Towards Usable Machine Learning

by

Alexandra Zytek

B.S., Rensselaer Polytechnic Institute (2018)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2021

© Massachusetts Institute of Technology 2021. All rights reserved.

Author .....
Department of Electrical Engineering and Computer Science
January 27, 2021

Certified by .....
Kalyan Veeramachaneni
Principal Research Scientist
Laboratory for Information and Decision Systems
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Towards Usable Machine Learning

by

Alexandra Zytek

Submitted to the Department of Electrical Engineering and Computer Science
on January 27, 2021, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Machine learning (ML) is being applied to a diverse and ever-growing set of domains. In many cases, domain experts — who often have no expertise in ML or data science — are asked to use ML predictions to make high-stakes decisions. Multiple ML usability challenges can appear as result, such as lack of user trust in the model, inability to reconcile human-ML disagreement, and ethical concerns about oversimplification of complex problems to a single algorithm output. In this thesis, we investigate the ML usability challenges present in non-technical, high-stakes domains, through a case study in the domain of child welfare screening. This study was conducted through a series of collaborations with child welfare screeners, which included field observations, interviews, and a formal user study. Through these collaborations, we identified four key ML usability challenges, and honed in on one promising ML augmentation tool to address them (local factor contributions). This thesis also includes list of design considerations to be taken into account when developing future augmentation tools for child welfare screeners and similar domain experts. Finally, we address the remaining challenges facing the ML community when making ML models more usable in diverse domains.

Thesis Supervisor: Kalyan Veeramachaneni
Title: Principal Research Scientist
Laboratory for Information and Decision Systems

Acknowledgments

I would like to express my deepest gratitude to my advisor, Kalyan Veeramachaneni, for his vital mentorship, support, and guidance throughout this research process. I am very grateful to Dongyu Liu for his endless support during this research. I would also like to thank the rest of the Data to AI team for discussions and guidance. I thank Michaela Henry for her support in collaborations and valuable insights.

I would like to extend special thanks to Rhema Vaithianathan, Diana Benavides Prado, Megh Mayur, Athena Ning, and Larissa Lorimer from the Centre for Social Data Analytics for their endless support in the child welfare case study. I thank Marie-Pascale Grimon and Chris Mills for insightful conversations on ML for child welfare screening. I thank the child welfare experts at Larimer County Department of Human Services for their invaluable insights and feedback on our work.

I thank Iulia Ionescu, Sergiu Ojoc, Ionut Radu, and Ionut Margarint for their work on developing the Sibyl application. I thank Arash Akhgari for his work on diagrams and visualizations. I thank Cara Giaimo and Max Suechting for their feedback. I thank Joshua Robinson for insightful conversations on ML interpretability. I thank Asher Norland for his support in systems development for my projects. I also thank Alice Zytek and Roman Zytek for always helping me extend the reach of my research.

I would like to acknowledge the funding provided to me by two National Science Foundation Grants, "Building a Scalable Infrastructure for Data-Driven Discovery and Innovation in Education" (#1443068) and "Data Science Foundry: A Collaborative Platform for Computational Social Science" (#1761812).

Finally, I would like to offer special thanks to my wonderful family and friends, at MIT and elsewhere, for their endless support.

Contents

1 Introduction **13**

2 Challenges Facing Usability **17**

3 Study Context: Child Welfare **21**

3.1 ML for Child Welfare Screening 22

3.2 Study Participants 23

3.3 Research Questions 23

4 Related Work **25**

4.1 Human-Centered Machine Learning 25

4.2 ML Augmentation Tools 26

4.2.1 Model-Focused Tools 26

4.2.2 Tools for Data-Heavy Domains 27

4.2.3 Tools for Non-Technical Domains 28

5 Understanding Context and ML Usability Challenges **31**

5.1 Understanding Existing Workflow 31

5.2 Identifying ML Usability Challenges 33

5.3 Interviewing Screeners 34

5.4 Summary of Findings 35

6 Sibyl: Design and Feedback **37**

6.1 Sibyl Interfaces and Mock-ups 37

6.1.1 Case-Specific Details: Factor Contributions 39
6.1.2 Sandbox: Investigating "What Ifs" 41
6.1.3 Similar Cases: Investigating Past Cases 41
6.1.4 Global Factor Importances: Understanding the Model 44
6.1.5 Factor Distributions: Understanding Past Predictions 44
6.2 Iterating on Mockup Feedback 48
6.3 Summary of Findings 49

7 Evaluating Tool Usage **51**
7.1 Study Procedure 51
7.2 Study Results 52
7.2.1 Helpful Interfaces 52
7.2.2 Reliance on Sibyl 53
7.2.3 Impact on Trust 53
7.2.4 Information Presentation 54
7.3 Summary of Findings 56

8 Discussion **57**
8.1 The Importance of Interpretable Factors 57
8.2 Accuracy versus Fidelity 60
8.3 Explaining Explanations 61
8.4 Non-Applicable Usability Challenges 62
8.5 Cognitive Biases 63
8.6 Evaluating Explanations 64
8.7 Systems for Explanations 66

9 Conclusion **69**

A Mitigating Tool Definitions **75**

B User Study Questions Asked **77**

List of Figures

2-1 Collaboration Types 20

5-1 The general child welfare screening process 32

6-1 The SIBYL case-specific details interface 40
6-2 The SIBYL sandbox interface 42
6-3 The SIBYL similar cases interface 43
6-4 Factor Distribution visualizations 44
6-5 The SIBYL global factor importance interface 46
6-6 The SIBYL factor distributions page 47
6-7 The side-by-side view of the Details interface 50

7-1 User study procedure. 52

8-1 A demonstration of the inconsistency of interpretations on a simple synthesized example. 65
8-2 The explainable ML workflow 66

List of Tables

2.1 List of possible ML usability challenges 18
2.2 Context factors for domains that may influence ML usability 19

6.1 The proposed SIBYL interfaces, and the challenges they were theorized to address. 38

7.1 Summary of reasons expert participants listed having more trust in the model 54
7.2 Summary of reasons expert participants listed having less trust in the model 54

B.1 Questions asked in the formal user study. 77

Chapter 1

Introduction

We define a machine learning (ML) model as *usable* if its outputs can be effectively used by or have a positive influence on human decision-making. To investigate this concept further, we must first introduce several related concepts.

A *decision-making workflow* is the process of analyzing data and using existing information to select from a set of actions or decisions [11]. The world is filled with such decision-making workflows across most domains, ranging from deciding which medical procedures to perform on a sick patient or determining which candidates to admit to a PhD program, to deciding which highway exit to take or what to eat for dinner. *A decision-maker* could be a human or team of humans, using a combination of their own knowledge and external tools to aid the process. The decision-maker could be a computer algorithm, autonomously making decisions and taking actions [11]. Or, the decision-maker could be a combination of these, with human and computer each taking over the decision-making depending on the circumstance.

Thanks to innovations in ML, computers now play a role in an increasing number of decision-making workflows previously performed by humans alone, promising both speed and precision. In the case of some workflows, such as self-driving cars, the goal may be to entirely replace the human decision-maker with an ML-based approach. In other workflows, humans provide essential insights that cannot currently be replaced by existing ML models. In such cases, the goal is to improve decision-making outcomes by using ML output to augment human decision-making. Finally, ML models may

make new decision-making workflows possible — ones that are based around new data sources and analysis made possible by ML models. Currently, there are a wide variety of decision-making workflows already in use that may be improved by ML, so we ground our focus on these existing workflows.

Two research themes emerge when trying to improve decision-making workflows with ML models.

The first theme revolves around the development and selection of the ML models themselves — finding and selecting data, training ML models [24], debugging to improve model performance [2] [35], selecting from model options [23], evaluating a model's ability to generalize and its expected performance in the real world, and validating the effects of model introduction. This theme focuses on an audience of ML experts and data scientists. It is based around mostly model-focused questions such as "how can we improve the performance of this model", "what combination of models would be most useful", and "what is the expected benefit of this model in the real world?"

The second theme revolves around the usage of the model itself — taking the information provided by a model and translating it into a real-world action. The audience of this theme is more diverse, typically consisting of both domain experts who may lack ML experience and laypeople engaging in everyday life. The questions relevant to this audience are often more domain- or decision-focused, such as "why should I trust the model in this case?" and "what about this situation caused the model to make this prediction?"

Both of these themes can benefit from a similar set of tools. For example, ML explainability and interpretability can reveal flaws in the model logic before deployment (theme 1) and help calibrate human trust when making decisions using ML (theme 2). Data visualization can shed light on which features may be helpful in model training (theme 1) and highlight when a model may be less helpful due to a case being out of the training distribution (theme 2).

Making ML models usable requires investigating a decision-making workflow through both themes of research. For a model to be usable, it must perform sufficiently well

with available data, and be useful to any humans involved in the decision-making process. This thesis will focus on the second theme, answering the question: **given an existing decision-making workflow and relevant ML model, what still needs to be done to make the model usable?**

To approach this question, we began by identifying the key ML usability challenges that may prevent an existing model from being effectively used in decision-making workflows, as well as the domain factors that may determine which usability challenges are most relevant. In order to investigate the problem of ML usability further, we then conducted a case study and followed a user-centered iterative design process [21] in the domain of child welfare screening.

The rest of this thesis is structured as follows. In Chapter 2, we further discuss the challenges involved in making ML models usable. In Chapter 3, we provide background on the domain of child welfare screening and our case study participants. In Chapter 4, we provide related work regarding ML usability and augmentation tools. In Chapter 5, we discuss the process to better understand the child welfare screening workflow and usability challenges. In Chapter 6, we discuss our initial augmentation tool design and screener feedback. In Chapter 7, we describe our formal user study and results. In Chapter 8, we discuss the key, generalizable findings of this work, as well as the remaining challenges for improving ML usability. Finally, we conclude in Chapter 9.

Chapter 2

Challenges Facing Usability

Many ML usability challenges stem from the fact that ML models often output only a single number or classification. This lack of further information can result in many challenges when trying to use the model.

We investigated the need for additional auxiliary information alongside ML predictions through an informal literature review of 55 papers on ML applications and explainability. We found ML usability challenges can arise in multiple places in the decision-making workflow — when deciding if and how the model should be used, when actively using the model to make decisions, and when retrospectively reviewing a model's decisions and performance. As explained in the introduction, in this thesis we focus on the usability challenges relevant to the second of these steps, active decision making.

Table 2.1 summarizes the set of usability challenges we codified that are relevant when actively using a model for decision-making. Some challenges stem from not understanding where a model's predictions come from, making it difficult for human decision-makers to trust the model (TR), and to handle any disagreements between their opinions and the model's output (RD). Others are caused by a lack of information about the real effects of a decision. A lone model prediction often does not explicitly indicate the expected results of a decision (UC), suggest accountability (AC), or provide ethical assurances (EC). Finally, challenges may arise when the output of the model is not a direct suggestion of a decision, but rather auxiliary

Usability Challenge	Code	Mitigating Tools
Lack of **TR**ust	TR	Global explanations, local explanations, performance metrics, historical predictions and results
Difficulty **R**econciling human-ML **D**isagreements	RD	Local explanations
Unclear **C**onsequences of actions	UC	Cost-benefit analysis, historical predictions and results
Lack of **AC**countability	AC	Local explanations, performance metrics
Ethical **C**oncerns (ex. possible bias, concerns about oversimplification)	EC	Global explanations, local explanations, ML fairness metrics, historical predictions and results
Confusing or unclear prediction **T**arget (ie. the measure predicted by the model has an unclear meaning or significance)	CT	Cost-benefit analysis, further analysis of prediction target impact
Unhelpful prediction **T**arget (ie. the measure predicted by the model is not relevant to the required decision)	UT	Retrain model with new prediction target

Table 2.1: List of challenges that could negatively impact the usability of an ML model (left column), their corresponding codes that we will use throughout the paper (middle column), and possible tools that may help address the challenges (right column). For definitions and examples of the mitigating tools, see Appendix A.

information. In this case, the output may be confusing (CT) or entirely irrelevant (UT).

The machine learning, data science, and data visualization communities have offered a multitude of algorithms and tools to augment ML predictions and address these usability challenges — we refer to these as *ML augmentation tools*. These tools, when chosen carefully for the domain, have the ability to greatly improve the usability of ML models for decision making. Examples of such tools include data visualizations, global and local explanations [5], cost-benefit analysis, performance metrics, and information about historic usage and results of the ML model. Unfortunately, research aimed at augmenting ML predictions often focuses on an audience of ML/data experts [38] [30] [13] or domain experts in fairly data-heavy fields such as medicine [20] [16].

