A Slow Algorithm Improves Users' Assessments of the Algorithm's Accuracy

JOON SUNG PARK, University of Illinois at Urbana-Champaign, USA
RICK BARBER, University of Illinois at Urbana-Champaign, USA
ALEX KIRLIK, University of Illinois at Urbana-Champaign, USA
KARRIE KARAHALIOS, University of Illinois at Urbana-Champaign, USA

With computational algorithms making an increasing number of deeply consequential, and often problematic judgments on our behalf, there is a growing interest in slowing down technology to encourage users to reflect on judgments made by algorithms. Prior work in slow technology has established slowness as an agent of reflection and serendipity; however, it has been unclear whether this waiting time actually helps users gain useful insight or any other benefits as they make judgments using an algorithm. To this end, we conducted a series of online and in-person between-subject user studies in which we isolate the impact of an algorithm's speed on how users incorporate the algorithm's advice when making judgments in the context of simple visual recognition tasks. We find that our participants followed good quality algorithms more and bad quality algorithms somewhat less if the response time of the algorithm is slower. Furthermore, qualitative analysis of the in-person study interviews reveals that the waiting was not time wasted, but was often used to reflect on the task and the estimation process of themselves and the algorithm, and to compare and reevaluate the two processes. Based on these findings, we outline design implications of future algorithmic systems.

CCS Concepts: • **Human-centered computing → Empirical studies in interaction design**.

Additional Key Words and Phrases: Slow technology; human-algorithm interaction; trust in automation; design guidelines

ACM Reference Format:
Joon Sung Park, Rick Barber, Alex Kirlik, and Karrie Karahalios. 2019. A Slow Algorithm Improves Users' Assessments of the Algorithm's Accuracy. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 102 (November 2019), 15 pages. https://doi.org/10.1145/3359204

1 INTRODUCTION

Algorithmic systems are ubiquitous; powered by the recent advances in machine learning, algorithms play an important role advising and influencing our actions anywhere from how we search for information [4] to how we make decisions like whom to send to jail [2] or whom to date [19]. But these algorithms are not perfect. Growing number of studies report cases of algorithms handcuffed by a flawed training dataset, returning to the users biased and seemingly inappropriate

Authors' addresses: Joon Sung Park, University of Illinois at Urbana-Champaign, 201 N Goodwin Ave, Urbana, Illinois, USA, 61801, jp19@illinois.edu; Rick Barber, University of Illinois at Urbana-Champaign, 201 N Goodwin Ave, Urbana, Illinois, USA, 61801, barber5@illinois.edu; Alex Kirlik, University of Illinois at Urbana-Champaign, 201 N Goodwin Ave, Urbana, Illinois, USA, 61801, kirlik@illinois.edu; Karrie Karahalios, University of Illinois at Urbana-Champaign, 201 N Goodwin Ave, Urbana, Illinois, USA, 61801, kkarahal@illinois.edu.

2573-0142/2019/11-ART102 $15.00
https://doi.org/10.1145/3359204

responses like marking a defendant likely to commit crimes in the future based on his race [2] or determining that patients with asthma has a better chance of surviving pneumonia [20].

Given an imperfect algorithmic system, it is important for the users to assess, as well as possible, the accuracy of the algorithm so as to know not only when to listen, but also when not to listen to the algorithm. To date, most of the effort in this regard has been focused on making the model learned by the algorithms interpretable and helping users understand why certain inputs map to certain outputs [8, 18, 25]. But in this paper, we go beyond the commonly focused topic of making these models interpretable by exploring how the speed of user's interaction with these algorithms could affect the user's ability to assess the algorithm's accuracy. In particular, resonating with the growing concern that the increasingly fast-paced digital environment that we inhabit might be toxic for the way we interact with these algorithmic systems [9, 33], we explore how slowness could improve user's ability to assess the algorithm's accuracy.

