assess the effectiveness of the proposed CX-ToM framework using more diverse and recent models as the underlying convolution neural network.

ResNet50

ResNet (He et al., 2016) is a relatively deeper convolution neural network than VGG-16. It incorporates skip connections and batch normalization which greatly improves model's generalization capability and performance. More specifically, each ResNet block is 3 layer deep consisting of 1 × 1, 3 × 3, 1 × 1 convolutions, respectively. The 1 × 1 convolution layers are useful in reducing and then restoring the dimensions. Finally, the average pooling is performed and ended itwith a fully connected layer.

We apply our CX-ToM framework to ResNet. As discussed in fault-lines as an alternative to attention based explanations and Algorithm 1, we mine xconcepts from ResNet by producing localization maps. The

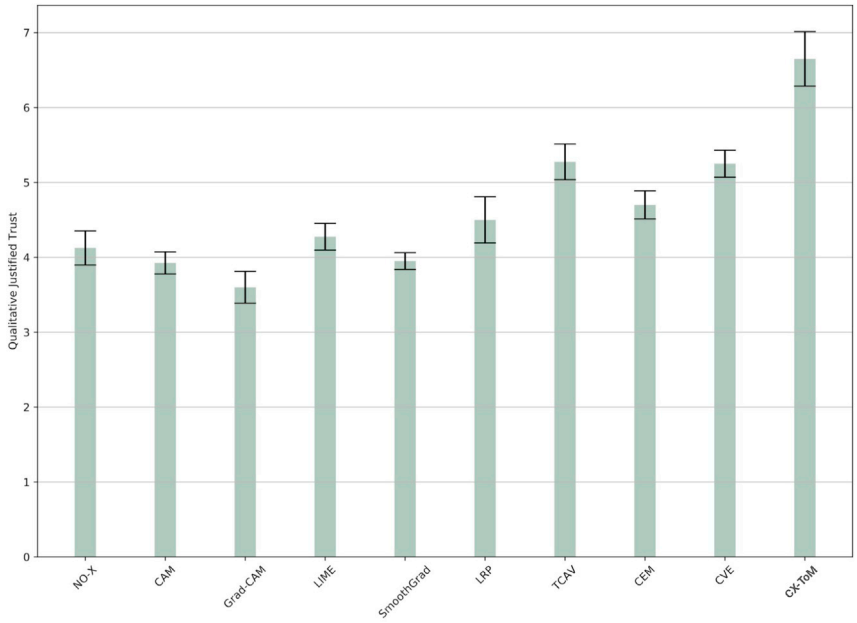


Figure 12. Average Qualitative Justified Trust (on a Likert scale of 0–9)
Error bars denote standard errors of the means.

average pooling layer is used to obtain importance weights of a feature map at a layer *L* for a given target class. We obtain class-specific xconcepts using concept activation vectors. Finally fault-lines are generated by solving the optimization problem in Equation (5).

We use the ILSVRC2012 dataset for our experiments. We compare our approach against the following baselines: Grad-CAM (Selvaraju et al., 2017a), LIME (Ribeiro et al., 2016), TCAV (Kim et al., 2018), and CVE (Goyal et al., 2019). Similar to our experiments with VGG-16, we use the metrics Justified Trust (JT) and Explanation Satisfaction (ES) to compare our approach with baselines. We recruited human subjects from our institution's Psychology subject pool. We apply between-subject design and randomly assigned subjects into six groups. Each group in the nonexpert pool is assigned 4 subjects and each group in the expert pool is assigned 2 subjects. We have identified 15 xconcepts and closely followed the experiment setup and design used in our experiments on the VGG-16 model.

Table 2 summarizes the JT and ES results of all the six groups. Similar to the results with VGG-16, trust improvements with Grad-CAM on both expert and nonexpert pools is the least compared to other baselines. Among the baselines, TCAV is the best performing model, implying that concept level explanations are relatively more scalable to deeper networks than attention based explanations. Our CX-ToM based framework shows significant improvements over the TCAV baseline. The subjective evaluation of JT and ES shows in Figure 14 further validate our hypotheses.

PAC networks

Recently a pixel-adaptive convolution network called PAC (Su et al., 2019) is proposed to address the content-agnostic limitations of traditional CNNs. Specifically, in traditional CNNs, the same convolutional filter banks are applied to all the input images irrespective of their content. However, image content varies substantially across the input images, and therefore, applying content-agnostic filter banks may not be optimal for all image types as well as different pixels in an image. In PAC networks, content-adaptive convolution operations are performed where a standard spatially invariant convolution filter *W* is multiplied with a adapting kernel *K*. These networks are shown to be effective in a wide range of computer vision problems such as depth and optical flow upsampling tasks (Su et al., 2019).

We apply our CX-ToM framework to PACNet. Using Algorithm 1, we extract xconcepts from PACNet and obtain class-specific xconcepts using concept activation vectors. Finally fault-lines are generated by