Context Factors	Categorizations	Example domain
Degree of Automation	Fully autonomous, humans not involved in decision making	Level 5 self-driving car [22]
	Machine makes decisions, humans monitor	Level 4 self-driving car [22]
	Machine suggests decisions, humans make decisions	Content violation flagging
	Machine provides auxiliary information, humans make decisions	Child welfare screening
Decision time	Immediate	Level 5 self-driving car [22]
	Seconds	Aircraft emergency response
	Minutes	Child welfare screening
	Hours	Non-emergency medical procedures
Technical expertise of humans	**Little to none**	Child welfare screening
	Experience with data science	Finance
	Machine learning/Data science expert	ML model training and debugging
Domain expertise of humans	Little to none	Autonomous aircraft (to passengers)
	Basic understanding/Intuition	Crowdsourcing
	Domain expert	Child welfare screening
Associated Risk	Low	Camera roll image sorting
	Medium	Mail sorting
	High	Emergency medical procedures

Table 2.2: Domain context factors that may influence the usability of an ML model. The context factors relevant to child welfare screening are in bold. By *technical expertise*, we refer to the ML or data science expertise of the end-user. This is not meant to be a complete list — there are many other factors that could also be relevant.

For example, Zhang et. al. [38] developed a framework for helping data scientists and ML experts interpret and debug ML models, and Lundberg et. al. [20] developed an interface for helping anaesthesiologists prevent hypoxaemia during surgery through detailed data visualization. In contrast, many fields are more qualitative in nature, with decisions following discussion more than data crunching. In this thesis, we focus on these more qualitative fields, and the usability challenges they face.

Determining which usability challenges exist, the best tools to address them, and the necessary design choices for these tools depends highly on specific aspects of the domain and the decision-makers involved. Based on our literature review, Table 2.2 lists some examples of factors that must be considered when working to make an ML

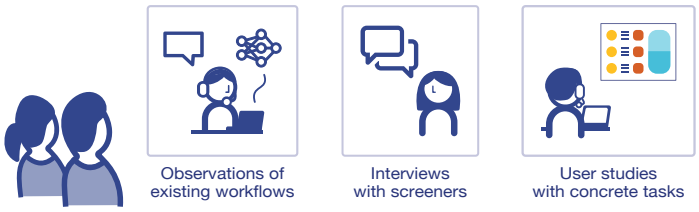

Figure 2-1: The three types of collaborations with child welfare screeners used in our investigation. In the first type, the ML practitioners (the authors), represented by the left-most figures, observed the end-users engaging in their existing decision-making workflow, to understand the constraints and needs of the systems. In the second type, the ML practitioners interviewed with the end-users, to get further insights on what improvements could be made to the system. In the third type, the end-users participated in a formal user study with proposed ML augmentation tools, to get concrete feedback.

model more usable in a particular context.

To investigate the problem of finding and mitigating usability challenges in more qualitative fields, we selected the domain of child welfare screening. In terms of the relevant context factors, child welfare screeners are domain experts without ML/data science expertise, making decisions using an ML model as an auxiliary tool, with about a few minutes per decision, in a high-risk field.

Addressing usability challenges is a non-trivial task that requires collaboration with end-users. For this thesis, we engaged in three forms of collaboration: *observations* to understand their existing workflow and its possible usability challenges, *interviews* to gain additional insight into the desires of end-users, and *user studies* with possible ML augmentation techniques to get concrete feedback on design. Figure 2-1 shows this process.

Chapter 3

Study Context: Child Welfare

Child abuse is an active issue affecting the health and well-being of communities. The Centers for Disease Control and Prevention (CDC) estimates at least 1 in 7 children have experienced child abuse and/or neglect in the past year [3]. Child abuse victims can suffer physical injuries and emotional problems, and may experience trauma resulting in long-term mental health problems [3]. More than one-third of American children are investigated as potential victims of abuse or neglect by age 18 [15]. Still, in 2018, there were 1,770 reported fatalities resulting from child abuse and neglect [3].

Both false negatives (real abuse cases that are screened out) and false positives (cases with no abuse that are screened in) can have heavy consequences. False negatives lead to prolonged child suffering and, in extreme cases, child fatality. False positives can lead to long-term emotional distress for parents, children, and other family and community members, as well as damaged financial, career and social prospects for parents and other caretakers [26].

In the U.S., regional Child Protective Services (CPS) agencies are tasked with handling child abuse and neglect referrals from concerned members of the community, including mandated reporters such as teachers, who are required by law to report any suspicion of abuse or neglect. These referrals are examined by child welfare specialists ("call screeners"), who decide whether to screen in or screen out each case. A screened-in case will be investigated further, while a screened-out case will be recorded but not

investigated.

In 2018, CPS agencies in the United States received 4.3 million referrals from concerned parties about potential child abuse [4]. 56% of these referrals were screened in and investigated, but only 16.8% of the screened in cases were found to involve abuse or neglect [4].

One important motivation for computerized assistance in child welfare call screening is repeated cases of missed abuse. Fatal child abuse cases in which children were referred several times but were never screened in are tragic, and although such cases are rare, they are avoidable [10]. An ML solution can quickly scan for red flags, such as repeated referrals, that busy human call screeners may miss in the overload of data.

3.1 ML for Child Welfare Screening

In recent years, predictive risk modelling (PRM) has been deployed in child welfare contexts in multiple counties, with the goal of enabling more efficient and consistent decision-making and improving the overall health and safety of county residents [32]. One example of such a model was deployed in Allegheny Country, PA by Vaithianathan et. al. in 2016 [32].

Currently, PRM is being introduced to our collaborating county in Colorado by Vaithianathan et. al. [31], through a LASSO regression model trained on 461 features, which include information such as the child and parents' age, past referrals and their outcome, and past court involvements. The model predicts the likelihood of removal from home in the next two years, translated to a 1 through 20 risk score where the higher the score, the higher the risk [32]. This thesis focuses on the usage of this model.

3.2 Study Participants

For this thesis, we collaborated over the course of a year with a pool of 19 social workers and supervisors working for the child welfare department in a collaborating county in Colorado. All participants regularly act as screeners in the county's child welfare screening decision making process.

Our collaborations began in December 2019, with two days of in-person field observations (Section 5.1). Following this, we observed a simulated case review session via video conferencing (Section 5.2), and conducted several interviews also via video conferencing (Sections 5.3 and 6.2). Finally, the collaborating screeners participated in a user study digitally (Chapter 7).

3.3 Research Questions

This thesis seeks to answer the following research questions:

RQ1 What ML model usability challenges exist in the domain of child welfare screening, and other similar domains (with high risk, high domain expertise, and low technical expertise)?

RQ2 What tools can be helpful in mitigating these ML model usability challenges?

RQ3 What design choices must be made when building these tools to optimize them for use by child welfare screeners and other experts in similar domains?

Importantly, our goal in this research was to aid user decision-making by boosting the usefulness of the ML model through augmentation, not to modify the decision making itself. Wang et. al. [33] describe a possible benefit of explanations as reducing the negative impact of cognitive biases on decision-making. While we believe this is an important use, aiming for this goal would require delving into the shortcomings of the existing decision-making process itself, which requires a very different type of expertise. For the sake of this thesis, we will assume that the currently approved decision-making process is effective, and the goal is to provide the right information

to aid this process. We do, however, investigate possible cognitive biases that may be *introduced* by augmentation, described in section 8.5.

Chapter 4

Related Work

4.1 Human-Centered Machine Learning

Past literature has advocated for a *human-centered* perspective to ML [7] — one that considers machines and algorithms as part of collaborative systems alongside humans. This perspective considers how humans use, interact with, adapt to, and evaluate ML applications [7]. A truly human-centered ML approach acts end-to-end, beginning with human-in-the-loop training systems and ending with evaluation systems based on the metrics that end users are most interested in [7]. In this thesis, we take a deeper look at one step of this extensive pipeline: the use of ML model predictions by humans for real-world decision making.

A common concern addressed by the literature is the black box nature of most ML models [28]. Humans struggle to use ML predictions because they do not understand where they came from. This concern is addressed through the fields of interpretable or explainable ML. Doshi-Velez and Kim proposed that the need for ML interpretability stems from an "incompleteness in the problem formulation" [5], which prevents the system from being thoroughly evaluated or trusted. This incompleteness can take several forms, including a need for scientific understanding, concerns about safety or ethics, or mismatched objectives between the model output and the human goal [5].

Doshi-Velez and Kim [5] also define three evaluation approaches for ML interpretability. In *application-grounded approaches*, domain experts work with interpre-

tations within a real application. This provides the most realistic quantification of interpretation quality, but may require high time commitments from a potentially small pool of domain experts. In *human-grounded approaches*, researchers develop simpler problems for experimentation using non-expert subjects. Finally, in *functionally-grounded evaluation*, a formal definition of interpretability is used as a proxy to evaluate an interpretation without using human subjects. This taxonomy of evaluation styles can also apply to the wider challenge of ML usability, and this thesis focuses on an example of such an application-grounded approach. As Doshi-Velez and Kim [5] point out, this is the best approach for evaluation if a willing pool of domain expert participants is available. A major focus of our process was to respect the participants' time by taking steps to ensure all time spent with them was optimized.

Wang et. al. [34] developed a human-driven conceptual framework for building explainable AI systems. They found that decision-makers seek explanations to justify unexpected occurrences, monitor for important events, or facilitate learning. They created a taxonomy of AI techniques based on how they support human reasoning and represent information. The authors also discuss how humans reason, using the dual process model. In this model, humans rely on system 1 reasoning, which happens quickly and intuitively, and system 2 reasoning, which is slower and more analytic. Finally, the authors discuss how explainable AI can mitigate cognitive biases. Our work builds on this by finding cognitive biases that can be *caused* by explainable AI, as listed in section 8.5.

4.2 ML Augmentation Tools

In this section, we reference some existing work that aims to improve ML usability, usually through explanations and/or visualizations.

4.2.1 Model-Focused Tools

Some work has been done on developing ML augmentation tools with the main focus of revealing the inner workings of the models themselves. These tools can be used for

debugging, validation, and ML education. A question of this thesis was if and how the needs of a less technical audience, focused on decision-making rather than model development, differed from the needs described in these papers.

Wang et. al. (2019) [36] developed a comprehensive tool to visualize convolutional neural networks, called CNN Explainer. This tool attempts to educate ML learners about CNNs' inner workings. It includes activation heatmaps for individual neurons, as well as visualizations of the model process.

Hohman et. al. (2019) [9] developed a visual analytics system called Gamut to investigate how machine learning practitioners and data scientists interact with machine learning. To develop this tool for use on generalized additive models (GAMs), the authors interviewed technical experts to generate a list of common questions asked about predictions. In total, they identified six question types, which they address using three views. The first view uses line charts to represent a feature's contribution to the model output at different values. The second view shows the feature contributions associated with each individual instance. The third view contains a table of all data inputs and their feature values, with information like their corresponding prediction accuracy and nearest neighbors. Gamut was tested by having 12 data scientists use the tool to investigate models while thinking out loud, followed by an interview. This study was focused on model investigation, rather than decision-making.

4.2.2 Tools for Data-Heavy Domains

In this section, we describe tools developed for the purposes of improving decision making — however, unlike the focus audience of this thesis of child welfare screeners, the domain experts in these papers were from more data-heavy fields.

Lundberg et. al. (2018) [20] developed Prescience, an explanatory ML system focused on preventing hypoxaemia during surgeries. This tool predicts the risk of hypoxaemia in the next five minutes using a gradient-boosting algorithm trained on time series. It includes several time-series visualizations for medical data features to explain the prediction, including SHAP feature contribution explanations.

Kwon et. al. (2019) [16] developed RetainVis, a visualization tool for explaining

recurrent neural networks (RNNs) applied to electronic medical records (EMRs). The tool was developed to help medical practitioners make medical decisions such as diagnosis and prescription. The tool was developed with active feedback from medical practitioners. RetainVis includes five different visualizations for looking into RNNs: 1) an overview of all patients' medical codes, contribution scores, and diagnosis risk, through multiple plot types, 2) a temporal patient summary, 3) a list of patients with information and comparison capabilities, 4) details about an individual patient, and 5) a what-if analysis tool that allows users to edit patient details.

Our work is similar to these tools in that it relies on collaboration with end users to develop a tool that provides additional information alongside an ML prediction. However, these tools were developed for domain experts who tend to actively work with complex medical data, and their visualizations include detailed information about a large number of features. Because the child welfare screeners are not expected to be as familiar with working with big data, we theorized that our visualizations would need to look different and rely less on data-heavy details.

4.2.3 Tools for Non-Technical Domains

Additionally, some work has looked at audiences who do not necessary have experience working with data science or in data-heavy environments.

Lakkaraju et. al. (2019) [17] developed a model agnostic framework called MUSE (Model Understanding through Subspace Explanations), which explains model output in terms of input subsets. This provides global explanations conditioned on certain feature values. The goal of the framework is to provide useful explanations to end users who may not have prior ML experience. The authors used three metrics to evaluate their explanation method: fidelity (how accurately the model explained model behavior), unambiguity (tendency for deterministic and unique explanations), and interpretability (how easy the explanation is to understand). The framework was tested by asking users to perform tasks using both MUSE explanations and baseline explanations (LIME, interpretable decision sets, and Bayesian decision lists).

