

UC Santa Cruz

UC Santa Cruz Electronic Theses and Dissertations

Title

Accurate, Fair, and Explainable: Building Human-Centered AI

Permalink

<https://escholarship.org/uc/item/4d80t5j5>

Author

Springer, Aaron

Publication Date

2019

Copyright Information

This work is made available under the terms of a Creative Commons Attribution License, available at <https://creativecommons.org/licenses/by/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**ACCURATE, FAIR, AND EXPLAINABLE:
BUILDING HUMAN-CENTERED AI**

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Aaron Springer

June 2019

The Dissertation of Aaron Springer
is approved:

Dr. Steve Whittaker, Chair

Dr. Marilyn Walker

Dr. Peter Pirolli

Lori Kletzer
Vice Provost and Dean of Graduate Studies

Copyright © by

Aaron Springer

2019

Table of Contents

List of Figures	vii
List of Tables	viii
Abstract	ix
Acknowledgments	x
1 Introduction	1
1.1 Contributions	5
1.1.1 Designing Accurate Personal Informatics Systems:	5
1.1.2 Auditing and Removing Algorithmic Bias:	6
1.1.3 When Do Users Want Transparency:	6
1.1.4 How Do Users Want to Interact with Transparency:	8
1.1.5 AI and Explanations in the Wild:	8
2 Literature Review	9
2.1 New Aspects of User Experience in Intelligent Systems	11
2.1.1 Accuracy	11
2.1.2 Transparency	14
2.1.3 Algorithmic Bias and Machine Learning Fairness	20
2.2 Domains	25
2.2.1 Emotional Analytics	26
2.2.2 Learning Analytics	32
2.2.3 Voice Interfaces	33
3 Improving Accuracy in Intelligent Systems	37
3.1 Introduction: Why we Need Accurate Mood Models to Promote Mental Wellbeing	38
3.2 Methods	43
3.2.1 EmotiCal System Overview	43
3.2.2 Users	51
3.2.3 Procedure	51

3.3	Results	52
3.3.1	Modeling the Impact of Activities on Daily Mood	52
3.3.2	Activities are Critical for Explaining Mood: Health and Social are Important	53
3.3.3	User Explanations Improve Mood Models	56
3.4	Discussion	64
4	Removing Bias in Intelligent Systems	70
4.1	Introduction	71
4.2	Methods	74
4.2.1	Prototype and Infrastructure	75
4.2.2	Identifying Underserved Content	76
4.3	Identifying Content Results	79
4.3.1	Voice Interfaces May Underserve Specific Genres	79
4.3.2	Typology of Underserved Content	82
4.4	Correcting Underserved Content	86
4.5	Correcting Content Results	91
4.5.1	Aliases Improve Content Accessibility	92
4.6	Discussion	94
4.6.1	Limitations and Trade-Offs for Practitioners	96
4.6.2	Bugs or Biases	97
4.6.3	Creative Intricacies	98
4.6.4	Conclusion	98
5	When Do Users Want Transparency	99
5.1	Introduction	100
5.1.1	Contribution	102
5.2	Why Emotional Analytics?	103
5.3	Research System: E-meter	104
5.3.1	Machine Learning Model	105
5.3.2	Version 1: Document-level:	106
5.3.3	Version 2: Word-level	106
5.4	Study 1	108
5.4.1	Method	108
5.4.2	Study 1 Results	111
5.4.3	Study 1 Summary	113
5.5	Study 2	114
5.5.1	Method	114
5.5.2	Users	115
5.5.3	Measures	115
5.5.4	Study 2 Results	115
5.5.5	Study 2 Discussion	122
5.6	Study 3	122

5.6.1	Method	122
5.6.2	Results	123
5.6.3	Discussion	127
5.7	Overall Discussion	127
5.7.1	Limitations	128
5.7.2	Synthesizing Contradictory Results	129
5.7.3	Social Communication Theories to Support Transparency	130
5.7.4	Design Implications for Intelligent Systems	131
5.7.5	Presenting Errors in Intelligent Systems	132
6	How do users want to interact with transparency?	134
6.1	Introduction	134
6.1.1	Contribution	137
6.2	Research System: E-meter	137
6.2.1	Machine Learning Model	138
6.2.2	System Versions	140
6.3	Study 1	141
6.3.1	Method	142
6.3.2	Results	143
6.3.3	Discussion	148
6.4	Study 2	149
6.4.1	Method	149
6.4.2	Results	151
6.4.3	Discussion	154
6.5	Discussion and Conclusions	155
6.5.1	Meeting the Competing Needs of Transparency Through Progressive Disclosure	156
6.5.2	Impact for Future Transparency Research	160
6.5.3	Limitations	160
6.5.4	Conclusion	161
7	AI and Explanations in the Wild	162
7.1	Introduction	162
7.2	Methods	163
7.2.1	Predictive Analytics System	164
7.2.2	Participants	171
7.2.3	Materials	172
7.2.4	Interview Questions	173
7.3	Results	175
7.3.1	Advisors Initially Evaluate System By Carefully Examining Students with Different Characteristics	177
7.3.2	Transparency Can Polarize System Perceptions	178
7.3.3	Explanations Undermine a System Prediction	179

7.3.4	Progressive Disclosure Principles: Explaining Features	180
7.3.5	Advisors Have Difficulty Understanding Engineered Features in Explanations	181
7.3.6	Discrepancies Between Advisor and System Models of Student Performance	182
7.3.7	Institutional Constraints the System Does Not Know	183
7.3.8	Conflicting Explanation Goals	184
7.3.9	Ethical Concerns About Using Predictive Analytics	184
7.4	Discussion	185
7.4.1	Error and Expectation Violation	186
7.4.2	Systems Should Deliberately Walkthrough Users Initially	186
7.4.3	Explanation Should Be Interactive	187
7.4.4	Resolving Conflicts in Explanation Goals	189
7.4.5	Conclusion	190
8	Discussion	192
8.1	Summary	192
8.2	Open Problems	195
	Bibliography	200

List of Figures

3.1	EmotiCal System Components. The first screen shows the mood-forecasting component. The second and third screens show parts of the logging.	45
3.2	Users record which trigger activities influenced their current mood on this screen of the application	48
3.3	Coefficients of Trigger Activities Predicting Mood. Bars indicate the standard error of each coefficient	54
3.4	Category Model Predicting Mood. Bars indicate the standard error of each coefficient	55
3.5	Mean Scores for Activities from Different Clusters Showing Differences in Aggregate Ratings by Cluster	61
3.6	Cluster Distribution as Illustrated by the First Two Principle Components	61
4.1	Genre Representation in Full Track Set and Anomalous Track Set	81
4.2	Crowdsourced Utterance Generation for Individual Tracks	87
4.3	Transcription and Search Process to Resolve URIs	89
4.4	Track Aliasing Decision Process	90
4.5	Alias Finds Improvement Over Baseline	93
5.1	E-meter Document-Level Feedback Condition	106
5.2	E-meter Word-Level Feedback Condition	107
5.3	Participants Found the E-meter Accurate	116
5.4	Participants Found the E-meter Moderately Trustworthy	117
5.5	Expectation Violation and Transparency Condition Interact to Form Accuracy Perceptions	119
5.6	Transparency View Time By Explanation Position	125
5.7	User Improvement in Understanding By Explanation Position	126
7.1	Dashboard Showing Student Risk Prediction and View Detail Link to Explanation	167
7.2	Explanation View: The Predictive Analytics Explains Ratings by Classifying and Showing Predictors	169
7.3	Advisors Found Most Predictions Agreed With Their Own Assessments	176

List of Tables

3.1	Remedial Plans Created by Participants. These remedial activities were recommended by EmotiCal based on the user responding positively to this activity in the past.	46
3.2	Contribution of Different Types of Information to Predicting Mood	56
3.3	Unigram Correlations with Mood from User Explanations After Removing LIWC Words. Many of these unigrams point towards the importance of implicit emotion recognition in text.	58
4.1	Summary of Mixed-Effects Poisson Regression	94
5.1	Effects of Transparency on Perceived Accuracy. ($R^2 = .55, p < .0001$).	117
5.2	Effect of Expectation Violation on Transparency View Time	124

Abstract

Accurate, Fair, and Explainable:

Building Human-Centered AI

by

Aaron Springer

Artificially intelligent systems play an increasingly large role in our everyday lives. These intelligent systems make decisions big and small—from who gets a mortgage to what articles we read online. However, these systems are often designed out of convenience; they are built using data that is available, they do not explain the decisions they make, and sometimes they are simply inaccurate. At best, these problems result in systems that are difficult to use; at worst, these problems result in reinforcing societal biases at scale. Rather than designing systems from a standpoint of convenience, my work centers the human experience in the design of intelligent systems. I show how to work with users to build maximally accurate systems. I demonstrate that common intelligent systems are biased and not equally usable by everyone but also that this can be fixed through careful data collection. I show what users need when a system explains itself and how we can build our systems to explain the right thing at the right time. Finally, I move this work out of the lab and into the wild through a field study with expert users of an explainable AI system. Together, I use these studies to make recommendations that ensure we meet the needs of our human users in the intelligent systems that we build.

Acknowledgments

To my parents I owe an eternal debt. Even without four-year degrees, they instilled the value of education and an undergraduate degree in my siblings and me. Thanks to my mother, Eileen, for modeling a love of reading and curiosity from my first days. Thank you to my father, Roger, for demonstrating persistence and a work ethic in ways that I still strive to emulate. Thanks to them also for raising me within the larger Mennonite community whose tenants have guided much of this work.

To my advisor, Steve. It goes without saying that I wouldn't be here without your help. I appreciate how much my graduate school experience was tailored to *my* goals, something I think few graduate students can say.

To my siblings: Jesse, Rachel, and Naomi; I absolutely could not ask for a better bunch of bulldogs. To my Santa Cruz family: Will, Matt, Julia, Kate, Carmen; I cannot imagine completing grad school without such a rock-solid home life. To my lab mates Artie, Charlotte, Jeff, Victoria, Ryan, Lee, and Joel; thank you for being research role models and support structures. And to my other Santa Cruz essentials Dhanya, Morteza, Joe, Vincent, and the Laguna House.

And finally to my partner Giselle. Thank you for helping me slow down and take the time to celebrate milestones. I look forward to celebrating many more.

Chapter 1

Introduction

Machine learning algorithms power the many artificially intelligent systems we routinely use. Intelligent systems recommend our routes to work; they curate our entertainment options; they choose our most “engaging” friends to show us content from. While it may seem that we are all happy to defer to these systems—in fact, these intelligent systems face mounting criticisms about how they make decisions. These criticisms are exacerbated by recent machine learning advances like deep learning that power systems that are difficult to explain in human-comprehensible terms. Academics, policymakers, and increasingly the public are also concerned that these systems encode societal biases and then perpetuate these biases at scale [30]. Other groups are concerned about the accuracy of these systems, especially when making high-stakes predictions [226]. Together such concerns have led to calls for transparency and explainability; with researchers to argue that machine learning must be ‘interpretable by design’ [1] and that transparency is essential for the adoption of many intelligent systems, e.g. for medical diagnoses [83, 217].

Artificially intelligent systems have existed for many years but more recently these systems have become pervasive in ways they never were before. Traditional systems are created in a fashion where they follow the logic they were programmed with and do not change unless that logic is changed. Most intelligent systems are different, they are probabilistic and based on statistical models of the world. Intelligent systems often make predictions about real-world phenomena—and given that they are probabilistic, sometimes they are wrong. Intelligent systems also shift over time, often these systems are continually updated as they see new information; this means that such systems can display emergent behavior, both positive and negative. The fact that these new intelligent systems are probabilistic is a differentiating factor that we must take into account when designing new systems. However, this has been problematic because we cannot simply apply the traditional way of developing systems to these new intelligent systems. Designing intelligent systems is more complex than designing traditional systems. Whereas designers could work independently when creating traditional systems, creating intelligent systems require that designers closely collaborate with data scientists and domain experts to understand how decisions about the machine learning model affect the design of the system [222]. Rather than developing purely driven by design, designers must now balance the dual constraints of design-driven and data-driven development [222].

Unfortunately, it seems the balance here has shifted more towards data-driven development and these systems are increasingly being developed out of convenience. These convenient systems are based on data that is already available and they do not properly make themselves understandable to end users; this results in systems that can be biased, inaccurate, and opaque to end users. This thesis intends to explore these problems in creating human-centric

intelligent systems. I explore user perceptions of algorithmic accuracy and bias and suggest technical solutions to both these problems. I also examine the role of explainability and transparency in user perceptions of intelligent systems, addressing how, when and why to implement system explanations. Together, these documents lay out a list of considerations for future developers of intelligent systems; each topic that I cover can be a stumbling block to creating human-centric intelligent systems.

This problem of accuracy is a major issue that designers of intelligent systems must confront. In traditional programs, users trust that the program author has created the program to behave reliably. Furthermore, the way the program generates its outputs is usually relatively easy to comprehend. In intelligent systems, this situation is fundamentally changed, with the potential to undermine user trust. Intelligent systems are trained on often noisy data, their outputs are stochastic and they may generate errors. Furthermore, they may not explain themselves well; users can no longer simply trust that the creator defined benevolent behavior because intelligent systems have emergent properties, inaccuracies in prediction, and their outputs may vary as their underlying dataset changes. This may have been the case with the COMPAS program also—the program was not accurate enough even without the bias problems [226]. Thus another challenge in designing human-centered AI is improving the accuracy of the algorithms that power these systems. And given challenges in designing accurate algorithms, it may be important to set expectations about accuracy.

Major public concerns have arisen following demonstrations of bias in intelligent systems with regards to gender, race, and other characteristics. These algorithmic biases manifest themselves in many ways. In some cases, algorithmic bias only results in a poor user experi-

ence because a core feature such as voice recognition struggles with the user's way of speaking [189, 199]. In more extreme cases, biased algorithms result in life-changing decisions being made about indirect stakeholders. For example, COMPAS is a program that predicts the recidivism risk for prisoners and these predictions are used to inform decisions about parole made by judges. COMPAS was also shown to unfairly predict higher risk levels for black defendants, even when the defendants did not go on to re-offend [5]. With such high stakes, it is clear that we must design systems in ways that make it easy to understand how such biases or inaccuracy can affect predictions.

Transparency may be one way to deal with issues of bias and properly setting expectations for accuracy in intelligent systems. Algorithmic transparency has been called for in response to problems like COMPAS's biased predictions and other "Weapons of Math Destruction" [148]. Such calls have been made by academics, policymakers, and media outlets [83, 148, 217, 30]. Unfortunately, these calls for transparency have not clearly articulated exactly what it means for a system to make itself transparent. Nor has research properly examined what impacts transparent systems have on the user experience. These calls for transparency have only supplied the why of transparency, it is left up to system designers and researchers learn the when and how of transparency. In an effort to create intelligent systems that are more human-centered, I focus on these 3 major problems of accuracy, bias, and transparency. In particular, I focus on these problems in the context of end-user systems. That is, I do not design systems for programmers or data scientists to improve their algorithms or audit them for biases. Instead, I study how accuracy, bias, and transparency affect the end user experience and how we can generally improve the user experience of intelligent systems by pushing back against

the trend of convenient intelligent systems and instead centering the human in our design of artificially intelligent systems.

1.1 Contributions

1.1.1 Designing Accurate Personal Informatics Systems:

Chapter 3 focuses on how we should design intelligent systems to collect information that improves system accuracy. This work is conducted in the domain of personal informatics, specifically mood tracking and prediction applications. My prior work with EmotiCal indicates that trusting the algorithm's predictions as accurate led to more usage of predictive application features [92]. This implies that if we can design systems to be more accurate and trustworthy then this should positively impact the user experience. I show that we can derive highly accurate models of mood by combining information about the activities that users engage in and also their self-reflections upon those activities. I also demonstrate that some commonly used features, such as knowledge of which activities a user engaged in or historical mood measures, may not be as important as previously thought. Overall my work demonstrates that importance of designing mood tracking and prediction systems in which users actively reflect on the importance of the activities that they engage in. I suggest design implications for future systems based upon this finding.

1.1.2 Auditing and Removing Algorithmic Bias:

A major problem that has come to light in intelligent systems is the issue of algorithmic bias. Intelligent systems that are powered by machine learning have been shown to harbor biases learned from the data that the machine learning model was trained on; these biases can negatively impact end-users and also third-party users that the application makes decisions about. Chapter 4 addresses this problem in the context of a voice interface for a worldwide music application. I first demonstrate that common voice interface libraries do not treat all phrases and utterances equally. I show that these voice interfaces systemically underserve content from specific musical genres such as hip-hop and country. Previous work in linguistics and sociology indicates that this may be due to the specific and creative sociolinguistic practices of the people who create these genres of music. I then categorize the errors and biases that the voice interface makes so that future users of voice interface libraries can avoid these issues. Finally, I demonstrate how to fix these biases without opening the black box. To do this I crowdsource a diverse set of people pronouncing specific track names that are not well recognized by current voice interface libraries. I use these to create ‘aliases’; links between misrecognition of the voice interface and the true transcription itself. After deploying this process, I demonstrate a significant improvement in the ability of users to access this content. My work improves the fairness of voice interfaces for both the creators and consumers of content within this application.

1.1.3 When Do Users Want Transparency:

I contribute to the growing literature on algorithmic transparency through three user evaluations of a working intelligent system in the Personal Informatics domain. Previous re-

search on transparency and intelligibility has had highly mixed results. Positive system perceptions can be built through transparency [56, 105, 124], even to the point of overconfidence [69]. At the same time, however, positive system perceptions can also be undermined by transparency [61, 105, 123, 143]. My approach draws on psychological and sociological theories of communication applied to HCI [70, 81, 161, 198] to explain when, why, and how users want transparency. My findings reveal that transparency can have both positive and negative effects depending on the context. I present a model that shows how the context of transparency and expectation violation interact in forming user perceptions of system accuracy. Transparency information has positive effects both in helping users form initial working models of system operation and reassuring those who feel the system is operating in unexpected ways. At the same time, negative effects can arise when transparency reveals algorithmic errors that can undermine confidence in those who already have a coherent view of system operation. Finally, I verify how these self-report data match actual user behaviors. I show that greater expectation violation leads participants to spend more time exploring transparency information. Results again are consistent with my prior qualitative results suggesting that transparency helps to build initial mental models. We, therefore, find that users spend more time when first exposed to transparency information and that time with spent transparency declines over time as they see explanations again. I explain my results using theories of occasioned explanation [70, 81], arguing that transparency information is anomalous for users who feel the system is operating correctly and therefore undermines their confidence in the system. Design implications include a greater focus on what situations necessitate a transparent explanation as well as improved algorithmic error presentation.

1.1.4 How Do Users Want to Interact with Transparency:

Much recent work on transparency has focused on technical explorations of self-explanatory systems. In contrast, I take an empirical user-centric approach to better understand how to design transparent systems. Two studies provide novel data concerning user reactions to systems offering transparent feedback vs overall prediction information. In Study 1, users anticipated that a transparent system would perform better, but retracted this evaluation after the experience with the system. Qualitative data suggest this may be because incremental transparency feedback is distracting, potentially undermining simple heuristics users form of system operation. Study 2 explored these effects in more detail suggesting that users may benefit from simplified feedback that hides potential system errors and assists users in building working heuristics about system operation. I use this data to motivate new progressive disclosure principles for presenting transparency in intelligent systems.

1.1.5 AI and Explanations in the Wild:

Finally, I move beyond the focus on toy systems in the current literature on explanation in intelligent systems. I examine a high-stakes deployed intelligent system that makes and explains predictions about student success. I find that many of the lessons learned from my previous work on explanation transfer to this real system. However, the context and complexity of real systems provide some further challenges regarding how advisors can understand the system and whether or not they believe it should be used. I derive design implications regarding how users should initially interact with intelligent systems and also how these systems should construct interactive explanation in order to become more human-centered.

Chapter 2

Literature Review

Interactions with intelligent systems are fundamentally different from traditional user interfaces. This is due to the complex machine learning capabilities that power these systems, leading intelligent systems to have new dimensions that we need to explore. This includes accuracy, bias, and explanation.

Accuracy is important because machine learning models are probabilistic—they are not always correct. For example, a speech system may misinterpret user inputs or a facial recognition system misidentify a person. The possibility of such errors means that systems must be prepared to recover from these errors and help users understand the likelihood of the system being correct in the future. Some work has tackled this through displaying confidence levels in a prediction [123], others have given users the ability to correct such errors [108].

Emergent biases are new to intelligent systems also. For example, systems may reflect biases in training data, leading them to misclassify cases that are infrequent in that training set [9]. While some scholars [67] note that system biases extend back to the 90s, the systems

they reference were hardcoded to favor certain organizations over others. Emergent biases, where intelligent systems unintentionally embody human biases are relatively new. These biases may disadvantage large groups of users because the application developers did not design their models with human-centricity in mind.

Finally, transparency is of growing importance as intelligent systems increasingly make high-impact policy decisions about people lives, involving education [58], recidivism [5], and health [1]. Awareness of errors and bias mean that users are increasingly skeptical about intelligent systems. Without the ability for intelligent systems to explain such high-impact decisions, users are often unsure whether they should trust the system prediction. These issues have led researchers to argue that machine learning must be ‘interpretable by design’ [1] and that transparency is essential for the adoption of many intelligent systems, e.g. for medical diagnoses [83, 217].

These emerging problems with intelligent systems potentially compromise the ability for developers to create human-centric systems. Traditionally designers have integrated the needs found from user-research and product goals in order to design new products that adhere to well understood GUI design principles. However, the advent of machine learning in intelligent systems means new approaches have to be found [4]. Designers are often unaware of how to use machine learning as a “design material” within their work [222], this leads to the creation of features and systems that are driven by engineers rather than researchers and system designers. So far this has resulted in less human-centered intelligent systems.

I first review solutions to these problems arising with current intelligent systems. Such solutions are essential to creating human-centric intelligent systems, these include: designing

systems to collect information that allows them to be maximally accurate, countering algorithmic bias, and creating explainable and transparent intelligent systems. I next review 3 domains in which intelligent algorithms are regularly used: voice interfaces, emotional analytics, and learning analytics. I explore the deployment of intelligent systems in the context of these three domains.

2.1 New Aspects of User Experience in Intelligent Systems

2.1.1 Accuracy

Trust is important in predicting acceptance of new technology especially in domains that contain sensitive information such as banking. In these domains, trust is highly predictive of intentions to use such systems. This is true of digital banking [52] and electronic health records usage by physicians is also highly predicted by trust [149]. A meta-analysis examining the integration of trust into the Technology Acceptance Model [46] found that the success of an application “depends not only on the benefits which it brings to the users but also on the level of trust which users have during the system’s usage” [220]. In many studies trust is construed as safety and integrity of data integrity, but intelligent systems have introduced a new concept that highly influences trust: accuracy.

It is essential to clarify how we use “accuracy” in research because it comes in many forms. On one hand, we have the accuracy of a machine learning model that we will refer to as ‘system accuracy’. For example, in a classification task, we can directly measure the accuracy of a predictive model as a percentage of how many examples it classifies correctly in a test set.

Assuming that the domain the predictive model is predicting in has not distributionally drifted from the test set, this measure of accuracy serves as ground truth for the systems actual capability. In other systems that have unbalanced classes or are predicting continuous outcomes, we may use other appropriate metrics such as precision/recall, R^2 , or Root Mean Squared Error (RMSE); these all broadly characterize the performance of the system and to fit with existing literature we simply group these under system accuracy. While system accuracy is very important, this ground truth is not always available to users. From the user's perspective, we have their concept of accuracy known as 'perceived accuracy'. Perceived accuracy is the user's judgment of how accurate the system is from their usage of the system. Perceived accuracy is influenced by 'system accuracy' but is also influenced by the heuristics and biases than humans use to judge systems. For example, users who see multiple errors in a row from a system will judge it much more harshly than if the errors were separated, even if the system accuracy is the same overall [227]. This distinction is critical—as we will see perceived accuracy can be influenced by much more than just the system accuracy.

As we begin to embed machine learning models in more systems, accuracy becomes an essential factor in creating human-centric intelligent systems. System accuracy is highly related to user trust in the system [227], which is an essential part of the user experience in intelligent systems. Additionally, empirical results suggest a threshold effect—a certain level of system accuracy is required to maintain trust in an intelligent system [227] otherwise trust declines over time. Other work shows that simply stating the tested accuracy of a specific model is not enough to ensure trust. One study provided users measures of the system's accuracy and then asked them to complete tasks to build a measure of perceived accuracy. The authors con-

cluded that users calibrate provided information, such as system accuracy, against their own subjective experience using the system and perceived accuracy in their own instances [225]. Both these results paint a complex picture of relations between trust and accuracy. They show that it is not simply the externally validated accuracy percentage that matters but also characteristics of the user experience such as early experiences and sequences of correct and incorrect system decisions.

High accuracy is essential for usable systems in many domains. For example, in voice recognition and voice transcription systems, accuracy is of utmost importance. Applications that recognize spoken digits for secure applications must have very low error rates [167]. And people who use transcribed voicemails for their tasks perform more quickly with accurate transcripts and consider them more readable and comprehensible [195]. Likewise, users are less likely to continue using the system if the transcripts they receive are inaccurate.

Accuracy interacts with other user experience features in intelligent systems. Low accuracy may impair other aspects of the user experience. Lim and Dey [123] explored the benefits of providing system explanations when system outputs were highly accurate versus inaccurate. Counter to their expectations, system explanations had negative effects on user experience. The authors of this study found that explanations can provoke doubt from users even when the overall prediction is correct. In recommender systems, accuracy is one of the most studied parts of the user experience, forming an essential construct in the modeling of user experience in recommender systems [134, 31]. However, this work also acknowledges that other aspects of intelligent systems such as transparency and serendipity are also important, and interact with accuracy in complex ways. Focusing on accuracy too much can come at

the expense of the user experience because accuracy is naturally a reduction of specific more complex interactions.

One important note is that different domains may have different levels of acceptability of prediction accuracy. While in a domain such as weather prediction, a user may consider 80% to be an acceptable level; users in another domain such as electricity usage forecasting may consider 80% to be below the usable threshold. Interestingly this same work found that users may prefer different models based on the false-positive versus false-negative distribution [103]. It's clear that accuracy and associated metrics like false-positive/false-negative ratio are essential considerations for a positive user experience within intelligent systems.

2.1.2 Transparency

Transparency has become a central issue for intelligent systems, with recent guidelines published by the EU that set explainable and transparent systems as a core requirement [77]. I review the problems that stem from a lack of transparency, I review how transparency affects the user experience, finally, I review social science theory that suggests how we should structure transparency and explanation.

2.1.2.1 Folk Theories of Algorithms

A wealth of prior work has explored issues surrounding algorithm transparency in the commercial deployments of systems for social media and news curation. Social media feeds are often curated by algorithms that may be invisible to users (e.g., Facebook, Twitter, LinkedIn). At one point, many users were unaware that Facebook newsfeeds filtered the posts that their

friends made [62]. These users reacted in surprise and sometimes anger when they were shown the filtered posts that were ‘missing’ from their newsfeed. Later research showed that many Facebook users develop ‘folk theories of their social feed [60], which are imprecise heuristics about how the system works, even going so far as to make concrete plans based upon their folk theories. This work also showed that making the design more transparent or seamful, allowed users to generate multiple folk theories and more readily compare and contrast between them [60].

Other work has illustrated issues regarding incorrect folk theories in the domain of intelligent personal informatics systems, showing specific challenges in how users understand these systems. Users are sometimes susceptible to blindly believing outputs from algorithmic systems, a phenomena referred to as algorithmic omniscience [61, 94, 190] and automation bias [42, 141]. For example, KnowMe [213] is a program that infers personality traits from a user’s posts on social media based on the Big Five personality theory. KnowMe users were quick to defer to algorithmic judgment about their own personalities, stating that the algorithm is likely to have greater credibility than their own personal statements (e.g., “...At the end of the day, that’s who the system says I am...”). Similar results were shown in [94]; participants expected intelligent personal informatics systems to serve as ground truth for their experiences and even attributed superhuman qualities to these devices, e.g., “...[it] could tell me about an emotion I don’t know that I am feeling...”. Other experiments indicate the risk of such trust, showing that users may believe even entirely random system outputs are moderately accurate [190]. Similarly, giving users placebo controls over an algorithmic interface shows corroborating results [207]; users with placebo controls felt more satisfied with their newsfeed. Similar placebo ef-

fects were found for heart rate; participants who were given placebo reports of lowered heart rate performed better than those given honest feedback [39]. Without a standard of transparency in intelligent systems, it may be possible to deceive end-users into believing false algorithmic outputs; this is a dangerous proposition when apps can be so easily distributed. However, the opposite reaction here seems to be possible also. One study indicates that users may not believe systems that measure and provide information about their own weight [102]. Users of these weighing systems felt that the fluctuations in measurements were too high, when they were actually within the bounds of what is expected. In a case like this, the authors conclude that providing more information about how the system is working and what is considered normal may allow users to be more accepting of the system.

2.1.2.2 Transparency Effects on User Experience

There is a long history of studying transparency and intelligibility in automated systems [13]. However, the results often indicate contradictory effects on user perceptions. Many experiments have indicated that transparency improves user perceptions of the system [56, 124]. Others have shown that interventions that simply showing the system's prediction confidence improve users system perceptions [6]. In extreme cases, animations that simulate transparency can cause users to be overconfident about systems even when they err [69].

Other studies show less positive effects of transparency for user perceptions. Exposing users to a transparent system using descriptive scenarios can lead users to question the system, resulting in reduced agreement with the system [123]. However, the effect may be the opposite for high certainty systems—transparency only increases user agreement. Older work

concludes that any hint of error in an automated system will decrease trust [143]. More recent work indicates other effects; explanations of how a system is working may increase trust [105], but explanations that include too much information may be harmful to user trust. However, these undermining effects are dependent upon the amount of expectation violation that a user experiences. User expectation violation occurs when the system makes a prediction that a user did not expect [105, 207]. Ideally, transparency should build user confidence in a system, whether or not the user is experiencing expectation violation. However, it may be that research communities have yet to find the correct interaction paradigms to achieve this.

Recently, the machine learning community has begun grappling with issues of transparency. This may be due to the rise of more inscrutable methods like deep learning. Some machine learning models are more straightforward to understand such as linear models and Generalized Additive Models [128, 216]. These understandable models can be “explained” to users as the simple linear contributions of their features. Other algorithms such as deep neural nets and random forests are inscrutable, as they may involve many layers of hidden features along with complex tuning parameters, making it difficult to explain how input features match to output predictions [216]. For example, seemingly random noise that is unnoticeable to humans can cause deep neural network image classifiers to fail catastrophically [76]. Various approaches aim to make these inscrutable algorithms understandable. These approaches often rely on approximating the inscrutable algorithm through a simple local or linear model that can be explained to the end user [129, 173]. However, even with these methods that attempt to render complex algorithms into more understandable models, there is no clear consensus on how to convey these models to users in a comprehensible way. Furthermore, many such attempts at

transparency are not validated; they have not been tested with real users or they simulate user behaviors [7, 139, 144]. The absence of real user feedback makes it challenging to operationalize transparency in ways that positively impact users.

There have been some recent attempts to draw general guidelines for explanation and broadly human-AI interaction, but they have approached the problem differently. One group analyzed the literature on this topic to derive 18 principles of good human-AI interaction design and then tested these principles by having designers evaluate products [4]. Transparent reasoning and explanations were included in these 18 principles as essential to successful human-AI interaction. Another group drew from the psychological literature on human reasoning and attempted to construct guidelines for systems that use explanations to counter human biases. They applied these guidelines in co-design sessions with clinicians and derived a number of design implications including allowing for hypothesis generation, integrating multiple explanations, and supporting coherent factors [211].

2.1.2.3 Explanation and Persuasion Theory

People interact with computers and intelligent systems in ways that mirror how they interact with other people [146, 171]. Given that transparency is essentially an explanation of why a model made a given prediction, we can turn to fields such as psychology and sociology for guidance about operationalizing explanations. These fields have a long history of studying explanation. One common analysis in philosophy and psychology argues that explanations involve drawing contrasts between facts (what actually happened) and foils (what was expected to happen). Users may then explore counterfactuals, i.e. what needs to be true for a currently

unmet expectation to occur (Lipton, Miller). Similar arguments have been made in sociology that explanations are ‘occasioned’, and should only be produced when there is a discrepancy between the current situation and what is expected [70]. Another approach is to model causal explanation as a form of conversation which is governed by common-sense conversational rules [91] such as Grice’s maxims [81]. In addition, when an explanation is needed and a communication breakdown occurs this is remedied by a phenomenon known as conversational repair. Conversational repair is interactional, participants in the conversation collaborate to achieve mutual understanding; this often happens in a turn-by-turn structure with repeated questions and clarifications [183]. These theories indicate that we should operationalize transparency in ways that fit human communication and repair strategies.

Additionally, there are parallels between how people interact with intelligent systems and theories of persuasion. The Elaboration Likelihood Model (ELM) is a dual process model of persuasion [161]. The ELM posits that two parallel processes are engaged when a person evaluates an argument, similar to Kahneman’s conception of System 1 and 2 thinking [99]. The central processing route involves careful consideration of the argument and complex integration into a person’s prior beliefs. The central route is often engaged in high stakes decisions. The peripheral route, in contrast, focuses on heuristic cues such as the attractiveness or status of the speaker, the listener’s current affect, the number and length of the arguments, and other cues not directly related to the content of the argument. Prior work on intelligent systems seems to align with this dual process model [105], people understand systems through peripheral routes if their expectations are met, only engaging in central processing when their expectations are violated. This is also demonstrated in the context of Google search suggestions; where users

can feel the cost of processing explanations outweighs their benefits [24]. Transparency needs to be operationalized in ways that allow users to understand transparency through both cursory heuristic routes and also through focused reflection.

2.1.3 Algorithmic Bias and Machine Learning Fairness

Algorithmic bias and machine learning fairness have recently come to public attention due to high profile incidents like COMPAS' negative predictions about black defendants [5]. Broadly machine learning fairness strives to address bias by creating algorithms that both treat people equally and are perceived as fair by their users. Deploying fair algorithms is essential for creating human-centered intelligent systems. Unfairness can lead to everything from unfair high stakes decisions about defendant parole [5] to poor user experiences due to a lack of trust and perceived unfairness (see big data). When users perceive an algorithm as unfair they may behave in ways that the system does not intend them to [118]. Other instances of unfairness such as unfairness in voice systems can diminish user acceptance of such systems.

Friedman and Nissenbaum define computational bias as “computer systems that systematically and unfairly discriminate against certain individuals or groups of individuals in favor of others” [67]. In addition, Friedman and Nissenbaum present a taxonomy of biases in computational systems with top-level categories of Preexisting Bias, Technical Bias, and Emergent Bias. While Friedman and Nissenbaum's work was prescient in many ways, it is difficult to use this taxonomy to address algorithmic and data bias problems in practice. Their categorization also does not point to underlying causes or solutions.

More recent taxonomies of algorithmic and data bias exist and allow us to classify

problems in ways that suggest how to intervene and correct biases. One helpful taxonomy for classifying algorithmic and data bias is presented by Ricardo Baeza-Yates [9]. The taxonomy consists of 6 types of bias: activity bias, data bias, sampling bias, algorithm bias, interface bias, and self-selection bias. These biases form a directed cyclic graph, where each bias feeds biased data into the next stage where more bias is introduced. While the cyclical nature of bias seems disheartening, this taxonomy also provides us with ways to intervene in the bias cycle. For example, voice interfaces often struggle with strong regional accents [199]. This deficiency can be classified into the above taxonomy which allows us to suggest corrective actions. The inaccuracy with strong accents could be due to a data bias; the company trained ASR models on data that did not include such accents and thus needs to collect that data. It could also be due to sampling bias for the algorithm, a sample over-representing unaccented voices could be corrected by using different stratified samplings for training data. The bias taxonomy from Baeza-Yates provides a common language to discuss problems with bias.

2.1.3.1 Algorithmic Biases In Voice

While biases in voice technologies are modality and domain-specific, they illustrate many of the common problems that larger algorithmic bias efforts must deal with. As such, we include this review of algorithmic bias in voice systems here and later review more broadly the topic of voice interfaces and the specific domain our later work will fall into: language usage in music. Automatic speech recognition systems may exhibit biases with regards to different voices and use of language. Tatman shows that different English dialects result in significantly different accuracy in automatic captions [199]. Other work shows that current nat-

ural language processing systems perform poorly with African-American English compared to Standard American English [17, 98]. This may mean that music creators who use their dialects when titling their compositions may cause their materials to be less accessible than those using more standard titles. Recent initiatives to create more open and diverse speech datasets for voice recognition models have yet to bear fruit [142]; nor will these efforts cover every domain. Many voice applications also re-use training data from applications in other modalities. For example, voice web search will at least partially exploit text web search data. However, voice queries are longer and closer to natural language than typed queries [84] so that voice naming may be atypical of long-form training text.

2.1.3.2 Solutions to Voice Biases and Challenges

As we explored with Baeza-Yates, there are many different sources of biases. These different sources lead to different approaches to overcoming bias, which we apply to voice interfaces. In voice interfaces, correcting biases can range from approaches focused on the interaction model itself to those dealing with the underlying data and algorithms. One approach is detecting when a user is having speech recognition problems and automatically adjusting the voice dialogue itself, e.g. switching from an open to a closed form questioning style. Other approaches may combine voice recognition with on-screen input. Goto et al. demonstrate this, showing options on-screen in response to uncertain voice commands [78]. However, these approaches do not necessarily solve problems where the training data itself is impoverished and particular content is inaccessible. We focus on the identification of inaccessible content and solutions through data collection.

One aspect of inaccessible content identification is common in ASR: recognizing Out-Of-Vocabulary (OOV) terms. Multiple ways are available to detect and deal with out of vocabulary terms. Parada et al. [151] describe multiple ways to deal with (OOV) terms. The first method, filler models, represents unrecognized terms using fillers, sub-words or generic word models. The second method uses confidence estimation scores to find unreliable OOV regions [88]. The third and final method Parada et al. describe uses the local lexical context of a transcription region. Other approaches model the pronunciation of OOV terms, see Can et al. [28]. Alternatively, Parada et al. [152] describe how, after OOV regions have been detected in transcriptions, they use the lexical context to query the web and retrieve related content from which they then derive OOV terms, often names. The above methods for recognizing OOV terms often assume that the developer has built a specialized ASR from the ground up and can modify it however they choose. With the advent of large-scale public ASR APIs, this assumption may no longer be true. In addition, bias studies find categories of problems that would exist even with a perfect vocabulary. For example, ‘100it racks’ is a creative way of spelling ‘hunnit racks’ which is slang for ‘hundred racks’, even a perfect vocabulary in an ASR could at best recognize ‘hunnit racks’ which still leaves a needed bridge between this recognition and the actual song title ‘100it racks’. Crowdsourcing is a relatively common part of dealing with ASR problems. Data collection through crowdsourcing can be used to learn pronunciations for named entities [181], and similar work exists for the generation of search aliases [33]. Ali et al. [3] describe the challenge in evaluating ASR output for languages without standard orthographic representation; where no canonical spelling exists. They use crowdsourced transcriptions to evaluate performance for Dialectal Arabic ASRs. Granell and Martínez-Hinarejos [80] use crowdsourcing to collect

spoken utterances to help transcribe handwritten documents, combining speech and text image transcription.