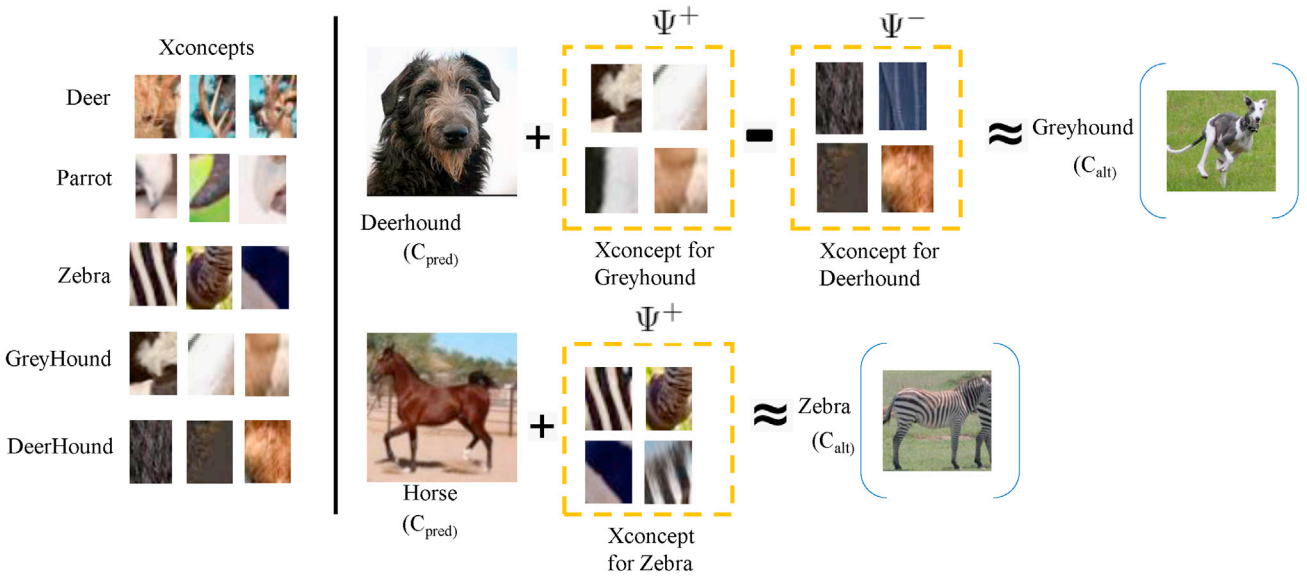


Figure 13. Examples of xconcepts (Left) and fault-line explanations (Right) identified by our method

solving the optimization problem in Equation 5. We use ILSVRC2012 dataset and consider the following baselines: Grad-CAM (Selvaraju et al., 2017a), LIME (Ribeiro et al., 2016), TCAV (Kim et al., 2018), and CVE (Goyal et al., 2019). We use the metrics Justified Trust (JT) and Explanation Satisfaction (ES) to compare our approach with baselines. We recruited human subjects from our institution's Psychology subject pool. We apply between-subject design and randomly assigned subjects into six groups. Each group in the nonexpert pool is assigned 4 subjects and each group in the expert pool is assigned 2 subjects. We have identified 18 xconcepts and closely followed the experiment setup and design used in our experiments on VGG-16 and ResNet models.

We present the JT and ES of all the six groups in Table 3. As we can see, trust improvements with Grad-CAM on PACNet are relatively lower compared to VGG-16 and ResNet. This indicates that attention based explanations need more fine-tuning on the non-traditional CNN architectures. TCAV and CVE clearly outperform other baselines. Our CX-ToM based framework shows relatively significant improvements over all the baselines indicating that our approach generalizes well to the recent CNN models. The subjective evaluation results of JT and ES shown in Figure 15 are consistent with our quantitative results.

Competency testing

We perform a competency testing experiment where we train two different CNNs, namely, AlexNet and ResNet-50. It may be noted that ResNet-50 is known to be more reliable and accurate than AlexNet. We show the predictions and the explanations from each of the two networks to the subjects (4 subjects in each of the above groups) and ask them to compare the reliability (competency) of the models relative to each other. We record the subjects' confidence scores in their answers on a Likert scale of 0–9. We chose only those images for which both models made the same prediction as ground truth. The assumption here is that an effective and useful explanation helps the subject to distinguish between a reliable model and an unreliable model easily. We find that human subjects, who are shown CX-ToM explanations, are able to identify the more accurate and reliable classifier (i.e., ResNet-50) with high confidence (average confidence score = 7.7). Human subjects who are shown explanations based on Grad-CAM, CEM, and TCAV also identified ResNet-50 as more reliable than AlexNet. However, they are not confident in their answers (avg. confidence scores are 2.6 (Grad-CAM), 4.9 (TCAV), and 4.2 (CEM)). Subjects in the remaining groups failed to identify the more reliable classifier.

Computational cost

We run all components of our framework on one RTX 2080ti GPU. The extraction of super-pixels using Grad-CAM, discussed in fault-lines as an alternative to attention based explanations, takes about 17 h

Table 2. Justified Trust and Explanation Satisfaction Results of CX-ToM and baselines on ResNet-50

