METHODS AND APPLICATIONS OF EXPLAINABLE MACHINE LEARNING

Joon Sik Kim

May 2023
CMU-ML-23-101

Machine Learning Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee

Ameet Talwalkar, Chair
Nihar Shah
Adam Perer
Chenhao Tan (University of Chicago)

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

This research was sponsored by: Defense Advanced Research Projects Agency award FA875017C0141; National Institutes of Health award R01GM117594; National Science Foundation awards IIS1661755, IIS1664720, IIS1705121, IIS1838017, IIS2046613; and contracts from JP Morgan Chase, Rakuten and ZestFinance.

Keywords: Machine Learning, Interpretable and Explainable Machine Learning, Algorithmic Fairness

To my mentors, colleagues, friends, and family.

Abstract

As machine learning models are more frequently deployed in various applications, there is a growing need to better understand, interact with, and regulate their behaviors. Explainable machine learning is a research field dedicated to this need, whose primary focus initially has been on *methodological developments* satisfying favorable algorithmic properties that elicit potentially useful information about the model predictions. However, critiques have also emphasized the need for more *critical evaluations* of these methods applied to concrete tasks with different users. In this thesis, we provide our individual contributions to both methodological and applied aspects of the field.

Methodologically, we present an efficient algorithm providing important information about the model behavior through influential training data points. We then propose a principled framework for understanding the model's trade-offs with performance and fairness metrics. Next, from an application-driven perspective, we discuss an evaluation framework that tests if the existing saliency methods on images are suitable for the practical task of spurious correlation detection. Lastly, motivated by practical issues in academic peer review, we present our findings on the utility of new and existing methods in helping human users perform the task of document matching.

Acknowledgments

Being part of the Ph.D. program at CMU has been an exceptionally rewarding experience. Such an experience would not have been possible without the support and guidance from many amazing people around me.

I express my deepest gratitude to my advisor Ameet Talwalkar. His guidance on asking the right question and approaching it with precision, not with myopic technicalities but with broader agenda helped me build a better perspective as a researcher. Also, if it were not for his kind and supportive advice throughout, I surely would not be here today.

I would also like to sincerely thank the rest of my thesis committee. Their works and expertise have helped shape my research directions. Nihar Shah, whom I had the fortune to collaborate on the work on peer review, opened up and introduced me an interesting research domain that is highly impactful yet less explored. Adam Perer and his work helped me to think more deeply about the role of humans interacting with machine learning systems. Chenhao Tan and his work encouraged me to consider the use of language models or interpretability methods as a tool for assisting the human users.

I would like to also thank my undergraduate advisor, Yisong Yue. I would not have considered taking this path if it were not for his active support throughout my initial studies in machine learning as an undergraduate.

I am also grateful for all the amazing research mentors and collaborators throughout. Jiahao Chen, whom I had the fortune to work with during my internship at JPMorgan, helped me to learn more about the field of algorithmic fairness and related areas with more practical viewpoint. Scott Lundberg and Marco Tulio Ribeiro, whom I had the honor to work with during my internship at Microsoft, helped me build better insights on causal lens of real-world problems. Other collaborators throughout the journey, including Chih-kuan Yeh, Ian En-Hsu Yen, Leqi Liu, David Inouye, Bryon Aragam, Pradeep Ravikumar, Jefferey Li, Wendy Yang, Wenbo Cui, Jian Ma, Keegan Harris, Hoda Heidari, Zhiwei Steven Wu, and Danish Pruthi, have all helped me significantly.

I want to express my gratitude to all members of SAGE Lab, whose discussions were invaluabe to shaping and sharpening many ideas. This is particularly true for the members of "Team Interpretability" in the lab: Gregory Plumb, Valerie Chen, and Nari Johnson. I learned a lot from all of them.

I would like to thank all my friends, especially the ones who have helped me feel at home despite being physically 6,000 miles away, including Byeoungju Ahn, Youngseog Chung, Byeongsu Jeon, Kwangho Kim, Kwangkyun Kim, Jisu Kim, Juyong Kim, Mark Moonyoung Lee, Andy Jonghyuk Song.

Last but not least, I always will be indebted to my parents, Taesuk Kim and Aehee Park, and my little sister, Yeonjee Kim, who have unconditionally supported me in and out through easy and hard times.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With impressive performance in practical domains like computer vision and natural language processing, complex machine learning models are increasingly adopted to assist humans in high-stakes decision-making, such as medical, financial, legal, and social applications. Such accelerated adoption creates a growing need for human users to better understand, regulate, and interact with these models.

Explainable machine learning[1] is a broad research field dedicated to this need [34, 79]. Many work in the literature focuses on methodological developments: developing new methods satisfying various technical objectives that can efficiently elicit important and useful information from a black-box machine learning model. However, various technical objectives used by these methods have no clear connection to the actual "importance" or "usefulness" of the information elicited, which inherently is dependent on the users using the information for some downstream task [28]. Hence, evaluating the developed methods based on concrete applications is critical to fully close the loop of developing novel methods with practical utility. In this thesis, we present individual contributions to both methodological and application-focused aspects of the field.

	Methodology		Application	
	Method Name	Information	Useful To	Useful For
Ch. 2	**Representer points** [143]	**Important samples**	ML practitioners	Dataset debugging, understanding misclassification
Ch. 3	**Fairness-Accuracy Trade-off diagnostic** [63]	**Fairness guarantees**	ML practitioners, regulators	Communicating trade-offs in fairness and accuracy for compliance and development
Ch. 4	Saliency methods	Important input features	**ML practitioners**	**Detecting spurious correlations** [64]
Ch. 5	Task-specific methods	Important input features	**Meta-reviewers**	**Matching submitted papers to potential reviewers** [65]

Table 1.1: Organization of our contributions presented in the thesis (bold parts indicate primary focus of each work). In Chapters 2 and 3, we focus on methodology, presenting novel methods that elicit different types of information about the model. In Chapters 4 and 5, we focus on application, where we study the practical benefits of given methods and information for different users and tasks. Considering both the methodology and application can lead to methods that are well-grounded on practical utility with desirable algorithmic properties.

[1] It is also often alternatively referred to as explainable AI (XAI in short), or interpretable machine learning.

Methodological Contributions

Picking Important Training Samples. One family of methods in explainable machine learning literature is *sample-based explanation* which attributes the model prediction to individual training samples. For instance, [68] approximate the influence function for deep neural networks to measure the impact of individual training data points on the trained model's loss function. Several follow-up works have suggested alternatives or improvements for selecting influential training samples for the model [45, 60, 98].

In Chapter 2, we present an efficient method of this family that selects a set of *representer points*, or a set of training data points, that are computationally influential to the model's loss function [143]. We do this by decomposing the prediction of the model into a linear combination of activations of training points. The quality of the selected representer points is evaluated in simulated tasks such as data label fixing and model debugging, which are generally important parts of a general workflow for ML practitioners (first row of Table 1.1). The proposed method is able to maintain the quality of the selection while being computationally efficient compared to the baselines for large-scale datasets.

Assessing Model Fairness. Another family of methods in the literature is *performance-based methods*, where they detect biases or blindspots of the model through relationships among model performance metrics on different subgroups of the test data based on specific features. For instance, a broad set of methods in algorithmic fairness assesses the model's bias using these relationships for different subgroups based on protected attributes [15, 21, 29, 36, 47, 67, 92] and try to mitigate it with data preprocessing [125, 148], regularization during training [145, 146], or model post-processing [47].

In Chapter 3, we present a principled framework that unifies different notions of algorithmic fairness to better inform trade-offs between fairness vs. accuracy and fairness vs. fairness [63]. The resulting *Fairness-Accuracy Trade-off (FACT) diagnostic* allows one to better formalize what the model is capable of in terms of satisfying multiple fairness criteria and predictive performance guarantees for a given dataset. For instance, ML practitioners who train models subject to compliance and regulators who establish a policy based on the model's capability can all potentially benefit from the framework (second row of Table 1.1).

Application-focused Contributions

Testing Correctness of Saliency Methods. *Saliency methods* refer to the set of methods that attribute scores (called feature attribution scores) to individual input features (e.g., pixels for images, tokens of text) based on how "influential" they are to the model's prediction [7, 11, 83, 101, 108, 113, 114, 117, 124, 147]. The output of these methods are typically presented in the form of heatmaps or highlights over the input features, based on the intensity of the feature attribution scores assigned per given input feature.

In Chapter 4, we develop a synthetic evaluation framework that tests if multiple leading saliency methods in the literature are capable of correctly detecting spurious correlations, an important task for ML practitioners where they check if the patterns a model relies on indeed align with expectations or domain knowledge (third row of Table 1.1)[64]. Although the existing saliency methods on image data are designed to highlight regions important to the model, we find that this may not be true and the methods are, in fact, more brittle. The presented evaluation framework, called `SMERF`, verifies if the methods can faithfully highlight the truly important region for a model under different conditions, allowing one to screen methods that are more suitable for practice.

Assisting Human Users for Matching. Compared to synthetic evaluations, *human-based evaluations* aim to capture the pros and cons of the methods applied to a specific task more realistically, via randomized controlled experiments with human users who directly interact with the information provided by the methods to complete some task. It is often the case that the methods previously claimed to be effective in synthetic evaluations may not be as effective in human-based evaluations [6, 9, 12, 53]. Such discrepancy speaks for the importance of human-based evaluation to fully gauge the method's practical utility.

In Chapter 5, we present our study of different types of explanation methods on a specific downstream task: assisting humans in document matching [65]. One motivating example of this application is paper-reviewer assignment in academic peer review for which the meta-reviewers are responsible (fourth row of Table 1.1). The increasing volume of papers submitted and the need to better understand existing matching algorithms call for more useful tools to assist their work. To this end, we not only develop new task-specific methods, but also test them, along with other existing methods, for their explicit utility on real human users. We present our findings on a simplified experimental setup, showing the potential advantages of the task-specific methods over the existing ones. We further present our efforts in adapting the methods to a more realistic setting of academic peer review, to further examine their true practical utility.

Chapter 2

Understanding Model Decisions through Influential Training Samples

2.1 Introduction

As machine learning systems start to be more widely used, we are starting to care not just about the accuracy and speed of the predictions, but also *why* it made its specific predictions. While we need not always care about the why of a complex system in order to trust it, especially if we observe that the system has high accuracy, such trust typically hinges on the belief that some other expert has a richer understanding of the system. For instance, while we might not know exactly how planes fly in the air, we trust some experts do. In the case of machine learning models however, even machine learning experts do not have a clear understanding of why say a deep neural network makes a particular prediction. Our work proposes to address this gap by focusing on improving the understanding of experts, in addition to lay users. In particular, expert users could then use these explanations to further fine-tune the system (e.g. dataset/model debugging), as well as suggest different approaches for model training, so that it achieves a better performance.

Our key approach to do so is via a representer theorem for deep neural networks, which might be of independent interest even outside the context of explainable ML. We show that we can decompose the pre-activation prediction values into a linear combination of training point activations, with the weights corresponding to what we call representer values, which can be used to measure the importance of each training point has on the learned parameter of the model. Using these representer values, we select representer points – training points that have large/small representer values – that could aid the understanding of the model's prediction.

Such representer points provide a richer understanding of the deep neural network than other approaches that provide influential training points, in part because of the meta-explanation underlying our explanation: a positive representer value indicates that a similarity to that training point is *excitatory*, while a negative representer value indicates that a similarity to that training point is *inhibitory*, to the prediction at the given test point. It is in these inhibitory training points where our approach provides considerably more insight compared to other approaches: specifically, what would cause the model to *not* make a particular prediction? In one of our examples, we see that the model makes an error in labeling an antelope as a deer. Looking at its most inhibitory training points, we see that the dataset is rife with training images where there are antelopes in the image, but also some other animals, and the image is labeled with the other animal. These thus contribute to inhibitory effects of small antelopes with other big objects: an insight that as machine learning experts, we found deeply useful, and which is difficult to obtain via other explanatory approaches. We demonstrate the utility of our class of *representer point* explanations through a range of theoretical and empirical investigations.

Source code supporting the contents of this chapter can be found here: `https://github.com/chihkuanyeh/Representer_Point_Selection`.

2.2 Related Work

There are two main classes of approaches to explain the prediction of a model. The first class of approaches point to important input features. Ribeiro et al. [101] provide such feature-based explanations that are model-agnostic; explaining the decision locally around a test instance by fitting a local linear model in the region. Ribeiro et al. [102] introduce Anchors, which are locally sufficient conditions of features that "holds down" the prediction so that it does not change in a local neighborhood. Such feature based explanations are particularly natural in computer vision tasks, since it enables visualizing the regions of the input pixel space that causes the classifier to make certain predictions. There are numerous works along this line, particularly focusing on gradient-based methods that provide saliency maps in the pixel space [11, 114, 117, 124].

The second class of approaches are sample-based, and they identify training samples that have the most influence on the model's prediction on a test point. Among model-agnostic sample-based explanations are prototype selection methods [16, 61] that provide a set of "representative" samples chosen from the data set. Kim et al. [62] provide criticism alongside prototypes to explain what are not captured by prototypes. Usually such prototype and criticism selection is model-agnostic and used to accelerate the training for classifications. Model-aware sample-based explanation identify influential training samples which are the most helpful for reducing the objective loss or making the prediction. Recently, Koh and Liang [68] provide tractable approximations of influence functions that characterize the influence of each sample in terms of change in the loss. Anirudh et al. [8] propose a generic approach to influential sample selection via a graph constructed using the samples.

Our approach is based on a representer theorem for deep neural network predictions. Representer theorems [105] in machine learning contexts have focused on non-parametric regression, specifically in reproducing kernel Hilbert spaces (RKHS), and which loosely state that under certain conditions the minimizer of a loss functional over a RKHS can be expressed as a linear combination of kernel evaluations at training points. There have been recent efforts at leveraging such insights to compositional contexts [18, 129], though these largely focus on connections to non-parametric estimation. Bohn et al. [18] extend the representer theorem to compositions of kernels, while Unser [129] draws connections between deep neural networks to such deep kernel estimation, specifically deep spline estimation. In our work, we consider the much simpler problem of explaining pre-activation neural network predictions in terms of activations of training points, which while less illuminating from a non-parametric estimation standpoint, is arguably much more explanatory, and useful from an explainable ML standpoint.

2.3 Representer Point Framework

Consider a classification problem, of learning a mapping from an input space $\mathcal{X} \subseteq \mathbb{R}^d$ (e.g., images) to an output space $\mathcal{Y} \subseteq \mathbb{R}$ (e.g., labels), given training points $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_n$, and corresponding labels $\mathbf{y}_1, \mathbf{y}_2, ...\mathbf{y}_n$. We consider a neural network as our prediction model, which takes the form $\hat{\mathbf{y}}_i = \sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta})) \subseteq \mathbb{R}^c$, where $\Phi(\mathbf{x}_i, \mathbf{\Theta}) = \mathbf{\Theta}_1 \mathbf{f}_i \subseteq \mathbb{R}^c$ and $\mathbf{f}_i = \Phi_2(\mathbf{x}_i, \mathbf{\Theta}_2) \subseteq \mathbb{R}^f$ is the last intermediate layer feature in the neural network for input $\mathbf{x}_i$. Note that c is the number of classes, f is the dimension of the feature, $\mathbf{\Theta}_1$ is a matrix $\subseteq \mathbb{R}^{c \times f}$, and $\mathbf{\Theta}_2$ is all the parameters to generate the last intermediate layer from the input $\mathbf{x}_i$. Thus $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2\}$ are all the parameters of our neural network model. The parameterization above connotes splitting of the model as a feature model $\Phi_2(\mathbf{x}_i, \boldsymbol{\Theta}_2)$ and a prediction network with parameters $\boldsymbol{\Theta}_1$. Note that the feature model $\Phi_2(\mathbf{x}_i, \boldsymbol{\Theta}_2)$ can be arbitrarily deep, or simply the identity function, so our setup above is applicable to general feed-forward networks.

Our goal is to understand to what extent does one particular training point $\mathbf{x}_i$ affect the prediction $\hat{\mathbf{y}}_t$ of a test point $\mathbf{x}_t$ as well as the learned weight parameter $\boldsymbol{\Theta}$. Let $L(\mathbf{x}, \mathbf{y}, \boldsymbol{\Theta})$ be the loss, and $\frac{1}{n}\sum_i^n L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})$ be the empirical risk. To indicate the form of a representer theorem, suppose we solve for the optimal parameters $\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} \left\{\frac{1}{n}\sum_i^n L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta}) + g(||\boldsymbol{\Theta}||)\right\}$ for some non-decreasing g. We would then like our pre-activation predictions $\Phi(\mathbf{x}_t, \boldsymbol{\Theta})$ to have the decomposition: $\Phi(\mathbf{x}_t, \boldsymbol{\Theta}^*) = \sum_i^n \alpha_i k(\mathbf{x}_t, \mathbf{x}_i)$. Given such a representer theorem, $\alpha_i k(\mathbf{x}_t, \mathbf{x}_i)$ can be seen as the contribution of the training data $\mathbf{x}_i$ on the testing prediction $\Phi(\mathbf{x}_t, \boldsymbol{\Theta})$. However, such representer theorems have only been developed for non-parametric predictors, specifically where Φ lies in a reproducing kernel Hilbert space. Moreover, unlike the typical RKHS setting, finding a global minimum for the empirical risk of a deep network is difficult, if not impossible, to obtain. In the following, we provide a representer theorem that addresses these two points: it holds for deep neural networks, and for any stationary point solution.

Theorem 2.3.1. *Let us denote the neural network prediction function by* $\hat{\mathbf{y}}_i = \sigma(\Phi(\mathbf{x}_i, \boldsymbol{\Theta}))$, *where* $\Phi(\mathbf{x}_i, \boldsymbol{\Theta}) = \boldsymbol{\Theta}_1 \mathbf{f}_i$ *and* $\mathbf{f}_i = \Phi_2(\mathbf{x}_i, \boldsymbol{\Theta}_2)$. *Suppose* $\boldsymbol{\Theta}^*$ *is a stationary point of the optimization problem:* $\arg\min_{\boldsymbol{\Theta}} \left\{\frac{1}{n}\sum_i^n L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})) + g(||\boldsymbol{\Theta}_1||)\right\}$, *where* $g(||\boldsymbol{\Theta}_1||) = \lambda ||\boldsymbol{\Theta}_1||^2$ *for some* $\lambda > 0$. *Then we have the decomposition:*

$$\Phi(\mathbf{x}_t, \boldsymbol{\Theta}^*) = \sum_i^n k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i),$$

where $\alpha_i = \frac{1}{-2\lambda n}\frac{\partial L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})}{\partial \Phi(\mathbf{x}_i, \boldsymbol{\Theta})}$ *and* $k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i) = \alpha_i \mathbf{f}_i^T \mathbf{f}_t$, *which we call a representer value for* $\mathbf{x}_i$ *given* $\mathbf{x}_t$.

Proof. Note that for any stationary point, the gradient of the loss with respect to $\boldsymbol{\Theta}_1$ is equal to 0. We therefore have

$$\frac{1}{n}\sum_{i=1}^n \frac{\partial L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}_1} + 2\lambda \boldsymbol{\Theta}_1^* = 0 \quad \Rightarrow \quad \boldsymbol{\Theta}_1^* = -\frac{1}{2\lambda n}\sum_{i=1}^n \frac{\partial L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}_1} = \sum_{i=1}^n \alpha_i \mathbf{f}_i^T \tag{2.1}$$

where $\alpha_i = -\frac{1}{2\lambda n}\frac{\partial L(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\Theta})}{\partial \Phi(\mathbf{x}_i, \boldsymbol{\Theta})}$ by the chain rule. We thus have that

$$\Phi(\mathbf{x}_t, \boldsymbol{\Theta}^*) = \boldsymbol{\Theta}_1^* \mathbf{f}_t = \sum_{i=1}^n k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i), \tag{2.2}$$

where $k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i) = \alpha_i \mathbf{f}_i^T \mathbf{f}_t$ by simply plugging in the expression (2.1) into (2.2). □

We note that α_i can be seen as the resistance for training example feature $\mathbf{f}_i$ towards minimizing the norm of the weight matrix $\boldsymbol{\Theta}_1$. Therefore, α_i can be used to evaluate the importance of the training data $\mathbf{x}_i$ have on $\boldsymbol{\Theta}_1$. Note that for any class j, $\Phi(\mathbf{x}_t, \boldsymbol{\Theta}^*)_j = \boldsymbol{\Theta}_{1j}^* \mathbf{f}_t = \sum_{i=1}^n k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i)_j$ holds by (2.2). Moreover, we can observe that for $k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i)_j$ to have a significant value, two conditions must be satisfied: (a) α_{ij} should have a large value, and (b) $\mathbf{f}_i^T \mathbf{f}_t$ should have a large value. Therefore, we interpret the pre-activation value $\Phi(\mathbf{x}_t, \boldsymbol{\Theta})_j$ as a weighted sum for the feature similarity $\mathbf{f}_i^T \mathbf{f}_t$ with the weight α_{ij}. When $\mathbf{f}_t$ is close to $\mathbf{f}_i$ with a large positive weight α_{ij}, the prediction score for class j is increased. On the other hand, when $\mathbf{f}_t$ is close to $\mathbf{f}_i$ with a large negative weight α_{ij}, the prediction score for class j is then decreased.

We can thus interpret the training points with negative representer values as inhibitory points that suppress the activation value, and those with positive representer values as excitatory examples that does the opposite. We demonstrate this notion with examples further in Section 2.4.2. We note that such excitatory and inhibitory points provide a richer understanding of the behavior of the neural network: it provides insight both as to why the neural network prefers a particular prediction, as well as *why it does not*, which is typically difficult to obtain via other sample-based explanations.

2.3.1 Training an Interpretable Model by Imposing L2 Regularization.

Theorem 2.3.1 works for any model that performs a linear matrix multiplication before the activation σ, which is quite general and can be applied on most neural-network-like structures. By simply introducing a L2 regularizer on the weight with a fixed $\lambda > 0$, we can easily decompose the pre-softmax prediction value as some finite linear combinations of a function between the test and train data. We now state our main algorithm. First we solve the following optimization problem:

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta}} \frac{1}{n} \sum_{i}^{n} L(\mathbf{y}_i, \Phi(\mathbf{x}_i, \mathbf{\Theta})) + \lambda ||\mathbf{\Theta}_1||^2. \tag{2.3}$$

Note that for the representer point selection to work, we would need to achieve a stationary point with high precision. In practice, we find that using a gradient descent solver with line search or LBFGS solver to fine-tune after converging in SGD can achieve highly accurate stationary point. Note that we can perform the fine-tuning step only on $\mathbf{\Theta}_1$, which is usually efficient to compute. We can then decompose $\Phi(\mathbf{x}_t, \mathbf{\Theta}) = \sum_i^n k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i)$ by Theorem 2.3.1 for any arbitrary test point $\mathbf{x}_t$, where $k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i)$ is the contribution of training point $\mathbf{x}_i$ on the pre-softmax prediction $\Phi(\mathbf{x}_t, \mathbf{\Theta})$. We emphasize that imposing L2 weight decay is a common practice to avoid overfitting for deep neural networks, which does not sacrifice accuracy while achieving a more interpretable model.

2.3.2 Generating Representer Points for a Given Pre-trained Model.

We are also interested in finding representer points for a given model $\Phi(\mathbf{\Theta}_{given})$ that has already been trained, potentially without imposing the L2 regularizer. While it is possible to add the L2 regularizer and retrain the model, the retrained model may converge to a different stationary point, and behave differently compared to the given model, in which case we cannot use the resulting representer points as explanations. Accordingly, we learn the parameters $\mathbf{\Theta}$ while imposing the L2 regularizer, but under the additional constraint that $\Phi(\mathbf{x}_i, \mathbf{\Theta})$ be close to $\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given})$. In this case, our learning objective becomes $\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given})$ instead of y_i, and our loss $L(\mathbf{x}_i, y_i, \mathbf{\Theta})$ can be written as $L(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$.

Definition 2.3.1. We say that a convex loss function $L(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$ is "suitable" to an activation function σ, if it holds that for any $\mathbf{\Theta}^* \in \arg\min_{\mathbf{\Theta}} L(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$, we have $\sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta}^*)) = \sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}))$.

Assume that we are given such a loss function L that is "suitable to" the activation function σ. We can then solve the following optimization problem:

$$\mathbf{\Theta}^* \in \arg\min_{\mathbf{\Theta}} \left\{ \frac{1}{n} \sum_{i}^{n} L(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta})) + \lambda ||\mathbf{\Theta}_1||^2 \right\}. \tag{2.4}$$

The optimization problem can be seen to be convex under the assumptions on the loss function. The parameter $\lambda > 0$ controls the trade-off between the closeness of $\sigma(\Phi(\mathbf{X}, \mathbf{\Theta}))$ and $\sigma(\Phi(\mathbf{X}, \mathbf{\Theta}_{given}))$, and

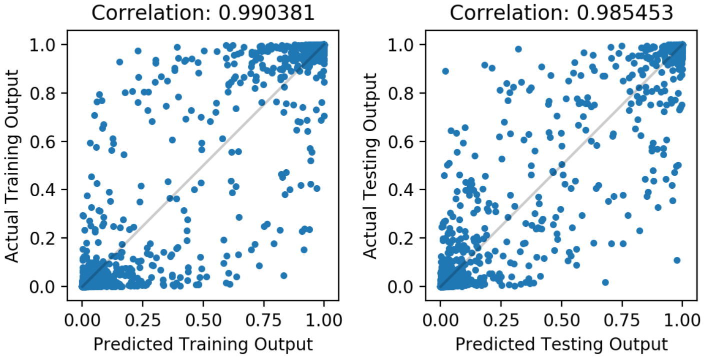


Figure 2.1: Pearson correlation between the actual and approximated softmax output (expressed as a linear combination) for train (left) and test (right) data in CIFAR-10 dataset. The correlation is almost 1 for both cases.

the computational cost. For a small λ, $\sigma(\Phi(\mathbf{X}, \mathbf{\Theta}))$ could be arbitrarily close to $\sigma(\Phi(\mathbf{X}, \mathbf{\Theta}_{given}))$, while the convergence time may be long. We note that the learning task in Eq. (2.4) can be seen as learning from a teacher network $\mathbf{\Theta}_{given}$ and imposing a regularizer to make the student model $\mathbf{\Theta}$ capable of generating representer points. In practice, we may take $\mathbf{\Theta}_{given}$ as an initialization for $\mathbf{\Theta}$ and perform a simple line-search gradient descent with respect to $\mathbf{\Theta}_1$ in (2.4). In our experiments, we discover that the training for (2.4) can converge to a stationary point in a short period of time, as demonstrated in Section 5.4.3.

We now discuss our design for the loss function that is mentioned in (2.4). When σ is the softmax activation, we choose the softmax cross-entropy loss, which computes the cross entropy between $\sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}))$ and $\sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta}))$ for $L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$. When σ is ReLU activation, we choose $L_{\text{ReLU}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta})) = \frac{1}{2}\max(\Phi(\mathbf{x}_i, \mathbf{\Theta}), 0) \odot \Phi(\mathbf{x}_i, \mathbf{\Theta}) - \max(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), 0) \odot \Phi(\mathbf{x}_i, \mathbf{\Theta})$, where $\odot$ is the element-wise product. In the following Proposition, we show that L_{softmax} and L_{ReLU} are convex, and satisfy the desired suitability property in Definition 2.3.1. The proof is provided in Appendix A.

Proposition 2.3.1. *The loss functions $L_{softmax}$ and L_{ReLU} are both convex in $\mathbf{\Theta}_1$. Moreover, $L_{softmax}$ is "suitable to" the softmax activation, and L_{ReLU} is "suitable to" the ReLU activation, following Definition 2.3.1.*

As a sanity check, we perform experiments on the CIFAR-10 dataset [71] with a pre-trained VGG-16 network [116]. We first solve (2.4) with loss $L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}), \Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}))$ for $\lambda = 0.001$, and then calculate $\Phi(\mathbf{x}_t, \mathbf{\Theta}^*) = \sum_{i=1}^{n} k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i)$ as in (2.2) for all train and test points. We note that the computation time for the whole procedure only takes less than a minute, given the pre-trained model. We compute the Pearson correlation coefficient between the actual output $\sigma(\Phi(\mathbf{x}_t, \mathbf{\Theta}))$ and the predicted output $\sigma(\sum_{i=1}^{n} k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i))$ for multiple points and plot them in Figure 2.1. The correlation is almost 1 for both train and test data, and most points lie at the both ends of $y = x$ line.

We note that Theorem 2.3.1 can be applied to any hidden layer with ReLU activation by defining a sub-network from input $\mathbf{x}$ and the output being the hidden layer of interest. The training could be done in a similar fashion by replacing L_{softmax} with L_{ReLU}. In general, any activation can be used with a derived "suitable" loss.

2.4 Experiments

We perform a number of experiments with multiple datasets and evaluate our method's performance and compare with that of the influence functions.[1] The goal of these experiments is to demonstrate that selecting the representer points is efficient and insightful in several ways. Additional experiments discussing the differences between our method and the influence function are included in Appendix A.

2.4.1 Dataset Debugging

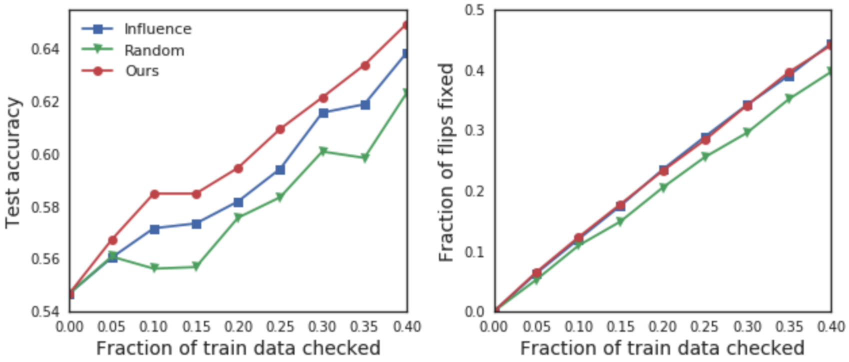


Figure 2.2: Dataset debugging performance for several methods. By inspecting the training points using the representer value, we are able to recover the same amount of mislabeled training points as the influence function (right) with the highest test accuracy compared to other methods (left).

To evaluate the influence of the samples, we consider a scenario where humans need to inspect the dataset quality to ensure an improvement of the model's performance in the test data. Real-world data is bound to be noisy, and the bigger the dataset becomes, the more difficult it will be for humans to look for and fix mislabeled data points. It is crucial to know which data points are more important than the others to the model so that prioritizing the inspection can facilitate the debugging process.

To show how well our method does in dataset debugging, we run a simulated experiment on CIFAR-10 dataset [75] with a task of binary classification with logistic regression for the classes automobiles and horses. The dataset is initially corrupted, where 40 percent of the data has the labels flipped, which naturally results in a low test accuracy of 0.55. The simulated user will check some fraction of the train data based on the order set by several metrics including ours, and fix the labels. With the corrected version of the dataset, we retrain the model and record the test accuracies for each metrics. For our method, we train an explainable model by mimimizing (2.3) as explained in section 2.3.1. The L2 weight decay is set to $1e^{-2}$ for all methods for fair comparison. All experiments are repeated for 5 random splits and we report the average result. In Figure 2.2 we report the results for four different metrics: "ours" picks the points with bigger $|\alpha_{ij}|$ for training instance i and its corresponding label j; "influence" prioritizes the training points with bigger influence function value; and "random" picks random points. We observe that our method recovers the same amount of training data as the influence function while achieving higher testing accuracy. Nevertheless, both methods perform better than the random selection method.

[1] Source code available at `github.com/chihkuanyeh/Representer_Point_Selection`.

2.4.2 Excitatory (Positive) and Inhibitory (Negative) Examples

We visualize the training points with high representer values (both positive and negative) for some test points in Animals with Attributes (AwA) dataset [140] and compare the results with those of the influence functions. We use a pre-trained Resnet-50 [48] model and fine-tune on the AwA dataset to reach over 90 percent testing accuracy. We then generate representer points as described in section 2.3.2. For computing the influence functions, just as described in [68], we froze all top layers of the model and trained the last layer. We report top three points for two test points in the following Figures 2.3 and 2.4. In Figure 2.3, which is an image of three grizzly bears, our method correctly returns three images that are in the same class with similar looks, similar to the results from the influence function. The positive examples excite the activation values for a particular class and supports the decision the model is making. For the negative examples, just like the influence functions, our method returns images that look like the test image but are labeled as a different class. In Figure 2.4, for the image of a rhino the influence function could not recover useful training points, while ours does, including the similar-looking elephants or zebras which might be confused as rhinos, as negatives. The negative examples work as inhibitory examples for the model – they suppress the activation values for a particular class of a given test point because they are in a different class despite their striking similarity to the test image. Such inhibitory points thus provide a richer understanding, even to machine learning experts, of the behavior of deep neural networks, since they explicitly indicate training points that lead the network away from a particular label for the given test point. More examples can be found in Appendix A.

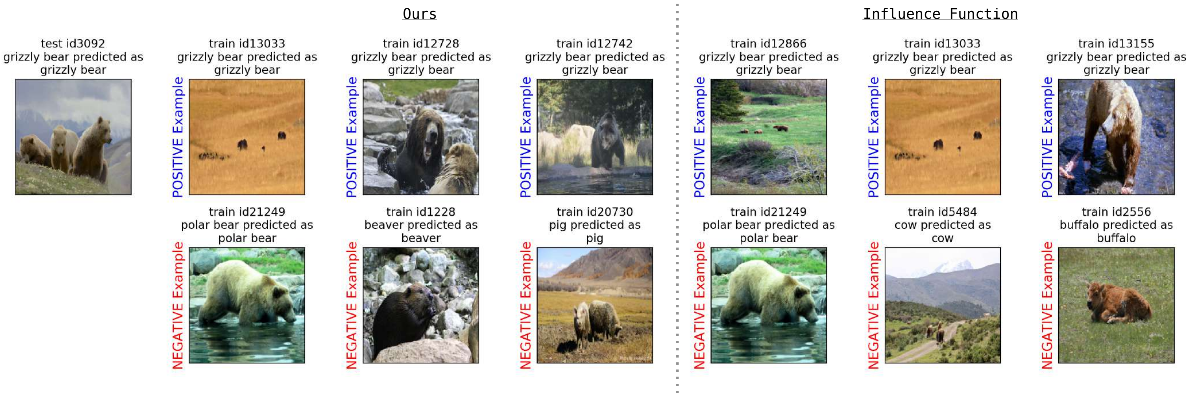


Figure 2.3: Comparison of top three positive and negative influential training images for a test point (left-most column) using our method (left columns) and influence functions (right columns).

2.4.3 Understanding Misclassified Examples

The representer values can be used to understand the model's mistake on a test image. Consider a test image of an antelope predicted as a deer in the left-most panel of Figure 2.5. Among 181 test images of antelopes, the total number of misclassified instances is 15, among which 12 are misclassified as deer. All of those 12 test images of antelopes had the four training images shown in Figure 2.5 among the top inhibitory examples. Notice that we can spot antelopes even in the images labeled as zebra or elephant. Such noise in the labels of the training data confuses the model – while the model sees elephant *and* antelope, the label forces the model to focus on just the elephant. The model thus learns to inhibit the antelope class given an image with small antelopes and other large objects. This insight suggests for instance that we use multi-label prediction to train the network, or perhaps clean the dataset to remove

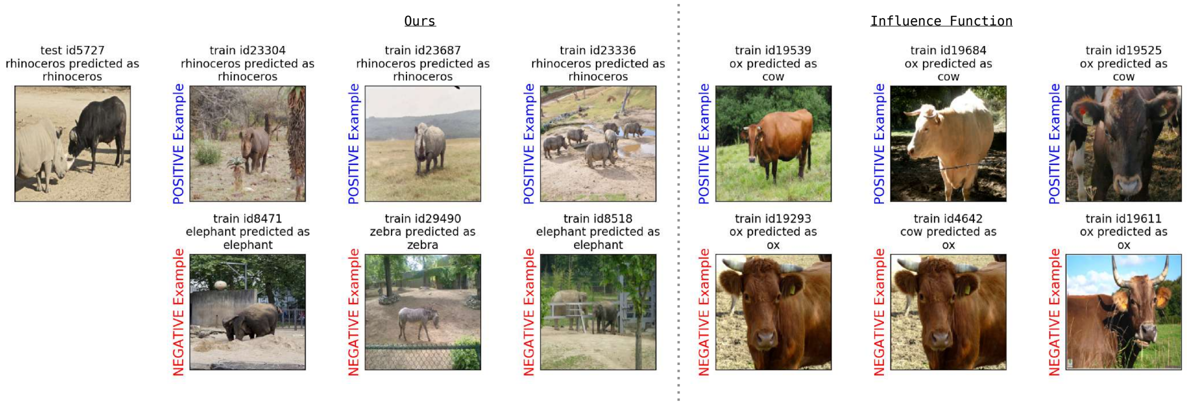

Ours
Influence Function
test id5727 rhinoceros predicted as rhinoceros
train id23304 rhinoceros predicted as rhinoceros
train id23687 rhinoceros predicted as rhinoceros
train id23336 rhinoceros predicted as rhinoceros
train id19539 ox predicted as cow
train id19684 ox predicted as cow
train id19525 ox predicted as cow
POSITIVE Example
train id8471 elephant predicted as elephant
train id29490 zebra predicted as zebra
train id8518 elephant predicted as elephant
train id19293 ox predicted as ox
train id4642 cow predicted as ox
train id19611 ox predicted as ox
NEGATIVE Example

Figure 2.4: Here we can observe that our method provides clearer positive and negative examples while the influence function fails to do so.

such training examples that would be confusing to humans as well. Interestingly, the model makes the same mistake (predicting deer instead of antelope) on the second training image shown (third from the left of Figure 2.5), and this suggests that for the training points, we should expect most of the misclassifications to be deer as well. And indeed, among 863 training images of antelopes, 8 are misclassified, and among them 6 are misclassified as deer.

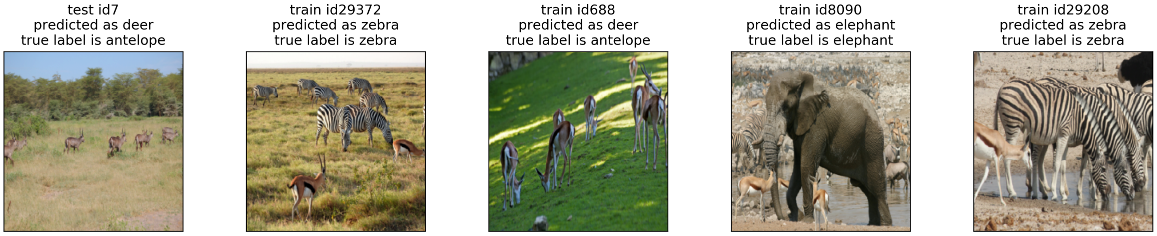


Figure 2.5: A misclassified test image (left) and the set of four training images that had the most negative representer values for almost all test images in which the model made the same mistakes. The negative influential images all have antelopes in the image despite the label being a different animal.

2.4.4 Sensitivity Map Decomposition

From Theorem 2.3.1, we have seen that the pre-softmax output of the neural network can be decomposed as the weighted sum of the product of the training point feature and the test point feature, or $\Phi(\mathbf{x}_t, \mathbf{\Theta}^*) = \sum_i^n \alpha_i \mathbf{f}_i^T \mathbf{f}_t$. If we take the gradient with respect to the test input $\mathbf{x}_t$ for both sides, we get $\frac{\partial \Phi(\mathbf{x}_t, \mathbf{\Theta}^*)}{\partial \mathbf{x}_t} = \sum_i^n \alpha_i \frac{\partial \mathbf{f}_i^T \mathbf{f}_t}{\partial \mathbf{x}_t}$. Notice that the LHS is the widely-used notion of sensitivity map (gradient-based attribution), and the RHS suggests that we can decompose this sensitivity map into a weighted sum of sensitivity maps that are native to each i-th training point. This gives us insight into how sensitivities of training points contribute to the sensitivity of the given test image.

In Figure 2.6, we demonstrate two such examples, one from the class zebra and one from the class moose from the AwA dataset. The first column shows the test images whose sensitivity maps we wish to decompose. For each example, in the following columns we show top four influential representer points in the the top row, and visualize the decomposed sensitivity maps in the bottom. We used SmoothGrad [120] to obtain the sensitivity maps.

For the first example of a zebra, the sensitivity map on the test image mainly focuses on the face of the zebra. This means that infinitesimally changing the pixels around the face of the zebra would cause the greatest change in the neuron output. Notice that the focus on the head of the zebra is distinctively the strongest in the fourth representer point (last column) when the training image manifests clearer facial features compared to other training points. For the rest of the training images that are less demonstrative of the facial features, the decomposed sensitivity maps accordingly show relatively higher focus on the background than on the face. For the second example of a moose, a similar trend can be observed – when the training image exhibits more distinctive bodily features of the moose than the background (first, second, third representer points), the decomposed sensitivity map highlights the portion of the moose on the test image more compared to training images with more features of the background (last representer point). This provides critical insight into the contribution of the representer points towards the neuron output that might not be obvious just from looking at the images itself.

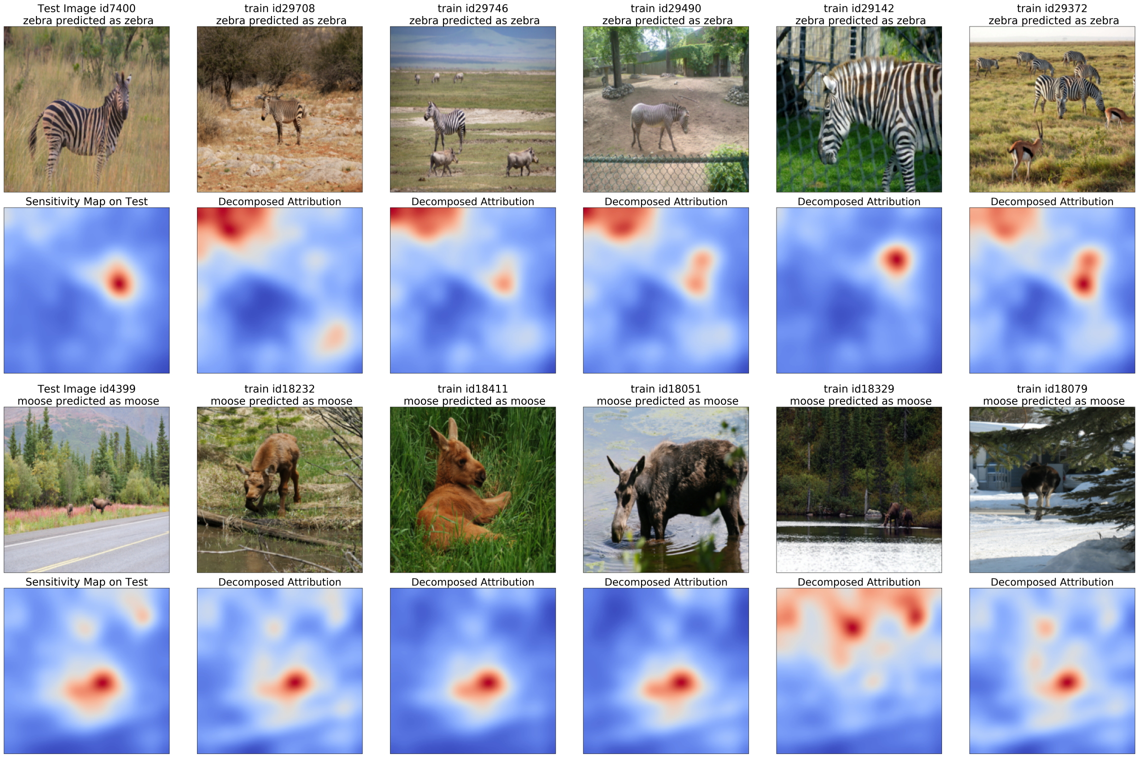


Figure 2.6: Sensitivity map decomposition using representer points, for the class zebra (above two rows) and moose (bottom two rows). The sensitivity map on the test image in the first column can be readily seen as the weighted sum of the sensitivity maps for each training point. The less the training point displays spurious features from the background and more of the features related to the object of interest, the more focused the decomposed sensitivity map corresponding to the training point is at the region the test sensitivity map mainly focuses on.

2.4.5 Computational Cost and Numerical Instabilities

Computation time is particularly an issue for computing the influence function values [68] for a large dataset, which is very costly to compute for each test point. We randomly selected a subset of test points, and report the comparison of the computation time in Table 2.1 measured on CIFAR-10 and AwA datasets. We randomly select 50 test points to compute the values for all train data, and recorded the average and standard deviation of computation time. Note that the influence function does not need the fine-tuning step when given a pre-trained model, hence the values being 0, while our method first optimizes for $\mathbf{\Theta}^*$ using line-search then computes the representer values. However, note that the fine-tuning step is a one time cost, while the computation time is spent for every testing image we analyze. Our method significantly outperforms the influence function, and such advantage will favor our method when a larger number of data points is involved. In particular, our approach could be used for *real-time explanations* of test points, which might be difficult with the influence function approach.

	Influence Function		Ours	
Dataset	Fine-tuning	Computation	Fine-tuning	Computation
CIFAR-10	0	267.08 ± 248.20	7.09 ± 0.76	0.10 ± 0.08
AwA	0	172.71 ± 32.63	12.41 ± 2.37	0.19 ± 0.12

Table 2.1: Time required for computing an influence function / representer value for all training points and a test point in seconds. The computation of Hessian Vector Products for influence function alone took longer than our combined computation time.