However, before such processes can be applied, we first need to assess potential problems occurring within a domain, and their prevalence. We can no longer assume voice application builders develop their own ASR nor that they have access to the internals of their ASR service; this creates challenges to correcting ASR errors that require pragmatic solutions. It may be that there is less accessible content that accesses cultural and linguistic practices that are not well-supported by current speech solutions. Most ASR systems rely on a language model that prioritizes high-probability word sequences over less likely utterances. These probabilities are trained from frequencies of word n-grams in corpora [169]. Probability of co-occurrence is also a significant predictor of ASR error [74]. For example, a popular track, at the time of writing, is "Two High" by Moon Taxi. The name of this song is pronounced [tu har] and can correspond to three possible English strings: "to high", "too high" or "two high". Of these strings, "too high" is the most statistically likely, and so when a user asks for the sound string [tu har], a generalized ASR system's language model is more likely to return the written string "Too high." This can be problematic for named content, creating confusion if the user is asking for a current popular track "Two High" or an older popular track like "Too High" by Stevie Wonder. While some ASR APIs accept custom language models and pronunciation dictionaries, these are usually quite limited. The additional vocabulary words still need to be detected, generated and supplied with a probability. This is especially an issue for domains where creative language usage is valued (e.g. music, art), or systems used by audiences with diverse linguistic backgrounds. These errors may require downstream solutions if the developers use off-the-shelf APIs where

language models are not directly modifiable.

2.2 Domains

Machine learning is now being integrated into many of the systems that we traditionally use (e.g. spelling correction, email composition, product recommendation) but also creating whole new classes of systems. In order to create more generalizable guidelines about creating human-centric intelligent systems, I review 3 different domains in which intelligent systems are used. The first of these is Emotional Analytics, an important ‘quantified self’ domain in which users track their mood and emotions and use intelligent systems that make predictions and help users gain self-knowledge about their emotions and mood. The next domain is voice interfaces, a major benefactor of the recent increases in accuracy in intelligent systems. Voice interfaces are important, as they are now used by over 46% of United States adults [147]. The final domain I examine is an example of high-stakes usage of intelligent systems. Learning analytics are increasingly used by higher education institutions to identify and allocate support to students who are diagnosed as underperforming. Learning analytics may, therefore, have significant effects on students’ success at college. The fact that they inform funding and support decisions mean that learning analytics systems may have different requirements around trust and explanation. I review the 3 aforementioned domains below in order to contextualize the work I later present.

2.2.1 Emotional Analytics

Chapters 3,5, and 6 focus is on how users interact with intelligent personal informatics (PI) systems. These systems are being increasingly deployed in commercial [219, 163] and research domains [14, 68, 92, 133, 230]. PI systems track how a person behaves on some dimension, whether physical, emotional, or mental, and may suggest improvements to this behavior through customized feedback and recommendations [92, 166]. Such personally relevant data potentially allows users to analyze and modify their behaviors to promote well-being [34, 35, 95]. My main focus will be on emotional analytics which has been identified by users as one of the key applications of PI technology [94].

Emotional Analytics is a subdomain of personal informatics (PI). There are many user experience challenges within PI, such as how to present negative feedback to users if they are not living up to the goals they have set [125]. With PI systems there is also a delicate balance about how much active support for sensemaking the system should provide. Some approaches have left the interpretation of data and identification of target behaviors that might be changed in the hands of the user. For example in Echo, the system provides support for recording and then actively reflecting on their past behaviors. However, the user is then left to derive insights to make changes and goals outside the system. Other systems attempt to ease this sensemaking burden by drawing correlations and connections between different behaviors and states that the user records [14]; however, the burden is still on the user to translate these connections into actionable behaviors and goals. Finally, some newer systems provide active sensemaking for the user; they internally model the user state and then use this to make recommendations

about behaviors and goals that the user should engage in [92, 166]. As we further devolve sensemaking to these PI apps, it is important that we implement these intelligent systems in ways that are human-centric.

However, there is clearly more work to be done in this realm. One example of an intelligent system used for emotional analytics is Monarca. Monarca uses both passive sensing of activity and self-report from users to predict and forecast mood. This allows patients and their caregivers to examine these relationships and form insights into what affects the patient's mood [54]. The system was deployed for 6 months with a small group of 6 patients and impressions were recorded [68].

Monarca exhibits the complexity of creating intelligent emotional analytics systems in a human-centered way. Physicians who used the system did not seem to trust the forecasts the system created for users [54]. Physicians were not sure how seriously to consider the forecast, whether to use it to change medication in advance of symptoms or just to use it as another indicator. Generally, the physicians ended up relying on their own clinical experience rather than using the mood forecasting for their patients. Patients themselves enjoyed seeing objective data about their lives but sometimes doubted the accuracy of the measurements. These results point to the need for further work about creating human-centered emotional analytics systems.

2.2.1.1 Prior Approaches to Modeling Mood

Important for my later research in emotional analytics is mood. A mood is a generalized emotional state that is typically described as having either a positive or negative valence. Moods tend to be longer lasting than discrete emotions such as anger, joy or fear. Moods are

less specific, less intense, and less likely to be triggered by a particular stimulus than discrete emotions [114]. Energy is a different concept that relates to the strength of the experienced emotion [179, 20], and is commonly synonymous with arousal. Mood and energy interact in complex ways, although there is no consensus about the exact nature of such interactions [179, 164, 200]. Theoretical and empirical work indicates that reporting both mood and energy together improves the accuracy of mood evaluations [179].

2.2.1.2 Relationships Between Activities, Mood and Wellbeing

Later I will examine how to create accurate models to predict mood from user activities. However, it is important that we understand how this has been accomplished prior to computational systems and personal informatics. Few studies have modeled how daily personal activities affect mood; most of these have focused on non-digital contexts. Daily personal activities have been examined by Stone et al. [196] who developed the Day Reconstruction Method (DRM). The DRM involved participants writing both about the activities they engaged in during the previous day, along with their accompanying feelings. Activities with highly positive effects on mood included exercising, socializing, intimate relations, and relaxing. Activities that engendered negative moods included doing housework, commuting, and working. However, other factors that are unrelated to specific activities also affected general mood levels, including pressure to work quickly and sleep quality. Another study found that people are prone to inaccurately evaluate how normal activities affect mood [218]. People in that study were more likely to rely on folk theories about mood (e.g., Fridays are happier days, Wednesdays are unhappy) than their own direct personal experiences.

A very different approach to understanding and improving mood is the Pleasant Events Schedule (PES). The PES explores the effects of different activities on mood using a combination of retrospective reporting and intervention methods. The PES is a behavioral self-report inventory that focuses on positive events in participants' lives [132]. In the PES users are asked to retrospectively report how frequently they engage in various activities and also to rate the 'pleasantness' of each activity. For example, seeing old friends is reliably judged as a highly pleasant activity, whereas physical discomfort is judged to be highly unpleasant [119]. While participants' subjective evaluations provide useful information about how different activities affect mood, these evaluations can take place as long as a month after the activity took place. This delayed evaluation may be problematic, as we know that emotional appraisals of activities and events change over time; current evaluations are not always reliable indicators of how people feel immediately after an activity [95, 175, 210].

2.2.1.3 Systems For Tracking Mood That Aim To Change Behavior

Until recently, most PI systems for wellbeing focused on physical health, for example, tracking goals relating to weight loss by documenting food consumption or daily step count. The implicit assumption behind these systems is that simply capturing and presenting detailed records of physical behavior will be sufficient to allow users to change target behaviors to meet goals. This approach has not always been successful. First, data about daily activities is often complex, requiring users to interpret multiple streams of time-varying data. Second, much of the general population has low data numeracy skills, creating challenges for end-users when making sense of such complex data, and making it critical to design effective end-user analytics

[160]. A further limitation of basic tracking systems is that they are overly rational; ignoring both users' affective reactions to their logged data as well as their emotional motivations for tracking and changing behavior [41]. It is clear that behavior change applications must facilitate better sensemaking for personal data.

In addition to systems that promote physical wellbeing, there are now more systems that primarily focus on emotional wellbeing. Echo was one of the first systems that addressed emotional reflection for psychological wellbeing [95, 106]. Echo is a digital diary in which users post about recent experiences as well as their current mood, writing text to explain their mood evaluation. Echo deployments have shown that both posting and reflection upon prior posts improve emotional wellbeing. Furthermore, the effects of Echo are long-lasting with improvements still evident after 4 months [106]. Other work has explored mood-dependent reflection, showing that reflection on positive memories can elevate a current negative mood, and that reflecting on negative events when in a positive mood leads those prior negative events to be more positively viewed [106]. Another reflection application, Pensieve, used a different design approach; emotional reflection was encouraged by having users reflect on prior social media content or respond to targeted prompts [156].

Early systems mainly supported manual text entry about mood. Recently, however, we have seen the emergence of new automatic sensing applications that aim to model mood based on a variety of automatically detected behaviors. MoodScope analyzes phone usage patterns such as the number of SMS messages, application usage, call frequency, and call length to infer mood [121]. BeWell is another application that incorporates passive sensing to detect physical, social, and sleep activities. BeWell blurs the line between physical and mental well-

being in an attempt to create a holistic system; it aims to explore how different activities affect mood as well as physical wellbeing. However, experimental trials to validate the models underlying BeWell are lacking [113]. Furthermore, there may be major limitations to approaches that involve passive data collection. One of the major questions we will examine in Chapter 3 is the role of active personal data recording compared to more passive approaches such as BeWell and MoodScope.

These research systems purport to support end-user sensemaking by providing correlations between data streams, e.g. relations between mood and sleep, but a different approach has been taken in MONARCA [11, 54, 63]. MONARCA is focused on emotional analytics; it allows bipolar patients to actively track activities and mood to better understand how trigger activities affect manic or depressive components of bipolar disorder. For example, a patient might experience more volatile moods if they skip medication, fail to exercise, or sleep poorly. Unlike many of the prior systems, MONARCA was deployed to a targeted clinical population to explicitly test the effects of such analytic support. Seventy-eight participants using the monitoring-only version of MONARCA showed no significant improvements and even a tendency for more depressive symptoms compared to a control group [63]. While an ongoing trial is exploring improved analytic support, this MONARCA evaluation highlights the need for additional work on emotional analytics and a greater exploration of possible benefits for nonclinical users.

2.2.2 Learning Analytics

Following the review of learning analytics undertaken by van Barneveld et al. we center our work specifically around the usage of predictive systems for academic analytics [208]. These are systems that deal with “extracting information using various technologies to uncover relationships and patterns within large volumes of data that can be used to predict behavior and events”. While this narrows the field of academic analytics quite a bit, predictive analytics can still mean many things within the field. Predictive analytics encompass many different systems including: early alert systems that predict which students are at risk of attrition, enrollment management systems that predict future enrollment and return class sizes, adaptive learning to create online course modules that personalize themselves to students, and many other intelligent systems [58]. These systems are increasingly deployed; over 70% of surveyed universities in 2018 had already implemented or planned to implement predictive analytics on their campus [59]. Campuses deploy these predictive analytics to save money through better student retention and more accurately predicted future class sizes ([58]. However, advisors and other practitioners on these campuses are now raising concerns about the usage of predictive analytics in higher education.

Higher education practitioners are concerned that predictive analytics may impact the students on their campus in negative ways. Practitioners are concerned that predictive analytics systems will discriminate, label, and stigmatize certain students [58]. Part of this concern stems from the ability of predictive analytics to embody societal biases and unintentionally reinforce them [5]. Another related concern is labeling and stigmatizing; there is a fear that labeling a stu-

dent as being high risk will unintentionally bias their advisor into treating them differently and thus create a self-fulfilling prophecy, known within education and sociology as the Pygmalion effect [176, 204]. There are fears about transparency of these systems also. Transparency here can mean two different things: the transparency of data collection to students and the transparency of the system predictions to advisors [58]. Even though some companies implement explanations within their systems, it remains to be seen if these are constructed in a way that improves the user experience.

It is important to note that none of these studies have directly examined whether advisors want to use predictive analytics nor how advisors use these systems. There is a lack of real evaluations in the literature with working predictive analytics implementations in education. We intend to fill this gap through our study in Chapter 7.

2.2.3 Voice Interfaces

Voice interfaces are computational systems where the primary modality of interaction happens through speaking. Voice interfaces became relatively mainstream in the 2000s, where people could call specific telephone numbers and interact with a voice interface in order to do specific tasks such as booking a flight or checking a schedule [45]. However, these voice user interfaces were limited to very specific tasks and were not available for general purpose or more conversational usage. We have recently come into an “Era of Digital Assistants” where many general purpose devices have digital assistant voice interfaces embedded within them [45]. In fact, recent studies have noted that almost half of United States adults interact with a digital assistant each day [147]. I briefly review the structure and technology underlying voice

interfaces in order to later examine how voice interfaces exhibit algorithmic bias and what we can do to fix them.

Voice interfaces rely on ASR (automatic speech recognition) systems to enable interaction. While different approaches exist, some recent deep-learning ASR systems for example directly map audio to characters, ASR systems are often made up of three different models [169]. The first, the acoustic model, translates the spoken language into distinct units of sound called phonemes; sounds that make up a language [111]. These phonemes are then mapped to words using the second model, the lexicon. For example, the English word “cat” has three phonemes: a [k], [æ], and [t], transcribed together as /kæt/, the lexicon would associate these sounds back to the word “cat”. Finally, these words are evaluated and changed according to the language model which is a probability distribution over word sequences.

Speech recognition has recently improved to the point where claims of human parity—in standard speech evaluation tasks—are beginning to surface [221]. However, this does not mean that ASR is a solved problem. Specific types of commands and words can still be hard to recognize. Each ASR technique, from neural networks to HMMs, comes with specific strengths and weaknesses [43]. Difficulties can be created by factors like disfluencies, repetitions, extreme prosodic values (e.g. pitch), and pairs of similar sounding words (e.g. ask/asked, says/said); regional accents and individual differences in pronunciations present additional problems [15]. Specific domains come with their own problems and potential consequences; see Henton’s discussion of ASR problems in recognizing medical terms in patient report dictation [90].

2.2.3.1 Creative Online Language Usage

Chapter 4 examines how these voice systems can be biased against users because of the users specific ways of speaking and creative choice of language. Here I review use of language that pushes the bounds of traditional speaking and writing and also how music and language are inherently tied.

Non-standard usage of language and symbols is a common practice when communicating. Text messaging does not always follow standard spelling and grammar [202]. Features like emojis are used in variety of functions ranging from adding shared meaning to making an interaction more engaging, complementing or even replacing text [40]. Similarly, online l33tsp34k, replaces letters with digits or other ASCII symbols, and has been around for decades. Even with minimal exposure, people are readily able to translate words in their ‘l33t form’ [159]. While the practice in art dates to at least the 1920’s (see e.g. the Dada poem w88888888 [185]), l33tsp34k’s origins aimed to make content harder to automatically process. This allowed to circumvent filtering of ‘forbidden words’ [159].

2.2.3.2 Language and Music

Language and music have a complex, intertwined relationship. Verbal language is integral in many types of music, music itself can be seen as language, and specific language is used to describe music; each of these constitutes its own whole field of study [64]. People use language to indicate belonging to specific social and cultural groups. Focusing on a Texas country community, Fox [66] discusses music as a identity preservation tool, emphasizing the importance of preserving linguistic forms, rather than solely meanings. Mastery of a specific language

can tie a speaker to a community; Cutler et al. [44] describe the phonological, grammatical, and lexical patterns that together form the linguistic style of American hip-hop. Additionally, Cutler explores the blending of local influences, including code switching with languages and dialects in Western European hip-hop. Such blending is also described by Dovchin [55] and exists in J-Pop blending English with Japanese lyrics, citing Western influences [140]. These cultural and linguistic practices have consequences for voice interactions. Differences in language use in different genres could cause differences in the accessibility of their content. Unintended biases can arise in what is not accessible.

Chapter 3

Improving Accuracy in Intelligent Systems

Intelligent systems have fundamental differences from traditional technical systems as evidenced by the creation of new guidelines for interaction with intelligent systems [4]. Often intelligent systems are making predictions or modeling phenomena; these systems do this using statistical and probabilistic techniques. This usage of probabilistic techniques introduces an important new factor in the user experience: accuracy. Accuracy is a measure of how often a system is correct or incorrect in its predictions of some phenomena. Systems with poor accuracy lead to a poor user experience [195]; they waste user time because the user must constantly be evaluating whether they agree with the system, as well as dealing with sometimes costly system errors. In order to remove this barrier, and create a positive user experience we should make our intelligent systems as accurate as possible. Studies have shown that user perceptions of accuracy and trust correlate to both more usage of the system overall and more usage of features that are predictive [92]. In this chapter, I explore what it takes to make an accurate intelligent system in an important domain, that of emotional analytics.

3.1 Introduction: Why we Need Accurate Mood Models to Promote Mental Wellbeing

Many mobile healthcare applications aim to improve both physical and mental wellbeing, taking a behavior change approach [125, 95, 121] by supporting detailed monitoring of target behaviors, such as diet, mood or exercise. The goal of these healthcare applications is to enable users to derive insights about the consequences of specific behaviors on their health. These user insights can then inform decisions about appropriate behavior modifications to improve personal health.

The first wave of personal healthcare systems focused on simple behavior tracking. The ubiquity of mobile phones provides a straightforward way for people to monitor health-relevant daily behaviors to better understand how these might be improved. For example, a user experiencing sleep problems might employ an application that allows them to log multiple sleep-relevant factors including exercise, diet, social activity, bedtime, work habits and so forth. By analyzing the effects of each of these factors, this sleep-deprived user it is argued should then be able to determine which of these trigger factors have critical impacts on sleep. In an ideal world, this careful analysis of system data should allow users to modify their behaviors to improve sleep. While there have been important successes with such monitoring systems [12, 27], they have crucial limitations. One critical issue is that these systems place heavy analytic demands on users. For example, users may struggle to disentangle what behavioral triggers affect their sleep, because there are so many possible factors, including exercise, social interactions, diet, bedtime, and more [162, 23, 25]. Users may also have further difficulty in-

terpreting the results of multi-factor tracking because factors can interact in complex ways. For example, it may be hard for users to generate the insight that a combination of minimal exercise and a late bedtime are maximally disruptive for sleep if neither factor alone is deleterious. But how can we better support the majority of users who lack the advanced data analytic abilities to interpret time-varying data with complex interactions between behavioral triggers [160]? This paper argues that such systems must provide user-centric data analysis tools allowing people to reason about these complex relations between triggers and target behaviors. In other words, these systems must support analytic sensemaking.

One response to the need for sensemaking has been to propose new types of analytic tools that provide interpretive support for complex personal health data. One class of tools offer simple visualizations illustrating correlations between trigger activities and health behaviors [65, 97]. Other systems provide natural language summaries describing how triggers are affecting target behaviors [14]. However, there are limitations to these approaches. For example, these systems tend to explore simple relations between trigger activities and goals (e.g. late bedtime affects sleep), whereas in many cases there are complex interactions between multiple variables (low exercise combined with late bedtime reduces sleep) [162, 23, 25]. And even when users do correctly interpret the effects of combined behavioral triggers this does not guarantee behavioral improvements. Sensemaking is necessary but not sufficient; it must also be supplemented by a remedial plan. To return to our example, simply understanding that bedtime and exercise together influence sleep is not enough to positively change behavior. Rather, users also have to plan effective new ways to change those two behaviors to achieve positive effects. For example, a user might decide they need to implement a combination of an early bedtime of 10

pm, allied with an exercise regime of at least 8000 steps to promote a good night's sleep. Other work has shown that such remedial planning works best if plans are concrete and executable [75], but such requirements are not always straightforward to satisfy.

Given these dual requirements of sensemaking and remedial planning, we explore a different approach to behavior change in personal healthcare systems. Our novel approach uses predictive algorithmic modeling to provide recommendations to users about how to modify activities to promote effective behavior change. Rather than devolving the burden of analytic sensemaking and devising remedial plans to users themselves, instead we scaffold both these processes. To aid sensemaking, we provide predictive end-user models. These models allow users to straightforwardly determine which trigger factors affect their wellbeing. More importantly, we also offer users actionable recommendations about what remedial actions they might undertake to positively change behavior. Our system addresses emotion regulation through the use of activity planning. It is well known that people experience major challenges in regulating their emotions; this has important consequences for emotional wellbeing. People are typically poor at predicting future emotions [72]; they overestimate the impacts that recent negative events will have on long-term affect. They also find it difficult to choose future activities that will improve long-term wellbeing [203]. Finally, when in a distressed state, many people tend to recall negative information rather than enhancing mood by remembering positive events [106, 215]. People who have difficulty overcoming these affective biases can experience severe negative consequences for their mental and physical wellbeing [41, 37, 206]. Access to algorithmic insights about one's own mood could alleviate these problems, especially if accompanied by remedial methods to positively improve mood [106]. These algorithmic insights may help

users regulate mood by providing recommendations that are not tainted by the affective biases everyone experiences. Supporting mood regulation with software is promising, but relatively little research attempts to provide this mood regulation support for the general population.

To help people better understand and regulate their emotions, we designed and implemented a mobile phone-based system called EmotiCal (Emotional Calendar, see Fig 3.1). Unlike many off-the-shelf applications, EmotiCal goes beyond simple mood and activity tracking. It supports predictive emotional analytics to help with sensemaking, allowing users to better understand how specific everyday activities influence their mood. EmotiCal also helps users generate remedial plans which take the form of personalized recommendations about new behaviors to improve mood. Elsewhere we describe a 3-week intervention study demonstrating how engaging with EmotiCal's predictive analytics improves wellbeing and increases users' sense of control over their emotions [92]. In addition, users who engaged with EmotiCal's emotion forecasting and remedial activity recommendations were more successful at choosing activities that improved their mood [92]. The current paper instead focuses on system design, specifically the underlying mood modeling that underpins both sensemaking and remedial planning. To develop these models, we examine users' active evaluations of trigger activities that users believe affect mood, as well as the explanations they provide of how these factors influence mood. Trigger activities are common social, work and health-related behaviors. Users log how often they engage in such activities within the application, and also provide information about each activity's effects on mood, e.g. a good night's sleep might enhance mood. We explore how these logged activities and explanations can be used to predict mood. Accurate mood models are critical to facilitating user sensemaking. With accurate mood models, users

are able to better understand the factors underlying mood because of the reliable relationships those models capture between trigger activities and mood. More importantly, accurate models are necessary to provide compelling recommendations about remedial activities users might perform to improve mood. We explore exactly what types of data are needed to generate these models and what system designs might promote the collection of such data. While our focus here is on emotion regulation, the insights we generate have direct implications for a broad class of quantified self and behavior change systems that aim to help users gain insight into, and hence modify important personal behaviors. This paper describes how we developed accurate mood models to support sensemaking and remedial planning. Data were derived from deployments of the EmotiCal system with 70 total users who generated 2,875 log files to provide this data. One critical question we wanted to address was the role of active user evaluation. In our procedure, users actively engaged in reflecting about their mood and trigger behaviors in contrast to recent automatic approaches [121, 113]. This active reflection involved identifying which activities affect mood, weighing the effects of those activities, and generating explanations for those effects. This active user approach generates rich, systematic data but imposes additional logging burdens on users. Given these burdens, in addition to examining the role of active reflection, we also assess the additional effects active data collection has on model accuracy.

We were also interested in individual differences between users; as it may be that certain classes of triggers have very different effects on different users. For example, for some people, activities related to work may be critical in determining their mood, whereas for others, their social behaviors might have much stronger effects. Our study explored these differences. Finally, we were interested in the effects of temporal context on mood. One's current mood

may be highly influenced by anticipated future events, such as an upcoming vacation, or by past events, such as a recent family reunion. We wanted to assess the contributions of recent past and upcoming future events on current mood.

We address the following questions:

- **Explanatory Models for Mood:** How do we derive accurate mood models? How do different sets of trigger activities, specifically social, health, and work, affect mood? Additional questions include: Does active user reflection about activities improve mood models? Do user explanations also improve models? Are there individual differences between users? How do past and future events affect current mood?
- **System design:** Based on the above models, how might we design effective systems to better track, capture, and predict mood? What types of data are needed for accurate modeling? What system design features allow such data to be collected?

3.2 Methods

3.2.1 EmotiCal System Overview

EmotiCal users actively record mood, energy level, and up to 14 trigger activities that users believe have influenced their mood. For example, they can track social interactions (e.g., time spent with a friend or coworker), aspects of physical health (e.g., sleep or exercise), and work activities (e.g. meetings) to log these activities' effects on mood. EmotiCal also encourages active reflection by evaluating exactly which activities have affected their mood. EmotiCal also prompts users to generate short explanations of how and why they think those activities

have affected mood. This active reflection has been shown to be important for behavior change [53, 95]. EmotiCal uses this logged information about mood and activities to create an individualized mood model for each user, predicting how different trigger activities influence the user's mood. Users are also encouraged to report energy levels separately from mood valence as this has been shown to provide more accurate information about one's emotional state [180, 179].

Fig 3.1 illustrates the main functions of EmotiCal. The left-hand panel shows the landing page visualization for EmotiCal users. This visualization allows users to engage in affective forecasting about their future mood, as well as sensemaking analytics to plan future remedial activities to improve mood. The center panel shows the mood-monitoring interface with options to rate mood and energy level, as well as contextual information, e.g. time and location. The right-hand panel shows the UI for evaluating trigger activities that led to current mood (e.g., that food had a positive impact on current mood). There are a total of 14 possible activities the user might select as affecting mood, although not all are shown in this UI view. Models of the relations between these activities and mood are used to generate recommendations about potential remedial plans shown in the left panel. We discuss mood scales and trigger activities in more detail below. EmotiCal uses this historical monitoring information to predict each user's expected general mood for two upcoming days. Past and future expected moods are presented to users in a visualization (see left-hand panel of Fig 3.1). Sensemaking and remedial planning are supported through interaction with this visualization. Users are encouraged to actively manipulate their future mood by adding recommended mood-enhancing activities to their schedule. Two slots (+s) are displayed above the visualization showing moods of today, tomorrow and the day after tomorrow. Users can click on a slot and see a list of

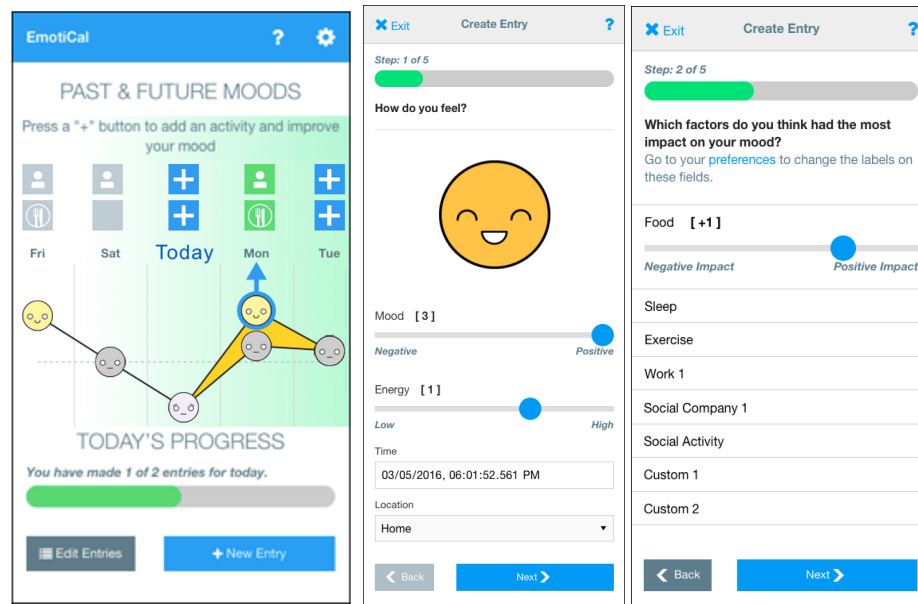


Figure 3.1: EmotiCal System Components. The first screen shows the mood-forecasting component. The second and third screens show parts of the logging.

recommended activities. Recommendations are derived from a user's logs (history-based) or their psychological needs profile (needs-based). History-based recommended activities are derived specifically from the user's own past data; this allows EmotiCal to propose actions that the user's own logging data indicate have positive past relationships with mood. Needs-based activities are generated by profiling each user's psychological needs as assessed using the Basic Psychological Needs Scale (BPNS) [47], and generating activities that meet those needs. This paper focuses on history-based profiling as these recommendations are directly derived from modeling.

After selecting a recommended activity by pressing a '+' above the visualization, users are prompted to schedule that activity. Past research shows that concrete implementation intentions improve the likelihood of following through with a plan [75]. Textual feedback then

User ID	Recommendation	Activity	User Explanation of Anticipated Mood Benefits of Planned Activity
80126	History-Based	Food	It helps me gain more energy and feel happiness. I will go to my favorite restaurant around 6pm tonight.
42968	History-Based	Work	I feel that I should do some work toward writing daily, not only does it keep up my abilities as a writer while I'm not in school it also feels like what I should be doing.

Table 3.1: Remedial Plans Created by Participants. These remedial activities were recommended by EmotiCal based on the user responding positively to this activity in the past.

summarizes the activity plan (e.g., “At 9am tomorrow, I will go for a run.”). The user then writes a brief description of the expected benefits from engaging in that activity and any additional planning information necessary; prior work again shows this to improve intervention effectiveness [206]. After finishing activity planning, the visualization then updates to show the predicted changes in mood resulting from adding the new action. In Table 3.1 are two examples of planned remedial activity entries showing sources of activity recommendations derived from log file modeling. The Activity column shows the type of mood boosting activity planned, and the Explanation column shows user-generated expectations of how activities will help along with anticipated benefits. Thus User 80126 plans to boost mood by scheduling a Food related activity involving going to her favorite restaurant at 6pm that night and 42968 plans to write each day to master that important skill. EmotiCal recommends user-specific enjoyable activities; adding an activity increases the expected mood for the planned day. For example, User

42968's Work activity plan of daily writing would increase his estimated mood for the next day from 'slightly happy' (+1 on the mood scale) to 'happy' (+2 on the mood scale) on the later-explained full mood scale of -3 (very negative) to 3 (very positive). The magnitude of this estimated mood change is calculated by the mood models we describe in this paper.

We now turn to our system deployment in which users used EmotiCal to monitor their emotions and activities over a 3-week period. 4.2 Mood Monitoring, Energy Level and Trigger Activities EmotiCal monitoring involved users logging information about mood and trigger activities that potentially contribute to current mood, and actively weighting the extent to which that trigger affected mood. Users also generated short text benefit descriptions explaining how and why they thought those triggers activities affected their mood. EmotiCal prompted users to create at least 2 mood entries per day with notifications that encouraged them to submit at least one morning entry and one evening entry. Prior research using similar methods indicated that allowing users to make entries on their own schedule led to more carefully considered entries and better compliance than system-generated prompts [95, 106]. However, this approach may preclude users from recording in specific situations, if they are stressed, busy or engaged in social interaction [95, 106]. Users were prompted via automatic text messages if they did not spontaneously submit a minimum of 2 entries per day.

Making a mood entry was lightweight and could typically be done in about 40 seconds. To create a mood entry, users first make a simple mood valence and strength decision, choosing a mood ranging from -3 (very negative) to +3 (very positive) (see center panel of Fig 3.1). This scale is similar to the valence row of the Self-Assessment Manikin (SAM) though extended to 7 ratings rather than 5 [21] to allow for finer mood granularity. Similar to SAM,

Which factors do you think had the most impact on your mood?
Go to your [preferences](#) to change the labels on these fields.

Food

Sleep

Exercise

Work 1 (Programming) **[+1]**


Negative Impact  *Positive Impact*

Figure 3.2: Users record which trigger activities influenced their current mood on this screen of the application

we showed a face that changed as different moods were selected. Following prior work, we also included energy level evaluations to increase the precision of mood ratings (e.g., a cheerful emoticon may have a connotation of excitement when the user wants to convey feeling calm) [180]. Energy ratings ranged from -3 (low energy) to +3 (high energy). Users could also optionally change the time and date of the entry or set a location (Home, Work or Other).

After selecting mood, users were prompted to engage in active mood analysis. Users were asked to identify which of 14 possible trigger activities influenced their mood and to rate that influence on a scale of -2 (negatively impacted mood) to +2 (positively impacted mood). The (-2, 2) scale was based loosely on the Positive Events Schedule [132], we extended this from a 3 item rating to a 5 item rating in order to allow users provide more sensitivity for our modeling algorithms. Users could choose as many activities as they felt were relevant, although most users chose relatively few per entry. The mood-analysis component of the UI is shown in Fig 3.2.

The 14 trigger activities that we incorporated into the system for mood analysis were identified from three different sources: a log file analysis from a previous study of mood-tracking [106], surveys (n=39) and interviews (n=12). In each of these contexts, informants were asked to identify activities that affected their mood. By far the most frequent trigger activities discussed were food, sleep, exercise and general social activity. However in addition to these common triggers, all informants mentioned several more esoteric personal activities that also had emotional impacts. Informants discussed the effects of highly customized, specific activities such as particular leisure activities (e.g., knitting) or socializing with a particular person (e.g., a romantic partner). These esoteric activities fell into three main categories: leisure, work and social domains, with informants generally mentioning more social factors than leisure or work triggers. Survey respondents who described work largely emphasized negative impacts on mood. Given the goal of EmotiCal was to use log files for positive activity recommendations, we prioritized customized leisure options in order to motivate the later activity recommendations. A final class of triggers fell outside these three domains, for example alcohol intake, menstrual cycle, or finances. To account for the wide range of other triggers for mood, we included 2 custom options. All participants were given an instructions document that explained how to set mood triggers, and discussed with a researcher in an onboarding phone call the trigger creation process. Participants were instructed to not change triggers once they were initially set.

This requirements data informed the design of the EmotiCal UI for tracking and analyzing triggers, where there were 3 main classes of trigger.

- Default activities: The UI first probed the four commonly reported default trigger activities (food, sleep, exercise, general social activity).

- Non-default: Then, to address the issue of esoteric triggers, we also allowed people to track non-default activities. Following our requirements analysis, and to provide users with some guidance, we explained during setup that non-default activities could be of three general types, work, leisure and social, and we provided some examples. A user could therefore decide that ‘Playing Music’ was often important for their mood, and so we allowed them to set it as a non-default trigger activity. Or they could decide that ‘Processing Email’ was a work activity that affected mood and decide to track that.
- Custom: Finally, because we did not want to restrict users, we also told them that they could also define other Custom activities that did not fall into these prior categories.

To ease tracking, non-default and custom activities have an editable title. Thus triggers discussed above might show up in the system as ‘Leisure 2 (Playing Music)’, or ‘Work 1 (Processing Email)’. Overall users could set up to 10 customized triggers of which 3 were social, 3 were leisure and 2 were work, and two were totally uncategorized, motivated by the triggers identified in our initial requirements samples.

After choosing activities that affected their mood, the UI encouraged users to submit a brief free-write explanation about how those triggers activities impacted their mood (e.g., “I really love the TV shows I watch. Class today was too demanding and draining.” – User 13489). Again, data entry was lightweight and took around 40 seconds on average, as users tended to select a small number of triggers (Mean=2.3, SD=1.21). For the trigger activities UI, see Fig 3.2.

3.2.2 Users

Ninety-two users were recruited through Craigslist, Facebook, Quantified Self forums, university classroom announcements and flyering. Users were eliminated from our analysis if they were noncompliant, which we define as entering fewer than 10 entries over the course of the three-week study. This criterion eliminated 22 users from our analysis, resulting in 70 compliant users. These compliance levels are consistent with those reported in similar [95, 93]. The final sample consisted of 48 females and 21 males and 1 unspecified person, (Mean age=30.7, SD=10.26). Users received a \$20 Amazon gift card as compensation for participating.

3.2.3 Procedure

Users were told that the research goal was to beta-test a new technology to help regulate mood and improve wellbeing. Users first completed an online pretest, consisting of a set of surveys to assess emotional wellbeing and behavior frequencies with enjoyment ratings for those behaviors [92]. We then emailed users a web-link to EmotiCal with login information. To maintain user compliance, researchers individually contacted users by text and phone within the first week; this ensured that users were consistently submitting entries and addressed any technical errors or confusion over study instructions. We also scanned server logs to confirm that users were indeed making daily entries, correctly following instructions and were not submitting records that would raise concern (e.g., self-harm). We contacted users to answer the post-test survey three weeks after the start date; they were debriefed, thanked, and given the opportunity to delete or modify any data they wished to keep private before data analysis. Overall

users generated a total of 2,875 logfiles.

3.3 Results

Accurate mood models are necessary to support compelling activity recommendations. If models are perceived to be highly accurate, users are more likely to engage with the system and use its recommendations [92]. Multiple different types of data potentially affect our models, including trigger activities, user explanations, and individual differences. We now explore the effects of each of these different types of information to model mood, with the aim of developing the most accurate models possible.

3.3.1 Modeling the Impact of Activities on Daily Mood

Mood ratings were recorded on a 7 point scale (-3 to +3). We chose regression rather than classification based machine learning to better quantify the exact contributions of each feature towards mood. Regression models were trained using Ordinary Least Squares regression from the statsmodels Python library [186]. Feature selection for textual models was done using scikit-learn [155]. All R^2 reported are adjusted for the number of features.

We first analyze trigger activities that users logged. Next we examine the textual explanations associated with each mood post, in which users analyzed how and why they felt that those activities would benefit mood. Finally, we explore individual differences in how different activities affect mood.

3.3.2 Activities are Critical for Explaining Mood: Health and Social are Important

We begin by regressing on the 14 trigger activities in order to predict overall mood across all users. The model was highly predictive $R^2 = .434$ and highly statistically significant ($p < .000001$). We refer to this model in Table 3.2 as the Activities model because it solely uses the trigger activities that users record. In order to compare beta coefficients of activities in this model, trigger activities, and mood were both normalized to [0,1]. The regression was calculated using all the entries created across users. These coefficients are graphed in Fig 3.3. We examined the model to determine which types of activities most affected mood, by exploring the relative weightings of the activity categories on the overall model. The beta weights for the different categories are shown in Fig 3.3, indicating that users felt that most activities had positive effects on mood. All 14 factors were significant at $\alpha = .001$.

The next critical question we wanted to address was the role of active user evaluation. In our procedure, users actively evaluate and weight the role of trigger activities; they determined both which activities affect mood, as well as weighting the effects of those chosen activities. How critical are such active weightings for generating accurate models, as opposed to less burdensome automatic approaches that track activities automatically? The accuracy of the model appears to result from users' active appraisal of the extent to which activities affected mood, rather than the simple fact that they engaged in these activities. For example, a user simply recording that they slept is not highly predictive of mood. In contrast, adding the user's active appraisal informs us both how well they slept and how that user feels that sleep

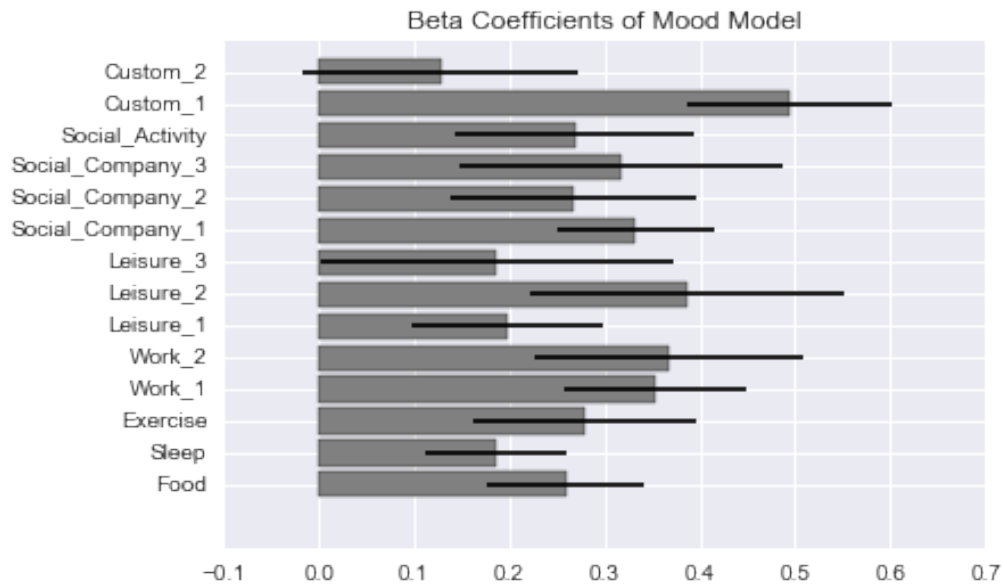


Figure 3.3: Coefficients of Trigger Activities Predicting Mood. Bars indicate the standard error of each coefficient

affected their mood. User 86753 does this exactly in one of their entries “I got enough sleep last night, . . . so I am in a good mood.” To evaluate the model difference between active evaluation of activities and simple presence/absence of activities, we removed the self-evaluation in user entries; we did this by mapping the [-2,2] scale for activities to a simple binary [0,1]. Ones replaced any non-zero entry and zeroes remained, representing an effect/no effect contrast. In other words, we ignored all weightings and treated all cases where users rated an activity as equivalent. We then calculated the relationship between mood and these binary effect/no effect features. This model without active evaluative weightings loses nearly all its predictive power, resulting in an $R^2 = .037$. Active user weighting is therefore critical for accurate mood models.