The work in this thesis similarly looks at domain experts without ML expertise,

but we have chosen to focus on a more specific subset of domains in order to thoroughly highlight their ML usability challenges.

Chapter 5

Understanding Context and ML Usability Challenges

To address **RQ1** (identifying ML usability challenges), we performed a series of field observations and interviews. In this chapter, we discuss the goals and the findings of these steps.

5.1 Understanding Existing Workflow

To better understand the existing child welfare screening workflow, we travelled to our collaborating county in Colorado to observe screeners' decision-making on referrals without using the ML model. This observation revealed:

1. Our collaborating county uses the general procedure for child welfare screening shown in Figure 5-1. In cases of immediate concern, a referral may be screened in immediately after it is received by CPS (this decision is made by a child welfare supervisor). In most cases, however, the decision as to whether to screen-in (investigate further) or screen-out (not investigate further) a referral is made by a team of child welfare experts. It is this team that receives the ML risk score prediction.

2. The team of child welfare experts meet in-person every morning, during which

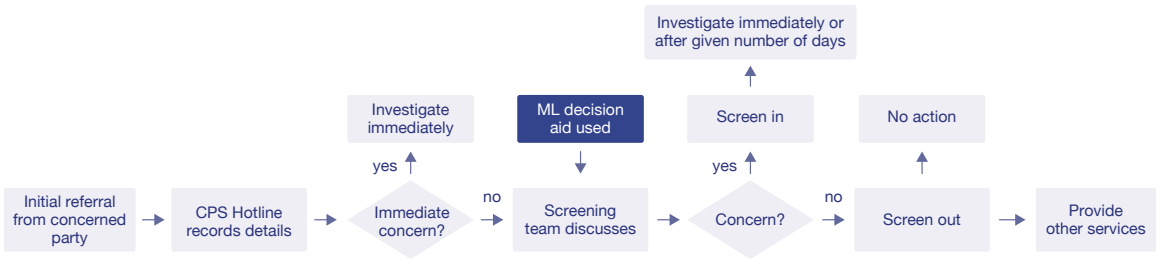

Figure 5-1: The general child welfare screening process used by the CPS department of our collaborating county. The referral is first received by the CPS hotline, and then sent to a child welfare supervisor. In a minority of cases, the supervisor will deem the child or children involved to be in serious and immediate risk of danger, and will screen-in the case for immediate further investigation. In most cases, however, the case will be reviewed by a team of child welfare screeners the next day. This team will be given the ML risk score prediction (dark blue box). If this team decides to screen-in the case, it will be investigated further through home visits, interviews, or other means. Otherwise, the case will be recorded but not investigated unless re-referred. In the case of a screen-out, the screeners may elect to provide the family with additional family services.

time they screen about seven cases. Five to ten minutes are spent on each case by this team. Most of this time is spent going over the details of the case. The screening decision is made after about one to two minutes of discussion.

3. A large portion of these five-to-ten minutes of screening time is dedicated to determining the factors that are associated with higher and lower likelihood of abuse — referred to as risk and protective factors, respectively — involved in a case, and weighing these against each other. The factors considered will vary based on the details of the case, but may include information such as child's age (very young children are more vulnerable), criminal record of adults involved, whether there are any trusted adults active in the child's life, and actions and statements made by adults involved (for example, a mother taking actions to separate an aggressive partner from her children).

5.2 Identifying ML Usability Challenges

To identify the ML usability challenges present in the domain of child welfare screening (**RQ1**), we observed a simulated case review session using the ML model. In this session, our collaborating screeners (split into 3 teams, labelled **T1**, **T2**, and **T3**) were asked to make decisions about historical referrals as they normally would — but with the addition of the risk scores provided by the ML model, which were provided before they made the final screening decision. Each team made decisions on 7 to 9 cases. The participants were then interviewed and asked to reflect upon how they used each score, whether the scores aligned with their expectations, and how the scores impacted their decisions.

Based on the answers to these interview questions, we categorized the concerns into four key usability challenges:

1. **Lack of Trust** **TR** Screeners expressed a lack of trust when making decisions using the ML model alone, evidenced by their tendency to not consider the model prediction at all when it disagreed with their intuition. For example, when asked if the score caused them to reconsider their decision, **T3** responded

   "No, [we were] surprised it is that low."

   In answer to the same question, **T2** responded

   "No - just questioned where the score came from."

2. **Difficulty reconciling human-ML disagreements** **RD** Screeners did not have a clear path forward when they disagreed with the model prediction, sometimes electing to ignore it entirely (see previous item) and sometimes trying to justify it based on how they thought the model worked. For example, **T2** reported in one case that the score made them

   "think a little deeper about why the score is so high [and caused us to] take another look at [the history]"

3. **Unclear prediction target** CT Because the model provides auxiliary information (1-20 risk score based on the likelihood of removal from home in 2 years) rather than a direct decision suggestion, there was some confusion about how to use the model prediction target. For example, when asked how the model affected their decision making process, **T3** responded

   > *"[we did not know] enough of what the score means to know how to accurately use it."*

   T3 also said they

   > *"Wish we knew how we got to the score."*

4. **Concerns about Ethics** EC As expected for such a sensitive domain, users were concerned about the ethics of using the ML model score. There was concern that the model may prevent critical thinking, and prevent children from being treated as individuals with unique circumstances. **T3** commented

   > *"[The model] could be dangerous for people just looking at the number, need to take everything into account. Makes you stop and think and ask yourself are you critically thinking..."*

5.3 Interviewing Screeners

To begin addressing **RQ2** (what tools can be helpful in mitigating usability challenges), we interviewed the 19 screeners about what kind of additional information they would be interested in getting alongside ML predictions. The format was a semi-structured open-floor session, with us (the ML practitioner team) asking questions to the room, and all screeners were encouraged to answer or provide other thoughts at any time. We began by asking whether they thought additional information would be useful and what specific information they would be interested in, and ended by proposing possible augmentation tools and asking if they seemed helpful.

Our findings from this step included:

1. Screeners were confident that they would want to know why the model made the predictions it made.

2. Some screeners believed that understanding how important each factor was to the score prediction would be helpful.

3. Some screeners wanted to know what steps the model takes in making predictions.

4. Some screeners were interested in getting "*what-if*" style explanations that give information about what could be changed about a child to reduce his or her risk. Note that the ML model is not trained on causal relationships, so explanations would not be able to provide such information.

5. Some screeners were interested in seeing similar cases they dealt with in the past. Others thought this would be too much information to digest in such a short period of time.

5.4 Summary of Findings

Here, we summarize the findings from our first set of collaborations (field observations and interviews). We identified four key usability challenge categories: lack of screener trust in the model (TR), difficulty reconciling disagreements between the model risk score and screener intuition (RD), confusion about what exactly the risk score entails (CT), and concerns about the ethics of simplifying a complex case down to a single number (EC). We determined that screeners wanted to know where the risk score predictions came from, and were interested in seeing a few different augmentation tool types including information about which factors contributed to predictions, "*what-if*" style explanations, descriptions about the model's process, and examples of similar past cases.

Chapter 6

Sibyl: Design and Feedback

To address **RQ2** and **RQ3**, and based on the usability challenges identified (section 5.2) and responses to our interview (section 5.3), we engaged in a user-centered iterative design process [21] to develop Sibyl, an ML augmentation tool. We began by designing high-fidelity mock-ups for five different augmentation interfaces, each with a separate purpose and goal. Table 6.1 summarizes the motivation behind each interface in terms of its theorized effect on addressing the usability challenges.

6.1 Sibyl Interfaces and Mock-ups

In this section, we describe the original five Sibyl interface designs, and provide the high-fidelity mock-ups.

A note on terminology: early in the design process, we learned that the word "factor" is more familiar to screeners than "feature" when referring to pieces of information used when making decisions. For the purposes of consistency, we will use the word factor when discussing the Sibyl tool and findings to refer to data inputs used by the model — this is synonymous in this case to model features.

Interface	Challenges Addressed	How does it address the challenge?
Case-Specific Details	TR	Reveals relevancy of considered factors
	RD	Highlights factors that may have been missed or misused
	CT	Translates score to a concrete factor list
	EC	Allows for critical thinking about factors and score
Sandbox	RD	Allows users to test theorized justifications
	EC	Allows for thinking through what-if scenarios
Similar Cases	TR	Provides information on past performance
	EC	Provides a deeper look into the nuance of cases
Global Factor Importance	TR	Reveals how the model generally makes predictions
	CT	Translates the score to a concrete factor list
Factor Distributions	TR	Shows how well the model performed on past cases
	CT	Shows the relationship between the risk score and removals from home

Table 6.1: The proposed SIBYL interfaces (left column), the challenges they were theorized to address (middle column, using the codes from Table 2.1), and the reasons we expected these interfaces to address the given challenges (right column). TR: Lack of trust in the model. RD: Difficulty reconciling disagreements. CT: Confusing prediction target. EC: Ethical concerns.

6.1.1 Case-Specific Details: Factor Contributions

The *Case-Specific Details* interface, shown in Figure 6-1, provides a simple local explanation of where an individual model prediction comes from through factor *contributions*. The table in this interface assigns each factor a contribution, either positive or negative.

For example, a local factor contribution explanation may reveal that the age of a particular child (infant) resulted in the risk score significantly increasing, while the number of past referrals (0) resulted in the risk score decreasing.

The local factor contributions were found using the Shapely Additive Explanations (SHAP) algorithm [19]. We chose to use SHAP because it is a theoretically sound approach that generates consistent and intuitive explanations for an ML model [19].

We realized that a possible source of confusion stems from the phrases *positive contribution* and *negative contribution*. In the ML community, a positive contribution indicates that the model prediction will increase in value. In the screener community, however, an increased risk score is only a negative occurrence. We decided to avoid using the terms "positive" and "negative" in the app or instructions. Instead, the factors are labelled as "risk factors" or "protective factors," which directly mirrors the screeners' language.

Additionally, we decided on a local explanation interface as we theorized it may help with the identified usability challenges: the screeners' lack of trust TR by demonstrating that the model relies in part on similar factors as the human screeners in making decisions; difficulty reconciling disagreements RD by highlighting differences in the human and model's logic; unclear prediction target CT by providing a concrete explanation of the scores' meaning; and concerns about ethics EC by making critical thinking about relevant factors easier.