While ranking the training points according to their influence function values, we have observed numerical instabilities, more discussed in the supplementary material. For CIFAR-10, over 30 percent of the test images had all zero training point influences, so influence function was unable to provide positive or negative influential examples. The distribution of the values is demonstrated in Figure 2.7, where we plot the histogram of the maximum of the absolute values for each test point in CIFAR-10. Notice that over 300 testing points out of 1,000 lie in the first bin for the influence functions (right). We checked that all data in the first bin had the exact value of 0. Roughly more than 200 points lie in range $[10^{-40}, 10^{-28}]$, the values which may create numerical instabilities in computations. On the other hand, our method (left) returns non-trivial and more numerically stable values across all test points.

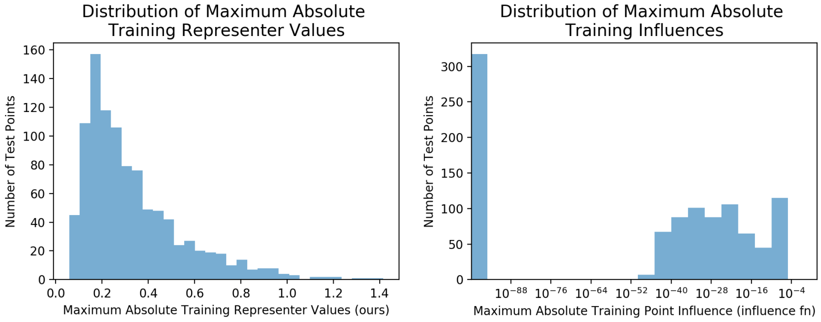


Figure 2.7: The distribution of influence/representer values for a set of randomly selected 1,000 test points in CIFAR-10. While ours have more evenly spread out larger values across different test points (left), the influence function values can be either really small or become zero for some points, as seen in the left-most bin (right).

Chapter 3

Understanding Model Decisions through Fairness-Performance Trade-offs

3.1 Introduction

As machine learning continues to be more widely used for applications with societal impact such as credit decisioning, predictive policing, and employment applicant screening, practitioners face regulatory, ethical, and legal challenges to prove whether or not their models are fair [32]. To provide quantitative tests of model fairness, the practitioners further need to choose between multiple definitions of fairness that exist in the machine learning literature [22, 92, 132]. Among them is a class of definitions called *group fairness*, which measures how a group of individuals with certain protected attributes are treated differently from other groups. From a technical point of view however, several notions of group fairness have been shown to be incompatible with one another [29, 67], sometimes with a necessary cost in loss of accuracy [80]. Such considerations complicate the practical development and assessment of machine learning models designed to satisfy group fairness, as the conditions under which these trade-offs must necessarily occur can be too abstract to understand. Previous works on these trade-offs have been presented in ad hoc and definition-specific manner, which further calls for a more general perspective addressing the trade-offs in practice.

As an example, suppose an engineer is responsible for training a loan prediction model from a large user dataset, subject to mandatory group fairness requirements shaped by regulatory concerns. One has many choices for how to train this fair model, with fairness enforced before [57, 85, 104, 121, 125, 148], during [145, 146], or after [36, 39, 47] training. However, the engineer must resort to trial and error to determine which of these myriad approaches, if any, will produce a compliant model with sufficient performance[1] to satisfy business needs. It may even turn out that despite one's best efforts, the fairness constraints set by the regulators are actually impossible to satisfy to begin with, due to limitations intrinsic to the prediction task and data at hand. If there was a tool to understand the potential trade-offs exhibited by the model, even before training, it would be easier for multiple parties to effectively reconcile the conflicting components in designing fair classifiers.

Motivated by such practical considerations, we propose the *FACT (**FA**irness-**C**onfusion **T**ensor) diagnostic* for exploring the trade-offs involving group fairness: the diagnostic provides a general framework under which the practitioners can understand both fairness–fairness trade-offs and fairness–performance trade-offs. At the core of our diagnostic lies the *fairness–confusion tensor*, which is the confusion matrix

[1] In this work, *performance* refers to classical metrics derived from the confusion matrix, e.g., accuracy, precision and fairness notions are not part of it.

divided along an additional axis for protected attributes. The FACT diagnostic first expresses the majority of group fairness notions as linear/quadratic functions of the elements of this tensor. The simplicity of these functions makes it easy for them to be naturally integrated into a class of optimization problems over the elements of the tensor (not over the model parameters), which we call *performance–fairness optimality problem* (PFOP). It essentially considers the geometry of valid fairness–confusion tensors that satisfy a specified set of performance and/or fairness conditions.

By noting that many settings involve only linear notions of fairness, in this work we focus on *least-squares accuracy–fairness optimality problem* (LAFOP) and *model-specific least-squares accuracy–fairness optimality problem* (MS-LAFOP), which are specific instantiations of PFOP, each representative of model-agnostic and model-specific scenarios. These problems allow one to not only understand group fairness incompatibility, but also analyze difficulty of learning a classifier under additional group fairness conditions imposed. In particular, for the model-agnostic case, the diagnostic allows for a comparative analysis of the *relative* difficulty of learning a classifier under additional group fairness constraints imposed. This difficulty is interpreted with respect to the Bayes error, which is the inherent difficulty of the fairness-unconstrained learning problem, hence a natural reference point.

Our contributions are:

1. to demonstrate how fairness–confusion tensor characterizes the majority of group fairness definitions in the literature as linear or quadratic functions, whose simplicity can be leveraged to formulate optimization problems suited for trade-off analysis,
2. to formulate the FACT diagnostic as a PFOP, LAFOP, and MS-LAFOP over the fairness–confusion tensor, enabling both model-agnostic and model-specific analysis of fairness trade-offs,
3. to provide a general understanding of group fairness incompatibility, which simplifies the existing results in the literature and extends them to new types,
4. to demonstrate the use of the FACT diagnostic on synthetic and real datasets, e.g. how it can be used for diagnosis of relative influence of the fairness notions on performance and other fairness conditions, and how it can be used as a post-processing method for designing fair classifiers.

Source code supporting the contents of this chapter can be found here: `https://github.com/wnstlr/FACT`.

3.2 Related Work

Fairness–confusion tensor is not a completely new notion – several work has implicitly mentioned it, mostly disregarding it as a simple computational tool that eases the computation on an implementation level [14, 24]. It is also a natural object considered in several post-processing methods in fairness [47, 96], a group of algorithms that fine-tune a trained model to mitigate the unfairness while keeping the performance change minimal. Here we take a closer look at the fairness–confusion tensor itself and study how this object naturally brings together several notions of group fairness, simplifying and generalizing the analysis of inherent trade-offs within.

Quantitative definitions of group fairness exist in many different variations [15, 21, 29, 36, 47, 67, 92] but few work exists to categorize these notions with a broader perspective encompassing the trade-off schemes. Verma and Rubin [131] categorized the existing group fairness definitions based on entries and rates derived from the fairness–confusion tensor but did not explore any trade-offs and incompatibilities within. Our work extends this effort and provides a versatile geometric formalism to study the trade-offs.

Fairness–performance trade-offs have been studied in many specific cases [22, 39, 57, 80, 89, 132, 151], for limited definitions of fairness, performance, and models. To our knowledge, these trade-offs have not been studied in the general way we present below. Zafar et al. [145, 146] presented an optimization-based analysis of the trade-offs, albeit over the parameter space of a particular model.

Fairness–fairness trade-offs describe the incompatibility of multiple notions of group fairness [15, 29, 67, 96] without some strong assumptions about the data and the model. Previous incompatibility results have been presented mostly in ad hoc and definition-specific manner, which our diagnostic addresses with a more general perspective for understanding incompatibilities. We show a general incompatibility result involving Calibration fairness condition, which naturally implies the result in Kleinberg et al. [67] along with many other new ones. To the best of our knowledge, our work is the first to provide a systematic approach to diagnose both fairness–fairness and fairness–performance trade-offs together for group fairness under the same formalism.

3.3 The Fairness–confusion Tensor

Our key insight is that the elements of the fairness–confusion tensor encode all the information needed to study many notions of performance and group fairness. The fairness–confusion tensor is simply the stack of confusion matrices for each protected attribute a, as shown in Table 3.1. We focus on the simplest case, with one binary protected attribute $a \in \{0, 1\}$, and a binary classifier $\hat{y} \in \{0, 1\}$ for a binary prediction label $y \in \{0, 1\}$.[2]

$a = 1$	$y = 1$	$y = 0$
$\hat{y} = 1$	TP$_1$	FP$_1$
$\hat{y} = 0$	FN$_1$	TN$_1$

$a = 0$	$y = 1$	$y = 0$
$\hat{y} = 1$	TP$_0$	FP$_0$
$\hat{y} = 0$	FN$_0$	TN$_0$

Table 3.1: The fairness–confusion tensor, showing the two planes corresponding to the confusion matrix for each of the favored ($a = 1$) and disfavored groups ($a = 0$).

Let us denote the elements of the fairness–confusion tensor as TP_a, FP_a, FN_a, TN_a, each element with subscripts indicating a, N be the number of data points, $N_a = TP_a + FN_a + FP_a + TN_a$ be the number of data points in each group $a \in \{0, 1\}$, and $M_a = TP_a + FN_a$ be the number of positive-class instances ($y = 1$) for each group. Assume N, N_a and M_a are known constants. Unraveling the fairness–confusion tensor into an 8-dimensional vector, we write it as

$$\mathbf{z} = (TP_1, FN_1, FP_1, TN_1, TP_0, FN_0, FP_0, TN_0)^T / N,$$

normalized and constrained to lie on $\mathcal{K} = \{\mathbf{z} \geq 0 : \mathbf{A}_{\mathsf{const}}\mathbf{z} \ = \mathbf{b}_{\mathsf{const}}, \|\mathbf{z}\|_1 = 1\}$, where $\mathbf{A}_{\mathsf{const}}$ and $\mathbf{b}_{\mathsf{const}}$ encode marginal sum constraints of the dataset (e.g., $TP_a + FN_a = M_a$) in matrix notations:

$$\mathbf{A}_{\mathsf{const}} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix},$$

$$\mathbf{b}_{\mathsf{const}} = (N_1, M_1, N_0, M_0)^T / N.$$

We show below that some typical notions of group fairness can be reformulated as simple functions of $\mathbf{z}$, namely as a form of $\phi(\mathbf{z}) = 0$.

[2] The arguments generalize to multiple and non-binary protected attributes with high-dimensional tensors.

Demographic parity (DP) states that each protected group should receive positive prediction at an equal rate: $\mathsf{Pr}(\hat{y} = 1|\mathbf{a} = 1) = \mathsf{Pr}(\hat{y} = 1|\mathbf{a} = 0)$, which is equivalent to $(TP_1 + FP_1)/N_1 = (TP_0 + FP_0)/N_0$, or also the linear system $\phi(\mathbf{z}) = \mathbf{A}_{\text{DP}}\mathbf{z} = 0$, where

$$\mathbf{A}_{\text{DP}} = \begin{pmatrix} N_0 & 0 & N_0 & 0 & -N_1 & 0 & -N_1 & 0 \end{pmatrix} / N. \tag{3.1}$$

The choice of normalization, $1/N$, ensures that the matrix coefficients are in $[0, 1]$. We will refer to these matrices $\mathbf{A}$ that encode information about the fairness conditions as fairness matrices.

Predictive parity (PP) [29] states that the likelihood of being in the positive class given the positive prediction is the same for each group: $\mathsf{Pr}(y = 1|\hat{y} = 1, \mathbf{a} = 1) = \mathsf{Pr}(y = 1|\hat{y} = 1, \mathbf{a} = 0)$, which is equivalent to $\frac{TP_1}{TP_1+FP_1} = \frac{TP_0}{TP_0+FP_0} \iff \frac{TP_1}{TP_0} = \frac{FP_1}{FP_0}$. Unlike for DP, the marginal sum constraints do not relate TP_a and FP_a, so this notion of fairness is *not* linear in the fairness–confusion tensor. PP actually can be expressed using a *quadratic* form:

$$\phi(\mathbf{z}) = \frac{1}{2}\mathbf{z}^T\mathbf{B}_{\text{PP}}\mathbf{z} = 0, \quad \mathbf{B}_{\text{PP}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.2}$$

Calibration within groups (CG) [67], when specialized to binary classifiers and binary protected classes, can be written as the system of equations $FN_a = v_0(FN_a + TN_a); TP_a = v_1(TP_a + FP_a)$, where the v_is are scores satisfying $0 \leq v_0 < v_1 \leq 1$ and have no implicit dependence on any entries of the fairness–confusion tensor. We can rewrite this this condition explicitly as the matrix equation $\phi(\mathbf{z}) = \mathbf{A}_{\text{CG}}\mathbf{z} = 0$ with a fairness matrix

$$\mathbf{A}_{\text{CG}} = \begin{pmatrix} 1 - v_1 & 0 & -v_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 - v_0 & 0 & -v_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 - v_1 & 0 & -v_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 - v_0 & 0 & -v_0 \end{pmatrix}. \tag{3.3}$$

Equalized odds (EOd) [47] states that true-positive rates and false-positive rates are the same for both groups, which can be expressed as a linear system $\phi(\mathbf{z}) = \mathbf{A}_{\text{EOD}}\mathbf{z} = 0$ with a fairness matrix

$$\mathbf{A}_{\text{EOD}} = \frac{1}{N}\begin{pmatrix} M_0 & 0 & 0 & 0 & -M_1 & 0 & 0 & 0 \\ 0 & 0 & N_0 - M_0 & 0 & 0 & 0 & -N_1 + M_1 & 0 \end{pmatrix} \tag{3.4}$$

where each row respectively corresponds to conditions for Equality of Opportunity (EOp) [47] and Predictive Equality (PE) [29]. Likewise, vertically stacking multiple fairness matrices results in a fairness matrix corresponding to the conjunction of different fairness notions.

In Table 3.2 we generalize this formulation to a wide majority of group fairness definitions in the literature, along with their abbreviations used throughout the paper. We find that most of the definitions take either linear or quadratic form with respect to $\mathbf{z}$. We further introduce a graphical notation to help visualize which components of the fairness–confusion tensor participate in the fairness definition. Depict the fairness–confusion tensor as ⊞ ⊞ , with the left matrix for the favored class ($a = 1$) and the right matrix for the disfavored class ($a = 0$). Since each component of $\mathbf{z}$ corresponds to some element of the fairness–confusion tensor, we shade each component that appears in the equation. Blue shading denotes the favored class, while red shading denotes the disfavored class. We further distinguish two kinds of dependencies. Components that have a nonzero coefficient in the matrix are shaded fully. However, the values of these coefficients themselves can depend on other components, albeit implicitly, and we shade these implicit components in a lighter shade. Putting this all together, we can represent DP in (3.1) graphically as 🟦 🟥, EOd as 🟦 🟥 $\wedge$ 🟦 🟥, PP as $($🟦 🟥$)^{(2)}$, with the superscript denoting the quadratic order of the term. As shown in the third column of Table 3.2, all group fairness notions can be effectively described in this notation.

Name of fairness	Definition and linear system	Terms in fairness–confusion tensor
Demographic parity (DP)	$\mathsf{Pr}(\hat{y}=1\vert\mathbf{a}=1)=\mathsf{Pr}(\hat{y}=1\vert\mathbf{a}=0)$ $\mathbf{A}_{\text{DP}}=\frac{1}{N}\begin{pmatrix}N_0 & 0 & N_0 & 0 & -N_1 & 0 & -N_1 & 0\end{pmatrix}$	🟦 🟥
Equality of opportunity (EOp)[47]	$\mathsf{Pr}(\hat{y}=1\vert y=1,\mathbf{a}=1)=\mathsf{Pr}(\hat{y}=1\vert y=1,\mathbf{a}=0)$ $\mathbf{A}_{\text{EOP}}=\frac{1}{N}\begin{pmatrix}M_0 & 0 & 0 & 0 & -M_1 & 0 & 0 & 0\end{pmatrix}$	🟦 🟥
Predictive equality (PE)[29]	$\mathsf{Pr}(\hat{y}=1\vert y=0,\mathbf{a}=1)=\mathsf{Pr}(\hat{y}=1\vert y=0,\mathbf{a}=0)$ $\mathbf{A}_{\text{PE}}=\frac{1}{N}\begin{pmatrix}0 & 0 & N_0-M_0 & 0 & 0 & 0 & -N_1+M_1 & 0\end{pmatrix}$	🟦 🟥
Equalized odds (EOd)[47]	EOp $\wedge$ PE	🟦 🟥 $\wedge$ 🟦 🟥
Equal false negative rate (EFNR) [2]	$\mathsf{Pr}(\hat{y}=0\vert y=1,\mathbf{a}=1)=\mathsf{Pr}(\hat{y}=0\vert y=1,\mathbf{a}=0)$ $\mathbf{A}_{\text{EFNR}}=\frac{1}{N}\begin{pmatrix}0 & M_0 & 0 & 0 & 0 & -M_1 & 0 & 0\end{pmatrix}$	🟦 🟥
Calibration within groups (CG)[67]	$\mathsf{Pr}(y=1\vert P_\theta(\mathbf{x})=s,\mathbf{a}=1)=\mathsf{Pr}(y=1\vert P_\theta(\mathbf{x})=s,\mathbf{a}=0)=s$ $\mathbf{A}_{\text{CG}}=\begin{pmatrix}1-v_1 & 0 & -v_1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1-v_0 & 0 & -v_0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1-v_1 & 0 & -v_1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1-v_0 & 0 & -v_0\end{pmatrix}$	🟦 ⊞ $\wedge$ 🟦 ⊞ $\wedge$ ⊞ 🟥 $\wedge$ ⊞ 🟥
Positive class balance (PCB)[67]	$\mathbb{E}(P_\theta\vert y=1,\mathbf{a}=1)=\mathbb{E}(P_\theta\vert y=1,\mathbf{a}=0)$ $\mathbf{A}_{\text{PCB}}=\min_a(M_a)\begin{pmatrix}\frac{v_1}{M_1} & \frac{v_0}{M_1} & 0 & 0 & -\frac{v_1}{M_0} & -\frac{v_0}{M_0} & 0 & 0\end{pmatrix}$	🟦 🟥
Negative class balance (NCB)[67]	$\mathbb{E}(P_\theta\vert y=0,\mathbf{a}=1)=\mathbb{E}(P_\theta\vert y=0,\mathbf{a}=0)$ $\mathbf{A}_{\text{NCB}}=\min_a(N_a-M_a)\begin{pmatrix}0 & 0 & \frac{v_1}{N_1-M_1} & \frac{v_0}{N_1-M_1} & 0 & 0 & -\frac{v_1}{N_0-M_0} & -\frac{v_0}{N_0-M_0}\end{pmatrix}$	🟦 🟥
Relaxed Equalized Odds (REod)[96]	$\alpha_0 FPR_0+\beta_0 FNR_0=\alpha_1 FPR_1+\beta_1 FNR_1$ $\mathbf{A}_{\text{REOD}}=\begin{pmatrix}0 & \frac{\beta_1}{M_1} & \frac{\alpha_1}{N_1-M_1} & 0 & 0 & \frac{\beta_0}{M_0} & \frac{\alpha_0}{N_0-M_0} & 0\end{pmatrix}/N$	🟦 🟥
Predictive parity (PP)[29]	$\mathsf{Pr}(y=1\vert\hat{y}=1,\mathbf{a}=1)=\mathsf{Pr}(y=1\vert\hat{y}=1,\mathbf{a}=0)$ $\frac{1}{2}\mathbf{z}^T\mathbf{B}_{\text{PP}}\mathbf{z}=(TP_1FP_0-TP_0FP_1)/N^2$	$($🟦 🟥$)^{(2)}$
Equal false omission rate (EFOR) [1]	$\mathsf{Pr}(y=1\vert\hat{y}=0,\mathbf{a}=1)=\mathsf{Pr}(y=1\vert\hat{y}=0,\mathbf{a}=0)$ $\frac{1}{2}\mathbf{z}^T\mathbf{B}_{\text{EFOR}}\mathbf{z}=(TN_1FN_0-TN_0FN_1)/N^2$	$($🟦 🟥$)^{(2)}$
Conditional accuracy equality (CA)[15]	PP $\wedge$ EFOR	$($🟦 🟥$)^{(2)}$ $\wedge$ $($🟦 🟥$)^{(2)}$

[1] To our knowledge, EFOR has not been described in literature in isolation, but is used in the definition of conditional accuracy equality (CA)[15].

[2] Defined implicitly in [29].

Table 3.2: Some common group fairness definitions and corresponding abbreviations used throughout the paper in terms of linear functions $\phi(\mathbf{z}) = \mathbf{Az}$ or quadratic functions $\phi(\mathbf{z}) = \frac{1}{2}\mathbf{z}^T\mathbf{Bz}$ that appear in the performance–fairness optimality problem (3.5). There are two groups separated by the horizontal line: those that are specified by linear functions (above), or quadratic functions (below). The graphical notation is described in Section 3.3. P_θ is the probability produced by a model (parameterized by θ) of $\hat{y} = 1$. The fairness functions ϕ are uniquely defined only up to a normalization factor and overall sign.

3.4 Optimization over the Fairness–confusion Tensor

The fairness–confusion tensor $\mathbf{z}$ allows for a succinct linear and quadratic characterization of group fairness definitions in the literature. We naturally consider the following family of optimization problems over $\mathbf{z} \in \mathcal{K}$, where the objective function is constructed so that the solution reflects trade-offs between fairness and performance.

Definition 3.4.1. *Let $f^{(i)} : \mathcal{K} \to [0, 1]$ be performance metrics (indexed by i) with best performance 0 and worst performance 1, $\phi^{(j)}(\mathbf{z})$ be fairness functions (indexed by j) with μ_i, λ_j be real constants with $\mu_0 = 1$. Then, the* performance–fairness optimality problem (PFOP) *is a class of optimization problem of form:*

$$\operatorname*{argmin}_{\mathbf{z}\in\mathcal{K}} \sum_{i\geq 0} \mu_i f^{(i)}(\mathbf{z}) + \sum_{j\geq 0} \lambda_j \phi^{(j)}(\mathbf{z}) \tag{3.5}$$

PFOP is a general optimization problem containing two groups of terms; the first quantifying performance loss; the second quantifying unfairness. The restriction $\mathbf{z} \in \mathcal{K}$ is necessary to ensure that $\mathbf{z}$ is a valid fairness–confusion tensor that obeys the requisite marginal sums. In our discussion below, it will be convenient to consider solutions with explicit bounds on their optimality.

Definition 3.4.2. *Let $\epsilon \geq 0$ and $\delta \geq 0$. Then, a* (ϵ, δ)-solution to the PFOP *is a* $\mathbf{z}$ *that satisfies* (3.5) *such that $\sum_j \lambda_j \phi^{(j)}(\mathbf{z}) \leq \epsilon$ and $\sum_i \mu_i f^{(i)}(\mathbf{z}) \leq \delta$.*

The parameters ϵ and δ represent the sum total of deviation from perfect fairness and perfect predictive performance respectively. Unless otherwise stated, the rest of the paper is dedicated to analyzing one of the simplest instantiations of PFOP, defined below.

Definition 3.4.3. *The* least-squares accuracy–fairness optimality problem (LAFOP) *is a PFOP with accuracy (or classification error rate) as the performance function $f^{(0)}$, and $K \geq 1$ fairness constraints in the form of a fairness matrix $\mathbf{A}$ (each row indexed by j), with*

$$\begin{aligned}
\phi^{(j)}(\mathbf{z}) &= (\mathbf{A}_{j,*}\mathbf{z})^2, \quad j = 0, ..., K-1 \\
f^{(0)}(\mathbf{z}) &= (\mathbf{c} \cdot \mathbf{z})^2, \\
\mathbf{c} &= (0, 1, 1, 0, 0, 1, 1, 0)^T, \\
\lambda &= \lambda_0 = ... = \lambda_{K-1}.
\end{aligned} \tag{3.6}$$

In other words, LAFOP is the problem

$$\underset{\mathbf{z} \in \mathcal{K}}{\operatorname{argmin}} \, (\mathbf{c} \cdot \mathbf{z})^2 + \lambda \|\mathbf{A}\mathbf{z}\|_2^2, \tag{3.7}$$

where $\mathbf{c} \cdot \mathbf{z}$ encodes the usual notion of classification error, and $\mathbf{A}$ encodes K linear fairness functions stacked together as the regularizer. A single hyperparameter λ specifies the relative importance of satisfying the fairness constraints while optimizing classification performance, with $\lambda = 0$ considering only performance and disabling all fairness constraints, and $\lambda = \infty$ imposing fairness constraints without regard to accuracy.

LAFOP is a convex optimization problem which is simple to analyze. Despite its simplicity, LAFOP encompasses many situations involving linear notions of fairness, allowing us to reason about multiple fairness constraints as well as fairness–accuracy trade-offs under versatile scenarios.

3.4.1 Reduction to a post-processing method for fair classification

PFOP and LAFOP do not assume anything about the model, therefore are designed to be model-agnostic. In this section we highlight the versatility of LAFOP by showing that adding a model-specific constraint on LAFOP reduces it to a post-processing algorithm for fair classification.

Post-processing method, in particular for EOd as introduced in Hardt et al. [47], solves the following optimization problem for $\tilde{Y}$, which is a post-processed, supposedly fair, classifier, given $\hat{Y}$, a vanilla classifier:

$$\min_{\tilde{Y}} \mathbb{E}l(\tilde{Y}, Y) \text{ such that } \gamma_0(\tilde{Y}) = \gamma_1(\tilde{Y}) \text{ and } \gamma_0(\tilde{Y}) \in P_0(\hat{Y}), \gamma_1(\tilde{Y}) \in P_1(\hat{Y}) \tag{3.8}$$

where $\gamma_a(\tilde{Y})$ represents EOd constraints for $\tilde{Y}$ as a tuple of (FPR_a, TPR_a), and $P_a(\hat{Y})$ is a model-specific set of feasible γ_a values, defined as $P_a(\hat{Y}) = \text{convhull}\{(0,0), \gamma_a(\hat{Y}), \gamma_a(1-\hat{Y}), (1,1)\}$. All the components of (3.8) can be rewritten in terms of $\hat{\mathbf{z}}$ and $\tilde{\mathbf{z}}$, the fairness–confusion tensors corresponding to the classifiers $\hat{Y}$ and $\tilde{Y}$ respectively. This yields a LAFOP over $\tilde{z}$ with additional model-specific constraints derived from $\hat{z}$ on the solution space. More formally, we have the following optimization problem for post-processing:

Definition 3.4.4. *Given a classifier to be post-processed and its corresponding fairness–confusion tensor $\hat{z}$, the* model-specific LAFOP *(MS-LAFOP) for EOd is the variant of LAFOP with model-specific constraints on the solution space as the following:*

$$\operatorname*{argmin}_{\tilde{\mathbf{z}}\in\hat{\mathcal{K}}} (\mathbf{c}\cdot\tilde{\mathbf{z}})^2 + \lambda\|\mathbf{A}_{\mathrm{EOD}}\tilde{\mathbf{z}}\|_2^2, \tag{3.9}$$

where

$$\hat{\mathcal{K}} = \left\{\tilde{\mathbf{z}} \geq 0 : \mathbf{A}_{\mathsf{const}}\tilde{\mathbf{z}} \ = \mathbf{b}_{\mathsf{const}}, \|\tilde{\mathbf{z}}\|_1 = 1, \beta_a(\tilde{z}) \in \textit{convhull}\,\{(0,0), \beta_a(\hat{z}), \beta_a(1-\hat{z}), (1,1)\}\,\forall a\right\}$$

with β_a expressing (FPR_a, TPR_a) tuples computed from the corresponding fairness–confusion tensor of group a.

From the solution of MS-LAFOP, it is possible to compute mixing rates for post-processing the given classifier. We note that MS-LAFOP can be extended to other group fairness notions as long as the model-specific constraints are accordingly set up for them. For more details, refer to Section B.7.3.

3.5 Incompatible Group Fairness Definitions

In this section, we show how LAFOP yields a more general view of understanding group fairness incompatibility results. As $\lambda \to \infty$, for linear fairness functions $\phi^{(i)}(\mathbf{z}) = \mathbf{A}^{(i)}\mathbf{z}$, LAFOP becomes equivalent to solving the following linear system of equations:

$$\begin{pmatrix} \mathbf{A}^{(0)} \\ \vdots \\ \mathbf{A}^{(K-1)} \\ \mathbf{A}_{\mathsf{const}} \end{pmatrix} \mathbf{z} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{b}_{\mathsf{const}} \end{pmatrix}, \ \mathbf{z} \geq 0, \tag{3.10}$$

Notice the compatibility of fairness conditions encoded by these K fairness matrices $\mathbf{A}^{(i)}$ is equivalent to having infinitely many solutions to the above linear system. We formally define (in)compatibility of fairness notions below based on this observation.

Definition 3.5.1. *Let $\Phi = \{\phi^{(i)}\}_{i=0}^{K-1}$ be a set of linear fairness functions, encoded in a fairness matrix $\mathbf{A}$ (of which each row corresponds to $\phi^{(i)}$), and let ρ be the number of solutions for the system in* (3.10). *If $\rho = 0$, then Φ is said to be incompatible. Otherwise, Φ is compatible. When Φ is incompatible, some additional set of constraints on the dataset or the model is required for it to be compatible.*

This means that in general, incompatibility results among the group fairness definitions can be proven simply by asking if and when solutions exist to their corresponding linear system of form (3.10).

3.5.1 The incompatibility involving CG

We introduce a general incompatibility result involving CG that leads to many other new results as well as the one from Kleinberg et al. [67].

Theorem 3.5.1. *Let $B = 2$ be the number of bins in the definition of calibration within groups fairness (CG) [67], and v_0, v_1 be the scores, with $0 \leq v_0 < v_1 \leq 1$, and $K > 1$ with $\phi^{(0)}(\mathbf{z}) = \mathbf{A}_{\mathrm{CG}}\mathbf{z}$. Then, the* *corresponding* (3.10) *has the only solution*

$$z_0 = \frac{1}{N(v_1 - v_0)} \begin{pmatrix} v_1(M_1 - N_1 v_0) \\ v_0(-M_1 + N_1 v_1) \\ (1-v_1)(M_1 - N_1 v_0) \\ (1-v_0)(-M_1 + N_1 v_1) \\ v_1(M_0 - N_0 v_0) \\ v_0(-M_0 + N_0 v_1) \\ (1-v_1)(M_0 - N_0 v_0) \\ (1-v_0)(-M_0 + N_0 v_1) \end{pmatrix}, \tag{3.11}$$

and only when

$$0 \le v_0 \le \min_a \left(\frac{M_a}{N_a}\right) \le \max_a \left(\frac{M_a}{N_a}\right) \le v_1 \le 1. \tag{3.12}$$

Otherwise, no solution exists.

Theorem 3.5.1 yields other extended results regarding the incompatibility of CG and other notions of fairness. As one canonical instance, simply substituting z_0 in (3.11) to the linear system of the form in (3.10) with PCB and NCB fairness matrices yields the following corollary, which is equivalent to the result presented in Kleinberg et al. [67] (proof is in Section B.2).

Corollary 3.5.1.1 (Re-derivation of [67])**.** *Consider a classifier that satisfies CG, PCB and NCB fairness simultaneously. Then, at least one of the following statements is true:*

1. *the data have equal base rates for each class a, i.e.* $M_0/N_0 = M_1/N_1$*, or*
2. *the classifier has perfect prediction, i.e.* $v_0 = 0$ *and* $v_1 = 1$.

Similar approach can be applied to derive incompatibilities of CG with other linear and quadratic notions of fairness as below (proofs in Section B.3, Section B.4).

Corollary 3.5.1.2. *(Linear notion of fairness: DP) Consider a classifier that satisfies CG and DP fairness simultaneously. Then, the data have equal base rates for each group* a.

Corollary 3.5.1.3. *(Quadratic notion of fairness: PP) Consider a classifier that satisfies CG and PP fairness simultaneously. Then, at least one of the following is true:*

1. $v_0 = (M_1 - M_0)/(N_1 - N_0)$.
2. $v_1 = 1$.

From Theorem 3.5.1 and its corollaries, we curate the extended incompatibility results involving CG in Table 3.3 along with conditions for compatibility. To our knowledge, all cases other than the bottom row of the table are new.

3.5.2 The incompatibility of {PE, EFNR, PP}

Using the same logic as the previous section, we re-derive an incompatibility result in Chouldechova [29] and provide more precise necessary conditions for compatibility. For details of the proof, refer to Section B.5.

Theorem 3.5.2 (Restatement of Chouldechova [29])**.** *Consider a classifier that satisfies* {*PE, EFNR, PP*}*. Then, at least one of these statements must be true:*

1. *The classifier has no true positives.*
2. *The classifier has no false positives.*
3. *Each protected class has the same base rate.*

Theorem 3.5.2 systematically shows that equal false positive rates, equal false negative rates, and predictive parity are compatible only under specific data/model-dependent circumstances, that were otherwise not clear in the original statements in Chouldechova [29].

Sets of fairness definitions	Necessary conditions
{CG, PP, DP, and any of EOp, PE, PCB, NCB, EFOR}	$M_0 = M_1$ and $N_0 = N_1$
{CG, DP, and any of EOp, PE, PCB, NCB, EFOR}	EBR only
{CG,EOp}, {CG,PCB}, {CG,EOp,PCB},{CG,EFOR,EOp}, {CG,EFOR,PCB},{CG,EFOR,EOp,PCB}	$v_0 = 0$ or EBR
{CG,PE}, {CG,NCB}, {CG,EOp,NCB}, {CG,EFOR,PE}, {CG,EFOR,NCB}, {CG,EFOR,EOp,NCB}	$v_1 = 1$ or EBR
{CG,EOd}[96], {CG, PCB, NCB} [67],{CG,EOd,PCB,NCB}, {CG,EFOR,EOd}, {CG,EFOR,PCB,NCB},{CG,EFOR,EOd,PCB,NCB}	($v_0 = 0$ and $v_1 = 1$) or EBR

Table 3.3: Some sets of fairness definitions containing Calibration(CG), which are incompatible in the sense of Definition 3.5.1 (left-column), together with their necessary conditions to be compatible (right column). EBR is the equal base rate condition, $M_0/N_0 = M_1/N_1$. For other abbreviations, refer to Table 3.2. These are all special cases of Theorem 3.5.1, while not exhaustive.

3.6 Experiments

In this section we show how the FACT diagnostic can practically show the relative impact of several notions of fairness on accuracy on synthetic and real datasets[3]. First we introduce FACT Pareto frontiers which characterize a model's achievable accuracy for a given set of fairness conditions, as a tool for understanding the trade-offs and contextualizing some recent works in fair classification (Section 3.6.2). We then explore a model-agnostic assessment of multiple fairness conditions via LAFOP (Section 3.6.3, Section B.7.2), as well as a model-specific assessment of post-processing methods in fair classification via MS-LAFOP (Section 3.6.4, Section B.7.3).

3.6.1 Datasets

We study a synthetic dataset similar to that in Zafar et al. [145], consisting of two-dimensional features along with a single binary protected attribute that is either sampled from an independent Bernoulli distribution ("unbiased" variant, denoted **S(U)**), or sampled dependent on the features ("biased" variant, denoted **S(B)**). The synthetic dataset consists of two-dimensional data $\mathbf{x} = (x_0, x_1)$ that follow the Gaussian distributions

$$\begin{aligned} \mathbf{x}|y = 1 \sim &\mathcal{N}\left(\begin{pmatrix}2\\2\end{pmatrix}, \begin{pmatrix}5 & 1\\1 & 5\end{pmatrix}\right) \\ \mathbf{x}|y = 0 \sim &\mathcal{N}\left(\begin{pmatrix}-2\\-2\end{pmatrix}, \begin{pmatrix}10 & 1\\1 & 3\end{pmatrix}\right). \end{aligned} \tag{3.13}$$

For the S(U) dataset, the protected attribute value is independent of $\mathbf{x}$ and y, and is instead distributed according to the Bernoulli distribution $a \sim \mathcal{B}\left(\frac{1}{2}\right)$. This notion of fairness was described in [22].

For the S(B) dataset, the protected attribute value is assigned as $a|\mathbf{x} = \mathrm{sgn}(x_0)$, which corresponds to a situation when some features (but not all) encode a protected attribute.

We also study the UCI Adult dataset [35], a census dataset used for income classification tasks where we consider sex as the protected attribute of interest.

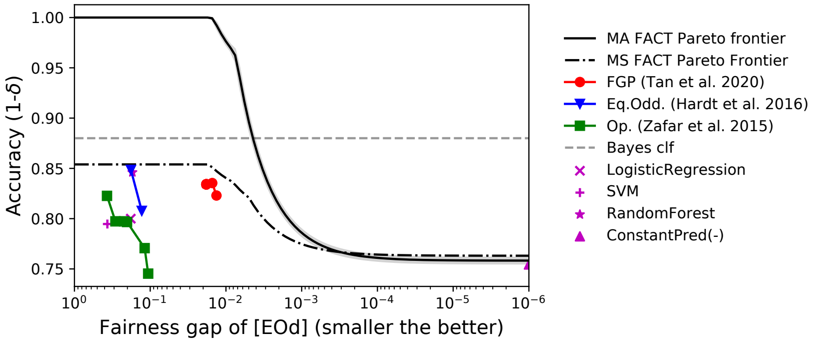


Figure 3.1: Model-agnostic (MA) and model-specific (MS) FACT Pareto frontiers of equalized odds on the Adult dataset. Three fair models (FGP, Eq.Odd., Op.) are shown in context by varying the strength of the fairness condition imposed, along with some baseline models (LR, SVM, RF, ConstantPrediction). The MA frontier should be interpreted relative to the Bayes error because it is oblivious to it — $\delta = 0$ means that the upper bound of the accuracy is the accuracy of the Bayes classifier, not 1. The MS frontier on the other hand provides realistic more bounds.

3.6.2 FACT Pareto frontiers

With LAFOP and MS-LAFOP, one can naturally consider a FACT Pareto frontier of accuracy and fairness by plotting (ϵ, δ) values of the (ϵ, δ)-solutions. In this section, we want to highlight the use of this frontier in the context of several published results in the literature as well as its implications.

The FACT Pareto frontier can be computed both in model-agnostic (MA) and model-specific (MS) scenarios by solving LAFOP and MS-LAFOP respectively, and Figure 3.1 shows such example on the Adult dataset for EOd fairness. We also consider three fair classification models: **FGP** [125], **Op.** [145], and **Eq.Odd.** [47], individually representing three different approaches one can take in training fair models (imposing fairness before, during, or after training). Some baseline models (logistic regression, SVM, random forest) are also plotted for reference, and a perfectly fair classifier (ConstantPredict: predicting all instances to be negative) on the bottom right corner is considered as an edge-case.

It is important to note that the MA FACT Pareto frontier should be interpreted as characterizing the model's achievable accuracy *relative* to the Bayes error (i.e., the degree to which the added fairness constraints adversely impact the Bayes error), which in this case is empirically estimated at around 0.12 from a wide range of ML models that have been tested on the Adult datset [25]. This relatively less realizable bound calls for a model-specific counterpart, the MS FACT Pareto frontier, which limits the frontier to be derived from a given pre-trained classifier. As shown in Figure 3.1, it indeed provides a more reasonable frontier for the models considered.

Placing different types of classifiers on the frontier, it is easy to visually grasp strengths and weaknesses of each models. FGP seems to outperform all other models in terms of the trade-off, while Op and EqOdd suffer more from early accuracy drops. The frontier further informs that for any model trained, only for fairness gaps below 10^{-2} will the accuracy start to suffer. Such understanding of the trade-offs will be helpful in anticipating practical limitations of models to be trained, as well as in comparing multiple models to determine which is better-suited for different situations.

In the rest of the following sections and figures, for the model-agnostic analysis, δ should be interpreted in reference to the Bayes error, i.e $\delta = 0$ means that the upper bound of the best-achievable accuracy

[3] Code available at `github.com/wnstlr/FACT`

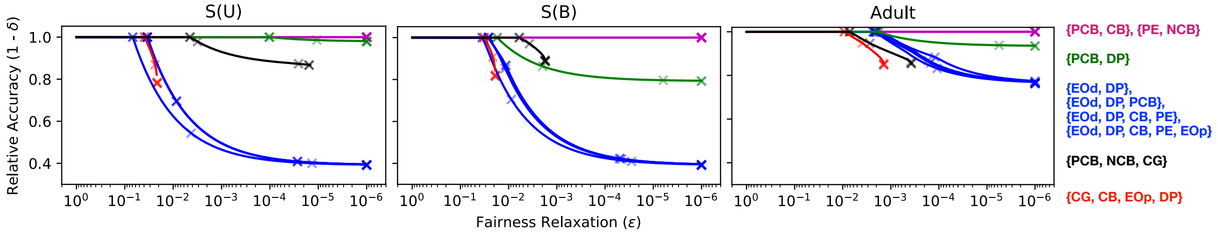


Figure 3.2: Model-agnostic FACT Pareto frontier for different groups of fairness notions (colored and grouped according to their convergence value as $\epsilon \to 0$) for three datasets (Section 3.6.1). The bottom two groups of fairness notions are incompatible (black, red), hence the halted trajectories before reaching smaller values of ϵ. Similar convergence behaviors within the fairness groups in blue reflect the dominance of $\{$EOd, DP$\}$ – any additional fairness notions added on top of these have no impact on the convergence value. Best viewed in color.

is the accuracy of the Bayes classifier, not 1.

3.6.3 Model-agnostic scenario with multiple fairness conditions

We are now interested in how a *group* of fairness conditions simultaneously affect accuracy. This can be assessed by looking at the shape of the MA FACT Pareto frontier of LAFOP with multiple fairness constraints, particularly δ values of (ϵ, δ)-solutions when ϵ is varied to be zero (or very close to it) on multiple fairness notions. Figure 3.2 shows this in two different ways: (i) (ϵ,δ)-solutions obtained when fairness conditions are imposed as hard inequality constraints instead of as regularizers, i.e. solving $\mathrm{argmin}_{\mathbf{z}\in\mathcal{K}} (\mathbf{c}\cdot\mathbf{z})^2$ s.t. $\|\mathbf{A}\mathbf{z}\|_2^2 \leq \epsilon$ (solid line), and (ii) (ϵ,δ)-solutions obtained from the LAFOP (3.7) while varying λs (crosses). Different groups of fairness notions are colored according to their convergence behaviors.

Similar trajectories and convergence of the curves allow us to identify fairness notions that come "for free" given some others, in terms of additional accuracy drops. In other words, the Pareto frontiers are effective at demonstrating the relative strength of the fairness notions within a group. For instance, under $\{$EOd, DP$\}$ (third group, blue) the best attainable accuracy drops by over 60 percent for S(U) and S(B), but we also observe that adding CB, PE, and/or PCB on top of them causes no additional accuracy drop – $\{$EOd, DP$\}$ essentially determines δ for the entire group of fairness notions in blue.

The MA FACT Pareto frontiers for multiple fairness conditions also show not only the existing incompatibility of the fairness notions, but also how much relaxation is required for them to be approximately compatible. The halted trajectories before hitting much smaller ϵ for the bottom two groups in black and red clearly verify this. Because the S(U) dataset has a smaller base rate gap between the groups compared to the Adult or the S(B) dataset by design, the incompatibility in S(U) becomes only visible at a much smaller ϵ value.

Taking a more macroscopic perspective, the MA FACT Pareto frontiers also show which dataset allows overall better trade-off scheme compared to the others. Because the S(U) dataset was designed to be less biased compared to the S(B) dataset, it exhibits significantly smaller drop in overall accuracy, particularly for the green group involving DP. The way S(U) was designed aligns with this observation, as the sensitive attributes were randomly sampled independently from the features. However, EOd and DP together (in blue) drives down the accuracy just like the biased counterpart, which demonstrates how conservative EOd fairness is for these datasets.

More observations and experiments are presented in Section B.7.2. It is possible to further extend these analyses to an arbitrary number of fairness constraints imposed on LAFOP, as well as to other performance metrics like precision or recall as seem fit.

3.6.4 Model-specific scenario with post-processing methods

While the MA FACT Pareto frontier shows a broader trade-off landscape for any classifiers, model-specific analysis using MS-LAFOP in (3.9) can be helpful in practice with more reasonable MS Pareto frontiers. Also after solving the MS-LAFOP, its solution can be used to compute the mixing rates for post-processing any given classifier just like done in Hardt et al. [47]. For more details, refer to Section B.7.3.

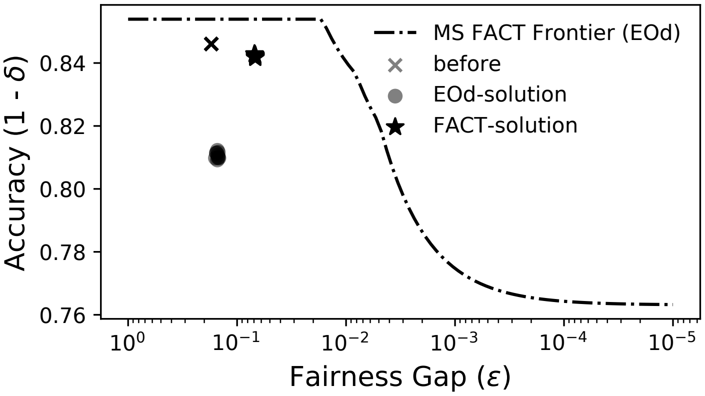


Figure 3.3: Model-specific FACT Pareto frontier of EOd on Adult dataset. Compared to the model-agnostic frontier, it yields a more realizable bounds on the trade-off between fairness and accuracy. Post-processed solutions for the given classifiers (crosses) using the algorithm in [47] (circles, EOd-solution) and FACT (stars, FACT-solution) are also shown. The FACT-solutions suffer significantly less from the trade-off, yielding competitive accuracy to the original classifiers while achieving smaller fairness gaps compared to the EOd-solutions.