In order to better compare differences between different classes of activities, we constructed a hierarchical model that aggregates the data from the 14 individual trigger activities

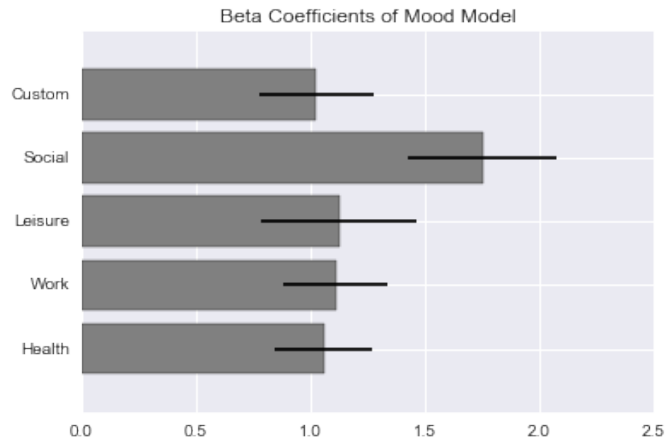


Figure 3.4: Category Model Predicting Mood. Bars indicate the standard error of each coefficient

into superordinate categories. We did this by combining the effects of the different health related activities, so that sleep, food, and exercise trigger activities were aggregated into a common ‘Health’ category. Likewise, we combined other activities into Work, Social, Leisure, and Custom. We modeled the relations between these superordinate categories on mood, to determine the mood effects of each. The model using superordinate categories was again highly predictive with $R^2 = 0.414, p < 0.000001$.

We then calculated significant differences between the superordinate categories; we did this by comparing the Pearson correlations between pairs of Activity triggers and mood while controlling for the correlation between each pair. Overall Health and Social factors have a larger impact than all other factors. There was no difference between the effects of each of the other activities. The other 3 activities: Work, Custom, and Leisure, were not significantly different from each other. This result supports Parks et al. [154] whose users reported that developing and maintaining social relationships was the most important and meaningful activity

Type(s) of Data Included in Model	R^2	Significance (p-value)
Activities	.434	< 0.000001
Explanations	.442	< 0.000001
Activities + Explanations	.565	< 0.000001
Activities + Explanations + Individual Differences	.613	< 0.000001

Table 3.2: Contribution of Different Types of Information to Predicting Mood

for user happiness.

Our primary goal was to develop as accurate model of mood as possible to enable sensemaking and remedial planning, where planning requires proposing specific activities. We therefore return to the original 14-factor Activity model to explore other factors that contribute to mood. The next factor we examined was user explanations.

3.3.3 User Explanations Improve Mood Models

In addition to actively weighting activity categories, users also added a short text description explaining how and why specific activities affected mood. Below are a few examples from users:

- “I woke up feeling really sick and then dealing with an hour commute on public transit made me feel worse” [User 10606]
- “I had four meetings today but being around my friends made it a little better.” [User 13489]
- “Secured a much coveted freelance gig” [User 71153]

We analyzed the text in the user explanations to determine whether it offered information that improved our prior activity based mood models. We took two approaches to analyzing this text in these explanations. The first approach used Linguistic Inquiry Word Count (LIWC). LIWC is a tool that analyzes individual words and categorizes them into different linguistic categories, e.g. the words ‘hate’, ‘fear’, and ‘rage’ are all classified by LIWC as negative emotions [158]. Using LIWC, we first analyzed the percentage of words that mentioned positive (e.g., happy, enjoyed, bliss) and negative emotions (e.g., sad, depressed, angry). Overall, 6.79% of total words were positive and 2.03% were negative. Sixty-six and a half percent of explanations contained positive and 25.3% negative words, with 15.4% containing both. We wanted to see the extent to which providing explanations that referred to positive or negative feelings predicted users’ mood judgements. So for each user explanation we determined the percentage of words within each explanation that were negative and the percentage that were positive and regressed this against the mood rating. Use of positive or negative emotion words was indeed related to mood in a multivariate regression model, $R^2 = .18$. Adding these linguistic categories to the Activity model, led to a modest improvement in that model from $R^2 = .434$ to $R^2 = .474$.

However, one limitation of LIWC is that it relies on fixed mappings between specific words and predefined emotional categories. However, people often talk implicitly about their emotions [10]. To address this implicit expression of emotion, we next modeled language in posts using unigrams in a second analysis. Unigram modeling examines whether the use of specific words by users correlates with changes in mood. Using unigrams also improved the basic activities model, but this time more significantly. Unigrams of user explanations alone generates a highly predictive model ($R^2 = .442$) and adding unigrams to the Activities base

Feature	Coefficient
negative	-3.980933
issues	-3.033704
negatively	-2.87173
all	-2.37859
don [don't]	-2.349698
sick	-2.168835
headache	-2.075584
not	-1.993266
able	1.600763
didn [didn't]	-0.883192
with	0.74027
tired	-0.492941
enough	0.416083

Table 3.3: Unigram Correlations with Mood from User Explanations After Removing LIWC Words. Many of these unigrams point towards the importance of implicit emotion recognition in text.

model improves it by .131 to $R^2 = .556$. We denote the added unigrams of user explanations as ‘Explanations’ in Table 3.3.

We hypothesized that this additional improvement with unigrams occurred because implicit emotions in users’ text entries might be missed by LIWC. To examine this, we identified the top 30 most predictive unigrams chosen by F-score. F-score is a common way to measure feature importance [32]. After excluding 1 proper name, we subtracted the intersection of the top unigrams with the list of words that LIWC categorizes as positive or negative emotion. This allowed us to identify terms not captured by LIWC classification. This subtraction left us with 13 unigrams in Table 3.3. Following Goyal et al. [79] we note that many of these unigrams are implicit expressions of events related to negative emotions. For example, negatively weighted terms refer to experiences such as having a ‘headache’ or being ‘sick’. These negative experiences are likely to affect our emotions negatively even though they aren’t

an explicit expression of emotion. In the same way, positively weighted terms such as ‘with’ may refer to important social experiences or ‘able’ may express competence, both of which are important for psychological wellbeing [47]. Such expressions would not be captured by LIWC, nor would they be captured by the Activities analysis, which focuses on specific (generally positive) activities. However, it’s clear these Explanations provide valuable data for our mood models.

3.3.3.1 Individual Differences

We explored model personalization to see whether incorporating individual differences would further improve model quality. During the deployment, we built individual regression models on a per-user basis rather than using the general models trained across all participants mentioned above. Combining and averaging the metrics of these 70 individual models gives us a rough picture of how they performed. Averaging the R^2 of the 70 individual models resulted in a R^2 of .45 with a standard deviation of .19. However, creating a model for each individual user has both advantages and disadvantages. One major disadvantage of creating individual models is the cold start problem [184]. User models don’t become accurate until they accrue enough data, which may take weeks. An alternative to this is to use the previous generalized Activity model and add a feature for each user.

To account for individual differences we use our general Activities model but add a binary feature for each user into the overall model, therefore adding 70 binary features. One unique binary feature of these 70 is set to 1 for each entry, depending on which user made the entry. The linear regression then learns a baseline for each of these 70 binary features, i.e. each

user. The binary feature denoting each user essentially acts as an individualized intercept. This model takes into account individual differences in base mood and increases the general model's R^2 to .515. This is notably higher than the initial Activities mood model where $R^2 = .434$. Table 3.2 shows that when we added Individual Differences to the Activities plus Explanations model, the R^2 increased from .565 to .613.

3.3.3.2 User Subgroup Modeling

Another way to tackle the cold start problem is by segmenting users into groups and building group mood models. This could allow a system to quickly categorize a starting user and provide them with an accurate model from the group. For example, one group of users' mood may be affected primarily by Social and Work Activities, whereas for others, mood may depend on Health and Leisure. We analyzed the group structure in our population using k-means clustering. Recall that each user provided a subjective emotional appraisal of some of the 14 activities for each entry; this appraisal ranged from -2 (very negative) to +2 (very positive). We applied Principle Components Analysis (PCA) to this data set to reduce the number of extraneous features. Following standard methods, the number of principal components ($n=2$) was chosen through examination of the scree plot [96]. The principal components of each user's activity evaluations were clustered using k-means. The number of clusters ($k=3$) was chosen by its silhouette score [177]. Silhouette scores were generated from $k=2$ to $k=10$ and were highest at $k=3$, averaging .72 from a maximum score of 1. A silhouette score of .72 indicates that a strong cluster structure has been found [127]. Figure 3.6 illustrates the clusters that resulted.

We see marked differences between the clusters as indicated by their mean subjective

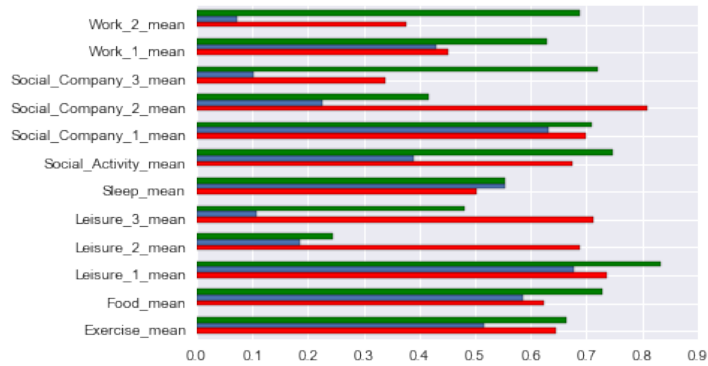


Figure 3.5: Mean Scores for Activities from Different Clusters Showing Differences in Aggregate Ratings by Cluster

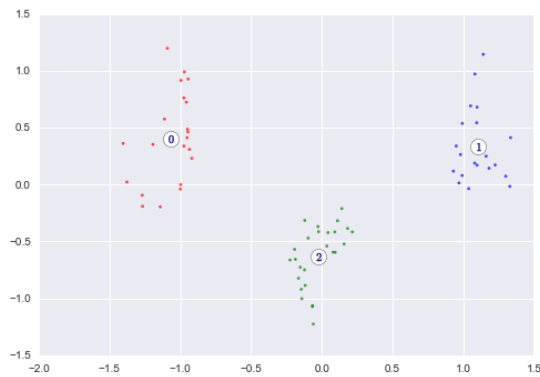


Figure 3.6: Cluster Distribution as Illustrated by the First Two Principle Components

scores for the activity categories in Figure 3.5. One of the clusters (shown in blue) is generally less positive about the activities they engage in than the other two clusters. The other two clusters (red and green) are generally more positive but differ significantly on a few categories (Work 2, Social Company 2, Leisure 2).

We then used these clusters to build group models for mood. We anticipated that these models would improve the accuracy of the generalized mood model across all users by taking advantage of the differences we saw in how those groups of users evaluated different activities.

We used the same features and process as the Activities model, and we trained mood prediction models at the cluster level. This resulted in three cluster models with varying R^2 of (.426, .421, .470). Models trained on random subsets of equivalent size to the clusters had $R^2 = (.450, .438, .422)$. As an additional test, we added the cluster labels back into the original regression as a feature; this only improved the original R^2 by .005. Despite the good separation of clusters according to the silhouette score, we fail to find cluster specific regression models that improve on the generalized models. It may be because clusters are redundant with information already present in our models.

These prior mood models have all taken a situational perspective, that mood is affected solely by activities that have occurred immediately prior to the mood judgment. However prior studies suggest that mood might also be affected by non-proximal events [106], which we now explore.

3.3.3.3 Inertia and Anticipation Have Small Effects on Mood

We went on to examine the inertia of everyday activities on mood. Our prior analyses looked exclusively at effects of current activities on mood. However, it is possible that mood is affected by activities that were carried out before or after the current time. For example, early morning exercise may have long-term inertia creating a positive mood throughout the day. We define these longer term effects as mood inertia effects. In the same way, anticipating upcoming events may prospectively influence current mood. Thinking about spending an evening with a good friend might elevate mood throughout the day. We define these prospective effects as anticipatory mood effects. To examine mood inertia effects, we calculated how activities from

previous entries influenced a future entry's moods. Given a series of entries $[t_1, t_2, \dots, t_n]$ where t is a feature vector and chronological entry number is denoted by the subscript. We predict mood of t_i using a lag of one entry by regressing on the concatenated feature vectors t_i and t_{i-1} . Mood inertia for 2 previous entries of activities was calculated in a similar fashion by concatenating feature vectors t_i, t_{i-1}, t_{i-2} to predict the mood score of entry t_i . To calculate anticipatory mood effects, the reverse was done (e.g., the mood of t_i is predicted by concatenating the features of t_i, t_{i+1}, t_{i+2}).

To examine the predictivity of mood inertia and anticipatory mood effects we augment the Activities model with entries from before or after. For mood inertia, if someone made mood entries in the morning and afternoon of a particular day, when predicting their afternoon entry, we added activities from their early morning entry to the model to measure inertia. Likewise we included the afternoon's activities in the morning mood model to examine anticipatory mood effects.

There was a marginal effect of augmenting Activities with these additional sources of information. We noted a small .016 increase in adjusted R^2 of the Activities model to $R^2 = .451$. Extending mood inertia effects by adding both the entry prior and 2 entries prior's features further increased the R^2 by .003 to .454. These results give credence to the view that mood is not lastingly affected by the past activities we asked users to log. For anticipatory mood effects we also noted rather small increases. Adding the next entry's features resulted in a .01 increase in R^2 to $R^2 = .444$ and adding the two consecutive entries further increased R^2 by .004 to .448.

We expected that mood inertia and anticipatory mood effects would add valuable information to our mood models. There are a few possible reasons that this analysis did not

produce more predictive results. One reason may be that users choose when they want to make entries; this can lead to inconsistent timings between users' entries which makes effects hard to detect. Some users occasionally missed consecutive days of entries, which would lead to our inertia predictions using stale data from multiple days before. With enough variation in this inconsistency, it could lead to more noise in the data rather than signal. A final possibility is simply that the activities that we record are mundane. It is plausible that only more major life events would have more notable temporal effects.

3.4 Discussion

Our modeling results are very encouraging in the context of systems like EmotiCal. Our models were indeed accurate as shown by the amount of variance in mood they could objectively explain, as well as by users' subjective evaluations of their accuracy. Simple Activities explain 43.4% of the variance and models additionally incorporating Individual Differences and User Explanations account for 61.3% of the variance. These are important findings because users are only likely to adopt recommended mood boosting activities when a system makes accurate predictions about projected effects of those activities on mood [92]. These mood models derive much of the predictive power from users' careful weighing of the extent to which each activity has affected mood, rather than the simple fact that the user engaged in those activities. We showed this by comparing models in which activities are given weights by users, with an alternate model in which each activity is represented in a binary fashion as relevant or irrelevant to mood score. The binary model had extremely weak ability to predict mood, showing the

importance of active user weightings. The highly predictive nature of the active user evaluations is further shown when we examined the users' textual explanations. It is clear that users are actively appraising their actions in text rather than simply describing the activities they engaged in. This active appraisal also explains the increase in predictive power that results from including textual features; suggesting users are more nuanced in text compared with a simple appraisal using an integer scale. Active evaluation provides essential information for accurate mood modeling.

These results have important general implications for the design of new technologies to detect and model mood. Our results argue for the importance of designs that encourage users to actively reflect and appraise the effects of activities on mood. However this design recommendation runs counter to proposals for new systems that include sensors and methods to automatically detect users' actions, e.g. physical activity, sleep, or social interaction using sensors [54, 212, 230]. A clear motivation for such automatic approaches is to reduce the burden on users to actively log their activities. In contrast, our findings suggest that simple activity detection has low explanatory value compared with active user evaluations. While passive tracking of activities is lightweight for users, we show that this approach overlooks important information contributed by a user's self-evaluation. Incorporating individual users' active weightings of the effects of activities on mood along with their textual self-evaluations increased our model's accuracy by 1657% from the $R^2 = .037$ from binary activity models to the $R^2 = .613$ of the full actively user weighted Activities + Explanations + Individual Differences model. Having said this, carefully reflecting and weighing what affects mood is a clear imposition on users. The demands of such active reflection may potentially reduce user compliance and willingness

to use such systems. While this clearly depends on the behaviors being analyzed, system designers must carefully consider how the accuracy of their models trades off against this user burden. Perhaps clearly informing users about the benefits of active user reflection may serve to motivate this behavior.

There are other design approaches that might promote active reflection while reducing the burden on users. We might, for example, seek to reduce the number of activities that users' actively track. We have shown that Health and Social activities are the most influential factors that need to be recorded. Tracking a few items relating to Health and Social activities might be done very quickly without overloading the user. Another approach might be to focus on individual differences. Modeling on a per user basis added considerably to the models' predictive power. Perhaps the tracking and reflecting interface might be adapted on a per-user basis to include a smaller number of user-relevant activities. Again there are trade-offs here, for example if users' lives undergo major changes it may be that new untracked activities begin to have effects on mood.

Individual mood modeling is an important consideration for both theory and for the development of future intervention systems, which will of course need to be highly personalized. We found three reliably distinct clusters of users, and although including such clusters did not improve our models overall, it may be because clusters are redundant with information already present in our individual user models. However, we believe that this clustering procedure holds promise for tackling the cold start problem in similar systems.

In addition, our findings have significant implications for mental health interventions. We have already incorporated activity models into our own EmotiCal system, a novel mobile

application to present personalized activity recommendations and mood forecasts. A field deployment has shown that people who use EmotiCal for a month display increased mood ratings and report greater insight into their emotions suggesting that our application overcame users' limitations in users' inherent abilities to forecast their own affect [92]. Increased well-being following the use of EmotiCal may have arisen because of increased affective awareness following active emotional reflection or because of direct changes to behavior resulting from carrying out mood boosting activities, or a combination of both. Our intervention does not enable us to disentangle these different mechanisms underpinning well-being improvements as EmotiCal led participants to do both. Furthermore, our approach extends current positive psychology application approaches. Many of these applications currently suggest thought-based exercises for mood improvement, such as gratitude exercises [187, 154]. Instead, our approach reflects Lewinsohn's behavioral approach to mood regulation [229] in recommending specific, personalized activities that are intended to increase positive mood and structuring recommendations with concrete planning to improve compliance [75]. Accurate mood modeling supports improved recommendations for mood-boosting activities, leading to measurable changes in wellbeing.

There are also important implications for social science. Our results show that we are able to accurately predict user mood from self-reported activity data. Better modeling of relations between activities and mood are critical for improved scientific understanding of health and wellbeing; in particular, in supporting effective emotion regulation. We also add to our scientific understanding about how activity influences mood using a new method. Our deployment allowed users to log exactly what they felt in the moment, giving our data fidelity with regards to how mood and activities interact.

Our findings also contribute to studies that characterize relations between activities and mood. We support findings using retrospective data [196, 154] showing that Social Activities contribute one of the largest positive effects on mood. However, unlike the Stone et al. study [196], we find that Health activities also impart large effects on mood. Nevertheless, the large positive effect of Health activities on mood is consistent with the findings of Parks et al. where users, in aggregate, found “Doing physical exercise or sports” to be the second “most important or meaningful” activity [154]. In contrast with Stone et al., we did not find that Work depressed mood [196]. These discrepancies with prior work may arise from a difference in subject populations. Stone et al. recruited all female users, from the same geographical region that largely worked as teachers, nurses or telemarketers. In contrast, our sample included both males and females from areas across the United States.

Nevertheless, there are limitations to our approach: in our procedure users choose which events to log, which may introduce logging biases. For example, users may be more likely to log positive rather than negative moods [93]. We addressed this by allowing user choice in log scheduling which results in more carefully considered entries [95]. Users could also navigate through the application to a page listing their previous entries, so it is possible they engaged in self-reflection facilitated by having these past entries.

Overall, our study presents a successful new method of modeling emotions that we deployed in the context of a successful emotional regulation system to promote wellbeing. We were able to accurately model which activities influenced mood by collecting active user logs about activities, as well as user explanations of how different activities influenced mood. Models were also improved by personalization. Users largely complied with active data entry over a

three-week deployment, suggesting the viability and promise of designing new personal health systems that analyze our activities and recommend new actions that can improve health. We have described one method to improve the accuracy of intelligent systems—constructing them to elicit information that is most predictive of the desired phenomena to be modeled. However, this comes at a cost of user effort needed to manually input some of this information. Given that accuracy is integral to the user experience, as we have previously reviewed, it is important that these results be tailored to individual applications. The required effort from users in some intelligent systems may be worth it due to the accuracy increases that it brings; while in other systems, it could worsen the user experience by simply burdening the user with no real benefit. Again, it is important that we center the human in our intelligent systems and design them in ways that result in the best overall experience for these users.

Chapter 4

Removing Bias in Intelligent Systems

A major problem with new intelligent systems is the differential user experience depending on user characteristics. Broadly this falls under the definition of algorithmic bias. These biases can be due to any number of user characteristics that fall outside of the user's own control. For example, facial recognition performs more poorly for users of color [168], and recidivism algorithms may unintentionally incorporate racial biases [5]. Such unintended biases are an emergent problem in intelligent systems that impairs the user experience. This user experience impairment can take the form of specific users repeatedly issuing the same command to voice interfaces because the interface does not recognize the users' accent. Other user experience problems look like people of color being unable to use many applications that require facial recognition. In order to improve the user experience, we must detect and remove bias from these systems. Only after removing biases and providing a system that works equally well for everyone can we consider our intelligent system to be truly human-centered. Here I contribute a model for understanding bias in voice interfaces involving the process of bias de-

tection, characterizing these biases, and also a process for removing bias. I show that this results in a user interface that is human centered—rather than the user adapting their way of speaking to the system, the system learns to better understand the user.

4.1 Introduction

Voice is a rapidly growing modality used to find and access a variety of content. Voice assistants are now used by 46% of United States adults [147]. Despite this rapid growth, voice interfaces may impact accessibility of content both positively and negatively. Content with long but simply pronounced names may be easier to access by voice compared to onerous text input. Other content may become inaccessible to users because of ambiguous pronunciations or automatic speech recognition (ASR) limitations. These changes in accessibility are an example of interface bias [9]; words that are easy to type may be less easy to say for particular populations. Another voice complication is that people may ask for the same content in different ways. People may not all agree on the same pronunciation, therefore confounding even a voice system trained ‘the right way’. These complications can make it hard for users to find the content that they want, and could disadvantage specific audiences. The accelerating deployment of voice interfaces combined with possible issues accessing specific types of content make it essential that we develop practical ways to examine these issues. We need methods to identify difficult voice commands and inaccessible content for users; furthermore, we need methods to rectify these issues.

Music is one of the primary use cases for voice-enabled devices [38] but music is also

associated with challenging and evolving socio-linguistic practices [64]. Music artists bend and extend language in ways that current voice systems do not accommodate. Take, for example, a user familiar with an artist from an on-screen interface, and asking a voice interface to play that artist: MSTRKRFT. A less informed user may assume the intended pronunciation is spelling the name one letter at a time, “M-S-T-R-K-R-F-T”. Other users may have seen similarly titled artists with dropped vowels and choose to pronounce the artist “Mister-craft” or “Mystery-craft”. Each of these pronunciations are reasonable. However, all these may be incorrect if the artist intended their name to be pronounced “Master-craft.” Even when pronounced correctly, a voice system may transcribe the phrase “Master craft”; this transcription has a large edit distance to “MSTRKRFT”, potentially rendering the artist unfound and the user frustrated.

Many more classes of content that are equally hard to surface using voice interfaces. Tracks such as ‘hot in herre’ have intentional alternative spellings. Some tracks use non-word sounds as their titles: OOOUUU by Young M.A. (spoken as “ooo-ooo” like the “oo” in “cool” with a descending tonal inflection starting in the middle) and Skrt (the sound that car tires make when skidding). Other artists use orthographically similar symbols as letter replacements, like 6LACK (pronounced “black”). Content titled using numbers also present a surprising amount of confusion: Quinn XCII is spoken as “Quinn Ninety-Three” and tracks like “Twenty-8” mix numeric representations. Users who want to access such content will face difficulty.

Language, names [85], music trends, and subcultures’ terminology evolve [44]. The changing context of language makes it imperative that we find ways to dynamically assess challenging content classes beyond the examples above. Music services have millions of tracks, any of which could present a problem to voice interfaces. Manually combing through all of this

content and testing it on voice interfaces is an infeasible task. Even if this task were feasible, end-users may not pronounce content names as expected and may struggle with names where the tester did not. Another option is re-training a speech model on the basis of the full content catalogue with vetted alternative pronunciations. However, this will also not be possible for many developers using off-the-shelf speech recognition APIs; nor feasible when the content space is extremely large with millions of items. Alternatively, the information retrieval and computational linguistics literature contains a multitude of large-scale approaches to learning aliases [33] from web corpora and learning machine transliterations [101] of terms from one language to another. However, light-weight approaches for voice applications in specific domains are still necessary. This especially applies when a multitude of unknown, varied linguistic issues are present. Each issue class could require dedicated detection and modeling efforts including constructing or acquiring a training corpora. Pragmatic, scalable ways are needed to find potentially problematic content classes so this content can be made more accessible.

Our contributions are three-fold:

- We present a method to automatically recognize content that is problematic for voice interfaces by leveraging engagement differences across input modalities, and apply it in a music case study
- We provide a typology of the content and practices difficult for voice interfaces to correctly surface
- We develop and test a process to make this content accessible, and describe challenges and considerations when applying such a process

Note that we use ‘accessibility’ here in the information retrieval sense, defined as the ability and likelihood to surface content in a particular system [8, 117], in this case voice interfaces. This type of work is essential as toolkits to design voice interactions become more widespread. Few individual developers have the resources to build their own ASR services; thus, many voice system designers will use off-the-shelf solutions. We demonstrate that relying on only off-the-shelf APIs may not suffice for certain content types. However, these APIs allow a much broader audience to build voice interfaces, and thus, methods are necessary to support these efforts. Our case study focuses on music, but our methods generalize to other applications.

4.2 Methods

In order to ensure that all content can be found via voice we must first understand which content is less accessible through voice interfaces. However, identification and classification of this content is not enough. We must then develop a method to improve the accessibility of the identified content. This results and methods section is structured in two parts:

- Identification: We present a method for identification of named content less accessible through our voice interface. We describe the choices and trade-offs that have to be made during this process. We analyze and describe the characteristics, including sociolinguistic practices, of this less accessible content.
- Correction: We present a way to correct these issues through a crowdsourcing method. We discuss pragmatic challenges and considerations in the application of this process. We then examine results of implementing this process and its performance improvements.

We apply this process in a music voice case study.

4.2.1 Prototype and Infrastructure

The authors were part of a team that developed an experimental mobile voice prototype to access music streaming service Spotify. This prototype was in use by thousands of end-users during this study. Voice requests for music through this interface were transcribed to text using an off-the-shelf ASR API service. Audio is sent to the API through the internet and then the prototype receives the most likely transcription in response. After the ASR API returns a transcription, the transcription is submitted to a search API connected to an index of track identifiers. This work is not meant as an evaluation of these component services' performance; such evaluation is highly domain dependent and machine learning APIs change over time. This work is an investigation of the classes of problems that developers should be ready for when using general purpose speech recognition services in specific domains, in our case music.

The prototype uses a hosted ASR provider; and thus did not have complete control over the ASR language model or lexicon. There are a number of practical challenges in this common type of set-up: the API is a black box to our prototype, we cannot modify the internals, the ASR vocabulary is not available to examine and is ever-changing, and not specialized for specific domains. The ASR API, as is a common feature, has a mechanism for adding custom vocabulary. Terms can be added to the lexicon with automatically derived pronunciations at runtime, and used to boost n-gram probabilities in the language model. These often have limitations. ASR APIs restrict the number of terms that can be added and/or considered at runtime. For a music application, a user could request any one of millions of artists and tracks.

Tens of millions of users request tracks and artists every month that employ linguistic practices problematic for standard ASR systems. The problem is even more pronounced for less popular long-tail content which is less likely to enter ASR API vocabularies. It was not possible to add all these track and artist names, and their multiple pronunciations by different audiences, as vocabulary additions need to stay within API limits. For a catalogue with millions of tracks, each requiring multiple vocabulary variations, this is not feasible. Localization and personalization may help narrow down potential vocabulary additions, but its constraints would still limit and bias the search space. This type of personalization also requires infrastructure that can be costly to build and maintain. Foreshadowing our results, we found that 7% of the content examined in this study would be affected by ASR limitations and that only 5 of 12 identified problem categories would have been solved by vocabulary additions.

Even if custom vocabulary input would be added, inaccessible content still needs to be detected and vocabulary additions generated. In addition, the entity to ASR output links we create using our method can be used to improve other services, such as textual search performance by accounting for users misspelling names they have only heard.

4.2.2 Identifying Underserved Content

4.2.2.1 Refinement of Method

Our first priority was to determine if the prototype suffered from differing levels of accessibility for different content. We mimicked a manual editorial process to assess quality for the most popular US-content, as counted by streams in the week of July 28th, 2017. One researcher, a male, US-native, Standard American English speaker, manually attempted to play

each of the most popular 200 tracks using the voice interface. This process explicitly focused on ASR misrecognition of named entities and not any other cause for lower voice performance; we ensured that all requests were in a syntax that would result in the correct result as long as the named entities were recognized correctly by the ASR (e.g. ‘Play [track name] by [artist]’). This manual editorial process was simply to validate our hypothesis that the prototype had difficulty with specific types of content.

Of the 200 tracks examined, around 7% could not easily be found using the voice interface. Some of these tracks were still accessible through spelling the entire track title aloud letter by letter or by mispronouncing the title in a way that cued the voice interface correctly. This method of identifying underserved content is informative, but clearly does not scale; manually checking millions of named content entities is not an efficient option. This approach also contributes bias itself, as the editor or researcher asking for the content has a specific accent and displays pronunciation patterns that may not be representative among the target application’s population. Even when users may be able to identify the occurrence of problems, assessing the severity and impact of errors in ASR output is hard for human judges [130]. Due to these factors, we developed a method to identify underserved content in a more generalizable and scalable way.

4.2.2.2 Identification at Scale

To identify underserved content at scale, we leverage the differences in input modalities across platforms the service is presented on. For example, a user searching for the artist ‘A\$AP Ferg’ can easily type in those characters and surface the track using the mobile or desk-

top client but may encounter issues using voice. They may pronounce A\$AP as [ei sæp], spoken as ‘a-sap’, or spell it aloud as ‘A-S-A-P’ or ‘a-dollar-sign-a-p’. These pronunciations will all result in different ASR transcriptions. A voice interface may not surface the correct artist for many of the possible pronunciations. The variability of pronunciations creates a disconnect between accessibility of content on voice interfaces compared to other mediums. Therefore, if content is very popular on the desktop and mobile interfaces and not popular on the voice interface, this may indicate that the users are not able to surface the content easily.

We create a univariate measure of voice accessibility in order to use anomaly detection techniques. For each track t , we calculate the voice findability $t_{findability}$, by dividing t_s , the total number of streams that track has experienced by t_v , the total number of voice finds the same track has experienced. The resulting distribution follows a power law distribution that we log transform in order to normalize for better anomaly detection. This equation for voice findability is shown below.

$$t_{findability} = \left(\frac{t_s}{t_v + 1} \right)$$

For related examples of accessibility metrics, see [8, 131]. We define an anomaly as a track with findability that lies over 1.645 standard deviations from mean findability. This threshold corresponds to a one-tailed t-test at $p=.05$. This threshold is lower than common anomaly detection thresholds at 2 or 3 standard deviations from the mean. We chose this threshold because false positives have a small cost in this context.

4.2.2.3 Limitations and Considerations in Detecting Tracks

The difference between popularity or finds in a voice versus a non-voice context may be caused by other issues or behavioral differences between platforms. For example, voice users may have different demographics, or situations where voice is used more may be associated with different types of music or playlists. These differences in our data would show up as false positives, anomalous tracks detected by the findability metric that are actually voice accessible. Our procedure was intentionally liberal with the definition of an anomalous track because false-positives are inexpensive (the cost of a limited number of crowdsourced utterances as described in the next section) whereas false-negatives could lead to content being inaccessible. We will later show that this anomaly identification method was accurate in surfacing tracks that are inaccessible by voice.

4.3 Identifying Content Results

4.3.1 Voice Interfaces May Underserve Specific Genres

We applied the anomaly detection procedure to the top 5000 tracks in the 28-day period from July 28th to August 24th, 2017. Before we provide a typology of the less accessible content, and its naming, we first examine the anomalous content through the lens of musical genre in order to gain a clearer view of the content that voice interfaces struggle to surface.

In order to discern which genres are underserved, we examine the proportions of genres in the top 5000 most streamed tracks as compared to the proportions of genres in the English-language titled anomalies from the top 5000 tracks. For example, if $\frac{1}{5}$ of the top 5000

tracks are pop tracks but $\frac{3}{5}$ of the anomalies are pop tracks, this may indicate that pop tracks are less accessible than other content. If all content were served equally then the proportions between the Top 5000 tracks and the anomalies would be the same. This process and its drawbacks parallel work done in auditing search engines for differences in satisfaction due to demographics [135]. As outlined above, differences in proportions could be caused by other demographic or contextual differences in music consumption through voice interfaces. However, this method provides an indication to developers that it would be worthwhile to further investigate accessibility of content in particular genres.

We use Spotify's metagenres that cluster genres, e.g. trap music and rap belong to the hip-hop metagenre. We use these metagenres to have more reliable and interpretable results. Certain genres are overrepresented in the anomaly set, indicating that these genres may contain a larger amount of content that voice interfaces have difficulty surfacing. These results are shown in figure 4.1. Hip hop rises from containing 36% of all tracks in the population to 58% of all anomalous tracks. Country music also experiences a disproportionate increase in the anomalous population, rising from 9% in the full sample to 12% of anomalies. This is in line with prior literature, showing that both hip-hop [44] and country music [66] have their own specific sociolinguistic practices.

Pop music goes in the reverse direction, indicating that pop music does not have as frequent issues with voice interfaces. In the overall sample, pop contains 32% of the tracks; in the anomalous sample, pop only contains 18% of the tracks. Rock genres experience quite large decreases, suggesting that they may struggle the least with voice interfaces; classic and modern rock combined drop from being 12% of the overall sample to only 2% of the anomalous

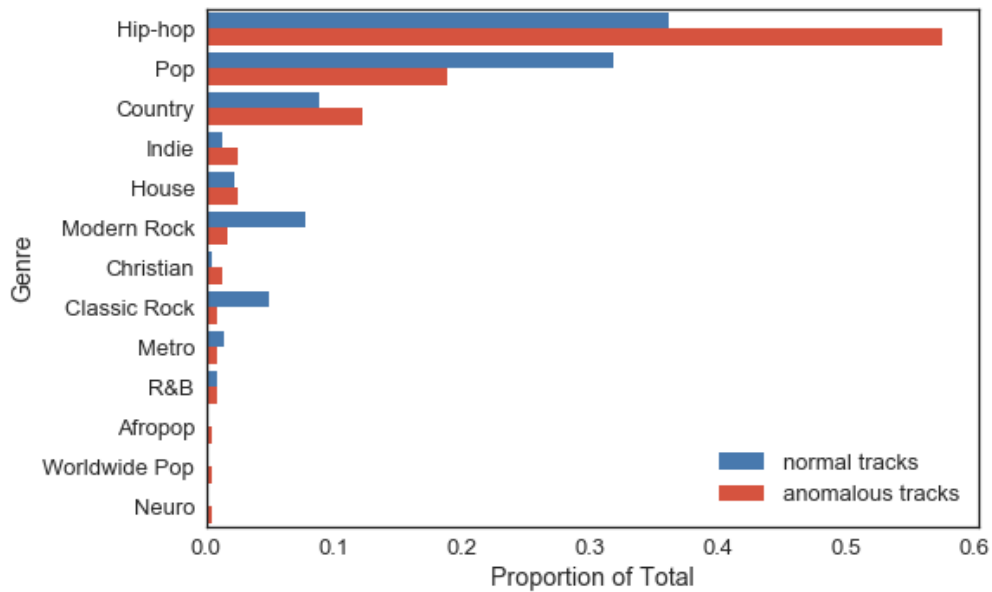


Figure 4.1: Genre Representation in Full Track Set and Anomalous Track Set

sample. In order to test for significant differences in major metagenres we eliminated 8 genres that had less than 5 tracks in the anomalous category. This limits us to only making conclusions about the changing distributions of the hip-hop, pop, country, indie, and house genres. Based on these 5 metagenres, the anomalous genres differ significantly in their distribution from the standard genre distribution as indicated by a Chi-Squared test for homogeneity was significant $\chi^2(4, N = 2075) = 1421, p < .0001$. We cannot be completely sure that this difference is due to voice user interface problems and not demographic differences between typical users and voice users. However, our later results on accuracy of our classification indicate that much of this variation is likely due to voice interface challenges.

4.3.2 Typology of Underserved Content

We now qualitatively examine the classes of content that suffer from inaccessibility due to their titles or names. This typology was created by coding the anomalies from the top 5000 tracks by number of streams in the 28-day period from July 28th to August 24th, 2017. One researcher went through the anomalies and organized them into prototype categories based upon their characteristics that created problems within the ASR system. This process created 11 different categories of content. Following this prototyping of categories, the two researchers annotated a sample of 100 of the anomalies in order to resolve conflicts and refine the categories until full agreement was reached on all 100. This co-annotation resulted in refining the definitions of 5 categories and the addition of a new category. The final typology consists of 12 categories of titles that are problematic for ASR systems.

English Dialects and Neologisms: English Dialects and Neologisms were defined as track titles that used new words that may contribute to a dialect or track titles that were spelled in a way intended to convey a certain dialect of English speech. Examples include ‘You Da Baddest’ by Future and ‘Any Ol’ Barstool’ by Jason Aldean. The determiner *da* (pronounced [də]) in ‘You Da Baddest’ is spoken distinctly from the Standard American English equivalent “the” (pronounced [ðə]). Even though these pronunciation differences are standard in the African American English dialect [201], ASR systems struggle for correctly form this dialect speech and often sanitize it to Standard American English. An example of the relationship between English dialects and Neologisms can be found in the track ‘Litty’ by Meek Mill and Tory Lanez. ‘Lit’ has referred to a status of being inebriated since the late 19th century [107]. Recently, in the

21st century, ‘lit’ has come to mean ‘exciting’ or ‘excellent’, pushed in large part by hip hop music [136]. ‘Litty’ is used as a drop-in replacement for ‘lit’ but has presented problems for voice interfaces, likely because litty was not in the ASR vocabulary.

Non-English Languages: As discussed earlier, recognizing multiple possible languages in the same system, let alone the same title, is an open problem in speech recognition [209, 228]. Current major ASR technology providers require that the implementer specify a single language that will attempt to be recognized. This produces challenges in linguistically heterogeneous regions. We do not attempt to tackle this issue using the method presented in this paper.