Our paper makes contributions to the existing literature by isolating the impact of an algorithm's speed on how users incorporate the algorithm's advice while performing a simple visual recognition task. Ever since Lars Hallnäs and Johan Redström first discussed the merits of slow technology in their seminal article [12], scholars in human-computer interaction and design have explored the topic of reevaluating the values of waiting time as moments of reflection, mental rest, and a catalyst for serendipity [22]. But despite the increasing interest the agenda of slow technology remains a work in progress with one of the outstanding criticisms being that reflection and insight do not go hand in hand [10, 28]. Some have expressed concern that reflection without gaining insight is pointless, or may even be harmful as the users may develop an inaccurate understanding of the systems they are using when not given enough information about the system [17, 28]. However, in a series of online (study 1, n=140; study 2, n=200) and in-person (study 3, n=32) between-subject user studies where the participants were asked to estimate the number of jelly beans in a jar with the help of an algorithm's suggestion, we find evidence that users are better at assessing the accuracy of an algorithm's advice if the speed of the algorithm is slow. More specifically, we find that:

- People adhere to a slower algorithm more if the output accuracy is good.
- People adhere to a slower algorithm somewhat less if the output accuracy is bad.

In addition to this, our work contributes to furthering our understanding of a user's mental model during the waiting time by qualitatively analyzing the content of the interviews that took place at the end of the in-person study. We find that for many of our participants, the waiting time was not time wasted; the time was often used to reflect on the problem at hand and the estimation process of themselves and the algorithm, and to introspectively compare and reevaluate the two processes. Some specifically appreciated the waiting time for giving them a chance to rethink their own estimation before being primed by the estimation of the algorithm, helping them avoid blindly following, or blindly dismissing the algorithm's outputs.

2 RELATED WORK

2.1 Pace as a Complex Element of Human-Computer Interaction

2.1.1 Study on System Response Time. The pace of interaction with our technology has been important, and widely studied area in human-computer interaction. In 1968, Robert B. Miller summarized 17 unique situations and tasks such as simple data entry and page navigation that can arise while using computers of his time, and qualitatively presented guidelines for an acceptable response time in each case [21]. He proposed that the context of the interaction is integral to the process of defining the appropriate response time. For example, if a user is simply pressing down on a key to enter a character in a command-line interface, the character should show up on the screen with almost no delay. However, if the user is engaged in a much more complex process like

restructuring multiple columns of tabular data, the user may be willing to wait significantly longer than two seconds. Following up on Miller's guidelines, for the next decade and a half, scholars experimentally tested the acceptable range of response times for various tasks, and expanded our understanding of what needs to be considered when deciding on acceptable wait times. The results suggest that faster does not necessarily mean better; while a short response time of under one second is preferred for user satisfaction and productivity in most tasks discussed by Miller [31, 37], it also leads to an increase in error rates for certain tasks, lowering the overall quality of user's work [3, 5].

In recent years, with the rise of algorithmic platforms such as search engines and social media news feeds, there has been renewed interest in response time in regards to algorithmic systems. But the efforts have been focused mostly on the users' satisfaction, concluding that fast interaction is almost always preferred. For example, Google conducted online experiments in which the response time of search outputs was intentionally delayed by 100 to 400 milliseconds and saw a significant drop in the number of searches per user [27]. Similarly, Bing experimented by adding an intentional server delay of 50 to 2,000 milliseconds and observed a decrease in not only the number of searches, but also in users' engagement with the search results [27]. These findings led to the technology platforms heavily optimizing for speed even at the cost of the output quality of the algorithms; search engines search through a previously cached, incomprehensive set of available documents even at the cost of returning less relevant information [32], while social media news feeds (e.g., Facebook's Newsfeed) prioritize on showing fast loading content [11].

2.1.2 The Slow Movement in Technology. The prime movement in technology that is challenging the recent focus on speed is the slow movement that began in 1986 with an activist in Italy protesting against the opening of a fast-food chain restaurant, advocating for a slower, traditional and mindful way of eating [36]. Over the years, the slow movement has also had its influence on thinkers in technology, as evident from Hallnäs and Redström's influential work from 2001 "Slow Technology – Designing for Reflection" that first presented us with a vision of designing technological artifacts that are focused not on efficiency and performance, but rather on reflection and mental rest by creating technological artifacts that are meant to be consumed slowly, over a long period of time [12].