Figure 3.3 shows the MS FACT Pareto frontier of EOd computed from MS-LAFOP for the Adult dataset (it is a zoomed-in version of the MA FACT Pareto frontier in Figure 3.1). We also plot two types of post-processed classifiers: EOd-solutions using the algorithm in Hardt et al. [47] (circles), and FACT-solutions using MS-LAFOP (stars). EOd solutions undergo steeper trade-off while the FACT-solutions are able to find a better configuration with smaller fairness gaps, retaining a competitive accuracy level to the original classifier (cross).

Chapter 4

Testing Model Explanations for Correctness

4.1 Introduction

Saliency methods are a popular tool to help understand the behavior of machine learning models. Given a model and an input image, these methods output a feature attribution that indicates which pixels they deem to be most "important" to the model's prediction [83, 117, 124]. Then, a natural question to ask is: *how do we define "important" and subsequently evaluate the efficacy of these methods?*

One intuitive approach to answering this question is to measure how well a saliency method locates the expected pixels of interest in the input image. In fact, this so-called "pointing game" [149] is one of the predominant evaluations used today [10, 27, 44, 108, 138, 153]. Current versions of this evaluation rely on external knowledge to define an *expected* feature attribution that highlights the region that a human would expect to be important for the given task. Then, the quality of a saliency method is measured using the overlap between its output and this expected feature attribution by metrics such as Intersection-Over-Union (IOU) (See Section 4.4).

Unfortunately, this approach has two key limitations. First, the results are unreliable when the model's ground-truth reasoning does not match human expectations, e.g., when the model is relying on spurious correlations. This is particularly problematic because detecting such discrepancies is one of the motivating use cases of saliency methods. Second, existing versions are based on relatively simple object classification tasks where we expect only a single region of the image, i.e., the object itself, to be relevant to the prediction. In practice, there exist more complex tasks, e.g., in medical imaging or autonomous driving, where considering interactions among multiple regions of the image may be necessary for the model to achieve high predictive accuracy.

These two limitations highlight the same fundamental concern: we do not know a priori what or how complex the model's reasoning will be, irrespective of how simple we think the underlying task is. For instance, the top panel of Figure 4.1 considers the seemingly simple task of identifying a baseball bat in an image. Based on the description of the task, we might expect the model to use *simple reasoning*, which we define as relying on a single region of the image, e.g., the bat itself, to make its prediction. If this is the case, the expected feature attribution should highlight the bat only. However, if the model actually uses more *complex reasoning*, which we define as relying on interactions among multiple regions of the image, e.g., using the presence of a hitter and a glove to identify a bat, the actual ground-truth feature attribution should highlight the hitter and the glove, not the bat. As illustrated through this example, the correct evaluation of saliency methods fundamentally depends on the model's ground-truth reasoning.

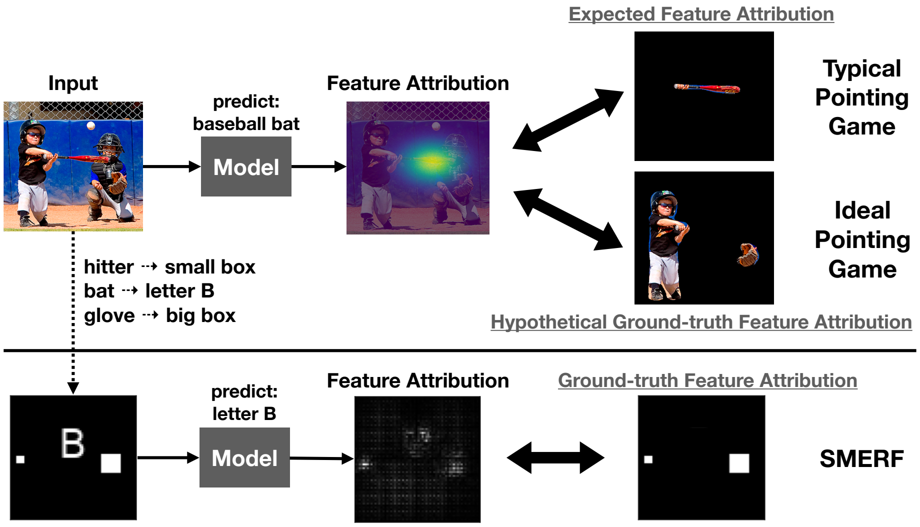


Figure 4.1: **Top.** Existing pointing game evaluations do not have access to ground-truth feature attributions but instead rely on *expected* feature attributions. For instance, while a model trained to identify a baseball bat is expected to rely on the baseball bat region of the image (top), it may rely on a more complex reasoning by using the presence of a hitter and a glove to identify a bat (bottom). **Bottom.** SMERF constructs a synthetic set of tasks that are stylized versions of real object classification tasks. Consider the task of identifying the letter 'B' in an image, where the letter and the two boxes correspond to the bat, the hitter, and the glove, respectively. SMERF controls the model's underlying reasoning via simulations over different data distributions, providing ground-truth feature attributions used to evaluate saliency methods (in this example, a model is relying on two boxes to identify the letter).

Consequently, we aim to address these key limitations by controlling the model's ground-truth reasoning. To do this, we start by generating synthetic images composed of simplified objects and consistent backgrounds that are stylized versions of real-world scenarios (e.g., Figure 4.1 Bottom). Then, by controlling the distribution and label of these images, we can induce and then verify a specific ground-truth reasoning for the model. By repeating this process for different levels of reasoning complexity, we build a benchmark called **S**imulated **M**od**E**l **R**easoning Evaluation **F**ramework (SMERF) that can evaluate saliency methods against ground-truth model reasoning.

Using SMERF, we consider seven distinct model reasoning settings with varying complexity, and perform an extensive evaluation of 10 leading saliency methods for each setting. Our analyses are summarized in Figure 4.2 and discussed at length throughout Section 4.4. We observe that for simple reasoning settings, leading saliency methods perform reasonably well on average, though still exhibit certain failure cases (Section 4.4.1). We further observe clear performance degradation as we increase model reasoning complexity. Indeed, in all complex reasoning settings, none of the methods meet our (lenient) definition of correctness[1], and all of them demonstrate acute failure cases (Section 4.4.2, 4.4.3).

Our results highlight major limitations of existing saliency methods, especially given the relative simplicity of SMERF's synthetic evaluation tasks. We further illustrate how SMERF's synthetic evaluations translate to more natural images, by presenting qualitatively similar yet generally worse results on analogous reasoning tasks that leverage natural image backgrounds instead of synthetic ones (Section 4.4.4).

[1] While we view the IOU > 0.5 as a lenient definition of correctness in synthetic settings, this value is commonly used in practice when evaluating on real tasks [38, 136].

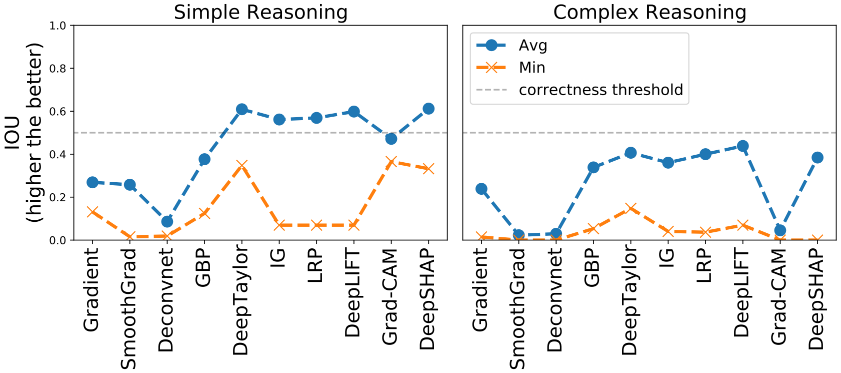


Figure 4.2: Summary of ground-truth-based evaluation of saliency methods via SMERF. **Left.** In simple reasoning settings, where the model relies on a single region of the image to make its prediction, average performance (blue) is reasonably good for most of the methods. However, all methods still demonstrate failure cases as shown by minimum performance (orange) over various tasks. **Right.** In more complex reasoning settings, where the model relies on interactions among multiple regions of the image, average performance drops with more acute failure cases.

Source code supporting the contents of this chapter can be found here: `https://github.com/wnstlr/SMERF`.

4.2 Related Work

Pointing Game Evaluation. The pointing game, which measures how well a saliency method identifies the relevant regions of an image, is one of the predominant ways to evaluate the efficacy of these methods [10, 27, 44, 108, 138, 153]. Many existing pointing game evaluations lack access to ground-truth model reasoning but instead rely on expected feature attributions generated by domain experts. Intuitively, this might appear to be reasonable by observing that the model has high test accuracy and concluding that it must be using the correct reasoning. However, datasets often contain spurious correlations and, as a result, a model may be able to achieve high test accuracy using incorrect reasoning. Consequently, these evaluations have confounded the correctness of the explanation with the correctness of the model. SMERF eliminates this confounding factor by leveraging the model's ground-truth reasoning, which allows us to demonstrate that several methods previously deemed to be effective are in fact sometimes ineffective for more complex model reasoning.

Adebayo et al. [2], Yang and Kim [142] try to address this same limitation using semi-synthetic datasets where the ground-truth reasoning is known by combining the object from one image with the background from another. Both analyses are based on the simple reasoning setting and, in that setting, our results roughly corroborate theirs. However, our analysis extends to more complex reasoning settings and demonstrates that methods that worked in the simple reasoning setting mostly perform much worse in these settings. It is important to consider the complex reasoning setting because we do not know how complex the model's reasoning is in practice (e.g., a model may rely on a spurious correlation and use complex reasoning for a simple task).

A concurrent work by Zhou et al. [155] introduces a similar semi-synthetic pipeline for testing saliency methods, where a family of image manipulations is applied to the input so that the ground-truth impact of specific features on the model prediction is known. Whereas Zhou et al. [155] focuses on model reasoning that relies on a single artificial feature in the image, we establish a complementary criteria for saliency methods by focusing on a more diverse set of model reasoning complexity induced by interactions among different features in the image.

Direct Criticisms of Saliency Methods. Adebayo et al. [1] uses two sanity checks that measure the statistical relationship between a saliency method and the model's parameters or the data it was trained on. They found that only a few methods (i.e., Gradient [117] and Grad-CAM [108]) passed these tests. Kindermans et al. [66] similarly tests several saliency methods for input invariance and finds that the Gradient satisfies the property while other methods generally do not. While `SMERF` is orthogonal to such types of analyses, it demonstrates that even methods that pass these tests have failure cases (as shown in Figure 4.2). Shah et al. [109] suggests that a model's adversarial robustness impacts how well the Gradient is able to correctly focus only on the relevant features. `SMERF` verifies this observation for simple reasoning, and further shows that the problem persists in complex reasoning settings even for robust models for all tested methods.

Other Evaluations. Beyond the pointing game, several proxy metrics have been proposed to evaluate saliency methods [4, 7, 11, 51]. Additionally, Liu et al. [82] introduces a benchmarking framework based on synthetic datasets sampled from different types of Gaussian distributions to evaluate the methods on these proxy metrics. However, Tomsett et al. [127] shows that several popular proxy metrics inherently depend on subtle hyperparameters that are not well understood and that this leads to analyses with inconsistent results. The unreliability of these popular proxy metrics further emphasizes the advantage of using a more intuitive evaluation like `SMERF`. Similar setups with humans in the loop are also being introduced to measure the user's perception of the feature attributions in detecting model biases [119].

4.3 Methods

`SMERF` is a synthetic evaluation benchmark where several types of ground-truth model reasoning, ranging from simple to complex, are generated to test a saliency method's ability to recover them. We first describe several types of model reasoning that are motivated by real-world examples and then captured in `SMERF`'s synthetic family of datasets called `TextBox`. We then explain the data generation and training process in `SMERF` with `TextBox`. Additional details are in Appendix C.1.

4.3.1 Types of Simple and Complex Model Reasoning

We are interested in evaluating the performance of saliency methods in capturing both simple and complex model reasoning, where the complexity of reasoning is defined based on whether the model relies on a *single* or *multiple* regions of the image. We next describe three different ways in which a model may exhibit simple or complex reasoning characterized by the model's reliance on the true set of features (X) and/or the set of spurious features (F).

Consider the model trained to detect a baseball bat from Figure 4.1. The model may correctly rely on the true set of features (e.g. the bat), without relying on spurious features (e.g. the glove or the hitter). We denote models that exhibit no reliance on spurious features as the *No-Reliance (NR)* setting (Figure 4.3, left). Another model could instead depend on the existence of the glove and the hitter, thus fully relying on spurious features, a setting we call *Full-Reliance (FR)* (Figure 4.3, middle). Finally, the model may rely on both the true and the spurious sets of features (Figure 4.3 right), a setting we denote as *Conditional-Reliance (CR)*. For instance, a model may learn to rely on the glove only when the hitter is present, but otherwise on the bat itself for the prediction.

CR by default prescribes complex reasoning due to its conditional nature. In contrast, the complexity

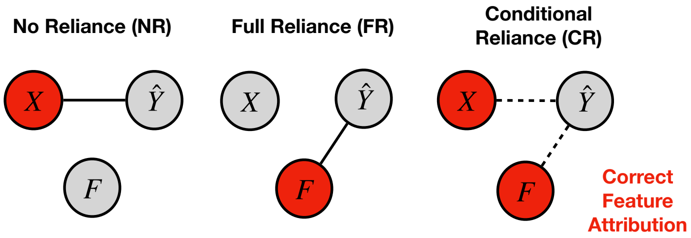


Figure 4.3: Given true features X and spurious features F in the training data, the model may exhibit simple or complex reasoning depending on how it relies on X and/or F. No-Reliance (NR) and Full-Reliance (FR) denote settings where the model relies solely on X or F, respectively. Conditional Reliance (CR) denotes settings where the model depends on both X and F. `SMERF` allows us to control the model's reasoning and to thus evaluate saliency methods against the ground-truth feature attribution (denoted in red) derived from this underlying reasoning.

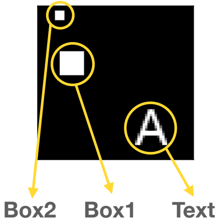


Figure 4.4: Features in the `TextBox` datasets.

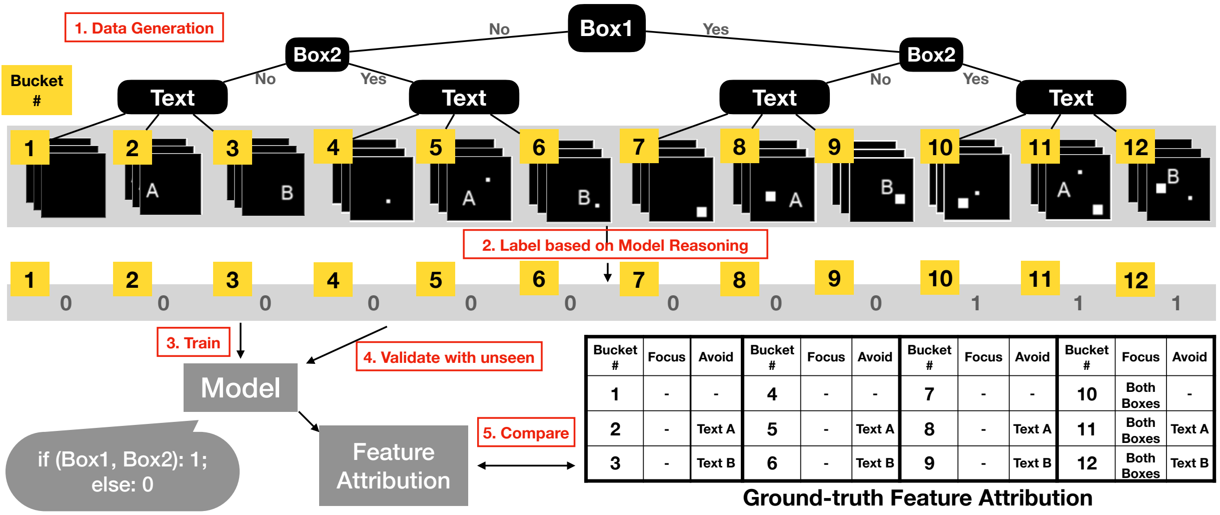
of model reasoning for NR and FR depends on how many objects are included in X and F: simple when X and F each consists of features corresponding to a single object, and complex when they consist of multiple objects. It is notable that existing pointing game evaluations in the literature are performed with respect to a simple reasoning under the assumption that the model exhibits NR, with X being the single object of interest [108, 153]. Moreover, previous controlled setups for evaluating saliency methods were limited to simple model reasoning in the FR setting, e.g., setting F as the background and X as a single object of interest [2, 142].

`SMERF` instantiates simple and complex model reasoning across these three settings by creating a family of datasets called `TextBox`. These datasets all consist of 64-by-64 pixel images with black background that include three types of white objects in random locations of each image (Figure 4.4): **Text**, "A", "B"; **Box1**, a 10-by-10 box; and **Box2**, a 4-by-4 box. `SMERF` simulates the objects' relationship with the labels (as in Figure 4.3) to control the model reasoning for X = Text, $F = \{\text{Box1}, \text{Box2}\}$. Note that the choice of these features are arbitrary – they can be replaced by other shapes, colors, and backgrounds.

4.3.2 Training Models with Ground-truth Reasoning

`SMERF` first creates an appropriate training dataset for a particular desired model reasoning. Specifically, `SMERF` starts by generating 12 *buckets*[2] of images (as shown in Figure 4.5), where each bucket contains images with particular X and F, and an associated label designated by the desired model reasoning. More formally, considering the joint distribution $p(X, F, Y)$, each bucket will be composed of images with features $X = x$, $F = f$ along with the label $y \sim p(Y|X = x, F = f)$ determined by the

[2] The total number of buckets depends on the cardinality of X and F, as we create buckets for all possible (X, F) value pairs; hence for `TextBox` datasets we consider 12 buckets: there are three different values for Text (Nothing, 'A', or 'B'), two for Box1, and two for Box2, resulting in a total of 12 distinct combinations (Appendix C.1.1).



Bucket #	Focus	Avoid	Bucket #	Focus	Avoid	Bucket #	Focus	Avoid	Bucket #	Focus	Avoid
1	-	-	4	-	-	7	-	-	10	Both Boxes	-
2	-	Text A	5	-	Text A	8	-	Text A	11	Both Boxes	Text A
3	-	Text B	6	-	Text B	9	-	Text B	12	Both Boxes	Text B

Figure 4.5: Workflow of SMERF for a model with FR on Box1 and Box2, where the model predicts 1 if both boxes are present, otherwise 0 (Complex-FR in Table 4.1). Twelve buckets of images each composed of different sets of features are generated. These are then labeled according to the model reasoning. The model is then trained/validated on samples from each bucket. The ground-truth feature attribution should focus/avoid certain objects in the image, as labels depend on specific objects only, e.g. labels do not depend on Text, but only on both Boxes (shown in the table). Feature attributions from saliency methods are compared against this ground-truth.

Name	Simple-FR	Simple-NR	Complex-FR	Complex-CR1	Complex-CR2	Complex-CR3	Complex-CR4

Reasoning (How buckets are labeled)	if B1: 1;
	otherwise: 0
		if T=A: 0;
		if T=B: 1
			if B1 & B2: 1;
			otherwise: 0
				if B2 & B1: 1;
				if B2 & NO B1: 0;
				if NO B2 & T=A: 0;
				if NO B2 & T=B: 1
					if B1 & B2: 1;
					if B1 & NO B2: 0;
					if NO B1 & T=A: 0;
					if NO B1 & T=B: 1
						if B2 & T=A: 0;
						if B2 & T=B: 1;
						if NO B2 & B1: 1;
						if NO B2 & NO B1: 0
							if B1 & T=A: 0;
							if B1 & T=B: 1;
							if NO B1 & B2: 1;
							if NO B1 & NO B2: 0

# of Buckets	12	8	12	10	10	10	10
ID of Buckets Labeled 0	1,2,3,4,5,6	2,5,8,11	1,2,3,4,5,6,7,8,9	2,4,5,6,8	2,5,7,8,9	1,2,3,5,11	1,2,3,8,11
ID of Buckets Labeled 1	7,8,9,10,11,12	3,6,9,12	10,11,12	3,9,10,11,12	3,6,10,11,12	6,7,8,9,12	4,5,6,9,12
ID of Undefined Buckets	None	1,4,7,10	None	1,7	1,4	4,10	7,10
Objects to Focus	1,2,3,4,5,6: N 7,8,9,10, 11,12: B1	2,3,5,6,8, 9,11,12: T	1,2,3,4,5 ,6,7,8,9: N 10,11,12: B1,B2	2,3,8,9: T 4,5,6: B2 10,11,12: B1, B2	2,3,5,6: T 7,8,9: B1 10,11,12: B1, B2	1,2,3: N 5,6,11,12: B2,T 7,8,9:B1	1,2,3: N 4,5,6: B2 8,9,11,12:B1,T
Objects to Avoid	1: N 2,3,8,9: T 4,10: B2 5,6,11,12: B2, T	2,3: N 5,6: B2 8,9: B1 11,12: B1,B2	1,4,7,10: N 2,3,5,6,11,12: T	2,3,4,10: N 8,9: B1 5,6,11,12: T	2,3,7,10: N 5,6: B2 8,9,11,12: T	1,5,6,7: N 2,3,8,9: T 11,12: B1	1,4,8,9: N 2,3,5,6: T 11,12: B2

Table 4.1: The seven model reasoning settings considered in the experiments (Section 4.4). Each column represents a model reasoning setting that belongs to one of the three categories depicted in Figure 4.3. **B1** stands for Box1, **B2** stands for Box2, **T** stands for Text, and **N** stands for None (see Figure 4.4). Depending on the reasoning, there are different numbers of buckets that belong to the positive/negative classes, and different objects that feature attributions should focus on/avoid. Bucket ID numbers are taken from Figure 4.5, which corresponds to the setting described in the third column of this table (Complex-FR). See Appendix C.1.1 for more details.

specified model reasoning, with different images in each bucket varying by the location of the features. A convolutional neural network[3] is trained on the entire set of buckets, which is then validated with unseen data points from each bucket to ensure that the ground-truth model reasoning has been properly learned.

[3] We consider a shallow CNN, AlexNet [72], and VGG16 [116] for the model architectures. The results in Section 3.6 are from the shallow CNN, and we observe similar results from other deep architectures, as reported in Section 4.4.3 and Appendix C.2.6.

Because the data distribution is simulated, we can generate arbitrary number of images from different buckets which differ only in terms of a single feature and use them to confidently verify that specific features are responsible for the model's prediction. Repeating this for all possible subsets of features effectively ensures that the model follows the intended reasoning. Ground-truth feature attributions are derived from this verified model reasoning and are later used for evaluating saliency methods.

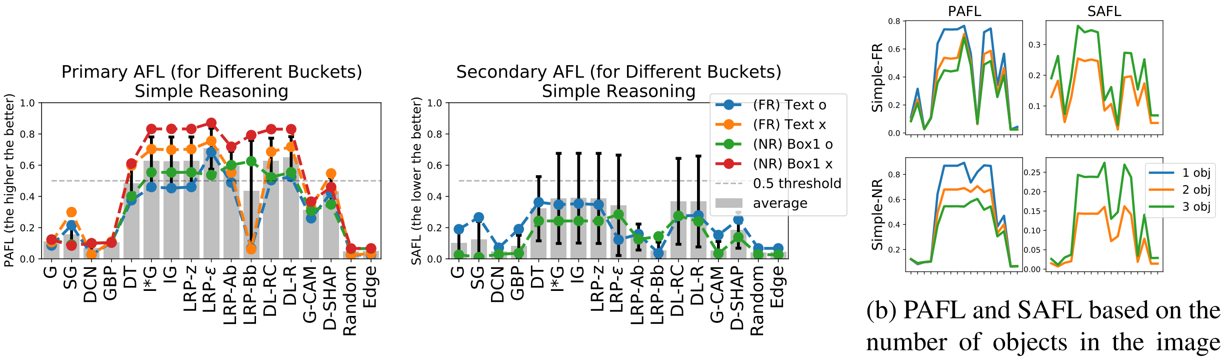
Figure 4.5 depicts the steps of dataset generation and model training/validation for complex model reasoning with FR. In this example, we want the labels to depend only on the presence of both boxes, thus providing positive labels only for the three buckets that include both boxes (buckets 10-12), and negative labels for the nine buckets that include at most one box (buckets 1-9). We verify that the model learns the desired model reasoning, as it achieves near-perfect accuracy on unseen samples from each bucket. The ground-truth model reasoning provides ground-truth feature attribution shown in the table of Figure 4.5, defining regions the saliency methods should focus and/or avoid for images from each buckets. This information will later be used to evaluate the feature attributions obtained from the saliency methods.

4.4 Experiments

We use SMERF and the TextBox datasets to show the performance of leading saliency methods for different types of model reasoning presented in Table 4.1 with varying complexity (the example from Figure 4.5 corresponds to Complex-FR in the third column). For simple reasoning (Section 4.4.1), we find that saliency methods perform reasonably well (with a few exceptions), which is generally consistent with previous pointing game evaluations. However, we observe a general trend of decreasing performance when input images become more "saturated" (i.e., filled with more objects), even when the model's underlying reasoning does not change. For complex reasoning (Section 4.4.2), the average performance for all methods decreases, with several failure cases due to methods focusing more on irrelevant objects. As a result, feature attributions qualitatively become indistinguishable across different model reasoning, raising practical concerns since users in general do not know the type of reasoning being used (and in fact would potentially rely on saliency methods to get this information). We further show that a similar trend is present when varying factors like model architecture choice and robust training (Section 4.4.3). And when the images contain more natural backgrounds (Section 4.4.4), even lower worst-case performance is observed. Additional details about the results are in Appendix C.2.

Saliency Methods and Baselines. We use a modified version of the open-source library iNNvestigate[3] which includes several implementations of leading saliency methods. We use the following methods: Gradient (G) [117], SmoothGradients (SG) [120], DeConvNet (DCN) [147], Guided Backpropagation (GBP) [122], Deep Taylor Decomposition (DT) [91], Input*Gradient (I*G) [114], Integrated Gradients (IG) [124], and Layerwise Relevance Propagation (LRP) [11] (four variations: LRP-z, LRP-ϵ, LRP-A$\flat$, LRP-B$\flat$), DeepLIFT (DL) [114] (two variations: DL-RC (using Reveal-Cancel rule), DL-R (using Rescale rule)), Grad-CAM (G-CAM) [108], and DeepSHAP (D-SHAP) [83]. We also add some simple baselines, like Random (random-valued feature attribution) and Edge-detection (Edge), both of which are model-independent and therefore are not useful in understanding the model reasoning.

Evaluation Metrics. Typical pointing game evaluations measure the performance of saliency methods with the Intersection-Over-Union (IOU) metric [38, 153]. This metric computes the ratio of the area of the intersection to the area of the union of the binary-masked feature attribution and the ground-truth feature attribution. The binary-masked feature attribution is obtained by first blurring the original feature attribution averaged across the three color channels, followed by thresholding the pixel intensity to select the top-K pixels, where K is equal to the number of pixels that contain the object of interest. Given the popularity of this metric, we perform extensive experiments with it: Figure 4.2 summarizes the results,



(a) PAFL and SAFL for Simple Reasoning

(b) PAFL and SAFL based on the number of objects in the image (x-axis is the same as Figure 4.6a)

Figure 4.6: **(a)** PAFL (left) and SAFL (right) for Simple-FR and Simple-NR (Table 4.1). The black vertical lines indicate the standard deviation across different buckets. Most methods perform well on average, with reasonable PAFL and low SAFL. The methods on the left part of the plot (G, SG, CGN, CBP) are the exception. The colored lines indicate average performance for different buckets that have either Text or Box1 (i.e. the irrelevant features) present (o) or absent (x). **(b)** PAFL decreases and SAFL increases as the number of objects (regardless of their relevance to the prediction) increases, which indicates that the methods perform worse as images become "saturated."

which are consistent with those of our main results.

However, we find that IOU loses information about raw attribution values on extra objects when thresholding to generate the binary-masked feature attribution for evaluation. It is therefore more likely to disregard non-trivial signals from extra objects in the image (see Appendix C.1.4).

To address these issues, we also consider a previously introduced [134] but unnamed metric that we call *Attribution Focus Level (AFL)*. This metric quantifies the proportion of the total attribution values that are assigned to the region of interest. Given the raw, normalized feature attribution values, it is the sum of values assigned to pixels inside the region of interest. Intuitively, values near 1 indicate a stronger level of focus on the region of interest. We define a threshold value of 0.5, chosen to indicate that more than half of the total attribution values is focused on the object, to roughly distinguish good and bad performance in terms of AFL.

To better account for the relationship among multiple features in the image, we use two types of AFL: (1) *Primary AFL (PAFL)*, which measures the level of focus on the *correct* (primary) features that are relevant for the prediction (corresponding to objects to focus in Table 4.1 and Figure 4.5), and (2) *Secondary AFL (SAFL)*, which measures the same quantity for the *incorrect* (secondary) features that are irrelevant for the prediction (corresponding to objects to avoid in Table 4.1 and Figure 4.5, excluding background). Notably, the sum of PAFL and SAFL is upper bounded by 1, so PAFL $>$ 0.5 implies that PAFL $>$ SAFL, which further indicates that the feature attribution correctly focuses more on the relevant features than the irrelevant ones (our definition of "success"). Conversely, when SAFL $>$ PAFL, the feature attribution incorrectly focuses more on irrelevant regions and is thus considered undesirable (our definition of "failure").

4.4.1 Simple Reasoning Setting

For a high-level understanding of how the methods generally perform, we plot PAFL and SAFL averaged across all buckets for simple reasoning instantiated with Simple-FR and Simple-NR (grey vertical bars in Figure 4.6a). We observe that most of the methods, except for G, SG, DCN, and GBP, perform

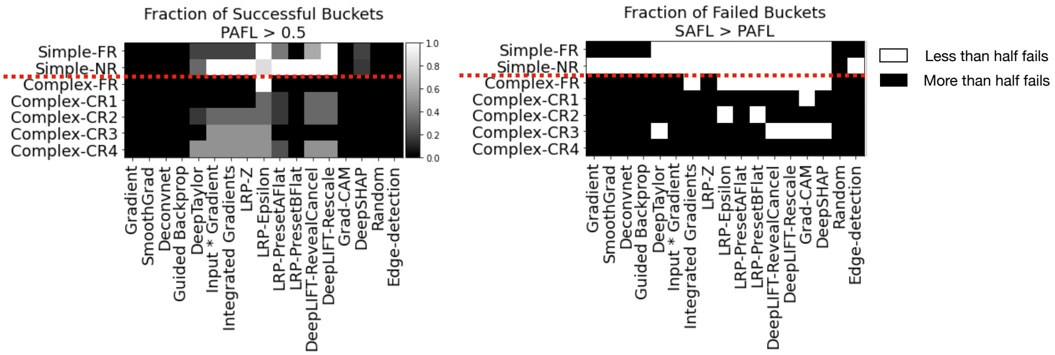
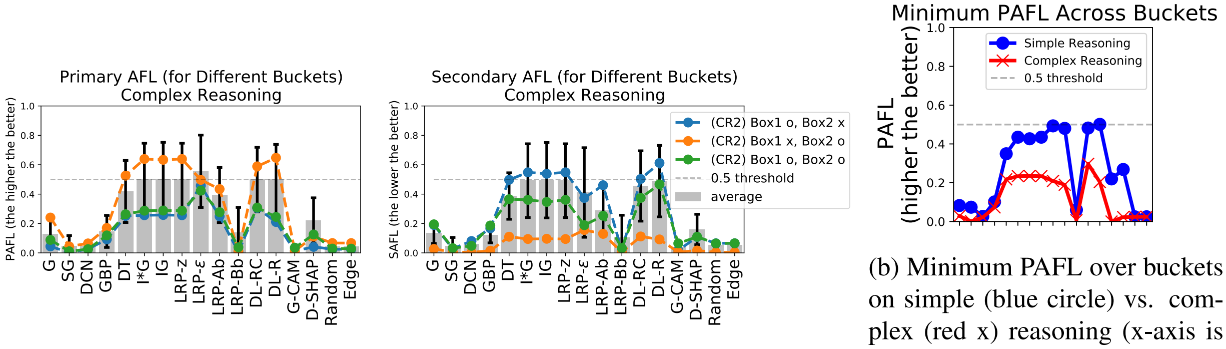


Figure 4.7: **Top:** The fraction of buckets for which the methods are successful (i.e., PAFL $>$ 0.5). Most success cases (white) are concentrated on simple reasoning setting. **Bottom:** The cases where the method fails due to wrong focus (i.e., SAFL $>$ PAFL) on more than half of the buckets for each reasoning (colored with black). Most failure cases are concentrated on complex reasoning, which aligns with increasing SAFL observed in Figure 4.8a.



(a) PAFL and SAFL for Complex Reasoning

(b) Minimum PAFL over buckets on simple (blue circle) vs. complex (red x) reasoning (x-axis is the same as Figure 4.8a).

Figure 4.8: **(a)** PAFL and SAFL for complex reasoning. Average performance on PAFL is mostly lower than 0.5, with worse performance compared to the results for simple reasoning. Further, SAFL increases significantly (to the point where it is on par with PAFL for some methods). Per-bucket performance variation for Complex-CR2 (from Table 4.1) is plotted with colored lines, each color corresponding to the methods' performance on a bucket with either Box1 or Box2 present (o) or absent (x). **(b)** Worst-case buckets show worse PAFL values in complex reasoning (red) with a sharp drop from simple reasoning setting (blue).

reasonably well, with PAFL exceeding the 0.5 correctness threshold and SAFL being lower. This reasonable level of performance aligns with what existing evaluations in the literature have shown with simple reasoning [2, 108].

Despite their reasonable performance in average, we observe a trend that PAFL decreases and SAFL increases as more objects are visible in the image, even though the model reasoning remains simple. The colored lines in Figure 4.6a exemplify this trend by showing per-bucket performance for two different buckets in Simple-FR (blue and orange) and Simple-NR (green and red) each. For both reasoning, all methods show lower PAFL on buckets where an irrelevant object (Text for Simple-FR in blue and Box1 for Simple-NR in green) is present, compared to buckets where that object is absent (orange and red). This means that part of the feature attribution originally assigned to the relevant object shifts to the irrelevant ones when they are visible in the image. Figure 4.6b verifies this trend for all buckets containing different number of visible objects: PAFL from buckets with fewer objects strictly upper bounds PAFL from buckets with more objects, while SAFL increases along with the number of objects. These variations in AFL based on the number of objects in the image lead to non-trivial variance across buckets as shown with black vertical error bars (standard deviation) in Figure 4.6a. Qualitative examples confirm this undesirable dependence of methods' AFL values on the number of objects in the image (Appendix C.2.2).

To better view the success/failure cases, we record the fraction of buckets that contain both relevant and irrelevant features for which each method is considered successful and indicate it with a color ranging from black (0) to white (1) (Figure 4.7 Top). For simple reasoning (the first two rows), methods in the middle (I*G through DL) have relatively higher success rates overall. Among these, DL-R is the only method that succeeds in *all* buckets for *both* types of simple reasoning. Methods like G, SG, DCN, and GBP fail to be successful for all buckets.

4.4.2 Complex Reasoning Setting

We consider 5 types of complex reasoning in Table 4.1: Complex-FR, and Complex-CR1 through Complex-CR4. Compared to the simple reasoning case, PAFL drops for all methods (grey vertical bars in Figure 4.8a). Methods with strong performance in simple reasoning settings (IG, LRP, and DL) narrowly meet the correctness threshold for complex reasoning settings, while those with decent performance (G-CAM and D-SHAP) suffer from bigger drops in PAFL. Methods like G, SG, DCN, and GBP, which showed weak performance in simple reasoning settings, continue to show low PAFL.

In addition to lower average PAFL overall, we see an increase in SAFL for complex reasoning. In some cases, SAFL approaches the 0.5 threshold, demonstrating a clear failure by focusing on the wrong object(s) more (grey bars in Figure 4.8a right). This is immediately verified in Figure 4.7 (top), where with a single exception (LRP-ϵ on Complex-FR), *none* of the methods are successful in all of the buckets for complex reasoning (rows below the red dotted line). Figure 4.7 (bottom) further demonstrates that the majority of the buckets show failures for complex reasoning (colored with black), contrary to simple reasoning (mostly white).

We next measure the worst-case performance of different methods (i.e., by evaluating the worst-performing buckets for each method, as we also visualized in Figure 4.2). We observe that the worst-case PAFL for complex reasoning is much lower than that for simple reasoning (Figure 4.8b). Qualitative samples from these worst-performing buckets speak for the low PAFL with clear lack of focus on the relevant features (Appendix C.2.2).

Moreover, we observe that per-bucket performance variation is more extreme in complex reasoning settings. For instance, per-bucket performance on Complex-CR2 (blue, orange, and green lines in Figure 4.8a) shows that most methods are successful in a bucket where only Box2 is present (orange), while for other buckets with only Box1 (blue) or both Boxes (green), all methods clearly fail. Such variation is also visible across other types of complex reasoning (Appendix C.2.3). These altogether contribute to higher variance of performance for complex reasoning settings (indicated with black vertical error bars in Figure 4.8a), higher than the simple reasoning settings (Figure 4.6a).

Finally, we contextualize the aforementioned failure cases of saliency methods from a practitioners' perspective, who are not aware of the type of model reasoning used, but are relying on the feature attributions for this information. By failing to point to only the correct set of features, these methods are likely to mislead practitioners. Figure 4.9 visualizes the feature attributions from IG (one of the best methods in our evaluation) for each model reasoning (see Appendix C.2.5 for other methods). Essentially all objects are highlighted in all samples, and it is thus not clear how to discern the underlying model reasoning from any of them. For example, Box1 appears to be the most important object according to the feature attributions in the third row, even when this object is clearly not part of the model reasoning for Simple-NR (3rd column) and Complex-CR1 (5th column).

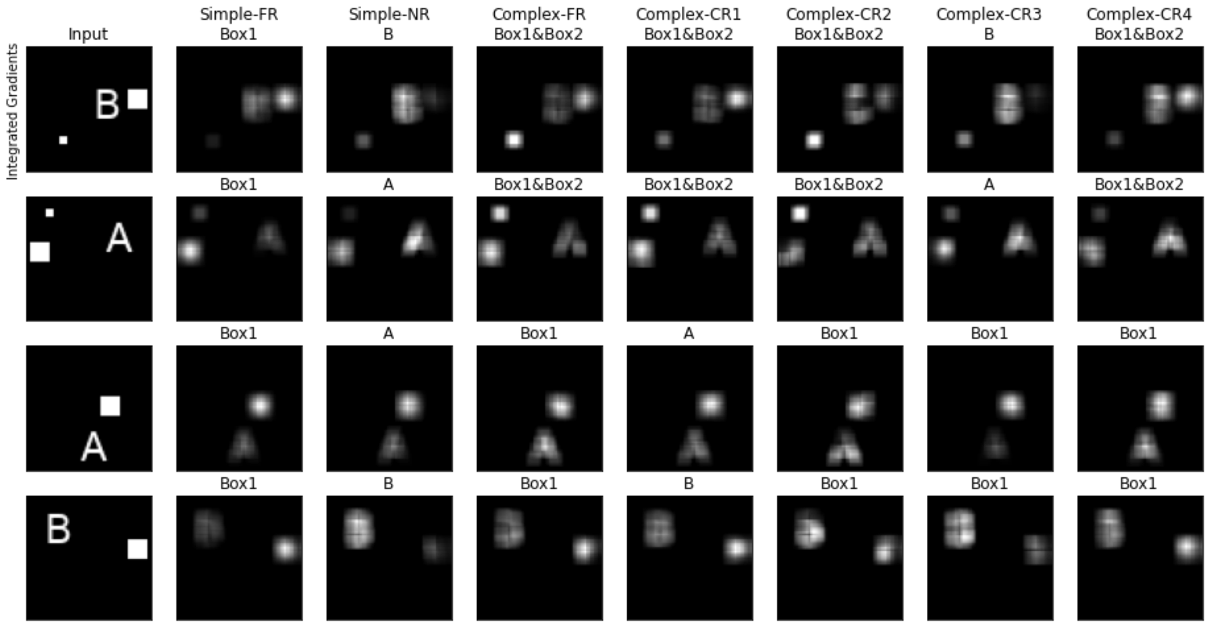


Figure 4.9: Feature attributions from Integrated Gradients for different model reasoning on four inputs from different buckets, labeled with their relevant features to highlight. Essentially all objects in the image are highlighted, making it difficult to identify which type of model reasoning is used for each column.

4.4.3 Impact of Model Architecture and Robustness

We next vary the model's architecture and its robustness and show that the trends we observed in previous sections persist.

Model Architecture. We repeat the same set of experiments with AlexNet [72] and VGG16 [116] to confirm that the trends we observe for the saliency methods are not the artifact of model architecture choice. Figure 4.10 shows similar trends we have observed so far: both the average (blue lines) and the worst-case performance drops (orange lines) as the reasoning becomes more complex (comparing solid against dotted lines). For more details, see Appendix C.2.6.

Adversarial Robustness. It has been previously suggested that Gradient feature attribution applied on adversarially robust models tend to better ignore the signals from spurious objects in the image [109]. We verify that this trend for Gradient feature attribution is somewhat true, yet the general problem persists for most of the methods even for robust models [86], in both simple and complex reasoning settings (Figure 4.11). While Gradient (G) and SmoothGradient (SG)'s performance on robust models (solid lines) is higher compared to plain models (dotted lines) for simple reasoning, we still observe performance drops in complex reasoning throughout all methods. For more details, see Appendix C.2.7.

4.4.4 Extending to Natural Image Backgrounds

In previous sections we focused on images with uniform black background. With this black background, we aimed to make it easy for the model to identify the objects present in the image. We also expected that this simplistic background would improve the performances of the saliency methods[4].

However, we can simulate more realistic scenarios by replacing the black background with natural images. To this end, we replace the background of the `TextBox` datasets with real images of baseball

[4] We also run experiments using random noise pixels for the background and observe performance drops (Appendix C.2.8).

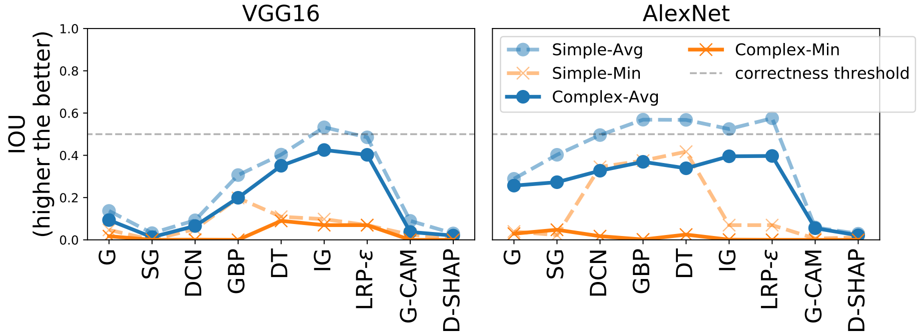


Figure 4.10: IOU results on VGG16 (Left) and AlexNet (Right). For both models, all methods show drops in both average (blue) and worst-case (orange) performance on complex reasoning (solid lines) compared to simple reasoning (dotted lines).

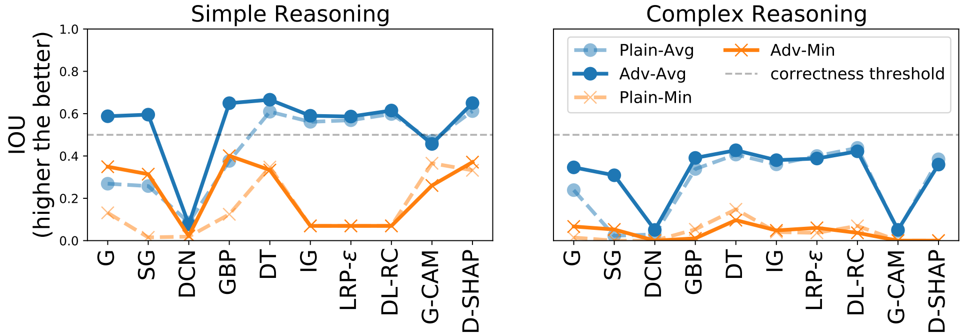


Figure 4.11: IOU results on robust models trained against PGD attacks (solid lines) and plain models (dotted lines, taken from Figure 4.2) for simple (left) and complex reasoning settings (right). Robust models show higher average performance for Gradient and SmoothGradient in simple reasoning compared to plain models, as suggested in [109]. Nevertheless, there still are performance drops for complex reasoning across all methods.

stadiums, chosen to simulate tasks that are more similar in spirit to the one depicted in the top panel of Figure 4.1, sampled from the Places dataset [154], while still reasoning over the same objects (Text, Box1, Box2), and repeat the experiments in Sections 4.4.1 and 4.4.2.

Figure 4.12 shows the summary of results on images with real backgrounds (solid lines), comparing them to what was observed in Sections 4.4.1 and 4.4.2 on images with black backgrounds (dotted lines, identical to Figure 4.8b). There are two types of performance drop observed. First, we observe a similar drop as we move from simple (blue solid line) to complex (red solid line) reasoning for both the black and real background settings. Second, we see a general performance drop in the real background setting as compared to the black background setting. For instance, in the real background setting, all methods are far from the threshold even for simple reasoning settings (solid blue line), which differs from results in the black background setting (dotted blue line). See Appendix C.2.9 for more details.

Collectively, these results suggest that the performance of these saliency methods is likely to further deteriorate under even more realistic, noisier scenarios. They thus highlight the importance of consistent success in controlled (synthetic) settings as a stepping stone to success in real-world settings.