Abbreviations and Ambiguous Acronyms: Abbreviations and ambiguous acronyms consist of tracks that include shortened or abbreviated words in their titles or textual cues that imply abbreviation or acronym. Examples of true acronyms include ‘E.T.’ by Katy Perry and ‘She’s Mine Pt. 1’ by J. Cole. Abbreviations are often ambiguous in their pronunciation. For the above tracks many people would say the first as ‘E-T’ (pronounced [i ti]) and the second ‘She’s Mine Part 1’ but ‘extra-terrestrial’ and ‘She’s Mine P-T 1’ would also be valid utterances. An ambiguous acronym can be seen in the track ‘LUV’ by Tory Lanez, while ‘LUV’ is intended solely as an alternative spelling, users may interpret the capitalization cues to imply that they should pronounce each letter individually.

Numbers, Dates, and Times: While seemingly simple to represent, numbers, dates, and times also present a challenge for surfacing correct content. For example: ‘Twenty 8’ by Kodak Black and ‘Confessions Part II’ by Usher. Similar to the abbreviations class, we have multiple textual representations of the same spoken phrases. ‘Confessions Part II’ could also

be transcribed as ‘Confessions Part 2’ or ‘Confessions Part Two’. This means that properly recognizing and translating between different transcriptions is essential to surfacing the correct content. Similarly, time and date can be represented in different ways; ‘seven hundred hours’ can be equivalent to ‘Seven AM’; ‘7/11’ could be ‘Seven eleven’, ‘July Eleventh’, or even ‘November Seventh’.

Removal of Spaces: Removing spaces in content names can also present challenges. The track title ‘DONTTRUSTME’ by 3OH!3 is one example of this. Removing spaces can increase the edit distance to the transcription and may result in incorrectly surfaced content.

Vocables: Vocables are modernly defined as utterances that are not words but do contain meaning. Commonly used examples are ‘uh-huh’ to agree with something and ‘ew’ to express disgust. Non-lexical vocables, a subclass of vocables that convey no lexical meaning are common in many types of traditional music such as Native American music and Irish music [64]. Today we see vocables in popular music like ‘do re mi’ by blackbear (or Julie Andrews) and ‘OOOUUU’ by Young M.A. These are particularly difficult for current ASR technology. Spelling for vocables is not clearly defined and subtle variations in capitalization or spelling may convey prosodic information that is ignored by the ASR. For example, vocalizing ‘OOOUUU’ like Young M.A. on her track gets transcribed as ‘ooh’, the exact same transcription as vocalizing the ‘Ouu’ portion of Lil Pump’s track ‘Flex Like Ouu’. These two sounds are vocalized quite differently in their respective tracks and current ASR technology does not differentiate.

Non-Replacement Symbols: Artists choose to use symbols in their tracks for many different reasons, a couple include: conveying a specific feeling (**Flawless by Beyoncé) and tagging to contextualize (NAVUZIMETRO#PT2 by NAV). These symbols can also carry

implied pronunciation such as Tay-K's track 'I < 3 My Choppa'. We cannot simply ignore the symbols when transcribing; if we drop the symbols in 'I < 3 My Choppa' we lose an implied word between 'I' and 'My' and will likely not find the correct track.

Orthographical and Semantic Replacement Symbols: Symbols can also be used as replacements to normal letters or words. Common examples of this include the plethora of artists prefixed with 'A\$AP'; this is pronounced [ei sæp], spoken as 'a-sap', but many less informed users may try to spell the word. Other artists' names are difficult or completely impossible to form with current voice interfaces such as V▲LH▲LL. Semantically similar replacement symbols include usage of '&' in place of 'and' others like Ed Sheeran's album '÷' (pronounced 'Divide').

Censored Swear Words: Many publishers will censor their own tracks before publishing them by replacing parts of the offensive words with asterisks. This censorship can complicate how easy it is to surface the track using voice. These tracks may be ambiguous, the censored word in 'P***** Print' by Gucci Mane has multiple plausible replacements and only knowledge of the track's lyrics can clarify which is correct.

Expressive and Alternative Spellings: Expressive and alternative spellings are closely related to dialect speech but differ in one key aspect. Alternative spellings are not intended to modify the pronunciation of the word. For example, 'Loving U' by 6LACK is still pronounced /lʌvɪŋ ju/, an identical pronunciation to the more standard spelling 'Loving You'. Alternative spellings may create issues because the actual title can be substantially different than the transcription that the ASR produces. Combinations of alternative spelling and dialects may be particularly challenging for ASR systems, e.g. '100it Racks', pronounced [hɒnɪr ræks], said

‘hunnit racks’.

Wordplay including Homophones, Puns, and Portmanteau: Words with similar pronunciations present issues for ASR systems because they may not be spelled in easily translatable ways. One artist ‘Knowmadic’ is difficult to surface because ASR will only form ‘Nomadic’, the name of another artist. Another relatively popular artist, ‘Cerebral Ballzy’, has an acoustically different name than the disease Cerebral Palsy, but the ASR will only form the name of the disease rather than the band. Presumably the association in the ASR language model between ‘cerebral’ and ‘palsy’ is highly probable and varying pronunciations of ‘palsy’ will not change the transcription.

Names: Proper nouns are a perennial difficulty for ASR systems because of the myriad spelling and pronunciation differences [116, 150]. We see evidence of this also. Some artists like SahBabii and NAV have created new names based on shortening their given name (NAV from ‘Navraj’) or permutations of their given names combined with other words (SahBabii from ‘Saheem Baby’).

4.4 Correcting Underserved Content

Now that we have examined what groups of content may be disadvantaged by current voice interfaces we move to the process needed to fix these accessibility issues. As machine learning technology continues to become commodified, downstream users of these technologies must find ways to adapt these systems to their specific context. Downstream users may not be able to explicitly change the machine learning model or add additional training data. We

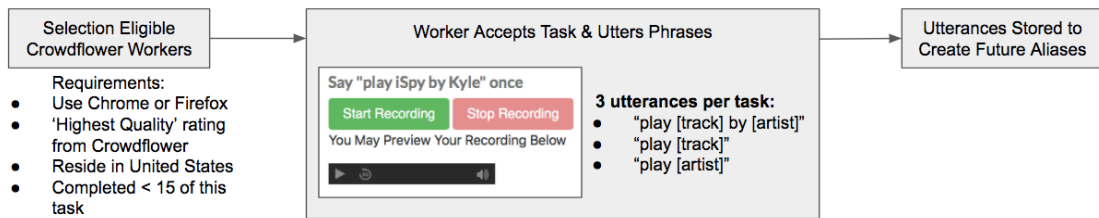


Figure 4.2: Crowdsourced Utterance Generation for Individual Tracks

cannot directly modify our ASR service or add training data, instead we create aliases for ASR mischaracterizations to ensure immediate fixes for underserved content. Each alias serves as a link to the content that the ASR struggles with. For example, the ASR system is unlikely to transcribe the string ‘Sk8er Boi’ from an utterance and will instead form the more standard English ‘skater boy’; this will not surface the right content. To direct the query to the correct content we need to create ‘skater boy’ as an alias for ‘Sk8er Boi’.

A simple way to create content aliases would involve an editor manually saying each of the detected anomalies aloud and recording the transcription from ASR as an alias. The editor may pronounce the track ‘LUV’ as [ɛl ju vi], said ‘ell-you-vee’, and then record the ASR’s transcription of ‘l u v’ as an alias for the original track. However, most of the population may actually pronounce ‘LUV’ as ‘Love’ and therefore the editor’s alias would be ineffective for many users. We need a way to sample the broad pronunciation space for such content that includes many different voices and accents in order to make generalizable aliases [199]. To generate a more diverse set of utterances than a manual editorial process we turn to crowdsourced audio generation. The crowdsourcing process and worker requirements are shown in Figure 4.2. Workers were paid \$0.50 for each completed task; tasks generally took less than 1

minute to complete. We collected utterances until each track had a minimum of 15 utterances for each type. After collection ended, we transcribed all of the utterances using the same ASR and settings that are used by the prototype application in order to ensure that the transcribed content works as an alias in the prototype. This process resulted in a variety of transcriptions for each track and utterance type.

The next step was to verify whether each of the collected utterances resulted in finding the correct content. We used the transcriptions produced by the ASR to calculate which entity would be surfaced if this utterance were made by a user. This process is shown in Figure 4.3. Each entity is identified by a Universal Resource Indicator (URI). In order to check if an utterance resulted in surfacing the correct content, we compared the original URI, from the track the crowd worker asked for, with the URI that resulted from transcribing and simulating their request. If these URIs matched, then the utterance was considered to be successful in surfacing the correct content.

The previous steps calculated whether any individual crowdsourced utterance resulted in surfacing the correct content. Now we use those steps to make a decision about the performance of the track as a whole. This step in the process verifies that the anomalous tracks our algorithm identified are anomalous due to voice interface difficulties and not due to differences in intent between modalities or failures in another part of the retrieval process. Similar to our anomaly detection threshold, we again set a relatively low accessibility threshold for when a track is accessible. We judged a detected anomalous track to be a false positive anomaly if more than $\frac{1}{3}$ of the ‘play [track] by [artist]’ queries resulted in the correct track URI. In Figure 4.4, we refer to this URI comparison process as “Examine URIs”. This $\frac{1}{3}$ accessibility threshold

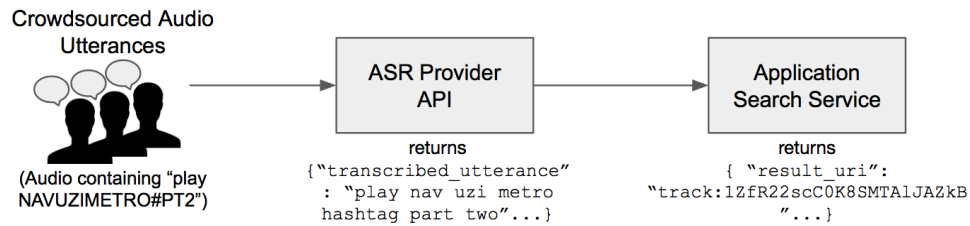


Figure 4.3: Transcription and Search Process to Resolve URIs

decision indicates that we are looking for tracks that are currently among the least accessible in our prototype. Decisions like these are part and parcel of the practitioner experience, this threshold may need to be set differently for other domains.

We outline the full alias decision process, including false positive decisions, for each track in Figure 4.4. If the ‘play [track] by [artist]’ queries do not surface the correct track URI $\frac{1}{3}$ of the time, this indicates there may be an ASR problem with the track or artist name. Examining the track and artist utterances separately allows us to tell whether the track title or artist name is the source of ASR error. We use the same $\frac{1}{3}$ correct URI threshold to determine whether or not these utterances are finding the correct content. Once the track or artist name has been identified as causing issues for our interface, we choose an alias from the collected utterances. Aliases are chosen by a simple frequency voting scheme. Aggregating generative work like spoken utterances rather than discriminative crowdsourced work is an open research problem; we use a simple voting scheme to demonstrate that this bias reduction method is robust even with simple aggregation functions.

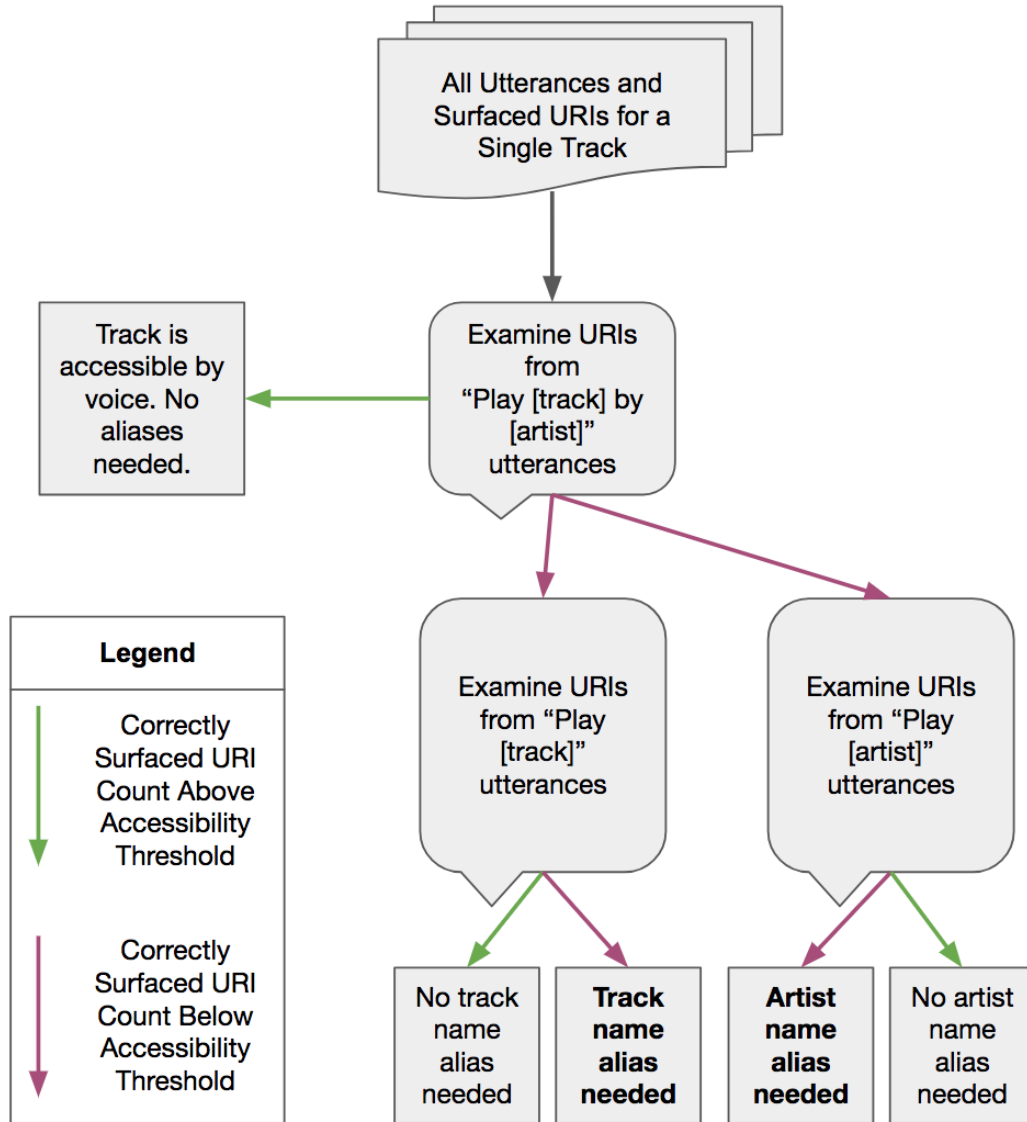


Figure 4.4: Track Aliasing Decision Process

4.5 Correcting Content Results

In the next section, we test our accessibility improvement method on the top 1000 most popular tracks from the US, in a 28-day period: July 28th to August 24th, 2017.

Fifty tracks from the anomalous set remained after eliminating anomalous tracks according to our 2 criteria: the track titles were in English and they did not create ethical concerns for our crowdsourcing experiment. In our solution in this paper we focus on English-language tracks. This is a difficult choice because code-mixing and code-switching [55] between languages happens in music applications. However, dealing with code-switching and multiple languages in a single ASR application is an open research problem itself [209, 228]. Among the tracks recognized as anomalous were tracks that contained ethnic slurs in the title. These presented an ethical concern for the researchers because we would be paying crowd workers to record themselves saying these slurs aloud. These tracks were discarded for the purposes of this study; note that they would however have to be addressed in live applications as artists can reclaim slurs or use them as social commentary. These types of tracks could be inaccessible if different pronunciations of slurs are not included.

Following the crowdsourcing of aliases for these 50 tracks, ten of the 50 tracks were deemed accessible through voice interfaces and false positives from the anomaly detection process. Recall that we set a low bar for false positives in our methodology, only $\frac{1}{3}$ of the crowdsourced ‘play [track] by [artist]’ utterances had to result in the correct track to be considered a false positive. We wanted to focus in on the most underserved tracks in order to test our method. We did not implement aliases for these 10 false positives.

In total, we used the remaining 40 tracks and aliases as input for the crowdsourcing method and alias testing. In order to eliminate potential confounds, we randomly sampled half of the 40 tracks to use as an experimental group to track alias performance. We added the aggregated crowdsourced aliases into a music streaming production environment for the 20 tracks that were in the experimental group. We now examine how the voice finds for these tracks changed after adding the produced aliases.

4.5.1 Aliases Improve Content Accessibility

In order to examine the effect of aliases on the underserved content, we examined the logs of our prototype voice interface. We examine the period directly around the implementation of the aliases in order to control for temporal effects. This time period includes 7 days before and 7 days after the aliases were implemented for the experimental group. As seen in Figure 4.5, we examine the sum of finds before the alias is implemented compared to after implementation and calculate the percent increase. A majority of tracks experienced explosive growth in their finds through the voice interface.

We test performance of our control and experimental groups using 2 methods. First, we use a pair of Wilcoxon Signed-Ranked Tests to examine control and experimental performance. Due to the fact that we planned two comparisons, we use a Bonferroni correction, thus our $\alpha=0.025$. A Wilcoxon Signed Rank Test indicates that the tracks in our experimental group were found significantly more using voice after aliases were added, $V=1$, $p < 0.001$. As expected, our control group did not significantly differ for the same time periods, $V=53$, $p = 0.093$. In addition, we specified a mixed-effects Poisson regression to better control for between

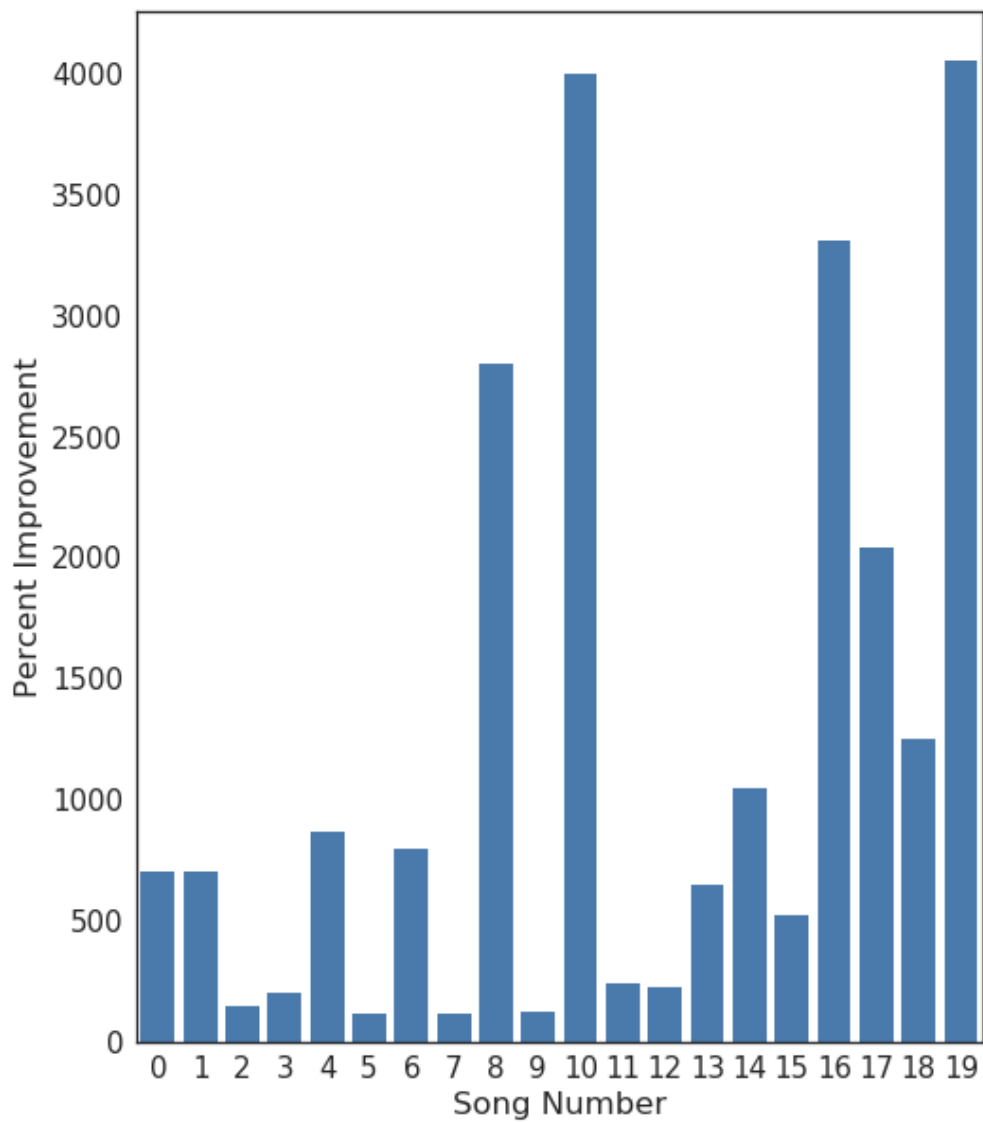


Figure 4.5: Alias Finds Improvement Over Baseline

Parameter	Coefficient	Std. Error	p-value
(Intercept)	1.771	0.312	< 0.001
Condition	-0.760	0.449	0.090
Time	0.598	0.301	0.046
Condition * Time	1.445	0.428	< 0.001

Table 4.1: Summary of Mixed-Effects Poisson Regression

group differences. The model included 2 random effects: track, to control for variation in initial voice finds, and time to control for natural variation in voice finds over time. Fixed effects in the model included an interaction between condition and time; we expect an interaction due to the implementation of the aliases in the experimental condition. The coefficient for this model are shown in Table 4.1. This mixed-effects Poisson regression found a significant interaction between condition and time, $p < 0.001$. Additionally, the time parameter was significant at $p=0.046$, this is likely due to increasing usage of our prototype over the time period. A pairwise post-hoc test indicated that the experimental group differed significantly before and after aliases were implemented, $p < 0.001$; the control group did not differ on the same time periods, $p = 0.191$. These results indicate that the implementation of aliases increased the accessibility of previously underserved content. A small number of tracks (6) experienced less growth overall, but the data at least illustrates that they were now more accessible than before.

4.6 Discussion

Voice is a rapidly growing way of interacting with consumer-facing applications. We have presented one approach to identify disadvantaged content which can be generalized to

other domains. Voice interfaces are made up of components based on textual, speech, and behavioral data. Groups that are underrepresented in training data, including those with different accents or members of sociolinguistic groups that do not use the majority dialect, will be disadvantaged. Similarly, content less likely to occur in large-scale speech training corpora, may be less likely to be recognized. This makes voice applications particularly prone to biases. Our case study shows that certain genres of content are more affected. We classified 12 linguistic and stylistic practices that present problems in current voice contexts. It is crucial to discover types of content that experience issues in scalable and easy to apply ways. In our evaluation, we showed our method increased accessibility of previously disadvantaged content.

Our method focuses specifically on enabling access to diverse content within the music space but this approach is extensible to many other domains. Developers are increasingly using public ASR APIs similar to what our prototype used. For example, take a developer creating an application containing many local, slang or dialectal terms, or app/company-specific terminology, or profession-specific scientific, medical, legal, industrial terms. While some domain-specialized ASR services are available (e.g. Nuance has medical and legal ASR products), for especially smaller developers with special purpose domains, these may not suffice. Similar issues will arise when automatically making apps voice-accessible; which commands will and will not work may not be clear. Terms may be comparably rare in the data that the general-purpose ASR API was trained on. This rarity in training data could then result in the ASR API transcribing more common similar-sounding phrases or words rather than the specialized terminology needed. Our method could identify these incorrect transcriptions and ensure that they still resolve to the action that the user desired.

4.6.1 Limitations and Trade-Offs for Practitioners

While this method presents a scalable and automated way of addressing accessibility problems, it is important to realize that there are limitations and potential improvements. It is worth considering how each decision in the process may affect the final outcome. Some voice interface problems we identified were related to representation challenges (e.g. multi-lingual content, numbers, dates, and times). Other were related to socio-linguistic practices also identified in music literature (e.g. Hip-hop [44] and country music specific [66]); or in online communication literature, such as 133tsp34k [159]. A tradeoff decision arises: if a particular problematic category becomes large enough, it may be worthwhile to develop a specific solution. However, those can be costly if requiring specific machine learning or domain expertise and datasets. Until then a method such as this one can be applied.

Our evaluation also illustrated the dynamic nature of speech recognition systems. Some problems ‘solve themselves’; two tracks in our control group became accessible without intervention, potentially through updates in the ASR system. Whether or not a developer can wait for ASR systems to update depends on the domain and expected use cases (e.g. new track releases). The size and variance of the domain-specific named content space will determine the anomaly threshold decisions and annotator decisions (crowdsourced or editorial) necessary. Anomaly detection improvements are possible by ensuring a close match of modality populations. Pronunciations can be provided by a broad population, or one closely matching the target audience, or in-house experts. Cost and access matter here.

4.6.2 Bugs or Biases

Our evaluation of the problems that voice interfaces have when catering to diverse ways of speaking forefronts a larger point about engineering challenges with new technologies. One perspective is that these problems are simply bugs within the voice interfaces that should be fixed. The voice libraries were built to understand Standard American English because, as the name suggests, our society holds it as the standard way of speaking. Therefore, speaking outside of Standard American English to a system that expects it is non-standard; the fact that the voice interface has issues with these ways of speaking is simply an engineering problem or a bug to be fixed in the next version. This perspective acknowledges that these problems should be fixed in the routine course of continually improving the system.

The other perspective holds that the problems these voice interfaces have affect specific groups of people and represent a form of bias. As we have shown, these problems are not distributed randomly in our dataset, the problems are more concentrated within specific genres that have sociolinguistic practices that differ from Standard American English. This perspective views these problems as a bias problem—the system was not designed to understand all ways of speaking and therefore privileges Standard American English over other valid ways of speaking. This bias perspective questions the tacit assumption that we should even assume the users and artists will be speaking Standard American English. Why were these voice libraries not originally created to handle the diversity of English language within the country they were created for? This perspective encourages us to better understand what populations will be using the systems we create in order to ensure that our systems work for all groups of people equally.

4.6.3 Creative Intricacies

Creators are deliberate in the way they name themselves and their content. In some cases, technological considerations are part of this process. We focused on making content accessible. It's worth noting that content creators may have different motivations. Obscurity or findability can both be treasured values. In the genre Witch House, with artists like GL▲SS †33†H, artists may intentionally obfuscate names [214]. In contrast, the electro-pop band Chvrches have claimed to spell their name "using a Roman 'v'" so Google wouldn't confuse the group with actual churches" [182]. Ironically, a general ASR system would have exactly the opposite result for users who pronounce the name correctly: churches would be found, not Chvrches. New interfaces and retrieval techniques may not necessarily align with all communities' practices, nor with content creators' existing technology strategies.

4.6.4 Conclusion

This study has demonstrated that intelligent systems can harbor societal biases. We also characterize these biases and build methods that identify and correct similar biases in voice interfaces. Truly centering the human in these intelligent systems involves designing them so that all people can use these systems equally well. Too often these systems are designed with convenient data in mind rather than spending the time to collect diverse data to improve the system like we have done.

Chapter 5

When Do Users Want Transparency

Algorithmic transparency is needed for many reasons. Greater transparency potentially increases end user control and improves acceptance of intelligent systems [108]. It can also promote user learning and insight from complex data, which is important as humans increasingly work with complex inferential systems for analytic purposes [108, 173]. Transparency can also enable oversight by system designers. Without such transparency it may be unclear whether an algorithm is optimizing the intended behavior [86, 126], or whether an algorithm accidentally promotes negative, unintended consequences (e.g. filter bubbles in social media; [19, 153]). Given these issues, it is increasingly possible that transparency, i.e. “a right to explanation”, may become a legal requirement in some contexts [77]. These issues have led researchers to argue that machine learning must be ‘interpretable by design’ [1] and that transparency is essential for the adoption of many intelligent systems, e.g. for medical diagnoses [83, 217].

5.1 Introduction

Machine learning algorithms power intelligent systems that pervade our everyday lives. These systems make decisions ranging from routes to work to recommendations about criminal parole [5, 16]. As humans with limited time and energy, we increasingly delegate responsibility to these systems with little reflection or oversight. Nevertheless, intelligent systems face mounting concerns about how they make decisions; concerns that are exacerbated by recent machine learning advances like deep learning that are difficult to explain in human-comprehensible terms. Major public concerns have arisen following demonstrations of unfairness in algorithmic systems with regards to gender, race, and other characteristics [18, 189, 199]. The need for explanation and transparency is a core problem that is threatening adoption of intelligent systems in many realms [83, 148, 217].

While such calls for transparency are admirable, it is unclear exactly what is needed to enact them in practice. Extensive research about how to operationalize transparency has risen from both machine learning and HCI communities but no clear consensus has resulted [1, 57, 216]. The ‘how’ of transparency is difficult—there are numerous implementation trade-offs involving accuracy and fidelity. Making a complex algorithm understandable to users might require explanatory simplification, which often comes at the cost of reduced accuracy of explanation [112, 174]. For example, methods have been proposed to explain neural network algorithms in terms of more traditional machine learning approaches, but these explanations necessarily present approximations of the actual algorithms deployed [129].

In addition, recent empirical studies have also attempted to present a case for the

‘why’ of transparency; however, these studies have shown puzzling and sometimes contradictory effects. In some settings there are expected benefits: transparency improves algorithmic perceptions because users may better understand system behavior [105, 108, 123]. But in other circumstances, transparency can have other quite paradoxical effects. Transparency may cause users to have worse perceptions of a system, trusting it less because the transparency led them to question the system even when it was correct [123]. Providing system explanations may also undermine user perceptions when users lack the attentional capacity to process complex explanations for example while they are executing a demanding task [24, 224]. Overall these results indicate mixed evidence as to why transparency should be implemented for users.

We also lack a clear explanation of why transparency has seemingly contradictory effects. The current study aims to provide such an explanation. We explore the possibility suggested by prior work, that explanations need to be provided at contextually appropriate times [24]. Prior studies indicate that users benefit most from transparency when their expectations are violated and when they are not overloaded with information. Rather than focusing on how to operationalize transparency or why to be transparent, we frame the problem differently: when is the best time to present transparency?

We approach this question using empirical mixed-methods to examine transparency in the context of a working algorithm that interprets a user’s description of an emotional experience. In three studies, we explore the connection between the impact of transparency and the user/system context it is presented in. We identify instances when transparency increases users’ understanding and confidence, as well as when it might undermine user confidence. We examine the relationship between user expectations, system output, and transparency. We address the

following research questions (RQs):

- (RQ1): When do users want to see transparency? (Study 1,2,3)
- (RQ2): Why does transparency help or hinder user understanding? (Study 1 & 2)
- (RQ3): How does transparency influence users' expectations and perceptions of system error? (Study 2)

To answer these questions, we conducted three studies. Study 1 explores how users engage with and understand a transparent working intelligent system using 'think-aloud' and semi-structured interviewing methods. Study 2 takes our observations from Study 1 and incorporates a quantitative design to test predictions regarding how users form perceptions of accuracy in the intelligent system. Study 3 builds on studies 1 and 2 by assessing how users respond to transparency information evaluating this in a naturalistic experiment. We then explain these results in terms of expectation violation and theories of social explanation.

5.1.1 Contribution

We contribute to the growing literature on algorithmic transparency through three user evaluations of a working intelligent system in the Personal Informatics domain. Previous research on transparency and intelligibility has had highly mixed results. Positive system perceptions can be built through transparency [56, 105, 124], even to the point of overconfidence [69]. At the same time, however, positive system perceptions can also be undermined by transparency [61, 105, 123, 143]. Our approach draws on psychological and sociological theories of communication applied to HCI [70, 81, 161, 198] to explain when, why, and how users want

transparency. Our findings reveal that transparency can have both positive and negative effects depending on context. We present a model that shows how the context of transparency and expectation violation interact in forming user perceptions of system accuracy. Transparency information has positive effects both in helping users form initial working models of system operation and reassuring those who feel the system is operating in unexpected ways. At the same time, negative effects can arise when transparency reveals algorithmic errors that can undermine confidence in those who already have a coherent view of system operation. Finally, we verify how these self-report data match actual user behaviors. We show that greater expectation violation leads participants to spend more time exploring transparency information. Results again are consistent with our prior qualitative results suggesting that transparency helps building initial mental models. We therefore find that users spend more time when first exposed to transparency information and that time with spent transparency declines over time as they see explanations again. We explain our results using theories of occasioned explanation [70, 81], arguing that transparency information is anomalous for users who feel the system is operating correctly and therefore undermines their confidence in the system. Design implications include a greater focus on what situations necessitate a transparent explanation as well as improved algorithmic error presentation.

5.2 Why Emotional Analytics?

Emotional analytics is a fruitful domain for transparency research for other reasons too. First it allows users to directly engage with personally relevant data. Other transparency

work has often asked users to evaluate algorithms in hypothetical scenarios where participants read about or watch algorithmic deployments and decisions [124, 123, 122]. Instead our aim was to have users directly experience the algorithm, evaluating it in situ as it made decisions about personally generated data [104]. A further critical aspect of emotional interpretation is that users are knowledgeable about their own feelings and experiences, allowing them to directly compare algorithmic interpretations with their own personal evaluations of those emotional experiences. This contrasts with other applications of smart algorithms, such as medical diagnostics, where the end user may not be a domain expert. As a non-expert, users might be less able to evaluate the results of algorithmic interpretations. In addition, emotion is highly variable between individuals and previous research demonstrates difficulty in accurately predicting emotion from text [120, 170]. One of the major challenges with intelligent systems is handling errors; explaining algorithmic prediction in this difficult domain of emotional analytics allows us to better understand how users make sense of output that contains errors.

5.3 Research System: E-meter

We developed a working system called the E-meter that uses textual entries to predict emotion. The E-meter (Figs 5.1, 5.2) presents users with a web page showing a system depiction, a short description of the system, instructions, and a text box to write in. The system was described as an “algorithm that assesses the positivity/negativity of [their] writing”.

The algorithm underlying emotion detection worked in the following way: each word that was written by the user was checked for its positive/negative emotion association in our

model. If it was found in the model, the overall mood rating in the system was updated. This constitutes an incremental linear regression that recalculates each time a word is written.

5.3.1 Machine Learning Model

As we outlined in the background, current processes for explanation of inscrutable models such as deep neural networks involve approximating the inscrutable model by a simpler, often linear, model [216]. Therefore, we focus on a linear model so that our transparency can be operationalized in a way that is faithful to current research, in examining a potentially explainable model.

Emotion predictions for users' experiences were generated using a linear regression model trained on text from the EmotiCal project [92, 191]. In EmotiCal, users wrote short textual entries about daily experiences and evaluated their mood in relation to those experiences. This data gave us a gold-standard supervised training set on which to train our linear regression. We trained the linear regression on 6249 textual entries and mood scores from 164 EmotiCal users. Text features were stemmed using the Porter stemming algorithm (Porter, 1980) and then the top 600 unigrams were selected by F-score, i.e. we selected the 600 words that were most strongly predictive of user emotion ratings. Using a train/test split of 85/15 the linear regression tested at $R^2 = 0.25$; mean absolute error was .95 on the target variable (mood) scale of (-3,3). In order to implement this model on a larger range for the E-meter, we scaled the predictions to (0,100) to create a more continuous and variable experience for users. The mean absolute error of our model indicates that the E-meter will, on average, err by 15.83 points on a (0,100) scale for each user's mood prediction.

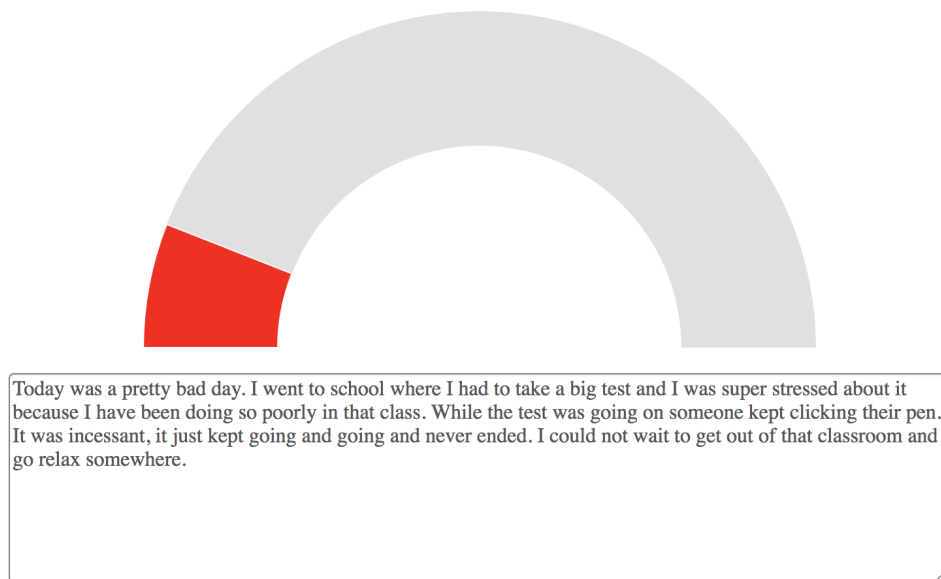


Figure 5.1: E-meter Document-Level Feedback Condition

5.3.2 Version 1: Document-level:

As users wrote, the E-meter showed the system's interpretation of the emotion of their writing. If the overall text was interpreted as positive, the meter filled the gauge to the right and turned more green (Fig 5.2); if the text was interpreted negatively, the gauge was emptied to the left and turned more red (Fig 5.1). This feedback represents the coarse and global feedback that many machine learning systems currently display. These systems give an overall rating but don't allow the user insight into the detailed workings of the algorithm.

5.3.3 Version 2: Word-level

In contrast with the Document-level version, the word-level condition provided fine-grained transparency. We operationalized transparency by highlighting the mood association

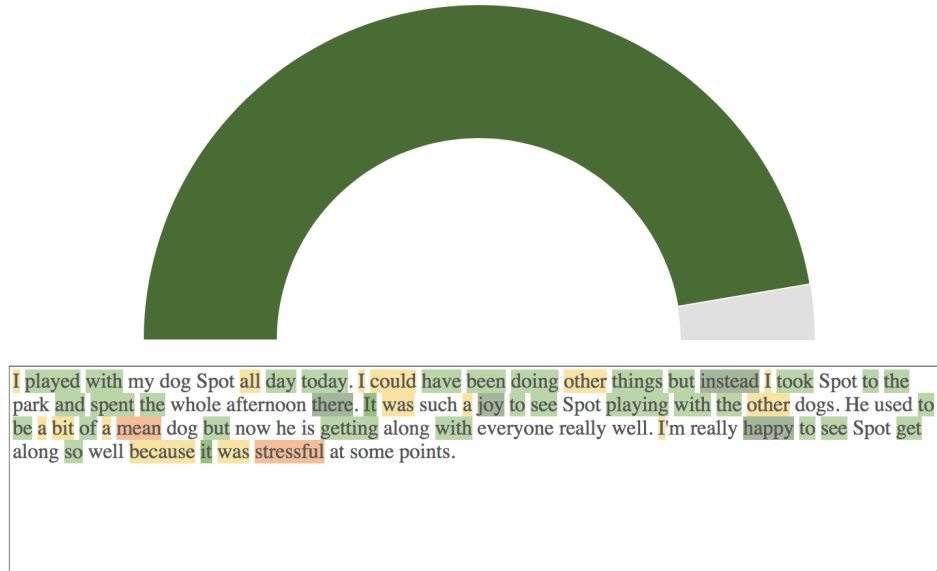


Figure 5.2: E-meter Word-Level Feedback Condition

of each word in the model; if a word is highly associated with a positive mood then it will be highlighted green, a word associated with a negative mood will be highlighted orange or red. The word-level version showed immediate incremental feedback of how the system interpreted each word as the user types. In this word-level condition, individual words are highlighted and color coded according to how the underlying algorithm interpreted that word's affect. This incremental feedback allows users to see how each individual word they wrote contributed to the overall E-meter rating. Furthermore, words remained highlighted as users continued to type allowing them to continue to assess their contribution to the overall score.