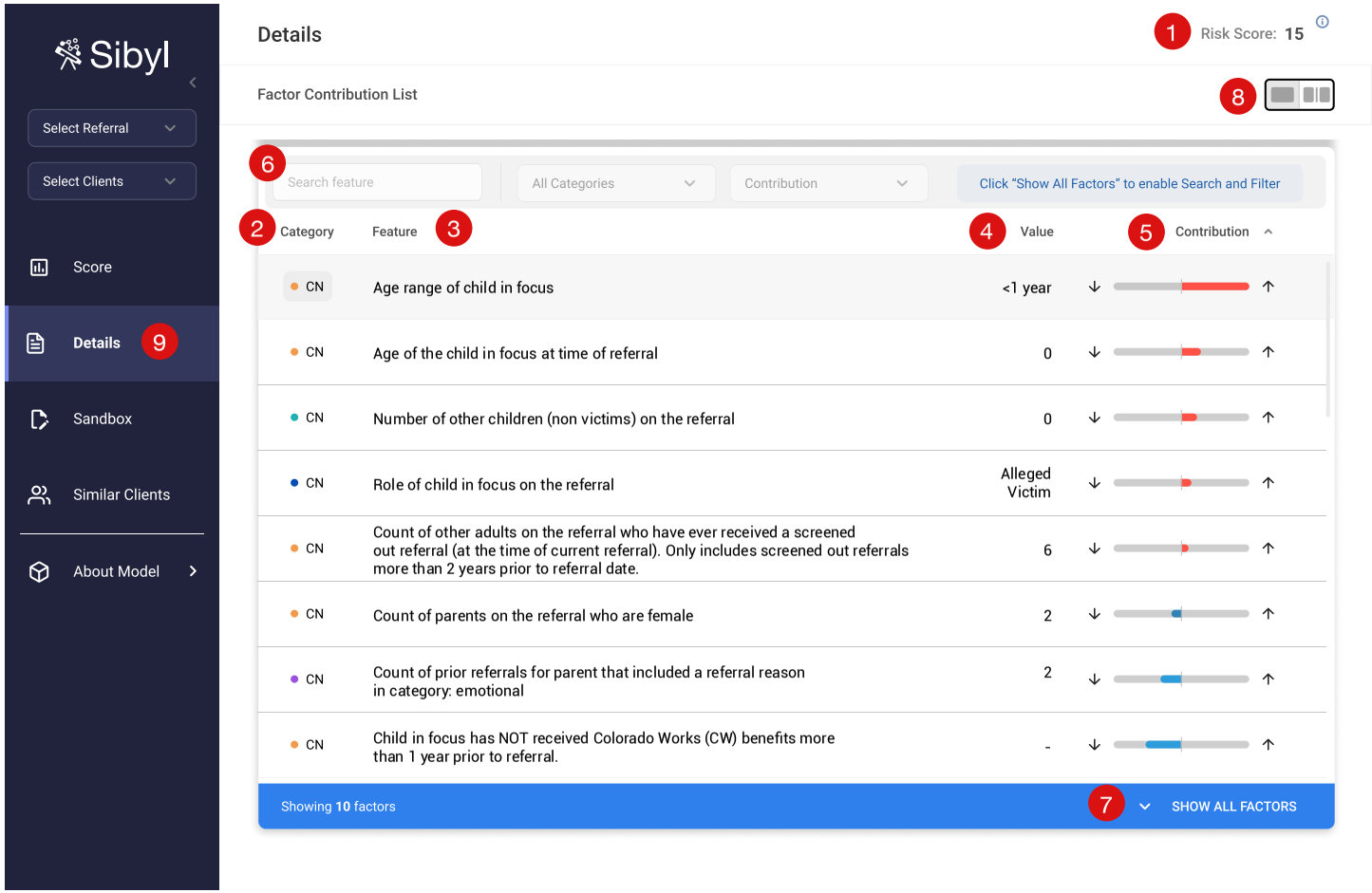

Figure 6-1: The SIBYL "Case-Specific Details" interface. The Case-Specific Details interface shows how factors contribute to predictions made by ML models about a child referred to child welfare services. Labeled elements are as follows: 1) The risk score for the case (1-20). 2) Categories for each factor, such as demographics (DG) or referral history. 3) A short description of each factor. 4) The value of numeric or categorical factors. 5) The contribution of each factor (the table can be sorted in ascending or descending order of contribution). 6) UI components for searching by factor name or filtering by category, enabled when the full factor list is shown. 7) A button for switching between a view that shows only the top 10 most contributing factors and one that shows all factors. 8) A button for switching between a single-table view and a side-by-side view, which splits factors that increase and decrease risk. 9) A sidebar that includes other interfaces, as described in chapter 6

6.1.2 Sandbox: Investigating "What Ifs"

The *Sandbox* interface, shown in Figure 6-2, allows users to experiment with and see how the model prediction would change if factors differed. It has two parts.

The *Experiment with Changes* box allows users to change up to four factor values at a time, to investigate specific "what-if" questions they may have. We decided to limit users to four factors changed at a time to prevent the information provided from shifting too far from the reality of the case at hand, which could cause confusion.

The *Model predictions if each value was changed* box shows the resulting prediction if each Boolean factor value was individually reversed.

This interface was added based on feedback from the interviews described in section 5.3. We theorized it may help with difficulty reconciling disagreements RD by allowing screeners to test their theorized justifications, and with concerns about ethics EC by making more detailed consideration about the model's output easier.

6.1.3 Similar Cases: Investigating Past Cases

The *Similar Cases* interface, shown in Figure 6-3, shows the complete history of child welfare involvement with past cases that had similar factor values. This interface includes a timeline for the current case and each similar case, and highlights events such as referrals to child welfare services, investigations, and removals.

The similar cases are found using a Nearest Neighbors algorithm. For design purposes, the algorithm we used weights all factors equally.

The interface was added as we theorized it may help with screeners' lack of trust TR by demonstrating past performance, and concerns about ethics EC by providing a deeper look into the nuance of individual cases.

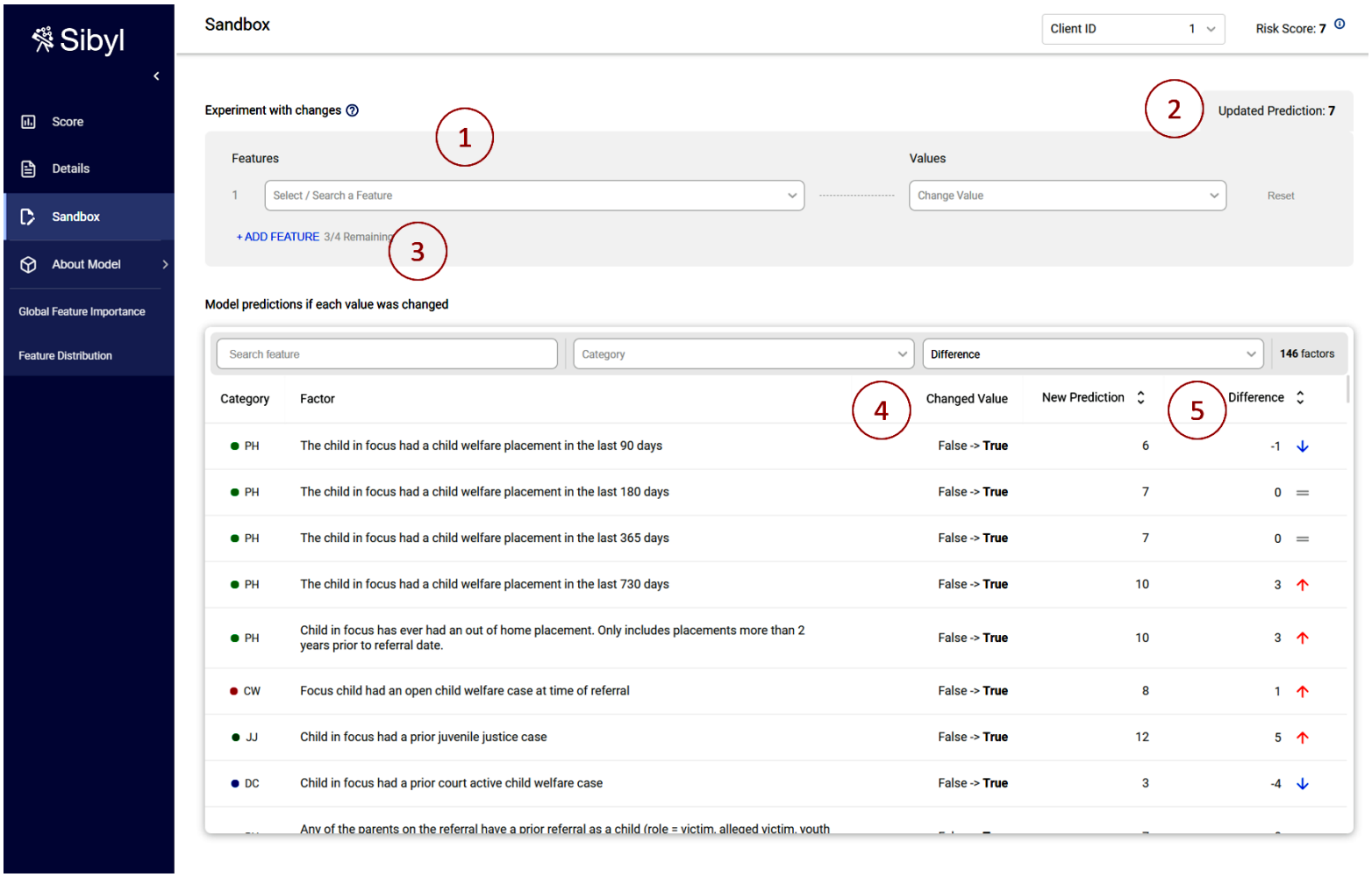

Figure 6-2: The sandbox interface of SIBYL includes the following elements. 1) Drop-down box functionality to temporarily change the values of up to four factors at a time, to investigate the effects on the model. 2) The updated prediction, factoring in the changes selected with component 1. 3) Allows for up to four factors to be added and experimented with at a time. 4) The current score prediction and flipped value of all binary factors in the client's dataset. 5) The resulting change in prediction, measured as difference in score, when each binary value is flipped.

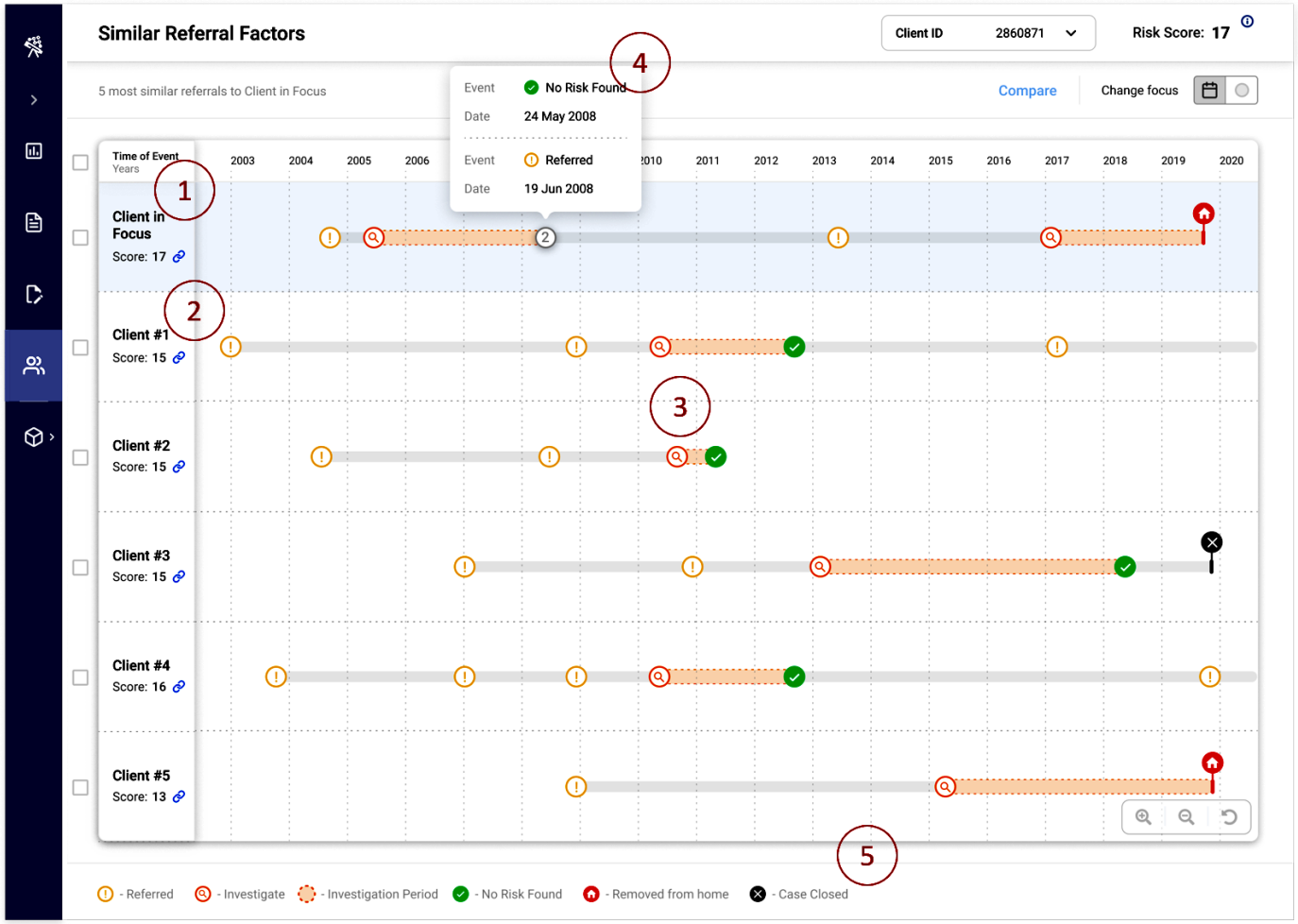

Figure 6-3: The similar cases interface of SIBYL includes the following elements (note that "client" refers to a child, per the screener's language): 1) The score and information about the current child in focus, with a link to their case file. 2) The score and information about the 5 most similar children in the database, based on factor values, with links to their case files. 3) A timeline of all CPS events that included each child, including referrals, investigations, and removals from home. 4) Detailed information about each event is available on hover. 5) The legend of event types.

6.1.4 Global Factor Importances: Understanding the Model

The *About Model* interfaces offer information about the model's general logic, outside of the context of a individual prediction.

The first *About Model* interface is the *Global Factor Importance* explanation, shown in Figure 6-5. This interface shows a global explanation in the form of the general, relative importance of each factor. It also provides a brief description of the model architecture and logic, as well as its performance metrics.

The global factor importance rankings were found using the Permutation Importance algorithm [6]. This algorithm computes the change in model performance if each factor is permuted individually. It therefore describes how closely each factor is linked to model performance. This interface was added as we theorized it may help screeners build trust in the model TR by seeing how it generally makes predictions, and because it may clarify the meaning of the prediction target CT.