XAI framework	Justified trust (±std)	Explanation satisfaction (±std)				
		Confidence	Usefulness	Appropriate detail	Understandability	Sufficiency
Non-expert pool						
Grad-CAM (Selvaraju et al., 2017a)	21.6 ± 2.8%	3.2 ± 1.5	3.2 ± 1.6	2.7 ± 2.8	3.0 ± 2.0	2.9 ± 0.9
LIME (Ribeiro et al., 2016)	26.9 ± 3.5%	3.3 ± 2.5	3.1 ± 2.1	3.7 ± 1.9	3.1 ± 1.8	4.0 ± 1.3
TCAV (Kim et al., 2018)	42.2 ± 2.6%	4.1 ± 2.7	3.2 ± 2.4	3.8 ± 1.9	4.0 ± 1.5	3.5 ± 1.8
CVE (Goyal et al., 2019)	38.1 ± 3.5%	2.7 ± 2.5	2.6 ± 1.5	3.0 ± 2.0	3.2 ± 1.1	3.2 ± 1.9
Fault-lines without ToM	54.2 ± 2.4%	6.1 ± 1.7	5.9 ± 1.2	6.6 ± 1.5	6.4 ± 0.9	6.2 ± 1.1
CX-ToM (fault-lines with ToM)	58.3 ± 1.8%	6.3 ± 1.8	6.2 ± 1.6	6.9 ± 1.1	7.2 ± 0.8	7.2 ± 1.6
Expert pool						
Grad-CAM (Selvaraju et al., 2017a)	20.1 ± 1.8%	2.5 ± 2.2	2.5 ± 1.8	1.7 ± 1.9	3.0 ± 1.9	3.0 ± 1.2
LIME (Ribeiro et al., 2016)	25.4 ± 2.7%	3.0 ± 1.6	3.2 ± 2.9	3.8 ± 2.1	2.6 ± 1.0	2.5 ± 2.9
TCAV (Kim et al., 2018)	46.0 ± 2.4%	3.5 ± 1.4	3.8 ± 1.7	3.6 ± 2.2	3.8 ± 2.1	4.0 ± 1.9
CVE (Goyal et al., 2019)	43.1 ± 3.1%	3.2 ± 2.3	3.2 ± 0.9	3.0 ± 1.8	3.0 ± 1.3	3.4 ± 1.8
Fault-lines without ToM	54.9 ± 1.6%	6.2 ± 2.1	6.0 ± 1.2	5.3 ± 1.6	6.0 ± 1.5	5.9 ± 1.5
CX-ToM (fault-lines with ToM)	56.0 ± 1.9%	5.8 ± 1.6	6.1 ± 1.0	6.1 ± 1.0	7.0 ± 1.5	7.0 ± 1.2

(15 min per 100 images in the training dataset). The clustering of these super-pixels is relatively fast and completes within 3 h to extract the 57 xconcepts from 80 image classes. Using TCAV technique to learn CAVs takes about 15 h on RTX 2080ti and then identifying the directional derivatives takes about 2 h for the extracted 57 xconcepts (discussed in fault-lines as an alternative to attention based explanations). Finally, the optimization step to select the appropriate fault-line takes about 40 s per image.

Conclusions

In this paper, we introduced a new explainable AI (XAI) framework, CX-ToM, based on Theory of Mind and fault-lines. We argue that, because of their iterative, conceptual, and counterfactual nature, CX-ToM based explanations are lucid, clear, and easy for humans to understand. We proposed a new method to automatically mine explainable concepts from a given training dataset and to derive fault-line explanations. Moreover, we show that estimating the human's understanding of the CNN model using Theory-of-Mind helps in generating more appropriate fault-lines. Using qualitative and quantitative evaluation metrics, we demonstrated that CX-ToM significantly outperforms baselines in improving human understanding of the underlying classification model.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
- ADDITIONAL RESOURCES

ACKNOWLEDGMENTS

The authors thank Zhao Weng (UCLA), Sparsh Arora (UCLA), Yujia Peng (UCLA), Debleena Sengupta (UCLA), Lawrence Chen (UCLA), and Yuhe Gao (UCLA) for their help with conducting human studies and experiment setup, Prof. Devi Parikh (Georgia Tech), Prof. Dhruv Batra (Georgia Tech), Prof. Stefan Lee (OSU), for helpful discussions and useful feedback.

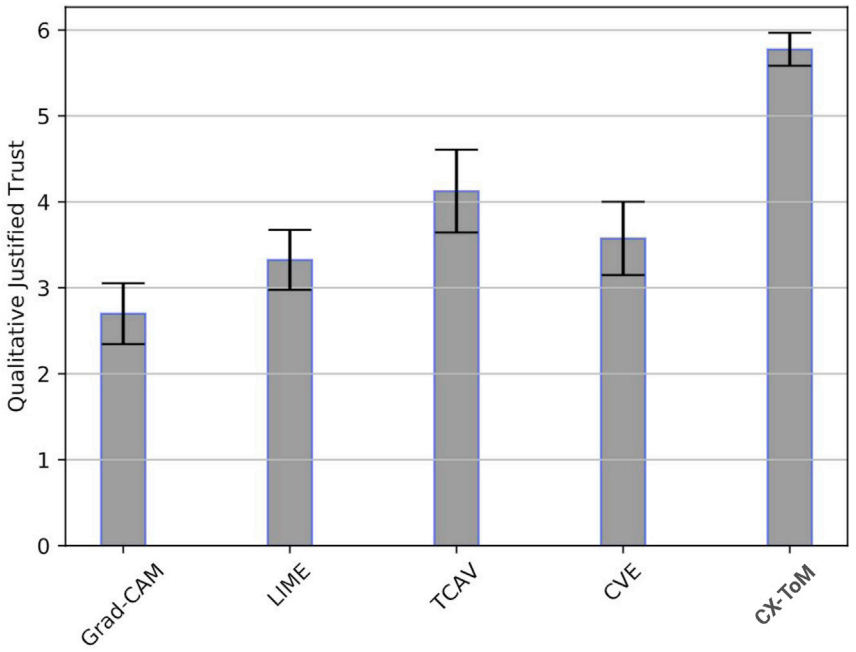


Figure 14. Average Subjective Justified Trust (on a Likert scale of 0–9) on ResNet-50

AUTHOR CONTRIBUTIONS

A.A. is the main lead author of the work. A.A. implemented the explanation learning framework and also designed, conducted human study experiments, and wrote the paper. K.W. provided feedback in the initial stages of designing the learning framework. He is also a key part of the experiment design section. C.L., S.S., and H.L. provided valuable feedback in improving the clarity of the paper and also helped in designing and conducting human studies. S.T., J.C., and S.C. played a major role in shaping up the overall explanation framework and also helped in designing experiments and in writing the work.