This form of transparency offers users insight into the underlying word-based regression model driving the E-meter visualization; it depicts how the regression model correlates each word with positive or negative emotion to arrive at an overall weighting for the entire

text that the user has entered. The fact that the visualization is persistent also allows users to reexamine what they have written, reconciling the overall E-meter rating with the fine-grained word-level connotations.

We could have operationalized transparency in other ways. Other researchers have operationalized transparency through natural language explanations [105] and diagrams [123]. However, in our case we can convey the majority of the the system operation through word highlighting. In addition, our operationalization allow the answering of counterfactual questions, an important part of explanation [138, 216]. Users can interactively change the prediction by inserting more text, allow them to quickly test their counterfactual hypotheses. Highlighting the text is a non-intrusive way of conveying to the user what drives the algorithm and gives direct clues about the underlying linear model. In addition, by varying the colors of the highlighting we also show how the model is interpreting the specific words.

5.4 Study 1

Our first exploratory study aimed to understand the processes by which participants make sense of a complex algorithm that interprets their emotions.

5.4.1 Method

5.4.1.1 Users

Twelve users were recruited from an internal participant pool at a large United States west-coast university. They received course credit for participation. Participants average age

was 19.54 years (sd=1.52) and 7/12 identified as female. This study was approved by an Institutional Review Board.

5.4.1.2 Measures

All survey questions requested Likert scale responses unless stated otherwise. Participants were asked about their experience with the E-meter including: “Select the number of times you looked at the visualization while you were writing” and subsequently “If you looked at the visualization more than once: Rate the extent to which looking at the visualization impacted or did not impact your writing.” We next probed user evaluations of system accuracy and their trust in the system: “How accurate or inaccurate did you find the E-meter?” and “How trustworthy or untrustworthy did you find the E-meter system?” In addition to these questions, we used a shortened version of the Psychological General Well-Being Index (PGWBI) to screen for mental health before participants began the study [82].

Additionally, we assessed users’ perceptions of the emotion of their writing: “How positive or negative did you feel our writing was?”, as well as the system’s evaluations of their writing: “How positive or negative did the E-meter assess your writing to be?” We used the absolute difference of these two measures to calculate an aggregate measure of expectation violation. If the E-meter were perfect, it would always predict exactly how the user felt and expectation violation would be 0. If a user felt that their writing was “Strongly Negative” (1) but the E-meter rated it as “Slightly Negative” (3) then the user’s expectation violation would be 2.

5.4.1.3 Procedure

The participants were randomly divided into one of two conditions. Both groups were given document-level affective feedback from the E-meter scale as shown in Fig 5.1.

Condition 1: Six participants received real-time incremental word-level feedback about the algorithm's interpretation of their affect as they typed each word.

Condition 2: The other six only obtained word-level feedback after they had finished the writing task; these users explicitly requested word-level feedback by clicking a button labeled "How was this rating calculated?".

The researcher explained the experiment and think-aloud procedure, demonstrating a think-aloud on an email client. The researcher asked participants to "Please write at least 100 words about an emotional experience that affected you in the last week." In cases where the participant had trouble thinking aloud, they were prompted to speak. After the think-aloud writing exercise, the experimenter conducted a semi-structured interview that included an on-screen survey that the participants continued thinking aloud as they answered. After the survey, participants in the word-level feedback condition 2 were presented with the final state of the E-meter, exactly as they saw it when they finished writing. Participants in the initial document-level condition 1 saw exactly the same screen with an added button labeled "How was this rating calculated" which they pressed to reveal word-level highlighting. Finally, all participants were presented with a printed version of the final E-meter state with which they marked up to indicate errors. The entire process took around 50 minutes.

5.4.1.4 Analysis

Interviews were recorded using both audio and screen-recording. Two interviews (one from each condition) were not audio recorded, thus only the remaining 10 are used for the analysis. Recall RQ1 for this study: When does transparency help versus hinder users' perceptions of complex systems? We analyzed the interviews with RQ1 in mind; responses were coded using theoretical thematic analysis [22]. We describe the major themes related to RQ1 below.

5.4.2 Study 1 Results

Participants engaged meaningfully with the feedback from the E-meter—all participants consulted feedback at least once and 8/12 of participants consulted it 'more than 5 times'.

Document-Level Feedback Results in Inaccurate Mental Models: Only half of the participants receiving document-level feedback formed accurate mental models of the system. Others with document-level feedback expressed confusion about how the algorithm operated, which in turn negatively affected their system perceptions. Participant 10 expressed this confusion and how it led them to consider the system as inaccurate: P10: I don't have a way of interpreting it, I wouldn't know if it's good or bad or what I'm writing is negative or positive. . . . it's inaccurate cause it doesn't seem to portray the true emotional state of what I wrote, and untrustworthy cause it doesn't give off the right feedback, it doesn't allow me to interpret it correctly.

Participant 11 who also received document-level feedback considered the E-meter may be working entirely off of "tone in my voice" or at random "...this could have been a whole

fake fluctuation.”

Word-level feedback Promotes More Accurate Mental Models: When we later showed word-level feedback to these document-level participants, their confusion seemed to dissipate, leading them to form more accurate mental models about how the algorithm worked. In this quote from Participant 10, note the stark difference from their prior quote above; they now are reassured that the system is actually working and make excuses why the system generated an incorrect rating.

P10: “it goes word by word, it tries to take positivity and negativity from each word, I don’t think it really goes by context of what I’m writing much more than the word itself, so saying something like ‘my best friend getting arrested’, it’s definitely in a negative light, but because I mention ‘my best friend’, it kinda took it in a positive light.”

In contrast, many participants who received word-level feedback throughout formed accurate mental models initially when using the system. Participant 5 said: P5: I think it tags certain words, for sure, with values probably 1-4, green, yellow, orange, red, or nothing, that there are certain words that are programmed into it. ... Maybe there’s words that it knows to look for across the system, like ‘death,’ ‘burial,’ negative words.. ‘celebration,’ ‘party,’ it can pick up on across the whole thing.

Word-level transparency information therefore seemed to help both groups by promoting insight into how the algorithm was making its evaluations, either in real-time or retrospectively. These observations suggest overall benefits for word-level information, confirming our initial expectations about the value of providing this type of transparency feedback. Despite these benefits, to our surprise, we also observed some negative effects for word-level feedback,

which on some occasions seemed to undermine some participants' views of the algorithm.

Participant 3 when examining the word-level feedback after they had written their text noted "...honestly I feel like I don't see a pattern at all and it's kind of bugging me."

Participant 8 first used the document-level feedback system and formed a mental model of system operation. Upon being shown the word-level feedback they began to question their established prior model; this resulted in worse perceptions of the system: P8: "Yeah, I'm actually not sure what—I don't know if it's just as simple as positive versus negative words. I had a solid theory before, but it's falling apart... Well I just assumed that some words would be coded as positive or negative and then it would just like do a ratio of those two."

Overall, it seems that word-level transparency offered people a useful heuristic, but this heuristic could be undermined by closer analysis.

Expectation Violation Predicts Accuracy Perceptions: We also wanted to quantitatively check whether participants' evaluations of accuracy related to their expectation violation. As expected, we see a strong correlation between the amount of expectation violation that users experienced in the visualization and their overall perception of the system's accuracy ($r(11) = -0.748, p = 0.005$). In other words, users judge the system as accurate if the system's interpretation is consistent with the user's evaluation of their writing.

5.4.3 Study 1 Summary

Transparency feedback seemed to provide a useful heuristic in helping users form working mental models, but closer scrutiny and perceived errors may undermine confidence in the system. When users felt the system was inaccurate in the document-level condition, they

were reassured later by word-level feedback showing how the system actually worked. However, we also see positive system perceptions undermined by the word-level feedback when users had established mental models. We set out to explore these seemingly contradictory effects in a larger scale quantitative study. Again, we compared both forms of feedback, but we also wanted to more directly explore potentially differing effects of word-level feedback in relation to expectation violation. How might word-level feedback both (a) help users who were unclear about how the algorithm operated while at the same time (b) reduce the confidence of users who already had a working theory that was subsequently undermined when they were confronted with word-level errors?

5.5 Study 2

5.5.1 Method

We used the same system and conditions as Study 1 with one important difference. While users in the Document-Level feedback condition in Study 1 eventually saw the transparent Word-Level feedback after they had completed their writing, in Study 2 the conditions are entirely separate, Document-Level users never see Word-Level feedback. Users were randomly divided into Word-Level and Document-Level conditions and instructed to write 100 words about an emotional experience in the E-meter system.

5.5.2 Users

We recruited 41 users to test the E-meter system who had previously passed a short mental health screening (PGWBI) [82]. Users were recruited from Amazon Turk and paid \$3.33. The evaluation took 13 minutes on average. This study was approved by an Institutional Review Board.

5.5.3 Measures

We asked the same questions as Study 1 with the addition of the following questions. “Please name 2 or more things you like about the system” and “Please name 2 or more things you dislike about the system”. “Please give 2-3 ways the algorithm affected your writing”, “Imagine that you were given personalized tips on how to improve your mood based on what you wrote. Would you make use of such suggestions?”, “Please explain how do you think the system judges your writing.”, “Did you experiment with or manipulate your writing to test how the system was working or how accurate it was? If so, how?”, "If you have any additional feedback from your interaction with the E-meter, please detail it here."

5.5.4 Study 2 Results

Participants followed the instructions to write at least 100 words, mean=107.74, sd=14.8. As shown in Fig 5.3, the majority of users across conditions found the E-meter to be “Accurate” or “Very Accurate” with the median being “Accurate”. Fig 5.4 shows that users found the E-meter to be “Moderately Trustworthy”. As in Study 1, we calculate a user’s expectation violation in relation to the system’s overall emotion rating. We find that transparency and ex-

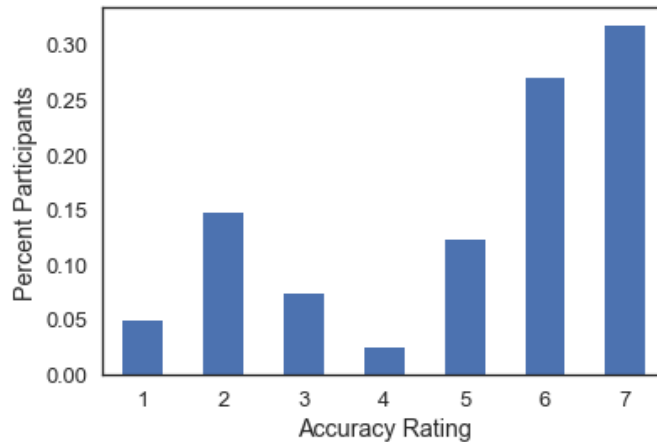


Figure 5.3: Participants Found the E-meter Accurate

pectation violation interact in a complex manner. We see a strong negative correlation between expectation violation and accuracy in the document-level group, ($r(21) = -.898, p < .00001$) confirming Study 1. In other words, with document-level feedback, when the system behaves as the user expects then it is perceived as accurate. However, this correlation between expectation violation and accuracy perceptions disappears in the word-level condition: $r(16) = -0.175, p = 0.488$. The relationship between expectation violation and accuracy perceptions is clearly more complex in the presence of word-level transparency.

We modeled the effects of the different types of transparency more systematically using linear regression predicting user accuracy perceptions as dependent variable (see Table 5.1). Given that Study 1 indicated people may respond to transparency differently based on expectation violation, independent measures in the regression model include expectation violation, condition, and an interaction between them. The overall regression was highly predictive $R^2 = .548, p < .0001$. As expected, both transparency and expectation violation are associated

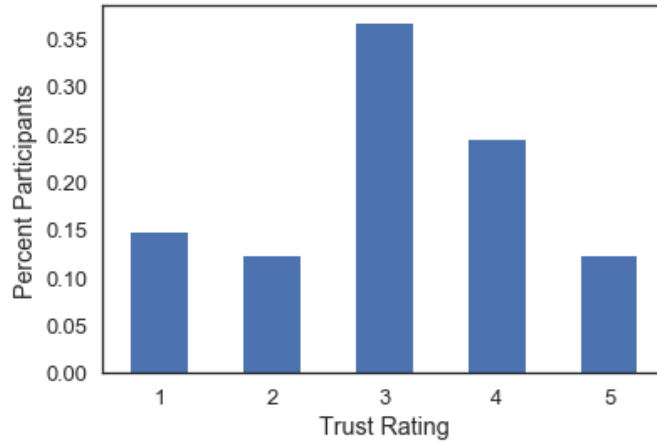


Figure 5.4: Participants Found the E-meter Moderately Trustworthy

Variable	Coefficient	Std. Error	p-value
Intercept	6.991	0.377	< 0.0001
Expectation Violation	-1.736	0.601	< 0.0001
Condition	-1.406	0.198	0.007
Condition * Expectation Violation	1.057	0.441	0.022

Table 5.1: Effects of Transparency on Perceived Accuracy. ($R^2 = .55, p < .0001$).

with perceived accuracy.

Counterintuitively, adding word-level transparency has an overall negative effect on perceptions of accuracy. However, this overall effect depends on expectation violation, as indicated by the interaction term in the regression. We depict the interaction in Fig 5.5, which shows that when compared with document-level feedback, word-level transparency has the expected positive effect when expectation violation is high. Confirming our qualitative results from Study 1, people using the word-level version of the system show higher levels of perceived accuracy

when their expectations are not met. However, the effects of word-level transparency are negative for lower levels of expectation violation, when compared with document-level feedback. Thus, word-level transparency is unhelpful when people perceive the system to be accurate.

One possible explanation could be that users in the Word-Level condition were consciously modifying and going back to edit their writing in order to achieve an accurate overall rating. Recall that we asked users whether they had modified their writing according to the feedback from the E-meter. Seventeen of our 41 users said that they changed their writing in order to influence the E-meter's response. However, this did not seem to modify perceptions of accuracy of the E-meter. Adding a binary variable for modifying writing based on the feedback into our above regression (Table 5.1) was not significant $\beta = -0.23, p = .61$. It seems that users' accuracy perceptions are not meaningfully affected by testing the E-meter.

To understand these effects further, we explored participant's qualitative responses to better understand the interaction between transparency and expectation violation.

Document-Level Feedback Results in Vague Mental Models: Most users in the document-level feedback condition were focused on major shifts in the movement of the meter in relation to their writing. For some of these users, document-level feedback was consistent with their expectations, helping them to form a global, if vague, model of how the algorithm was operating. For example, P24 wrote about how the accuracy of the system related to the system meeting their expectations: "I was writing about a negative topic and it continued to read in the negative state. The more upset I was writing, the further the dial went into the red." Users for whom the document level feedback matched their expectations maintained high confidence that the system was accurate.

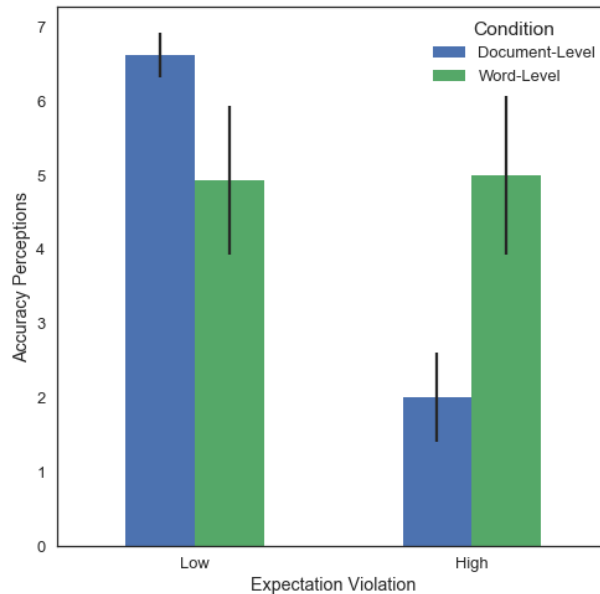


Figure 5.5: Expectation Violation and Transparency Condition Interact to Form Accuracy Perceptions

In contrast, other document-level users drastically lowered their confidence in the system’s accuracy when they felt that the system outputs contradicted their expectations. P4 felt that the system was inaccurate overall but could not form a clear hypothesis about exactly why it was failing: “It was highly inaccurate because the experience was clearly a negative one, I specifically explained how awful I felt, I don’t think that it could measure the sentiment of what I’m writing.” Others also thought that the system was inaccurate, but the absence of transparency led them to speculate about other ways it could be working. Participant 11 said “it looked at length and speed of what I was typing”; Participant 19 concurred, saying “i thought it was only reacting to my WPM [words per minute]”.

Overall, document-level users are confident in system operation if their expectations are met. If their expectations are violated, they seem to doubt the system is working as indicated

and, lacking any reassurance, this undermined their confidence in the system.

Word-Level feedback effects depend on expectations: Users in the word-level feedback condition formed clear mental models of how the system was working in rating individual words to arrive at an interpretation. However, even though these users seemed relatively confident about the algorithm's operation, they were often distracted or undermined by the system's interpretations of particular words. Participant 28 explained their rating of the system's accuracy as being downgraded by specific errors: "It went way down when I typed the word "mad" but that was only a small part of the whole situation. The words that were good or bad seemed kind of arbitrary too." Similarly, participant 30 said: "I wrote, 'I was not thrilled' which is a negative statement, but this meter took the word 'thrilled' as a very positive thing."

These examples suggest that, paradoxically, word-level errors might undermine some participants whose expectations are met and already have a good working theory of the algorithm's operation. To assess this further we examined cases of low expectation violation (instances of where users rated the E-meter as within 1 point of their own evaluation of their writing). Overall for these users the system is operating as expected. We analyzed these low expectation violation users to see why presenting word-level transparency information reduced perceptions of system accuracy. For these users, word-level transparency seems to create more questions than it answered; the additional information provided by the word-level highlighting seemed to confuse rather than clarify. One participant noted that while the system's final negative rating was consistent with their overall judgement of their own writing, the highlighting didn't make sense "...because the rating did not correspond to the number of identified words"; this user also noted "It gave a positivity rating of 1 even though it only highlighted one or two

words as red.” The word-level highlighting revealed to other users that the model worked in a different way from the user themselves. While users in the document-level condition were not able to discern this as a problem, word-level feedback users took issue with this. Participant 41 said: “The key is to measure the overall emotional tone of the passage and it seems to fail at this.” Participant 36 said this simply: “I disliked that it cannot understand context.” For these users, transparency revealed that the algorithm did not conform to their mental models of the task.

We also examined users who had the opposite experience. We looked at users who initially felt that the algorithm was violating their expectations, but transparency seemed to help them. For them word-level transparency seemed to provide reassurance and explanation of the system behavior. One user, participant 27 seemed to note that the system was trying, even if it violated their expectations: “I think that it was measuring words i used and rated them almost correctly.” In the same vein, Participant 30 said “Even though it got several individual things wrong, I think it actually did a good job on the whole.”

Overall, word-level feedback seemed to have somewhat contradictory effects depending on the user’s assessment of system performance. For users who felt that the system was behaving appropriately, noticing word-level errors and non-conformance to their mental models undermined their views of system operation. For others who were less sure about overall system accuracy, word-level feedback had the opposite effect, as it boosted confidence in the system. These data are consistent with Fig 5.5 showing that compared with document-level feedback, word-level feedback reduces accuracy judgments in low expectation violation users, but increases it for those who have high violations.

5.5.5 Study 2 Discussion

In this study we presented both qualitative and quantitative data showing that algorithmic transparency has complex effects that depend on users' expectation violation. Word-level transparency users with the most violated expectations had better perceptions of the E-meter's accuracy compared to their document-level counterparts. However, users in the word-level condition were less likely to regard the system as highly accurate when it did not violate their expectations.

Study 2 elucidates how people's attitudes and feelings towards a system change but does not clarify how behavior changes. We follow up with another study that measures user behavior in this system to provide further evidence about when users want to view transparency.

5.6 Study 3

5.6.1 Method

We use the same E-meter system as the previous conditions but in a more naturalistic deployment. In this study, participants used the E-meter system twice. The E-meter in this study generally provided document-only feedback condition but would display word-level feedback 3 different times when the user was writing; this feedback displayed when the user was 1/3rd through the writing task, 2/3rds, and then also when they were almost complete. The user answered a very short questionnaire, to assess expectation violation before seeing the word-level transparency, they then viewed the transparency as long as they wished—during which point they could not continue writing, and then pressed a button labeled “Press this button to

turn off highlighting and continue writing” which they pressed to continue the task.

5.6.1.1 Users

We recruited 53 users to test the E-meter system who had previously passed a short mental health screening (PGWBI) [82]. Users were recruited from Amazon Turk and paid \$3.33. The evaluation took 17 minutes on average. This study was approved by an Institutional Review Board.

5.6.1.2 Measures

The system was implemented to record how long the users took to examine the transparency before they continued with the writing task. We will refer to this timing as transparency view time. Before the users viewed the transparency they were asked how far the current document-level rating was from the user’s own evaluation of their content, we refer to this as the in-the-moment expectation violation, and this is measured multiple times per session. After seeing the transparency users were asked about how their understanding changed due to viewing the transparency. Finally after using the system twice, participants were asked about their overall perceptions of the E-meter’s accuracy, trust, and their mental model of the system’s operation.

5.6.2 Results

Confirming the result of studies 1 and 2 we again found that word-level feedback promotes more accurate mental models. In this study 52 out of 53 participants had accurate

Variable	Coefficient	Std. Error	p-value
Intercept	8.25	0.16	< 0.0001
Expectation Violation	0.07	0.03	0.03

Table 5.2: Effect of Expectation Violation on Transparency View Time

mental models of the system where they understood that the system rated individual words as positive or negative and used these to calculate an overall rating. One example of this accurate mental model is given by Participant 53: “It uses key words in the writing and the frequency and ratio of these word to calculate how positive or negative a passage is”. The one remaining participant who did not have an accurate mental model felt that the word highlighting was assigned at random and the meter moved randomly also.

We find that participants who experience expectation violation spend more time examining transparency than those who experience less or no expectation violation. This is a within subjects experiment; we measured expectation violation 3 times per trial with 2 trials per experiment so we analyze the experiment using a linear mixed model with crossed random effects from the R package lme4. The random effects are to control for intra-participant variance and intra-position variance (essentially order effects) and these are crossed because each participant experiences each position. The fixed effect in the model is the in-the-moment expectation violation, because we expected that users would spend more time examining explanations when their expectations are violated. Thus, the dependent variable is the measured time the user took with the explanation. Before fitting the model, we take the natural log of the time variable because it is distributed exponentially.

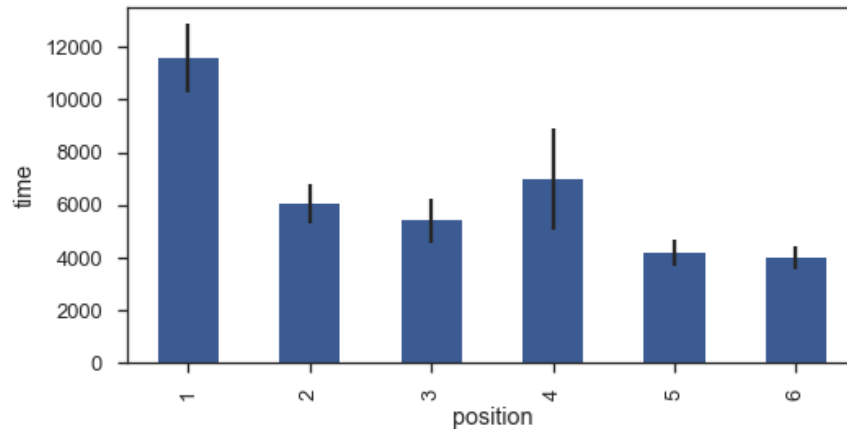


Figure 5.6: Transparency View Time By Explanation Position

The model is presented in Table 5.2. We note that the inclusion of the random effects in the model are significant at $p < .0001$ as determined by a likelihood-ratio test comparing the given model with a comparison model that drops either the participant or position random effect. This means there is significant variation between participants and also between the order effects within each participant.

Additionally, expectation violation has a significant effect on in-the-moment expectation violation. A one unit increase in expectation violation is associated with a 7.25% increase in transparency view time. Given that expectation violation ranges from [0,3], the maximum expectation violation is associated with a 23% increase in transparency view time. The model itself overall is very predictive with $R^2 = .48$.

We were interested in further examining the effects of order on transparency view time. As noted above, including the random effect for position was significant, indicating that there are significant differences in transparency view time based on what part of the task the user

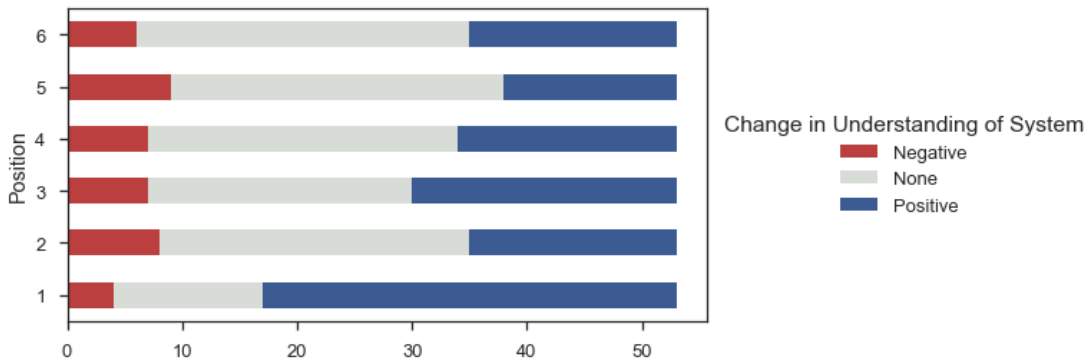


Figure 5.7: User Improvement in Understanding By Explanation Position

was in. In Fig 5.6 we plot the transparency view time according to position the user saw. The first position, where the user first sees the transparency, greatly exceeds all other view times. The fourth position also has a higher mean than the surrounding positions though the variance here is very large; recall that this fourth position is the first time the users sees the transparency for the second example of their writing. Another thing to note is that the transparency view time decreases over time; we see that users spend less time with transparency for subsequent viewings for the same example, but also decreasing between examples over time.

Next we examine how the users understanding changes over time. Recall that we have an ordinal 7-point Likert item outcome for understanding change from ‘Strongly Decreased My Understanding’ to ‘Strongly Increased my Understanding’ that users answer after each time they see the transparency. We find that user responses are significantly different using a Kruskal Wallis Rank Sum test $\chi^2(5) = 16.94, p < .01$. Visually examining the differences shown in Fig 5.7 shows that the first time users view the transparency there are improvements in understanding for the majority of people. Subsequent viewings of transparency are more

neutral overall, a sizable contingent of users still gain understanding but many feel their understanding is not changed either way by the transparency. A very small amount of users find that their understanding changes negatively when viewing the transparency.

5.6.3 Discussion

We examined when users want to receive transparency in a naturalistic experiment. Our primary outcome of interest is how long users examine transparency when they are shown it. We find that users spend significantly more time with transparency when their expectations are violated. Additionally, we see strong effects of order in how much time users spend with transparency. On the first viewing of transparency, users spend significantly more time. This time spent with transparency decreases over time as are likely forming more accurate mental models of the system. There is some indication that showing transparency on new examples (position 4 in Fig 5.6) may lead users to spending more time with the transparency but further research is needed.

5.7 Overall Discussion

Given their widespread deployment, it is imperative that we derive new theories and design approaches to improve users' understanding of intelligent systems. Three studies show when fine-grained algorithmic transparency is needed during user interaction with an intelligent system. Study 1 helps explain some inconsistencies reported in prior research into the benefits of transparency for improving user understanding and trust. Study 1 qualitatively showed the

complex relations between transparency and expectation violation; some users are reassured when they view word-level transparency; it shows them that the system is working even if it did not generate exactly the interpretation they expected. However, for other users, transparency seemed to undermine their experience; while overall the system worked as they expected, they also detected specific word-level feedback that seemed anomalous. Study 2 resolves these apparently contradictory observations. We show that there is no one-size fits all solution, as responses to word-level transparency depend on users' level of expectation violation.

Finally, we provide more systematic evidence supporting this interpretation in Study 3. Study 3 indicates that users spent more time viewing transparency information when their expectations are violated, indicating that transparency is addressing their lack of comprehension about how the system is operating. Additionally, Study 3 provides further evidence for the usefulness of transparency in facilitating the creation of accurate mental models—users looked at transparency less over time and gained more understanding when looking at transparency early on. This indicates that the users looked at the transparency for less time because they already knew how the system was working. Overall these studies indicate that intelligent systems should focus on showing transparency in two situations. One, when users expectations are violated. Second, early on in the users usage of the system so proper mental models are facilitated from the start.

5.7.1 Limitations

The current chapter explores one algorithmic domain, emotion regulation and clearly other contexts need to be explored. Furthermore, our deployment of a working algorithm meant

that results were obtained for situations where our algorithm generated moderate numbers of errors, and future research should evaluate contexts where there are different levels of errors, including extremes of multiple versus few errors. Additionally, while users generated their own data while using a real working system, results were not directly used to inform other aspects of user's personal behavior such as emotion tracking or emotion regulation, so the costs of system errors were low. While this is appropriate for exploring understanding of initial algorithms with moderate error rates, future work might explore user system models and trust in more high stakes contexts. Also, our work explores one aspect of transparency namely a dynamic visualization of the algorithm, and there are many other ways to depict how an algorithm operates including verbal explanations, concrete user exploration and so forth [105, 109, 124, 173].

5.7.2 Synthesizing Contradictory Results

Our results also serve to synthesize and explain previous research on transparency and intelligibility that has generated highly mixed and sometimes paradoxical results. On the one hand, trust can be built through transparency [56, 105, 124]), even to the point of overconfidence [69]. At the same time, however, trust can also be undermined by transparency [105, 123, 143]. Our findings indicate that this prior work may be rationalized by attending to expectation violation. For users who have low confidence in a system, with little idea how it works, then transparency can help form working system models, thus boosting confidence and trust. In contrast, trust can be undermined if users have a working theory of the system but are exposed to anomalous behaviors such as system errors.

5.7.3 Social Communication Theories to Support Transparency

The interactions between transparency and expectations are also consistent with social analyses of when explanations are needed. Theories of human communication (e.g. Garfinkel, Grice) argue that explanations are occasioned, i.e. that explanations are only provided on an ‘as needed’ basis when a situational expectation is not met [70, 81]. Grice embodies this in the "Maxim of Relation" where one tries to only say things that are relevant to the discussion. This coincides with Garfinkel’s work where explanations are considered anomalous when provided without reason. According to these theories, transparency is therefore occasioned for expectation violations, making it contextually appropriate to provide an account for why system behavior is unexpected or unusual. But if expectations are met, i.e. the system and situation are perceived as going according to plan, then an explanation is anomalous and contextually marked, potentially reducing user confidence: ‘everything is going fine so why are you providing this unnecessary information?’. Results are also consistent with the elaboration likelihood model [161]; which suggests that users might be generally happy to operate with imprecise working heuristics about how an algorithm operates, only invoking complex analysis when the algorithm behaves anomalously.

Our results draw attention to an important and often overlooked aspect of transparency. Most research has focused on questions of how to explain algorithmic operation, e.g. using approximate methods to explain neural nets using methods that everyday users will comprehend. In contrast our focus here has been on better understanding of when to deploy transparency, as we show that providing detailed information about an algorithm’s operation

can be counterproductive for users who have a working theory of system operation. Other work suggests that users do not want to be exposed to detailed algorithmic explanations when they are cognitively overloaded [94], suggesting a need to develop more systematic accounts of when algorithmic explanations are occasioned and useful.

Of course, deciding when to provide algorithmic explanations also gives rise to ethical concerns: if users are operating under false working assumptions about an algorithm then we need to expose and counteract these. Again, this suggests the need for an increased research focus on understanding users' working models of systems allowing us to diagnose when these are accurate, and when we need to intervene. This area is extraordinarily complex however given applications where positive placebo effects have been obtained using algorithms that falsely inform users that their stress levels are low [39]. In this case, an inaccurate user model and imperfect algorithmic understanding has beneficial outcomes.

5.7.4 Design Implications for Intelligent Systems

Our results also have implications for design. Whether or not transparency is beneficial depends on the timing of the transparency. In situations where a user's expectations are violated, users will spend more time looking at transparent feedback and also generate a more positive view of the system. In situations where expectations are not violated, users may generate a more negative view of system because transparency can reveal system errors. However, Study 3 indicates that users will not look at transparency as closely when expectations are not violated, so there is room here to examine interaction patterns aside from always-on transparency and their role in helping ameliorate the problems transparency produces in conditions

of low expectation violation.

We find that in some contexts, transparency decreases perceptions of accuracy. One reason for this decrease seemed to be when transparency revealed that the way the system was operating was different from a user's mental model of the task. For example, users took issue with the fact that the system did not seem consider context surrounding some words. Considering context was impossible in our machine learning model given that we were using solely individual words as features. Decisions made when training a machine learning algorithm including feature selection and algorithm choice necessarily constrain how transparency can be operationalized. This demonstrates that algorithmic decisions have direct user experience impacts in transparent applications. Current research indicates that we should present transparency in ways that bridge the gap between user's mental models and expert mental models of the task [57]. Our work dovetails with this, demonstrating that correct choices need to be made concerning the machine learning models to even allow for transparency that bridges this gap.

5.7.5 Presenting Errors in Intelligent Systems

Our results also draw attention to the critical problem of error presentation. All machine learning systems generate errors and while users need to be aware of this, our data suggests that in some cases users are overly focused on errors even when these are relatively infrequent and the system is operating well overall. This may be consistent with psychological theories suggest that people are generally poor at evaluating probabilities [100]. In any case, it indicates the need for much more research on error visualization as well as algorithmic success. Other designs suggested by our work could address the effects of errors undermining

confidence. For example, we might only show highlighting on words that the system is very confident will be positive or negative in any context. Other users wanted to know about the relative weighting of positive versus negative words and the algorithm might provide more explicit models of this. These design improvements may improve perceptions of accuracy across the board allowing users to generate more stable models and reduced questioning. However, given the ubiquity of errors in all intelligent systems, much more research is needed to explore how errors might be presented and explained in ways that do not undermine the development of accurate working models of system operation.

For consumer facing applications it may be beneficial to operationalize transparency so as to promote user confidence in approximate system models allowing underlying complexity to be hidden from the user. In other contexts, however, we may want to have users very carefully evaluate a system prediction. For example, we may not want doctors to simply defer to the predictions of a medical decision support system. We may instead want to lead them to question their underlying model while using the system. For such cases, it may be beneficial to operationalize transparency to promote careful consideration with moderate amounts of expectation violation. This may be a difficult balance to achieve between questioning and validating a system model so that while skeptical users would continue to trust and use the algorithm.

Overall our results offer a new perspective on when users want explanations from intelligent systems, this perspective matches with what we would expect from social science theory. We also suggest new ways to design intelligent systems that center the human desires for explanation.

Chapter 6

How do users want to interact with transparency?

6.1 Introduction

I have reviewed many reasons that transparency is needed for truly human-centered intelligent systems; they include increased understanding in the system [108, 197], improved trust in system decisions [105], clarity about the system optimizing intended functionality [86]. In the last chapter, I created design implications about when transparency is needed in an intelligent system. In this chapter, I study how we should operationalize transparency to fit user expectations. While there are many calls for transparency [30], it remains unclear exactly how to enact calls for transparency in practice. Extensive research about operationalizing transparency has emerged in the machine learning community but no clear consensus has resulted [1, 57, 216]. Deciding exactly how to implement transparency is difficult—there are numerous implemen-

tation trade-offs involving accuracy and fidelity. Making a complex algorithm understandable to end users might require simplification, which often comes at the cost of reduced accuracy of explanation [112, 173]. For example, methods have been proposed to explain deep neural network algorithms in terms of more traditional machine learning approaches, but these explanations necessarily present approximations of the original algorithms deployed [129]. These studies often approach transparency from a technical perspective, asking: “what is possible from an algorithmic standpoint?” rather than “what does the user need?”

However, some recent empirical studies attempt to examine the effects of transparency on users. But these studies reveal puzzling and sometimes contradictory effects. In some settings there are expected benefits: transparency improves algorithmic perceptions because users better understand system behavior [105, 108, 123]. But in other circumstances, transparency has other quite paradoxical effects. Transparency may erode confidence in a system, with users trusting it less because transparency led them to question the system even when it was correct [123]. Providing system explanations may also undermine user perceptions when users lack the attentional capacity to process complex explanations, for example, while they are executing a demanding task [24, 224]. Overall, these results indicate mixed evidence for the benefits of transparent systems. The above research suggests that we have yet to identify the appropriate interaction paradigms to present transparency. Machine learning research communities are forging ahead with foundational research on how to generate transparent systems [216] but studies often stop short of actually testing these systems with users [36, 7]. User evaluation is critical because, as we have seen, user reactions to transparency show quite contradictory results [24, 105, 123]. Some research suggests that the way we present transparency may account for

these contradictory results [69, 105]. Our research seeks to bridge this gap between generating explanations and user-centric presentation. In two studies, we explore users' direct reactions to a transparent personal informatics system that interprets emotions that users express in written text. The domain of emotional personal informatics is fruitful for transparency research for several reasons. Emotional analytics is perceived as a key application by many users [94]. And users are the ultimate experts on their own emotions, providing them with ground truth assessments about whether the system is correct. Further, making accurate predictions about emotions is also difficult, allowing us to examine a key challenge for intelligent systems: how to deal with system error.

We examine user preferences and reactions to transparency by comparing two versions of a working analytics system that predicts users' emotions from written text. Each version uses the same underlying algorithm. The first, transparent, version makes overall predictions about the user's affective state and supplements these overall predictions with incremental real-time feedback showing how the algorithm made these judgments. The second non-transparent version simply predicts the user's overall affective state, without underlying transparency feedback. We also examine the role of cognitive load in explaining these preferences and explore problems with current presentations of transparency.

We address the following research questions:

- RQ1: Do users prefer to use transparent systems? Do they prefer systems providing transparent feedback or those that simply present overall predictions? (Study 1)
- RQ2: Do cognitive load and distraction affect preferences for transparency? What other

factors influence reactions to transparency? (Studies 1 and 2)

- RQ3: How might we support effective transparency? (Studies 1 and 2)

6.1.1 Contribution

Much recent work on transparency has focused on technical explorations of self-explanatory systems. In contrast, we take an empirical user-centric approach to better understand how to design transparent systems. Two studies provide novel data concerning user reactions to systems offering transparent feedback vs overall prediction information. In Study 1, users anticipated that a transparent system would perform better, but retracted this evaluation after an experience with the system. Qualitative data suggest this may be because incremental transparency feedback is distracting, potentially undermining simple heuristics users form of system operation. Study 2 explored these effects in more detail suggesting that users may benefit from simplified feedback that hides potential system errors and assists users in building working heuristics about system operation. We use this data to motivate new progressive disclosure principles for presenting transparency in intelligent systems.

6.2 Research System: E-meter

We designed a system called the E-meter that tests users' reactions to a transparent system that actively interprets their own data. The E-meter system and protocol have been successfully deployed in multiple previous experiments [190, 192]. The E-meter predicts the emotional valence of a user's written personal experience. The E-meter is shown in Figures 5.1

and 5.2, showing non-transparent and transparent versions of the same algorithm. The system was described to users as an “algorithm that assesses the positivity/negativity of [their] writing”. It was built as a client-side web application using HTML5 and JavaScript, technologies that give us flexibility in deploying to various populations including in-lab studies and Amazon Mechanical Turk experiments.