6.1.5 Factor Distributions: Understanding Past Predictions

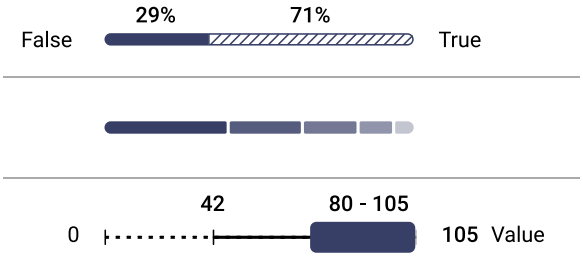

Figure 6-4: Factor Distribution visualizations. The top visualization is for binary factors. The middle visualization is for categorical factors, and can be hovered over for more information about the categories and their percentages. The bottom visualization is for numeric factors, and shows a simplified box-and-whiskers plot. The box-and-whiskers plot includes the global minimum and maximum values for a factor, the minimum and maximum for the selected risk score, and the first and third quartile values for the selected risk score.

The second *About Model* interface is the **Factor Distributions** explanation, shown in Figure 6-6. This interface shows the distribution of factor values among past cases of a given score, as well as the percentage of children with that score who

were removed from the home. This visualization gives a quick retrospective view of how the model performed in the past.

The Factor Distributions explanation uses three different kinds of visualizations, depending on the factor type. For binary and categorical features, a horizontal bar visualization, divided by percentage of value, was used. For numerical features, a simplified box-and-whiskers plot was used. These visualizations can be seen in Figure 6-4.

This interface was added as we theorized it may help screeners build trust in the model TR by seeing how it generally performs, and it may clarify the value of the prediction target CT by showing how it relates to a more tangible output of removals from the home.

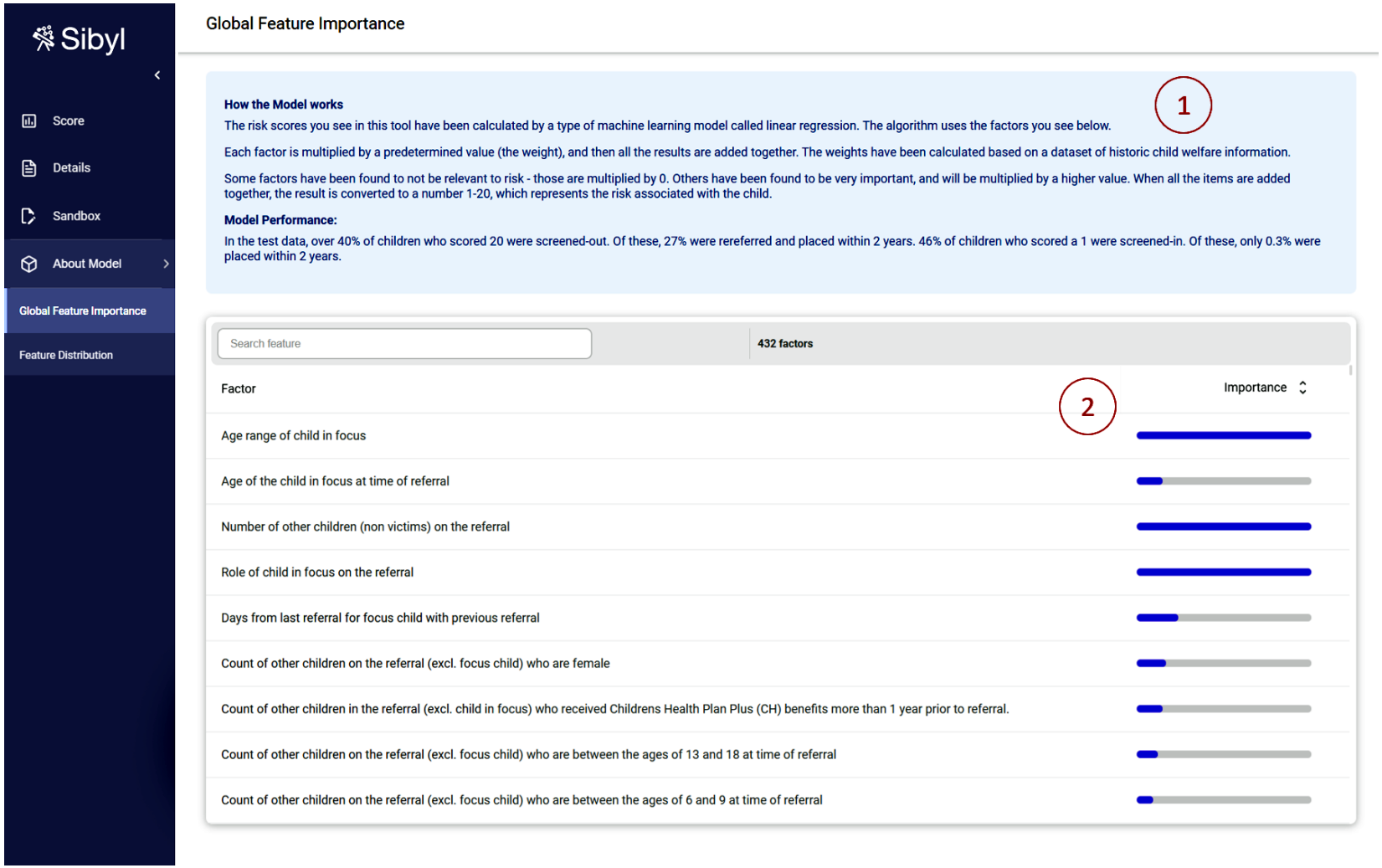

Figure 6-5: The global factor importance interface of SIBYL includes the following elements: 1) A natural language description of how the model works, worded to be understandable by end-users with little to no machine learning expertise. 2) The overall, global importances of all factors.

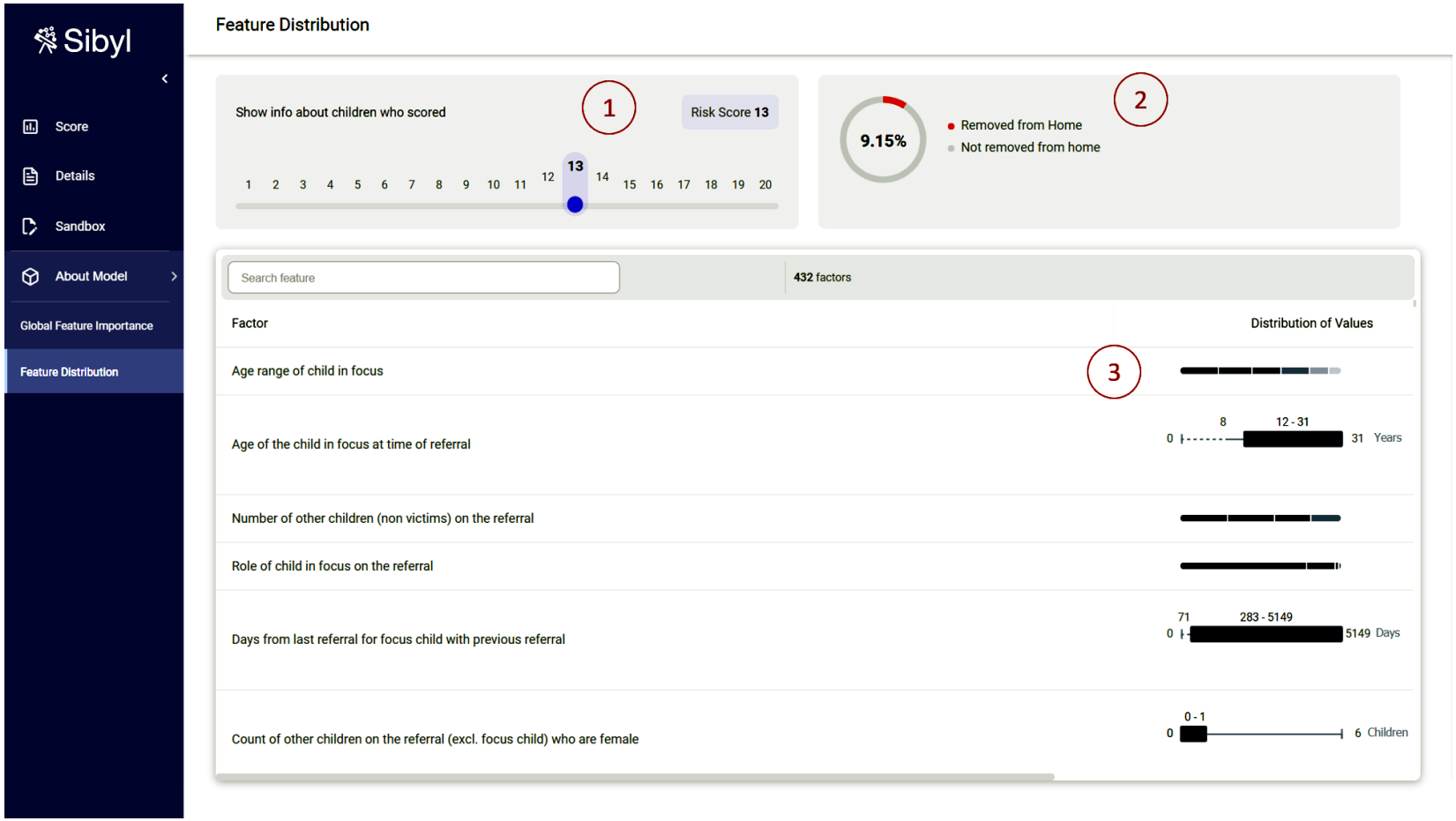

Figure 6-6: The Factor Distributions page of SIBYL includes the following components. 1) A selection bar of risk scores. Selecting a number from this bar provides the factor distributions for past children who scored each number. 2) A donut chart showing the percentage of children who scored the selected risk score and were removed from the home within 2 years (the model target). 3) The factor distribution visualizations for the selected risk score.

6.2 Iterating on Mockup Feedback

To garner feedback on our initial design, we showed the mock-ups of the five interfaces to our 19 collaborating child welfare screeners who provided feedback. We conducted an hour-long, semi-structured, open-floor style interview, where we showed the screeners the mock-ups and encouraged anyone to offer any comments or feedback.

Our goal from this step was to get additional, domain-expert insights on the proposed interfaces, giving them a final chance to give concrete, grounded feedback on the designs before implementation. Our findings from this step included:

1. **Case-Specific Details** Initial interview feedback suggested that the Case-Specific Details interface would be the most helpful, so it was kept as the first and default option. Additionally, screeners said that in their usual workflow, they would list "Risk" and "Protective" factors side by side. To mirror this, the updated version of this interface has a split-view toggle that shows the negative- and positive-contribution factors in two side-by-side tables, shown in figure 6-7

2. **Sandbox** Feedback received confirmed that the Sandbox interface has a risk of being misconstrued as suggesting action or reflecting real-world causal structures. However, screeners also said that they saw value in this interface as a supervision tool, used to review model and human decisions rather than actively during the decision-making process.

3. **Similar Cases** Screeners were concerned that this interface may cause poor decision making, as making decisions on a case based on past cases that seem similar is discouraged. Screeners reported that this kind of thinking can lead to biases or self-fulfilling prophecies. Therefore, it was decided that this interface would not be used at decision time.

   However, county officials pointed out that this kind of explanation could be used retroactively (outside of decision-making) to investigate unusual predictions made by the model for the purposes of model evaluation.

4. **About Model — Factor Importance** Screeners said that the Factor

Importance interface seemed intuitive, but may provide too much information for use during active decision making. Instead, they said it may be useful for training and education.

5. **About Model — Factor Distributions** Like the Factor Importance interface, screeners expressed concerns that the Factor Distributions interface shows too much information for use during active decision making. However, they said that it may be helpful for use in training, and for finding gaps in provided services.

6.3 Summary of Findings

We designed five ML augmentation interfaces, displayed through Sibyl. We presented high-fidelity mockups of these interfaces to screeners, which revealed which ones were most likely to be helpful, and what changes were needed to make the interfaces fit better with the existing workflow. For example, the *Case-Specific Details* interface was updated to have a side-by-side view to mirror the way screeners record risk and protective factors during their discussions.

Based on screener feedback on our designs, we decided to further test four of these interfaces — the *Similar Cases* interface was deemed to be likely to cause poor decision making through encouraging comparisons across unique cases.