Since then, the agenda of slow technology has been gradually applied to various causes like supporting better social connections through online messaging with temporal delay [34] and experiences of anticipation through a printer that prints nostalgic images from the user's photo library each month [23]. But recently, there has been a growing interest in applying the framework of slow technology to how we interact with algorithms. Search engines today, for example, not only retrieve simple facts and related documents, but also return answers to complex questions like where one should take vacation, or have dinner. In cases like these, a "slow search" as proposed by Jaime Teevan et al. can take extra time to look at a more comprehensive set of information available and return more relevant and useful information to the users [14, 32]. Not only that, for the users, the waiting time can be used to reflect and let the mind wander to increase the chance of serendipitous discovery [7], or to slowly think about the decisions being offered by the algorithm and ponder on its potential biases or flaws [9].

There certainly have been many doubts and critiques toward the idea of simply slowing down technology [17]. Long response times can be frustrating for all stakeholders of the system, and simply waiting a long time may not result in new useful insight for the user [10, 28]. However, the recent trend of overly focusing on fast interaction that is in part driven by behavioral advertising that benefits from high user engagement and satisfaction [14, 29] leaves room for revisiting the question of how fast we should interact with algorithms. A successful design of human-computer

interaction goes well beyond allowing for users' productivity, efficiency, and engagement, and have to take into consideration different contexts in which algorithms are deployed. For algorithmic systems like GPS navigators that are used often and needs to deliver instructions to drivers in real time, it might be preferred, or even necessary, for the speed of the interaction to be fast. But when designing for algorithmic systems that are not time-sensitive, and that are responsible for giving us consequential advice like who will go to jail or who needs serious medical attention, we should consider slowing down our interaction with these systems if it means we are better at noticing its potential flaws. In this way, we see the contribution of slow technology as not advocating for unreasonable and meaningless delays in our digital lives, but offering us a framework to think about the optimal amount of time needed to process the information and decisions presented to us much the same way the early pioneers of human-computer interaction approached system response time.

2.2 Users' Interaction with Algorithms

Our work contributes to the growing body of literature that explores users' interaction with algorithms by revisiting the effect of the waiting time on the users. In particular, we study whether the slower interaction with an algorithmic system could benefit the users' ability to assess the accuracy of the algorithm. This topic is particularly relevant today as the usage of algorithms become ubiquitous despite their flaws. The number of cases where algorithmic systems, handicapped by training datasets that are often flawed and biased, returning inappropriate responses like marking a defendant more likely to commit crimes in the future based on his race [2] or judging that patients with asthma has a better chance of surviving pneumonia [20] are growing. Rather than blindly following, or rejecting the suggestion made by an algorithm, it is important to encourage the users to become the judge of when to follow the algorithm.

To this end, numerous studies have explored the relationship between an algorithm's interpretabilty, transparency, and users' assessment of the algorithm's performance [8, 18, 25, 38]. For example, Poursabzi-Sangdeh et al. found that transparency towards an algorithm's attributes affect the users' abilities to detect the algorithm's mistakes [24]. At the same time, the recent human-computer interaction approach to this challenge has focused on studying how to communicate the accuracy of an algorithm to the users. In a controlled experiment where the participants were given an estimate of the algorithm's accuracy, Yin et al. found that people's trust in an algorithm correlates with the stated accuracy of the algorithm [38]. In this paper, we look at this challenge through the lens of the slow movement by studying how the pace of interaction could benefit the users' ability to assess the algorithm's accuracy.

Research Question (RQ). *Would a slow algorithm improve users' assessments of the algorithm's accuracy?*

3 METHOD

We conducted two online (study 1, n=140; study 2, n=200) and one in-person (study 3, n=32) between-subject user studies in which participants were assigned with a simple visual challenge of estimating the number of jelly beans in a jar with the help of an algorithm. In all three studies, participants were presented with five images of jelly bean jars one at a time, and were asked to make an initial estimation of how many jelly beans were in each jar. After each time they recorded their initial estimation, the participants were given advice from an algorithm of varying response times and accuracy about what the correct number of jelly beans in the jar might be. The participants then had a chance to change their answer and record their final estimation.