There are a number of tradeoffs that need to be made when testing user reactions to transparency. In contrast to many previous hypothetical scenario-based studies, where users are presented with descriptions of a system along with its outputs [124, 123, 122] we instead chose to deploy a working system to better engage and elicit direct user reactions to personally relevant data. The system also provided real-time feedback which indicated to users that the system was authentic and not a deception [190]. But implementing a real-time transparent system imposes trade-offs regarding the performance of the underlying model. In particular, we needed to sustain performance across multiple hardware platforms. Therefore, we limited the complexity of our machine learning model and constrained our underlying feature set so that it could run in real-time within a browser.

6.2.1 Machine Learning Model

Little empirical work has examined how users respond to even simple operating models of transparency. This motivates our current approach, which we have successfully deployed in several prior studies [192, 190]. We chose a modeling approach that is accurate enough to be convincing, but simple enough to explain to end-users. We implemented a unigram-based regression model. While such a model may be less accurate than state of the art methods us-

ing deep neural networks [50], it provides a straightforward operationalization of transparency that is intuitively understandable while still appearing accurate and convincing to users [192]. And although the algorithm makes some errors, evaluating user reactions to these errors is an important element of our approach.

The emotion detection algorithm works as follows: each word written by the user is evaluated for its positive/negative emotion association in our model. If a word is found in the model, the overall mood rating in the system is updated. This constitutes an incremental linear regression that recalculates each time a word is written. In prior experiments, users judged the same algorithm to be accurate but not perfect; the median user response for accuracy was ‘Accurate’ (6 on a 7-point Likert scale with the highest rating (‘7’) being ‘Very Accurate’) [192].

Models were trained on text gathered the EmotiCal deployment [92, 191]. In that deployment, users reported and analyzed activities and emotions over several weeks by writing short textual entries about daily experiences and directly evaluating their mood in relation to those experiences. This data gave us a gold-standard supervised training set where users labeled their own descriptions for underlying affect. We trained the linear regression on 6249 textual entries and mood scores from 164 EmotiCal users. Text features were stemmed using the Porter stemming algorithm [165] and then the top 600 unigrams were selected by F-score, i.e. we selected the 600 words that were most strongly predictive of user emotion ratings. Using a train/test split of 85/15 the linear regression tested at $R^2 = 0.25$; mean absolute error was .95 on the target variable (mood) scale of (-3,3). As we noted above, previous users found the system to be accurate to their experience, judging median accuracy to be 6 out of 7 on a scale where 7 is ‘Very Accurate’.

6.2.2 System Versions

The underlying machine-learning model was evaluated by users for two different system versions: non-transparent and transparent.

Non-transparent version: As users wrote, the E-meter showed the system's overall, document-level, interpretation of the emotional valence of their writing. If the overall text was interpreted negatively, the gauge emptied to the left and turned redder (see Figure 5.1). If the text was judged to be positive, the meter filled the gauge to the right and turned more green. This continuous scale feedback represents the coarse global information that many machine learning systems currently display. Such systems give an overall rating but do not offer the user an insight into the detailed workings of the underlying algorithm.

Transparent version: In contrast, the transparent version provided fine-grained feedback about how the algorithm was making its prediction. This transparency was operationalized as word-level feedback; in this system version, the E-meter signaled the affective association of each word the user typed. Individual words are highlighted and color-coded according to how the underlying algorithm interpreted that word's affect (Figure 5.2). If a word is highly associated with positive mood then it will be highlighted green, whereas a word associated with negative mood will be highlighted red. This incremental feedback allows users to see how each individual word they write contributes to the overall E-meter rating. Furthermore, words remain highlighted as users continue to type allowing them to assess each word's relative contribution to the overall score.

This form of transparency offers users insight into the underlying word-based regres-

sion model driving the E-meter evaluation; it depicts how the regression model correlates each word with positive or negative emotion to arrive at an overall weighting for the entire text that the user has entered. The fact that the visualization is persistent also allows users to reexamine what they have written, reconciling the overall E-meter rating with the fine-grained word-level transparency.

Of course, we could have operationalized transparency in other ways. Other researchers have implemented transparency through natural language explanations [105] and diagrams [123]. However, in our case, we can convey key aspects of the underlying system through word highlighting. In addition, our operationalization allows the answering of counterfactual questions, an important part of explanation [138, 216]. Highlighting the text helps directly convey to the user what drives the algorithm and gives clear clues about the underlying linear model. In addition, by varying the colors of the highlighting we also show how the model is interpreting the specific words.

6.3 Study 1

Study 1 was intended to explore user reactions to transparent and non-transparent versions of the algorithm. For each version, we assessed user evaluations both before and after using the system as well as their perceptions of cognitive load.

6.3.1 Method

Participants experienced both versions of the E-meter system. After using each version, they were asked a series of follow-up questions. The study design was counterbalanced; half the users experienced the non-transparent system version first. Questions and procedure were carefully piloted and had been used before in multiple prior studies.

6.3.1.1 Users

We recruited 100 users who had previously completed a subset of the Psychological General Well-being Index (PGWBI) [82]. Following recommendations from an IRB panel, the PGWBI screening was intended to exclude users who might find the emotional reflection distressing. Users were recruited from Amazon Mechanical Turk and paid \$3.33. The evaluation took 14.68 minutes on average. Following prior methodological recommendations in [48], we eliminated 26 respondents based on their responses to open-ended questions, leaving us with a sample of 74 users. This study was approved by an Institutional Review Board.

6.3.1.2 Measures

Before actually using the system, participants saw simulations of both transparent and non-transparent versions of the system and were asked to predict accuracy for each. We simulated both versions of the system showing what happened as filler Latin text was typed. We used Latin text because we wanted high-level system comparisons that were not based on users' reactions to specific English words. Following the animations, we asked the predicted accuracy question "This program evaluates the positivity/negativity of emotional experiences

that users write about. How accurate or inaccurate do you think this program would be for you? The program works with English also.” Users then provided qualitative explanations for their ratings—“Please give 2 or more reasons for the accuracy ratings you made on the previous page.”

Participants then began their writing activity. Following a protocol deployed in prior work, users were presented with one of the two system versions with the instructions “Please write at least 100 words about an emotional experience that affected you in the last week.” After this system experience, users completed the TLX workload assessment [87]. Users then answered the questions: “How positive or negative did you feel your writing was?” (subjective affect), “How positive or negative did the E-meter assess your writing to be?” (system affect), “How accurate or inaccurate was the E-meter in its assessment of your writing?” (retrospective accuracy), “How trustworthy or untrustworthy did you find the E-meter system?” (subjective trust). These questions were all 7-point Likert items. For example, the subjective trust questions items were from “Very Untrustworthy” to “Very Trustworthy”.

Users then repeated this process for the other system version. After using both versions, users answered a final experience-based system preference question “If you were to use the E-meter again, which system would you prefer?”. They then supplied reasons for this: “Please give 2 or more reasons for the choice you made above”.

6.3.2 Results

System Perceived as Moderately Accurate and Trustworthy: Overall, the median user found the E-meter to be ‘Slightly Accurate’ and ‘Slightly Trustworthy’. Both retrospective

accuracy and subjective trust distributions were bimodal (due to the 7-point Likert item containing an infrequently used neutral item) and there was no difference between conditions ($p = .24$, $p = .41$). This replicates our previous work [192] where the median user found the E-meter to be ‘Accurate’.

Transparent version is predicted to be more accurate before usage: Before any hands-on experience with the system, participants generated predicted accuracy judgments for both the transparent and non-transparent system versions. We statistically compared the difference in these system evaluations using a paired t-test. Participants anticipated greater accuracy for the transparent system, $t(73) = 5.452$, $p = 0.022$, although the effect was small—means were 4.24 and 4.57 respectively. Qualitative user comments supported these anticipated benefits. On being asked to explain why they expected the transparent system to be more accurate, users drew attention to the benefits of transparent color-coded feedback giving a clearer sense of how the system was operating and boosting their confidence that the system was operating appropriately. Participant 73 wrote “The second one had a legend with it and actually changed the color of the words I would have written. . . . It was also more catchy and the colors stood out to me.” In the same vein, participant 50 wrote: “One meter is more transparent than the other. I can see how it works. I feel more confident in knowing exactly how it comes up with its answers. I tend to think it is more reliable.” Overall then, before actually using either E-meter version, users anticipated that a system offering word-level transparency would be more accurate.

Recall that after experiencing each system version, we asked users for retrospective accuracy and a final experience-based preference about which system version they would choose for future usage. User perceptions of retrospective accuracy with both versions of the

system highly correlate with their final experience-based system preference in a logistic regression model (p 's = [0.019, 0.0001]). Given only retrospective accuracy scores from both versions, we can predict the version choice with 69.5% accuracy in a 5-fold cross-validated test. Therefore, knowing users' predictive accuracy was higher for the transparent version, we would expect that this would lead to users ultimately preferring the more transparent version of the system.

After experiencing the system there was no preference for either system version:

However, this positive predictive evaluation of transparent system accuracy did not persist after the actual experience with the system when we analyzed final experience-based preferences. As we expected, many of those who preferred the transparent system after usage did so because it illustrated the inner workings of the system. Participant 72 said "I know what the first [transparent] version is doing. I cannot tell what the second version [non-transparent] is doing. Because the second version does not give real feedback, I cannot make an informed decision when writing if I should be using it or not." Participant 28 concurred, saying "I think it [the transparent version] provides more engaging feedback and helps me better understand the reasons it gives for the amount in the meter."

However, to our surprise, the benefits of transparency did not generalize to all users. After the experience with both systems overall users were evenly split in which version of the system they preferred to use in the future: fifty percent of participants (37) said that they would prefer the non-transparent version if they were to use the system again. The other 50% (37) chose the transparent condition. Consistent with these preference judgments, participants also showed no overall differences in trust after using the two different system versions ($t(73)=.910$,

$p=.343$). How can we interpret these findings? While users had better impressions of the transparent condition initially, those preferences disappear after using both systems. It is important to note here that although the only objective differences between the systems lay in their transparency, users seemed to treat them as operating quite differently. As we will see later, even though participants knew both versions of the system existed at the start of the experiment, they seemed to attribute different qualities to each version after experiencing them.

Overall, both the final system choice and trust showed no differences between system versions despite people being confident initially that the transparent version would be more accurate. What might explain these changed perceptions after usage? Role of Cognitive Load: One possible explanation for this changed perception is cognitive load. Transparency may demand attention and distract participants, in contrast to the non-transparent system version which presents less information [24]. Some participants' explanations for their experience-based preference seem to support this. These participants ($n=13$) cited the distracting nature of the word highlighting. Users called word-based transparent feedback "annoying" (P3) and "obtrusive" (P7). Participant 20 said that the non-transparent version was "a lot less distracting". P57 said, "The individual highlighting of the words was distracting during writing; I wouldn't have minded it as much if I could turn it on and off." However, these subjective reports were not borne out by our quantitative analysis of cognitive load as assessed by the TLX survey. A paired t-test comparing the overall TLX measures for both versions of the system indicated no difference in workload: $t(73) = -.05, p=.95$.

Reduced transparency may lead users to overestimate system capabilities: Another potential reason why users may prefer the non-transparent system relates to user infer-

ences about algorithmic capability. Our qualitative analysis of experience-based preference suggests that users who know less about the working of the system, seem to ascribe more advanced abilities to it. Nearly a quarter (24%) of users who chose the non-transparent E-meter as their preferred version stated that they preferred it because it took their overall writing context into account, incorporating information beyond simple lexical weightings. Participant 66 said, “I think the [non-transparent version] takes into account everything you are writing and makes a decision better than just by focusing on word choice.” Participant 17 concurred, saying “I like the second [non-transparent version] as it seems to focus on the whole and not each word.” While these non-transparent inferences are positive, they are also inaccurate. Recall that both systems use the same underlying machine learning model which uses solely individual word features.

What might explain this overestimation of system capabilities? One possibility is that non-transparent feedback can hide low-level errors from users. In contrast, many transparent condition users identified highlighted words they felt were misclassified, leading them to downgrade their system evaluation. For example, participant 40 chose the non-transparent version, justifying it by saying: “...the biggest reason is that the most negative thought I had was expressed by the word "isolated" in the text I wrote and the e-meter marked that one word as "Unimportant" I couldn't get past that.” Participant 70 said: “Some associations don't make any sense, while others do.” In contrast, non-transparent feedback did not expose these errors. If the algorithm was behaving consistently with their overall expectations, users in the non-transparent condition judged it very positively. This did not seem to be the case in the transparent condition, users judged the system more negatively even if the overall prediction was correct.

Together these observations suggest that error hiding and the absence of detailed information in non-transparent feedback leads some participants to form approximate but positive working heuristics about how the system operates.

6.3.3 Discussion

Our initial hypothesis was that providing detailed, transparent word-level feedback would be more helpful to users than non-transparent overall predictions. However, our first study unearthed some unexpected findings, showing that user interpretations of transparency feedback are far from straightforward. Consistent with our initial expectations and the prior literature on transparency [69, 124], users anticipated a preference for transparency before using the system. Users justified this preference by making arguments that such feedback would provide detailed incremental information about algorithmic decisions. But to our surprise, many participants did not retain this preference after using the system, at which point participants were evenly split between the systems in their trust and transparency preferences. As we had originally anticipated, some users continued to prefer the transparent version and they were likely to cite the increased insight that it facilitated into the algorithm's underlying operation. In contrast, others preferred the non-transparent version but offered very different reasons for their preferences. Some of these users seemed to find highlighting to be distracting, although our cognitive load results do not support this. Others may have preferred non-transparent feedback because it did not expose word-level errors, potentially leading users to overestimate the competence of the underlying algorithm with the consequence that they believed it to be more advanced than it was. For these users who preferred the non-transparent version, it seemed that

incremental feedback was providing more information than they required[24, 105].

6.4 Study 2

Ideally, a system should mitigate the negative distracting elements of incremental transparency while providing improved understanding to users. However, it is difficult from our initial study to know how to operationalize transparency in a way that achieves this. In order to better understand how to convey transparency to users in effective ways, we employed a semi-structured interviewing process in our second study. We used think-aloud interviewing methods to examine in depth how the type and timing of algorithmic transparency can inform decisions about how to design effective transparency. In particular, given that Study 1 indicated that some users felt incremental feedback provided too much information, Study 2 examines letting users view increased transparency only when they explicitly request it after they have finished writing. Overall Study 2 gathered richer contextual qualitative data to illuminate what factors influence the interpretation and uptake of transparency information. In particular, we wanted to better understand why ostensibly richer feedback was not providing anticipated benefits to some participants.

6.4.1 Method

6.4.1.1 Users

Twelve users were recruited from an internal participant pool at a large United States west-coast university. They received course credit for participation. Participants average age

was 19.54 years (s.d.=1.52) and 7/12 identified as female. This study was approved by an Institutional Review Board.

6.4.1.2 Measures

Users again completed a shortened version of the PGWBI to screen for mental health before the study [82]. Users answered a similar set of survey questions to Study 1 in a think-aloud style; however, these were primarily used to prompt explanation and structure the interview. As such, we do not present the results in this paper.

6.4.1.3 Procedure

The participants were randomly divided into one of two conditions. Both groups were given overall prediction feedback from the E-meter (Figure 5.1).

Passive Transparency: Six participants received real-time transparent word-level feedback about the algorithm's interpretation of their affect as they typed each word.

Requested Transparency: The other six only obtained transparent word-level feedback after they had finished the writing task; these users explicitly requested transparency by clicking a button labeled "How was this rating calculated?".

The researcher explained the experiment and think-aloud procedure, demonstrating a think-aloud on an email client. As in Study 1, the researcher asked participants to "Please write at least 100 words about an emotional experience that affected you in the last week." After the think-aloud writing exercise, the experimenter conducted a semi-structured interview that included an on-screen survey. After the survey, participants in the initial non-transparent

condition 1 saw exactly the same screen with an added button labeled “How was this rating calculated” which they pressed to reveal the transparent word-level highlighting. The entire process took around 50 minutes.

6.4.1.4 Analysis

Interviews were recorded using both audio and screen-recording. Two interviews (one from each condition) were not audio recorded, thus only the remaining 10 are used for the analysis. For this qualitative analysis, responses were coded using thematic analysis [22] specifically targeting RQ3: What problems must be mitigated to support effective transparency?

6.4.2 Results

More Transparency is Not Always Better: The transparent version of the E-meter again operationalized transparency using color highlighting to show how each word contributes to the calculation of the overall mood score. However, several users took issue with this level of detail. P10 felt that only the “big emotionally heavy words” should be highlighted. Other users felt similarly, P4 talked about select “trigger” words that that “trigger the foundation of the issue” and were essential to understanding the text. P8 felt that there were a few important words and the rest just added noise: “it’s taking into account words like ‘stressful’ and ‘regretful’ and stuff but then like everything else in between adds like an extra layer that complicates it”. While the machine learning model was limited to the 600 most predictive words of mood, that model still seemed to present too many extraneous words that users felt were unimportant. From a design perspective, it may be that users need to identify a small number of clear examples of

words showing strongly positive and negative affect, in order to form a working model of system operation.

Transparency May Violate User Expectations Even When the System is Correct:

Similar to how participants felt there were many extraneous highlighted words, participants also often focused on specific words that they judged had been misinterpreted by the system. Many users wrote about their experiences in the first person, often using the word “I”. In our machine learning model, “I” has a slightly negative connotation which confused our users, because many of them thought “I” should be neutral. P8 said, “So I was gonna say that yellow words would be neutral because it has highlighted ‘I...’”. Along the same lines, when analyzing the highlighting of different words, P6 said “‘I’? Mmm, I don’t understand that either”. Participant 5 even started conjecturing about the actual system model saying “‘I’ doesn’t seem like it would have... [participant trails off] unless people speak in objective terms when they’re talking about more positive experiences.” Clearly, these users don’t feel that the word “I” has negative connotations. However, extant literature confirms that system feedback is correct as high usage of first-person singular pronouns is correlated with depression and negative mood [157, 178]. These examples indicate a problem when system feedback reveals information that contradicts the users’ expectations. Even if the system is objectively correct when displaying transparency information, it can still cause users to take issue and result in poorer perceptions of that system.

User Heuristics May Contradict Transparency: Other problems arose because users formed working heuristics of how the system operated, which were sometimes contradicted by word-level feedback. Four users felt that there were discrepancies between the overall rating and the transparent word-level feedback within the system. When Participant 8

viewed the word-level transparency they started off by saying “I’m very confused” and then explained how they felt the overall rating should just be calculated as a ratio of positive/negative words—“Well I just assumed that some words would be coded as positive or negative and then it would just like do a ratio of those two.” Participant 6 explained it similarly, the overall rating showed a slightly negative emotion rating for their written passage but, in participant 6’s words: “So when you look at the comparison with the meter, and you look at my paragraph itself, right? There’s more words that are highlighted in green.” These users are using simple heuristics to relate the overall document rating and word-level transparency. They feel that the overall rating should reflect a simple ratio of word-level transparency. For example, if the highlighted words are primarily green, then the overall rating should be very positive. However, in our machine learning model, a single word such as “angry” could be rated negatively enough that it would cancel out multiple mildly positive words. Some users arrive at this correct model after consciously engaging with the system. For example, recall how participant 6 talked about how they felt the word-level highlighting and the overall rating were incongruent; however, after thinking more deeply, this participant later said: “... it’s weighing certain words, right? Because obviously these two words, right? Like “upset” and like—er yeah, “upset” really polarize the meter.” This quote demonstrates that this user has moved beyond their initial heuristic that all words are weighted equally; they are now noting how one word seemed to have a larger effect in the system.

Together these observations suggest that users initially form simple working hypotheses about system operation. Users seem to engage with transparency first by operating with these simple hypotheses and only scrutinizing these when their expectations are not met. As

with other areas of reasoning, it may be that in interacting with a system users first engage in rapid, approximate, System 1 thinking and only engage in deeper, more analytic, System 2 thinking when directly prompted or confronted with clearly anomalous information [99, 161].

6.4.3 Discussion

The second study again revealed the complexity involved in presenting transparency information. Confirming our first study, we again showed that providing more detailed transparency information is not always better. While some users saw potential benefits to word level transparency information, others argued against “complete” transparency, preferring to see only a subset of ‘important’ words. This observation suggests a principle for transparency presentation that limits the amount of information presented. This principle might involve explaining as much variation as possible using the smallest number of explanatory features. Therefore, we might aim to weigh the overall number of features we present against the information they provide. This is consistent with machine learning approaches to developing models with high dimensional feature sets that aim to identify features with the greatest explanatory power. While we know of technical methods that support this [137], we have not seen such transparency actually tested with users. A second observation is that users may take issue with detailed transparency and predictions, even when underlying system models are objectively correct. This creates quite a difficult problem for system designers. If it were possible to know which features prompt mistaken beliefs, then these features could be excluded from transparency. Unfortunately, short of testing user beliefs about all features, this may be very hard to do. We also saw that users don’t actively interrogate transparency information to deeply analyze all its

implications. Instead, users often look for quick heuristic routes to confirm or discredit simple working theories. Therefore, we should design intelligent systems in ways that allow users to develop simple working heuristics but also invite them to evolve more accurate mental models when they are motivated to do so.

6.5 Discussion and Conclusions

Unlike much technically oriented work that aims to develop new transparency algorithms, we explored user perceptions of, and reactions to, transparency. Specifically, we examined users engaging with a real system that interprets self-generated personally relevant emotional data. We deployed a necessarily simple algorithm that users find accurate [192] in order to create simple transparent feedback that conveys the underlying operation of the algorithm. Both exploratory studies indicate that developing and deploying transparent smart systems is complex in practice. In both cases, we observed unexpected user reactions to our attempts to provide detailed information about algorithmic operation. In Study 1 we found that before actual usage, participants initially anticipate greater accuracy for the transparent version of the E-meter but this is altered by their system experiences. After using both system versions, users are split 50-50 in their preference, and trust data showed similar ambivalence, even though both versions of the algorithm were perceived as accurate. We identified several possible reasons for this shift. As anticipated, participants who preferred the transparent version valued the increased system understanding that transparency afforded. But two different reasons led other users to prefer the non-transparent version: some attributed more sophisticated abilities to the

less transparent system while others felt distracted by transparency. Although many user comments mentioned how incremental word level affective feedback was highly distracting, these comments were not reflected by a reliable overall difference in cognitive load between transparency conditions as assessed by NASA TLX. However, more sensitive real-time measures of cognitive load might yield different results.

These initial results raise the question of how we can operationalize transparency in ways that don't distract, while simultaneously allowing users to engage with the system using simple heuristics and facilitating advanced understanding. Study 2 gathered richer contextual data to illuminate exactly how to operationalize transparency to fit the goals from Study 1. To address distraction and improve clarity, we discovered that some users prefer to see only major contributing features to the overall algorithm rating. More detailed transparency information can lead some users to falsely believe the system is operating incorrectly. Furthermore, users often evaluate transparency using simple heuristics rather than deep reflection. Together these results suggest that supporting transparency is complex and there are myriad decisions that affect the user experience when deciding how to operationalize it. We now discuss future design approaches that build on these observations.

6.5.1 Meeting the Competing Needs of Transparency Through Progressive Disclosure

Studies 1 and 2 revealed requirements that transparency must meet to be effective for users. This is a challenge because some of these requirements seem internally inconsistent. How can we allow users to develop preliminary working heuristics, while at the same

time facilitating detailed understanding for those users who value it? One design solution is suggested by an interaction that took place in Study 2. Recall that in Study 2, some users described their emotional experience using the non-transparent version and only later saw the word-level transparency after clicking a button labeled “How was this rating generated?” After clicking the button to reveal transparency, Participant 11 had further questions. Quite naturally, the participant pointed at the button again and had this exchange with the researcher (R):

P11: Can I click this? Does this...?

R: I don't think it shows any more than that.

P11: Damn it.

R: Yeah. If you were to click it, what would you expect to see more about?

P11: I want bullet points to tell me why it works the way it does.

Even after seeing the exhaustive transparent feature contributions to the overall rating, this user still had more questions about the system's inner workings. This data point suggests an interaction paradigm that meets the competing needs these two studies have generated: progressive disclosure.

Progressive disclosure has a long history in UI research, dating back to the Xerox Star and early word processing systems [29, 145, 188]. The original concept involved hiding advanced interface controls; allowing users to make fewer initial errors and learn the system more effectively [29]. In other words, advanced information and explanation are provided on an ‘as needed’ basis, only when the user requests it. Progressive disclosure is also consistent with the literature on explanation from the social sciences, which argues that in human-human interaction, explanations are ‘occasioned’, being provided only when the situation demands it

[70, 81, 91, 183]. Additionally, Grice's Maxims that govern the use of explanation include the "Maxim of Quantity"—that one should give as much information as is needed but no more. Progressive disclosure accomplishes exactly this by providing more explanation on demand progressive disclosure ensures that the user is receiving only as much information as they need.

We can apply the principles of progressive disclosure directly to transparency in intelligent systems. For example, similar to Study 2, the E-meter could show a "How was this rating calculated?" button. In this setting, the E-meter might start with only a document-level rating, which reduces distraction and avoids unnecessary complexity. Upon first press of the "How was this rating calculated?" button, the E-meter shows a brief natural language explanation e.g. "This rating was calculated using the positive and negative weighting of the words you have written." On the next press, the E-meter might show a subset of the high confidence and high impact words. This second press satisfies the users in Study 2 who only wanted to see the major factors influencing the algorithm. However, some users (like those in Study 1) may want yet more transparency. Another press of the button could reveal further word features that contributed to the overall score. More presses could reveal details about how the training data was collected or a textual summary of the machine learning model. In this progressive disclosure approach, following UX and human conversational principles, explanation is presented as a two-way communication with the user driving exactly when and how explanations are provided. Note too that because transparency is provided 'on demand' this removes confusions and inefficiencies arising from spurious, unwanted explanations, and adjusts explanations to the users' requirements. These design suggestions are consistent with recent work on folk theories in algorithmic systems [49, 60] as well as a large body of social science theory; these theo-

ries show that people are content to operate with simple (often inaccurate) situational heuristics unless they are deeply invested in a decision or the situation is strikingly anomalous [99, 161].

Progressive disclosure of transparency is not limited to predictions from text. For example, other researchers have examined transparency in the context of deep learning models predicting patient outcomes in the medical realm [110]. In this context, patients have a predicted diagnosis risk of a disease and a high number of features (e.g. previous medical diagnoses, number of doctor visits, type of care). Each feature can be visualized and ranked for its contribution to the overall predicted diagnosis risk score. In addition, users can compare outcomes between patients and conduct what-if analyses by modifying patient attributes and seeing how predicted risk changes. Our results suggest that exposing such complexity at once could overwhelm certain users, leading them to reject the tool. To operationalize progressive disclosure in this setting, we might present the predicted risk score by default with a short natural language explanation following natural language explanations in [89]. Should the user request more information, we can incorporate visualizations of features that most contribute to the predicted risk score (e.g. [173] and our transparent version of the E-meter). A request for further information could show similar patients and previous outcomes that have informed the current prediction. Finally, if the user continues requesting increasing amounts of disclosure about a prediction, we can infer that user is invested enough to truly engage with the system and we can present an exploratory explanation (e.g. the explorable predictions in [92]). Such a tool might allow users to modify a patient's features and see how this changes the risk prediction, helping the user to consciously build an accurate mental model of the machine learning predictions. Again, the major contribution of progressive disclosure is avoiding overwhelming users

with information, but instead slowly increasing transparency as users indicate a willingness to engage meaningfully with it. We believe this can improve the acceptance of intelligent systems in many realms.

6.5.2 Impact for Future Transparency Research

Another issue arising from our research is the role of individual differences. It was apparent that different users have varying reactions to, and expectations about intelligent systems. One possibility is that these differences arise from individual user traits, such as need for control [2] or need for cognition [26]. Future work might examine the relationship between such individual traits and reactions to transparency, allowing designers to profile users and deploying personalized system versions.

Our results also have important methodological implications. One method used in prior studies is to provide potential system users with hypothetical scenarios describing system operation and eliciting reactions to those systems. These methods offer ways to collect controlled user data at scale [69, 123]. However, results from Study 1 indicate the importance of direct user experience when making system evaluations; users' perceptions of the system were very different following actual usage compared with their projected reactions prior to usage. Care needs to be taken with the usage of scenario-based methods.

6.5.3 Limitations

The current study examines the important algorithmic domain of emotion analytics, but clearly other contexts need to be explored. Furthermore, our deployment of a working algo-

rithm meant that results were obtained for situations where our algorithm generated moderate numbers of errors—future research should compare contexts where there are different levels of errors [123]. Additionally, while users generated their own data in our system, results were not directly used to inform other aspects of the user’s personal behavior so the costs of system errors were low. While this is appropriate for exploring the understanding of initial algorithms with moderate error rates, future work might explore user reactions to transparency in higher risk contexts. While we believe our implications regarding progressive disclosure are generalizable, our work derived these insights from one operationalization of transparency, namely a dynamic visualization of an algorithm. There are many other ways to depict how an algorithm operates including verbal explanations, concrete user exploration, and so forth [105, 109, 124, 173].

6.5.4 Conclusion

Overall, our data suggest empirically motivated challenges in designing effective UX methods to support transparency for complex algorithms. Our results also reveal potential new research questions regarding user traits, system heuristics, and workload. The current study indicates a promising design approach involving progressive disclosure which we intend to explore in future work. It is critical to answer these questions as we continue to deploy intelligent systems with increasing ubiquity and impact.

Chapter 7

AI and Explanations in the Wild

7.1 Introduction

I have extensively examined transparency and explanation in the previous two chapters in a lab setting involving a relatively short low-stakes interaction between user and system. This parallels much of the current research on machine learning systems and explanations. However, findings may be very different in high stakes contexts where users are deploying intelligent systems with real-life implications. I now examine a real-world intelligent system to examine how the E-meter system findings generalize to user interactions with deployed intelligent systems.

I address this shortcoming in the current literature by examining an educational predictive analytics system. This system makes predictions about the performance of university students using machine learning and explains these predictions using its own interface. The system conforms to current recommendations that machine learning should become an ‘unre-

markable' component of a larger system [223]. The system displays a students' predictive rating as a small badge on the student profile page which advisors regularly access to write notes and look up other information. Advisors use this system to better support students and reach out to those who are having trouble academically. We consider this system high-stakes because of the potential impacts that it can have on individual students; getting support like tutoring can increase the retention of at-risk college students [172, 115]. However, if advisors cannot identify which students to proactively contact, students may fall through the cracks and not receive the support that they need.

Our primary research questions are broad, reflecting the novelty of examining a high-stakes intelligent system in context. We ask:

- How do expert users initially evaluate an intelligent system?
- How do users evaluate intelligent predictions in real-world contexts?
- How do users engage with intelligent predictions and explanations?

7.2 Methods

We first conducted pilot interviews with 8 advisors to better understand the advising context and their workflows with the technical systems they were using. Additionally, the pilot study focused on advisor knowledge and opinions about predictive analytics. The pilot study informed the questions we asked participants, helping us to understand the advising workflow, how the system fits into this workflow, and gave us language and context that enabled us to ask what we wanted. Using the results of this pilot study, we developed a protocol to examine how

advisors engage with the predictive analytics systems initially and to elicit opinions about how the system should be used at the institution.

The pilot interviews indicated that when advisors first used the predictive analytics system they wanted to engage in ‘vetting’, that is they wanted to examine a number of students to understand how the system rated them and explained those ratings. Therefore, we incorporated this as a task within our protocol: we asked users how they would go about determining how accurate or inaccurate the system was and let them examine a number of students within the predictive analytics system. We allowed advisors to choose the students to examine because we wanted to obtain advisor’s reactions to the algorithm in a real-world contexts. Depending on interview time remaining, after examining 4 or 5 students, the interviewer moved on to the semi-structured interviewing process detailed later.

7.2.1 Predictive Analytics System

As noted by the vendor, the system we examined is widely used across hundreds of higher educational institutions, with thousands of staff users. However, each institution has its own implementation of the system which contains their students’ data and allows for minor modifications of system features. The system contains multiple functions besides predictive analytics that include: student-advisor appointment scheduling, note-taking and sharing across advisors, information like GPA and coursework for every student, and a variety of system views on measures of student success. All of this functionality is identical across the campuses that it is deployed at, however the predictive model is trained on specific data for each institution to better fit each unique context.

At the point when this research was conducted, the overall system had been implemented at the examined institution for 3 years. For these three years, advisors had been using the system's non-predictive features like notes sharing, academic progress visualization, and student search abilities. The system was mandated for daily use for collegiate advisors who specialize in first and second-year students; all of these collegiate advisors had been trained in system usage. The other group of advisors, major advisors, who work with students in their major coursework, had the option to use the system if they wanted. Software developers at the research institution had modified it to operate in the context of the research campus, a process which involved creating different roles and privileges and modifying the framing around the predictive analytics feature.

Of particular interest to this research is the predictive analytics feature. The overall system contains the ability to embed and individualized predictive analytics model for each campus that it is implemented at. Each campus has different lower division classes, majors and credit requirement. Implementing the model involves the campus specifying a number of choices about the model (e.g. whether to build individual models for transfers/non-transfers and what thresholds indicate a student is high/low risk) and then providing historical student data that is used to train the predictive model. At the time of the study, the predictive analytics feature was implemented but had not been deployed to the advisors yet. As such, our interviews with these advisors were their first exposure to the predictive analytics in the larger system, although advisors had been using other system features for several years.

The predictive analytics are described by the vendor as predicting 'risk'. Universities are focused on student retention, and therefore want to identify those students who need

advice and assistance if they are to avoid dropping out. At the researchers' institution, a 'risk' prediction is the chance of not returning to the university in the following Fall quarter. These predictions are created using a series of "penalized logistic regression models" that have been trained on over 6 years of previous student performance data at the given university. The predictive analytics are a "series of models" because there are different models for students depending on how far in their academic career the student is and whether they were transfer/non-transfer students. For example, students in the 1-45 credit range are typically first years and the predictions are made by a model specifically trained to predict for students in this 1-45 credit range. The models used 60 features for training. These included multiple features regarding academic performance (GPA, Grade Variance, High School GPA, standardized test scores, etc.), institutional factors (credits attempted per term, average success in major, transfer status), demographic factors (international student indicator, veteran status, age at first term), and some proprietary measures that the vendor calculates (major-skill alignment, estimated skills) which we will examine further later in this section.

An important distinction in the model to make is the difference between natural features that are simply given to the vendor by the university and "engineered" features that the vendor creates from this data. As defined in [71], "Feature engineering involves the selection of a subset of informative features and/or the combination of distinct features into new features in order to obtain a representation that enables classification." Some of the features included in the model are provided directly by the university and are pre-existing, e.g. GPA, standardized test scores, and international student indicator. Others are "engineered" using domain knowledge to synthesize existing features in a way that improves representation to the learning

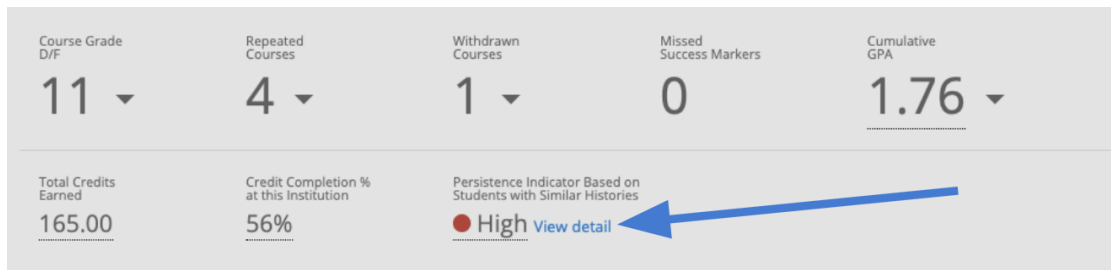


Figure 7.1: Dashboard Showing Student Risk Prediction and View Detail Link to Explanation

algorithm. For example, the model includes “Average Success in Major” which is a measure of aggregate retention within the major as a whole. Other engineered features include the aforementioned “Major-Skill Alignment” which is “A proprietary measure of how well a student’s current major(s) are aligned with their previously demonstrated academic ‘skills’.” Academic ‘skills’ themselves are used as a feature, ‘skills’ defined as: “... underlying patterns in the grades students earn in different courses – e.g., some students may have a history of excelling in math-related courses but not writing-related courses – and call the discrete factors behind these patterns ‘skills’.” The difference between natural and engineered features is important from an explanation perspective, because engineered features are by definition less intuitive as they do not map directly to natural categories.

The output of these logistic regression models is a risk score categorized into 3 levels: Low, Medium, and High risk. These scores indicate the following: the low-risk population contains students where the model predicts between a 66% and 100% chance of returning the next fall, the moderate population contains students with return rate predictions between 33% and 66%, the high-risk population contains students with predictions between 0% and 33% return rates. The risk score appears on the default student profile which advisors use very often,

this is shown in Fig 7.1. Fig 7.1 shows this default student profile page which shows aggregate statistics about the students and some system features. These aggregate statistics include information that may be pertinent for advisors to quickly see. The student profile includes: the students GPA, the number of D/F grades the student has received, the number of withdrawn and repeated courses, and information about the students accrued credits at the institution. Additionally, the system features include the risk score which is labeled ‘Persistence Indicator Based on Students with Similar Histories’ and ‘Missed Success Markers’ which represents whether students have failed to pass required courses to enter or complete their major. Additionally, users can access an explanation of the risk score by clicking the “View Detail” link next to the score itself.

Regarding the accuracy of the predictive models overall, the vendor provided a report concerning the accuracy of the “risk model”, in summary, the report wrote “Your [model] is high-performing; it can be used confidently to both assess individual students and efficiently design effective, targeted intervention campaigns” [words in brackets modified by the authors for anonymity]. Recall that the system uses a series of models for different credit ranges and specific student attributes. Overall, the average AUC for identifying high-risk students in these models used in the system was .82. Regarding AUC, .5 represents pure chance and 1.0 represents a system that would be exactly correct every time. In other words the system was moderately accurate in its predictions.

One point of interest is the relabeling of these “Risk Scores”, as ‘Persistence Indicator Based on Students with Similar Histories’. After initial informational meetings with advisors about the predictive analytics, the implementing body at the university felt that the language

	● Negative	● Neutral	● Positive
Program of Study	-	-	<ul style="list-style-type: none"> ● Average Outcome In Major 0.53 ● Percentile Rank in Major 0.01
Performance	<ul style="list-style-type: none"> ● Total Number of D/F Grades Earned 11 	<ul style="list-style-type: none"> ● Cumulative GPA 1.76 	<ul style="list-style-type: none"> ● GPA Trend -0.44
Progress	-	<ul style="list-style-type: none"> ● Average Outcome in Credit Range 136-180 credits 	<ul style="list-style-type: none"> ● Average Credits per Term 20.6 ● Earned to Attempted Credit Ratio 0.56 ● Lifetime Accumulated Credits 165
Pre-Enrollment Data	<ul style="list-style-type: none"> ● Proportion Transfer Credits 0.65 ● Transfer Student Y 	-	-

Persistence Indicator Based on Students with Similar Histories ● High

Figure 7.2: Explanation View: The Predictive Analytics Explains Ratings by Classifying and Showing Predictors

around “risk scores” was judgmental and could lead to a less equity-oriented advising relationship with the students. The university therefore decided that these scores should be called “Support Scores”, indicating the amount of support a student may need to succeed at the institution. Later these were changed to “Persistence Indicator Based on Students with Similar Histories.” A major factor in these changes was strongly voiced concerns from a subset of advisors who believed that implementing predictive analytics for student populations was unethical.