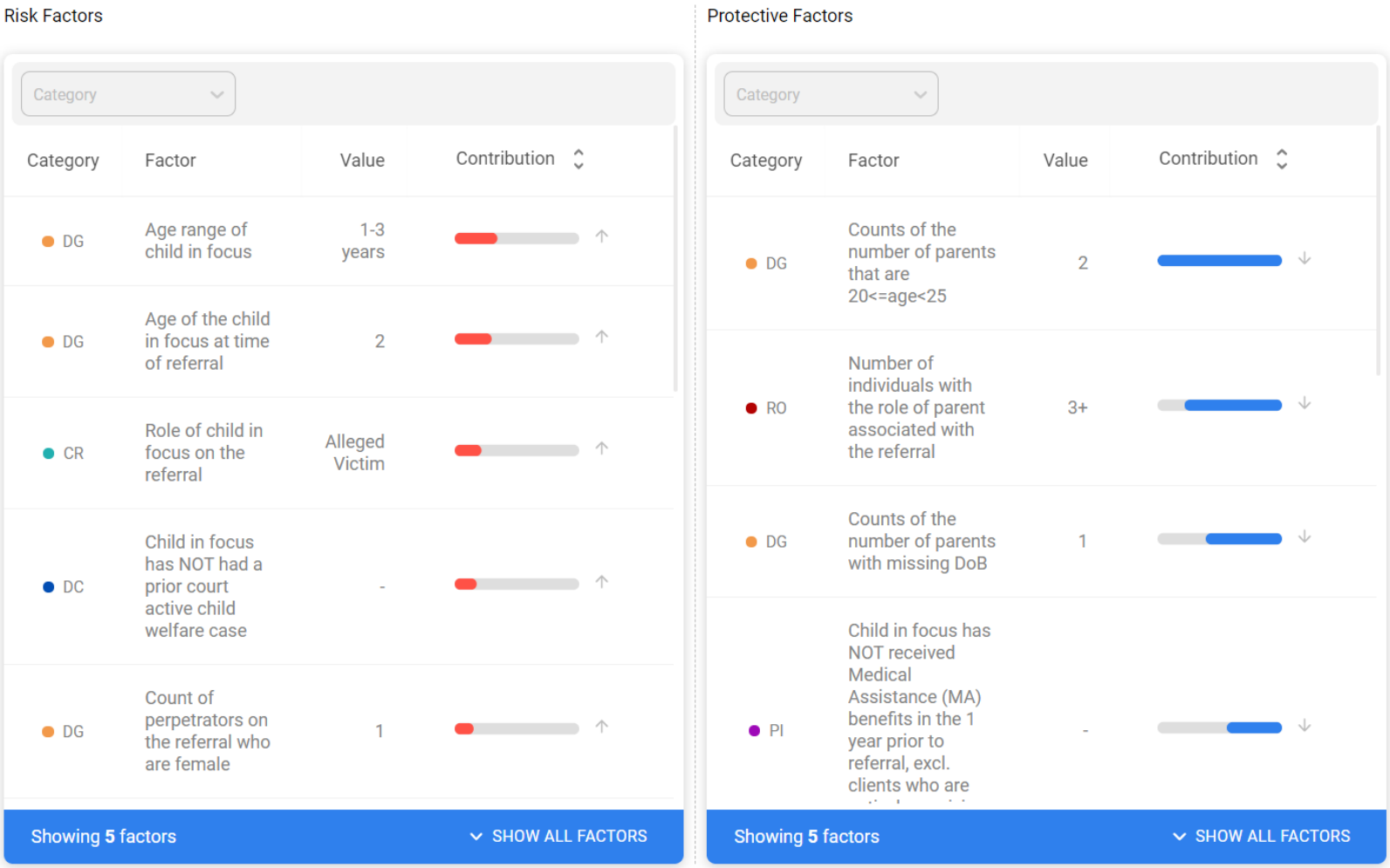

Figure 6-7: The side-by-side view of the Details interface. This view was added because it mirrors the current workflow of screeners, who tend to write down the protective and risk factors of the child side-by-side in a form.

Chapter 7

Evaluating Tool Usage

To evaluate SIBYL, we ran two formal user studies — first with non-expert participants (people without experience in child welfare screening), and then with expert participants (child welfare screeners). By running a study with non-expert participants first, we were able to fix obvious usability problems with SIBYL and avoid wasting experts' time. For brevity, we discuss the results of both user studies together in this section. Small usability fixes were made to the SIBYL design after the first user study based on feedback; otherwise, the studies had the same format.

In total, 13 of our collaborating child welfare experts participated in the study, as well as 12 non-expert participants. The non-expert participants included data and social scientists. 2 of the expert participants completed the task while video conferencing and screen-sharing with experimenters.

For data privacy reasons, all data used in this section was simulated or deidentified.

7.1 Study Procedure

The procedure for our user study is summarized in Figure 7-1. Participants were first shown a short video explaining how to use SIBYL. Next, they were shown 7 historical, de-identified case descriptions, accompanied by the model's prediction and SIBYL interfaces. The case descriptions were paragraph-form narratives with the information provided by the concerned party when making the referral. For example,

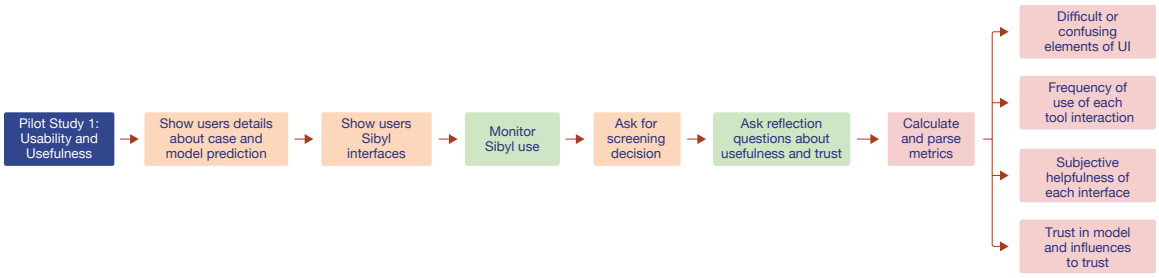

Figure 7-1: The procedure for our formal user study. Our participants were first shown the description of potential abuse from a child welfare referral, as well as the corresponding ML prediction risk score. Next, they were given the opportunity to interact with the SIBYL interfaces. Once they were ready, they were asked to make a screen-in or screen-out decision, and then asked a series of reflection questions.

a case description may include text like:

> *"Caller (teacher) says Abby (age 5) came into school with bruise on arm. Caller says Abby often comes in bruised. Abby told teacher she fell off bike. Teacher asked Abby's mother about this and mother started acting aggressive..."*

Next, participants were asked to make a screen-in or screen-out decision on the case, and asked to answer a series of reflection questions. These questions can be found in Appendix B.

Not all participants completed all 7 cases. In total, experts completed 73 individual case analyses, and non-experts completed 75.

7.2 Study Results

7.2.1 Helpful Interfaces

To address **RQ2**, we analyzed the self-reported helpfulness of each augmentation interface.

The Case-Specific Details interface was considered the most helpful interface by a large margin by both experts and non-experts. It was labelled as being helpful by experts in 91.8% of case analyses, and by non-experts in 90.7% of case analyses.

This was significantly higher than Sandbox (experts: 16.4%, non-experts: 22.6%) and Factor Distributions (experts: 20.5%, non-experts: 8.0%). Factor Importance was never listed as helpful by either group.

7.2.2 Reliance on Sibyl

Unsurprisingly, non-experts were more likely to report listening to the model without considering the added case information in Sibyl. One non-expert participant commented,

> "*No idea what is going on in this case description – so completely defer to the model here.*"

Another non-expert participant commented

> "*I found the score useful - and used it as a justification for screening out without exploring in detail all the factors.*"

Additionally, non-experts reported that they used the model "*a lot*" or "*a great deal*" on 46.4% of cases, while experts chose these options in 15.6% of cases.

7.2.3 Impact on Trust

The interfaces in Sibyl were reported to both increase and decrease expert user trust of the model, for different reasons. Tables 7.1 and 7.2 list the common reasons given by expert participants for trusting the model more or less, respectively. We see that agreeing with the model's score increases trust of the model the most. Beyond this, the Case-Specific Details page was frequently cited as increasing model trust, either due to specific factors listed or more general elements of the page, such as the number of factors. Trust was reduced when there was confusion or inconsistencies in the presented information, or when the model did not consider important factors that participants knew about.

Category	# of answers	Sample answers
General comments about being shown protective and risk factors (Details page)	8	"*The details and the risk and protective factors and the contribution they have*" "*Info in the details and risk factors*"
Specific factors listed as risk or protective (Details page)	8	"*The past number of child welfare involvements (listed in the features listed)*", "*The risk factors involved, especially prior placements, benefits, and current CPS involvement*"
The number of factors listed	5	"*Very few risk factors*", "*The lack of protective factors*"
The score agreed with screener intuition	10	"*...the children are not residing with the alleged perpetrators which I would assume would reduce the risk score*", "*Model prediction makes sense*"
The explanation agreed with screener intuition	3	"*Risk factors made sense for the model prediction number*"
General comments about sandbox page	1	"*Details under sandbox of why the risk level was so high*"
General comments about the explanation providing more information or understanding	6	"*Allowed for more understanding*", "*History clarification*"

Table 7.1: Summary of answers to the question "what made you trust the model more/less" among expert participants who listed having "*a great deal*", "*a lot*" or "*a moderate amount*" of trust in the model. The first column lists the general themes we found in the answers. The second column list the number of answers that fell within each theme. The final column lists selected answers for each theme.

Category	# of answers	Sample answers
A specific, key factor was not considered by the model	3	"*This is a case for law enforcement, not CPS*", "*...it may have been handled during the open case*"
The importance weighting of factors was off	2	"*Young, vulnerable children being left alone is still cause for concern, despite past involvement*"
The score disagreed with screener intuition	3	"*Seems high, no health history, no real proof of any drug use, no proof child is at any risk of abuse*"
There was some confusion about information presented	2	"*There were discrepancies between the info in the referral and the info provided by the tool*"
The screener wanted more information	2	"*I would want to see other referrals for the family*"

Table 7.2: Summary of answers to the question "what made you trust the model more/less" among expert participants who listed having "*a little*" or "*not at all*" trust in the model. The first column lists the general themes we found in the answers. The second column list the number of answers that fell within each theme. The final column lists selected answers for each theme.

7.2.4 Information Presentation

To address **RQ3**, we categorize and summarize the comments made by users regarding Sibyl design choices, as well as the steps we took to address them.

1. **Too many factors shown** The model was originally trained on over 400 factors, but many of these factors have zero or near-zero weight. One expert participant commented:

   > "*Too many factors listed. I only want to see the material risk and protective factors.*"

   Another said

   > *"There are a lot of features to go through ... Some of those features don't seem to be meaningful in terms of increasing/decreasing risk"*

   Our updated version of SIBYL only shows 10 factors by default, with an option to show more.

2. **Confusion caused by correlated factors** The model uses some engineered factors, resulting in factors that have deterministic relationships. For example, there is a numeric factor called `AGE OF CHILD`, and then a set of binary factors referring to each age group: i.e. `CHILD IS LESS THAN 1 YEAR OLD`, `CHILD IS BETWEEN THE AGES OF 1 AND 3`, etc. These factors may cause confusion when shown directly to users. In addition to increasing the cognitive load on users without providing additional useful information, explanations using these factors may reveal seemingly contradictory or unusual relationships. For example, an age category may contribute greatly, while the numeric age factor does not.

   Additionally, having these correlated factors causes confusion on the sandbox page, as it is possible to change one factor without changing all of the other deterministically-correlated factors in its set. One participant commented,

   > *"I'm not sure, in the sandbox, if I change one feature, other features will be changed automatically."*

   To solve these problems, we combined the correlated factors in the SIBYL interface, forming categorical factors out of binary one-hot encoded factors, and

summing the additive contributions. This process is inverse of what ML engineers do when preparing data for modelling.

3. **Confusion caused by Boolean terminology** One source of confusion was the method of displaying Boolean factors. In our original design, we displayed the description of the factor, with a value of `True` or `False`. This is the most accurate way of representing the model's logic, but it is not the most intuitive way for our end-users. One non-expert participant said,

   > *"The 'true' and 'false' is hard to interpret... Would rather have a positive statement (e.g., no perpetrator named)"*

   Therefore, our final version of SIBYL instead states only true statements about the child — including by negating descriptions of false factors. For example, the factor `CHILD HAS SIBLINGS` with a value of `False` will be displayed as `CHILD DOES NOT HAVE SIBLINGS`.

7.3 Summary of Findings

Our formal user study revealed that child welfare non-experts had an easier time trusting the model and relying on its risk score predictions than experts. Experts had the easiest time trusting the risk score prediction when it agreed with their own intuition. Screeners also trusted the model more when they saw contributing factors that they agreed were relevant, or saw a high or low risk score associated with a large number of risk factors or protective factors, respectively. They trusted the model less when the risk score prediction differed greatly from their intuition, or when the information presented appeared incorrect or confusing. Screeners wanted the interfaces to mirror their own language, and be simple enough to be parsable quickly without excessive effort.

Chapter 8

Discussion

The process of collaborating with domain experts to find and address the ML usability challenges present in child welfare screening revealed a real need for ML augmentation tools. At the same time, it revealed that the existing work in ML augmentation may not be sufficient to effectively address the ML usability challenges for less technical domains.

In this section, we summarize the broad, generalizable lessons about improving the usability of ML models through augmentation tools, based on our experiences in the domain of child welfare screening. Because improving the usability of ML models is such a highly context-dependent task, as described in the introduction, we will limit this discussion to lessons that may apply to domains that are similar in their context factors — i.e. users with high domain-expertise and low technical-expertise, making decisions using ML models as auxiliary tools in high-risk/high-impact domains.