It is worth noting that quantitatively measuring how closely a participant follows an algorithm's estimation is challenging. To this end, the task of estimating the number of jelly beans was carefully

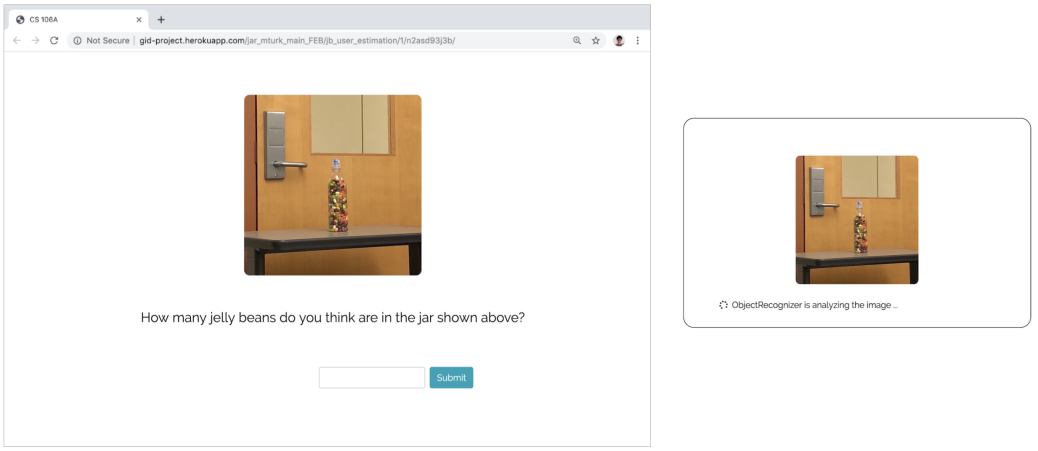

Fig. 1. Example page of the custom web page used in the studies with one of the images of a jelly bean container (left), and the loading GIF beneath the image that was shown to the participants while they were waiting for the algorithm to return its response (right).

chosen based on the prior literature. Although the task could be considered simple and somewhat artificial, early studies in psychology that studied people's tendency to conform to the other's opinion [13], or that studied the wisdom of crowds [30] have frequently employed this particular task because it allows for an easy comprehension on the part of the subjects, quantitative standards of measurement, central point from which to measure dispersion, and sufficient range for expression of opinion so that no one might hold a more extreme opinion or judgment than is provided for in the scale of measurement [13]. And importantly for this study, for many of the participants, the task was not so mechanical in the sense that it was not immediately clear how difficult it would be for a computer algorithm to make an accurate estimation.

Given this, despite its simplicity, the jelly bean task gives us a convincing experimental conditions to study how people's adherence to an algorithm's suggestion changes by observing how much the participants would change their initial estimation on the number of jelly beans to adhere to the algorithm's advice given the varying response times and algorithm's accuracy. If a slower response time improves users' assessments of the algorithm's accuracy, we should observe that the participants' confidence in the slow algorithm's output will strongly correlate with its accuracy.

Hypothesis 1 (H1). *Given an accurate advice from the algorithm, participants will adhere to the algorithm's advice more and exhibit higher confidence in the algorithm's accuracy if the response time of the algorithm is slower.*

Hypothesis 2 (H2). *Given an inaccurate advice from the algorithm, participants will adhere to the algorithm's advice less and exhibit less confidence in the algorithm's accuracy if the response time of the algorithm is slower.*

In the remainder of this section, we describe our study procedure and participants for each of the three studies.

3.1 Study 1 (Online)

3.1.1 Participants. A total of 140 participants on Amazon Mechanical Turk (MTurk) completed Study 1 that took 14.3 minutes on average to finish. Participants consented to participate once at the beginning of the study, and once at the end of the study when they were debriefed. They had to be at least 18 years old, living in the US, and have completed at least 100 Human Intelligence task (HITs – MTurk's task unit) with at least a 95% HIT approval rate. The mean age score was 4.60 (SD=0.96; 3="18-24 years old," 4="25-34 years old"), and 45 of them identified themselves as female. In addition, 59 of the participants held a bachelor's degree, 19 held a higher degree, and the rest a high school diploma or some high school-level education. The sample was 76.43% Caucasian, 8.05% Hispanic, 6.45% Asian, 5.00% African American, and 2.01% Native American, and 0.07% other. After discussions and pilot studies, we expected the participants to take roughly 12 minutes or less. To this end, the participants were initially paid $1.50 for their time through the standard payment system of MTurk. Our post-study analysis revealed, however, that the participants in Study 1 took longer than our expectation. Therefore, following the recent practice of using MTurk's bonus system that allows requesters to pay the workers extra money after the initial payment, we paid every participants in this study extra $0.30 (for an example, see [35]). This ensured us that participants were paid at least the US Federal minimum wage of $7.25 per hour.