The system shows both an overall rating of Low/Medium/High risk and also an explanation of the overall rating. This explanation can be accessed through the “View Detail” link indicated by the blue arrow and also through opening an adjacent tab in the interface. This explanation capability was explained to advisors through the document provided at the start of the interview; the document included language from the vendor where the vendor defines this explanation capability as “They represent a subset of predictors – one of 60 variables that our

models find correlation between that variable and student success. The point of these [explanations] is to distill down an array of variables to the few things that are truly having an impact on students.” A picture of this explanation capability is shown in Fig 7.2. The explanation shows predictors which are broad representations of the features that the logistic regression model uses to make its overall prediction about student risk. The explanation divides these predictors in the model into 3 main categories across the top horizontal labels. Predictors are either Negative, Neutral, or Positive, depending on how they influence the student’s rating towards Low or High support. Each of these predictors in the figure can move between the Negative, Neutral, and Positive categories depending on their value. For example, a student with a GPA of 3.95 may have their “Cumulative GPA” predictor in the Positive column while a student with a GPA of 1.2 may have their “Cumulative GPA” predictor in the negative column. Predictors are also classified as being about the “Program of Study”, “Performance”, “Progress”, or “Pre-Enrollment Data”. This second group of classifications intends to provide structure so that advisors could quickly see where the student may be having issues. Alongside each predictor is the numeric value of that predictor for example in Fig 7.2, “Total Number of D/F Grades Earned” has an “11” beside it indicating that this student has earned 11 D/F grades in their college career. As we will talk about later, it may be important to note that this form of explanation can expose information that violates expectations; in Fig 7.2, having a downward “GPA Trend” of -0.44 is categorized as a positive influencer which may not be what users expect.

7.2.2 Participants

We interviewed 17 advising faculty at a large west coast university in the United States. As previously mentioned, the advisers had been using the broader student success system for up to 3 years. The predictive analytics component was in beta deployment as the researchers conducted this study. The primary responsibility for advising faculty at this university is to ensure that students are progressing towards their chosen degree. The advisors help the student choose classes for their major, they reach out to students experiencing academic difficulty, and they point students towards resources that will help improve the students' academic experience. Through our pilot interviews, it was clear that advisors view the stereotypical advising jobs like course planning and helping students choose majors as a small subset of their relationship with the students. The majority of advisors strive to advise holistically and this often involves becoming a confidant and counselor to students who may be experiencing familial and mental health issues. These issues necessarily affect the students' academic performance and advisors are quick to point towards information like this as enabling them to contextualize the students' experiences and academic performance in order best help their students.

The experience range of these advisors was vast, from less than a year to having advised at the same institution for over 40 years. We were only able to collect data regarding the advisors education level for half the advisors we talked to. Among those, 44 percent had a Masters degree in higher education, 44 percent had degrees mostly in the humanities and social sciences, and around 12 percent had a bachelors in physical sciences. We include this information to illustrate that the majority of the advisors were non-technical and did not have

extensive experience with statistics and predictive analytics as we will further examine later.

These advisors contributed to different aspects of the student experience. At the university, there are 3 primary types of advisors. College Advisors are the first point of contact for students when they enter the university, these college advisors help students strategize about their general education requirements and also for classes they must have to later propose and enter their desired major. College advisors are also often those who reach out to students in academic difficulty and help them find the resources they need. Major advisors specialize in helping students plan and take the required courses to complete the major they desire. Major advisors have extensive knowledge about the specific classes within their major and can advise students on common paths through majors, course-load, and also considerations that the college advisors may not know about the specific major. Finally, EOP (Education Opportunity Program) advisors have a mission to “provide support of first-generation, low-income and educationally disadvantaged students.” These advisors work with underrepresented populations in traditional college environments and advise them in ways that help students to navigate academic life. Of the 17 total advisors, 10 were College Advisors, and 7 were Major Advisors.

7.2.3 Materials

Advisors were provided with a 2-page document explaining the predictive analytics system. This document was adapted from training materials by the vendor of the predictive analytics software. It contained information about what predictive analytics for student success are, what features the system may use to predict student success, what different support scores mean, and suggested action items for each support score. The document described what the

“risk score” was and explained what each individual rating meant. Along with the individual ratings, the document described what types of behaviors students in these ratings would exhibit and what type of support could be provided by the advisor to them. The document also included information about what the predictors in the explanations were, how they were categorized, and how they should be interpreted. The document did not supply any information about the performance of the predictive analytics, only instructions about how to interpret the system information.

7.2.4 Interview Questions

The interviews followed a semi-structured format, and were based on prior pilots which had explored typical advisor tasks, their interactions and main goals when reviewing student files and advising students. Following the semi-structured interview procedure, the interviewers had precompiled questions but were free to ask advisors follow-up questions or to encourage advisors to further elaborate on specific points. We began by informing the advisors about the study, how the results would be used, and asked advisors for their consent to participate in the research and be recorded. The interview questions were as follows:

- Have you used predictive analytics systems before?
- How accurate or inaccurate do you think this system will be? (with follow-up for explanations)
- How would you determine how accurate or inaccurate the system is? (with follow-up for explanations)

- Are you comfortable proceeding and looking at examples of student scores? (with follow-up for explanations)

For each specific student the advisor chose to look up we asked:

- Why are you choosing this student?
- What prediction do you expect for this student?
- Does the prediction match your expectation?
- What do you think of the predicted influencers? [the explanation feature]

We then moved to general questions about the system experience as a whole:

- In general, what did you think of the support level predictions?
- In general, what did you think of the predicted influencers?
- What are positive or negative effects you envision if predictive analytics were implemented like this for all advisors at [university]?
- Do you think this system should be used at [university]?
- Do you think advisors should be trained the use of predictive analytics? If so, what should the training focus on?
- Is there anything I should have asked you about that I didn't?

The answers to these questions to analyzed using thematic analysis following [22]. Our specific research questions focused on how users understood and interacted with predictions in the system. As such, the majority of our analysis focuses on these parts of the interview.

7.3 Results

All but two of the advisors had no prior experience with predictive analytics systems. Of the two interviewed advisors who had used predictive analytics before: one used predictive analytics in a previous advising position at another university and another advisor used predictive analytics in their job search to answer questions about advancement and salary.

In aggregate, advisors seemed to find the predictions from the system accurate. Fig 7.3 shows the distribution of expectation violation for the students that advisors examined. Recall that the ratings are ordinal, students are either labeled “Low”, “Moderate”, or “High”. In Fig 7.3, students labeled ‘Correct’, means that the rating was exactly what the advisor expected. ‘Off By One Class’ indicates that the rating was off by one class in the ordinal measure; ‘Off By Two Classes’ indicates that the rating was off by two classes, for example rating a student as “High” when the advisor expected “Low”. While we present this information in aggregate, it may be important to acknowledge the diversity of advisor experience that this represents. For example, advisor 8 examined 4 students and found that all of their ratings were exactly as expected. On the other hand, advisor 9 examined 5 students; they found only one that was correct, 3 that were off by 1, and one more that was off by two. As we have examined in Chapter 5, this expectation violation is highly correlated with the advisors’ perceptions of accuracy in the system. In our cases, we saw that advisors had varied perceptions of accuracy based on the students that they chose to examine.

After using the system, there was no clear consensus between advisors whether or not they wanted to see the predictive analytics implemented in their workplace. Forty-two percent

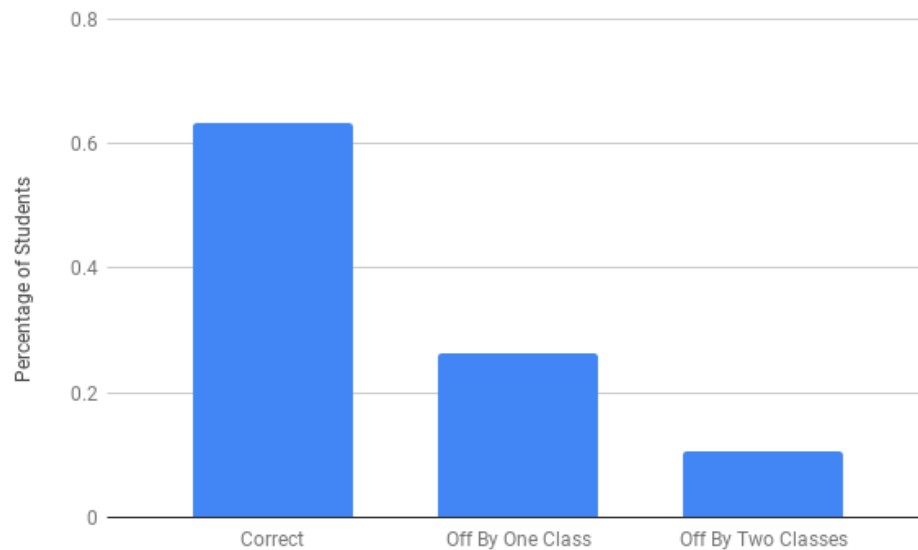


Figure 7.3: Advisors Found Most Predictions Agreed With Their Own Assessments

of the advisors we interviewed wanted to see the predictive analytics used; they cited the ability to reach out to students in a fine-grained fashion and the ability to more quickly find students who need help as major reasons for implementation. One-quarter of advisors felt that the system should not be implemented, they cited data privacy concerns, concerns about the system being used to push students out of preferred majors, and performance concerns about the system. Another quarter of advisors did not directly answer yes or no, these advisors saw positives and negatives with the system and did not have a strong opinion either way.

7.3.1 Advisors Initially Evaluate System By Carefully Examining Students with Different Characteristics

Advisors were moderate in their expectations of system accuracy when probed before their use of the system, in general, most felt that it would be pretty accurate. Though, when asked what they would like to do first with the predictive analytics systems, advisors were quick to respond that their first step was to verify that it was accurate in identifying high-risk students. They did this by exploring predictions for a range of students of differing abilities. advisor 8 referred to this as ‘Trust, but verify’. Rather than simply choosing students that were convenient and at hand, advisors deliberately explored multiple system predictions for a variety of different types of students. Advisor 8 said, “I’m trying to get a good spread of different demographics from freshman, sophomore, junior, seniors, and international students, transfer students et cetera”. After examining a couple of students experiencing academic difficulty, Advisor 11 sought out different students, saying that they wanted to see a range of students from “High achieving student who gets A’s all the time” to “Medium Achieving Students—the ‘C’s get degrees’ student” all the way to the “Low achieving student who consistently gets academic probations and is subject to disqualification”.

In addition to sampling a diverse group of students in their initial interactions, advisors did so while simultaneously selecting students that they knew well so that they could more accurately judge the prediction from the system, advisor 9 encapsulated this saying: “Just trying to pick people I’ve seen recently because I’ve got a really clear... I’m looking for something that’s not low. That’s, that’s what I think I should do because we’ve got lots of low, we’ve

got all low readings so far.” advisors manipulating their testing to explore different student attributes and chose examples they knew the most about. Overall they deliberately manipulated their initial system experiences to accurately evaluate how the system was performing across the spectrum of students.

7.3.2 Transparency Can Polarize System Perceptions

Part of our protocol involved asking the advisors not only whether the overall rating for a student matched their expectations, but also whether the explanation of that rating made sense to the advisor. In situations where the advisors expectations were violated and the overall rating did not match what the advisor expected, advisors were eager to examine the explanation for this rating. In cases where it was what the advisor expected, the interviewer often had to prompt the advisor to examine the explanation. These behavior from advisors are expected given the analysis in Chapter 5 that indicates that expectation violation is the primary time when advisors would need explanations from the system.

Similar to recent research on explainability and transparency [192, 194], we again find that transparency has both positive and negative effects on the user experience. Context and timing seem to determine whether transparency is helpful or detracts from the user experience with predictive analytics. Explanations Provide Reassurance about the Correctness of a System Prediction: Some advisors initially disagreed with the risk score for a student but the explanation provided reassurance and allowed the advisors to better understand the reasons for the prediction of the system. For example, one advisor expected a student to be a high-risk level after assuming that the student was domestic, however the system predicted low risk and the system explanation

showed the student's international status as a key variable in its prediction leading the advisor to reflect, saying: "Yeah, in this case, I didn't pay attention that this student is transfer and international student. I judged this student status based on frosh four-year student. I said, 'no way he needs strong support'. But he's international, so it makes sense." In another case, the advisor disagreed with the overall risk prediction but upon seeing the explanation began to change their mind and said "So here's, here's part of what's going on. The first term GPA is 3.3. And so while the student has two non-passing grades, it looks like the GPA is higher than if the student had, you know, a couple of F's." In these cases, explanations from the system were successful; they resolved the disagreement with the advisor and reassured them the system was working properly.

7.3.3 Explanations Undermine a System Prediction

However, in other instances, explanations led to further questioning of the system even when the advisor agreed with the overall risk score. One advisor agreed with the overall moderate risk rating, but when looking at the explanation began to question the data the system was using, saying "Moderate which I would agree with at this point. The reasons make sense to me. ... first term GPA—that's not accurate because their first term was it a 3.3? I don't know. Maybe it was a 1.8. So. That is off. So it seems wrong ... It's miscalculated or their first term including all of their transfer credits may be a 3.3 but their actual first term GPA was a 1.8." Advisor 9 agreed with the overall risk rating but then noted discrepancies between this student and another they examined: "It's interesting. She's got the same number as the previous one, but hers is marked as negative." It's not always possible to easily tell whether the system

is right or the advisor is right, however, there are significant consequences. These explanations are provoking expectation violation and previous research indicates how this is related to a lack of trust and reduced perceptions of accuracy in the system [192, 194].

7.3.4 Progressive Disclosure Principles: Explaining Features

Confirming prior work in testing explanatory systems [194], we also see users' desire for more detailed information about the explanations themselves. Advisors often had further questions after viewing the predictive analytics. These were often concerned with the nature of the variables in the model. Some advisors asked for the ability to have mouse-over descriptions appearing for variables. These would provide further information about what the specific feature was: Advisor 1 said, "If they could explain it in like a little asterisk thing at the box—like this is how we factor that..." Others asked to know in more detail why a specific feature was rated as positive or negative and what the thresholds to switch between ratings was. Advisor 12 expressed this about a specific feature, the count of D's/F's a student has, as needing to know "what is considered positive versus negative or neutral. The negative in terms of like how many D's and F's does it take to be negative." Interestingly, advisors wanted these more detailed justifications for some and not others. We further examine these requests for feature explanations in the next section.

7.3.5 Advisors Have Difficulty Understanding Engineered Features in Explanations

A common practice in the data science and machine learning workflow is to “engineer features.” As we reviewed in the methods, engineering features often means subdividing or combining provided data into new features that didn’t previously exist. An example of this in our system is the “Average Outcome in Credit Range”. Average Outcome in Credit Range is an engineered feature which essentially measures retention at different points during the college experience. For example, the 1-45 credit range (students in this range are typically in their first year) is considered to have a low outcome because many students drop out in this 1-45 credit range. On the other hand, the credit range 135-180 (typically fourth-year students) have high average outcomes because most students who get to this point go on to graduate. This is an engineered feature because it combines aggregate outcomes for many students to then form a feature that is helpful in predicting outcomes for an individual student.

As might be expected, advisors had much more difficulty understanding engineered features than other more intuitive features such as GPA and Transfer/International Student indicators. Advisors often asked questions trying to determine what they meant. advisor 12 asked ‘What does rank in major mean?’ and advisor 10 asked: “I have a question about what this is, average outcome in major?” Other advisors understood what information the feature was trying to capture but had trouble understanding the specifics. advisor 1 noted, “Percentile rank in the major. This makes me wonder: is it major courses or is it the major as an overall including GEs, so you’re looking at the percentile rank in major and you’re wondering is that including just

lower division or is it just courses required for the major or is it courses required for the major and general education?” Overall, most of the difficulties that advisors had in understanding features in the system’s explanation came from questions about what specific engineered features meant.

7.3.6 Discrepancies Between Advisor and System Models of Student Performance

A key problem that advisors ran into repeatedly in the explanations was a mismatch between their own versus the system’s models of student performance. Recall that the system is using many cross-sectional measurements such as GPA, SAT/ACT scores, and number of credits; these measurements often represented single points of time where the student had their performance evaluated. Advisors had trouble reconciling the system’s point-in-time explanations with how they thought of student performance; advisors saw student performance as unfolding over time. Advisors viewed student performance as a narrative, in which there were specific milestones and challenging events (e.g. the student’s first quarter, preparing to declare a major). Advisor 2 talked about this discrepancy between the system and her own model: “My concern was. How he did in his beginning quarter because that’s the most challenging quarter for, for students to address. That’s when they blow it, and they pick up at the second quarter, but he did well, he did little less than well and then into third quarter he dipped so something happened there. It could be emotional, family, finances and also it can contribute to him being undeclared and that, not having a goal to look for.” After advisor w explained this, they continued to say “I mean it says the numbers, the GPA, The, only numbers, and it doesn’t say anything about the spring quarter. Of course, the system will not you know, think like we think...” Advisor 4 talked

similarly about how they think of student success as being milestone-based, whereas the system did not present this view: “Yeah, you know, I guess what I would say is that you know, we looked at things like unit completion. We looked at things like GPA. . . . But I look at milestones, you know”

7.3.7 Institutional Constraints the System Does Not Know

Advisors struggled to understand how the system could have such apparently complete data about students but fail to include in institutional factors that determine student success. At the university this research was conducted at, there are classes required to propose a major, general education classes, major classes. All these different classes must be completed on a specific schedule. The predictive analytics platform was built to generalize across campuses, and so does not incorporate all these subtle factors concerning differences between class types and schedules. Advisors found this troublesome. Advisor 5 looked at a student who had a moderate risk rating and said “This is a student who falls into the qualification issue. . . . This student is going into his fifth year with no declared major but he’s only at a moderate level. Right? So, for me, that’s a huge red flag.” Advisor 3 talked explicitly about this also “But it doesn’t look to me as if there’s any factoring in of like progress markers. . . . If it’s 50 upper division units [to graduate] and she only has 15 by x amount of time then you’re running out of time.” Advisors felt that without the system factoring in these institutional limits, it was missing clearly identifiable cases where students needed help. It is difficult to say that these missing institutional limits are an inherent factor of the complexity of putting hard constraints on statistical models or because the model itself is derived from a platform meant to generalize across

many campuses.

7.3.8 Conflicting Explanation Goals

We found that advisors have two primary goals for explanations; advisors expected explanations to provide auditability for the system and to provide guidance for which actions advisors should take to better support their students. We saw advisors in this study asking for these auditable explanations; Advisor 8 in our study said “I want to know why it works and how it exactly functions and what its purpose is. And so if I don’t know where the data it is deriving its analytics is coming from, to me that doesn’t really make a ton of sense.”

Finally, a user goal that we have not seen written about is that our advisors wanted explanations to be actionable. The advisors in our study saw this system as an aid to help them support students and therefore wanted the explanations to contain information that they could use to inform the type of support they could give to the student whether it is referrals to group tutoring, talking to their faculty professors, or to basic needs support programs. After viewing an explanation, Advisor 12 said it succinctly: “This is interesting information to know, but it doesn’t lead me to do anything about support.” Often system designers, whether they know it or not, intend their explanations to be persuasive and convince users the system is working; this may be at odds with user goals that want auditable and actionable explanations.

7.3.9 Ethical Concerns About Using Predictive Analytics

Most advisors felt that predictive analytics could be helpful in executing their job but some expressed ethical concerns about how the use of predictive analytics would affect stu-

dents. One major concern was student data privacy and consent. One advisor, who declined to be recorded, also declined to explore the system at all because she wanted further explicit consent from students that their data be used for predictive analytics, rather than the blanket consent that students agree to when entering an institution. Another major concern of advisors was that the system creates self-fulfilling prophecies. Advisors feared that by labeling students as high risk, they would doom these students to be treated differently and disadvantaged academically. One experienced advisor referenced the Pygmalion effect directly, citing previous research that indicated that labeling arbitrary students as low or high performing created self-fulfilling prophecies; the advisor said, “I don’t want to be feeling the Pygmalion effect when you go into the classroom and you have this assumption about students...”

7.4 Discussion

Our results suggest that there are many unsolved problems in how people interact with and evaluate intelligent systems. We saw how explanations can polarize perceptions of the system, how mismatching mental models can lead to system doubt, and how users may have ethical concerns with the task the system is performing overall. Our interviews with these advisors surface a number of different design implications for intelligent systems that people work with every day. We continue on to examine these design implications that address the problems we found in our analyses.

7.4.1 Error and Expectation Violation

Contrary to our previous studies in Chapter 5 and 6, we did not explore what error means within the predictive analytics system. This was a purposeful decision and again highlights the importance of system context in studies with intelligent systems. In the predictive analytics system we studied, it makes predictions about the future performance of students. As such, it is difficult if not downright impossible to make judgments about whether the system prediction is an error. As such we can only talk about whether advisors' expectations were violated by the system. Advisors had complex mental models about how the system would predict, what information it could take into account, and their own opinions on predictions. These responses from advisors open up the idea of expectation violation from the previous definition of simple disagreement with system predictions [192, 105]. Advisors had expectations for how the system would behave that accounted for unknowable information from the system's perspective. In addition, they had their own predictions for the student which may be correct or not. And of course, the system had its predictions also. Advisors often compared and navigated between these 3 different expectations in order to form their perceptions of system accuracy and ability; it seemed that this was necessary because the advisors could not evaluate the true ground truth error levels of the system through their own experience.

7.4.2 Systems Should Deliberately Walkthrough Users Initially

Nearly all the advisors that we interviewed were very systematic in how they evaluated the system for accuracy and trustworthiness. When testing the system, advisors chose students that they knew well and sampled these students across dimensions such as academic

performance and tenure at the institution. The system, however, did little to facilitate this initial testing and expectation setting. It is particularly important to set expectations and create shared mental models quickly when using a new intelligent system because mental models are hard to modify, being difficult to change once they have been developed [205].

Intelligent systems should include initial system walkthroughs that enable advisors to engage in this expectation setting in facilitated ways. For example, when first entering an educational predictive analytics system like this, the system should show advisors examples of students at least from each of the predicted risk categories and maximize the differences in attributes for each of students shown. This will allow advisors to calibrate expectations for reliability and help them to construct more reliable mental models that will be useful for the rest of their system usage.

7.4.3 Explanation Should Be Interactive

Similar to previous research [192, 194, 123, 105], we again see that explanations and transparency can have both positive and negative effects. In some instances, they reassure users that the system is operating correctly, while in other instances, they can provoke doubt that the system is working as intended. These problems both follow from the simplistic ways that we currently structure explanation within intelligent systems. When users ask for an explanation, the system displays as much information as about how the rating was calculated as it can. Rather than explaining the predictions based upon why the user asked for an explanation, these systems simply dump all the information they know and expect the user to figure it out themselves. Additionally, another theme in our interview was that advisors often wanted to know

yet more about the model even after the explanation had been supplied. For example, we saw advisors asking why certain variables in the explanation were considered positive or negative and what their thresholds were. Finally, another related theme was that advisors had trouble understanding explanations that contained specific features. In particular, advisors had problems in understanding engineered features. These themes all indicate major problems with how we explain intelligent systems currently.

We suggest that these problems arise because users expect richer forms of explanation than at present. For example, rather than being able to ask “Why is this student rated as lower risk than I expect”, advisors can only ask “Why this rating” or “View Detail” to determine what led to this rating. Understanding how specific user expectations are violated and why an explanation is needed would allow the system to tailor the explanation in ways that avoid some of the problems we discovered. If systems provided a way for users to ask for explanation that hinted at why their expectations are violated then the system could hone in on specific features of the explanation of interest to the user, rather than showing everything and expecting the user to engage meaningfully with all of it. As we have seen, this approach can lead to further expectation violation even when the system is predicting correctly.

Similarly, viewing explanation as more interactive allows us to implement the progressive disclosure principles and explain the engineered features that advisors had trouble understanding. Rather than the explanation ending after a single interaction, users should be able to continually drill down into the explanation to better understand specific parts on an ‘as-needed’ basis. For example, advisors currently wanted to know about thresholds for certain features but had no ability to ask for this information in the current system. We must re-examine

how we explain in order to better serve users within intelligent systems.

7.4.4 Resolving Conflicts in Explanation Goals

One problem present in both the literature and our current study here is that explanations may serve multiple goals, and it is important to know which are being addressed. Some systems explain in an effort to convince users that the intelligent system is correct and the user should trust it [73]. Other systems explain in order to make themselves auditable so that users can verify how the system is working. In particular, users in our study wanted auditability for two reasons: to ensure the system was not using information that could enforce societal biases (e.g. race) and also to vet that the system is using information that makes sense for this task in general. For our population of advisors, enough had heard of algorithms perpetuating biases similar to what we explored in Chapter 4; this made them leery of using a predictive system if they couldn't see how it was working. This distinction between auditable and persuasive explanations has been written about previously, where [193] noted that explanations are used both to persuade users/improve user experience and for auditability also but that these goals are in conflict and may require separate explanations for each purpose. Explaining to persuade the user of system accuracy may involve hiding information that could induce doubt in the system which is the exact opposite of what a user requiring auditability would want. Before creating explanations in these types, we must first figure out what our users want.

We believe that user requirements for explanation requirements depend on the larger system context in which the explanations exist. Users may not take issue with persuasive explanations in cases where the system is low-stakes such as content recommender systems; de-

cisions in these systems are easily reversible. However, in our case the system we tested with advisors is high stakes, failure to properly advise and support students can lead to students failing to achieve their educational goals. Advisors in such a high stakes system understandably wanted auditability from the explanations within the system, they needed to know how far they could trust it. Additionally, advisors wanted these explanations to provide information about the students that indicated which actions the advisors should take in the student's interest. Actionable explanations again may be dependent on the system context, here advisors are intended to use the rating to know how much support students should need therefore advisors felt that the explanations of "how much support" should include information that also pointed towards "what kinds of support". An example of this within the predictive analytics system is "academic skills". the vendor of the predictive analytics system uses a feature called "Major-Skill Alignment" which is "A proprietary measure of how well a student's current major is aligned with their previously demonstrated academic skills." Surfacing a student's "skill" feature could lead advisors to recommend targeted programming or tutoring that could support the student in building their skills so they could be successful in the major. In order to deploy successful explanations, we need to understand whether our users want explanations to be persuasive, auditable, or actionable.

7.4.5 Conclusion

We completed a study of expert users with a deployed high-stakes intelligent system used in the educational context. We find that current intelligent systems do not properly support users in building mental models of system operation through explanation, nor do they explain in

ways that expert users seem to expect. These results motivate design implications that involve explicit and deliberate introduction to intelligent system outputs and moving towards more interactive explanations. Additionally, system designers should examine the context of their system when deciding what the primary goal of explanations is within it. Designing with these results in mind will lead to improved user experiences within intelligent systems.

Chapter 8

Discussion

I have presented 5 studies that examine core aspects of intelligent systems. These studies focus on accuracy, fairness, and explanation; all of which are essential to building human-centered intelligent systems. Many current systems fall short of this human-centered bar, so my work provides guidance for how to best create new systems in a human-centered fashion.

8.1 Summary

My work on accuracy shows that we must be intentional in our design of intelligent systems so that they manage trade-offs between collecting enough data that allows them to be maximally accurate while also not overly burdening the user. Using an intelligent system that predicts mood in order to recommend positive activities, I demonstrate that actively asking users to reflect on their daily activities greatly improves predictive models compared to simply knowing the user engaged in a specific activity. I also examine how this active reflection pro-

vides further implicit signals to textual predictive models that allow for increased accuracy. I conclude by highlighting how we must balance the user's desire for accuracy versus the cost of users manually entering data which could lead to attrition in usage. My work on fairness shows that common machine learning speech recognition libraries exhibit voice biases that may lead to specific users being disadvantaged. I develop a new method to recognize voice biases in a music application at scale. I then characterize these biases to allow other researchers to understand the limitations of current voice interfaces. Finally, I develop a method that crowdsources further pronunciations of 'disadvantaged' content in order to create aliases that address the voice biases in the voice user interface. Again I conduct this work in order to center the human in this intelligent system; intelligent systems must be designed to work for all humans and not designed out of convenience due to what data is available or what is simple to build.

Next, I examine a common approach to intelligent system explanations and compare this with what users actually desire. I demonstrate that constantly showing explanations can have negative effects on user system perceptions. I continue by examining when users actually want to receive explanations. My work indicates that users prefer explanations to occur only when their expectations of correct system performance are violated. In these cases, users spend longer looking at explanations and come away with better perceptions of the system overall. This work centers the human in intelligent system creation by catering to their needs of when the user wants explanation to happen.

My next exploration of explanations aims to uncover how users want to interact with explanations in intelligent systems. I examine how always-on explanation can be disruptive and increase user skepticism about system results. However, this study examines how users want

to interact with explanations, rather than focusing on when users want explanations. I find that current explanations present many of the problems for users and suggest design principles like progressive disclosure that may solve these problems. Progressive disclosure means providing more interactive explanations that users can repeatedly request information from. Designing explanations in this way conforms better to what users want from their intelligent systems.

Finally, I move beyond the lab studies methods used in current studies of explanation in intelligent systems. I examined a high-stakes deployed intelligent system that makes predictions about student success and explains these predictions through an educational analytics interface. This system is currently being trialed by student advisors as a means to identify and allocate resources to students in need of assistance. I find that many of the lessons learned from my previous lab studies of explanation transfer to this real context. However, the context and complexity of real systems provide some further challenges regarding how advisors understand the system and whether or not they believe it should be used. I derive design implications regarding how users should initially interact with intelligent systems and also how these systems should construct interactive explanation in order to become more human-centered.

The recommendations and methods I have generated in these studies identify various ways that designing for intelligent systems differs from designing traditional systems. My recommendations push back against system design that prioritizes available data and ease of system creation. Rather than creating from a place of convenience, my work advocates for working directly with users and their needs in order to create intelligent systems that are accurate, fair, and explainable.

8.2 Open Problems

While I have contributed towards solutions for the problems I have identified in designing human-centric intelligent systems, this work has also generated new problems and questions about the relationships between the core concepts of accuracy, fairness, and explanation. I briefly discuss these outstanding questions and problems providing a roadmap to future researchers who intend to research human-centered intelligent systems.

One problem concerns the framing of an intelligent system. Recall that one problem in Chapter 7 was that the institution identified a presentational problem with the predictive analytics system they implemented; the advisors had issues with the framing of the “Risk Scores” that it predicted for each individual student. Members of the advising community felt that this was a negative framing that could lead to advisers of this system pushing students away from their desired paths rather than finding support for students. The issues concerning framing and acceptability of machine learning implied here are interesting to study in and of themselves. Advisors feared that the “Risk Scores” would be used unfairly to label students and this was compounded by their fears that the system could be harboring hidden biases against underrepresented students. Seeing these students as “riskier” could lead to advisors acting in ways that compromise that student’s future, e.g. not placing that student in an impacted major that the student desired. However, the framing change towards “Support Level” predictions turns this fairness and bias problem on its head. Rather than seeing students as “risky”, now they are simply labeled as needing more support to succeed. The interesting point here is how this affects the acceptability of the intelligent system for the advisors. The system is identical but the framing

has changed which could lead to differences in uptake and deployment of the system. There is already interesting research in personal informatics and biofeedback that examines this framing [94, 39]; future researchers should explore these framing effects in other contexts, particularly high stakes contexts like student support.

Another open problem involves the practicalities of implementing my design recommendations for user-centric explanations. Chapters 5, 6, and 7 all explored different aspects of how explanations should function. Chapter 5 showed that users want explanations when their expectations are violated, but explanations may provoke doubt at other times. This result raises two possible ways to build explanations.

One possibility is that users explicitly request explanation when they so desire. There are HCI and technical challenges here, how should we allow users to ask for explanations in ways that they provide information about how their expectations are violated. If users can provide this data seamlessly then the system can tailor the explanation to the users' needs. My recommendation is to structure explanation interactively but how to operationalize this is an open problem. Additionally, allowing users to request on-demand explanations creates other problems. For example, users may think they understand how the system is working when they really do not. Another possible problem is that users may not make the effort to ask for an explanation and simply give up on the system when their expectations are violated; this mirrors the trade-offs in Chapter 3 where users are asked to provide more information to make the system more accurate. Similarly, for interactive explanations, we must make sure the payoff is worth the additional cost of the interaction.

The other possibility besides providing on-demand explanations is to detect when and

how the users' expectations are violated. Detecting expectation violation may be a very difficult task that elicits comparisons to detecting user intent. However, there are some approaches that may make it somewhat tractable. One of these is selective explanation, for example, only explaining a prediction if it lies close to a decision boundary in the machine learning model. This could be accomplished quite easily for models such as Logistic Regression and SVMs that have a simple decision boundary but may not generalize easily to new techniques such as deep neural networks. In addition to detecting expectation violation, Further open problems involve real-world effects of explanations.

As I have previously explained, my experiments in Chapter 7 are some of the first results regarding explanation from real-world commercially deployed systems. Given the context, there are a number of follow up studies that this Chapter surfaces. A major one of these is looking at the usage of explanation over time. My study focused on how expert users initially form impressions of an algorithm and explanations but did not examine how interactions with these explanations behave over time. It could be that users interact with explanations frequently initially and then once the user has more calibrated expectations of the system they will interact with the explanations less. These interactions over time should be examined. Additionally, there are questions surrounding the long-term benefit or detriments of explanation. Recall that much of my work has focused on how explanations can have both positive and negative effects on user perceptions of the system and the more recent work shows that users and system designers can have conflicting goals in creating explanations. How do our short term user perception changes due to explanation relate to longer-term satisfaction with the system? In my educational predictive analytics system advisors wanted explanations to point towards actionable steps in helping

students. Would changing the explanations to support further action results in a better user experience for advisors?

My work also creates more questions about when people believe or avoid algorithmic systems. On one hand, we have concepts such as algorithmic omniscience, which I have previously demonstrated in studies [190, 213, 61], and automation bias [42]; both concepts indicating that users may blindly trust intelligent systems and their predictions. On the other hand, we have algorithmic aversion, where users tend to avoid algorithms after seeing them err, even when they would not punish a human prediction the same way [51]. What tips the scale in favor of algorithmic omniscience or algorithm avoidance? I hypothesize that the context matters here but this remains to be verified by future researchers.

In fact, the context of intelligent systems must be explored further in general. As I discussed at the end of Chapter 7, explanations may have different requirements based on system context and user needs. There we explored how users may want auditable or actionable explanations while system designers are often providing persuasive explanations. There are also issues around system context and expectation violation that should be further explored. As we have shown in Chapter 5, expectation violation plays a major role in users' perceptions of the system. We measure expectation violation in a short series of interactions with in-lab system. There are big questions about how this plays out over the lifetime of using an application. Do users begin to understand where the application fails and compensate for this with their behavior? We saw similar behavior in Chapter 7 where users had distinct mental models of what the system could not know. How should we tune systems with regards to precision and recall to create the best user experience? How does this tuning depend on context the system

is deployed in? For example, in a context that has a high cost for false positives, we may want to tune the system differently than a context with a low false positive cost. For example, with the student success predictive analytics system users may want to tune this to have high-recall; given the framing of the system about providing more support to detected students, it is better to identify false positives than it is to have false-negatives.

My work with human-centered AI has generated exciting new research areas to explore. There are extensive open problems regarding interactive explanation, intelligent system context, algorithmic omniscience and avoidance, and how to frame intelligent systems. I hope that myself and other future researchers can use this roadmap to build towards a future that has truly human-centered AI.

Bibliography

- [1] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y. Lim, and Mohan Kankanhalli. Trends and Trajectories for Explainable, Accountable and Intelligible Systems: An HCI Research Agenda. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pages 1–18, Montreal QC, Canada, 2018. ACM Press.
- [2] Icek Ajzen. The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, 50(2):179–211, 1991.
- [3] Ahmed Ali, Walid Magdy, Peter Bell, and Steve Renais. Multi-reference WER for evaluating ASR for languages with no orthographic rules. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 576–580. IEEE, 2015.
- [4] Saleema Amershi, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, Eric Horvitz, Dan Weld, Mihaela Vorvoreanu, Adam Fournery, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, and Paul N. Bennett. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–13, Glasgow, Scotland Uk, 2019. ACM Press.
- [5] Julia Angwin and Jeff Larson. Machine Bias, May 2016.
- [6] Stavros Antifakos, Nicky Kern, Bernt Schiele, and Adrian Schwaninger. Towards improving trust in context-aware systems by displaying system confidence. In *Proceedings of the 7th international conference on Human computer interaction with mobile devices & services - MobileHCI '05*, page 9, Salzburg, Austria, 2005. ACM Press.
- [7] Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. "What is relevant in a text document?": An interpretable machine learning approach. *PLOS ONE*, 12(8):e0181142, August 2017.
- [8] Leif Azzopardi and Vishwa Vinay. Retrievalability: an evaluation measure for higher order information access tasks. In *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, page 561, Napa Valley, California, USA, 2008. ACM Press.
- [9] Ricardo Baeza-Yates. Data and algorithmic bias in the web. pages 1–1. ACM Press, 2016.

- [10] A. Balahur, J. M. Hermida, and A. Montoyo. Building and Exploiting EmotiNet, a Knowledge Base for Emotion Detection Based on the Appraisal Theory Model. *IEEE Transactions on Affective Computing*, 3(1):88–101, January 2012.
- [11] Jakob E. Bardram, Mads Frost, Károly Szántó, and Gabriela Marcu. The MONARCA self-assessment system: a persuasive personal monitoring system for bipolar patients. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, pages 21–30. ACM, 2012.
- [12] Faisal A. Barwais, Thomas F. Cuddihy, and L. Michaud Tomson. Physical activity, sedentary behavior and total wellness changes among sedentary adults: a 4-week randomized controlled trial. *Health and Quality of Life Outcomes*, 11:183, 2013.
- [13] Victoria Bellotti and Keith Edwards. Intelligibility and Accountability: Human Considerations in Context-Aware Systems. *Human-Computer Interaction*, 16(2):193–212, December 2001.
- [14] Frank Bentley, Konrad Tollmar, Peter Stephenson, Laura Levy, Brian Jones, Scott Robertson, Ed Price, Richard Catrambone, and Jeff Wilson. Health Mashups: Presenting Statistical Patterns between Wellbeing Data and Context in Natural Language to Promote Behavior Change. *ACM Transactions on Computer-Human Interaction*, 20(5):1–27, November 2013.
- [15] Catherine T Best, Jason A Shaw, and Elizabeth Clancy. Recognizing Words Across Regional Accents: The Role of Perceptual Assimilation in Lexical Competition. page 6.
- [16] Reuben Binns, Max Van Kleek, Michael Veale, Ulrik Lyngs, Jun Zhao, and Nigel Shadbolt. 'It's Reducing a Human Being to a Percentage'; Perceptions of Justice in Algorithmic Decisions. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pages 1–14, 2018. arXiv: 1801.10408.
- [17] Su Lin Blodgett, Lisa Green, and Brendan O'Connor. Demographic Dialectal Variation in Social Media: A Case Study of African-American English. *arXiv:1608.08868 [cs]*, August 2016. arXiv: 1608.08868.
- [18] Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam T. Kalai. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357, 2016.
- [19] Engin Bozdag. Bias in algorithmic filtering and personalization. *Ethics and Information Technology*, 15(3):209–227, September 2013.
- [20] Margaret M. Bradley, Mark K. Greenwald, Margaret C. Petry, and Peter J. Lang. Remembering pictures: Pleasure and arousal in memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18(2):379–390, March 1992.