We also discuss the remaining challenges that these lessons suggest may need to be addressed before ML models can be truly usable to non-technical domain experts for decision making.

8.1 The Importance of Interpretable Factors

Simple models, such as regression, are often cited as being inherently interpretable [28]. However, our case study suggested that even simple models may cause confusion

in the target audience, and lead to challenges when attempting to explain model predictions for active decision making.

Instead, our work found that, for the purposes of making models usable for end-users, the interpretability of the model factors (in ML terms, features) may be most important. In our study, the screeners were often confused when explanations used factors that did not have clear implications on risk. For example, in our user study, one participant said

> *"... 2 parents have missing date-of-birth is shown as a significant blue bar which I can't imagine is protective."*

Additionally, as discussed in section 7.2.4, one-hot-encoded factors were not interpretable, and many of the reasons screeners trusted the model more or less (Tables 7.1 and 7.2) related to the specific factors.

The phrase *interpretable factors/features* still lacks a formal and thorough definition. In our case study, all the factors we were working with were interpretable by the strictest definition — humans could understand exactly which real-world concepts they represented, and their association to these concepts. This level of interpretability was not sufficient to prevent confusion.

The ML augmentation literature may benefit from a more precise definition of *interpretable factors/features*; one that changes based on context. As a start for this, we propose some possible definitions for interpretable features by context here; however, this thesis only provides empirical support for the first of these:

1. **Non-technical domain experts, using explanations while making fast, high-impact decisions**

   This context matches the child-welfare context.

   Users need features to be immediately **human-readable**, as they do not have time to consult code-books or remember meanings of cryptic phrases. For the same reason, features should be in **the most natural wording** — for example, described using positive or negative language rather than having an attached `TRUE` or `FALSE` value.

Users need to **understand the expected effect of features**. In this domain, users are not looking to learn new patterns about the domain — such learning would need to happen offline, prior to decision making. With so much at stake, users only want to get access to more information about the case at hand. Therefore, features such as `PARENTS ARE MISSING DATE OF BIRTH` from our use-case, which do not have an obvious effect on the output, may cause confusion.

Features need to **make sense together**. A machine learning model treats each feature as a mathematical variable, but a human looks at them all as a collection representing an individual. For example, in one of our sample cases, a child was listed as both being an infant and having zero past juvenile justice cases. Both of these pieces of information were actively used by the model and seen as important; however, one screener in an interview pointed out that they seem redundant when presented together (*it can be safely assumed that no infant will have had a juvenile justice case*). Explanations that use these features together come across as overly-informative and add additional confusion.

2. **Machine-learning experts using explanations to debug a model**

   In this use case, the features in the explanation should **exactly match the form used by the model**, at least in some point of the explanation process. Here, users want an explanation of the model that is accurate to be useful in adjusting the model.

   Time restraints may be fewer in this context, so users are able to more deeply consider the effects of unusual features.

3. **Technical domain experts making data-driven decisions.**

   This use case is similar to the first, in that the end goal is to make an informed decision, not necessarily to understand the model itself. Features should still be **human-readable** and presented with **the most natural wording**.

   However, experts who are more used to making strictly data-driven decisions

may be equipped to interpret features that do not have clear implications.

The most natural way to address this concern is by avoiding using confusing or un-interpretable features in model training. In some cases, this can be done without significant effect on model performance. However, in other cases another approach may be necessary — either through finding ways to clarify features through augmentations, or through hiding/modifying the truth of the explanations (see section 8.2). More work will need to be done on determining the best approach.

8.2 Accuracy versus Fidelity

Robnik-Sikonja and Bohanec [27] define the *accuracy* of an explanation as how well it generalizes to other unseen examples (i.e., how accurately these rules predict what happens in the real world), and *fidelity* as how well an explanation describes the model itself.

Our users were mostly interested in getting *accurate* explanations that provided information about *the case at hand*. As evidenced by all three findings about the interface design (section 7.2.4), users wanted to receive information about the model in a *language* and *format* that mirrored their own, not the format used by the model itself. This is also evidenced by design requests like using the terms *risk and protective factors*, rather than the more ML-centric terms *negative and positive features*.

Another example of accuracy causing confusing comes from the information provided by SHAP. SHAP values represent the contribution of a feature, given a specific feature coalition, to the model prediction compared to the average prediction. This is often considered a way of explaining with high fidelity, but can be very confusing to end-users. In one case in the child welfare case study, the feature `NUMBER OF PAST REFERRALS WITH EMOTIONAL ABUSE` with a value of 8 was listed as being protective (reducing risk). This is likely because the average referred child has a large number of referrals, and emotional referrals may be less likely to lead to removal from home than other kinds. However, describing emotional abuse referrals as "protective" appears to be wrong to most domain experts.

Another example was the desire for information shown to form a coherent description of a child, even though the model does not consider this. An example of this is describing an infant as having no juvenile justice cases, as described in section 8.1.

This finding reveals a important ethical question: to what extent is it acceptable to hide the truth, or even more questionably, modify the truth, when providing ML explanations? The truth of what works best for improving model performance may differ from what makes users most likely to trust a model prediction.

8.3 Explaining Explanations

Another finding of this work is that some explanations may require a second layer of explanation, answering the question "*why does the model use this logic to make this prediction.*"

Understanding how, for example, a feature contributes to the model prediction will not satisfy users unless they understand why the feature has this contribution. This is especially important in cases where there is a logical reason for feature to contribute in a given way, but the users do not know the reason — in this case, the explanation may cause them to wrongfully trust the model less.

In our formal user study, one screener commented that the explanations seemed inconsistent because, in one case, age contributed significantly, and in another case it contributed very little. They wrote,

> *"it would really help to see the why behind the weight [of the contribution of the age feature]"*

In effect, they are asking for an additional layer of explanation, based on why the correlation that the model found exists.

This need for more explanation is also often caused by less interpretable features (for example, the question as to why parents' missing date-of-birth is protective, described in section 8.1).

The most natural way to provide this additional information may be to have a knowledgeable human hand-write reasonable explanations for model explanations.

For example, in the case of feature contribution explanations, a domain expert may write a sentence that will be presented alongside the `AGE` feature — "Infants are almost always investigated because of their vulnerability. In the case of adolescents, age becomes less important compared to other, more revealing features." However, this approach is time consuming for the domain experts, and can only be used when all these explanations can be reasoned out.

Other approaches might include carefully combining explanations or using data visualization to demonstrate the relevant patterns in the training data. For example, many confusions about feature contributions could be clarified by showing relevant example-based explanations. Further work on finding ways to explain explanations is needed.

8.4 Non-Applicable Usability Challenges

One interesting finding of our work was the usability challenges we did not see evidence of in our case study.

For example, one possible use of explanations is to give humans the ability to actively correct errors in a model's logic. We did not see evidence of this behavior from our users, however. There are several possible reasons for this. First, our users are making decisions in a very limited time, and do not have additional time to review the model's quality. Second, our users are thoroughly analyzing every case on their own and were only using the model as an extra flag. Finally, the users already have some discomfort about the model, likely due to the high associated risk. As a result, our users tended to discount the model altogether if they did not believe it was correct about a particular case.

Additionally, users expressed almost no interest in learning about the model itself through explanations. A common explanation need addressed by the literature [18] is to understand how the model works (model transparency), possibly for debugging. In our case study, however, only once (see section 5.3, item 3) did any screener express interest in understanding the details of how the model worked under the hood — and

even then, they were mostly looking for a broad overview. It is reasonable to believe that this finding would generalize to any domain with non-technical domain experts, or even in most cases where the ML augmentation tool is being used to assist with decision making unrelated to the model itself.

8.5 Cognitive Biases

Wang et. al. [34] introduced a list of the cognitive biases explanations can help address. Our experience with child welfare screeners additionally suggested some of these cognitive biases could be *encouraged* by the explanations and other forms of further information. For example:

1. **Representativeness Bias [34]**: Case-based explanations that offer similar examples to the case at hand (such as our Similar Cases page) risk encouraging users to make decisions based on similarities to another case.

2. **Causation vs Correlation**: Counterfactual-based explanations, which consider how the model prediction would change under different circumstances, made participants more likely to interpret the explanations as containing information about the causal structure of the world.

3. **Availability Bias [34]**: A factor-contribution explanation that is sorted in ascending order (and therefore lists negative contributions first) may result in different decisions than one that is sorted in descending order (and therefore lists positive contributions first) due to availability bias, which causes humans to put too much importance on recent or memorable events or information.

Further work and user studies may better reveal the extent to which these biases are caused or exacerbated by ML augmentation tools.

8.6 Evaluating Explanations

There is still a lack of formal frameworks for evaluating ML explainability algorithms, which can make it difficult to select a specific explanation algorithm even once the need for one has been found. For the work in this thesis, we selected popular explanation algorithms with theoretical backings, such as SHAP. However, we could have selected alternative choices, and the resulting explanations and effect on user decision-making may have differed.

Different explanation algorithms may produce different explanations for the same input with the same model. Figure 8-1 illustrates the problem. On a very simple, synthetic input, the disagreements from popular interpretation methods (Saliency [29], SHAP [19], LIME [25], and Occlusion [37]) become clear.

A similar concern was raised by Adebayo et. al. [1], who tested the fidelity of sanity map explanation methods by comparing the resulting map after randomizing convolutional neural network layer weights. This work found that explanations can look reasonable even with an entirely random model, thereby suggesting that visual or common-sense inspections on explanations are not sufficient for evaluation.

When two explanations generated by different algorithms disagree with each other on the same input with the same model, or act in a way that seems contrary to common sense, there are multiple possible explanations:

1. One (or both) of the explanations are misrepresenting the model.

2. The explanations are selecting different parts of the same explanation. The true explanation of models is often too large and complex to show entirely to humans, so an important part of an explanation algorithm is selecting which part of the explanation to show.

3. The explanations are explaining different things. For example, the global explanation modification of SHAP explains which features contribute most to outputs [19], while Permutation Feature Importance explains which features are most important to model performance [6]. This subtle difference may lead to very

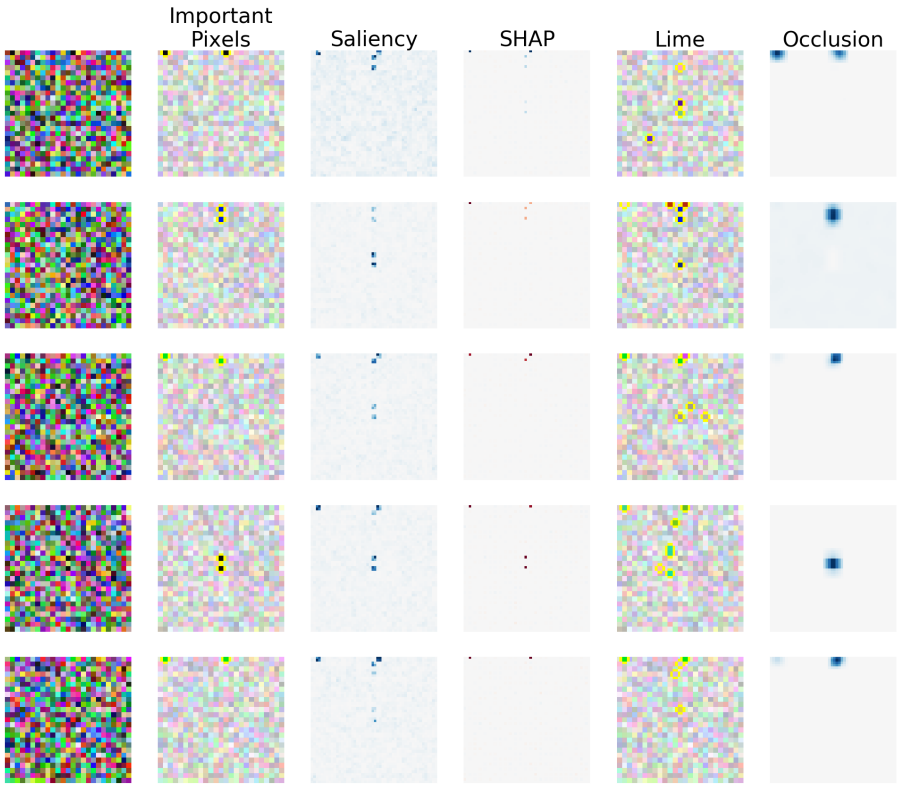

Figure 8-1: A demonstration of different interpretations on a very simple synthesized example, using three popular methods. Each image in a class is differentiated from other classes by two specific 4x4 super-pixels. Saliency and SHAP find the same pixels, but disagree on whether the entire super-pixel is important or not. Occlusion finds some, but not all, important super-pixels. LIME, a highly cited model-agnostic interpretation method, fails on this example.

different results.

Note that in case 2 and 3, the explanations are both still valid, while this is not necessarily so in case 1. Being able to effectively differentiate between these three cases is essential when deciding which explanations to use.