Table 3. Justified Trust and Explanation Satisfaction Results of CX-ToM and baselines on PACNet

		Explanation satisfaction (±std)				
XAI framework	Justified trust (±std)	Confidence	Usefulness	Appropriate detail	Understandability	Sufficiency
Non-expert pool						
Grad-CAM (Selvaraju et al., 2017a)	15.2 ± 1.5%	2.4 ± 1.8	2.6 ± 1.2	2.5 ± 1.5	2.9 ± 1.7	3.0 ± 1.2
LIME (Ribeiro et al., 2016)	22.2 ± 2.4%	3.1 ± 2.2	2.7 ± 2.0	3.5 ± 1.9	2.7 ± 1.2	3.8 ± 1.6
TCAV (Kim et al., 2018)	40.1 ± 2.2%	3.9 ± 1.7	3.6 ± 1.1	4.1 ± 2.5	4.0 ± 1.2	3.6 ± 1.8
CVE (Goyal et al., 2019)	41.5 ± 3.2%	3.1 ± 1.5	3.3 ± 1.0	3.8 ± 2.1	3.8 ± 2.0	3.9 ± 1.2
Fault-lines without ToM	53.8 ± 1.9%	6.3 ± 2.0	5.6 ± 1.1	6.1 ± 1.9	5.9 ± 0.6	6.6 ± 1.6
CX-ToM (fault-lines with ToM)	54.8 ± 2.0%	6.2 ± 2.0	6.5 ± 1.8	6.2 ± 1.0	7.0 ± 1.9	6.8 ± 1.9
Expert pool						
Grad-CAM (Selvaraju et al., 2017a)	16.8 ± 1.9%	2.3 ± 1.2	2.9 ± 1.4	2.0 ± 1.9	3.1 ± 1.5	3.2 ± 2.2
LIME (Ribeiro et al., 2016)	23.7 ± 2.0%	2.9 ± 1.3	3.2 ± 2.5	3.0 ± 2.1	2.5 ± 1.6	2.9 ± 2.0
TCAV (Kim et al., 2018)	38.6 ± 3.1%	3.9 ± 1.3	3.2 ± 1.5	3.9 ± 2.0	4.0 ± 1.0	3.7 ± 1.1
CVE (Goyal et al., 2019)	39.1 ± 2.0%	3.5 ± 2.2	3.7 ± 1.6	3.2 ± 1.2	3.9 ± 1.1	3.0 ± 1.5
Fault-lines without ToM	57.0 ± 1.8%	6.0 ± 1.5	6.2 ± 1.7	5.8 ± 1.9	5.5 ± 1.1	6.1 ± 1.9
CX-ToM (fault-lines with ToM)	59.8 ± 1.6%	6.3 ± 1.1	6.5 ± 1.7	7.0 ± 1.5	6.7 ± 1.7	6.5 ± 1.0

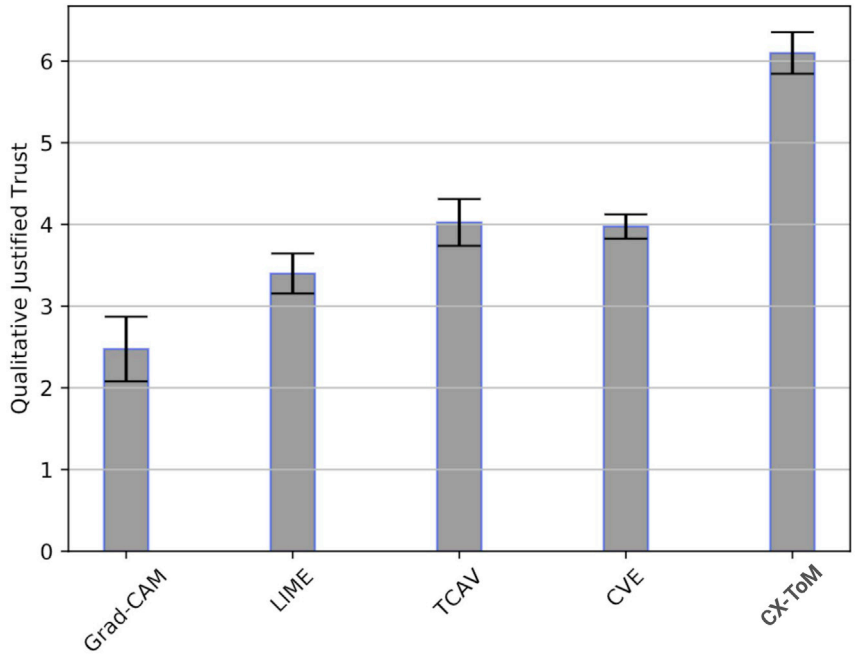


Figure 15. Average Subjective Justified Trust (on a Likert scale of 0–9) on PACNet

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 28, 2020
Revised: February 15, 2021
Accepted: December 5, 2021
Published: January 21, 2022

REFERENCES

Agarwal, S., Aggarwal, V., Akula, A.R., Dasgupta, G.B., and Sridhara, G. (2017). Automatic problem extraction and analysis from unstructured text in it tickets. IBM J. Res. Dev. *61*, 4–41.

Agarwal, S., Akula, A.R., Dasgupta, G.B., Nadgowda, S.J., Nayak, T.K., 2018. Structured representation and classification of noisy and unstructured tickets in service delivery. US Patent 10,095,779.

Akula, A.R. (2015). A Novel Approach towards Building a Generic, Portable and Contextual NLIDB System (International Institute of Information Technology Hyderabad).

Akula, A.R., and Zhu, S.C. (2019). Visual discourse parsing. arXiv, preprint abs/1903.02252. https://arxiv.org/abs/1903.02252.

Akula, A., Sangal, R., and Mamidi, R. (2013). A novel approach towards incorporating context processing capabilities in NLIDB system. In Proceedings of the Sixth International Joint Conference on Natural Language Processing, pp. 1216–1222.