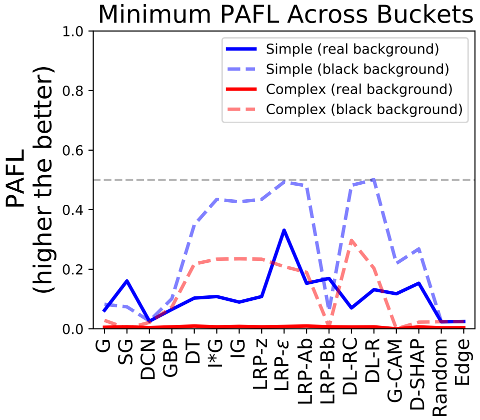


Figure 4.12: Minimum PAFL comparison between cases with uniform black background (dotted lines) and real background (solid lines). We observe a decrease in performance when moving from simple (blue) to complex reasoning (red) even for the real background case. Due to more noise from the background, the overall PAFL values are lower for the real background case.

Chapter 5

Assisting Human Decisions in Document Matching

5.1 Introduction

An important application in which human decision makers play a critical role, is document matching, i.e., when a *query document* needs to be matched to one of the many *candidate documents* from a larger pool based on their relevance. Concrete instances of this setup include: *academic peer review*, where meta-reviewers—associate editors in journals (e.g., `https://jmlr.org/tmlr/ae-guide.html`) or area chairs in conferences [111] or program directors conducting proposal reviews [59]—are asked to assign one or more candidate reviewers to submitted papers with relevant expertise based on their previously published work (illustrated in Figure 5.1, solid arrows); *recruitment*, where recruiters screen through a list of resumes from candidate applicants for an available position at the company [97, 106]; and *plagiarism detection*, where governing members (e.g., ethics board members of a conference, instructors of a course) review submissions to determine the degree of plagiarism [42]. Because the pool of candidate documents is typically large and the decision makers have limited time, they first use automated matching models to pre-screen the candidate documents. These matching models typically base their screening on affinity scores, which measure the relevance of each candidate document to the query document [5, 26, 30, 77]. The human decision makers subsequently determine the best-matching document, taking both their expertise and the affinity scores computed by the matching models into account. Such intervention by human decision makers is required for such tasks, as often times either errors made by the models are so consequential that they warrant human oversight, or the overall performance can be considerably improved by incorporating the domain knowledge of human experts.

Despite the growing prevalence of automated matching models and human decision makers working jointly for such practical matching tasks, humans generally find it difficult to completely rely on the models due to a lack of assistive information other than the models' output itself. For instance, in peer review, 20% of the meta-reviewers from past NLP conferences found the affinity scores from the matching model to be *"not very useful or not useful at all"* in a recent survey [126]. The survey also reports that the affinity scores rank the least important for the respondents, compared to more tangible and structured information about the candidate reviewers such as whether they have worked on similar tasks, datasets, or methods. Additionally, the survey finds that providing just the affinity scores increases the meta-reviewers' workload as they *"have to identify the information they need from a glance at the reviewer's publication record."* and *"are presented with little structured information about the reviewers."* Similarly, in hiring, the recruiters need to manually evaluate more profiles further down the search result pages due to too

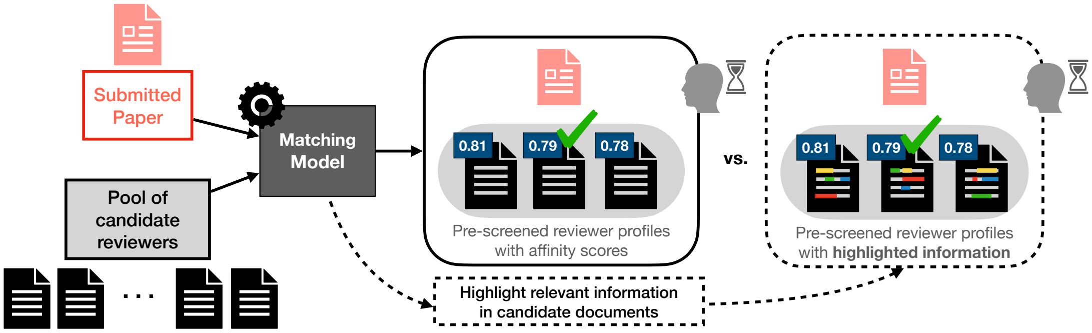

Submitted Paper
Pool of candidate reviewers
Matching Model
0.81
0.79
0.78
Pre-screened reviewer profiles with affinity scores
vs.
Pre-screened reviewer profiles with highlighted information
Highlight relevant information in candidate documents

Figure 5.1: An example document matching application of peer review. For each submitted paper, the matching model pre-screens a list of candidate reviewers via affinity scores (solid arrows). Meta-reviewers, typically under a time constraint, then select the best match to the submitted paper among the pre-screened reviewer (box with a solid line). We study whether providing additional assistive information, namely highlighting potentially relevant information in the candidate documents, can help the meta-reviewers make better decisions (dotted arrows and boxes). We do so by focusing on a proxy matching task on a crowdsourcing platform that is representative of real-world applications not limited to peer review, including recruitment and plagiarism detection which follow the similar setup with different documents and decision makers.

generalized matches suggested by the model [77].

To address the lack of additional assistive information in the document matching setup, we conduct the first evaluation of what additional information can help the human decision makers to find matches *accurately* and *quickly* (Figure 5.1, dotted arrows). To do so, we first design a proxy task of summary-article matching that is representative of the general setup so that several methods providing different types of assistive information can be readily tested at scale via crowdsourced users (Section 5.3.1). The choice of proxy task addresses the logistical difficulty and expenses of directly experimenting with real domain-specific decision makers.

On this proxy task, we explore different classes of methods that have been previously suggested as tools for users to understand model outputs or document content. To standardize the format of assistance, we focus on methods that highlight assistive information within the candidate documents that the decision makers can utilize for matching (Section 5.3.2):

- SHAP [83], a popular *black-box model explanation* [28, 34], highlights input tokens in the document that contribute both positively and negatively to the affinity scores. The utility of SHAP on several concrete downstream tasks remain controversial with conflicting results [6, 53, 58], and has yet to be evaluated for its effectiveness in document matching.
- BERTSum [81], a state-of-the-art *text summarization method*, which highlights key sentences in the candidate documents to help reduce the user's cognitive load for the task.
- Two task-specific methods, that we design ourselves (Section 5.3.2), to highlight details in the candidate documents relevant to the details in the query (by using sentence and phrase-level similarity measures).

With assistive information provided by these methods as treatments, and a control group provided with just the affinity scores and no additional assistive information, we conduct a pre-registered user study (with 271 participants) on a crowdsourcing platform.[1] The study finds that (Section 5.4):

- Despite its usage in numerous applications, SHAP decreases the participants' matching performance

[1] Pre-registration document is available here: https://aspredicted.org/LMM_4K9

compared to the control group.

- Contrary to the expectation that summarizing long articles could improve task efficiency, the summaries generated by BERTSum adversely impact the participants. Participants take longer to finish and are less accurate compared to the control group.
- Our task-specific methods, which are tailored to better identify details useful for the task, help the participants to be quicker and more accurate compared to the control group.
- An overwhelming number of participants in *all* treatment groups perceive that the highlighted information is helpful, whereas the quantitative performance (accuracy and time) says otherwise.

The results suggest the benefits of designing task-specific assistive tools over general black-box solutions, and highlight the importance of quantitative evaluation of the methods' utility that is grounded on a specific task over subjective user perceptions [28]. The code from the study is available at `https://github.com/wnstlr/document-matching`.

5.2 Related Work

Prior Evaluation of Assistive Information. We discuss how our proposed evaluation of different types of assistive information, which include affinity scores, black-box model explanations, and text summaries, differs from how they have been previously evaluated.

Affinity scores, computed by comparing the similarity of representations learned by language models, are commonly used in practice to rank or filter the candidate documents [26, 30, 90, 103, 128, 137] Their quality has been evaluated both with or without human decision makers: some may evaluate them based on the user's self-reported confidence score [90], while others may use performance from proxy tasks like document topic classification, where a higher test accuracy of the classification model using the learned representation indicates better ability to reflect more meaningful components in the documents [30]. However, the utility of affinity scores for assisting human decision makers for the document matching task is less studied.

While information provided by black-box model explanations have been evaluated for their utility to assist human decision makers in various downstream tasks, the results have been lackluster. On the deception detection task, where users are asked to determine if a given hotel review is fake or not, prior work have shown that only some explanation methods improve a user's task performance [73, 74]. Arora et al. [9] further show that none of the off-the-shelf explanations help the users better understand the model's decisions on the task. On more common NLP tasks like sentiment classification and question-answering, providing explanations to the users decreases the task performance compared to providing nothing when the model's prediction is incorrect [12]. For the fraud detection task with domain experts, providing some model explanations showed conflicting effects on improving the performance [6, 53]. In this work, we expand user evaluations of black-box model explanations to the document matching task and propose alternatives that could be more helpful.

Summaries generated by text summarization models [76, 81, 112, 150] are typically either evaluated by metrics like ROUGE with respect to the annotated ground-truth summary in a standardized dataset, or by a human's subjective rating of the quality. To the best of our knowledge, the usefulness of these automatically summarized information to the human decision makers in concrete downstream tasks is rarely studied. Even for a few applied works that utilize these methods to practical documents in legal or business domains, the final evaluations do not explore beyond these task-independent metrics [13, 37, 52]. In this work, we explicitly evaluate whether the generated summaries can help improve the decision makers' task performance in document matching.

Practical Concerns in Document Matching Applications. There are a number of real-world document matching applications including peer review, hiring, and plagiarism detection. For each application, we discuss practical issues that have been raised by users that can be mitigated by providing more assistive information about the data and the model.

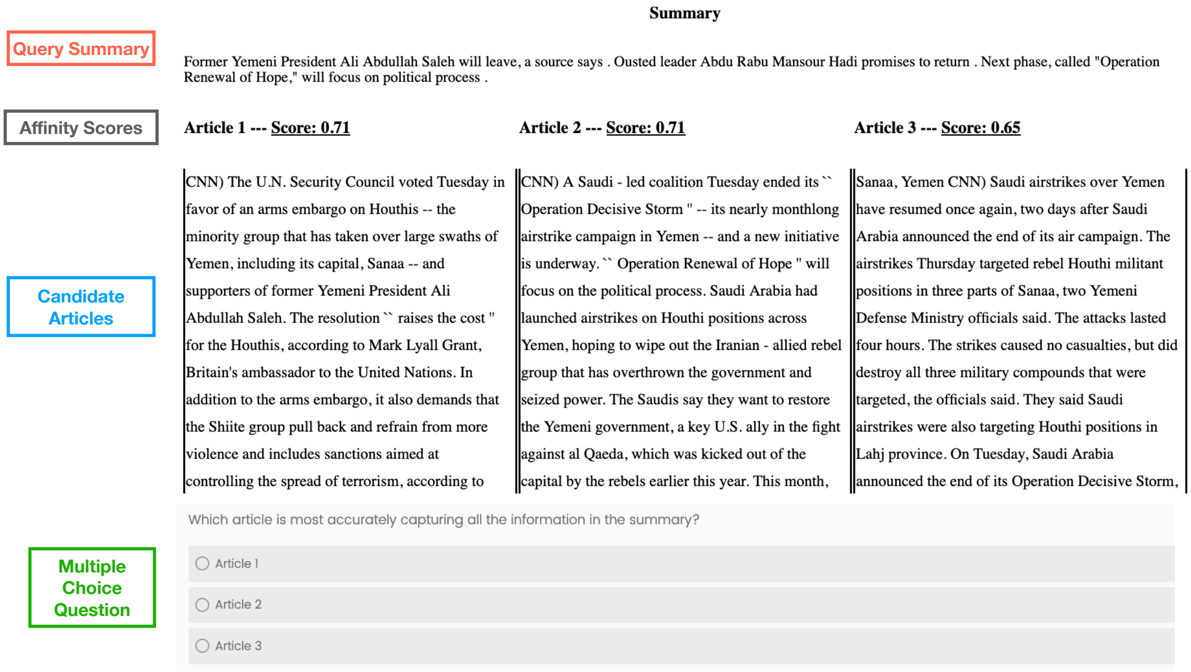
In scientific peer review, submitted papers need to be matched to appropriate reviewers with proper expertise or experience in the paper's subject area. First, a set of candidate reviewers are identified using an affinity scoring model based on representations learned by language models [26, 30, 90, 103, 128, 137]. Additional information such as reviewer bids or paper/reviewer subject areas may also be elicited [41, 88, 110]. Based on this information, meta-reviewers may either be asked to directly assign one or more reviewers to each paper, or to modify the assignment that has been already made as they see appropriate. For example, in the journal Transactions on Machine Learning Research, for any submitted paper the meta-reviewer (action editor) is shown a list of all non-conflicted reviewers sorted according to the affinity scores. The meta-reviewer may also click on any potential reviewer's name to see their website or list of publication. The meta-reviewer is then required to assign three reviewers to the paper based on this information. However, a recent survey of meta-reviewers from past NLP conferences reveal that the affinity scores alone are not as useful, and most respondents prefer to see more tangible and structured information about the reviewers [126].

In hiring, many companies resort to various algorithmic tools to efficiently filter and search for suitable candidates for a given job listing [17, 40, 97]. Many recruiters, while using these tools, express difficulties in reconciling a mismatch between algorithmic results and the recruiter's own assessments. This is mainly attributed to "too generalized and imprecise" relevant matches suggested by the model, which lead to more "manually evaluating more profiles further down the search result pages" increasing the task completion time [77]. Also, the general lack of understanding about the algorithmic assessments makes the recruiters more reluctant to adopt them.

In plagiarism detection, many existing software tools aim to reduce the governing members' workload by providing detailed information about the match, e.g., what specific parts of the query document are identical or similar to parts of the candidate documents. However, their performance in identifying various forms of plagiarism (e.g., ones involving paraphrasing or cross-language references) is still limited [55]. Also many existing tools lack user-friendly presentation of information that can better assist the task [43]. As the governing members need to ultimately assess the proposed evidence by the model to determine the degree of penalty [42], additional assistive information about the match may improve their experience.

5.3 Task Setup and Methods

In Section 5.3.1, we describe the design of a summary-article matching task, which is an instance of the document matching tasks. We use this task as a proxy for other document matching tasks (e.g., matching reviewers to papers in peer-review), as it is more amenable for crowdsourcing experiments at scale. The summary-article matching task addresses common difficulties encountered when directly experimenting on real-world applications like recruiting real domain-specific decision makers (e.g., meta-reviewers in academia), building on complex systems in practice (e.g., internal systems that govern workflows in academic conferences), and coordinating logistical issues (e.g., longer turnaround for receiving feedback for each paper assignment). Our task may also be useful for early prototyping and validation of different methods. Then in Section 5.3.2, we present existing and our proposed methods that provide assistive information that we evaluate with human users on the the summary-article matching task.



Summary

Former Yemeni President Ali Abdullah Saleh will leave, a source says . Ousted leader Abdu Rabu Mansour Hadi promises to return . Next phase, called "Operation Renewal of Hope," will focus on political process .

Article 1 --- Score: 0.71	Article 2 --- Score: 0.71	Article 3 --- Score: 0.65
CNN) The U.N. Security Council voted Tuesday in favor of an arms embargo on Houthis -- the minority group that has taken over large swaths of Yemen, including its capital, Sanaa -- and supporters of former Yemeni President Ali Abdullah Saleh. The resolution `` raises the cost '' for the Houthis, according to Mark Lyall Grant, Britain's ambassador to the United Nations. In addition to the arms embargo, it also demands that the Shiite group pull back and refrain from more violence and includes sanctions aimed at controlling the spread of terrorism, according to	CNN) A Saudi - led coalition Tuesday ended its `` Operation Decisive Storm '' -- its nearly monthlong airstrike campaign in Yemen -- and a new initiative is underway. `` Operation Renewal of Hope '' will focus on the political process. Saudi Arabia had launched airstrikes on Houthi positions across Yemen, hoping to wipe out the Iranian - allied rebel group that has overthrown the government and seized power. The Saudis say they want to restore the Yemeni government, a key U.S. ally in the fight against al Qaeda, which was kicked out of the capital by the rebels earlier this year. This month,	Sanaa, Yemen CNN) Saudi airstrikes over Yemen have resumed once again, two days after Saudi Arabia announced the end of its air campaign. The airstrikes Thursday targeted rebel Houthi militant positions in three parts of Sanaa, two Yemeni Defense Ministry officials said. The attacks lasted four hours. The strikes caused no casualties, but did destroy all three military compounds that were targeted, the officials said. They said Saudi airstrikes were also targeting Houthi positions in Lahj province. On Tuesday, Saudi Arabia announced the end of its Operation Decisive Storm,

Which article is most accurately capturing all the information in the summary?

○ Article 1

○ Article 2

○ Article 3

Figure 5.2: Interface for our summary-article matching task, an instance of the general document matching task. For each question, the participants are provided with the summary, three candidate articles to select from, and affinity scores for each candidate. The articles here are abridged to save space.

5.3.1 Instantiating Document Matching

In the general document matching task, a matching model pre-selects a set of candidate documents based on the affinity scores, which capture the relevance between the query document and the candidate document. These affinity scores facilitate filtering candidates from a large pool of documents, but are nevertheless prone to errors. The user therefore goes over the candidate documents with the scores and selects the most relevant candidate document. A practical concern which we would like to address is when the affinity scores from the matching model alone may not provide sufficient information to determine a match quickly and accurately. We outline how we instantiate the summary-article matching task that captures these details.

Task setup. We instantiate the general document matching tasks with a *summary-article matching task*. Here, the query and candidate documents are each sampled from human-written summaries and news articles in the CNN/DailyMail dataset [49, 107], a common NLP dataset used for summarization task. We select this dataset because the contents are accessible to a general audience, which enables us to evaluate a variety of assistive methods by employing crowdworkers as in Lai et al. [74], Wang and Yin [135]. So in our task, the participants are given a series of questions composed of a query summary with three candidate articles[2], and are asked to select an article from which the summary is generated under a time constraint (Figure 5.2).

As in the general document matching task, each candidate article is presented with an affinity score computed by a language model, which captures the similarity between the article and the summary. The affinity scores are computed by taking a cosine similarity between the final hidden representation of a

[2] While the decision makers in a general matching task may observe more than three candidate articles, we devise a simpler instantiation here to reduce the complexity of the task, which will be better suited for the crowdsourcing task.

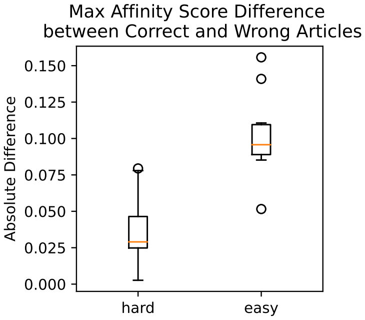


Figure 5.3: Distribution of affinity scores—computed by the matching model—for hard and easy questions. The box plot shows maximum absolute difference in affinity scores between the correct and wrong candidate articles for each of the hard and easy questions. The smaller the absolute difference is, the smaller the gap between the correct and the wrong article, making the scores less helpful in identifying the correct article (e.g., for the hard questions).

language model for the article and the summary [26, 139]. We use the representations from the DistilBART [112] model fine-tuned on the CNN/DailyMail dataset.

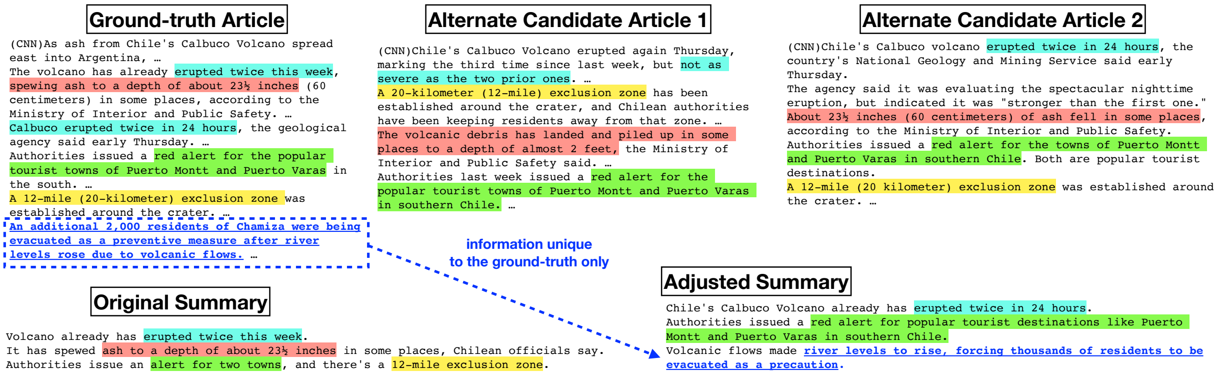
Question types. In practice, there are some questions where the correct (document) match is obvious, whereas other questions require a more thorough inspection of the specifics. For instance, in scientific peer review, a paper about a new optimization method in deep learning may be assigned to a broad range of candidate reviewers whose general research area is within deep learning. However, a reviewer who has worked both on optimization theory and deep learning may be a better fit compared to others who have primarily worked on large-scale deep-learning based vision models. Even among the reviewers in optimization theory, the reviewers who have worked on similar type of methods to the one proposed may be better suited for the match. Such subtleties require more careful examination by the meta-reviewers.

We capture such scenarios by creating a data pool composed of two types of questions via manual inspection: easy and hard. Easy questions have candidate answers (articles) from different topics or events that are easily discernible from one another, and therefore can be easily matched correctly. On the other hand, hard questions have candidate articles with a shared topic that only differ in small details, requiring a more careful inspection by the users.

On easy versus hard questions, the affinity scores naturally show distinctive behaviors. The gap of the scores between the correct and the wrong matches is smaller for the hard questions than for the easy ones (Figure 5.3). Because the scores for all candidate articles are similar to one another in the hard questions, the affinity scores are not as helpful in identifying the best match. Additionally, it is more likely that the candidate article with the highest affinity score is not the correct match in the hard questions. If a hypothetical user was to simply select a candidate article which has the highest affinity score by completely relying on the matching model's output, they would be accurate only for 33.3 percent of the time for the hard questions, compared to 100 percent of the time for the easy questions. We believe that providing users with assistive information might be critical for improving outcomes when making decisions on the hard questions, when the model is less accurate and the correct match is more difficult to find.

Defining ground-truth matches. A ground-truth match for a given summary and a set of candidate articles is necessary to measure participant performance. To construct pairs of summary and candidate articles, we first sample a summary-article pair from raw dataset and consider the article as the ground-truth for the given summary. We then select two other articles from the dataset which have the highest affinity scores with respect to the given summary as the incorrect candidate articles for the given summary.

There are several instances where the two alternate candidate articles, which should be incorrect



(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, …
The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. …
Calbuco erupted twice in 24 hours, the geological agency said early Thursday. …
Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. …
A 12-mile (20-kilometer) exclusion zone was established around the crater. …
An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. …

(CNN)Chile's Calbuco Volcano erupted again Thursday, marking the third time since last week, but not as severe as the two prior ones. …
A 20-kilometer (12-mile) exclusion zone has been established around the crater, and Chilean authorities have been keeping residents away from that zone. …
The volcanic debris has landed and piled up in some places to a depth of almost 2 feet, the Ministry of Interior and Public Safety said. …
Authorities last week issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in southern Chile. …

(CNN)Chile's Calbuco volcano erupted twice in 24 hours, the country's National Geology and Mining Service said early Thursday.
The agency said it was evaluating the spectacular nighttime eruption, but indicated it was "stronger than the first one."
About 23½ inches (60 centimeters) of ash fell in some places, according to the Ministry of Interior and Public Safety.
Authorities issued a red alert for the towns of Puerto Montt and Puerto Varas in southern Chile. Both are popular tourist destinations.
A 12-mile (20 kilometer) exclusion zone was established around the crater. …

Volcano already has erupted twice this week.
It has spewed ash to a depth of about 23½ inches in some places, Chilean officials say.
Authorities issue an alert for two towns, and there's a 12-mile exclusion zone.

Chile's Calbuco Volcano already has erupted twice in 24 hours.
Authorities issued a red alert for popular tourist destinations like Puerto Montt and Puerto Varas in southern Chile.
Volcanic flows made river levels to rise, forcing thousands of residents to be evacuated as a precaution.

Figure 5.4: Ensuring a single correct match for the questions. For an original summary sampled from the dataset, the three candidate articles appear to be all correct matches – the critical information highlighted with different colors in the original summary is contained in all three candidate articles (highlighted with the same colors). To resolve having multiple correct matches for the question, we manually extract information unique to the ground-truth article only (underlined text in dotted box) and add it to the adjusted summary to ensure that only the ground-truth article is the correct match for the summary.

choices, may arguably be a suitable choice for the given summary. This happens because the dataset contains multiple articles covering the same event. To resolve this issue of having multiple ground-truths, we manually modify the given summary so that it is consistent only with the ground-truth article. Specifically, we manually identify unique information in the ground-truth article that is not part of the alternate candidate articles and add that information to the summary (Figure 5.4).

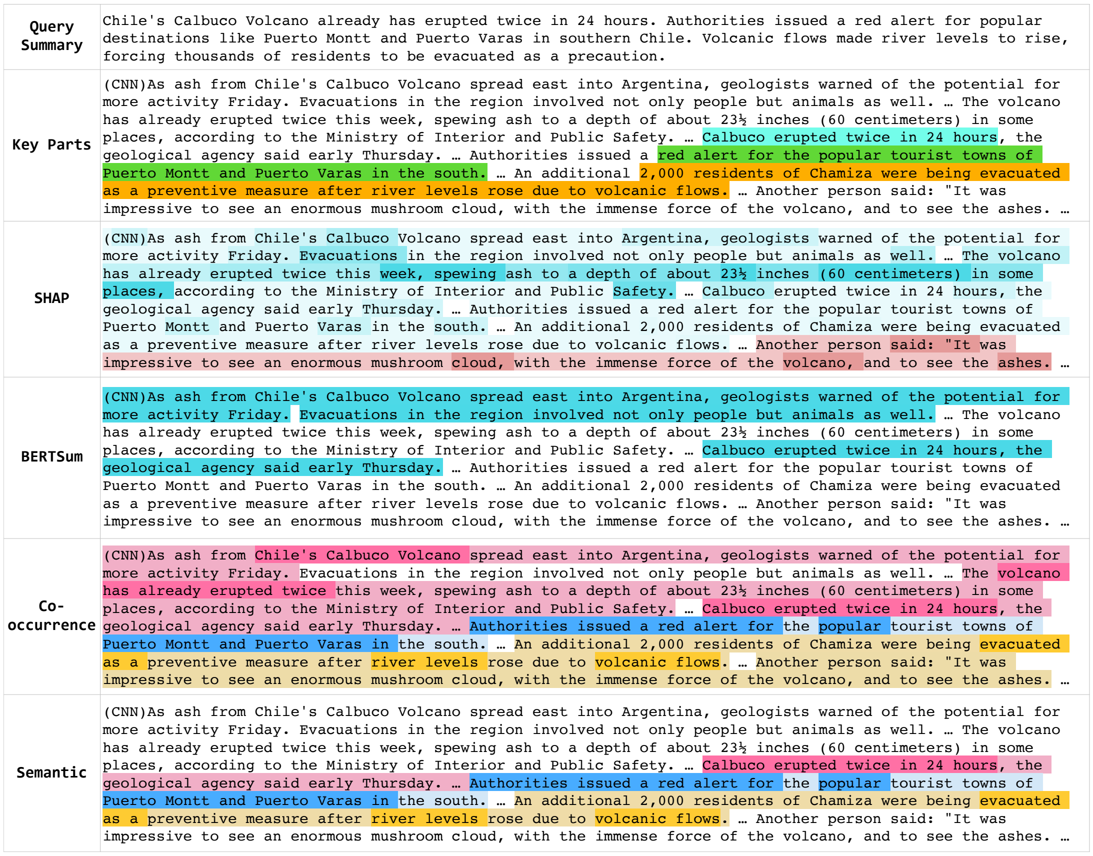
5.3.2 Tested Methods

In this section, we describe the methods used to highlight assistive information that we evaluate in our study and how they are presented to the users.

Black-box Model Explanation. Black-box model explanations include techniques that aim to highlight important input tokens for a model's prediction [83, 113, 117, 124]. While there are several candidates to consider, we use a widely-applied method called SHAP [83]. SHAP assigns attribution scores to each input token that indicate how much they contribute to the prediction output. We select SHAP from a pool of prominent explanation methods (which include Integrated Gradients [124] and Input x Gradients [113]) by examining how much the distribution of attribution scores deviate from random distribution of attribution scores (see Appendix D.1 for details).

We visualize SHAP (example shown in Figure 5.5, third row) by highlighting the input tokens according to their attribution scores. Tokens that contribute to increasing the affinity score (i.e., those with positive attribution scores) are highlighted in cyan, while those that decrease the score (i.e., those with negative attribution scores) are highlighted in pink. The color gradients of the highlights indicate the magnitude of the attribution scores: the darker the color, the bigger the magnitude.

Extractive Summarization. Summarization methods are trained to select key information within a large body of text. These methods can potentially help users process multiple lengthy articles in a shorter amount of time [76, 81, 150, 152]. Summaries generated by these methods are typically either generative (i.e., the summary is a newly-generated text that may not be part of the original text) or extractive (i.e., the summary is composed of text pieces extracted from the original text) [46]. Because generative summaries are more susceptible to hallucinating information not present in the original text [23, 54, 87], we focus on evaluating extractive summaries. In particular, we use BERTSum [81], which achieves state-of-the-



Query Summary	Chile's Calbuco Volcano already has erupted twice in 24 hours. Authorities issued a red alert for popular destinations like Puerto Montt and Puerto Varas in southern Chile. Volcanic flows made river levels to rise, forcing thousands of residents to be evacuated as a precaution.
Key Parts	(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, geologists warned of the potential for more activity Friday. Evacuations in the region involved not only people but animals as well. … The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. … Calbuco erupted twice in 24 hours, the geological agency said early Thursday. … Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. … An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. … Another person said: "It was impressive to see an enormous mushroom cloud, with the immense force of the volcano, and to see the ashes. …
SHAP	(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, geologists warned of the potential for more activity Friday. Evacuations in the region involved not only people but animals as well. … The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. … Calbuco erupted twice in 24 hours, the geological agency said early Thursday. … Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. … An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. … Another person said: "It was impressive to see an enormous mushroom cloud, with the immense force of the volcano, and to see the ashes. …
BERTSum	(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, geologists warned of the potential for more activity Friday. Evacuations in the region involved not only people but animals as well. … The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. … Calbuco erupted twice in 24 hours, the geological agency said early Thursday. … Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. … An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. … Another person said: "It was impressive to see an enormous mushroom cloud, with the immense force of the volcano, and to see the ashes. …
Co-occurrence	(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, geologists warned of the potential for more activity Friday. Evacuations in the region involved not only people but animals as well. … The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. … Calbuco erupted twice in 24 hours, the geological agency said early Thursday. … Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. … An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. … Another person said: "It was impressive to see an enormous mushroom cloud, with the immense force of the volcano, and to see the ashes. …
Semantic	(CNN)As ash from Chile's Calbuco Volcano spread east into Argentina, geologists warned of the potential for more activity Friday. Evacuations in the region involved not only people but animals as well. … The volcano has already erupted twice this week, spewing ash to a depth of about 23½ inches (60 centimeters) in some places, according to the Ministry of Interior and Public Safety. … Calbuco erupted twice in 24 hours, the geological agency said early Thursday. … Authorities issued a red alert for the popular tourist towns of Puerto Montt and Puerto Varas in the south. … An additional 2,000 residents of Chamiza were being evacuated as a preventive measure after river levels rose due to volcanic flows. … Another person said: "It was impressive to see an enormous mushroom cloud, with the immense force of the volcano, and to see the ashes. …

Figure 5.5: Highlighted information using different methods on the ground-truth article of the summary-article pair example in Figure 5.4. Highlights for "Key Parts" (second row) indicate information relevant to the query summary (first row), all of which ideally should be visibly highlighted by the methods that follow. SHAP (third row) and BERTSum (fourth row) fail to fully highlight all key parts. Critically, they fail to visibly highlight the key part about river levels rising (yellow highlights in Key Parts), the unique information that distinguishes the ground-truth from other candidate articles (as described in Figure 5.4), which can directly impact the participant's performance. On the other hand, our task-specific methods, both Co-occurrence (fifth row) and Semantic (sixth row) ones, are able to visibly highlight all key parts.

art performance on the CNN/DailyMail dataset [94], to extract three key sentences from the article. We visualize the extracted summary by highlighting the selected sentences from the original text with a single solid color (example shown in Figure 5.5, fourth row).

Task-specific Methods. The summary-article matching task requires users to accurately and quickly identify whether all details in the summary are correctly presented in each article. This is particularly challenging for hard questions, where the ground-truth can only be identified by looking at the right part of the articles due to their subtle differences.

Next we propose two *task-specific* methods that are more tailored to addressing this challenge. The methods operate at sentence and phrase-level information in the summary and candidate articles. Specifically, we select and show the top K sentences[3] from each article with the highest similarity measure to

[3] We pick $K = 3$, but this can be tuned for different levels of detail, depending on the length of the summary or the article.

each sentence in the summary[4]. To further provide more fine-grained detail on why that sentence could have been selected, we then show exactly-matching phrases within those selected sentences. Essentially, the methods are designed to guide the users to relevant parts in the article for each summary sentence by presenting the relevance hierarchically – by first showing the key sentences and then the key phrases within.

We consider two versions of the method which use different similarity measures to select the sentences:

- **Co-occurrence method** uses F1 score of ROUGE-L [78], a common performance metric used to capture the degree of n-gram co-occurrence between two texts.
- **Semantic method** uses the cosine similarity between the sentence representations from Sentence-BERT [100], a transformer model trained for sentence-level tasks. These scores are more sensitive to semantic similarities among texts like paraphrased components that may not be effectively captured by ROUGE-L.

Once we select K sentences based on the similarity measures, we visualize them using different colors to differentiate sentences in the article related to different sentence in the summary. Like before, we use color gradients to indicate the magnitude of the similarity score for each sentence (the higher the similarity, the darker the color). We then color the exactly-matching phrases using the darkest shade. For instance, in Figure 5.5 (fifth and sixth rows), the pink, blue and yellow highlights indicate relevant parts to first, second, and third sentence in the summary respectively. We include additional examples from each of the explored methods in Appendix D.2.

5.4 Experiments

We run a pre-registered[5] user study on the summary-article matching task introduced in Section 5.3.1 to evaluate the methods described in Section 5.3.2 as treatment conditions. In this section, we outline the details of the user study (Section 5.4.1), followed by our main hypotheses (Section 5.4.2) and results (Sections 5.4.3).

5.4.1 User Study Design

We present 16 questions to each participant. The 16 questions comprise 4 easy and 12 hard questions in random order. Participants complete all questions in one sitting. For each question, participants see a query summary followed by three longer candidate articles (see Figure 5.2 for an example). To incorporate the time constraints typical decision makers may face in practical settings, as similarly done in [95], we limit participants to spend 3 minutes to answer each question, after which they automatically see the next question. We offer bonus payments to encourage high-quality responses in terms of both accuracy and time (more details in Appendix D.3.4).

We recruit 275 participants from a balanced pool of adult males and females located in the U.S. with minimum approval ratings of 90% on Prolific (www.prolific.co), with diverse demographic

[4] Note that one could alternatively consider applying SHAP to the sentence-level information from the summary instead of the entire summary. While seemingly providing an apples-to-apples comparison to the sentence-level task-specific methods, a sentence-level application of SHAP would be unconventional and inconsistent with SHAP's intended use. Indeed, SHAP is designed to explain a model's prediction, and in our set up the model makes a prediction for the affinity score between the *entire summary* and the article. Hence, the most natural and consistent way of using SHAP is applying it to the entire input. For text summarization method, a similar argument holds: the method summarizes a longer text by extracting a subset of information and therefore is meant to take in the entire document content as the input.

[5] Pre-registration document is available at https://aspredicted.org/LMM_4K9

background (more details in Appendix D.3.2). The sample size is determined from Monte Carlo power analysis based on data collected from a separate pilot study, for a statistical power above 0.8 (more details in Appendix D.3.1). Each participant is then randomly assigned to one of five groups:

- *Control*: participants see the basic information (summary, articles, affinity scores)
- *SHAP*: participants see the basic information + highlights from SHAP
- *BERTSum*: participants see the basic information + highlights from BERTSum
- *Co-occurrence*: participants see the basic information + highlights from Co-occurrence method
- *Semantic*: participants see the basic information + highlights from Semantic method

We include two attention check questions in the study in addition to the 16 questions above. 271 out of 275 participants pass both attention-check questions, and we exclude responses from the 4 non-qualifying participants from our further analysis. We include mode details about the user study in Appendix D.3.

5.4.2 Main Hypotheses

We pose the following null hypotheses with two-sided alternatives about the mean accuracy of participants on the hard questions, using different kinds of assistive information:

(**H1**) The mean accuracy of participants using SHAP is not different from that of the control.

(**H2**) The mean accuracy of participants using BERTSum is not different from that of the control.

(**H3**) The mean accuracy of participants using Co-occurrence method is not different from the control.

(**H4**) The mean accuracy of participants using Semantic method is not different from the control.

To test each hypothesis, we compare the mean accuracy of the participants in different treatment settings against the control group with two-tailed permutation tests, where the test statistic is the difference in the mean accuracy. We account for multiple comparisons with Sidak correction [115] for the family-wise error rate of 0.05.

5.4.3 Results

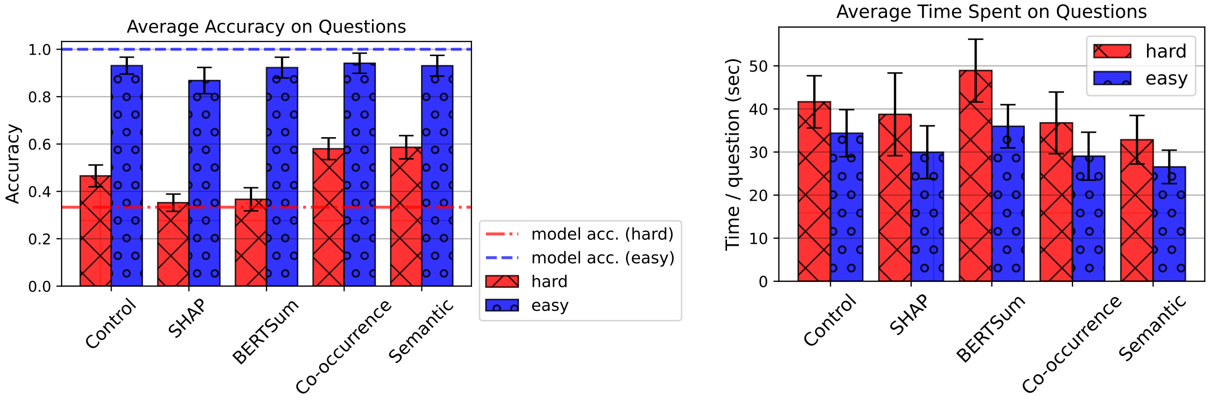
We now discuss the participants' task accuracy, completion time, and qualitative responses in different treatment groups.

Accuracy Difference

We find a statistically significant difference in the mean accuracy of all treatment groups compared to the control and reject the null hypotheses **H1** through **H4** in Section 5.4.2.

- Participants using SHAP perform significantly *worse* than the control ($p = 0.001599 < 0.05$).
- Participants using BERTSum perform significantly *worse* than the control ($p = 0.0056 < 0.05$).
- Participants using Co-occurrence method perform significantly *better* than the control ($p = 0.002997 < 0.05$).
- Participants using using the Semantic method perform significantly *better* than the control ($p = 0.002997 < 0.05$).

Comparing the participants' accuracy against the model accuracy on different question types, we verify that the assistive information is particularly helpful for the hard questions (Figure 5.6a). Note that the model is only accurate around 33.3% of the time in hard questions (red dotted line) while being



(a) Average accuracy with 95% confidence intervals.

(b) Average time with 95% confidence intervals.

Figure 5.6: (a) On hard questions, we observe significantly higher accuracy in groups presented with assistive information from Co-occurrence and Semantic methods compared to the control, and lower accuracy in groups presented with assistive information from SHAP and BERTSum. The dotted lines indicate the accuracy of the matching model, i.e., accuracy when selecting the article with the highest affinity score. For easy questions, it is more effective to simply follow the affinity scores without support from additional assistive information. For hard questions where the correct match is less obvious, using Co-occurrence or Semantic methods may be effective. (b) We observe a lower average time for groups given SHAP, Co-occurrence, and Semantic methods compared to control in both types of questions, but higher average time for BERTSum.

100% accurate on easy questions (blue dotted line). The information from Semantic method is the most effective for the hard questions with the highest average accuracy of 58.6%, which is a 26% increase in accuracy compared to the control (46.6%) and a 77% increase compared to the model accuracy (33.3%), while SHAP (35.2%) and BERTSum (36.7%) remain less effective (red checker-patterned bars). On the other hand, there appears no significant difference in accuracy among the methods for the easy questions (blue dotted bars), all of them slightly less accurate than the model accuracy. The results suggest that while it is more efficient to rely on the affinity scores for the easy questions, assistive information via Semantic methods can be particularly helpful for the hard questions, when the correct match is less obvious for both the models and humans. This further suggests that for the best results in practice, it may be useful to consider first identifying the difficulty of the question and then determine if additional information is necessary.

Time Difference

We record the average response time (in seconds/question) for participants in each treatment group. We observe that on average the participants using SHAP, Co-occurrence, and Semantic methods respond more quickly compared to the control group for both types of questions (Figure 5.6b). For both easy and hard questions, the participants using the Semantic method take the shortest average time (26.6 seconds for easy and 32.9 seconds for hard), which is approximately a 20% improvement over the control (34.4 seconds for easy and 41.7 seconds for hard). The participants using BERTSum take the longest (36 seconds for easy and 48.9 seconds for hard), where they experience a 17% increase in time for the hard questions. Given that the Semantic method is also able to significantly boost the task accuracy, it is the most effective method among the tested ones. On the other hand, as BERTSum simultaneously decreases the task accuracy and increases the completion time, it may be considered the least effective method.

Qualitative Responses

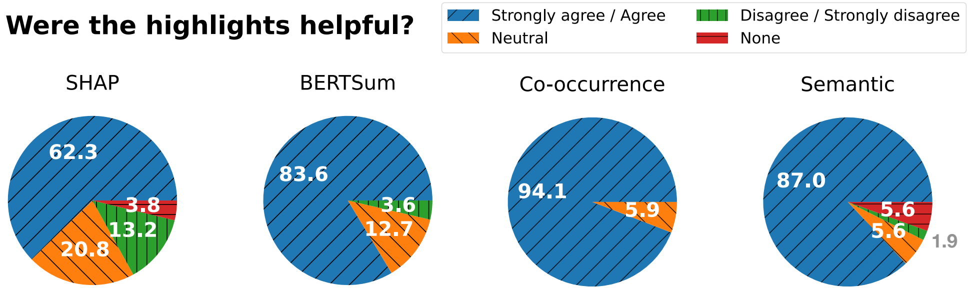


Figure 5.7: Participants' responses (in percentage) for "Were the highlights helpful?". For all the methods, the majority of the participants respond positively to the question regardless of their actual task accuracy.

At the end of the user study, the participants are asked several qualitative questions about the task.

"Were the highlights helpful?" Participants from all of the treatment groups generally respond positively to this question – Figure 5.7 shows the proportion of different responses from the participants in each group, and positive responses in blue form the majority in all groups. While the participants *believe* the highlights to be helpful, their task performance shows the contrary for participants using SHAP and BERTSum. Such discrepancy between the subjective perception of a tool's utility and the objective utility measured by task-grounded performance metrics corroborate similar previous observations on different assistive tools [12, 58].

"What information was the most helpful in answering the question?" While the majority of the participants using BERTSum, Co-occurrence, and Semantic methods respond that the highlights were the most helpful, the participants using SHAP have more diverse responses that showed no particular preference (Figure 5.8). It is interesting to note that the participants using either Co-occurrence or Semantic methods find the role of highlights to be significantly helpful when compared to other methods.

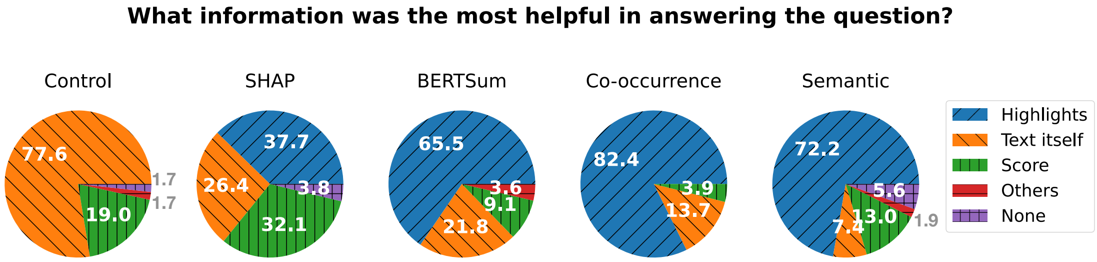


Figure 5.8: Participants' responses (in percentage) to a question "What information was the most helpful?"

"Were there too many highlights?" We find that the participants using SHAP most strongly agree to this sentiment (Figure 5.9). One factor that could have contributed to this is the default output values from SHAP used to generate the highlights, which are not post-processed for more succinct representation of information. Appropriate post-processing of the attribution scores may be necessary to better account for this—the impact of the amount of highlights on the task performance is an open research question that requires future work.