- [21] Margaret M. Bradley and Peter J. Lang. Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, 25(1):49–59, 1994.
- [22] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, January 2006.
- [23] V. A. Brockton West Roxbury. Sleep, Sleep Deprivation, and Daytime Activities A Randomized Controlled Trial of the Effect of Exercise on Sleep. *Sleep*, 20(2):95–101, 1997.
- [24] Andrea Bunt, Matthew Lount, and Catherine Lauzon. Are explanations always important?: a study of deployed, low-cost intelligent interactive systems. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 169–178. ACM, 2012.
- [25] John T. Cacioppo, Louise C. Hawkley, Gary G. Berntson, John M. Ernst, Amber C. Gibbs, Robert Stickgold, and J. Allan Hobson. Do Lonely Days Invade the Nights? Potential Social Modulation of Sleep Efficiency. *Psychological Science*, 13(4):384–387, 2002.
- [26] John T. Cacioppo, Richard E. Petty, and Chuan Feng Kao. The Efficient Assessment of Need for Cognition. *Journal of Personality Assessment*, 48(3):306–307, June 1984.
- [27] Lisa A. Cadmus-Bertram, Bess H. Marcus, Ruth E. Patterson, Barbara A. Parker, and Brittany L. Morey. Randomized Trial of a Fitbit-Based Physical Activity Intervention for Women. *American journal of preventive medicine*, 49(3):414–418, September 2015.
- [28] D. Can, E. Cooper, A. Sethy, C. White, B. Ramabhadran, and M. Saraclar. Effect of pronunciations on OOV queries in spoken term detection. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3957–3960, April 2009.
- [29] John M. Carroll and Caroline Carrithers. Training Wheels in a User Interface. *Commun. ACM*, 27(8):800–806, August 1984.
- [30] Electronic Privacy Information Center. EPIC - Algorithms in the Criminal Justice System, 2018.
- [31] Li Chen and Pearl Pu. Interaction design guidelines on critiquing-based recommender systems. *User Modeling and User-Adapted Interaction*, 19(3):167–206, August 2009.
- [32] Yi-Wei Chen and Chih-Jen Lin. Combining SVMs with various feature selection strategies. In *Feature extraction*, pages 315–324. Springer, 2006.
- [33] Pu-Jen Cheng, Jei-Wen Teng, Ruei-Cheng Chen, Jenq-Haur Wang, Wen-Hsiang Lu, and Lee-Feng Chien. Translating Unknown Queries with Web Corpora for Cross-language Information Retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '04*, pages 146–153, New York, NY, USA, 2004. ACM. event-place: Sheffield, United Kingdom.

- [34] Eun Kyoung Choe, Bongshin Lee, Haining Zhu, Nathalie Henry Riche, and Dominikus Baur. Understanding Self-reflection: How People Reflect on Personal Data Through Visual Data Exploration. In *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth '17*, pages 173–182, New York, NY, USA, 2017. ACM.
- [35] Eun Kyoung Choe, Nicole B. Lee, Bongshin Lee, Wanda Pratt, and Julie A. Kientz. Understanding quantified-selfers' practices in collecting and exploring personal data. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 1143–1152, Toronto, Ontario, Canada, 2014. ACM Press.
- [36] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3504–3512. Curran Associates, Inc., 2016.
- [37] Cindy Chung and James W. Pennebaker. The psychological functions of function words. *Social communication*, pages 343–359, 2007.
- [38] Comscore. State of the U.S. Online Retail Economy in Q1 2017, 2017.
- [39] Jean Costa, Alexander T. Adams, Malte F. Jung, François Guimbretière, and Tanzeem Choudhury. EmotionCheck: A Wearable Device to Regulate Anxiety Through False Heart Rate Feedback. *GetMobile: Mobile Comp. and Comm.*, 21(2):22–25, August 2017.
- [40] Henriette Cramer, Paloma de Juan, and Joel Tetreault. Sender-intended functions of emojis in US messaging. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '16*, pages 504–509, Florence, Italy, 2016. ACM Press.
- [41] Pim Cuijpers, Annemieke van Straten, and Lisanne Warmerdam. Behavioral activation treatments of depression: A meta-analysis. *Clinical Psychology Review*, 27(3):318–326, April 2007.
- [42] Mary L. Cummings. Automation bias in intelligent time critical decision support systems. In *AIAA 1st Intelligent Systems Technical Conference*, volume 2, pages 557–562. AIAA, 2004.
- [43] Michelle Cutajar, Edward Gatt, Ivan Grech, Owen Casha, and Joseph Micallef. Comparative study of automatic speech recognition techniques. *IET Signal Processing*, 7(1):25–46, February 2013.
- [44] Cecelia Cutler. Hip-Hop Language in Sociolinguistics and Beyond. *Language and Linguistics Compass*, 1(5):519–538, September 2007.
- [45] Ritwik Dasgupta. Voice User Interface Design. page 114, 2018.

- [46] Fred D. Davis. *A technology acceptance model for empirically testing new end-user information systems : theory and results*. Thesis, Massachusetts Institute of Technology, 1985.
- [47] Edward L. Deci and Richard M. Ryan. The "What" and "Why" of Goal Pursuits: Human Needs and the Self-Determination of Behavior. *Psychological Inquiry*, 11(4):227–268, October 2000.
- [48] Sean A. Dennis, Brian M. Goodson, and Chris Pearson. MTurk Workers' Use of Low-Cost 'Virtual Private Servers' to Circumvent Screening Methods: A Research Note. SSRN Scholarly Paper ID 3233954, Social Science Research Network, Rochester, NY, August 2018.
- [49] Michael A. DeVito, Jeremy Birnholtz, Jeffery T. Hancock, Megan French, and Sunny Liu. How People Form Folk Theories of Social Media Feeds and What it Means for How We Study Self-Presentation. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pages 1–12, Montreal QC, Canada, 2018. ACM Press.
- [50] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, October 2018. arXiv: 1810.04805.
- [51] Berkeley J. Dietvorst, Joseph P. Simmons, and Cade Massey. Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General*, 144(1):114–126, 2015.
- [52] Sergios Dimitriadis and Nikolaos Kyrezi. Linking trust to use intention for technology-enabled bank channels: The role of trusting intentions. *Psychology & Marketing*, 27(8):799–820, 2010.
- [53] K. S. Dobson and R. Joffe. The role of activity level and cognition in depressed mood in a university sample. *Journal of Clinical Psychology*, 42(2):264–271, March 1986.
- [54] Afsaneh Doryab, Mads Frost, Maria Faurholt-Jepsen, Lars V. Kessing, and Jakob E. Bardram. Impact factor analysis: combining prediction with parameter ranking to reveal the impact of behavior on health outcome. *Personal and Ubiquitous Computing*, 19(2):355–365, February 2015.
- [55] SENDER DOVCHIN. Performing Identity Through Language: The Local Practices of Urban Youth Populations in Post-Socialist Mongolia. *Inner Asia*, 13(2):315–333, 2011.
- [56] Mary T. Dzindolet, Scott A. Peterson, Regina A. Pomranky, Linda G. Pierce, and Hall P. Beck. The Role of Trust in Automation Reliance. *Int. J. Hum.-Comput. Stud.*, 58(6):697–718, June 2003.

- [57] Malin Eiband, Hanna Schneider, Mark Bilandzic, Julian Fazekas-Con, Mareike Haug, and Heinrich Hussmann. Bringing Transparency Design into Practice. In *23rd International Conference on Intelligent User Interfaces, IUI '18*, pages 211–223, New York, NY, USA, 2018. ACM.
- [58] Manuela Ekowo and Iris Palmer. THE PROMISE AND PERIL OF PREDICTIVE ANALYTICS IN HIGHER EDUCATION. page 36.
- [59] Ellucian. 2019 Trends to Watch: Higher Education. *Higher Education*, page 15, 2019.
- [60] Motahhare Eslami, Karrie Karahalios, Christian Sandvig, Kristen Vaccaro, Aimee Rickman, Kevin Hamilton, and Alex Kirlik. First I like it, then I hide it: Folk Theories of Social Feeds. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 2371–2382. ACM, 2016.
- [61] Motahhare Eslami, Sneha R. Krishna Kumaran, Christian Sandvig, and Karrie Karahalios. Communicating Algorithmic Process in Online Behavioral Advertising. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18*, pages 432:1–432:13, New York, NY, USA, 2018. ACM.
- [62] Motahhare Eslami, Aimee Rickman, Kristen Vaccaro, Amirhossein Aleyasen, Andy Vuong, Karrie Karahalios, Kevin Hamilton, and Christian Sandvig. I always assumed that I wasn't really that close to [her]: Reasoning about Invisible Algorithms in News Feeds. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 153–162. ACM, 2015.
- [63] M. Faurholt-Jepsen, M. Vinberg, E. M. Christensen, M. Frost, J. Bardram, and L. V. Kessing. Daily electronic self-monitoring of subjective and objective symptoms in bipolar disorder—the MONARCA trial protocol (MONitoring, treAtment and pRediCtion of bipolar disorder episodes): a randomised controlled single-blind trial. *BMJ Open*, 3(7):e003353–e003353, July 2013.
- [64] Steven Feld and Aaron A. Fox. Music and Language. *Annual Review of Anthropology*, 23(1):25–53, 1994.
- [65] Fitbit. Fitbit App & Dashboard, March 2017.
- [66] Aaron A. Fox. *Real Country: Music and Language in Working-Class Culture*. Duke University Press, October 2004. Google-Books-ID: rMqmJ3o9AzwC.
- [67] Batya Friedman and Helen Nissenbaum. Bias in computer systems. *ACM Transactions on Information Systems (TOIS)*, 14(3):330–347, 1996.
- [68] Mads Frost, Afsaneh Doryab, Maria Faurholt-Jepsen, Lars Vedel Kessing, and Jakob E. Bardram. Supporting disease insight through data analysis: refinements of the monarca self-assessment system. page 133. ACM Press, 2013.

- [69] Pedro García García, Enrico Costanza, Jhim Verame, Diana Nowacka, and Sarvapali D. Ramchurn. Seeing (Movement) is Believing: The Effect of Motion on Perception of Automatic Systems Performance. *Human–Computer Interaction*, 0(0):1–51, April 2018.
- [70] Harold Garfinkel. *Studies in Ethnomethodology*. Wiley, January 1991.
- [71] Vijay N. Garla and Cynthia Brandt. Ontology-guided feature engineering for clinical text classification. *Journal of Biomedical Informatics*, 45(5):992–998, October 2012.
- [72] Daniel T. Gilbert, Elizabeth C. Pinel, Timothy D. Wilson, Stephen J. Blumberg, and Thalia P. Wheatley. Immune neglect: a source of durability bias in affective forecasting. *Journal of personality and social psychology*, 75(3):617, 1998.
- [73] Sofia Gkika and George Lekakos. The persuasive role of Explanations in Recommender Systems. page 10, 2014.
- [74] Sharon Goldwater, Dan Jurafsky, and Christopher D. Manning. Which words are hard to recognize? Prosodic, lexical, and disfluency factors that increase speech recognition error rates. *Speech Communication*, 52(3):181–200, March 2010.
- [75] Peter M. Gollwitzer. Implementation intentions: Strong effects of simple plans. *American psychologist*, 54(7):493, 1999.
- [76] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv:1412.6572 [cs, stat]*, December 2014. arXiv: 1412.6572.
- [77] Bryce Goodman and Seth Flaxman. European Union Regulations on Algorithmic Decision-Making and a “Right to Explanation”. *AI Magazine*, 38(3):50–57, October 2017.
- [78] Matasaka Goto, Koji Kitayama, Katunobu Itou, and Tetsunori Kobayashi. Speech Spotter: On-demand Speech Recognition in Human-Human Conversation on the Telephone or in Face-to-Face Situations / Masataka Goto. *8th International Conference on Spoken Language Processing*, page 4, 2004.
- [79] Amit Goyal, Ellen Riloff, and Hal Daumé III. Automatically producing plot unit representations for narrative text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 77–86. Association for Computational Linguistics, 2010.
- [80] Emilio Granell and Carlos-D. Martínez-Hinarejos. A Multimodal Crowdsourcing Framework for Transcribing Historical Handwritten Documents. In *Proceedings of the 2016 ACM Symposium on Document Engineering, DocEng ’16*, pages 157–163, New York, NY, USA, 2016. ACM.
- [81] H. P Grice. *Logic and conversation*. 1975. OCLC: 57327364.

- [82] Enzo Grossi, Nicola Groth, Paola Mosconi, Renata Cerutti, Fabio Pace, Angelo Compare, and Giovanni Apolone. Development and validation of the short version of the Psychological General Well-Being Index (PGWB-S). *Health and Quality of Life Outcomes*, page 8, 2006.
- [83] Chloe Gui and Victoria Chan. Machine learning in medicine. *University of Western Ontario Medical Journal*, 86(2):76–78, December 2017.
- [84] Ido Guy. Searching by Talking: Analysis of Voice Queries on Mobile Web Search. pages 35–44. ACM Press, 2016.
- [85] Matthew W. Hahn and R. Alexander Bentley. Drift as a mechanism for cultural change: an example from baby names. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270(suppl_1), August 2003.
- [86] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of Opportunity in Supervised Learning. *arXiv:1610.02413 [cs]*, October 2016. arXiv: 1610.02413.
- [87] Sandra G. Hart and Lowell E. Staveland. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In Peter A. Hancock and Najmedin Meshkati, editors, *Advances in Psychology*, volume 52 of *Human Mental Workload*, pages 139–183. North-Holland, January 1988.
- [88] Timothy J. Hazen and Issam Bazzi. A comparison and combination of methods for OOV word detection and word confidence scoring. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 1, pages 397–400. IEEE, 2001.
- [89] Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. Grounding Visual Explanations. page 16.
- [90] Caroline Henton. Bitter Pills to Swallow. ASR and TTS have Drug Problems. *International Journal of Speech Technology*, 8(3):247–257, September 2005.
- [91] Denis J. Hilton. Conversational processes and causal explanation. *Psychological Bulletin*, 107(1):65–81, 1990.
- [92] Victoria Hollis, Artie Konrad, Aaron Springer, Chris Antoun, Matthew Antoun, Rob Martin, and Steve Whittaker. What Does All This Data Mean for My Future Mood? Actionable Analytics and Targeted Reflection for Emotional Well-Being. *Human-Computer Interaction*, January 2017.
- [93] Victoria Hollis, Artie Konrad, and Steve Whittaker. Change of Heart: Emotion Tracking to Promote Behavior Change. pages 2643–2652. ACM Press, 2015.
- [94] Victoria Hollis, Alon Pekurovsky, Eunika Wu, and Steve Whittaker. On Being Told How We Feel: How Algorithmic Sensor Feedback Influences Emotion Perception. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(3):114:1–114:31, September 2018.

- [95] Ellen Isaacs, Artie Konrad, Alan Walendowski, Thomas Lennig, Victoria Hollis, and Steve Whittaker. Echoes from the past: how technology mediated reflection improves well-being. page 1071. ACM Press, 2013.
- [96] Donald A. Jackson. Stopping Rules in Principal Components Analysis: A Comparison of Heuristical and Statistical Approaches. *Ecology*, 74(8):2204–2214, December 1993.
- [97] Jawbone. Jawbone | JAMBOX Wireless Speakers | UP Wristband | Bluetooth Headsets, March 2017.
- [98] Anna Jørgensen, Dirk Hovy, and Anders Søgaard. Challenges of studying and processing dialects in social media. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 9–18, Beijing, China, July 2015. Association for Computational Linguistics.
- [99] Daniel Kahneman. *Thinking, Fast and Slow*. Macmillan, October 2011.
- [100] Daniel Kahneman, Paul Slovic, and Amos Tversky. *Judgment Under Uncertainty: Heuristics and Biases*. Cambridge University Press, April 1982. Google-Books-ID: _0H8gwj4a1MC.
- [101] Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. Machine transliteration survey. *ACM Computing Surveys*, 43(3):1–46, April 2011.
- [102] Matthew Kay, Dan Morris, mc schraefel, and Julie A. Kientz. There’s no such thing as gaining a pound: reconsidering the bathroom scale user interface. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing - UbiComp ’13*, page 401, Zurich, Switzerland, 2013. ACM Press.
- [103] Matthew Kay, Shwetak N. Patel, and Julie A. Kientz. How Good is 85%?: A Survey Tool to Connect Classifier Evaluation to Acceptability of Accuracy. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI ’15, pages 347–356, New York, NY, USA, 2015. ACM. event-place: Seoul, Republic of Korea.
- [104] SeungJun Kim, Jaemin Chun, and Anind K. Dey. Sensors Know When to Interrupt You in the Car: Detecting Driver Interruptibility Through Monitoring of Peripheral Interactions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI ’15*, pages 487–496, Seoul, Republic of Korea, 2015. ACM Press.
- [105] René F. Kizilcec. How Much Information?: Effects of Transparency on Trust in an Algorithmic Interface. pages 2390–2395. ACM Press, 2016.
- [106] Artie Konrad, Simon Tucker, John Crane, and Steve Whittaker. Technology and Reflection: Mood and Memory Mechanisms for Well-Being. *Psychology of Well-Being*, 6(1), December 2016.
- [107] Willam J Kountz. Billy Baxter’s Letters, 1899.

- [108] Todd Kulesza, Margaret Burnett, Weng-Keen Wong, and Simone Stumpf. Principles of Explanatory Debugging to Personalize Interactive Machine Learning. In *Proceedings of the 20th International Conference on Intelligent User Interfaces - IUI '15*, pages 126–137, Atlanta, Georgia, USA, 2015. ACM Press.
- [109] Todd Kulesza, Weng-Keen Wong, Simone Stumpf, Stephen Perona, Rachel White, Margaret M. Burnett, Ian Oberst, and Andrew J. Ko. Fixing the program my computer learned: barriers for end users, challenges for the machine. In *Proceedings of the 13th international conference on Intelligent user interfaces - IUI '09*, page 187, Sanibel Island, Florida, USA, 2008. ACM Press.
- [110] B. C. Kwon, M. Choi, J. T. Kim, E. Choi, Y. B. Kim, S. Kwon, J. Sun, and J. Choo. RetainVis: Visual Analytics with Interpretable and Interactive Recurrent Neural Networks on Electronic Medical Records. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2018.
- [111] Peter Ladefoged and Keith Johnson. *A Course in Phonetics*. Cengage Learning, January 2014. Google-Books-ID: NOYbCgAAQBAJ.
- [112] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Interpretable & Explorable Approximations of Black Box Models. *arXiv:1707.01154 [cs]*, July 2017. arXiv: 1707.01154.
- [113] Nicholas D. Lane, Mu Lin, Mashfiqui Mohammad, Xiaochao Yang, Hong Lu, Giuseppe Cardone, Shahid Ali, Afsaneh Doryab, Ethan Berke, Andrew T. Campbell, and Tanzeem Choudhury. BeWell: Sensing Sleep, Physical Activities and Social Interactions to Promote Wellbeing. *Mobile Networks and Applications*, 19(3):345–359, June 2014.
- [114] Randy J. Larsen. Toward a Science of Mood Regulation. *Psychological Inquiry*, 11(3):129–141, July 2000.
- [115] Marcia L Laskey and Carole J Hetzel. Investigating Factors Related to Retention of At-risk College Students. page 13.
- [116] Antoine Laurent, Sylvain Meignier, and Paul Deléglise. Improving recognition of proper nouns (in ASR) through generation and filtering of phonetic transcriptions. *Computer Speech and Language*, 28(4):979–996, 2014.
- [117] Steve Lawrence and C Lee Giles. Accessibility of information on the web. 400:3, 1999.
- [118] Min Kyung Lee, Daniel Kusbit, Evan Metsky, and Laura Dabbish. Working with Machines: The Impact of Algorithmic and Data-Driven Management on Human Workers. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, pages 1603–1612, Seoul, Republic of Korea, 2015. ACM Press.
- [119] Peter M. Lewinsohn and Christopher S. Amenson. Some relations between pleasant and unpleasant mood-related events and depression. *Journal of Abnormal Psychology*, 87(6):644–654, 1978.

- [120] Shoushan Li, Lei Huang, Rong Wang, and Guodong Zhou. Sentence-level Emotion Classification with Label and Context Dependence. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1045–1053, Beijing, China, July 2015. Association for Computational Linguistics.
- [121] Robert LiKamWa, Yunxin Liu, Nicholas D. Lane, and Lin Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 389–402. ACM, 2013.
- [122] Brian Y. Lim and Anind K. Dey. Assessing demand for intelligibility in context-aware applications. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 195–204. ACM, 2009.
- [123] Brian Y. Lim and Anind K. Dey. Investigating intelligibility for uncertain context-aware applications. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 415–424. ACM, 2011.
- [124] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. *Why and why not* explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*, page 2119, Boston, MA, USA, 2009. ACM Press.
- [125] James J. Lin, Lena Mamykina, Silvia Lindtner, Gregory Delajoux, and Henry B. Strub. Fish’n’Steps: Encouraging physical activity with an interactive computer game. In *International Conference on Ubiquitous Computing*, pages 261–278. Springer, 2006.
- [126] Zachary C. Lipton. The Mythos of Model Interpretability. *arXiv:1606.03490 [cs, stat]*, June 2016. arXiv: 1606.03490.
- [127] Shen Liu, Yang Xie, James Mcgree, and Zongyuan Ge. Computational and statistical methods for analysing big data with applications, 2016.
- [128] Yin Lou, Rich Caruana, and Johannes Gehrke. *Intelligible Models for Classification and Regression*.
- [129] Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. page 10.
- [130] Daniel Luzzati, Cyril Grouin, Ioana Vasilescu, Martine Adda-Decker, Eric Bilinski, Nathalie Camelin, Juliette Kahn, Carole Lailier, Lori Lamel, and Sophie Rosset. Human annotation of ASR error regions: Is "gravity" a sharable concept for human annotators? In *LREC*, 2014.
- [131] Hao Ma, Raman Chandrasekar, Chris Quirk, and Abhishek Gupta. Improving search engines using human computation games. In *Proceeding of the 18th ACM conference*

on *Information and knowledge management - CIKM '09*, page 275, Hong Kong, China, 2009. ACM Press.

- [132] Douglas MacPhillamy and Peter Lewinsohn. The Pleasant Events Schedule: Studies on Reliability, Validity, and Scale Intercorrelation. *Journal of Consulting and Clinical Psychology*, 50(3):363–380, 1982.
- [133] Daniel McDuff, Amy Karlson, Ashish Kapoor, Asta Roseway, and Mary Czerwinski. AffectAura: an intelligent system for emotional memory. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 849–858. ACM, 2012.
- [134] Sean M. McNee, John Riedl, and Joseph A. Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI '06 extended abstracts on Human factors in computing systems - CHI EA '06*, page 1097, Montré#233;al, Qu#233;bec, Canada, 2006. ACM Press.
- [135] Rishabh Mehrotra, Ashton Anderson, Fernando Diaz, Amit Sharma, Hanna Wallach, and Emine Yilmaz. Auditing Search Engines for Differential Satisfaction Across Demographics. *arXiv:1705.10689 [cs]*, May 2017. arXiv: 1705.10689.
- [136] Merriam-Webster. What Does 'Lit' Mean? | Merriam-Webster.
- [137] B. Micenková, R. T. Ng, X. Dang, and I. Assent. Explaining Outliers by Subspace Separability. In *2013 IEEE 13th International Conference on Data Mining*, pages 518–527, December 2013.
- [138] Tim Miller. Explanation in Artificial Intelligence: Insights from the Social Sciences. *arXiv:1706.07269 [cs]*, June 2017. arXiv: 1706.07269.
- [139] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, February 2018.
- [140] Andrew Moody and Yuko Matsumoto. “Don’t Touch My Moustache”: Language Blending and Code Ambiguation by Two J-Pop Artists. *Asian Englishes*, 6(1):4–33, June 2003.
- [141] Kathleen L. Mosier, Linda J. Skitka, Susan Heers, and Mark Burdick. Automation Bias: Decision Making and Performance in High-Tech Cockpits. *The International Journal of Aviation Psychology*, 8(1):47–63, January 1998.
- [142] Mozilla. Common Voice by Mozilla.
- [143] Bonnie M. Muir and Neville Moray. Trust in automation. Part II. Experimental studies of trust and human intervention in a process control simulation. *Ergonomics*, 39(3):429–460, March 1996.

- [144] Saurabh Nagrecha, John Z. Dillon, and Nitesh V. Chawla. MOOC Dropout Prediction: Lessons Learned from Making Pipelines Interpretable. In *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*, pages 351–359, Perth, Australia, 2017. ACM Press.
- [145] Lloyd H. Nakatani and John A. Rohrlich. Soft machines: A philosophy of user-computer interface design. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems - CHI '83*, pages 19–23, Boston, Massachusetts, United States, 1983. ACM Press.
- [146] Clifford Nass and Youngme Moon. Machines and Mindlessness: Social Responses to Computers. *Journal of Social Issues*, 56(1):81–103, January 2000.
- [147] Kenneth Olmstead. Voice assistants used by 46% of Americans, mostly on smartphones | Pew Research Center.
- [148] Cathy O’Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown, 2016.
- [149] José Manuel Ortega Egea and María Victoria Román González. Explaining physicians’ acceptance of EHCR systems: An extension of TAM with trust and risk factors. *Computers in Human Behavior*, 27(1):319–332, January 2011.
- [150] Aasish Pappu, Teruhisa Misu, and Rakesh Gupta. Investigating Critical Speech Recognition Errors in Spoken Short Messages. In Alexander Rudnicky, Antoine Raux, Ian Lane, and Teruhisa Misu, editors, *Situated Dialog in Speech-Based Human-Computer Interaction*, pages 71–82. Springer International Publishing, Cham, 2016.
- [151] Carolina Parada, Mark Dredze, Denis Filimonov, and Frederick Jelinek. Contextual Information Improves OOV Detection in Speech. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10*, pages 216–224, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
- [152] Carolina Parada, Abhinav Sethy, Mark Dredze, and Frederick Jelinek. A spoken term detection framework for recovering out-of-vocabulary words using the web. In *Eleventh Annual Conference of the International Speech Communication Association*, 2010.
- [153] Eli Pariser. *The Filter Bubble: What The Internet Is Hiding From You*. Penguin Books Limited, May 2011.
- [154] Acacia C. Parks, Matthew D. Della Porta, Russell S. Pierce, Ran Zilca, and Sonja Lyubomirsky. Pursuing happiness in everyday life: The characteristics and behaviors of online happiness seekers. *Emotion*, 12(6):1222–1234, 2012.
- [155] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent

- Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:28252830, October 2011.
- [156] S. Tejaswi Peesapati, Victoria Schwanda, Johnathon Schultz, Matt Lepage, So-yaee Jeong, and Dan Cosley. Pensieve: Supporting Everyday Reminiscence. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2027–2036, New York, NY, USA, 2010. ACM.
- [157] James W. Pennebaker. *The Secret Life of Pronouns: What Our Words Say About Us*. Bloomsbury Press, 1 edition edition, August 2011.
- [158] JW Pennebaker, RJ Booth, and ME Francis. Linguistic Inquiry and word count: LIWC [Computer Software], 2007.
- [159] Manuel Perea, Jon Andoni Duñabeitia, and Manuel Carreiras. Masked associative/semantic priming effects across languages with highly proficient bilinguals. 2008.
- [160] Ellen Peters, Judith Hibbard, Paul Slovic, and Nathan Dieckmann. Numeracy Skill And The Communication, Comprehension, And Use Of Risk-Benefit Information. *Health Affairs*, 26(3):741–748, May 2007.
- [161] Richard E. Petty and John T. Cacioppo. The Elaboration Likelihood Model of Persuasion. *Advances in Experimental Social Psychology*, 19:123–205, January 1986.
- [162] Katri Peuhkuri, Nora Sihvola, and Riitta Korpela. Diet promotes sleep duration and quality. *Nutrition Research*, 32(5):309–319, May 2012.
- [163] PIP. Home I, 2018.
- [164] Robert Plutchik. The Nature of Emotions. *American Scientist*, 89:344–350, 2001.
- [165] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, March 1980.
- [166] Mashfiqui Rabbi, Min Hane Aung, Mi Zhang, and Tanzeem Choudhury. MyBehavior: automatic personalized health feedback from user behaviors and preferences using smart-phones. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 707–718. ACM, 2015.
- [167] Mazin G. Rahim and Chin-Hui Lee. String-based minimum verification error (SB-MVE) training for speech recognition. *Computer Speech & Language*, 11(2):147–160, April 1997.
- [168] Inioluwa Deborah Raji and Joy Buolamwini. Actionable Auditing: Investigating the Impact of Publicly Naming Biased Performance Results of Commercial AI Products. page 7.

- [169] Antoine Raux and Maxine Eskenazi. Optimizing Endpointing Thresholds Using Dialogue Features in a Spoken Dialogue System. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, SIGdial '08, pages 1–10, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. event-place: Columbus, Ohio.
- [170] Lena Reed, Jiaqi Wu, Shereen Oraby, Pranav Anand, and Marilyn Walker. Learning Lexico-Functional Patterns for First-Person Affect. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 141–147, Vancouver, Canada, July 2017. Association for Computational Linguistics.
- [171] Byron Reeves and Clifford Nass. *The Media Equation: How People Treat Computers, Television, and New Media like Real People and Places*. Cambridge University Press, September 1996.
- [172] David Reinheimer and Kelly McKenzie. The Impact of Tutoring on the Academic Success of Undeclared Students. *Journal of College Reading and Learning*, 41(2):22–36, March 2011.
- [173] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, New York, NY, USA, 2016. ACM.
- [174] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [175] Michael D. Robinson and Gerald L. Clore. Belief and feeling: Evidence for an accessibility model of emotional self-report. *Psychological Bulletin*, 128(6):934–960, 2002.
- [176] Robert Rosenthal. Teacher Expectancy Effects: A Brief Update 25 Years after the Pygmalion Experiment. *Journal of Research in Education*, 1(1):3–12, 1991.
- [177] Peter J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.
- [178] Stephanie Rude, Eva-Maria Gortner, and James Pennebaker. Language use of depressed and depression-vulnerable college students. *Cognition and Emotion*, 18(8):1121–1133, December 2004.
- [179] James A. Russell. Culture and the categorization of emotions. *Psychological Bulletin*, 110(3):426–450, November 1991.
- [180] James A. Russell and Geraldine Pratt. A description of the affective quality attributed to environments. *Journal of Personality and Social Psychology*, 38(2):311–322, 1980.

- [181] Attapol T. Rutherford, Fuchun Peng, and Françoise Beaufays. Pronunciation learning for named-entities through crowd-sourcing. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.
- [182] Mark Savage. BBC Sound of 2013: Chvrches - BBC News, 2012.
- [183] Emanuel A. Schegloff. Repair After Next Turn: The Last Structurally Provided Defense of Intersubjectivity in Conversation. *American Journal of Sociology*, 97(5):1295–1345, 1992.
- [184] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. Methods and metrics for cold-start recommendations. page 253. ACM Press, 2002.
- [185] Schwitters. w88888888. 1923.
- [186] Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.
- [187] Martin E. P. Seligman, Tracy A. Steen, Nansook Park, and Christopher Peterson. Positive Psychology Progress: Empirical Validation of Interventions. *American Psychologist*, 60(5):410–421, 2005.
- [188] David Canfield Smith. Designing the Star User Interface. page 21.
- [189] Aaron Springer and Henriette Cramer. "Play PRBLMS": Identifying and Correcting Less Accessible Content in Voice Interfaces. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 296:1–296:13, New York, NY, USA, 2018. ACM.
- [190] Aaron Springer, Victoria Hollis, and Steve Whittaker. Dice in the Black Box: User Experiences with an Inscrutable Algorithm. March 2017.
- [191] Aaron Springer, Victoria Hollis, and Steve Whittaker. Mood modeling: accuracy depends on active logging and reflection. *Personal and Ubiquitous Computing*, pages 1–15, March 2018.
- [192] Aaron Springer and Steve Whittaker. What are You Hiding? Algorithmic Transparency and User Perceptions. In *2018 AAAI Spring Symposium Series*, 2018.
- [193] Aaron Springer and Steve Whittaker. Making Transparency Clear. *Algorithmic Transparency for Emerging Technologies Workshop*, page 5, 2019.
- [194] Aaron Springer and Steve Whittaker. Progressive Disclosure: Designing for Effective Transparency. In *Proceedings of the 24th International Conference on Intelligent User Interfaces - IUI '19*, 2019.
- [195] Litza Stark, Steve Whittaker, and Julia Hirschberg. ASR Satisficing: The effects of ASR accuracy on speech retrieval. In *In Proceedings of International Conference on Spoken Language Processing*, pages 1069–1072, 2000.

- [196] Arthur A. Stone, Joseph E. Schwartz, David Schkade, Norbert Schwarz, Alan Krueger, and Daniel Kahneman. A population approach to the study of emotion: Diurnal rhythms of a working day examined with the day reconstruction method. *Emotion*, 6(1):139–149, 2006.
- [197] Simone Stumpf, Vidya Rajaram, Lida Li, Weng-Keen Wong, Margaret Burnett, Thomas Dietterich, Erin Sullivan, and Jonathan Herlocker. Interacting meaningfully with machine learning systems: Three experiments. *International Journal of Human-Computer Studies*, 67(8):639–662, August 2009.
- [198] Lucy A. Suchman. *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge University Press, November 1987. Google-Books-ID: AJ_eBJtHxmsC.
- [199] Rachael Tatman. Gender and Dialect Bias in YouTube’s Automatic Captions. *EACL 2017*, page 53, 2017.
- [200] Auke Tellegen. Structures of mood and personality and their relevance to assessing anxiety, with an emphasis on self-report. In A. H. Tuma and J. D. Maser, editors, *Anxiety and the anxiety disorders*, pages 681–706. Lawrence Erlbaum Associates, Inc, Hillsdale, NJ, US, 1985.
- [201] Erik R. Thomas. Phonological and Phonetic Characteristics of African American Vernacular English. *Language and Linguistics Compass*, 1(5):450–475, September 2007.
- [202] Crispin Thurlow. DAOL: Generation Txt?.. page 27, 2003.
- [203] Dianne M. Tice, Ellen Bratslavsky, and Roy F. Baumeister. Emotional distress regulation takes precedence over impulse control: If you feel bad, do it! *Journal of Personality and Social Psychology*, 80(1):53–67, 2001.
- [204] David O. Trouilloud, Philippe G. Sarrazin, Thomas J. Martinek, and Emma Guillet. The influence of teacher expectations on student achievement in physical education classes: Pygmalion revisited. *European Journal of Social Psychology*, 32(5):591–607, September 2002.
- [205] Joe Tullio, Anind K. Dey, Jason Chalecki, and James Fogarty. How It Works: A Field Study of Non-technical Users Interacting with an Intelligent System. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 31–40, New York, NY, USA, 2007. ACM.
- [206] R Turner, M Ward, and D Turner. BEHAVIORAL TREATMENT FOR DPRESSION: AN EVALUATION OF THERAPEUTIC COMPONENTS. *Journal of Clinical Psychology*, 35(1):167–175, January 1979.
- [207] Kristen Vaccaro, Dylan Huang, Motahhare Eslami, Christian Sandvig, Kevin Hamilton, and Karrie Karahalios. The Illusion of Control: Placebo Effects of Control Settings. In

Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI '18, pages 16:1–16:13, New York, NY, USA, 2018. ACM.

- [208] Angela van Barneveld, Kimberly E Arnold, and John P Campbell. Analytics in Higher Education: Establishing a Common Language. page 11.
- [209] Ewald van der Westhuizen and Thomas Niesler. Automatic Speech Recognition of English-isiZulu Code-switched Speech from South African Soap Operas. *Procedia Computer Science*, 81:121–127, 2016.
- [210] W. Richard Walker, John J. Skowronski, and Charles P. Thompson. Life is pleasant—and memory helps to keep it that way! *Review of General Psychology*, 7(2):203–210, 2003.
- [211] 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.
- [212] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T. Campbell. StudentLife: assessing mental health, academic performance and behavioral trends of college students using smart-phones. pages 3–14. ACM Press, 2014.
- [213] Jeffrey Warshaw, Tara Matthews, Steve Whittaker, Chris Kau, Mateo Bengualid, and Barton A. Smith. Can an Algorithm Know the "Real You"?: Understanding People's Reactions to Hyper-personal Analytics Systems. pages 797–806. ACM Press, 2015.
- [214] Angela Watercutter. Crazy Characters Help Indie Bands Outsmart Google.
- [215] Philip C. Watkins, Andrew Mathews, Donald A. Williamson, and Richard D. Fuller. Mood-congruent memory in depression: Emotional priming or elaboration? *Journal of Abnormal Psychology*, 101(3):581, 1992.
- [216] Daniel S. Weld and Gagan Bansal. The Challenge of Crafting Intelligible Intelligence. *arXiv:1803.04263 [cs]*, March 2018. arXiv: 1803.04263.
- [217] Jenna Wiens and Erica S. Shenoy. Machine Learning for Healthcare: On the Verge of a Major Shift in Healthcare Epidemiology. *Clinical Infectious Diseases*, 66(1):149–153, January 2018.
- [218] Elizabeth J. Wilson and Daniel L. Sherrell. Source effects in communication and persuasion research: A meta-analysis of effect size. *Journal of the Academy of Marketing Science*, 21(2):101–112, March 1993.
- [219] Woebot. Woebot - Your charming robot friend who is here for you, 24/7, 2018.

- [220] Kewen Wu, Yuxiang Zhao, Qinghua Zhu, Xiaojie Tan, and Hua Zheng. A meta-analysis of the impact of trust on technology acceptance model: Investigation of moderating influence of subject and context type. *International Journal of Information Management*, 31(6):572–581, December 2011.
- [221] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig. Achieving Human Parity in Conversational Speech Recognition. *arXiv:1610.05256 [cs, eess]*, October 2016. arXiv: 1610.05256.
- [222] Qian Yang, Alex Scuito, John Zimmerman, Jodi Forlizzi, and Aaron Steinfeld. Investigating How Experienced UX Designers Effectively Work with Machine Learning. In *Proceedings of the 2018 on Designing Interactive Systems Conference 2018 - DIS '18*, pages 585–596, Hong Kong, China, 2018. ACM Press.
- [223] Qian Yang, Aaron Steinfeld, and John Zimmerman. Unremarkable AI: Fitting Intelligent Decision Support into Critical, Clinical Decision-Making Processes. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–11, Glasgow, Scotland Uk, 2019. ACM Press.
- [224] Rayoung Yang and Mark W. Newman. Learning from a learning thermostat: lessons for intelligent systems for the home. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing - UbiComp '13*, page 93, Zurich, Switzerland, 2013. ACM Press.
- [225] Ming Yin, Jennifer Wortman Vaughan, and Hanna Wallach. Understanding the Effect of Accuracy on Trust in Machine Learning Models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–12, Glasgow, Scotland Uk, 2019. ACM Press.
- [226] Ed Yong. A Popular Algorithm Is No Better at Predicting Crimes Than Random People - The Atlantic, 2018.
- [227] Kun Yu, Shlomo Berkovsky, Ronnie Taib, Dan Conway, Jianlong Zhou, and Fang Chen. User Trust Dynamics: An Investigation Driven by Differences in System Performance. pages 307–317. ACM Press, 2017.
- [228] Emre Yılmaz, Henk van den Heuvel, and David van Leeuwen. Investigating Bilingual Deep Neural Networks for Automatic Recognition of Code-switching Frisian Speech. *Procedia Computer Science*, 81:159–166, 2016.
- [229] Antonette M. Zeiss, Peter M. Lewinsohn, and Ricardo F. Muñoz. Nonspecific improvement effects in depression using interpersonal skills training, pleasant activity schedules, or cognitive training. *Journal of consulting and clinical psychology*, 47(3):427, 1979.
- [230] Miriam Zisook, Sara Taylor, Akane Sano, and Rosalind Picard. SNAPSHOT Expose: Stage Based and Social Theory Based Applications to Reduce Stress and Improve Well-being. 2016.