In addition, as demonstrated by the child welfare case study, understanding the theoretical soundness of explanations is not enough to judge their usefulness. Evaluating the effectiveness of explanations requires involvement from the end-users. This can be done through a series of user studies, set up to empirically evaluate explanations on multiple axes.

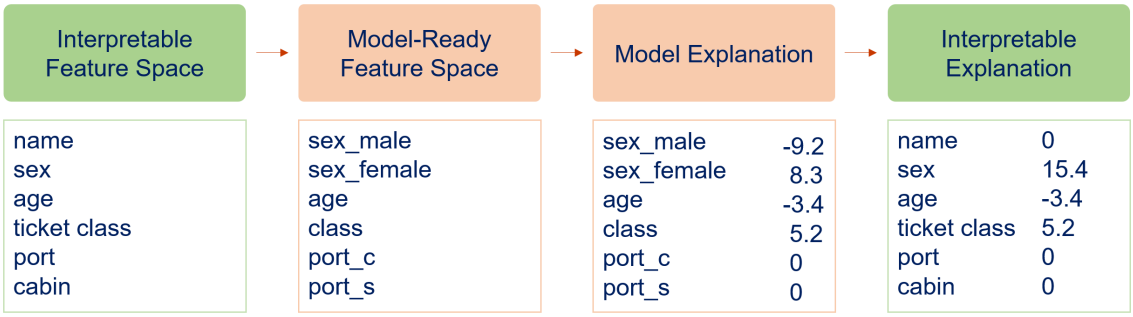

Figure 8-2: The explainable ML workflow begins and ends at an interpretable state, factoring in the fact that some explanation algorithms work only on model-ready data. The figure provides an example of a feature importance explanation using the Titanic dataset [12]

For example, to quantify the change in outcome that comes from interpretability, we define the following user study. In the control condition, subjects are asked to take an action (such as buying and selling stocks or giving medication to patients), and are provided with relevant predictions from an ML model (such as stock price forecasts or patient diagnoses). The outcome of this action is recorded. In the experimental condition, subjects are also given an explanation of the model's prediction. Again, the outcome is recorded. These outcomes may take the form of the summation of several outcome values, each with a domain-expert specified weight $(w_1, \ldots, w_n)$.

We then define the quality of an explanation as

$$Q_{out} = \frac{\sum_i w_i O_i^e}{\sum_i w_i O_i^{me}} \tag{8.1}$$

8.7 Systems for Explanations

Currently, there are few systems developed for the explicit purpose of explaining and augmenting ML predictions to end-users, and existing ML systems may not be sufficient for usability.

ML workflows are usually run by preprocessing data and then making predictions. Because the output is this prediction, there is no need to reverse the transformations back to the original state (which is often also the most interpretable state). In other

words, the transformation workflow goes in one direction: from the original data to the model prediction. When explanations are applied, they are often applied at the machine-ready data state, which reduces the interpretability of the resulting explanations.

For the child welfare case study described in this thesis, we need explanations in terms of the human-readable, original data state. To accomplish this, we began the process of developing an ML library, `Pyreal`, that allows for ML prediction and explanation pipelines that begin and end at a human-readable, fully interpretable state are needed. Figure 8-2 shows an example of this kind of workflow.

Additionally, there is a need for a configurable API framework for developing ML augmentation tools. For this thesis, we developed the SIBYL interface and corresponding API, which acts as a first step towards filling this need. All the representations featured in SIBYL are generalizable, and could be used in a wide variety of domains.

Chapter 9

Conclusion

In this thesis, we investigated the ML usability challenges and possible mitigating tools in the domain of child welfare screening and made several findings.

First, we confirmed that augmenting real-world decision-making workflows with ML models is a non-trivial task that requires time and effort from both ML practitioners and end-users. Additional information alongside a model prediction is often necessary.

Second, we began the process of establishing a generalizable framework for selecting and applying tools that provide this additional information, based on our findings from the case study. Here, we list a set of guidelines that ML practitioners should consider when attempting to deploy usable ML applications.

Prior to developing ML augmentation tools, practitioners should determine precisely which usability challenges are relevant to the domain, likely through observations and interviews. As a starting point to this process, our case study suggested that lack of trust (TR), difficulty reconciling disagreements (RD), confusing prediction target (CT), and ethical concerns (EC) may be most relevant to high-stakes domains with non-technical expert decision-makers. Pinpointing relevant challenges will prevent wasting time on non-applicable challenges (Section 8.4).

ML explanations should provide the information that users are looking for, which often may not be about how the model works itself (Section 8.2). To aid this, explanations may themselves require further information to make sense (Section 8.3).

Additionally, while this thesis focused on usability challenges that arise in existing ML models, we did find that addressing these challenges may be difficult or impossible unless if the model uses features that are not appropriately interpretable for the intended audience (Section 8.1).

Finally, we identified remaining gaps that will need to be filled by future research to ensure ML models can efficiently be made usable. We need systems developed for this explicit purpose (Section 8.7), and formal methodologies for evaluation (Section 8.6).

Bibliography

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity Checks for Saliency Maps. *32nd Conference on Neural Information Processing Systems*, page 11.

[2] Benjamin Bengfort and Rebecca Bilbro. Yellowbrick: Visualizing the Scikit-Learn Model Selection Process. *Journal of Open Source Software*, 4(35):1075, March 2019.

[3] Centers for Disease Control and Prevention (CDC). Preventing Child Abuse and Neglect Factsheet, 2020.

[4] Children's Bureau. Child Maltreatment 2018: Summary of Key Findings, 2020.

[5] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning, February 2017.

[6] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research*, 20:1–81, December 2019.

[7] Marco Gillies, Rebecca Fiebrink, Atau Tanaka, Jérémie Garcia, Frédéric Bevilacqua, Alexis Heloir, Fabrizio Nunnari, Wendy Mackay, Saleema Amershi, Bongshin Lee, Nicolas d'Alessandro, Joëlle Tilmanne, Todd Kulesza, and Baptiste Caramiaux. Human-Centred Machine Learning. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 3558–3565, San Jose California USA, May 2016. ACM.

[8] Google. Machine Learning Glossary: Fairness. https://developers.google.com/machine-learning/glossary/fairness.

[9] Fred Hohman, Andrew Head, Rich Caruana, Robert DeLine, and Steven M. Drucker. Gamut: A Design Probe to Understand How Data Scientists Understand Machine Learning Models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–13, Glasgow, Scotland Uk, 2019. ACM Press.

[10] Dan Hurley. Can an Algorithm Tell When Kids Are in Danger? *The New York Times*, January 2018.

[11] Varghese S. Jacob, James C. Moore, and Andrew B. Whinston. An analysis of human and computer decision-making capabilities. *Information & Management*, 16(5):247–255, May 1989.

[12] Kaggle. Titanic - Machine Learning from Disaster. https://kaggle.com/c/titanic.

[13] Minsuk Kahng, Pierre Y. Andrews, Aditya Kalro, and Duen Horng Chau. ActiVis: Visual Exploration of Industry-Scale Deep Neural Network Models. *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 24, April 2017.

[14] Will Kenton. How Cost-Benefit Analysis Process Is Performed. https://www.investopedia.com/terms/c/cost-benefitanalysis.asp.

[15] Hyunil Kim, Christopher Wildeman, Melissa Jonson-Reid, and Brett Drake. Lifetime Prevalence of Investigating Child Maltreatment Among US Children. *American Journal of Public Health*, 107(2):274–280, February 2017.

[16] Bum Chul Kwon, Min-Je Choi, Joanne Taery Kim, Edward Choi, Young Bin Kim, Soonwook Kwon, Jimeng Sun, and Jaegul Choo. RetainVis: Visual Analytics with Interpretable and Interactive Recurrent Neural Networks on Electronic Medical Records. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):299–309, January 2019.

[17] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Faithful and Customizable Explanations of Black Box Models. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 131–138, Honolulu HI USA, January 2019. ACM.

[18] Zachary C. Lipton. The Mythos of Model Interpretability. *arXiv:1606.03490 [cs, stat]*, June 2016.

[19] Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, page 10, 2017.

[20] Scott M. Lundberg, Bala Nair, Monica S. Vavilala, Mayumi Horibe, Michael J. Eisses, Trevor Adams, David E. Liston, Daniel King-Wai Low, Shu-Fang Newman, Jerry Kim, and Su-In Lee. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, 2(10):749–760, October 2018.

[21] Tamara Munzner. A Nested Model for Visualization Design and Validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, November 2009.

[22] National Highway Traffic Safety Administration (NHTSA). Automated Vehicles for Safety. https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety, September 2017.

[23] Jorge Piazentin Ono, Sonia Castelo, Roque Lopez, Enrico Bertini, Juliana Freire, and Claudio Silva. PipelineProfiler: A Visual Analytics Tool for the Exploration of AutoML Pipelines. *arXiv:2005.00160 [cs]*, September 2020.

[24] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, and David Cournapeau. Scikit-learn: Machine Learning in Python. *MACHINE LEARNING IN PYTHON*, page 6.

[25] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, February 2016.

[26] Darrel Richardson. The Effects of a False Allegation of Child Sexual Abuse on an Intact Middle Class Family. *IPT*, 2, 1990.

[27] Marko Robnik-Šikonja and Marko Bohanec. Perturbation-Based Explanations of Prediction Models. In Jianlong Zhou and Fang Chen, editors, *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, Human–Computer Interaction Series, pages 159–175. Springer International Publishing, Cham, 2018.

[28] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019.

[29] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv preprint arXiv:1312.6034*, December 2013.

[30] Hendrik Strobelt, Sebastian Gehrmann, Hanspeter Pfister, and Alexander M. Rush. LSTMVis: A Tool for Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks. *arXiv:1606.07461 [cs]*, October 2017.

[31] Rhema Vaithianathan, Haley Dinh, Allon Kalisher, Chamari Kithulgoda, Emily Kulick, Megh Mayur, Athena Ning, Diana Benavides Prado, and Emily Putnam-Hornstein. Implementing a Child Welfare Decision Aide in Douglas County, December 2019.

[32] Rhema Vaithianathan, Emily Putnam-Hornstein, Nan Jiang, Parma Nand, and Tim Maloney. Developing Predictive Models to Support Child Maltreatment Hotline Screening Decisions: Allegheny County Methodology and Implementation, 2017.

[33] Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y. Lim. Designing Theory-Driven User-Centric Explainable AI. In *Proceedings of the 2019 CHI Conference*

on Human Factors in Computing Systems - CHI '19, pages 1–15, Glasgow, Scotland Uk, 2019. ACM Press.

[34] Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y. Lim. Designing Theory-Driven User-Centric Explainable AI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–15, Glasgow, Scotland Uk, 2019. ACM Press.

[35] Qianwen Wang, Yao Ming, Zhihua Jin, Qiaomu Shen, Dongyu Liu, Micah J. Smith, Kalyan Veeramachaneni, and Huamin Qu. ATMSeer: Increasing Transparency and Controllability in Automated Machine Learning. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12, May 2019.

[36] Zijie J. Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng Chau. CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. *arXiv:2004.15004 [cs]*, April 2020.

[37] Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. *European conference on computer vision*, November 2013.

[38] Jiawei Zhang, Yang Wang, Piero Molino, Lezhi Li, and David S. Ebert. Manifold: A Model-Agnostic Framework for Interpretation and Diagnosis of Machine Learning Models. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):364–373, January 2019.

Appendix A

Mitigating Tool Definitions

Here, we define the types of mitigating tools discussed in this paper and Table 2.1.

Global Explanation: An explanation of a model's general logic, achieved through methods such as quantifying the overall importance of features or visualizing the model boundary [5].

Local Explanation: An explanation as to why a model made an individual prediction, achieved through methods such as quantifying how much each feature contributed to this particular prediction [5].

Cost-Benefit Analysis: A measurement of the expected total reward from taking an action, defined by the expected benefits minus the costs [14]. In the case of machine learning, this would involve providing information about the expected results of a prediction alongside the prediction itself.

ML Fairness Metrics: Mathematical approaches to measuring the level of bias present in models [8]

Appendix B

User Study Questions Asked

Table B.1 contains the complete list of questions we asked during our user study.

When?	**Response Type**	**Question**
Beginning	Multiple choice	What is your experience with child welfare screening?
After each case	Multiple choice	Would you choose to screen-in or screen-out?
	5-point Likert scale	How confident are you in your decision?
	5-point Likert scale	How much did the prediction score impact your decision?
	5-point Likert scale	How much did you trust the model's prediction for this case?
	Free response	What caused you to trust the model more or less?
	Multiple-multiple choice	What explanations, if any, did you find helpful?
	Free response	Any other comments?
End	5-point Likert scale	How helpful did you find the model's predictions overall?
	5-point Likert scale	How helpful did you find the Sibyl tool's explanations?
	Multiple-multiple choice	Which explanation did you find most helpful overall?
	Free response	Were there any feature categories you found more or less helpful?

Table B.1: Questions asked in the formal user study.