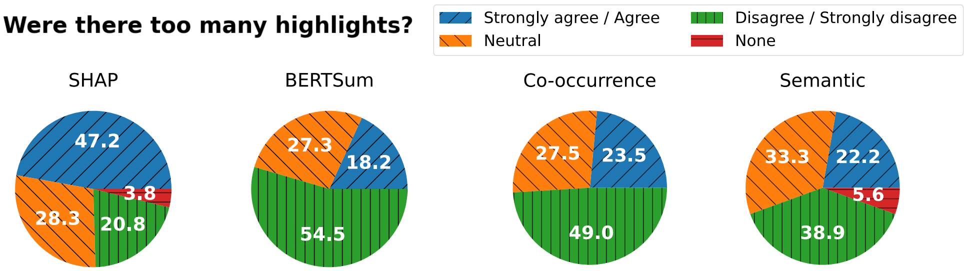


Figure 5.9: Participants' responses (in percentage) to a question "Were there too many highlights?"

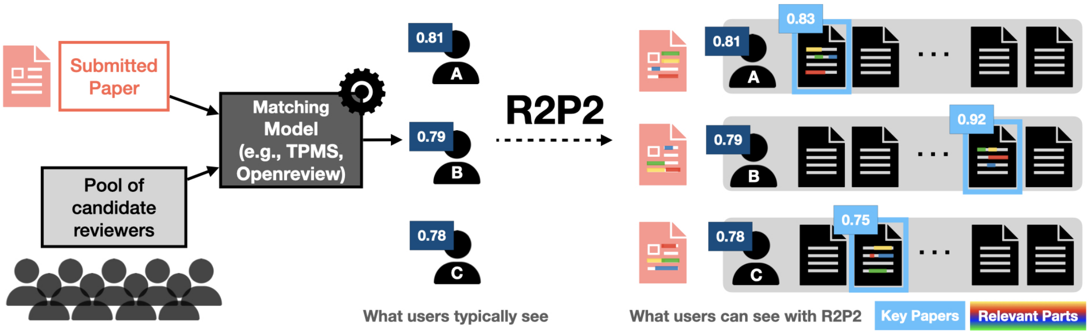

Submitted Paper
Pool of candidate reviewers
Matching Model (e.g., TPMS, Openreview)
0.81
A
0.79
B
0.78
C
R2P2
What users typically see
0.83
0.92
0.75
What users can see with R2P2
Key Papers
Relevant Parts

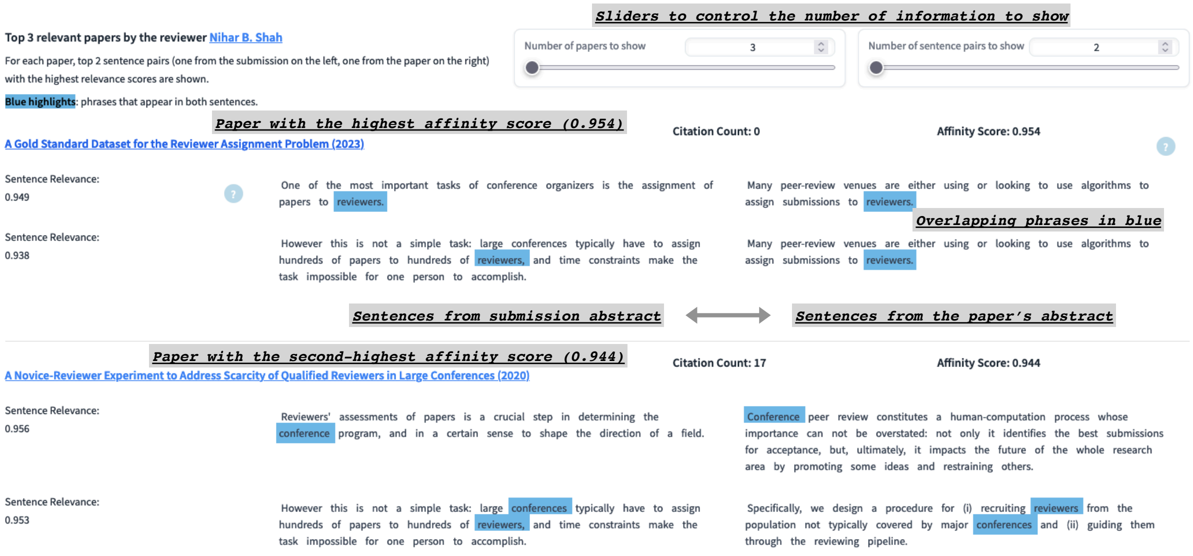
Figure 5.10: Overview of R2P2. While typical meta-reviewers only see the ranked list of potential reviewers and their respective affinity scores (left), R2P2 helps provide more context to the match, by searching for top relevant papers by the reviewer, as well as relevant pieces of information within those papers (right).

5.5 Towards Application to Peer Review

Based on the results obtained in our user study for summary-article matching, we apply the semantic method to a more specific set of documents for peer review: academic abstracts. Abstracts contain rich and compact information about what the paper is about and are therefore widely utilized in practice. For instance, large language models use the embeddings of the abstracts to subsequently assign affinity scores to each candidate reviewer [31]. In this section, we introduce a tool we developed that uses the academic abstracts and the semantic method to assist meta-reviewers check the quality of a potential match between a submitted paper and a reviewer.

Meta-reviewers generally "are presented with little structured information about the reviewers" [126]: what the meta-reviewers typically see is just a list of candidate reviewers, sorted according to their affinity scores (Figure 5.10, left). They can then click on each reviewer's profile to see more details about the reviewer (e.g., Google scholar page, personal websites, etc.). The tool, which we call R2P2 (**R**eviewer **TO** **P**aper for **P**eer Review), is designed to provide more structured information about each reviewers relevant to the submission. It takes in a title and an abstract of the submitted paper, and a potential reviewer's profile (Semantic Scholar profile link). It then searches for previous papers written by the reviewer and ranks each paper based on its abstract's similarity to the submitted abstract, computed with embeddings from a language model of choice[6]. Once the papers are ranked and top papers are retrieved, it further uses the language model to highlight relevant sentences within the abstracts of the retrieved papers

[6] In our current version, we use Specter2[31, 118].


Sliders to control the number of information to show
Paper with the highest affinity score (0.954)
Overlapping phrases in blue
Sentences from submission abstract
Sentences from the paper's abstract
Paper with the second-highest affinity score (0.944)

Top 3 relevant papers by the reviewer Nihar B. Shah

For each paper, top 2 sentence pairs (one from the submission on the left, one from the paper on the right) with the highest relevance scores are shown.

Blue highlights: phrases that appear in both sentences.

Number of papers to show: 3

Number of sentence pairs to show: 2

A Gold Standard Dataset for the Reviewer Assignment Problem (2023)

Citation Count: 0

Affinity Score: 0.954

Sentence Relevance: 0.949

One of the most important tasks of conference organizers is the assignment of papers to reviewers.

Many peer-review venues are either using or looking to use algorithms to assign submissions to reviewers.

Sentence Relevance: 0.938

However this is not a simple task: large conferences typically have to assign hundreds of papers to hundreds of reviewers, and time constraints make the task impossible for one person to accomplish.

Many peer-review venues are either using or looking to use algorithms to assign submissions to reviewers.

A Novice-Reviewer Experiment to Address Scarcity of Qualified Reviewers in Large Conferences (2020)

Citation Count: 17

Affinity Score: 0.944

Sentence Relevance: 0.956

Reviewers' assessments of papers is a crucial step in determining the conference program, and in a certain sense to shape the direction of a field.

Conference peer review constitutes a human-computation process whose importance can not be overstated: not only it identifies the best submissions for acceptance, but, ultimately, it impacts the future of the whole research area by promoting some ideas and restraining others.

Sentence Relevance: 0.953

However this is not a simple task: large conferences typically have to assign hundreds of papers to hundreds of reviewers, and time constraints make the task impossible for one person to accomplish.

Specifically, we design a procedure for (i) recruiting reviewers from the population not typically covered by major conferences and (ii) guiding them through the reviewing pipeline.

Figure 5.11: R2P2 first provides relevant sentence pairs from the submission abstract (left) and the top papers by the reviewer (right). Example here is for the TPMS paper [26] as the submission and Nihar Shah as the reviewer. From the presented information, it is easy to catch that both the reviewer and the submission have a common interest in peer review.

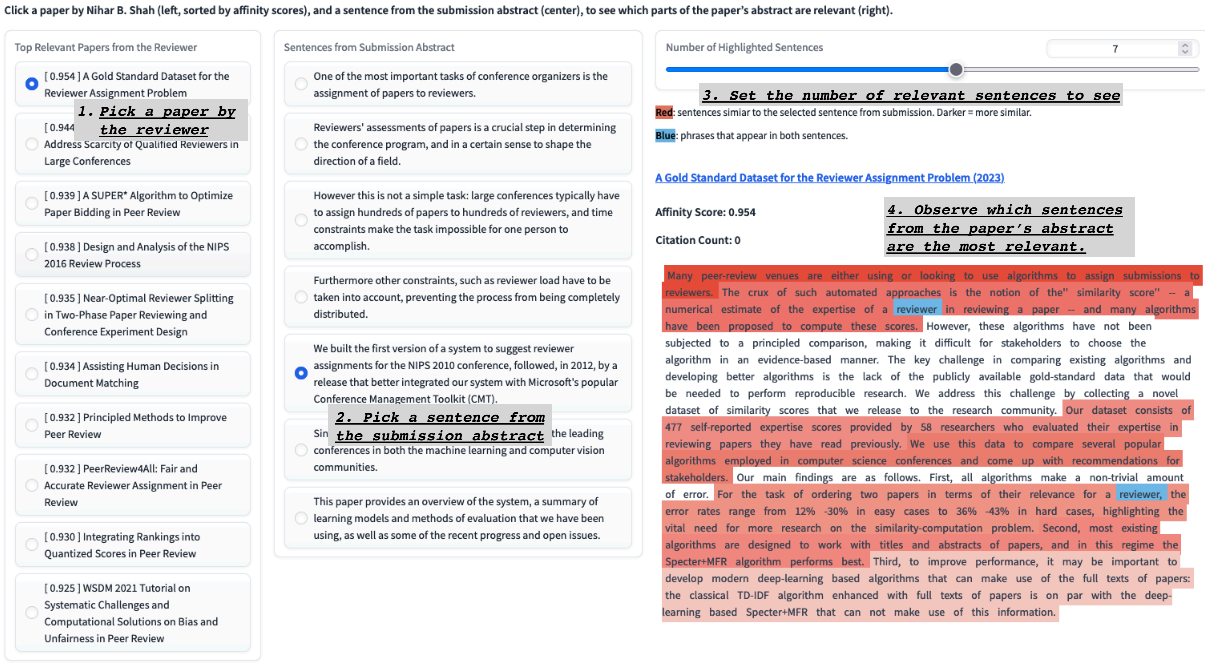
(Figure 5.10, right), just as we have done in the summary-article matching setup. The tool is available for trial here[7].

Unlike the summary-article matching setup where the users are simply asked to process three candidate texts and a short paragraph of summary text, this setup involves longer input text (e.g., submitted abstract) as well as multiple candidate texts (e.g., reviewer's previous papers) to process. This requires a change in how the highlights and relevant information are presented, as visualizing all highlights with different colors at once with all text (as we did in the summary-article matching task) is not practical. R2P2 displays a more stratified set of information as described below.

Initial Information. First, the users are shown succinct characterization of what parts from the abstracts (one from the submitted paper, one from the reviewer's previous paper) make them similar. In particular, we pick the top relevant sentence pairs (based on relevance scores computed by the language model) from the submitted abstract and the abstracts of the reviewer's top relevant papers and display them side-by-side (Figure 5.11). We further allow users to change the number of top papers by the reviewer and top sentence pairs to show for each paper. We highlight common phrases that are present in both sentences with blue. Noting the common phrases and the content of the top pairs, the similarity between the submission and the reviewer

Additional Information. If the provided information above is not sufficient, the users have a choice to explore each of the reviewer's previous papers in more detail. This part gives the users more degrees of freedom to select a specific reviewer's paper, a specific sentence from the submitted abstract, and the number of relevant sentences to show from the abstract of the selected reviewer's paper (Figure 5.12).

[7] https://huggingface.co/spaces/jskim/paper-matching



Click a paper by Nihar B. Shah (left, sorted by affinity scores), and a sentence from the submission abstract (center), to see which parts of the paper's abstract are relevant (right).

Top Relevant Papers from the Reviewer

- ● [0.954] A Gold Standard Dataset for the Reviewer Assignment Problem
- ○ [0.944 … Address Scarcity of Qualified Reviewers in Large Conferences
- ○ [0.939] A SUPER* Algorithm to Optimize Paper Bidding in Peer Review
- ○ [0.938] Design and Analysis of the NIPS 2016 Review Process
- ○ [0.935] Near-Optimal Reviewer Splitting in Two-Phase Paper Reviewing and Conference Experiment Design
- ○ [0.934] Assisting Human Decisions in Document Matching
- ○ [0.932] Principled Methods to Improve Peer Review
- ○ [0.932] PeerReview4All: Fair and Accurate Reviewer Assignment in Peer Review
- ○ [0.930] Integrating Rankings into Quantized Scores in Peer Review
- ○ [0.925] WSDM 2021 Tutorial on Systematic Challenges and Computational Solutions on Bias and Unfairness in Peer Review

1. Pick a paper by the reviewer

Sentences from Submission Abstract

- ○ One of the most important tasks of conference organizers is the assignment of papers to reviewers.
- ○ Reviewers' assessments of papers is a crucial step in determining the conference program, and in a certain sense to shape the direction of a field.
- ○ However this is not a simple task: large conferences typically have to assign hundreds of papers to hundreds of reviewers, and time constraints make the task impossible for one person to accomplish.
- ○ Furthermore other constraints, such as reviewer load have to be taken into account, preventing the process from being completely distributed.
- ● We built the first version of a system to suggest reviewer assignments for the NIPS 2010 conference, followed, in 2012, by a release that better integrated our system with Microsoft's popular Conference Management Toolkit (CMT).
- ○ Sin … the leading conferences in both the machine learning and computer vision communities.
- ○ This paper provides an overview of the system, a summary of learning models and methods of evaluation that we have been using, as well as some of the recent progress and open issues.

2. Pick a sentence from the submission abstract

Number of Highlighted Sentences 7

3. Set the number of relevant sentences to see

🟥 Red: sentences simiar to the selected sentence from submission. Darker = more similar.

🟦 Blue: phrases that appear in both sentences.

A Gold Standard Dataset for the Reviewer Assignment Problem (2023)

Affinity Score: 0.954

Citation Count: 0

4. Observe which sentences from the paper's abstract are the most relevant.

Many peer-review venues are either using or looking to use algorithms to assign submissions to reviewers. The crux of such automated approaches is the notion of the" similarity score" -- a numerical estimate of the expertise of a reviewer in reviewing a paper -- and many algorithms have been proposed to compute these scores. However, these algorithms have not been subjected to a principled comparison, making it difficult for stakeholders to choose the algorithm in an evidence-based manner. The key challenge in comparing existing algorithms and developing better algorithms is the lack of the publicly available gold-standard data that would be needed to perform reproducible research. We address this challenge by collecting a novel dataset of similarity scores that we release to the research community. Our dataset consists of 477 self-reported expertise scores provided by 58 researchers who evaluated their expertise in reviewing papers they have read previously. We use this data to compare several popular algorithms employed in computer science conferences and come up with recommendations for stakeholders. Our main findings are as follows. First, all algorithms make a non-trivial amount of error. For the task of ordering two papers in terms of their relevance for a reviewer, the error rates range from 12% -30% in easy cases to 36% -43% in hard cases, highlighting the vital need for more research on the similarity-computation problem. Second, most existing algorithms are designed to work with titles and abstracts of papers, and in this regime the Specter+MFR algorithm performs best. Third, to improve performance, it may be important to develop modern deep-learning based algorithms that can make use of the full texts of papers: the classical TD-IDF algorithm enhanced with full texts of papers is on par with the deep-learning based Specter+MFR that can not make use of this information.

Figure 5.12: R2P2 then allows users to explore more details for each of the reviewer's paper. One can select a particular paper, select a particular sentence from the submission abstract, to see which sentences of the paper's abstract are the most relevant to the selected sentence.

The red highlights indicate the relevant sentences from the selected paper's abstract, where the darkness of the color scales with the magnitude of the relevance score (the darker the higher). Again, we highlight common phrases in blue. The users can interactively click on different parts to better understand the similarities and differences between the reviewer and the submission. It is also useful to look at the list of titles on the left to quickly verify if the reviewer has frequently published on a similar topic to that of the submission. In Figure 5.12, we see that all papers have affinity scores above 0.92, which indicate a potential strong match.

We perform interviews with few colleagues to collect qualitative feedback about the tool. To make them interact with the tool in a more meaningful manner (and therefore to make them more likely to provide quality feedback), we devise a small task that emulates the meta-reviewer's job of evaluating the quality of the given paper-reviewer assignment. Specifically, they are asked to use the tool to rate the quality of 5 different paper-reviewer assignments, using a score from 1 - 5 (5 being the strongest match, 1 being the weakest). The paper-reviewer assignments are sampled from a gold-standard dataset, which consists of "self-reported expertise scores provided by researchers who evaluated their expertise in reviewing papers they have read previously" [123].

Generally, the feedback was positive with suggestions for improving the UI and the algorithm. One common feature that the participants found useful was the title of top relevant papers from the reviewer's profile. Part of the workflow was to go through the list of these titles to deduce how the reviewer may be relevant. One participant also pointed out that skimming through the reviewer's publication list manually without such information would make it easy to miss few papers that could be highly relevant.

Sentence-by-sentence comparison presented as the initial information was generally useful, but the participants suggested it could be improved in terms of selecting the sentences to display. There were cases where the selected sentence pairs were introductory sentences of respective abstracts which contain less specific information about the assignment. While they were useful in checking for the general alignment of expertise, more detail was needed to ensure the quality of the match (hence the participants needed to look at the additional information section to ascertain their impressions). One participant suggested a possibility of improving this by leveraging a structural nature of academics abstracts (e.g., first sentence of the abstract is almost always about the general field of study; last few sentences mostly discuss specific details that are unique to the paper) to diversify the way we pick information to present.

Next steps would include addressing these feedback and running a scaled-up controlled study based on the task described above with domain experts. The gold-standard dataset contains multiple paper-reviewer assignments that can be utilized to design such study, so it would be possible to objectively test the effectiveness of the tool by comparing the control and treatment groups' performance. However, one needs to modify the task format so that it accounts for calibration issues, as the scores given in the datasets will be different from scores given by the participants. Additionally, setting up the study would require more care due to the high cost of recruiting domain experts in scale. It would also be helpful to pitch the tool to few domain experts with meta-reviewer experience and ask for their direct feedback.

Chapter 6

Conclusion

6.1 Summary

Explainable machine learning is a broad research field actively developing, both in terms of methodology and application. In this thesis, we presented our contributions to both of these aspects.

In Chapter 2, we proposed a novel method of selecting representer points, the training examples that are influential to the model's prediction. To do so we introduced the modified representer theorem that could be generalized to most deep neural networks, which allows us to linearly decompose the prediction (activation) value into a sum of representer values. The optimization procedure for learning these representer values is tractable and efficient, especially when compared against the influence functions proposed in [68]. We have demonstrated our method's advantages and performances on several large-scale models and image datasets, along with some insights on how these values allow the users to understand the behaviors of the model. Source code is available here.[1]

In Chapter 3, we introduced the FACT diagnostic, which facilitates systematic reasoning about different kinds of trade-offs involving arbitrarily many notions of performance and group fairness notions, which all can be expressed as functions of the fairness–confusion tensor. In our formalism, the majority of group fairness definitions in the literature are in fact linear or quadratic thus are easy to be imposed as constraints to the PFOP. The FACT diagnostic further benefits from elementary linear algebra and convex optimization to provide a unified perspective of viewing fairness–fairness trade-offs and fairness–performance trade-offs. We have also empirically demonstrated the practical use of the FACT diagnostic in several scenarios. Source code is available here.[2]

In Chapter 4, using `SMERF`, we created seven stylized prediction tasks which revealed significant shortcomings of existing saliency methods, especially in their ability to recover complex model reasoning. We further corroborated the main results with additional results using natural image backgrounds, demonstrating similar qualitative trends but degraded quantitative performance. `SMERF` serves as a natural benchmark to evaluate saliency methods' correctness, and our results suggest that it can roughly provide an optimistic upper bound of a method's performance in more complicated real-world scenarios. Source code is available here.[3]

In Chapter 5, motivated by practical concerns in document matching tasks with human decision makers, we conducted a user study to investigate the utility of different kinds of assistive information for the summary-article matching task. We found that even well-established black-box model explanations can

[1] `https://github.com/chihkuanyeh/Representer_Point_Selection`

[2] `https://github.com/wnstlr/FACT`

[3] `https://github.com/wnstlr/SMERF`

potentially impair the users' decisions, while task-specific approaches can effectively assist them. Existing methods are typically not explicitly optimized for the task's objective: Model explanations are contingent on the matching model; it attempts to explain what the *model* considers important, not necessarily what *human* users find important to perform well in the task. General text summarization methods can be helpful for succinctly expressing a high-level topic of the article, but may lack the precision of picking the details directly related to the given summary. Furthermore, we observed that the users' subjective perception on the utility of (assistive) information was misaligned with their performance on the task. These results altogether emphasize that it is important for the developers of such assistive tools to articulate the specific use (and users) it serves, and rigorously evaluate their proposals. We also made a step forward wrapping the methods tested into a concrete tool that people can openly utilize for paper-reviewer matching process in peer review. Source code is available here.[4]

6.2 Future Work

There remains many interesting open research directions moving forward for each of the presented work. To truly see if RPS is applicable to different domains other than image datasets with different types of neural networks, applying them to some signature language model architectures like LSTM [50] or transformers [130] would be interesting. Preliminary experiment results on LSTM for sentiment classification task is presented in Appendix A.

In FACT diagnostics, many of the presented results require only linear fairness functions and accuracy, as in the LAFOP/MS-LAFOP setting. Nevertheless, it is easy to extend this to quadratic fairness functions with more varied performance metrics depending on different use cases. We believe there are more interesting theoretical results to explore that stem from the formulation. For instance, we briefly introduce a small theoretical result regarding fairness–accuracy trade-offs using the FACT diagnostic in Appendix B.6, which deserves further analysis.

We believe that SMERF can broadly play an important role in guiding future methodological advances to overcome the demonstrated shortcomings of current saliency methods by systematically and quantitatively defining what correct behavior looks like on various tasks. Moreover, SMERF can be extended over time as new methods are developed that perform consistently well on these stylized settings, by generating more sophisticated perception and reasoning tasks, e.g., by introducing semi-synthetic objects and/or encoding more complex reasoning through which the objects impact predictions. An interested reader can take the source code to (i) run the entire pipeline from generating datasets to computing results; and (ii) evaluate new tasks by encoding new model reasoning and new methods. Finally, generalizing the main ideas behind SMERF may also be useful in settings where saliency methods are inherently not appropriate, e.g. problems that involve counterfactual model explanations [133].

The summary-article matching task developed as part of our work on document matching can be used as a first-pass test to validate promising methods (and promote development of new approaches). Relaxing some assumptions in our setup can provide further insights on strengths and weaknesses of individual methods in more complex scenarios (e.g., allowing multiple or no ground-truths in questions). Also, recent developments in generative large language models [19, 20, 130] provide broader family of models that can be explored as the core basis for picking the relevant information from text and explaining them. We are looking forward to continue testing the developed tool and iterate with meta-reviewers so that it could be potentially integrated with the peer review system in practice.

[4] https://github.com/wnstlr/document-matching

Chapter 7

Bibliography

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31:9505–9515, 2018. 4.2

[2] Julius Adebayo, Michael Muelly, Ilaria Liccardi, and Been Kim. Debugging tests for model explanations. In *Advances in Neural Information Processing Systems*, volume 33, pages 700–712, 2020. 4.2, 4.3.1, 4.4.1, C.2.4

[3] Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T. Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. innvestigate neural networks! *Journal of Machine Learning Research*, 20(93):1–8, 2019. 4.4, C.1.3

[4] David Alvarez Melis and Tommi Jaakkola. Towards robust interpretability with self-explaining neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 4.2

[5] Salha M. Alzahrani, Naomie Salim, and Ajith Abraham. Understanding plagiarism linguistic patterns, textual features, and detection methods. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(2):133–149, 2012. 5.1

[6] Kasun Amarasinghe, Kit T Rodolfa, Sérgio Jesus, Valerie Chen, Vladimir Balayan, Pedro Saleiro, Pedro Bizarro, Ameet Talwalkar, and Rayid Ghani. On the importance of application-grounded experimental design for evaluating explainable ml methods. *arXiv preprint arXiv:2206.13503*, 2022. 1, 5.1, 5.2

[7] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *International Conference on Learning Representations*, 2018. 1, 4.2

[8] Rushil Anirudh, Jayaraman J Thiagarajan, Rahul Sridhar, and Timo Bremer. Influential sample selection: A graph signal processing approach. *arXiv preprint arXiv:1711.05407*, 2017. 2.2

[9] Siddhant Arora, Danish Pruthi, Norman Sadeh, William W. Cohen, Zachary C. Lipton, and Graham Neubig. Explain, edit, and understand: Rethinking user study design for evaluating model explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5277–5285, Jun. 2022. 1, 5.2

[10] Nishanth Arun, Nathan Gaw, Praveer Singh, Ken Chang, Mehak Aggarwal, Bryan Chen, Katharina

Hoebel, Sharut Gupta, Jay Patel, Mishka Gidwani, et al. Assessing the trustworthiness of saliency maps for localizing abnormalities in medical imaging. *Radiology: Artificial Intelligence*, 3(6), 2021. 4.1, 4.2

[11] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015. 1, 2.2, 4.2, 4.4

[12] Gagan Bansal, Tongshuang Wu, Joyce Zhou, Raymond Fok, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, and Daniel Weld. Does the whole exceed its parts? the effect of ai explanations on complementary team performance. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–16, 2021. 1, 5.2, 5.4.3

[13] Neha Bansal, Arun Sharma, and RK Singh. A review on the application of deep learning in legal domain. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 374–381. Springer, 2019. 5.2

[14] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, et al. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018. 3.2

[15] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 2018. doi: 10.1177/0049124118782533. 1, 3.2, **??**, **??**

[16] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, pages 2403–2424, 2011. 2.2

[17] J Stewart Black and Patrick van Esch. Ai-enabled recruiting: What is it and how should a manager use it? *Business Horizons*, 63(2):215–226, 2020. 5.2

[18] Bastian Bohn, Michael Griebel, and Christian Rieger. A representer theorem for deep kernel learning. *arXiv preprint arXiv:1709.10441*, 2017. 2.2

[19] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 6.2

[20] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 6.2

[21] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2), September 2010. doi: 10.1007/s10618-010-0190-x. 1, 3.2

[22] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops*, 2009. 3.1, 3.2, 3.6.1

[23] Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. Faithful to the original: Fact aware neural abstractive summarization. In *thirty-second AAAI conference on artificial intelligence*, 2018. 5.3.2

[24] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on*

Fairness, Accountability, and Transparency, 2019. 3.2

[25] Navoneel Chakrabarty and Sanket Biswas. A statistical approach to adult census income level prediction. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*. IEEE, 2018. 3.6.2

[26] Laurent Charlin and Richard Zemel. The toronto paper matching system: an automated paper-reviewer assignment system. 2013. (document), 5.1, 5.2, 5.3.1, 5.11

[27] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018. 4.1, 4.2

[28] Valerie Chen, Jeffrey Li, Joon Sik Kim, Gregory Plumb, and Ameet Talwalkar. Interpretable machine learning: Moving from mythos to diagnostics. *Commun. ACM*, 65(8):43–50, July 2022. ISSN 0001-0782. 1, 5.1

[29] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2), 2017. 1, 3.1, 3.2, 3.3, 3.3, **??**, **??**, **??**, 3.5.2, 3.5.2, 3.5.2

[30] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.207. URL `https://aclanthology.org/2020.acl-main.207`. 5.1, 5.2

[31] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. Specter: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, 2020. 5.5, 6

[32] Kate Crawford, Roel Dobbe, Theodora Dryer, Genevieve Fried, Ben Green, Elizabeth Kaziunas, Amba Kak, Varoon Mathur, Erin McElroy, Andrea Nill Sánchez, Deborah Raji, Joy Lisi Rankin, Rashida Richardson, Jason Schultz, Sarah Myers West, and Meredith Whittaker. Ai now 2019 report, 2019. 3.1

[33] George B. Dantzig. Linear programming and extensions. Technical Report R-366-PR, RAND Corporation, Santa Monica, California, 1963. URL `https://www.rand.org/content/dam/rand/pubs/reports/2007/R366part1.pdf`. B.7.1

[34] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017. 1, 5.1

[35] Dheeru Dua and Casey Graff. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences, 2017. URL `http://archive.ics.uci.edu/ml`. 3.6.1

[36] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 2012. 1, 3.1, 3.2

[37] Ahmed Elnaggar, Christoph Gebendorfer, Ingo Glaser, and Florian Matthes. Multi-task deep learning for legal document translation, summarization and multi-label classification. In *Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference*, pages 9–15, 2018. 5.2

[38] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and An-

drew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 1, 4.4, C.2.1

[39] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, 2015. 3.1, 3.2

[40] Carmen Fernández and Alberto Fernández. Ethical and legal implications of ai recruiting software. *Ercim News*, 116:22–23, 2019. 5.2

[41] T Fiez, N Shah, and L Ratliff. A SUPER* algorithm to optimize paper bidding in peer review. In *Conference on Uncertainty in Artificial Intelligence*, 2020. 5.2

[42] Tomáš Foltỳnek, Norman Meuschke, and Bela Gipp. Academic plagiarism detection: a systematic literature review. *ACM Computing Surveys (CSUR)*, 52(6):1–42, 2019. 5.1, 5.2

[43] Tomáš Foltỳnek, Dita Dlabolová, Alla Anohina-Naumeca, Salim Razı, Július Kravjar, Laima Kamzola, Jean Guerrero-Dib, Özgür Çelik, and Debora Weber-Wulff. Testing of support tools for plagiarism detection. *International Journal of Educational Technology in Higher Education*, 17(1): 1–31, 2020. 5.2

[44] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 4.1, 4.2

[45] Han Guo, Nazneen Rajani, Peter Hase, Mohit Bansal, and Caiming Xiong. Fastif: Scalable influence functions for efficient model interpretation and debugging. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10333–10350, 2021. 1

[46] U. Hahn and I. Mani. The challenges of automatic summarization. *Computer*, 33(11):29–36, 2000. doi: 10.1109/2.881692. 5.3.2

[47] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, 2016. (document), 1, 3.1, 3.2, 3.3, 3.3, **??**, **??**, 3.4.1, 3.6.2, 3.6.4, 3.3, 3.6.4, B.7.3

[48] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2.4.2

[49] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28, 2015. 5.3.1

[50] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997. 6.2

[51] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 9737–9748, 2019. 4.2

[52] Yuxin Huang, Zhengtao Yu, Junjun Guo, Zhiqiang Yu, and Yantuan Xian. Legal public opinion news abstractive summarization by incorporating topic information. *International Journal of Machine Learning and Cybernetics*, 11(9):2039–2050, 2020. 5.2

[53] Sérgio Jesus, Catarina Belém, Vladimir Balayan, João Bento, Pedro Saleiro, Pedro Bizarro, and João Gama. How can i choose an explainer? an application-grounded evaluation of post-hoc

explanations. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 805–815, 2021. 1, 5.1, 5.2, D.1

[54] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 2022. 5.3.2

[55] M Jiffriya, MA Jahan, and R Ragel. Plagiarism detection tools and techniques: A comprehensive survey. *Journal of Science-FAS-SEUSL*, 2(02):47–64, 2021. 5.2

[56] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL `http://www.scipy.org/`. B.7.1

[57] Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining*, 2010. 3.1, 3.2

[58] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–14, 2020. 5.1, 5.4.3

[59] Wolfgang E Kerzendorf, Ferdinando Patat, Dominic Bordelon, Glenn van de Ven, and Tyler A Pritchard. Distributed peer review enhanced with natural language processing and machine learning. *Nature Astronomy*, 4(7):711–717, 2020. 5.1

[60] Rajiv Khanna, Been Kim, Joydeep Ghosh, and Sanmi Koyejo. Interpreting black box predictions using fisher kernels. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3382–3390. PMLR, 2019. 1

[61] Been Kim, Cynthia Rudin, and Julie A Shah. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in Neural Information Processing Systems*, pages 1952–1960, 2014. 2.2

[62] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, pages 2280–2288, 2016. 2.2

[63] Joon Sik Kim, Jiahao Chen, and Ameet Talwalkar. FACT: A diagnostic for group fairness tradeoffs. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5264–5274. PMLR, 13–18 Jul 2020. **??**, 1

[64] Joon Sik Kim, Gregory Plumb, and Ameet Talwalkar. Sanity simulations for saliency methods. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 11173–11200. PMLR, 17–23 Jul 2022. **??**, 1

[65] Joon Sik Kim, Valerie Chen, Danish Pruthi, Nihar B Shah, and Ameet Talwalkar. Assisting human decisions in document matching. *arXiv preprint arXiv:2302.08450*, 2023. **??**, 1

[66] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 267–280. Springer, 2019. 4.2

[67] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, 2017. 1, 3.1, 3.2, 3.3, **??**, **??**, **??**, 3.5.1, 3.5.1, 3.5.1, 3.5.1.1, **??**

[68] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In

International Conference on Machine Learning, pages 1885–1894, 2017. 1, 2.2, 2.4.2, 2.4.5, 6.1, A.2

[69] Dieter Kraft. A software package for sequential quadratic programming. Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt (DFVLR), 1988. B.7.1

[70] Dieter Kraft. Algorithm 733: TOMP–Fortran modules for optimal control calculations. *ACM Transactions on Mathematical Software*, 20(3), 1994. doi: 10.1145/192115.192124. B.7.1

[71] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009. 2.3.2

[72] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 3, 4.4.3, C.1.2, C.2.6

[73] Vivian Lai and Chenhao Tan. On human predictions with explanations and predictions of machine learning models: A case study on deception detection. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 29–38, 2019. 5.2

[74] Vivian Lai, Han Liu, and Chenhao Tan. "why is 'chicago' deceptive?" towards building model-driven tutorials for humans. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2020. 5.2, 5.3.1

[75] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2.4.1

[76] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL `https://aclanthology.org/2020.acl-main.703`. 5.2, 5.3.2

[77] Lan Li, Tina Lassiter, Joohee Oh, and Min Kyung Lee. Algorithmic hiring in practice: Recruiter and hr professional's perspectives on ai use in hiring. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '21, page 166–176, New York, NY, USA, 2021. Association for Computing Machinery. 5.1, 5.2

[78] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-1013`. 5.3.2

[79] Zachary C Lipton. The mythos of model interpretability. *Queue*, 16(3):31–57, 2018. 1

[80] Lydia T. Liu, Max Simchowitz, and Moritz Hardt. The implicit fairness criterion of unconstrained learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. 3.1, 3.2, B.6

[81] Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1387. URL `https://aclanthology.org/D19-1387`. 5.1, 5.2, 5.3.2

[82] Yang Liu, Sujay Khandagale, Colin White, and Willie Neiswanger. Synthetic benchmarks for scientific research in explainable machine learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 4.2

[83] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30:4765–4774, 2017. 1, 4.1, 4.4, 5.1, 5.3.2, D.1

[84] Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pages 142–150. Association for Computational Linguistics, 2011. A.4

[85] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *Proceedings of the 35th International Conference on Machine Learning*, 2018. 3.1

[86] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 4.4.3, C.2.7

[87] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.173. URL `https://aclanthology.org/2020.acl-main.173`. 5.3.2

[88] Reshef Meir, Jérôme Lang, Julien Lesca, Natan Kaminsky, and Nicholas Mattei. A market-inspired bidding scheme for peer review paper assignment. In *Games, Agents, and Incentives Workshop at AAMAS*, 2020. 5.2

[89] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, 2018. 3.2

[90] David Mimno and Andrew McCallum. Expertise modeling for matching papers with reviewers. In *KDD*, 2007. 5.2

[91] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017. 4.4

[92] Arvind Narayanan. Translation tutorial: 21 fairness definitions and their politics. In *Proceedings of the Conference on Fairness, Accountability and Transparency, New York, USA*, 2018. 1, 3.1, 3.2

[93] Gherman Novakovsky, Nick Dexter, Maxwell W Libbrecht, Wyeth W Wasserman, and Sara Mostafavi. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nature Reviews Genetics*, pages 1–13, 2022. D.1

[94] Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, 2021. 5.3.2

[95] Elizabeth Pier, Joshua Raclaw, Anna Kaatz, Markus Brauer, Molly Carnes, Mitchell Nathan, and Cecilia Ford. Your comments are meaner than your score: score calibration talk influences intra- and inter-panel variability during scientific grant peer review. *Research Evaluation*, 26(1):1–14, 2017. 5.4.1

[96] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, 2017. 3.2, **??**, **??**

[97] P. Poovizhi, K. Ezhilarasi, G. Gayathri, R. Megala, and D. Anisha. Automatic scraping of employment record using machine learning—an assistance for the recruiter. In *Smart Data Intelligence*, pages 561–577. Springer Nature Singapore, 2022. 5.1, 5.2

[98] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33: 19920–19930, 2020. 1

[99] Fabrizio Pucci, Martin Schwersensky, and Marianne Rooman. Artificial intelligence challenges for predicting the impact of mutations on protein stability. *Current opinion in structural biology*, 72: 161–168, 2022. D.1

[100] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL `http://arxiv.org/abs/1908.10084`. 5.3.2

[101] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016. 1, 2.2

[102] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. 2018. 2.2

[103] Marko A. Rodriguez and Johan Bollen. An algorithm to determine peer-reviewers. In *ACM Conference on Information and Knowledge Management*, 2008. 5.2

[104] Samira Samadi, Uthaipon Tantipongpipat, Jamie H Morgenstern, Mohit Singh, and Santosh Vempala. The price of fair PCA: One extra dimension. In *Advances in Neural Information Processing Systems*, 2018. 3.1

[105] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. A generalized representer theorem. In *International conference on computational learning theory*, pages 416–426. Springer, 2001. 2.2

[106] Candice Schumann, Jeffrey Foster, Nicholas Mattei, and John Dickerson. We need fairness and explainability in algorithmic hiring. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2020. 5.1

[107] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1099. URL `https://aclanthology.org/P17-1099`. 5.3.1

[108] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 1, 4.1, 4.2, 4.3.1, 4.4, 4.4.1

[109] Harshay Shah, Prateek Jain, and Praneeth Netrapalli. Do input gradients highlight discriminative features? *Advances in Neural Information Processing Systems*, 34, 2021. (document), 4.2, 4.4.3, 4.11, C.2.7, C.16, C.2.7

[110] Nihar Shah, Behzad Tabibian, Krikamol Muandet, Isabelle Guyon, and Ulrike Von Luxburg. Design and analysis of the NIPS 2016 review process. *JMLR*, 19(1):1913–1946, 2018. 5.2

[111] Nihar B Shah. Challenges, experiments, and computational solutions in peer review. *Communications of the ACM*, 65(6):76–87, 2022. 5.1

[112] Sam Shleifer and Alexander M Rush. Pre-trained summarization distillation. *arXiv preprint arXiv:2010.13002*, 2020. 5.2, 5.3.1

[113] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016. 1, 5.3.2, D.1

[114] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153. PMLR, 06–11 Aug 2017. 1, 2.2, 4.4

[115] Zbynek Sidak. Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62(318):626–633, 1967. ISSN 01621459. URL `http://www.jstor.org/stable/2283989`. 5.4.2

[116] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2.3.2, 3, 4.4.3, C.1.2, C.2.6

[117] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 1, 2.2, 4.1, 4.2, 4.4, 5.3.2

[118] Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. Scirepeval: A multi-format benchmark for scientific document representations. *ArXiv*, abs/2211.13308, 2022. 6

[119] Leon Sixt, Martin Schuessler, Oana-Iuliana Popescu, Philipp Weiß, and Tim Landgraf. Do users benefit from interpretable vision? a user study, baseline, and dataset. In *International Conference on Learning Representations*, 2021. 4.2

[120] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 2.4.4, 4.4

[121] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. Learning controllable fair representations. In *Proceedings of Machine Learning Research*, 2019. 3.1

[122] J Springenberg, Alexey Dosovitskiy, Thomas Brox, and M Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015. 4.4

[123] Ivan Stelmakh, John Wieting, Graham Neubig, and Nihar B Shah. A gold standard dataset for the reviewer assignment problem. *arXiv preprint arXiv:2303.16750*, 2023. 5.5

[124] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR, 06–11 Aug 2017. 1, 2.2, 4.1, 4.4, 5.3.2, D.1

[125] Zilong Tan, Samuel Yeom, Matt Fredrikson, and Ameet Talwalkar. Learning fair representations for kernel models. In *Proceedings of The 23rd International Conference on Artificial Intelligence and Statistics*, 2020. 1, 3.1, 3.6.2

[126] Terne Thorn Jakobsen and Anna Rogers. What factors should paper-reviewer assignments rely on? community perspectives on issues and ideals in conference peer-review. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4810–4823, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.354. URL `https://aclanthology.org/2022.naacl-main.354`. 5.1, 5.2, 5.5

[127] Richard Tomsett, Dan Harborne, Supriyo Chakraborty, Prudhvi Gurram, and Alun Preece. Sanity checks for saliency metrics. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04): 6021–6029, Apr 2020. 4.2

[128] H. D. Tran, G. Cabanac, and G. Hubert. Expert suggestion for conference program committees. In *2017 11th International Conference on Research Challenges in Information Science (RCIS)*, pages 221–232, May 2017. doi: 10.1109/RCIS.2017.7956540. 5.2

[129] Michael Unser. A representer theorem for deep neural networks. *arXiv preprint arXiv:1802.09210*, 2018. 2.2

[130] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 6.2

[131] Sahil Verma and Julia Rubin. Fairness definitions explained. In *Proceedings of the International Workshop on Software Fairness*, 2018. doi: 10.1145/3194770.3194776. 3.2

[132] Indrė Žliobaitė. On the relation between accuracy and fairness in binary classification. In *Proceedings of the Second Workshop on Fairness, Accountability, and Transparency in Machine Learning*, 2015. 3.1, 3.2

[133] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017. 6.2

[134] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 24–25, 2020. 4.4

[135] Xinru Wang and Ming Yin. Are explanations helpful? a comparative study of the effects of explanations in ai-assisted decision-making. In *26th International Conference on Intelligent User Interfaces*, pages 318–328, 2021. 5.3.1

[136] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019. 1, C.2.1

[137] John Wieting, Kevin Gimpel, Graham Neubig, and Taylor Berg-Kirkpatrick. Simple and effective paraphrastic similarity from parallel translations. In *ACL*, pages 4602–4608, Florence, Italy, July 2019. 5.2

[138] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4.1, 4.2

[139] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, and Ming Zhou. MIND: A large-scale dataset for news

recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3597–3606, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.331. URL `https://aclanthology.org/2020.acl-main.331`. 5.3.1

[140] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *arXiv preprint arXiv:1707.00600*, 2017. 2.4.2

[141] Guang Yang, Qinghao Ye, and Jun Xia. Unbox the black-box for the medical explainable AI via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. *Information Fusion*, 77:29–52, 2022. D.1

[142] Mengjiao Yang and Been Kim. Benchmarking attribution methods with relative feature importance, 2019. 4.2, 4.3.1, C.2.4

[143] Chih-Kuan Yeh, Joon Kim, Ian En-Hsu Yen, and Pradeep K Ravikumar. Representer point selection for explaining deep neural networks. *Advances in neural information processing systems*, 31, 2018. **??**, 1

[144] Éloi Zablocki, Hédi Ben-Younes, Patrick Pérez, and Matthieu Cord. Explainability of deep vision-based autonomous driving systems: Review and challenges. *International Journal of Computer Vision*, pages 1–28, 2022. D.1

[145] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the Second Workshop on Fairness, Accountability, and Transparency in Machine Learning*, 2015. 1, 3.1, 3.2, 3.6.1, 3.6.2

[146] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, 2017. 1, 3.1, 3.2

[147] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014. 1, 4.4

[148] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning*, 2013. 1, 3.1

[149] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126 (10):1084–1102, 2018. 4.1

[150] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020. 5.2, 5.3.2

[151] Han Zhao and Geoffrey J Gordon. Inherent tradeoffs in learning fair representations. In *Advances in Neural Information Processing Systems*, 2019. 3.2

[152] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.552. URL `https://aclanthology.org/2020.acl-main.552`. 5.3.2

[153] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4.1, 4.2, 4.3.1, 4.4

[154] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 4.4.4, C.2.9

[155] Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*. AAAI, Feb 2022. 4.2

Appendices

Appendix A

Supplementary for Chapter 2

A.1 Proof of Proposition 2.3.1

Proof. The convexity can be checked easily. If $\mathbf{\Theta}^* \in \arg\min_{\mathbf{\Theta}} L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$, by the first order condition we have

$$\frac{\partial L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}^*))}{\partial \mathbf{\Theta}_1^*} = \mathbf{0} \tag{A.1}$$

By chain rule we obtain

$$\frac{\partial L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}^*))}{\partial \Phi(\mathbf{x}_i, \mathbf{\Theta}^*)} \frac{\partial \Phi(\mathbf{x}_i, \mathbf{\Theta}^*)}{\partial \mathbf{\Theta}_1^*} = \mathbf{0}, \tag{A.2}$$

and thus we have

$$\left(\frac{\partial L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}^*))}{\partial \Phi(\mathbf{x}_i, \mathbf{\Theta}^*)}\right) \mathbf{f}_i^T = \mathbf{0}, \tag{A.3}$$

When $\mathbf{f}_i$ is not a zero vector, this reduces to

$$\frac{\partial L_{\text{softmax}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}^*))}{\partial \Phi(\mathbf{x}_i, \mathbf{\Theta}^*)} = \mathbf{0}, \tag{A.4}$$

and we show that $\sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta})) - \sigma(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given})) = \mathbf{0}$. As a result, L_{softmax} is "suitable to" the softmax activation.