Akula, A.R., Dasgupta, G.B., Nayak, T.K., 2018. Analyzing tickets using discourse cues in communication logs. US Patent 10,067,983.

Akula, A.R., Liu, C., Saba-Sadiya, S., Lu, H., Todorovic, S., Chai, J.Y., and Zhu, S.C. (2019a). X-tom: explaining with theory-of-mind for gaining justified human trust. arXiv, preprint arXiv:1909.06907.

Akula, A.R., Liu, C., Todorovic, S., Chai, J.Y., and Zhu, S.C. (2019b). Explainable AI as collaborative task solving. In CVPR Workshops, pp. 91–94.

Akula, A.R., Todorovic, S., Chai, J.Y., and Zhu, S.C. (2019c). Natural language interaction with explainable AI models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 87–90.

Akula, A.R., Gella, S., Al-Onaizan, Y., Zhu, S.C., and Reddy, S. (2020a). Words aren't enough, their order matters: on the robustness of grounding visual referring expressions. arXiv, preprint arXiv:2005.01655.

Akula, A.R., Wang, S., and Zhu, S. (2020b). Cocox: generating conceptual and counterfactual explanations via fault-lines. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, the Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, the Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020 (AAAI Press), pp. 2594–2601. https://aaai.org/ojs/index.php/AAAI/article/view/5643.

Akula, A., Gella, S., Wang, K., Zhu, S.C., and Reddy, S. (2021a). Mind the context: the impact of contextualization in neural module networks for grounding visual referring expressions. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 6398–6416.

Akula, A., Jampani, V., Changpinyo, S., and Zhu, S.C. (2021b). Robust visual reasoning via language guided neural module networks. Adv. Neural Inf. Process. Syst. *34*.

Akula, A.R., Changpinyo, B., Gong, B., Sharma, P., Zhu, S.C., Soricut, R., 2021c. CrossVQA: scalably generating benchmarks for systematically testing VQA generalization.

Akula, A.R., Dasgupta, G.B., Ekambaram, V., Narayanam, R., 2021d. Measuring effective utilization of a service practitioner for ticket

resolution via a wearable device. US Patent 10,929,264.

Alang, N., 2017. Turns out algorithms are racist.[online] the new republic.

Alvarez-Melis, D., and Jaakkola, T.S. (2018). On the robustness of interpretability methods. arXiv, preprint arXiv:1806.08049.

Augasta, M.G., and Kathirvalavakumar, T. (2012). Reverse engineering the neural networks for rule extraction in classification problems. Neural Process. Lett. *35*, 131–150.

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*, e0130140.

Bara, Christian-Paul, Wang, CH-Wang, and Chai, Joyce (2021). Mindcraft: Theory of Mind Modeling for Situated Dialogue in Collaborative Tasks. In Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021 (EMNLP).

Beck, A., and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. *2*, 183–202.

Berry, D.C., and Broadbent, D.E. (1987). Explanation and verbalization in a computer-assisted search task. Q. J. Exp. Psychol. *39*, 585–609.

Biran, O., and Cotton, C. (2017). Explanation and justification in machine learning: a survey. In IJCAI-17 Workshop on Explainable AI (XAI), p. 1.

Bivens, A., Ramasamy, H., Herger, L., Rippon, W., Fonseca, C., Pointer, W., Belgodere, B., Cornejo, W., Frissora, M., Ramakrishna, V. et al., 2017. Cognitive and contextual analytics for it services.

Bornstein, A.M. (2016). Is artificial intelligence permanently inscrutable? Nautilus.

Byrne, R.M. (2002). Mental models and counterfactual thoughts about what might have been. Trends Cogn. Sci. *6*, 426–431.

Byrne, R.M. (2017). Counterfactual thinking: from logic to morality. Curr. Dir. Psychol. Sci. *26*, 314–322.

Champlin, C., Bell, D., and Schocken, C. (2017). AI medicine comes to Africa's rural clinics. IEEE Spectr. *54*, 42–48.

Chancey, E.T., Bliss, J.P., Proaps, A.B., and Madhavan, P. (2015). The role of trust as a mediator between system characteristics and response behaviors. Hum. Factors *57*, 947–958.

Clark, H.H., and Schaefer, E.F. (1989). Contributing to discourse. Cogn. Sci. *13*, 259–294.

Clark, K., Khandelwal, U., Levy, O., and Manning, C.D. (2019). What does Bert look at? An analysis of Bert's attention. arXiv, preprint arXiv:1906.04341.

Darlington, K. (2013). Aspects of intelligent systems explanation. Univ. J. Control Autom. *1*, 40–51.

Dasgupta, G.B., Nayak, T.K., Akula, A.R., Agarwal, S., and Nadgowda, S.J. (2014). Towards auto-remediation in services delivery: context-based classification of noisy and unstructured tickets. In International Conference on Service-Oriented Computing (Springer), pp. 478–485.

Devin, S., and Alami, R. (2016). An implemented theory of mind to improve human-robot shared plans execution. In 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (IEEE), pp. 319–326.

Dhurandhar, A., Chen, P.Y., Luss, R., Tu, C.C., Ting, P., Shanmugam, K., and Das, P. (2018). Explanations based on the missing: towards contrastive explanations with pertinent negatives. In Advances in Neural Information Processing Systems, pp. 592–603.

Doshi-Velez, F., and Kim, B. (2017a). A roadmap for a rigorous science of interpretability. arXiv, preprint arXiv:1702.08608 150.

Doshi-Velez, F., and Kim, B. (2017b). Towards a rigorous science of interpretable machine learning. arXiv, preprint arXiv:1702.08608.