3.1.2 Procedure. When the participants accepted the task on MTurk platform, they were redirected to a custom built website for this study and randomly placed into one of 14 categories: a combination of seven different response times (1, 5, 15, 30, 45, 60, and 75 seconds) and two algorithm accuracy (high accuracy in which the algorithm's advice were off by only 2% from the correct answer, and low accuracy in which the algorithm overestimated the correct answer by 100%). There were 10 participants in each category. Similar to a procedure used in a study that explored human's perception of algorithms [16], we started the study by providing the participants the following definition of algorithms: "Algorithms are processes or sets of rules that a computer follows in calculations or other problem-solving operations" [16]. The participants were then given a brief explanation that machine vision algorithms are actively researched type of algorithms which focus on understanding the contents of videos or images, and that a group of university researchers have developed a version of a machine vision algorithm named ObjectRecognizer that can count the number of jelly beans in a container from a photo of the container.

In the study, the participants were shown five images one at a time, each of unique and transparent jars with 520, 450, 660, 730, and 590 jelly beans in this order of appearance. After each image was shown, the participants were asked to record how many jelly beans they thought were in the container. They were then presented with a button on the website to start running the algorithm to get its estimation with the following explanation: "Now, you will run our machine vision algorithm, ObjectRecognizer, in real-time. Once you receive its suggested answer, you are welcome to change your final answer as much, or as little as you want." When the participant pressed the button, all participants were shown a commonly used loading GIF until the algorithm returned its estimation. The amount of time participants had to wait before the algorithm returned its estimation, and the accuracy of its estimation, were based on categories the participants were placed in. No further instructions were given during the waiting time. Finally, the study ended with a short survey that included a manipulation check and a short survey about the participants' confidence level in the algorithm's output accuracy in a 7 point Likert scale, and demographics.

3.2 Study 2 (Online)

3.2.1 Participants. A total of 200 participants on Amazon Mechanical Turk (MTurk) completed Study 2 that took 13.9 minutes on average to complete. Participants in Study 2 were recruited and

paid through the same procedure as Study 1; the participants were initially paid $1.50 through the standard payment system of MTurk, and later received a bonus of $0.30 to ensure that they were compensated at least the US Federal minimum wage for their time. The mean age score was 4.60 (SD=1.05; 3="18-24 years old," 4="25-34 years old"), and 74 of them identified themselves as female. Also, 94 of the participants held a bachelor's degree, 21 held a higher degree, and the rest a high school diploma or some high school-level education. The sample was 75.0% Caucasian, 7.50% Hispanic, 5.0% Asian, 8.50% African American, and 2.0% Native American, and 1.50% other.

3.2.2 Procedure. Study 2's procedure was identical to that of Study 1. However, informed by the results from Study 1, we narrowed down our participant categories to four: a combination of two different response times (1 second and 45 seconds) and two algorithm accuracy, which were the same as the ones stated in Study 1. Each category had 50 participants.

3.3 Study 3 (In-person)

3.3.1 Participants. We recruited a total of 32 participants around a university town in the Midwest region of the United States through flyers and an online newsletter for an in-person study that took around 45 minutes to complete. Participants consented to participate once at the beginning of the study, and once at the end of the study when they were debriefed. The participants were asked to come in to the lab and were paid $10 for their time. The mean age score was 4.13 (SD=1.58; 3="18-24 years old," 4="25-34 years old"), and 22 of them identified themselves as female. And 90 of the particpants held a bachelor's degree, 6 held a higher degree, and the rest a high school diploma or some high school-level education. The sample was 59.3% Caucasian, 12.5% Hispanic, 21.9% Asian, and 6.25% African American.