If $\mathbf{\Theta}^* \in \arg\min_{\mathbf{\Theta}} L_{\text{ReLU}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta}))$, we can rewrite L_{ReLU} as

$$\begin{aligned}
&L_{\text{ReLU}}(\Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi(\mathbf{x}_i, \mathbf{\Theta})) \\
&= \frac{1}{2}\max(\Phi(\mathbf{x}_i, \mathbf{\Theta}), 0) \odot \Phi(\mathbf{x}_i, \mathbf{\Theta}) - \Phi(\mathbf{x}_i, \mathbf{\Theta}_{given}) \odot \Phi(\mathbf{x}_i, \mathbf{\Theta}) \\
&= \sum_{j=0}^{c} \frac{1}{2}\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}), 0)\Phi_j(\mathbf{x}_i, \mathbf{\Theta}) - \Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given})\Phi_j(\mathbf{x}_i, \mathbf{\Theta}),
\end{aligned} \tag{A.5}$$

and we note that $\mathbf{\Theta}_{1j}$, the j-th row for $\mathbf{\Theta}_1$, is only related to $\Phi_j(\mathbf{x}_i, \mathbf{\Theta})$. Therefore, we have $\mathbf{\Theta}_{1j}^* \in \arg\min_{\mathbf{\Theta}_{1j}} L_{\text{ReLU}}(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi_j(\mathbf{x}_i, \mathbf{\Theta}))$. We now consider the cases where $\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), 0) = 0$ and $\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), 0) > 0$.

When $\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), 0) = 0$,

$$L_{\text{ReLU}}(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi_j(\mathbf{x}_i, \mathbf{\Theta})) = \frac{1}{2}\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}), 0) \cdot \Phi_j(\mathbf{x}_i, \mathbf{\Theta}). \tag{A.6}$$

$\mathbf{\Theta}_{1j}$ obtains the minimum when $\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}), 0) = 0$, therefore $\sigma(\Phi_j(\mathbf{x}_i, \mathbf{\Theta})) = \sigma(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}))$.

When $\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), 0) > 0$,

$$L_{\text{ReLU}}(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}), \Phi_j(\mathbf{x}_i, \mathbf{\Theta})) = \frac{1}{2}\max(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}), 0) \cdot \Phi_j(\mathbf{x}_i, \mathbf{\Theta}) - \Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}) \cdot \Phi_j(\mathbf{x}_i, \mathbf{\Theta}).$$

For $\Phi_j(\mathbf{x}_i, \mathbf{\Theta}) \leq 0$, the minimum for L_{ReLU} is 0. For $\Phi_j(\mathbf{x}_i, \mathbf{\Theta}) > 0$, the minimum for L_{ReLU} is $-\frac{1}{2}\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given})^2$ only if $\Phi_j(\mathbf{x}_i, \mathbf{\Theta}) = \Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given})$. Therefore, the minimum is reached again when $\sigma(\Phi_j(\mathbf{x}_i, \mathbf{\Theta})) = \sigma(\Phi_j(\mathbf{x}_i, \mathbf{\Theta}_{given}))$. As a result, L_{ReLU} is "suitable to" the ReLU activation. $\square$

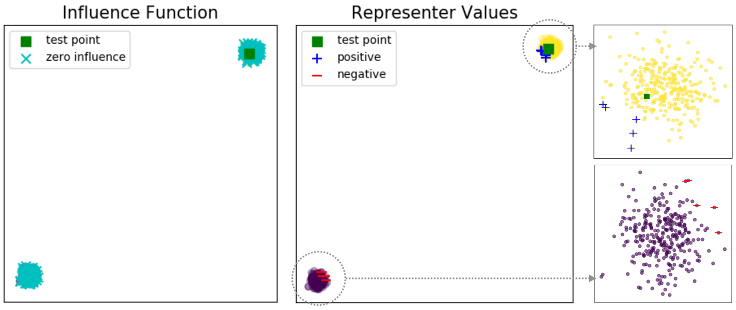


Figure A.1: Demonstration of numerical stability on a 2-D toy data. Even with a big margin, our values faithfully provide positive examples from the same class and negative examples from the different class near the decision boundary. Influence functions return zeros for all training points, thus is not able to provide influential points.

A.2 Relationship with the Influence Function

In this section, we compare the behaviors of our method and the influence functions [68]. Recall that for a training point $\mathbf{x}_i$ and a test point $\mathbf{x}_t$, the influence function value is computed in terms of the gradients/hessians of the loss, and it reflects how the loss at a test point $\mathbf{x}_t$ will change when the training point $\mathbf{x}_i$ is perturbed (weighted more/less). Because the influence function is defined in terms of the loss, its value can easily become arbitrarily close to zero if the the loss is flat in some region. On the other hand, our representer values are computed using the neurons' activation values, which may result in comparatively larger values in general. We verify this in a toy dataset in 2-D with a large margin shown in Figure A.1.

We train a multi-layer perceptron with ReLU activations as a binary classifier. The influential points, we would expect, are the points that are closer to the decision boundary. As shown on the left of Figure A.1, the influence function does not provide positive or negative examples from each class for the given test point in green square because all training points have exactly zero influence function values, marked with cyan crosses. However, our method provides correct positive and negative points near the decision boundary as we can see from the rightmost panel of Figure A.1.

In Figure A.2, we compare the behaviors of several metrics (Euclidean distance, influence function, representer value) for selecting training points that are most similar to a test point from CIFAR-10 dataset. We use a pre-trained VGG-16. As we can observe from the first two plots, the Euclidean distance does not reflect the class each training point is in – even if the training point is far away from the test point it may still be a similar image in the same class. On the other hand, representer and influence function values tell us about which class each training point is in. From the third plot, we observe that influence function and representer values agree on selecting images of horses as harmful and dogs as helpful.

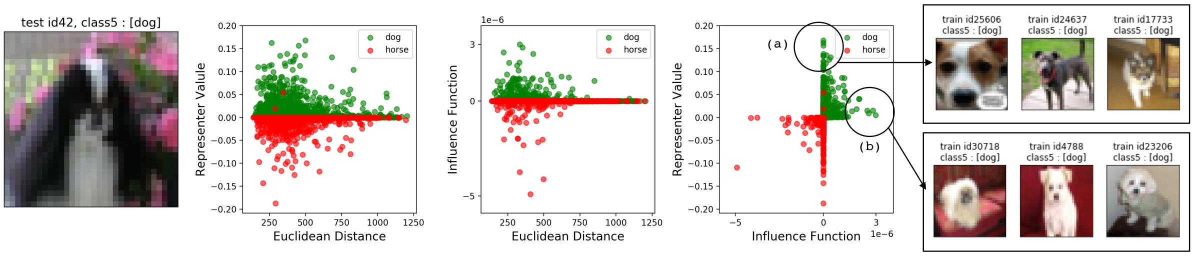


Figure A.2: Euclidean vs Influence Function vs Representer Value (ours). Representer values have bigger scale in general. Some examples from regions (a) where the influence function value is zero while our representer value is big and (b) where our representer value is small while the influence function value is big are shown.

A.3 More Examples of Positive/Negative Reperesenter Points

More examples of positive and negative representer points are shown in Figure A.3. Observe that the positive points all have the same class label as the test image with high resemblance, while the negative points, despite their similarity to the test image, have different classes.

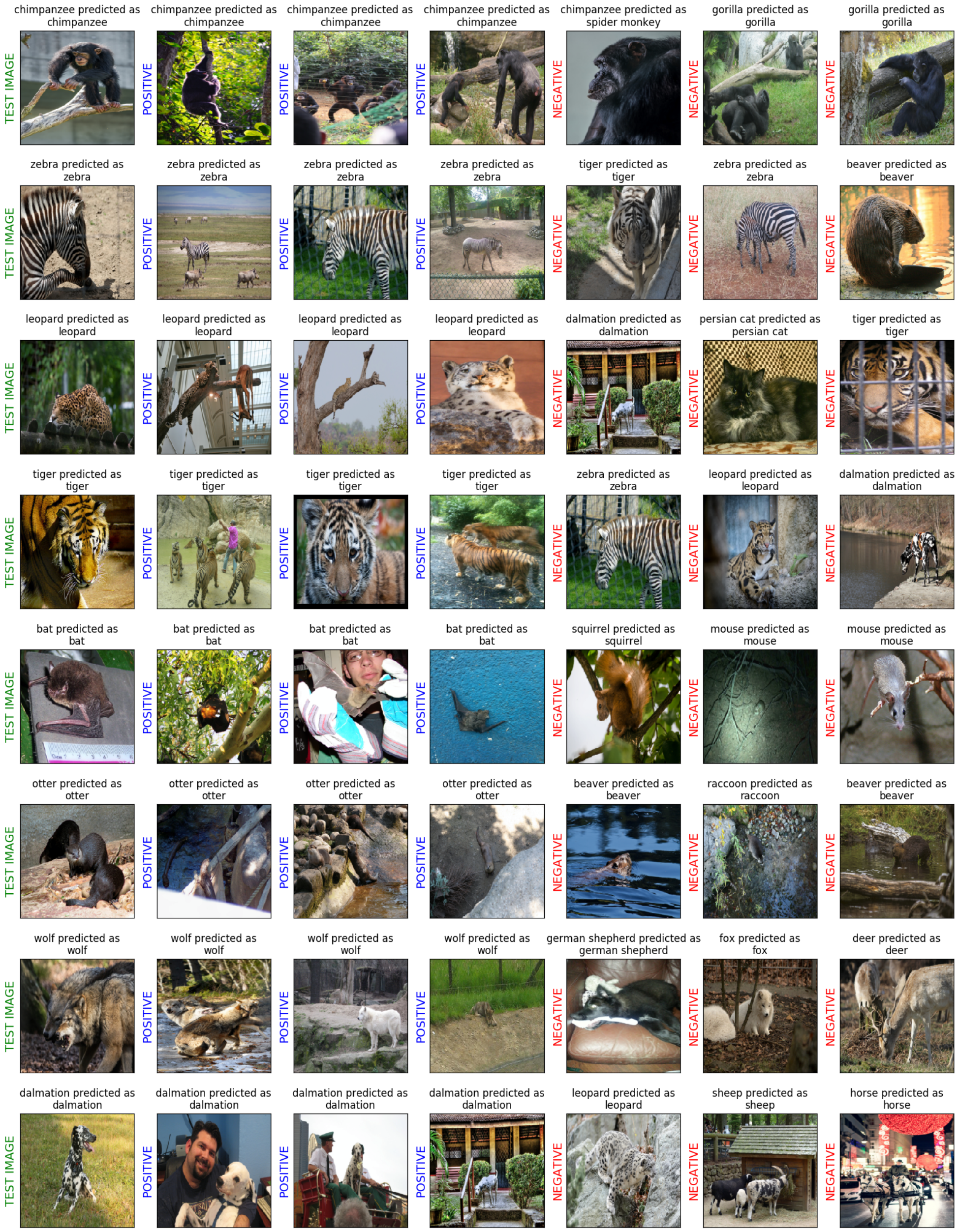


Figure A.3: More examples of positive and negative representer points. The first column is composed of test images from different classes from AwA dataset; the next three images are the positive representer points; the next three are negative representer points for each test image.

A.4 Representer Points of LSTM on NLP Data

We perform a preliminary experiment on an LSTM network trained on a IMDB movie review dataset [84]. Each data point is a review about a movie, and the task is to identify whether the review has a positive or negative sentiment. The pretrain model achieves 87.5% accuracy. We obtain the positive and negative repesenter points with methods described in section 2.3.2. In Table A.1, we show the test review which is predicted as a negative review by the LSTM network. We observe that both the top-1 positive and negative representer contain negative connotations just like the test review, but in the negative representer (which is a positive review about the movie) the negative connotation comes from a character in the movie rather than from a reviewer.

Test Review	¡START¿ when i first saw this movie in the theater i was so angry it completely blew in my opinion i didn't see it for a decade then decided what the hell let's see i'm watching all ¡¿ movies now to see where it went wrong my guess is it was with sequel 5 that was the first to ¡¿ the whole i am in a dream ¡¿ i see weird stuff oh ¡¿ it's not a dream oh wait i see something spooky oh never mind ¡¿ storyline those which made it so scary in the first place nothing fantasy nothing weird the box got opened boom they came was the only one that could bargain her way out of it first
Positive Representer	¡START¿ no not the ¡¿ of ¡¿ the ¡¿ ¡¿ ¡¿ but the mini series ¡¿ ¡¿ lifetime must have realized what a dog this was because the series was burned off two episodes at a time most of them broadcast between 11 p m friday nights and 1 a m saturday ¡¿ as to why i watched the whole thing i can only ¡¿ to ¡¿ sudden ¡¿ attacks of ¡¿ br br most of the cast are ¡¿ who are likely to remain unknown the only two ¡¿ names are shirley jones and rachel ward who turn in the only decent performances jones doesn't make it through the entire series lucky woman ward by the way is aging quite well since her
Negative Representer	¡START¿ this time around ¡¿ is no longer royal or even particularly close to being any such thing instead rather a butler to the prince ¡¿ portrayed by hugh ¡¿ who ¡¿ tim ¡¿ who presence is ¡¿ missed and that hole is never filled his character had an innocent charm was a bumbling and complete ¡¿ we can't help but care for him which isn't at all true of his ¡¿ as being ¡¿ which he apparently was according to the ¡¿ page not to mention loud ¡¿ and utterly non threatening ¡¿ can now do just about what he ¡¿ and does so why is he so frustrated and angry honestly it gets depressing at times yes his master is a ¡¿ they

Table A.1: An example of top-1 positive and negative representer points for IMDB dataset. ¡¿ stands for unknown words since only top 5000 vocabularies are used.

Appendix B

Supplementary for Chapter 3

B.1 Proof of Theorem 3.5.1

A useful strategy is to solve (3.10) for a set of solutions, then ask if any of these solutions satisfies an additional fairness constraint $\phi^{(K)}(\mathbf{z}) = 0$. This proof, as well as many of the ones below, illustrate this strategy in practice.

Proof. First, set $K = 1$ and $\mathbf{A}^{(0)} = \mathbf{A}_{\text{CG}}$ in (3.10). Since $v_0 \neq v_1$, the matrix $\mathbf{A}$ is full rank and therefore admits the solution (3.11). Considering $\mathbf{z}_0 \geq 0$ yields immediately the condition (3.12).

Next, set $K > 1$. Then either $\mathbf{z}_0$ is a solution (which is the case when all other fairness notions are linear and linearly dependent on $\begin{pmatrix} \mathbf{A}_{\text{CG}} \\ \mathbf{A}_{\textsf{const}} \end{pmatrix}$), or otherwise no solution exists to both (3.10) and $\phi^{(1)}(\mathbf{z}) = \cdots = \phi^{(K-1)}(\mathbf{z}) = 0$ simultaneously. □

This theorem states that $\Phi = \{\text{CG}\}$ is incompatible when $v_0 \neq v_1$, since it is a singleton set of incompatible fairness.

The condition $v_0 \neq v_1$ is necessary in Theorem 3.5.1, which is reasonable to assume as we would expect the positive class to have a higher score than the negative class in the definition of CG. We can prove the necessity of this condition by contradiction. In the degenerate case $v_0 = v_1 = v$, $\Phi = \{\text{CG}\}$ is a set of compatible fairness notions. It turns out that (3.10) with $K = 1$ is only on rank 6. Denoting $\textcircled{i}$ as the ith row of the matrix, we have two linear dependencies, $\textcircled{5} + \textcircled{6} + v\textcircled{1} = \textcircled{2}$ and $\textcircled{7} + \textcircled{8} + v\textcircled{3} = \textcircled{4}$. There is no longer a unique solution to the (3.10); instead, we have a two-parameter family of solutions,

$$\mathbf{z}(\alpha, \beta) = \frac{1}{N(1-v)} \begin{pmatrix} v(N_1(1-v) - \alpha) \\ v\alpha \\ (1-v)(N_1(1-v) - \alpha) \\ (1-v)\alpha \\ v(N_0(1-v) - \beta) \\ v\beta \\ (1-v)(N_0(1-v) - \beta) \\ (1-v)\beta \end{pmatrix}, \tag{B.1}$$
$$0 \leq \alpha \leq (1-v)N_1, \quad 0 \leq \beta \leq (1-v)N_0.$$

Furthermore, this family of solutions satisfies $\mathbf{A}_{\textsf{const}}\mathbf{z}_0 = \mathbf{b}_{\textsf{const}}$ if and only if $v = M_0/N_0 = M_1/N_1$, i.e. the base rates are equal and furthermore the score for both bins is equal to the base rate.

B.2 Proof of Corollary 3.5.1.1

Proof. Consider the product

$$\begin{pmatrix} \mathbf{A}_{\text{PCB}} \\ \mathbf{A}_{\text{NCB}} \end{pmatrix} z_0 = \frac{M_1 N_0 - M_0 N_1}{N} \begin{pmatrix} \frac{v_0 v_1}{M_0 M_1} \\ \frac{(1-v_0)(1-v_1)}{(M_0-N_0)(M_1-N_1)} \end{pmatrix}. \tag{B.2}$$

This product equals the zero vector (and hence satisfies both PCB and NCB) if and only if either of the conditions of the Corollary hold. (The last solution, $v_0 = 1$ and $v_1 = 0$, is inadmissible since $v_0 < v_1$ by assumption.) □

B.3 Proof of Corollary 3.5.1.2

Proof. The result follows from solving

$$\mathbf{A}_{\text{DP}} \mathbf{z}_0 = \frac{M_1 N_0 - M_0 N_1}{N^2 (v_1 - v_0)} = 0. \tag{B.3}$$

□

B.4 Proof of Corollary 3.5.1.3

Proof. The result follows from solving

$$\phi_{\text{PP}}(\mathbf{z}_0) = v_1(1 - v_1) \left((M_1 - N_1 v_0)^2 - (M_0 - N_0 v_0)^2 \right) = 0 \tag{B.4}$$

which is true if and only if either condition in the Corollary is true. (The last case, $v_1 = 0$, is inadmissible by assumption.) □

In addition, here is a situation of fairness "for free", in the sense that one notion of fairness automatically implies another.

Corollary B.4.0.1. *Consider a classifier that satisfies CG fairness. Then, the classifier also satisfies EFOR fairness. In other words, {CG, EFOR} is incompatible.*

Proof. $\phi_{\text{EFOR}}(\mathbf{z}_0) = 0$ vanishes identically. □

B.5 Proof of Theorem 3.5.2

Proof. Finding the solution to $\phi_{\text{PP}}(\mathbf{z}) = \phi_{\text{EFPR}}(\mathbf{z}) = \phi_{\text{EFNR}}(\mathbf{z}) = 0$ and also the linear system $\mathbf{A}_{\textsf{const}} \mathbf{z} = \mathbf{b}_{\textsf{const}}$ yields the three conditions of the Theorem. □

B.6 CG–accuracy trade-offs

In the paper, we have only considered the case when $\lambda = \infty$ in the LAFOP: we only consider when the fairness criteria are satisfied exactly yielding several fairness–fairness trade-off results without heed to the accuracy of the classifiers. Nonetheless, recall that LAFOP allows us to express both fairness–accuracy and fairness–fairness trade-offs by introducing an accuracy objective along with a fairness regularizer. In this section, we show how the LAFOP can be used to theoretically analyze a simple fairness–accuracy trade-off. We present a small result that is relevant to the CG–accuracy trade-off considered in [80].

Theorem B.6.1. *Let $\alpha = (M_0 + M_1)/N$ be the base rate. Consider a classifier that satisfies CG with $0 \leq v_0 < v_1 \leq 1$. Then, perfect accuracy is attained if and only if*

$$\frac{v_0(1-2v_1)}{1-v_1+v_0} = \alpha \leq \frac{1}{8}, \quad \left|v_0 - \frac{1}{4}\right| \leq \frac{\sqrt{1-8\alpha}}{4}. \tag{B.5}$$

Proof. The case of necessity ($\Rightarrow$) follows immediately from solving $\mathbf{c} \cdot \mathbf{z}_0 = 0$, where $\mathbf{z}_0$ is defined in Theorem 3.5.1. The inequality conditions follow immediately from the constraint $0 \leq v_0 < v_1 \leq 1$. The case of sufficiency ($\Leftarrow$) follows immediately from Theorem 3.5.1 and substituting the equality condition. □

The condition of this theorem relates the scores v_0 and v_1 to the base rate of the data, thus providing simple, explicit data dependencies that are necessary and sufficient.

B.7 Experiment Details

B.7.1 Optimization

For solving the optimization problems, we used solvers in the `scipy` package for Python [56]. For linear fairness constraints, we used the simplex algorithm [33], and for other constrained optimization forms, we used sequential least-squares programming (SLSQP) solver [69, 70].

B.7.2 Model-agnostic multi-way fairness–accuracy trade-offs

We have only considered situations where zero or one parameter is sufficient to simultaneously specify the fairness strength for every fairness function, i.e. $\lambda = \lambda_0 = \cdots = \lambda_{K-1}$. In this section, we generalize this and allow each regularization parameter to vary freely. It is then natural to consider the multilinear least-squares accuracy–fairness optimality problem (MLAFOP): $\operatorname{argmin}_{\mathbf{z} \in \mathcal{K}} (\mathbf{c} \cdot \mathbf{z})^2 + \sum_{i=0}^{K-1} \lambda_i \|\mathbf{A}^{(i)}\mathbf{z}\|_2^2$, where the regularization parameters λ_i now take different values across each of the K fairness constraints. This allows for a general inspection of the individual effect of fairness constraints in a group.

For instance, a three-way trade-off among EOd, DP, and accuracy can be visualized as a contour plot, similar to the one shown in Figure B.1. And for general $(K+1)$-way trade-offs involving K fairness constraints and accuracy, we visualize two-dimensional slices along the $K+1$-dimensional surface. For example, consider a four-way trade-off between a group of three fairness definitions (DP, EOd, PCB) and accuracy. Figure 3.2 already showed that imposing PCB given (DP, EOd) does not affect δ, which implies that PCB is the weakest in terms of its influence on δ. To get more information, for the S(B) dataset, we show in Figure B.2 three cases of varying one λ for one fairness constraint while keeping the other λ values fixed in MLAFOP. Sweeping through PCB condition (left) does not affect $1 - \delta$ at fixed EOd and DP levels, confirming the observation from Figure 3.2. Sweeping through DP conditions while keeping PCB and EOd strengths fixed (middle) results in a slight drop, but not big enough to make all levels to converge to values reported in Figure 3.2 (0.392). Sweeping through EOd while keeping PCB and DP strengths fixed (right) on the other hand results in significant changes for all levels and convergence to the value 0.392, suggesting EOd is stronger than DP in terms of its influence on changing δ. This notion of relative influence of fairness deserves further investigation, to see if these preliminary results are robust across other slices and datasets. Nonetheless, such analysis demonstrates a clear picture of how different notions of fairness interact with one another when they are to be imposed together.

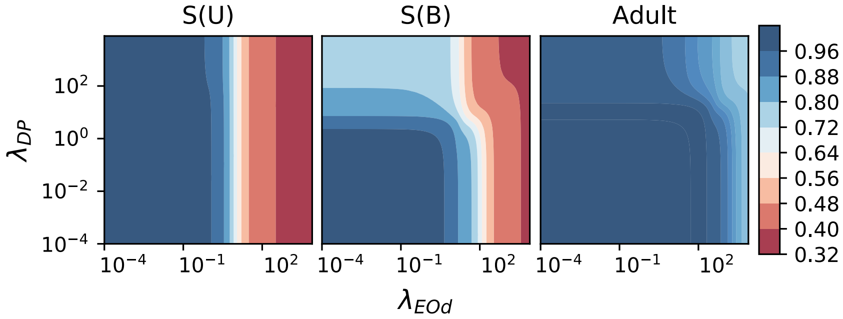


Figure B.1: Fairness–fairness–accuracy trade-off analysis using contour plot of accuracy with varying regularization strengths of Demographic Parity (DP) and Equalized Odds (EOd) for the unbiased synthetic dataset (left), biased synthetic dataset (middle), and Adult dataset (right). The contours show how the regularization strength of each fairness individually influence the accuracy $(1-\delta)$ given the other (accuracy of 1.0 being the accuracy of the Bayes classifier). For the unbiased synthetic data, the accuracy change along the vertical axis (DP) is practically nonexistent given EOd, while along the horizontal axis (EOd) the change is drastic. Other datasets demonstrate more complex relationships.

B.7.3 Connection to the post-processing methods for fair classification

We can explicitly rewrite the constraints in (3.8) using $\hat{z}$ and $\tilde{z}$, which respectively correspond to the fairness–confusion tensor of the given pre-trained classifier $\hat{Y}$ and the derived fair classifier $\tilde{Y}$:

$$\gamma_0(\tilde{Y}) = \gamma_1(\tilde{Y}) \iff \mathbf{A}_{\mathrm{EOD}}\tilde{\mathbf{z}} = 0$$

$$\gamma_0(\tilde{Y}) \in P_0(\hat{Y}) \iff \left(\frac{\tilde{\mathbf{z}}_7}{\tilde{\mathbf{z}}_7+\tilde{\mathbf{z}}_8}, \frac{\tilde{\mathbf{z}}_5}{\tilde{\mathbf{z}}_5+\tilde{\mathbf{z}}_6}\right) \in \text{convhull}\left\{(0,0), \left(\frac{\hat{\mathbf{z}}_7}{\hat{\mathbf{z}}_7+\hat{\mathbf{z}}_8}, \frac{\hat{\mathbf{z}}_5}{\hat{\mathbf{z}}_5+\hat{\mathbf{z}}_6}\right), \left(\frac{\hat{\mathbf{z}}_8}{\hat{\mathbf{z}}_7+\hat{\mathbf{z}}_8}, \frac{\hat{\mathbf{z}}_6}{\hat{\mathbf{z}}_5+\hat{\mathbf{z}}_6}\right), (1,1)\right\} \tag{B.6}$$

$$\gamma_1(\tilde{Y}) \in P_1(\hat{Y}) \iff \left(\frac{\tilde{\mathbf{z}}_3}{\tilde{\mathbf{z}}_3+\tilde{\mathbf{z}}_4}, \frac{\tilde{\mathbf{z}}_1}{\tilde{\mathbf{z}}_1+\tilde{\mathbf{z}}_2}\right) \in \text{convhull}\left\{(0,0), \left(\frac{\hat{\mathbf{z}}_3}{\hat{\mathbf{z}}_3+\hat{\mathbf{z}}_4}, \frac{\hat{\mathbf{z}}_1}{\hat{\mathbf{z}}_1+\hat{\mathbf{z}}_2}\right), \left(\frac{\hat{\mathbf{z}}_4}{\hat{\mathbf{z}}_3+\hat{\mathbf{z}}_4}, \frac{\hat{\mathbf{z}}_2}{\hat{\mathbf{z}}_1+\hat{\mathbf{z}}_2}\right), (1,1)\right\} \tag{B.7}$$

where the subscript i of the fairness–confusion tensor corresponds to the i-th element in their vector representation as in Section 3.3. By setting the objective function to be the classification error, imposing EOd fairness constraint and the model-dependent feasibility constraints in (B.6) and (B.7), MS-LFAOP is the same optimization problem as the post-processing methods, now over the space of the fairness–confusion tensors. The FACT Pareto frontier obtained by solving MS-LAFOP therefore can assess the trade-off exhibited by any classifier post-processed in such ways.

In practice, the post-processing method solves (3.8) by parameterizing $\tilde{Y}$ with two variables for each group $a = 0, 1$: $\mathsf{Pr}(\tilde{Y} = 1|\hat{Y} = 1, A = a), \mathsf{Pr}(\tilde{Y} = 1|\hat{Y} = 0, A = a)$. [47]. These values are called the *mixing rates*, as they indicate the probability of labels that should be flipped or kept for each group when post-processing the given classifier $\hat{Y}$. The algorithm then randomly selects the instances for each group to flip according to these mixing rates. These mixing rates can also be written in terms of the

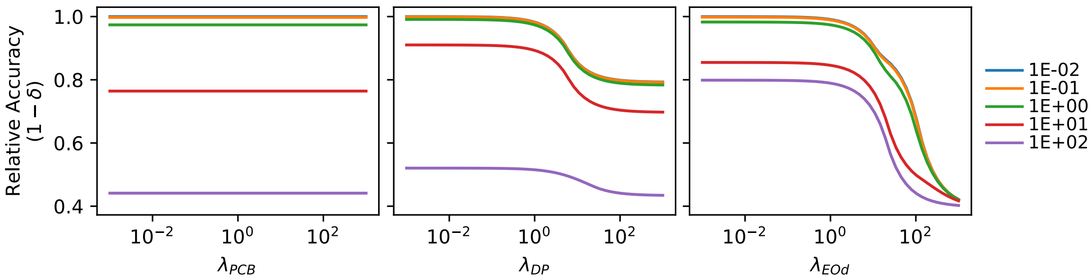


Figure B.2: The four-way trade-off between accuracy, PCB, EOd, and DP in the biased synthetic dataset (Section 3.6.1). Shown here is the $(1-\delta)$ value as a function of some regularization strength λ_ϕ for some fairness function ϕ, while holding all other $\lambda_{\phi'}$s constant (accuracy of 1.0 being the accuracy of the Bayes classifier). The value next to each colored line in the legend represents constant values for the fixed $\lambda_{\phi'}$s. Sweeping through PCB while keeping DP and EOd fixed (left) does not change the accuracy, whereas the other plots show multiple levels of variations. For EOd (right), the accuracy levels converge quickly to the limiting value of 0.392 as shown in Figure 3.2, suggesting that the accuracy is more sensitive to changes in EOd constraint strength compared to the others.

fairness–confusion tensor $\tilde{z}$ and $\hat{z}$, by using the fact that

$$\begin{aligned}\mathsf{Pr}(\tilde{Y}=\tilde{y}|Y=y,A=a) = \mathsf{Pr}(\tilde{Y}=\tilde{y}|\hat{Y}=1,A=a)\mathsf{Pr}(\hat{Y}=1|Y=y,A=a)+\\ \mathsf{Pr}(\tilde{Y}=\tilde{y}|\hat{Y}=0,A=a)\mathsf{Pr}(\hat{Y}=0|Y=y,A=a),\end{aligned}$$

and that $\mathsf{Pr}(\tilde{Y}=\tilde{y}|Y=y,A=a)$, $\mathsf{Pr}(\hat{Y}=\hat{y}|Y=y,A=a)$ terms are essentially what $\tilde{z}$ and $\hat{z}$ encode. Therefore, by using $\tilde{z}$ obtained from the MS-LAFOP above, we can compute the mixing rates to post-process the given classifier.

Appendix C

Supplementary for Chapter 4

C.1 Experiment Details

The repository of code used to generate the dataset and the results in the paper is available here[1].

C.1.1 `TextBox` Dataset

The model for each reasoning was trained on a different number of training data points, based on the number of buckets available to the dataset, as shown below in Table C.1. The total number of buckets depend on how the reasoning is set up (according to Table 4.1), i.e. how the labels are given based on the various combinations of features present/absent in the image. Based on this reasoning, primary and secondary objects are determined, which are objects in the image that should be highlighted (relevant for the prediction) and that should not be highlighted (not relevant for the prediction) respectively. If the object of interest is absent in the image, the corresponding primary/secondary metrics are not computed for further evaluations.

Name	**Simple-FR**	**Simple-NR**	**Complex-FR**	**Complex-CR1**	**Complex-CR2**	**Complex-CR3**	**Complex-CR4**
Total Training	24000	16000	36000	150000	150000	150000	150000
Training/bucket	2000	2000	2000 (for 0) 6000 (for 1)	15000	15000	15000	15000
Total Validation	6000	4000	6000	4000	4000	4000	4000
Validation/bucket	500	500	500	400	400	400	400

Table C.1: The seven model reasoning settings considered in the experiments (Section 3.6) number of data points used. The total number of data points depend on a potential class imbalance between positive and negative samples, along with the total number of buckets.

Note that while all seven settings start with twelve buckets of images (since the same set of features is used throughout), some buckets may end up with undefined labels and will not be used to train the model. For instance, Simple-NR (second column of Table 4.1) has 8 labeled buckets instead of 12, because images from four buckets without any Text (Buckets 1, 4, 7, 10 in Figure 4.5) have undefined labels given that the specified model reasoning relies on Text to produce a label. To balance the number of positive and negative samples in the dataset, more instances are sampled from buckets 10-12 for the training data for Complex-FR in particular. Finally, we note that even for a specified model reasoning, the object to focus on and/or avoid may differ across buckets. For instance, because Simple-NR relies only on Text

[1] https://github.com/wnstlr/SMERF

for predictions, Text is always the object to focus for buckets in which it is present In contrast, Box1 and Box2 are objects to avoid in buckets when they appear, but of course cannot be avoided in buckets 2 and 3 where they do not appear in the first place.

`TextBox` dataset is generated by sampling a random vector, each element indicating a specific feature the image should exhibit. The features include: the type of Text, location of the Text, color of the character, the background color, existence of Box1 and/or Box2. When randomly placing Box1 and/or Box2 on top of the Text images, the locations of these objects are constrained to avoid overlapping with one another. The resulting images have a dimension of 64-by-64 with three color channels. The pixel values are scaled to be between 0 and 1. In Figure C.1, we show some examples of data points for each reasoning, along with their labels, primary objects, and secondary objects. For each images, we also show what our trained model predicted (all of which are correct). The code for generating the dataset can be adapted to create different settings.

We show in Figure C.1 some sample data points from different buckets with bounding boxes around primary and secondary objects for each reasoning. For each images, we also show what the trained model predicted (all of which are correct).

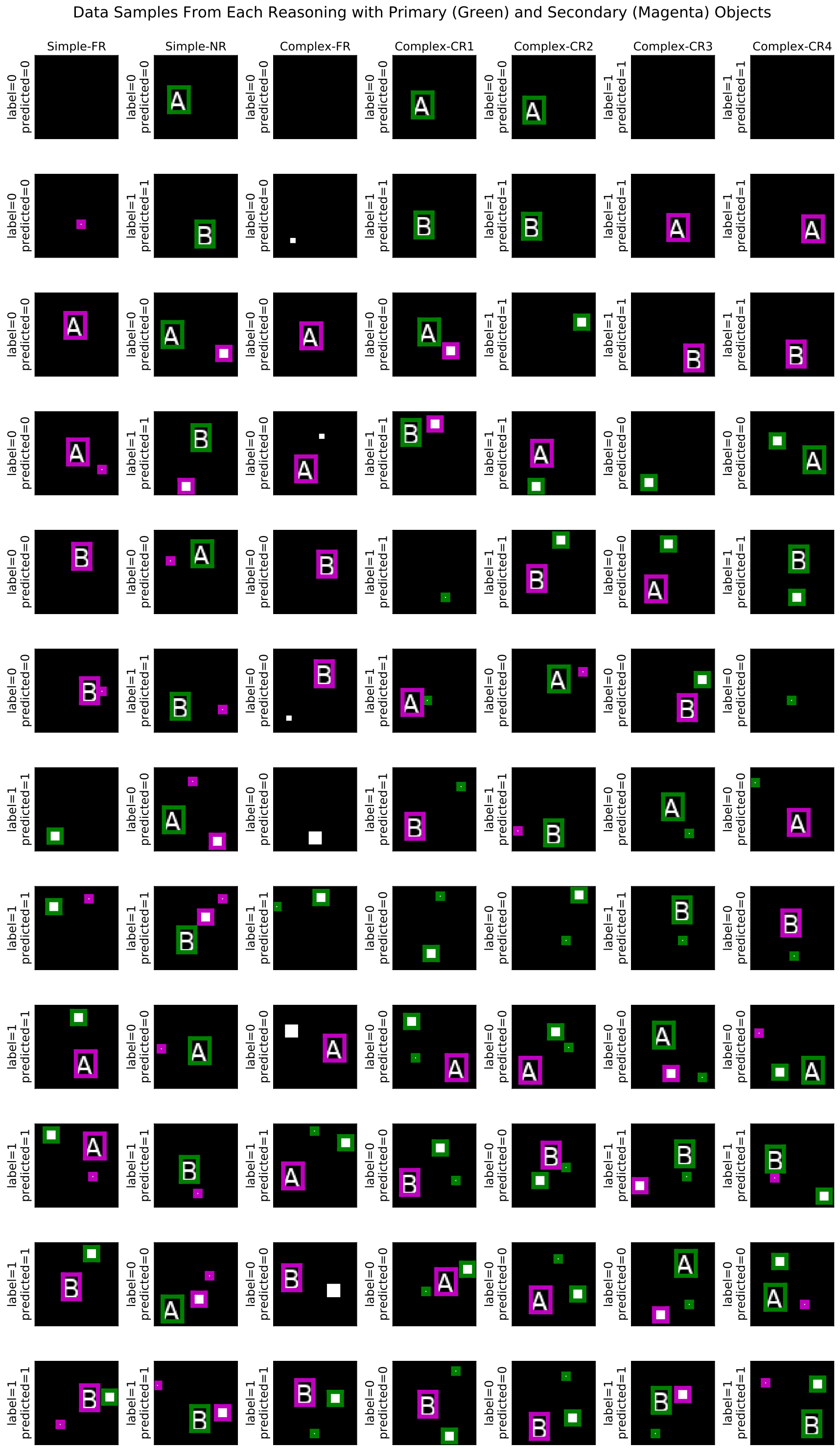


Figure C.1: Data samples from each reasoning, with bounding boxes for primary (green) and secondary (magenta) objects, along with the trained model's predictions on them (all of which are correct).

C.1.2 Model Architecture and Hyperparameters for Training

For all results presented in the Experiments section (Section 3.6), we trained separate convolutional neural networks for different types of model reasoning, but kept the same architecture of the same type. For simple reasoning, we have three convolutional layers (32 filters, kernel-size 3, stride (2,2); 64 filters, kernel-size 3, stride (2,2); 64 filters, kernel-size 3, stride (2,2)), followed by two fully-connected layers (200 units; 2 units), all with ReLU activation functions except for the output layer. For complex reasoning, however, to account for the more complex feature relationships encoded in the dataset (lower number of parameters for the model could not achieve near-perfect accuracy on these datasets) the model has more parameters: four convolutional layers (64 filters, kernel-size 3, stride (2,2); 128 filters, kernel-size 3, stride (2,2); 256 filters, kernel-size 3, stride (2,2); 64 filters, kernel-size 3, stride (2,2)), followed by three fully-connected layers (200 units; 200 units; 2 units), all with ReLU activation functions except for the output layer.

Learning rate was set as 0.0001, trained with Adam optimizer minimizing the binary cross entropy loss, with maximum epoch of 10. No particular grid search on these hyperparameters was performed, but the models were trained up to near-perfect accuracy for each model reasoning within several runs with different initializations (Table C.2 shows bucket-wise test performance of the trained model for each reasoning). SMERF overall does not require much computational load as the model sizes are not big; the entire pipeline was tested out on a machine with a single GPU (GTX 1070, 8GB), with a system memory of 16GB.

To further ensure that our findings are not affected by the specific choice of model architecture as described above, we additionally experimented with more complex model architectures: AlexNet [72] and VGG16 [116]. The results are presented in Appendix C.2.6.

	Simple-FR	**Simple-NR**	**Complex-FR**	**Complex-CR1**	**Complex-CR2**	**Complex-CR3**	**Complex-CR4**
Bucket1	1.00	-	1.00	-	-	1.00	1.00
Bucket2	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Bucket3	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Bucket4	1.00	-	1.00	1.00	-	-	0.9975
Bucket5	1.00	1.00	1.00	0.95	1.00	0.9875	0.95
Bucket6	1.00	1.00	1.00	1.00	0.9975	1.00	0.9225
Bucket7	1.00	-	1.00	-	1.00	1.00	-
Bucket8	1.00	1.00	1.00	1.00	1.00	1.00	0.9975
Bucket9	1.00	1.00	1.00	0.9575	1.00	0.995	0.9975
Bucket10	1.00	-	1.00	1.00	1.00	-	-
Bucket11	1.00	1.00	0.968	1.00	0.975	1.00	1.00
Bucket12	1.00	1.00	0.976	0.975	0.9125	0.9	0.995

Table C.2: Bucket-wise test accuracy of the trained models.

C.1.3 Saliency Methods Tested

We used various saliency methods as implemented in iNNvestigate[2] library [3], which is licensed under the BSD License. For methods that are not included in the library, we used the existing implementations for Grad-CAM[3] (Unlicensed) and DeepSHAP[4] (MIT Licensed) and integrated them into the pipeline. The details of the hyperparameters used for these methods are all available in the source code, specifically in smerf/explanations.py file. No particular grid search was performed for these hyperparameters.

[2] https://github.com/albermax/innvestigate

[3] https://github.com/wawaku/grad-cam-keras

[4] https://github.com/slundberg/shap

C.1.4 Evaluation Metric: Intersection-Over-Union (IOU) and Attribution Focus Level (AFL)

Instead of using the typical IOU value as our main metric[5], we use Attribution Focus Level (AFL), which alternatively computes the proportion of attribution values assigned to specific objects out of the total attribution value (normalized to be 1). Below we will demonstrate the advantage of using AFL over IOU in the context of `SMERF`.

IOU loses information from raw feature attribution values, so misses the impact of signals from other objects in the image.

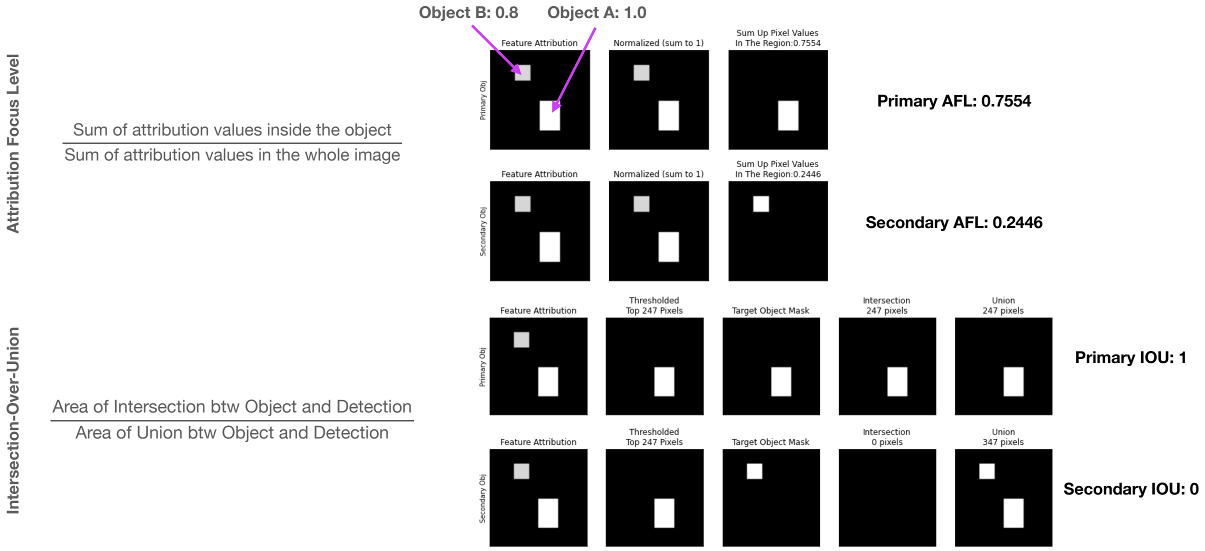


Figure C.2: Computing AFL vs IOU on a toy example where attribution values from a secondary object are non-trivial. In this example, object A (primary) and B (secondary) are both highlighted by the feature attribution as shown, where the attribution values for each pixel of A being 1, B being 0.8, and background being 0.001. AFL is computed by taking the sum of values inside the object of interest and dividing that to the sum of attribution values in the whole image. This results in PAFL value of 0.75, and SAFL value of 0.24. Notice the non-zero SAFL that indicates non-trivial signal from the secondary object B. On the other hand, typical IOU, due to its thresholding to contain only a limited number of pixels for the evaluation, fails to capture this non-trivial signal from object B, resulting in the Secondary IOU value of 0.

Consider a feature attribution which focuses on two objects A and B in an image (A being primary and B being secondary), where the attribution values of pixels inside object A are all 1, those inside object B are all 0.8, and 0.001 elsewhere in the background. The example is shown in Figure C.2. PAFL and SAFL for this feature attribution are 0.75 and 0.24 respectively. Note that the SAFL value is non-zero, correctly reflectin the fact that the feature attribution values are non-trivial on the secondary object despite most of the attribution still focused on the primary object.

However, IOU fails to correctly show this non-trivial signal on the secondary object (ie.e Secondary IOU is zero). This is mainly due to the thresholding process which selects only top-K pixels with the highest attribution values and masking only those to be included when computing the areas of intersection

[5] Nevertheless, we provide full results on IOU in Appendix C.2.1, which are consistent with our results with AFL.