Douglas, W. (2007). Dialogical models of explanation. In ExaCt 2007, pp. 1–9.

Edmonds, M., Gao, F., Liu, H., Xie, X., Qi, S., Rothrock, B., Zhu, Y., Wu, Y.N., Lu, H., and Zhu, S.C. (2019). A tale of two explanations: enhancing human trust by explaining robot behavior. Sci. Robot. *4*. https://doi.org/10.1126/scirobotics.aay4663.

Erhan, D., Bengio, Y., Courville, A., and Vincent, P. (2009). Visualizing higher-layer features of a deep network. In Technical Report, *1341* (University of Montreal), p. 1.

Fisher, A., Rudin, C., and Dominici, F. (2018). Model class reliance: variable importance measures for any machine learning model class, from the "rashomon" perspective. arXiv, preprint arXiv:1801.01489.

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, pp. 3429–3437.

Friedman, J.H. (2001). Greedy function approximation: a gradient boosting machine. Ann. Stat. *29*, 1189–1232.

Goldman, A.I. (2012). Theory of mind. The Oxford Handbook of Philosophy of Cognitive Science (Oxford University Press).

Goodfellow, I.J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv, preprint arXiv:1412.6572.

Goodman, B., and Flaxman, S. (2017). European Union regulations on algorithmic decision-making and a "right to explanation". AI Mag. *38*, 50–57.

Gosselin, F., and Schyns, P.G. (2001). Bubbles: a technique to reveal the use of information in recognition tasks. Vis. Res. *41*, 2261–2271.

Goyal, Y., Wu, Z., Ernst, J., Batra, D., Parikh, D., and Lee, S. (2019). Counterfactual visual explanations. In ICML 2019.

Gulshan, V., Peng, L., Coram, M., Stumpe, M.C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., et al. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. JAMA *316*, 2402–2410.

Gupta, A., Akula, A., Dasgupta, G., Aggarwal, P., and Mohapatra, P. (2016). Desire: deep semantic understanding and retrieval for technical support services. In International Conference on Service-Oriented Computing (Springer), pp. 207–210.

Gupta, A., Akula, A., Malladi, D., Kukkadapu, P., Ainavolu, V., and Sangal, R. (2012). A novel approach towards building a portable nlidb system using the computational paninian grammar framework. In 2012 International Conference on Asian Language Processing (IEEE), pp. 93–96.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). The Elements of Statistical Learning. Springer Series in Statistics (Springer New York Inc.).

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Hendricks, L.A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., and Darrell, T. (2016). Generating visual explanations. In European Conference on Computer Vision (Springer), pp. 3–19.

Hendricks, L.A., Hu, R., Darrell, T., and Akata, Z. (2018). Generating counterfactual explanations with natural language. In ICML Workshop on Human Interpretability in Machine Learning.

Hernández-García, A., and König, P. (2018). Do deep nets really need weight decay and dropout? arXiv, preprint arXiv:1802.07042.

Hilton, D.J. (1990). Conversational processes and causal explanation. Psychol. Bull. *107*, 65.

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. arXiv, preprint arXiv:1503.02531.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. Neural Comput. *9*, 1735–1780.

Hoffman, R.R. (2017). A taxonomy of emergent trusting in the human–machine relationship. In Cognitive Systems Engineering: The Future for a Changing World (CRC Press), pp. 137–163.

Hoffman, R.R., Hancock, P.A., and Bradshaw, J.M. (2010). Metrics, metrics, metrics, part 2: universal metrics? IEEE Intell. Syst. *25*, 93–97.

Hoffman, R.R., and Klein, G. (2017). Explaining explanation, part 1: theoretical foundations. IEEE Intell. Syst. *32*, 68–73.

Hoffman, R.R., Mueller, S.T., and Klein, G. (2017). Explaining explanation, part 2: empirical foundations. IEEE Intell. Syst. *32*, 78–86. https://doi.org/10.1109/MIS.2017.3121544.

Hoffman, R.R., Mueller, S.T., Klein, G., and Litman, J. (2018). Metrics for explainable AI:

challenges and prospects. arXiv, preprint arXiv:1812.04608.

Jain, S., and Wallace, B.C. (2019). Attention is not explanation. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), arXiv:1902.10186. http://arxiv.org/abs/1902.10186.

Johnson, S., and Everingham, M. (2010). Clustered pose and nonlinear appearance models for human pose estimation. In BMVC, p. 5.

Kahneman, D., and Tversky, A. (1981). The simulation heuristic. In Technical Report (Stanford Univiversity California Department of Psychology).

Karpathy, A., Johnson, J., and Fei-Fei, L. (2015). Visualizing and understanding recurrent networks. arXiv, preprint arXiv:1506.02078.

Keil, F.C. (2006). Explanation and understanding. Annu. Rev. Psychol. *57*, 227–254.

Kim, B., Rudin, C., and Shah, J.A. (2014). The bayesian case model: a generative approach for case-based reasoning and prototype classification. In Advances in Neural Information Processing Systems, pp. 1952–1960.

Kim, B., Shah, J.A., and Doshi-Velez, F. (2015). Mind the gap: a generative approach to interpretable feature selection and extraction. In Advances in Neural Information Processing Systems, pp. 2260–2268.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., and Sayres, R. (2018). Interpretability beyond feature attribution: quantitative testing with concept activation vectors (TCAV). In International Conference on Machine Learning, pp. 2673–2682.

Kingma, D.P., and Ba, J. (2015). Adam: a method for stochastic optimization. In International Conference on Learning Representations (ICLR).

Kulesza, T., Burnett, M., Stumpf, S., Wong, W.K., Das, S., Groce, A., Shinsel, A., Bice, F., and McIntosh, K. (2011). Where are my intelligent assistant's mistakes? A systematic testing approach. In International Symposium on End User Development (Springer), pp. 171–186.