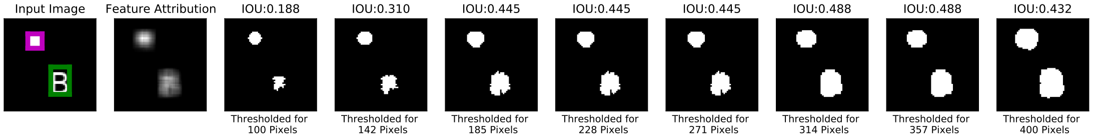

Input Image
Feature Attribution
IOU:0.188
IOU:0.310
IOU:0.445
IOU:0.445
IOU:0.445
IOU:0.488
IOU:0.488
IOU:0.432
B
Thresholded for 100 Pixels
Thresholded for 142 Pixels
Thresholded for 185 Pixels
Thresholded for 228 Pixels
Thresholded for 271 Pixels
Thresholded for 314 Pixels
Thresholded for 357 Pixels
Thresholded for 400 Pixels

Figure C.3: Change of thresholded masks based on the number of pixels to include, and the corresponding Primary IOU values on feature attributions obtained with Integrated Gradients. The IOU values are sensitive to the amount of thresholding.

and union. For this example, as shown in the image on the second column of Figure C.2 for IOU, 247 pixels are selected from thresholding, which ends up selecting only the region of object A, as the pixels within this region all have higher attribution values (1) compared to those in object B (0.8). This results in Primary IOU and Secondary IOU of 1 and 0 respectively, which fails to capture non-trivial strength of focus on the secondary object. As SMERF's model reasoning settings require analyzing the feature attributions' strength on both primary and secondary objects, it is important to correctly address this issue.

IOU is sensitive to the number of pixels to include when thresholding.

Another problem with thresholding is that it is usually unclear what the thresholding value should be. In the above example, 247 pixels are selected to include the number of pixels that consists the primary object. While this is an intuitive choice, this value can be arbitrary and the resulting IOU values can vary significantly based on the choice. Figure C.3 shows Primary IOU from a feature attribution obtained by Integrated Gradients on one of the data points where the Text is primary and the Box1 is secondary.

AFL provides more intuitive understanding about the saliency methods' level of focus among various objects.

The sum of PAFL and SAFL is upper-bounded by one, as the sum of all attribution values in the image is normalized to sum up to 1. By comparing these values, it is intuitive to deduce whether more focus is present on the primary/secondary object, or on the background. Also it is easy to understand how the level of focus shifts among these objects. Figure C.4 shows an example of how different degrees of attribution values placed on the secondary object (top) and the background (bottom) can change the corresponding PAFL and SAFL values. Note that as more we can observe PAFL decreasing and SAFL increasing, as more proportion of the total attributions are assigned to the secondary object (top). Also when the background has higher attribution values than the objects, both PAFL and SAFL values plummet.

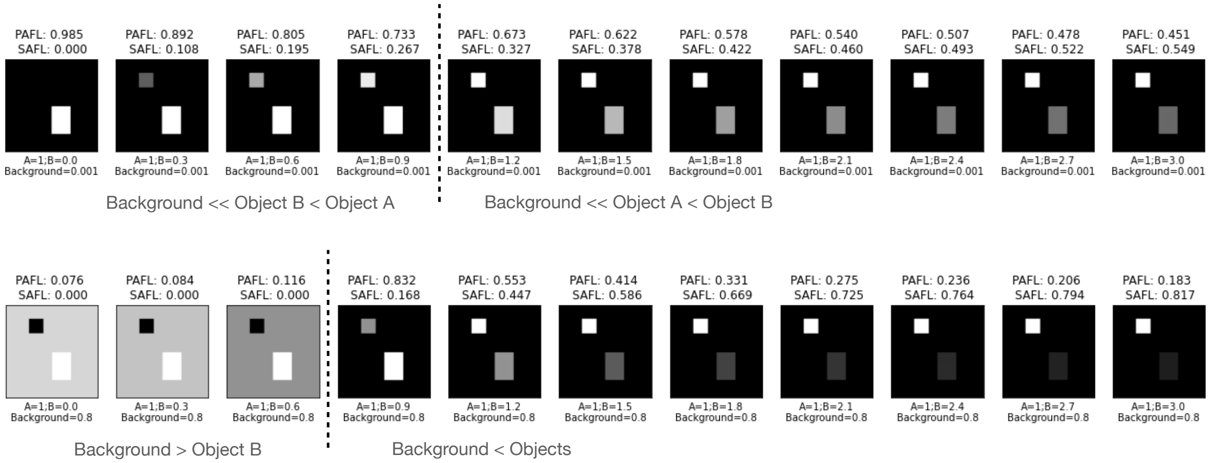


Figure C.4: Taking the example of Objects A/B from Figure C.2, (Top) as the strength of attribution values placed on object B increases, the PAFL values decrease and SAFL values increase. With the background attribution values being significantly smaller than the objects, PAFL and SAFL sums up to one. Naturally, when SAFL $>$ PAFL, it more focus is attributed to the secondary object compared to the primmary. (Bottom) When the attribution values on the background are high (higher than objects), PAFL and SAFL approaches zero as the background dominates the share in the total attribution (first three images). So when both PAFL and SAFL is low, it represents the case where most of the signals are coming from the background.

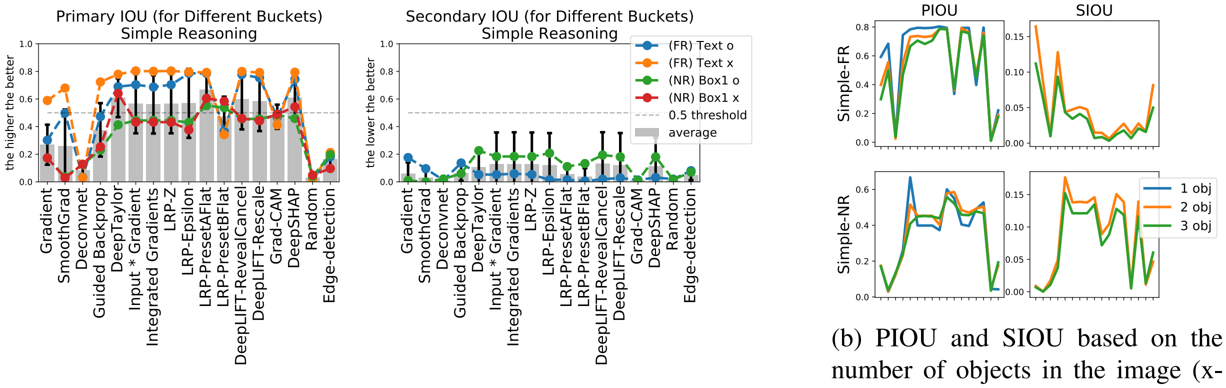
C.2 Additional Experimental Results

C.2.1 Results with IOU Metric

Intersection-Over-Union (IOU) metric is a ratio of the intersecting area to the union area of the 0-1 masked feature attribution and the ground-truth feature attribution. The ground-truth feature attributions are predefined from the data generation process (by the ground-truth model reasoning). The 0-1 masked feature attribution from the saliency methods is obtained by first blurring the original feature attribution averaged across the three color channels, followed by thresholding the pixel intensity to select top-K pixels, where K is equal to the number of pixels that correspond to the primary object. Given the 0-1 masked feature attribution and the ground-truth feature attribution, we compute two types of IOU values just like AFL: (1) primary IOU (PIOU), which measures how much of the 0-1 masked feature attribution overlaps with the ground-truth for the region *relevant* to the model prediction; and (2) secondary IOU (SIOU), which measures the same value with respect to the region *not relevant* to the model prediction. Just like AFLs, PIOU should be high and SIOU should be low for methods that are more effective and correct. Throughout this section we use the threshold value of 0.5 to roughly distinguish good and bad performance in terms of IOU as commonly done in practice [38, 136].

Figure C.5a shows PIOU and SIOU values for simple reasoning settings. Note that most methods achieve PIOU higher than 0.5, just like the results from AFL. SIOU values are mostly low throughout, even lower than what we observed from AFL. However, we notice that the IOU values fail to capture the unexpected variation of level of focus given to multiple objects based on how saturated the images are, unlike what we observed from AFLs (Figure C.5b).

Figure C.6a shows the PIOU and SIOU values for complex reasoning settings. In this case, all methods score PIOU below 0.5 correctness threshold, just like AFL. Also the minimum PIOU across different



(a) PIOU and SIOU for Simple Reasoning

(b) PIOU and SIOU based on the number of objects in the image (x-axis is the same as Figure C.5a)

Figure C.5: (a) PIOU (left) and SIOU (right) for Simple-FR and Simple-NR from Table 4.1. The black vertical lines indicate the standard deviation across different buckets. The dotted horizontal line is the correctness threshold of 0.5. Most methods perform well on average, with reasonable PIOU and low SIOU, except for a few methods on the left. (b) There is not much difference between the PIOU and SIOU values based on the number of objects in the image.

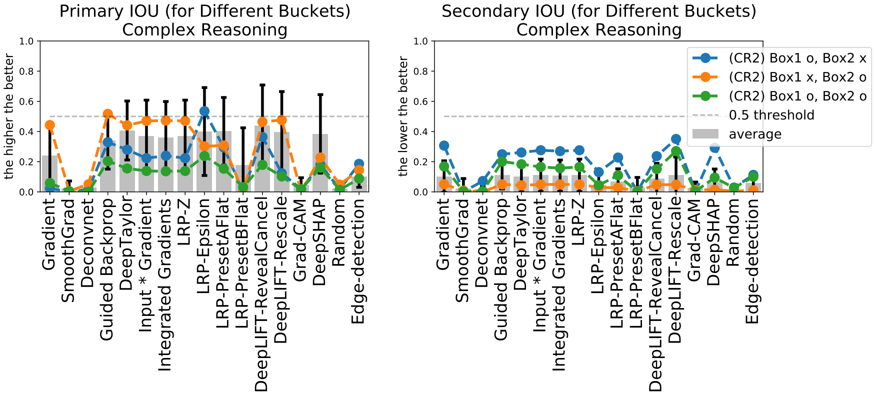
buckets from complex reasoning is strictly lower than those from simple reasoning for all methods (Figure C.6b). Therefore, general trend of methods being reasonably good for simple reasoning and being bad for complex reasoning still is observable using IOU. However, note that the SIOU values remain relatively low for both cases despite the decrease in PIOU for complex reasoning. This is where the aforementioned limitation of IOU metric is exposed, where it is difficult to clearly deduce where the changes in the primary values are coming from, contrary to what we observe from AFL.

Relatedly, IOU metric is relatively more generous to primary objects due to the way they are thresholded and it is more prone to ignoring potential non-trivial signals from objects that are not primary. This naturally leads to low SIOU values and a lack of clear relationship between primary and secondary metric: an increase in one would not necessarily imply a decrease in another, and vice-versa. In addition, the loss of information from the attribution values makes it difficult to better understand why one value decreased and how much of that change could be attributed to different objects in the image. Due to these limitations (as well as examples mentioned earlier in Appendix C.1.4, we present AFL results in the main text for better insights and highlighting the shortcomings of the methods in different settings.

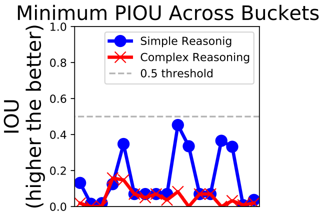
C.2.2 Qualitative Results

Figure C.7 and C.8 show samples from simple reasoning (each from Simple-FR and Simple-NR) in buckets where the methods record the lowest PAFL (indicated with red). For instance, in Simple-FR setup (Figure C.7) almost all methods achieve the lowest PAFL on the bucket where all features are present (top panel). Recall that Simple-FR relies fully on Box1; however we observe that several methods highlight all objects that are present in the image to a certain degree. Such tendency explains the unexpected variation of AFL based on the number of objects present in the image. Nevertheless, some minimum PAFL values are still above 0.5 for a method like LRP-Epsilon, which makes it the most effective method for Simple-FR that is less prone to errors (it also correctly focuses on Box1 without being distracted by other objects in the image). Compared to Simple-FR, Simple-NR shows more evenly distributed failure cases across different buckets shown (Figure C.8).

In Figure C.9 we have the samples from buckets with minimum PAFL for complex reasoning setup,



(a) PIOU and SIOU for Complex Reasoning



(b) Minimum PIOU across buckets on simple (blue circle) vs. complex (red x) reasoning (x-axis is the same as Figure C.6a).

Figure C.6: (a) PIOU and SIOU for complex reasoning. Average performance on PIOU is mostly lower than 0.5, with worse performance compared to the results from simple reasoning. (b) Worst-case buckets show worse PIOU values in complex reasoning.

in particular, Complex-CR2. This time most of the cases with minimum performance is concentrated on the bucket with Box1 and Text B (top panel; 13 out of 15 methods have minimum PAFL on this bucket). Recall that for this bucket, the ground-truth feature attribution should highlight just Box1. For most of the methods, more focus is given to the Text, which results in a particularly low PAFL across all methods. Notice that all the values are below 0.5: this aligns with our earlier observation that the worst-case performance of the methods on complex reasoning setup is much worse than simple reasoning setup.

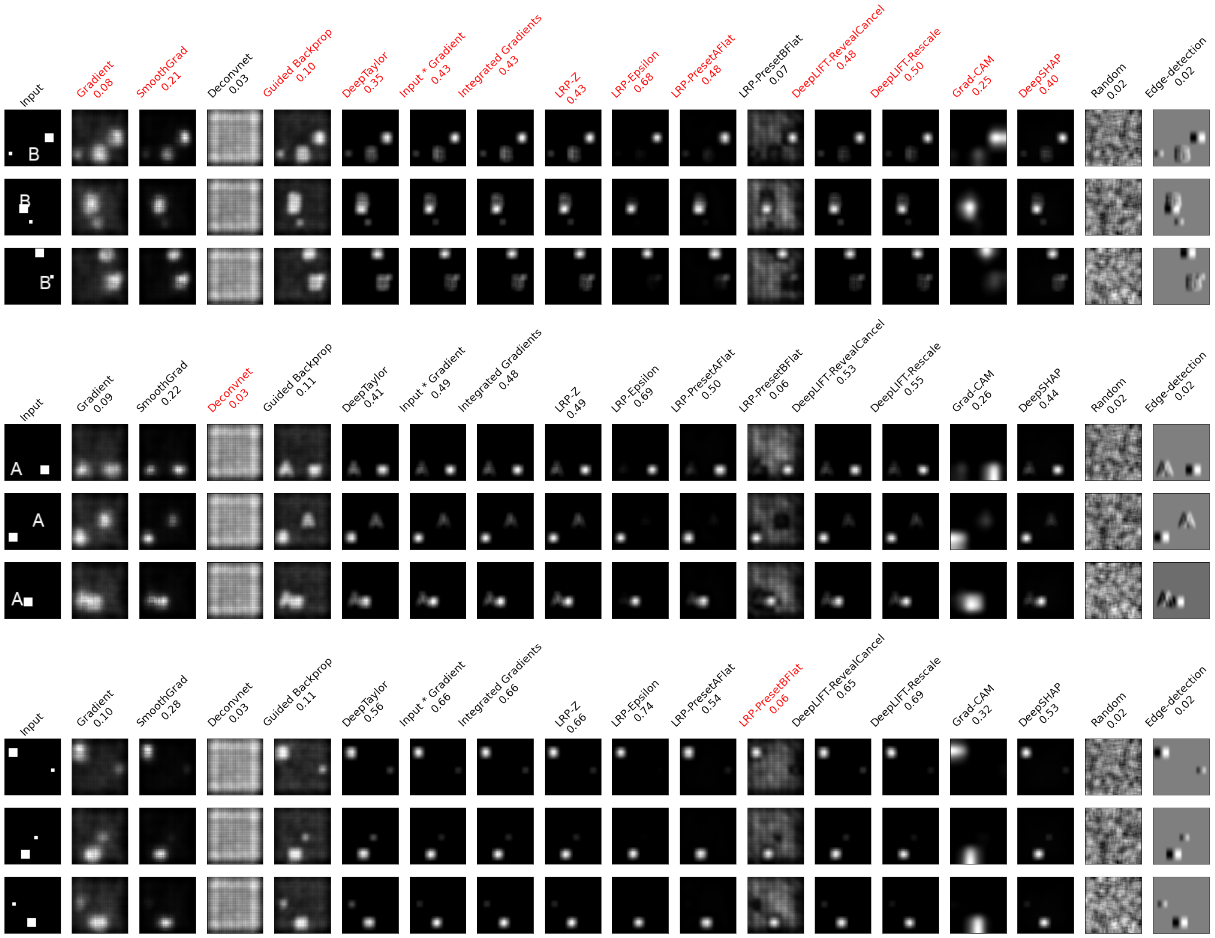


Figure C.7: Image samples from buckets where minimum PAFL values are recorded for different methods (marked with red), for FR-Simple. The ground-truth is to focus on Box1 only.

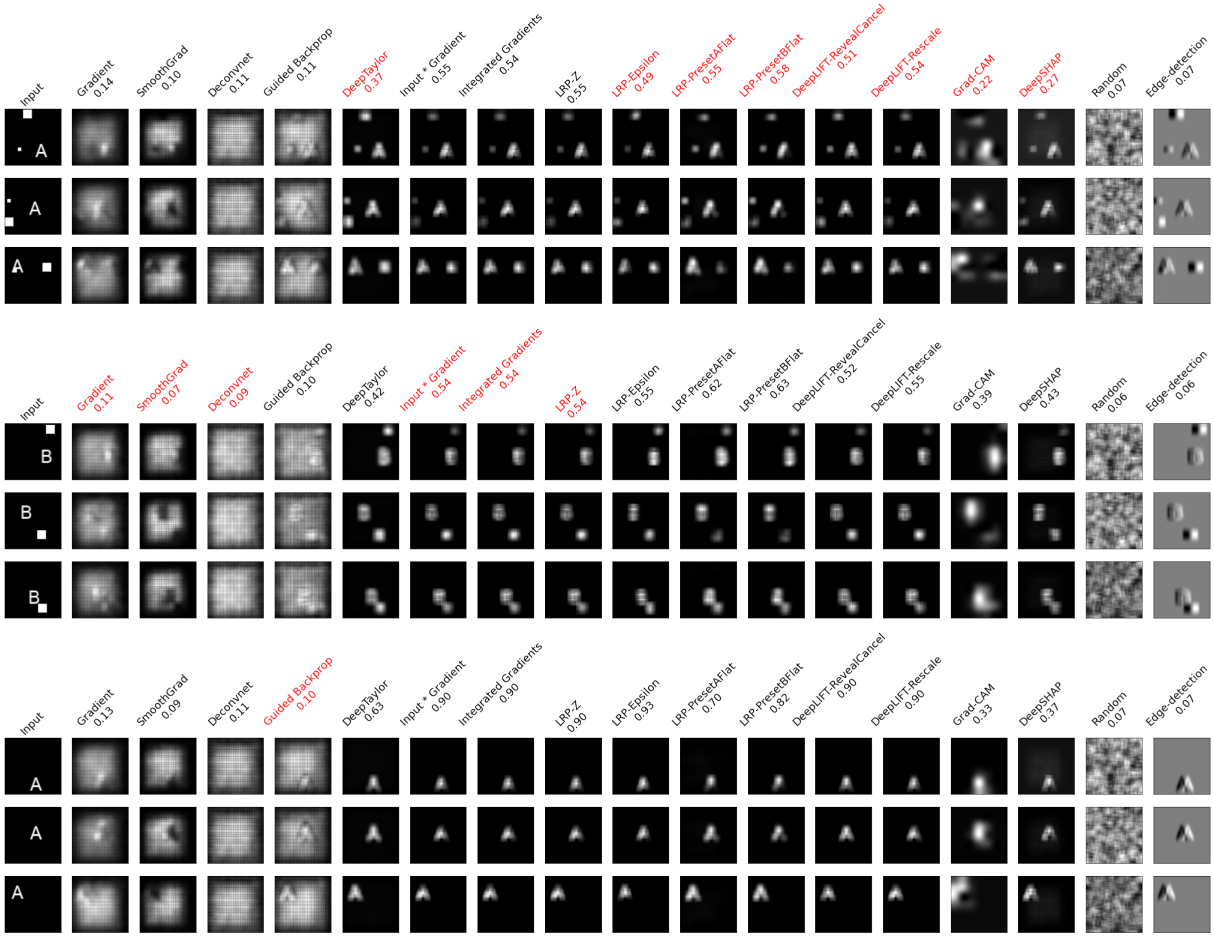


Figure C.8: Image samples from buckets where minimum PAFL values are recorded for different methods (marked with red, along with corresponding PAFL values), for NR-Simple. The ground-truth is to focus on the Text only.

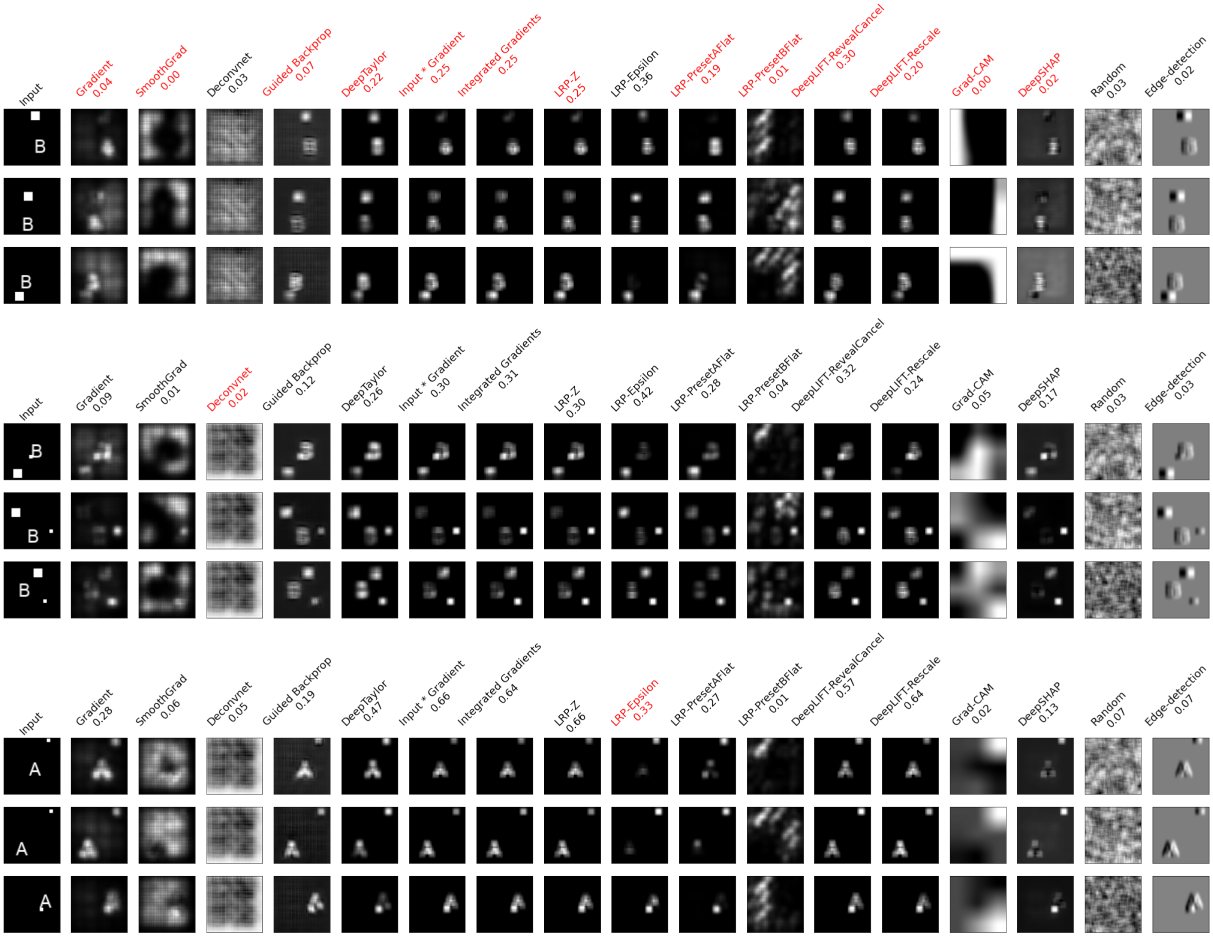


Figure C.9: Image samples from buckets where minimum PAFL values are recorded for different methods (marked with red), for CR-Complex2. The ground-truth is to focus on the Box1 and Box2 if Box1 is present, otherwise on Text.

C.2.3 Per-bucket Performance Variation for Complex Reasoning

In Figure C.10, we show additional details from different types of complex reasoning that show high variance of PAFL and SAFL we observed from Figure 4.8a. From top to bottom, we plot the variation of the values per bucket for Complex-CR1, Complex-CR3, Complex-CR4, and Complex-FR, in addition to the results from Complex-CR2 reported in the main text. We similarly observe that the variation is quite extreme for different buckets.

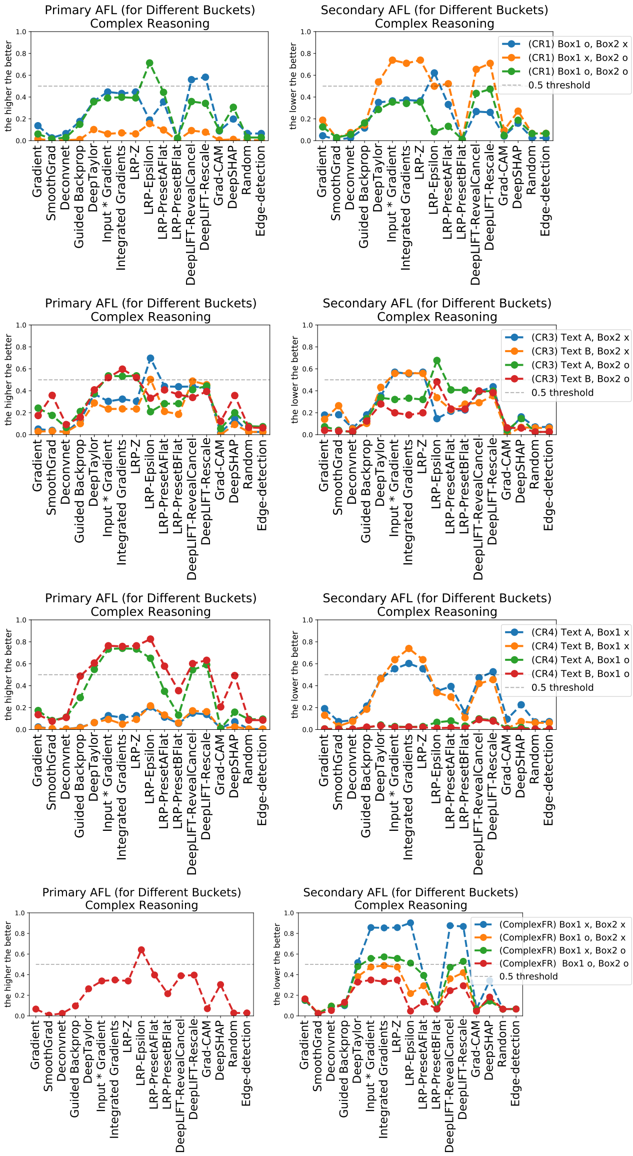


Figure C.10: Additional information for the variance of PAFL and SAFL across different buckets for complex reasoning (from top to bottom, Complex-CR1, Complex-CR3, Complex-CR4, Complex-FR). The performance varies significantly from bucket to bucket, just as we described in the main text.

C.2.4 Additional Failure Case Analysis

While the AFL may be more suited to give a high-level understanding of the methods' performance, to further understand the degree to which the model focuses more on the incorrect region compared to the correct region (i.e. to identify failure due to wrong focus), we also compute the mean of the attribution values inside the relevant (which we call *Primary Mean-AFL, or PMAFL*) and irrelevant (which we call *Secondary Mean-AFL, or SMAFL*) regions for comparison. For a successful method, primary MAFL should always upper bound secondary MAFL because on average the correct regions should always be assigned higher attribution values compared to the incorrect regions. In this analysis, the actual values of primary and secondary MAFL do not matter; only a relative comparison does.

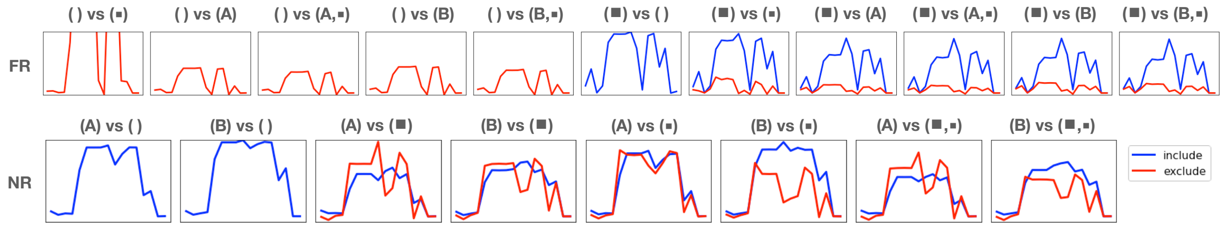


Figure C.11: MAFL value comparison within buckets for simple reasoning with FR and NR. Title of each panel indicates the bucket with "(correct feature) vs (incorrect feature)", and the blue and red lines show the general trend of primary and secondary MAFL values across different methods (x-axis being the same as Figure 4.6a, omitted for brevity). Primary MAFL should ideally always upper-bound secondary MAFL: while that is the case for FR, there are occasional failure cases in NR where the opposite happens.

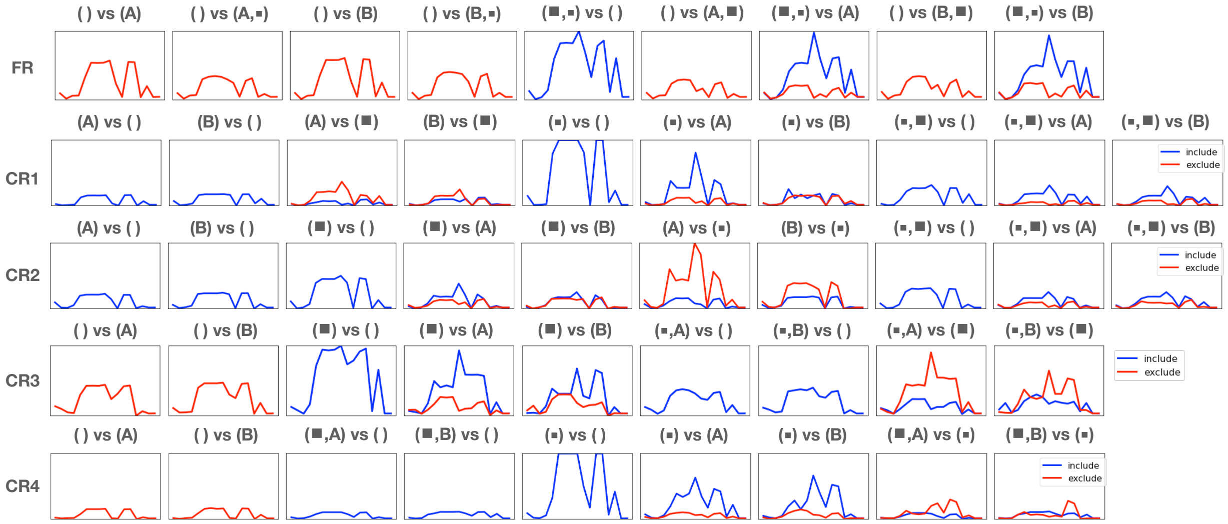


Figure C.12: MAFL values for complex reasoning. While PMAFL (blue) should be always bigger than SMAFL (red), some buckets exhibit much larger SMAFL. This means the methods are failing due to focusing significantly more on irrelevant regions features than relevant ones.

Figure C.11 plots MAFL value comparison for different buckets in Simple-NR and Simple-FR. It is important that the blue line (PMAFL) should always upper-bound the red line (SMAFL). While that is the case for FR for all buckets, there are occasional cases in NR with the opposite relationship for certain methods. Although the gap between the values are not big, this still indicates that the feature attribution may sometimes be wrongly focusing on irrelevant regions of the image more than the correct ones. Such failure cases have not been actively discussed in previous works that dealt models with simple reasoning [2, 142].

As further shown in Figure C.12 it is observed that the gap between PMAFL (blue) and SMAFL (red) is bigger for some buckets in complex reasoning compared to the simple reasoning case in Figure C.11. Such larger gaps clearly demonstrate and verify the methods' more frequent failure due to wrong focus from complex reasoning.

C.2.5 Identifiability Problem Samples

Figure C.13 shows samples that illustrate the difficulty of clearly distinguishing model reasoning based on the feature attributions (i.e. identifiability). Along with Integrated Gradients (as in Figure 4.9), all other methods including the ones presented here (Gradients, LRP-Z, DeepLIFT) highlight all objects in the image up to a certain level regardless of the model reasoning used, making it difficult to discern the reasoning based on the feature attributions.

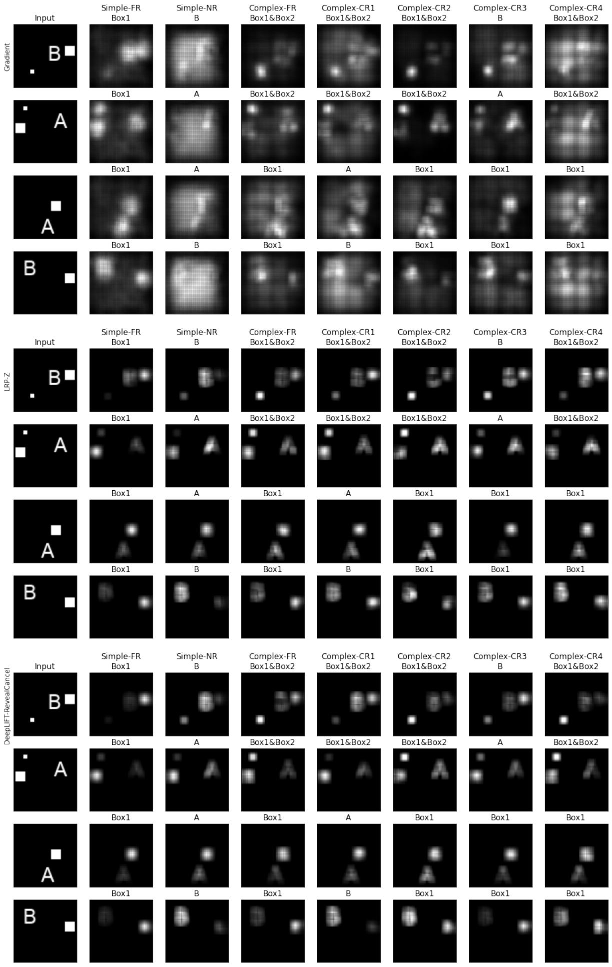


Figure C.13: Identifiability problem samples with other methods (Gradient, LRP-Z, DeepLIFT from top to bottom). All methods highlight all objects in the image up to a certain degree regardless of the model reasoning, making it difficult for the users to clearly distinguish among different types of model reasoning used.

C.2.6 Results on Other Model Architectures

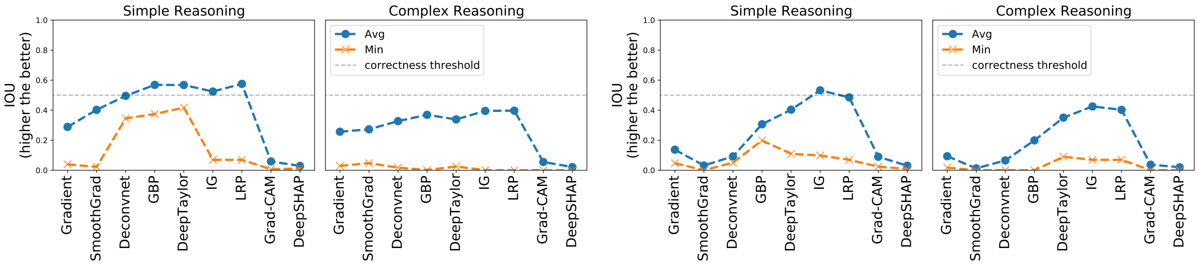


Figure C.14: IOU results using AlexNet (left) and VGG16 (right) architecture for the model.

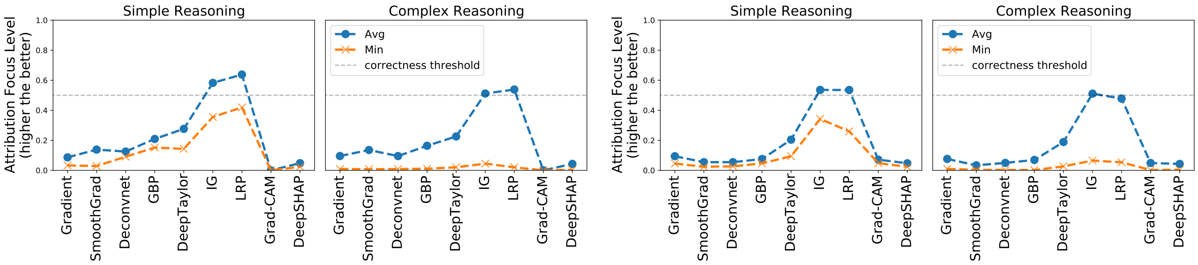


Figure C.15: PAFL results using AlexNet (left) and VGG16 (right) architecture for the model.

We repeat the same set of experiments with the model architecture using AlexNet [72] and VGG16 [116], to confirm that the trends we observe for the saliency methods are not the artifact of architecture choice. IOU and AFL results for these models are shown in Figures C.14 and C.15[6], which show similar trends we have observed so far in the simple CNN case: the average and the worst-case performance drops as the reasoning gets more complex.

[6] Note that DeepLIFT was left out from the experiments on these models as the library for DeepLIFT does not support MaxPooling2D layer at the moment.

C.2.7 Relationship with Adversarial Robustness

Shah et al. [109] showed that gradient feature attributions applied on more adversarially robust models tend to do better in ignoring the signals from spurious objects in the image. We verify that such trend for gradient attribution is somewhat true, yet the general problem persists for most of the other methods even for the robust models, in both simple and complex settings.

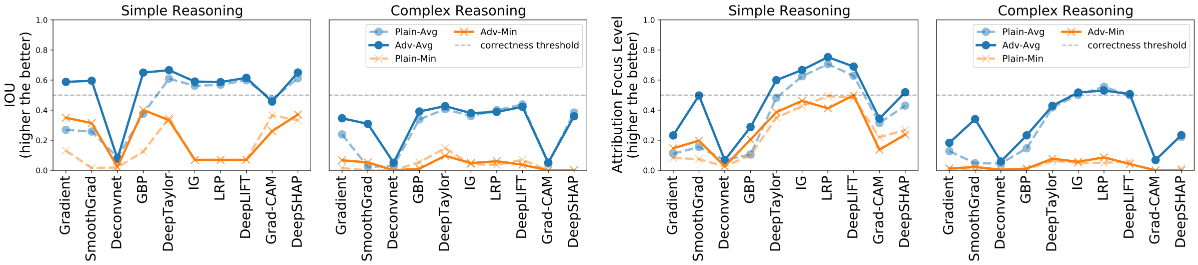


Figure C.16: IOU (left) and PAFL (right) results on models trained against PGD attack (solid lines) and plain models (dotted lines, taken from Figure 4.2) for simple and complex reasoning settings. We observe increase in average performance for Gradient and SmoothGradient for simple reasoning compared to plain models, as suggested in [109]. Nevertheless, in both metrics and across all methods, there still are sharp performance drops for complex reasoning.

Figure C.16 shows results for models trained against PGD attacks [86][7]. Throughout all methods, we still observe sharp performance drops in complex reasoning. We also observe the trend reported in [109] as well, where Gradient and SmoothGradient's performance for simple reasoning (solid lines) is generally higher compared to plain models (dotted lines). As shown, while training the model against adversarial attacks can help some methods in simple reasoning settings, SMERF suggests that the overarching problem of methods not being able to reliably recover complex reasoning still persists.

[7] Used a Python library https://github.com/Trusted-AI/adversarial-robustness-toolbox

C.2.8 `TextBox` with Noisy Background

Instead of having zero-valued black pixels for the background, we set the background to consist of random pixel values between 0-150 for each of the RGB channels (before being normalized to [0,1]).

The models were trained on these images to achieve near-perfect accuracy for all seven types of reasoning. Figure C.17 shows PAFL and SAFL for simple reasoning settings, and Figure C.18 for complex reasoning settings. Similar to our earlier observations, methods under complex reasoning settings show sharp degradation of performance compared to the simple reasoning settings (lower PAFL, higher SAFL). Also we notice that additional noise from the background lowers the worst-case PAFL values throughout, even for the simple reasoning settings (Figure C.18b).

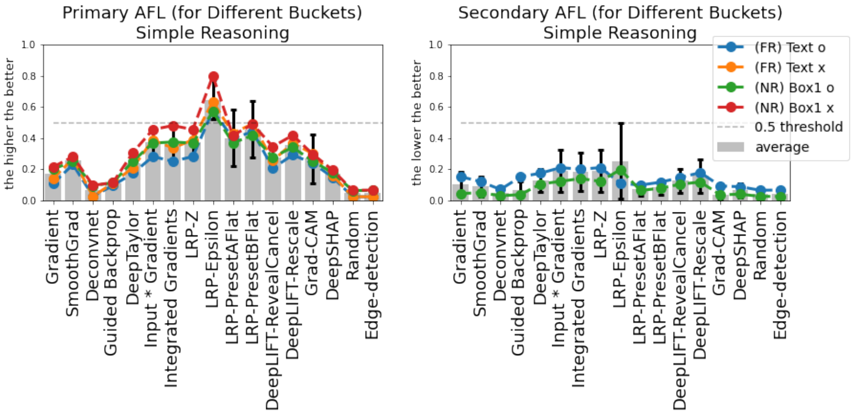
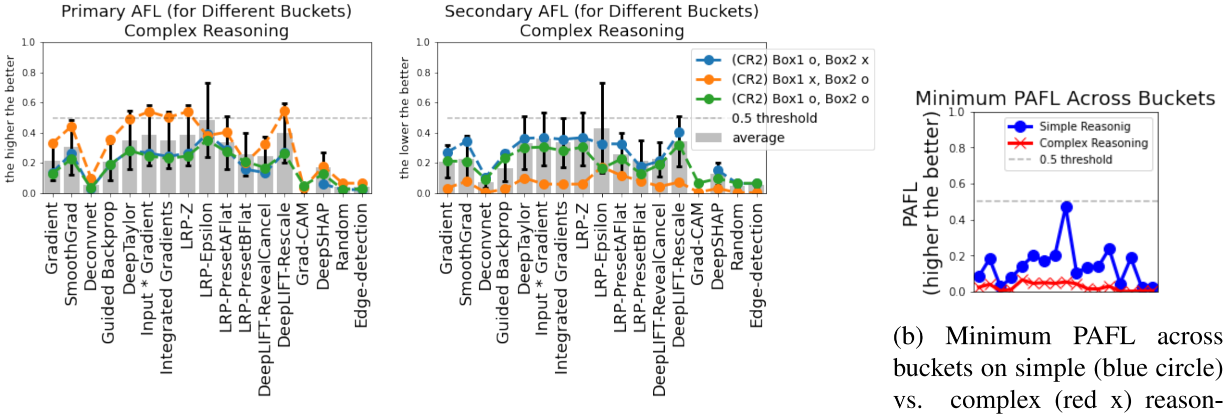


Figure C.17: PAFL and SAFL for Simple Reasoning, tested with non-zero random noisy background.



(a) PAFL and SAFL for Complex Reasoning, tested with non-zero random background.

(b) Minimum PAFL across buckets on simple (blue circle) vs. complex (red x) reasoning (x-axis is the same as Figure C.18a).

Figure C.18: (a) PAFL and SAFL for complex reasoning. (b) Worst-case buckets show worse PAFL values in complex reasoning.

C.2.9 `TextBox` with Realistic Backgrounds

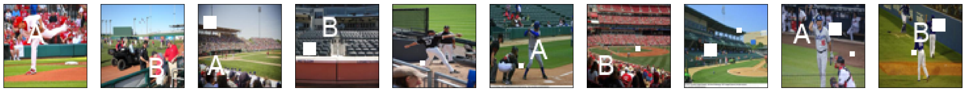


Figure C.19: Images with real backgrounds used for the experiments.

We replace the background of `TextBox` images with real images of different scenes taken from the Places dataset[8][154]. In particular, we replaced the background with images of baseball stadium and ran the same set of experiments (Figure C.19). The models trained on these images achieved near-perfect accuracy on both simple and complex settings (test accuracy for each reasoning: Simple-FR: 0.99, Simple-NR: 0.99, Complex-FR: 0.93, Complex-CR1: 0.98, Complex-CR2: 0.99, Complex-CR3: 0.93, Comlpex-CR4: 0.99).

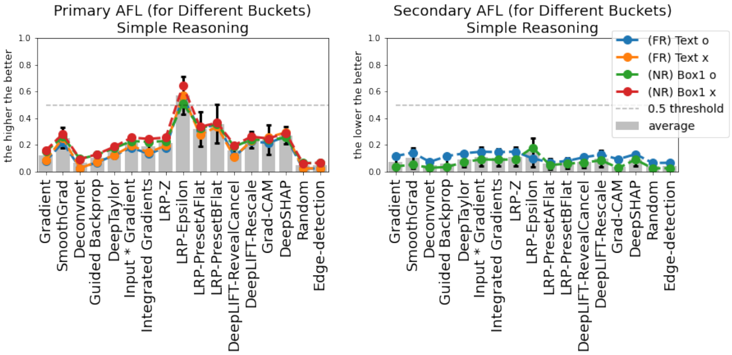
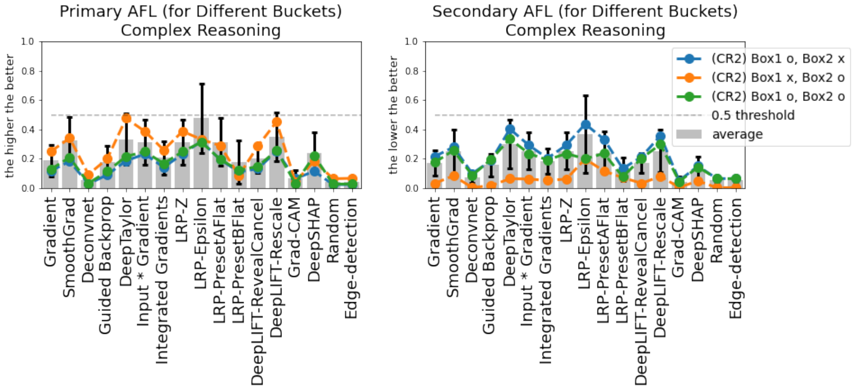
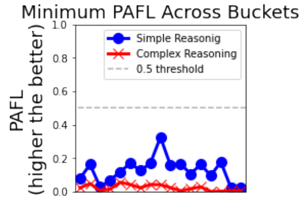


Figure C.20: PAFL and SAFL for Simple Reasoning, for images with real baseball stadium as background.



(a) PAFL and SAFL for Complex Reasoning, tested with real background of baseball stadiums.



(b) Minimum PAFL across buckets on simple (blue circle) vs. complex (red x) reasoning (x-axis is the same as Figure C.21a).

Figure C.21: (a) PAFL and SAFL for complex reasoning, for images with real baseball stadium as background. (b) Worst-case buckets show overall lower PAFL values in complex reasoning compared to simple reasoning.

[8] http://places2.csail.mit.edu/

Figure C.20 and Figure C.21 respectively shows AFL results on simple and complex reasoning settings. Figure C.21b in particular shows that similar to the black background scenario studied earlier, there is a performance degradation moving from simple to complex reasoning, but at a lower level compared to the black background scenario. Notably, the general performance drop relative to the black background setting was larger in this case than what we observed for the noisy background setting in Appendix C.2.8. The results suggest that synthetic results of `SMERF` on the black background provides an optimistic upper bound for the methods' performance on the real background.

Appendix D

Supplementary for Chapter 5

D.1 Proxy Test of Black-box Model Explanations

There are several black-box model explanations to consider for the task. While testing all of them on real users can be an interesting research on its own, as we are more broadly interested in what distinct types of information could be helpful, we decide to select one representative method in the literature. As there is no absolute answer to which method is superior, we conduct a simple proxy test of what method can be a better choice for the task.

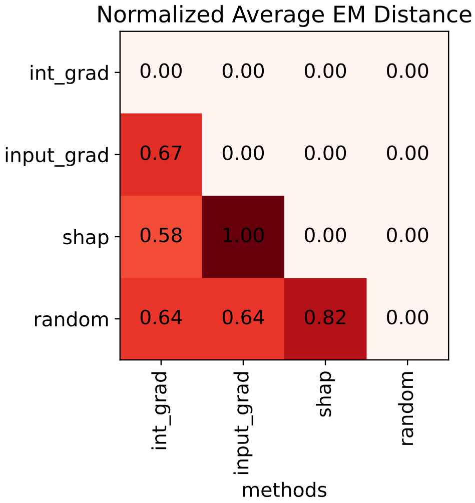


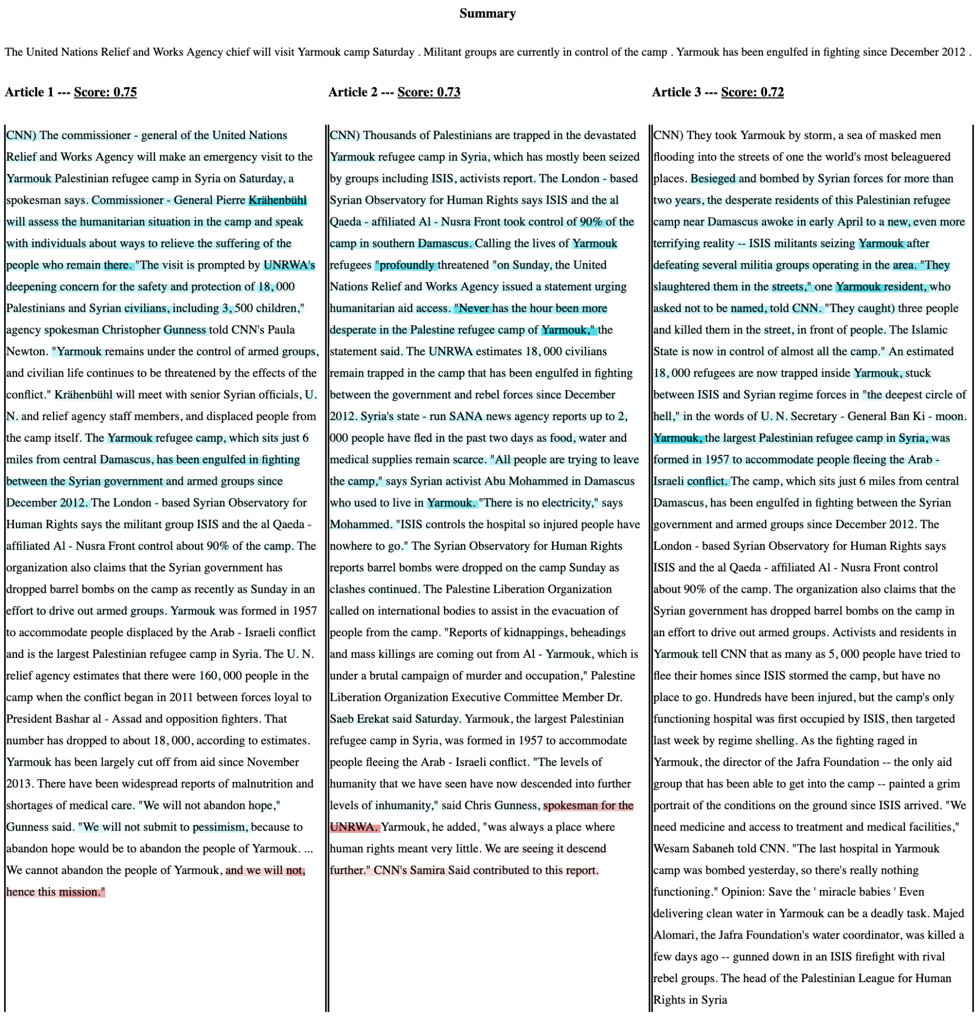
Figure D.1: Proxy quality test for the black-box model explanations using average EM distances between the distributions of attribution scores of input tokens. The higher the values (the darker the color), the more different the distribution of the attribution scores. SHAP shows the most distinct distribution from the random attributions (bottom row).

We consider the following feature attribution methods: Integrated Gradients [124], Input x Gradients [113], and SHAP [83]. In Figure D.1, we plot the mean EM distance (averaged across 50 different random attributions, normalized to be between 0 and 1) between the distribution of attribution scores for the input tokens in our ground-truth articles. The higher the value (darker the color), the more distinct the distribution of the attribution scores computed by respective methods. Notice that SHAP shows the most distinct distribution from random attributions compared to other methods, indicating it may be a better choice that carry more information about the important tokens. We have also qualitatively verified that the highlights from other two methods were not as meaningful as SHAP on the articles.

Note also that SHAP is a promising candidate to apply to the task due to its popularity and its common presence in more sophisticated domains like biology, physics, chemistry, and finance [53, 93, 99, 141, 144].

D.2 Method Examples

We show below some example highlights presented to the users using different methods.



Summary

The United Nations Relief and Works Agency chief will visit Yarmouk camp Saturday . Militant groups are currently in control of the camp . Yarmouk has been engulfed in fighting since December 2012 .

Article 1 --- Score: 0.75

CNN) The commissioner - general of the United Nations Relief and Works Agency will make an emergency visit to the Yarmouk Palestinian refugee camp in Syria on Saturday, a spokesman says. Commissioner - General Pierre Krähenbühl will assess the humanitarian situation in the camp and speak with individuals about ways to relieve the suffering of the people who remain there. "The visit is prompted by UNRWA's deepening concern for the safety and protection of 18, 000 Palestinians and Syrian civilians, including 3, 500 children," agency spokesman Christopher Gunness told CNN's Paula Newton. "Yarmouk remains under the control of armed groups, and civilian life continues to be threatened by the effects of the conflict." Krähenbühl will meet with senior Syrian officials, U. N. and relief agency staff members, and displaced people from the camp itself. The Yarmouk refugee camp, which sits just 6 miles from central Damascus, has been engulfed in fighting between the Syrian government and armed groups since December 2012. The London - based Syrian Observatory for Human Rights says the militant group ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90% of the camp. The organization also claims that the Syrian government has dropped barrel bombs on the camp as recently as Sunday in an effort to drive out armed groups. Yarmouk was formed in 1957 to accommodate people displaced by the Arab - Israeli conflict and is the largest Palestinian refugee camp in Syria. The U. N. relief agency estimates that there were 160, 000 people in the camp when the conflict began in 2011 between forces loyal to President Bashar al - Assad and opposition fighters. That number has dropped to about 18, 000, according to estimates. Yarmouk has been largely cut off from aid since November 2013. There have been widespread reports of malnutrition and shortages of medical care. "We will not abandon hope," Gunness said. "We will not submit to pessimism, because to abandon hope would be to abandon the people of Yarmouk. ... We cannot abandon the people of Yarmouk, and we will not, hence this mission."

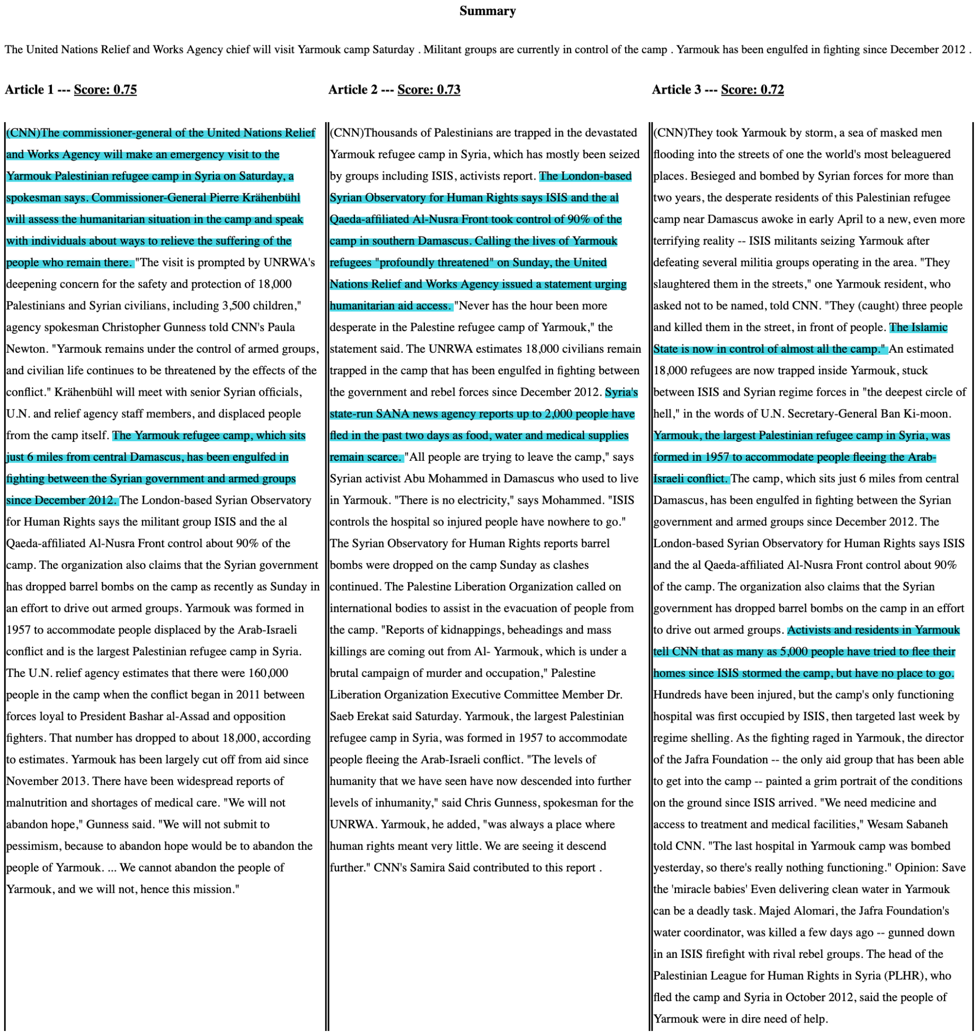
Article 2 --- Score: 0.73

CNN) Thousands of Palestinians are trapped in the devastated Yarmouk refugee camp in Syria, which has mostly been seized by groups including ISIS, activists report. The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front took control of 90% of the camp in southern Damascus. Calling the lives of Yarmouk refugees "profoundly threatened "on Sunday, the United Nations Relief and Works Agency issued a statement urging humanitarian aid access. "Never has the hour been more desperate in the Palestine refugee camp of Yarmouk," the statement said. The UNRWA estimates 18, 000 civilians remain trapped in the camp that has been engulfed in fighting between the government and rebel forces since December 2012. Syria's state - run SANA news agency reports up to 2, 000 people have fled in the past two days as food, water and medical supplies remain scarce. "All people are trying to leave the camp," says Syrian activist Abu Mohammed in Damascus who used to live in Yarmouk. "There is no electricity," says Mohammed. "ISIS controls the hospital so injured people have nowhere to go." The Syrian Observatory for Human Rights reports barrel bombs were dropped on the camp Sunday as clashes continued. The Palestine Liberation Organization called on international bodies to assist in the evacuation of people from the camp. "Reports of kidnappings, beheadings and mass killings are coming out from Al - Yarmouk, which is under a brutal campaign of murder and occupation," Palestine Liberation Organization Executive Committee Member Dr. Saeb Erekat said Saturday. Yarmouk, the largest Palestinian refugee camp in Syria, was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict. "The levels of humanity that we have seen have now descended into further levels of inhumanity," said Chris Gunness, spokesman for the UNRWA. Yarmouk, he added, "was always a place where human rights meant very little. We are seeing it descend further." CNN's Samira Said contributed to this report.

Article 3 --- Score: 0.72

CNN) They took Yarmouk by storm, a sea of masked men flooding into the streets of one the world's most beleaguered places. Besieged and bombed by Syrian forces for more than two years, the desperate residents of this Palestinian refugee camp near Damascus awoke in early April to a new, even more terrifying reality -- ISIS militants seizing Yarmouk after defeating several militia groups operating in the area. "They slaughtered them in the streets," one Yarmouk resident, who asked not to be named, told CNN. "They caught) three people and killed them in the street, in front of people. The Islamic State is now in control of almost all the camp." An estimated 18, 000 refugees are now trapped inside Yarmouk, stuck between ISIS and Syrian regime forces in "the deepest circle of hell," in the words of U. N. Secretary - General Ban Ki - moon. Yarmouk, the largest Palestinian refugee camp in Syria, was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict. The camp, which sits just 6 miles from central Damascus, has been engulfed in fighting between the Syrian government and armed groups since December 2012. The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90% of the camp. The organization also claims that the Syrian government has dropped barrel bombs on the camp in an effort to drive out armed groups. Activists and residents in Yarmouk tell CNN that as many as 5, 000 people have tried to flee their homes since ISIS stormed the camp, but have no place to go. Hundreds have been injured, but the camp's only functioning hospital was first occupied by ISIS, then targeted last week by regime shelling. As the fighting raged in Yarmouk, the director of the Jafra Foundation -- the only aid group that has been able to get into the camp -- painted a grim portrait of the conditions on the ground since ISIS arrived. "We need medicine and access to treatment and medical facilities," Wesam Sabaneh told CNN. "The last hospital in Yarmouk camp was bombed yesterday, so there's really nothing functioning." Opinion: Save the ' miracle babies ' Even delivering clean water in Yarmouk can be a deadly task. Majed Alomari, the Jafra Foundation's water coordinator, was killed a few days ago -- gunned down in an ISIS firefight with rival rebel groups. The head of the Palestinian League for Human Rights in Syria

Figure D.2: Example highlights for SHAP.



Summary

The United Nations Relief and Works Agency chief will visit Yarmouk camp Saturday . Militant groups are currently in control of the camp . Yarmouk has been engulfed in fighting since December 2012 .

Article 1 --- Score: 0.75

(CNN)The commissioner-general of the United Nations Relief and Works Agency will make an emergency visit to the Yarmouk Palestinian refugee camp in Syria on Saturday, a spokesman says. Commissioner-General Pierre Krähenbühl will assess the humanitarian situation in the camp and speak with individuals about ways to relieve the suffering of the people who remain there. "The visit is prompted by UNRWA's deepening concern for the safety and protection of 18,000 Palestinians and Syrian civilians, including 3,500 children," agency spokesman Christopher Gunness told CNN's Paula Newton. "Yarmouk remains under the control of armed groups, and civilian life continues to be threatened by the effects of the conflict." Krähenbühl will meet with senior Syrian officials, U.N. and relief agency staff members, and displaced people from the camp itself. The Yarmouk refugee camp, which sits just 6 miles from central Damascus, has been engulfed in fighting between the Syrian government and armed groups since December 2012. The London-based Syrian Observatory for Human Rights says the militant group ISIS and the al Qaeda-affiliated Al-Nusra Front control about 90% of the camp. The organization also claims that the Syrian government has dropped barrel bombs on the camp as recently as Sunday in an effort to drive out armed groups. Yarmouk was formed in 1957 to accommodate people displaced by the Arab-Israeli conflict and is the largest Palestinian refugee camp in Syria. The U.N. relief agency estimates that there were 160,000 people in the camp when the conflict began in 2011 between forces loyal to President Bashar al-Assad and opposition fighters. That number has dropped to about 18,000, according to estimates. Yarmouk has been largely cut off from aid since November 2013. There have been widespread reports of malnutrition and shortages of medical care. "We will not abandon hope," Gunness said. "We will not submit to pessimism, because to abandon hope would be to abandon the people of Yarmouk. ... We cannot abandon the people of Yarmouk, and we will not, hence this mission."

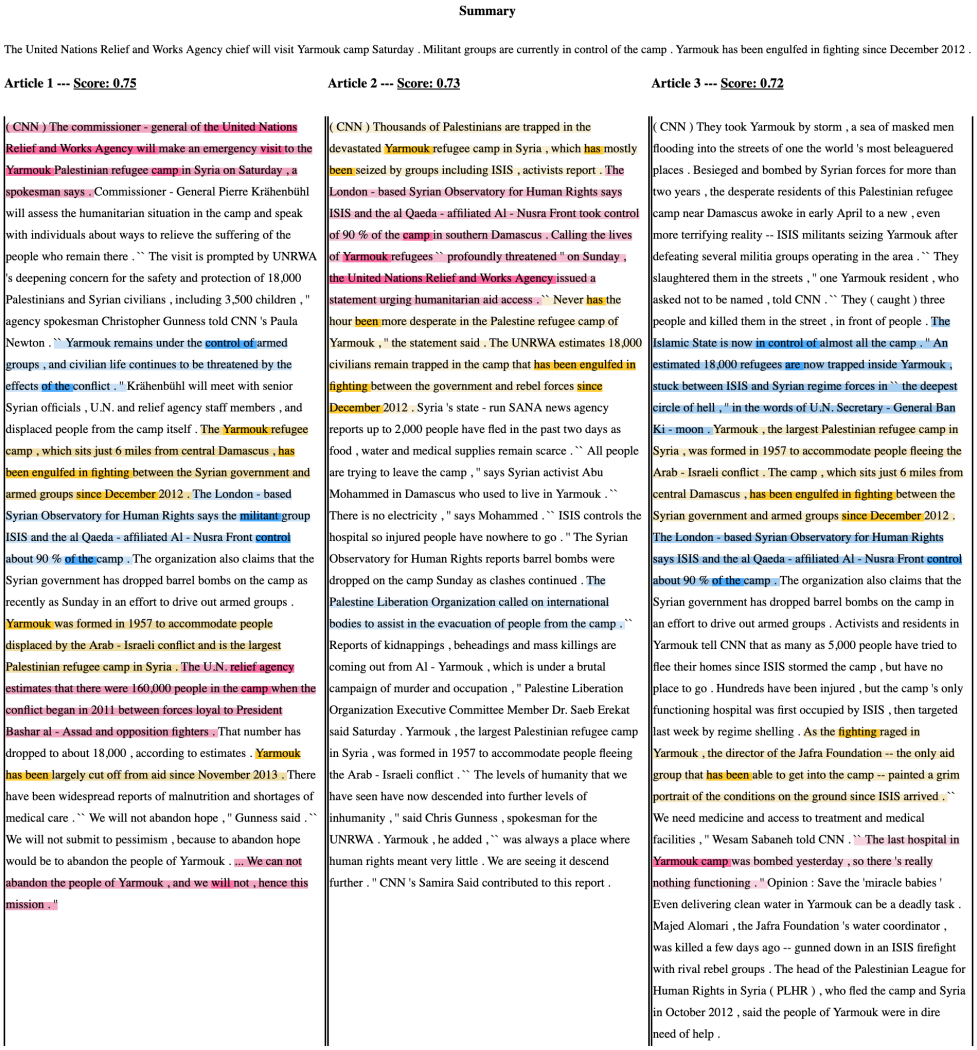
Article 2 --- Score: 0.73

(CNN)Thousands of Palestinians are trapped in the devastated Yarmouk refugee camp in Syria, which has mostly been seized by groups including ISIS, activists report. The London-based Syrian Observatory for Human Rights says ISIS and the al Qaeda-affiliated Al-Nusra Front took control of 90% of the camp in southern Damascus. Calling the lives of Yarmouk refugees "profoundly threatened" on Sunday, the United Nations Relief and Works Agency issued a statement urging humanitarian aid access. "Never has the hour been more desperate in the Palestine refugee camp of Yarmouk," the statement said. The UNRWA estimates 18,000 civilians remain trapped in the camp that has been engulfed in fighting between the government and rebel forces since December 2012. Syria's state-run SANA news agency reports up to 2,000 people have fled in the past two days as food, water and medical supplies remain scarce. "All people are trying to leave the camp," says Syrian activist Abu Mohammed in Damascus who used to live in Yarmouk. "There is no electricity," says Mohammed. "ISIS controls the hospital so injured people have nowhere to go." The Syrian Observatory for Human Rights reports barrel bombs were dropped on the camp Sunday as clashes continued. The Palestine Liberation Organization called on international bodies to assist in the evacuation of people from the camp. "Reports of kidnappings, beheadings and mass killings are coming out from Al- Yarmouk, which is under a brutal campaign of murder and occupation," Palestine Liberation Organization Executive Committee Member Dr. Saeb Erekat said Saturday. Yarmouk, the largest Palestinian refugee camp in Syria, was formed in 1957 to accommodate people fleeing the Arab-Israeli conflict. "The levels of humanity that we have seen have now descended into further levels of inhumanity," said Chris Gunness, spokesman for the UNRWA. Yarmouk, he added, "was always a place where human rights meant very little. We are seeing it descend further." CNN's Samira Said contributed to this report .

Article 3 --- Score: 0.72

(CNN)They took Yarmouk by storm, a sea of masked men flooding into the streets of one the world's most beleaguered places. Besieged and bombed by Syrian forces for more than two years, the desperate residents of this Palestinian refugee camp near Damascus awoke in early April to a new, even more terrifying reality -- ISIS militants seizing Yarmouk after defeating several militia groups operating in the area. "They slaughtered them in the streets," one Yarmouk resident, who asked not to be named, told CNN. "They (caught) three people and killed them in the street, in front of people. The Islamic State is now in control of almost all the camp." An estimated 18,000 refugees are now trapped inside Yarmouk, stuck between ISIS and Syrian regime forces in "the deepest circle of hell," in the words of U.N. Secretary-General Ban Ki-moon. Yarmouk, the largest Palestinian refugee camp in Syria, was formed in 1957 to accommodate people fleeing the Arab-Israeli conflict. The camp, which sits just 6 miles from central Damascus, has been engulfed in fighting between the Syrian government and armed groups since December 2012. The London-based Syrian Observatory for Human Rights says ISIS and the al Qaeda-affiliated Al-Nusra Front control about 90% of the camp. The organization also claims that the Syrian government has dropped barrel bombs on the camp in an effort to drive out armed groups. Activists and residents in Yarmouk tell CNN that as many as 5,000 people have tried to flee their homes since ISIS stormed the camp, but have no place to go. Hundreds have been injured, but the camp's only functioning hospital was first occupied by ISIS, then targeted last week by regime shelling. As the fighting raged in Yarmouk, the director of the Jafra Foundation -- the only aid group that has been able to get into the camp -- painted a grim portrait of the conditions on the ground since ISIS arrived. "We need medicine and access to treatment and medical facilities," Wesam Sabaneh told CNN. "The last hospital in Yarmouk camp was bombed yesterday, so there's really nothing functioning." Opinion: Save the 'miracle babies' Even delivering clean water in Yarmouk can be a deadly task. Majed Alomari, the Jafra Foundation's water coordinator, was killed a few days ago -- gunned down in an ISIS firefight with rival rebel groups. The head of the Palestinian League for Human Rights in Syria (PLHR), who fled the camp and Syria in October 2012, said the people of Yarmouk were in dire need of help.

Figure D.3: Example highlights for BERTSum.



Summary

The United Nations Relief and Works Agency chief will visit Yarmouk camp Saturday . Militant groups are currently in control of the camp . Yarmouk has been engulfed in fighting since December 2012 .

Article 1 --- Score: 0.75

(CNN) The commissioner - general of the United Nations Relief and Works Agency will make an emergency visit to the Yarmouk Palestinian refugee camp in Syria on Saturday , a spokesman says . Commissioner - General Pierre Krähenbühl will assess the humanitarian situation in the camp and speak with individuals about ways to relieve the suffering of the people who remain there . `` The visit is prompted by UNRWA 's deepening concern for the safety and protection of 18,000 Palestinians and Syrian civilians , including 3,500 children , " agency spokesman Christopher Gunness told CNN 's Paula Newton . `` Yarmouk remains under the control of armed groups , and civilian life continues to be threatened by the effects of the conflict . " Krähenbühl will meet with senior Syrian officials , U.N. and relief agency staff members , and displaced people from the camp itself . The Yarmouk refugee camp , which sits just 6 miles from central Damascus , has been engulfed in fighting between the Syrian government and armed groups since December 2012 . The London - based Syrian Observatory for Human Rights says the militant group ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90 % of the camp . The organization also claims that the Syrian government has dropped barrel bombs on the camp as recently as Sunday in an effort to drive out armed groups . Yarmouk was formed in 1957 to accommodate people displaced by the Arab - Israeli conflict and is the largest Palestinian refugee camp in Syria . The U.N. relief agency estimates that there were 160,000 people in the camp when the conflict began in 2011 between forces loyal to President Bashar al - Assad and opposition fighters . That number has dropped to about 18,000 , according to estimates . Yarmouk has been largely cut off from aid since November 2013 . There have been widespread reports of malnutrition and shortages of medical care . `` We will not abandon hope , " Gunness said . `` We will not submit to pessimism , because to abandon hope would be to abandon the people of Yarmouk . ... We can not abandon the people of Yarmouk , and we will not , hence this mission . "

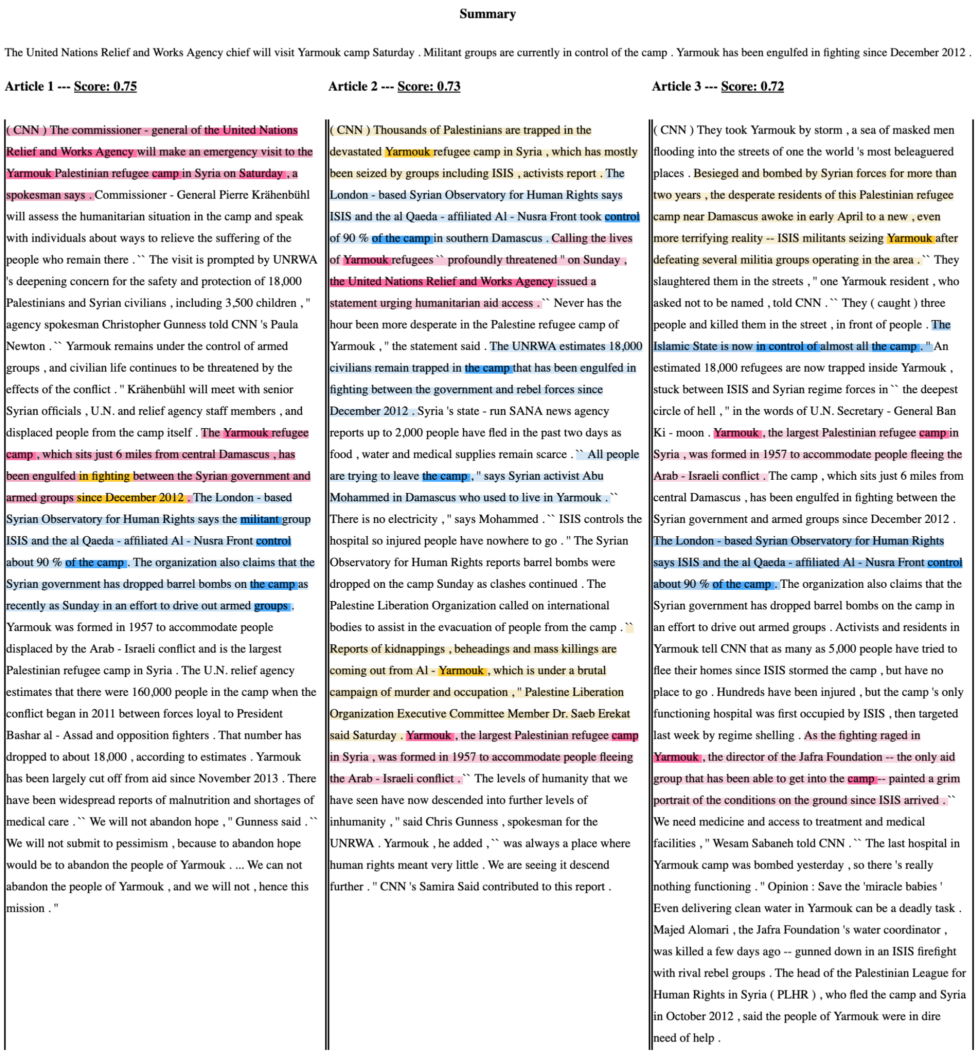
Article 2 --- Score: 0.73

(CNN) Thousands of Palestinians are trapped in the devastated Yarmouk refugee camp in Syria , which has mostly been seized by groups including ISIS , activists report . The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front took control of 90 % of the camp in southern Damascus . Calling the lives of Yarmouk refugees `` profoundly threatened " on Sunday , the United Nations Relief and Works Agency issued a statement urging humanitarian aid access . `` Never has the hour been more desperate in the Palestine refugee camp of Yarmouk , " the statement said . The UNRWA estimates 18,000 civilians remain trapped in the camp that has been engulfed in fighting between the government and rebel forces since December 2012 . Syria 's state - run SANA news agency reports up to 2,000 people have fled in the past two days as food , water and medical supplies remain scarce . `` All people are trying to leave the camp , " says Syrian activist Abu Mohammed in Damascus who used to live in Yarmouk . `` There is no electricity , " says Mohammed . `` ISIS controls the hospital so injured people have nowhere to go . " The Syrian Observatory for Human Rights reports barrel bombs were dropped on the camp Sunday as clashes continued . The Palestine Liberation Organization called on international bodies to assist in the evacuation of people from the camp . `` Reports of kidnappings , beheadings and mass killings are coming out from Al - Yarmouk , which is under a brutal campaign of murder and occupation , " Palestine Liberation Organization Executive Committee Member Dr. Saeb Erekat said Saturday . Yarmouk , the largest Palestinian refugee camp in Syria , was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict . `` The levels of humanity that we have seen have now descended into further levels of inhumanity , " said Chris Gunness , spokesman for the UNRWA . Yarmouk , he added , `` was always a place where human rights meant very little . We are seeing it descend further . " CNN 's Samira Said contributed to this report .

Article 3 --- Score: 0.72

(CNN) They took Yarmouk by storm , a sea of masked men flooding into the streets of one the world 's most beleaguered places . Besieged and bombed by Syrian forces for more than two years , the desperate residents of this Palestinian refugee camp near Damascus awoke in early April to a new , even more terrifying reality -- ISIS militants seizing Yarmouk after defeating several militia groups operating in the area . `` They slaughtered them in the streets , " one Yarmouk resident , who asked not to be named , told CNN . `` They (caught) three people and killed them in the street , in front of people . The Islamic State is now in control of almost all the camp . " An estimated 18,000 refugees are now trapped inside Yarmouk , stuck between ISIS and Syrian regime forces in `` the deepest circle of hell , " in the words of U.N. Secretary - General Ban Ki - moon . Yarmouk , the largest Palestinian refugee camp in Syria , was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict . The camp , which sits just 6 miles from central Damascus , has been engulfed in fighting between the Syrian government and armed groups since December 2012 . The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90 % of the camp . The organization also claims that the Syrian government has dropped barrel bombs on the camp in an effort to drive out armed groups . Activists and residents in Yarmouk tell CNN that as many as 5,000 people have tried to flee their homes since ISIS stormed the camp , but have no place to go . Hundreds have been injured , but the camp 's only functioning hospital was first occupied by ISIS , then targeted last week by regime shelling . As the fighting raged in Yarmouk , the director of the Jafra Foundation -- the only aid group that has been able to get into the camp -- painted a grim portrait of the conditions on the ground since ISIS arrived . `` We need medicine and access to treatment and medical facilities , " Wesam Sabaneh told CNN . `` The last hospital in Yarmouk camp was bombed yesterday , so there 's really nothing functioning . " Opinion : Save the 'miracle babies ' Even delivering clean water in Yarmouk can be a deadly task . Majed Alomari , the Jafra Foundation 's water coordinator , was killed a few days ago -- gunned down in an ISIS firefight with rival rebel groups . The head of the Palestinian League for Human Rights in Syria (PLHR) , who fled the camp and Syria in October 2012 , said the people of Yarmouk were in dire need of help .

Figure D.4: Example highlights for Co-occurrence method.



Summary

The United Nations Relief and Works Agency chief will visit Yarmouk camp Saturday . Militant groups are currently in control of the camp . Yarmouk has been engulfed in fighting since December 2012 .

Article 1 --- Score: 0.75

(CNN) The commissioner - general of the United Nations Relief and Works Agency will make an emergency visit to the Yarmouk Palestinian refugee camp in Syria on Saturday , a spokesman says . Commissioner - General Pierre Krähenbühl will assess the humanitarian situation in the camp and speak with individuals about ways to relieve the suffering of the people who remain there . `` The visit is prompted by UNRWA 's deepening concern for the safety and protection of 18,000 Palestinians and Syrian civilians , including 3,500 children , " agency spokesman Christopher Gunness told CNN 's Paula Newton . `` Yarmouk remains under the control of armed groups , and civilian life continues to be threatened by the effects of the conflict . " Krähenbühl will meet with senior Syrian officials , U.N. and relief agency staff members , and displaced people from the camp itself . The Yarmouk refugee camp , which sits just 6 miles from central Damascus , has been engulfed in fighting between the Syrian government and armed groups since December 2012 . The London - based Syrian Observatory for Human Rights says the militant group ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90 % of the camp . The organization also claims that the Syrian government has dropped barrel bombs on the camp as recently as Sunday in an effort to drive out armed groups . Yarmouk was formed in 1957 to accommodate people displaced by the Arab - Israeli conflict and is the largest Palestinian refugee camp in Syria . The U.N. relief agency estimates that there were 160,000 people in the camp when the conflict began in 2011 between forces loyal to President Bashar al - Assad and opposition fighters . That number has dropped to about 18,000 , according to estimates . Yarmouk has been largely cut off from aid since November 2013 . There have been widespread reports of malnutrition and shortages of medical care . `` We will not abandon hope , " Gunness said . `` We will not submit to pessimism , because to abandon hope would be to abandon the people of Yarmouk . ... We can not abandon the people of Yarmouk , and we will not , hence this mission . "

Article 2 --- Score: 0.73

(CNN) Thousands of Palestinians are trapped in the devastated Yarmouk refugee camp in Syria , which has mostly been seized by groups including ISIS , activists report . The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front took control of 90 % of the camp in southern Damascus . Calling the lives of Yarmouk refugees `` profoundly threatened " on Sunday , the United Nations Relief and Works Agency issued a statement urging humanitarian aid access . `` Never has the hour been more desperate in the Palestine refugee camp of Yarmouk , " the statement said . The UNRWA estimates 18,000 civilians remain trapped in the camp that has been engulfed in fighting between the government and rebel forces since December 2012 . Syria 's state - run SANA news agency reports up to 2,000 people have fled in the past two days as food , water and medical supplies remain scarce . `` All people are trying to leave the camp , " says Syrian activist Abu Mohammed in Damascus who used to live in Yarmouk . `` There is no electricity , " says Mohammed . `` ISIS controls the hospital so injured people have nowhere to go . " The Syrian Observatory for Human Rights reports barrel bombs were dropped on the camp Sunday as clashes continued . The Palestine Liberation Organization called on international bodies to assist in the evacuation of people from the camp . `` Reports of kidnappings , beheadings and mass killings are coming out from Al - Yarmouk , which is under a brutal campaign of murder and occupation , " Palestine Liberation Organization Executive Committee Member Dr. Saeb Erekat said Saturday . Yarmouk , the largest Palestinian refugee camp in Syria , was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict . `` The levels of humanity that we have seen have now descended into further levels of inhumanity , " said Chris Gunness , spokesman for the UNRWA . Yarmouk , he added , `` was always a place where human rights meant very little . We are seeing it descend further . " CNN 's Samira Said contributed to this report .

Article 3 --- Score: 0.72

(CNN) They took Yarmouk by storm , a sea of masked men flooding into the streets of one the world 's most beleaguered places . Besieged and bombed by Syrian forces for more than two years , the desperate residents of this Palestinian refugee camp near Damascus awoke in early April to a new , even more terrifying reality -- ISIS militants seizing Yarmouk after defeating several militia groups operating in the area . `` They slaughtered them in the streets , " one Yarmouk resident , who asked not to be named , told CNN . `` They (caught) three people and killed them in the street , in front of people . The Islamic State is now in control of almost all the camp . " An estimated 18,000 refugees are now trapped inside Yarmouk , stuck between ISIS and Syrian regime forces in `` the deepest circle of hell , " in the words of U.N. Secretary - General Ban Ki - moon . Yarmouk , the largest Palestinian refugee camp in Syria , was formed in 1957 to accommodate people fleeing the Arab - Israeli conflict . The camp , which sits just 6 miles from central Damascus , has been engulfed in fighting between the Syrian government and armed groups since December 2012 . The London - based Syrian Observatory for Human Rights says ISIS and the al Qaeda - affiliated Al - Nusra Front control about 90 % of the camp . The organization also claims that the Syrian government has dropped barrel bombs on the camp in an effort to drive out armed groups . Activists and residents in Yarmouk tell CNN that as many as 5,000 people have tried to flee their homes since ISIS stormed the camp , but have no place to go . Hundreds have been injured , but the camp 's only functioning hospital was first occupied by ISIS , then targeted last week by regime shelling . As the fighting raged in Yarmouk , the director of the Jafra Foundation -- the only aid group that has been able to get into the camp -- painted a grim portrait of the conditions on the ground since ISIS arrived . `` We need medicine and access to treatment and medical facilities , " Wesam Sabaneh told CNN . `` The last hospital in Yarmouk camp was bombed yesterday , so there 's really nothing functioning . " Opinion : Save the 'miracle babies ' Even delivering clean water in Yarmouk can be a deadly task . Majed Alomari , the Jafra Foundation 's water coordinator , was killed a few days ago -- gunned down in an ISIS firefight with rival rebel groups . The head of the Palestinian League for Human Rights in Syria (PLHR) , who fled the camp and Syria in October 2012 , said the people of Yarmouk were in dire need of help .

Figure D.5: Example highlights for Semantic method.

D.3 User Study Details

D.3.1 Pilots and Sample Size

Prior to conducting the actual user study, we ran pilot studies on a smaller number of participants. Using the data points collected from these studies, we conducted a Monte Carlo simulation-based power analysis to determine the effective sample size. We determined to recruit 55 participants per condition (so total of 275 = 55 × 5 conditions) for a statistical power over 0.8 with the effect size (Cohen's d) of 0.5 (orange line with circle markers in Figure D.6). This effect size corresponds to 0.1 difference in the mean accuracy between the control and the treatment.

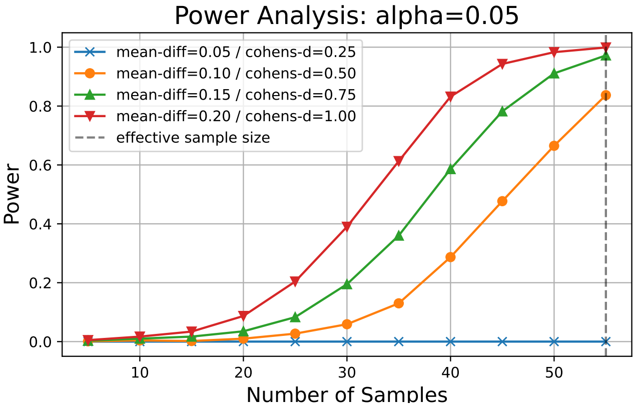


Figure D.6: Power analysis for the effective sample size. We collect 55 samples per group (vertical dotted line) for a statistical power over 0.8 for the effect size (Cohen's d) of 0.5 (orange line with circle markers).

D.3.2 Demographic Background

In Figure D.7, we provide demographic background of the participants (age, ethnicity, student status, employment status) recruited for the study. 275 participants were recruited from a balanced pool of adult males and females located in the U.S. with minimum approval ratings of 90% using Prolific (`www.prolific.co`).

D.3.3 Tutorial

We provide the participants with a set of instructions laying out what the highlights indicate and how one might use them for the task. The instruction is followed by two sample questions on which the participants could take unlimited time to get an understanding of what the questions look like. For the sample questions, the participants were provided the correct answers and the justification behind them as feedback.

D.3.4 Payments

Base payment per participants was \$3.15, determined based on the minimum hourly payment set by the platform and the median completion time of all participants, resulting in an average reward of \$12.07 per hour. To encourage quicker and more accurate responses, we designed bonus payments so that each participant could earn additional \$ (base payment for the question × multiplier) for each correctly

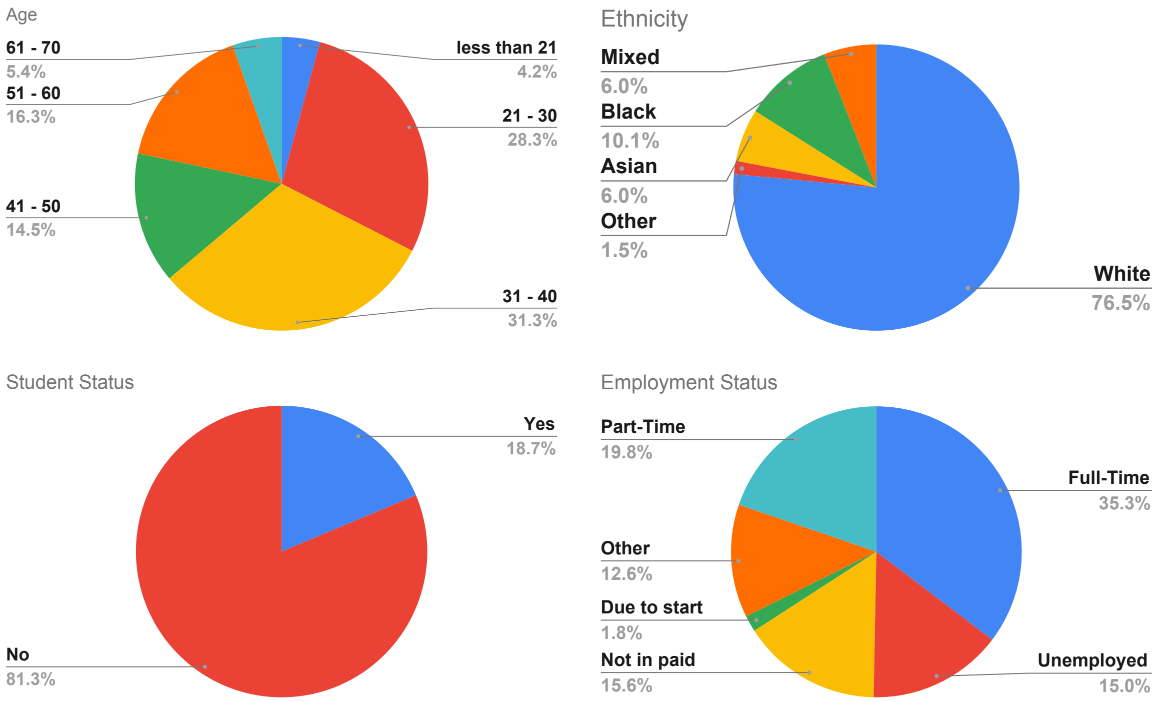


Figure D.7: Demographic background of the participants (age, ethnicity, student status, and employment status).

answered questions, where the multiplier is determined by the response time on the question (Table D.1). One could ideally earn up to ×1.5 the base payment by answering all questions correctly, all within 30 seconds. All payments (base and bonus) were processed after the data collection was complete, accounting for invalid responses.

Response Time (seconds)	< 30	< 60	< 90	< 120	> 120
Multiplier	x0.5	x0.4	x0.3	x0.2	x0.0

Table D.1: Reward multiplier based on response time for correct answers. Incorrect answers have the multiplier of zero.