Kulesza, T., Stumpf, S., Burnett, M., Wong, W.K., Riche, Y., Moore, T., Oberst, I., Shinsel, A., and McIntosh, K. (2010). Explanatory debugging: supporting end-user debugging of machine-learned programs. In 2010 IEEE Symposium on Visual Languages and Human-Centric Computing (IEEE), pp. 41–48.

Lei, T., Barzilay, R., and Jaakkola, T. (2016). Rationalizing neural predictions. arXiv, preprint arXiv:1606.04155.

Li, B., Hu, W., Wu, T., and Zhu, S.C. (2013). Modeling occlusion by discriminative and-or structures. In Proceedings of the IEEE International Conference on Computer Vision, pp. 2560–2567.

Lipton, Z.C. (2016). The mythos of model interpretability. In ICML Workshop on Human Interpretability in Machine Learning.

Lombrozo, T. (2006). The structure and function of explanations. Trends Cogn. Sci. *10*, 464–470.

Lundberg, S.M., and Lee, S.I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems, *30*, I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds (Curran Associates, Inc.), pp. 4765–4774. http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

Lyons, J.B., Clark, M.A., Wagner, A.R., and Schuelke, M.J. (2017). Certifiable trust in autonomous systems: making the intractable tangible. AI Mag. *38*, 37–49.

Miller, T. (2018). Explanation in artificial intelligence: insights from the social sciences. Artif. Intell. *267*, 1–38.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M.A. (2013). Playing Atari with deep reinforcement learning, arXiv:1312.5602. http://arxiv.org/abs/1312.5602.

Molnar, C. (2019). Interpretable machine learning. https://lulu.com/.

Moore, J.D., and Swartout, W.R. (1990). Pointing: a way toward explanation dialogue. In AAAI, pp. 457–464.

Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., and Frossard, P. (2017). Universal adversarial perturbations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1765–1773.

Palakurthi, A., Ruthu, S., Akula, A., and Mamidi, R. (2015). Classification of attributes in a natural language query into different SQL clauses. In Proceedings of the International Conference Recent Advances in Natural Language Processing, pp. 497–506.

Park, S., Nie, B.X., and Zhu, S.C. (2018). Attribute and-or grammar for joint parsing of human attributes, part and pose. IEEE Trans. Pattern Anal. Mach. Intell. *40*, 1555–1569.

Pearce, C., Meadows, B., Langley, P., and Barley, M. (2014). Social planning: achieving goals by altering others' mental states. In Proceedings of the AAAI Conference on Artificial Intelligence.

Pedreschi, D., Giannotti, F., Guidotti, R., Monreale, A., Pappalardo, L., Ruggieri, S., and Turini, F. (2018). Open the black box data-driven explanation of black box decision systems. arXiv, preprint arXiv:1806.09936.

Polino, A., Pascanu, R., and Alistarh, D. (2018). Model compression via distillation and quantization. arXiv, preprint arXiv:1802.05668.

Premack, D., and Woodruff, G. (1978). Does the chimpanzee have a theory of mind? Behav. Brain Sci. *1*, 515–526.

Pulijala, V., Akula, A.R., and Syed, A. (2013). A web-based virtual laboratory for electromagnetic theory. In 2013 IEEE Fifth International Conference on Technology for Education (IEEE), pp. 13–18.

Rabinowitz, N., Perbet, F., Song, F., Zhang, C., Eslami, S.A., and Botvinick, M. (2018). Machine theory of mind. In International Conference on Machine Learning, PMLR, pp. 4218–4227.

Raileanu, R., Denton, E., Szlam, A., and Fergus, R. (2018). Modeling others using oneself in multi-agent reinforcement learning. In International Conference on Machine Learning, PMLR, pp. 4257–4266.

Ramırez, M., and Geffner, H. (2011). Goal recognition over POMDPS: inferring the intention of a POMDP agent. In IJCAI, Citeseer, pp. 2009–2014.

Ramprasaath, R., Abhishek, D., Ramakrishna, V., Michael, C., Devi, P., and Dhruv, B. (2016). Grad-cam: why did you say that? Visual explanations from deep networks via gradient-based localization. Comput. Vis. Pattern Recognit. https://doi.org/10.1007/s11263-019-01228-7.

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, ACM, pp. 1135–1144.

Ribeiro, M.T., Singh, S., and Guestrin, C. (2018). Anchors: high-precision model-agnostic explanations. In Thirty-Second AAAI Conference on Artificial Intelligence.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. *115*, 211–252.

Ruth, B. (2007). The Rational Imagination, How People Create Alternatives to Reality (MIT Press).

Sato, M., and Tsukimoto, H. (2001). Rule extraction from neural networks via decision tree induction. In IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222) (IEEE), pp. 1870–1875.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017a). Grad-cam: visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017b). Grad-cam: visual explanations from deep networks via gradient-based localization. In ICCV.

Sheh, R., and Monteath, I. (2018). Defining explainable AI for requirements analysis. Künstl. Intell. *32*, 261–266.

Sheh, R.K.M. (2017). "Why did you do that?" Explainable intelligent robots. In Workshops at the Thirty-First AAAI Conference on Artificial Intelligence.

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, *70* (JMLR.org), pp. 3145–3153.

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv, preprint arXiv:1409.1556.

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

Stone, A., Wang, H., Stark, M., Liu, Y., Phoenix, D.S., and George, D. (2017). Teaching compositionality to CNNs. In CVPR.

Strobelt, H., Gehrmann, S., Huber, B., Pfister, H., and Rush, A.M. (2016). Visual analysis of hidden state dynamics in recurrent neural networks. arXiv, preprint arXiv:1606.07461.

Su, H., Jampani, V., Sun, D., Gallo, O., Learned-Miller, E., and Kautz, J. (2019). Pixel-adaptive convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11166–11175.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. In 34th International Conference on Machine Learning.

Sutton, R.S., McAllester, D.A., Singh, S.P., and Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In Advances in Neural Information Processing Systems, pp. 1057–1063.

Szafron, D., Greiner, R., Lu, P., Wishart, D., MacDonell, C., Anvik, J., Poulin, B., Lu, Z., and Eisner, R. (2003). Explaining Naïve Bayes Classifications. TR03-09 (Department of Computing Science, University of Alberta).

Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., and Fidler, S. (2016). Movieqa: understanding stories in movies through question-answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4631–4640.

Van Looveren, A., and Klaise, J. (2019). Interpretable counterfactual explanations guided by prototypes. arXiv, preprint arXiv:1907.02584.

Wachter, S., Mittelstadt, B., and Russell, C. (2017). Counterfactual explanations without opening the black box: automated decisions and the GDPR. Harv. J. Law Technol. *31*, 2018.

Walton, D. (2004). A new dialectical theory of explanation. Philos. Explor. *7*, 71–89.

Walton, D. (2011). A dialogue system specification for explanation. Synthese *182*, 349–374.

Wang, T., Rudin, C., Velez-Doshi, F., Liu, Y., Klampfl, E., and MacNeille, P. (2016). Bayesian rule sets for interpretable classification. In 2016 IEEE 16th International Conference on Data Mining (ICDM) (IEEE), pp. 1269–1274.

Wang, Z., Bapst, V., Heess, N., Mnih, V., Munos, R., Kavukcuoglu, K., and de Freitas, N. (2017). Sample efficient actor-critic with experience replay. In Proceedings of the 5th International Conference on Learning Representations, ICLR.

Williams, J.J., Kim, J., Rafferty, A., Maldonado, S., Gajos, K.Z., Lasecki, W.S., and Heffernan, N. (2016). Axis: generating explanations at scale with learnersourcing and machine learning. In Proceedings of the Third (2016) ACM Conference on Learning@ Scale (ACM), pp. 379–388.

Yang, S., Gao, Q., Saba-Sadiya, S., and Chai, J. (2018). Commonsense justification for action explanation. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 2627–2637.

Yoshida, W., Dolan, R.J., and Friston, K.J. (2008). Game theory of mind. Plos Comput. Biol. *4*, e1000254.

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

Zhang, Q.S., and Zhu, S.C. (2018). Visual interpretability for deep learning: a survey. Front. Inf. Technol. Electron. Eng. *19*, 27–39.

Zhang, Q., Cao, R., Nian Wu, Y., and Zhu, S.C. (2017). Mining object parts from cnns via active question-answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 346–355.

Zhang, Q., Nian Wu, Y., and Zhu, S.C. (2018a). Interpretable convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8827–8836.

Zhang, Q., Yang, Y., Yu, Q., and Wu, Y.N. (2018b). Network transplanting. arXiv, 1804.10272.

Zhang, Q., Wang, X., Wu, Y.N., Zhou, H., and Zhu, S.C. (2019a). Interpretable cnns for object classification. arXiv, preprint arXiv:1901.02413.

Zhang, Q., Yang, Y., Ma, H., and Wu, Y.N. (2019b). Interpreting cnns via decision trees. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6261–6270.

Zhang, Q., Ren, J., Huang, G., Cao, R., Wu, Y.N., and Zhu, S.C. (2020a). Mining interpretable AOG representations from convolutional networks via active question answering. IEEE Trans. Pattern Anal. Mach. Intell. *43*, 3949–3963.

Zhang, Q., Wang, X., Cao, R., Wu, Y.N., Shi, F., and Zhu, S.C. (2020b). Extracting an explanatory graph to interpret a CNN. IEEE Trans. Pattern Anal. Mach. Intell. https://doi.org/10.1109/TPAMI.2020.2992207.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (IEEE), pp. 2921–2929.

Zhu, S.C., and Mumford, D. (2007). A stochastic grammar of images. Found. Trends Comput. Graph. Vis. *2*, 259–362.

Zhu, Y., Gao, T., Fan, L., Huang, S., Edmonds, M., Liu, H., Gao, F., Zhang, C., Qi, S., Wu, Y.N., et al. (2020). Dark, beyond deep: a paradigm shift to cognitive AI with humanlike common sense. Engineering *6*, 310–345.

Zilke, J.R., Mencía, E.L., and Janssen, F. (2016). Deepred–rule extraction from deep neural networks. In International Conference on Discovery Science (Springer), pp. 457–473.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Software and algorithms		
TCAV algorithm	https://github.com/rakhimovv/tcav	https://github.com/rakhimovv/tcav
FISTA Optimization algorithm	https://arxiv.org/pdf/1802.07623.pdf	https://arxiv.org/pdf/1802.07623.pdf

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact (Full Name: Arjun Reddy Akula; Email Address: aakula@ucla.edu).

Materials availability

This study did not generate new unique reagents.

Data and code availability

The code is available publicly at this github page: https://github.com/arjunakula/faultline_explainer.

METHOD DETAILS

In our human study experiments, we recruited 120 human subjects from our institution's Psychology subject pool. These experiments were reviewed and approved by our institution's IRB. We applied between-subject design and randomly assigned each subject into one of the experiment and control groups. We did not leverage any dataset from other publications. We leveraged the TCAV Kim et al. (2018) code to generate explainable concepts.

ADDITIONAL RESOURCES

Our study has not generated or contributed to a new website.