3.3.2 Procedure. Study 3 is an in-person replication of Study 2. We invited participants who responded to our flyers and online newsletter to our lab. The participants were then randomly placed in one of the four categories used in Study 2, with each category having 8 participants. The participants were then directed to the same custom website used in Study 2 on a laptop that was provided by the researcher who conducted all in-person sessions. Rest of the procedure follows that of Study 2. At the end of the study, however, the researcher conducted a 10 to 15 minutes exit interview with the participants to explore the participants' mental model while waiting for the algorithm to return its estimation.

3.4 Measures

Below are the measures we used to test our hypotheses in all three studies. As mentioned above, Study 3 included an exit interview in addition to these measures.

3.4.1 Adherence to the algorithm's advice. To measure how closely the participants followed the advice from the algorithm, we calculated how much the participants changed their initial estimation of the number of jelly beans towards the algorithm's suggestion. In our analysis, we only focus on the first iteration of estimating the number of jelly beans out of the five due to the learning effect that occurs as the iterations continue.

Additionally, we highlight the following observations in our results to help justify this measure:

- *Consistent distribution:* For all three studies, participants were randomly assigned into one of the categories in the study. Our results show that the distribution of the average initial estimation during the first iteration were not significantly different between different categories.
- *Similar starting point:* Participants in all three studies started from relatively similar initial estimations with most of them underestimating the number of jelly beans, on average by

around 200. To further restrict the variance of the initial condition, we also analyzed our results with only the participants whose first initial estimation was within one standard deviation away from the mean, and found our findings to replicate.

- *A benefit from the good algorithm:* That a lot of our participants underestimated the number of jelly beans by around 200 on average meant the participants who received advice from a good algorithm (off by only 2% from the correct answer) almost always benefited from adhering closely to the algorithm.
- *A harm from the bad algorithm:* This also meant that for the participants who received advice from a bad algorithm (overestimated the correct answer by 100%), they were almost always better off not listening to the algorithm's suggestion.

3.4.2 Confidence in the algorithm's accuracy. Complementing the above measure, we also measured the participants' level of confidence during the exit survey in 7 point Likert scale with the following question adopted from a previous research [6]: "How confident were you in the ObjectRecognizer algorithm's estimate?" Although self-reported measures are not as strong as behavioral measures, in our results, we find this measure to correlate with the behavioral measure described above of how closely the participants adhered to the algorithm's advice.

3.4.3 Definition of algorithm manipulation check. Following previous research, the participants were asked the following open-ended question [16]: "In your own words, please briefly explain what you think algorithms are." The answers to this question confirmed that our participants understood algorithms as autonomous decision-makers.

3.5 Analysis

We used the one-sided Mann-Whitney U test in order to test our hypotheses that 1) the participants will adhere more to the advice of a slow algorithm given highly accurate output, and that 2) the participants will adhere less to the advice of a slow algorithm given inaccurate output. To analyze the main themes discussed by the participants during the exit interview in Study 3, two researchers labeled the interview transcription using line-by-line open coding. We revised our labeling through a collaborative and iterative process, and then used axial coding to extract the relationship between themes.

4 RESULTS

We summarize our findings from the three studies here. In subsection 4.1 and 4.2, we focus on the quantitative measures described above to explore how the response time of our algorithm affected our participants' process of estimation, and their confidence in the algorithm's output accuracy. In subsection 4.3, 4.4, and 4.5, we take a qualitative approach and thematically analyze the contents of the exit interview in Study 3 to help elucidate what is driving the results in the earlier subsections.

4.1 Users are Better at Assessing the Accuracy of the Slower Algorithm

4.1.1 For an accurate algorithm, users trust the slower algorithm more (H1). Our results confirm H1. In all three studies, when given a good (2% error rate) advice from the algorithm, participants who received advice from a somewhat slow algorithm with a response time of 45 seconds changed their initial estimation to a number much closer to the algorithm's estimation than the participants who received advice from a fast algorithm with a response time of 1 second (Study 1, $Z=84.0$, $p=0.0056$; Study 2, $Z=1548.0$, $p=0.02$; Study 3, $Z=52.0$, $p=0.02$). Additionally, we also notice that given an accurate algorithm, participants in the slow algorithm group were a little more confident in the output of the algorithm than participants in fast algorithm group for all three studies.

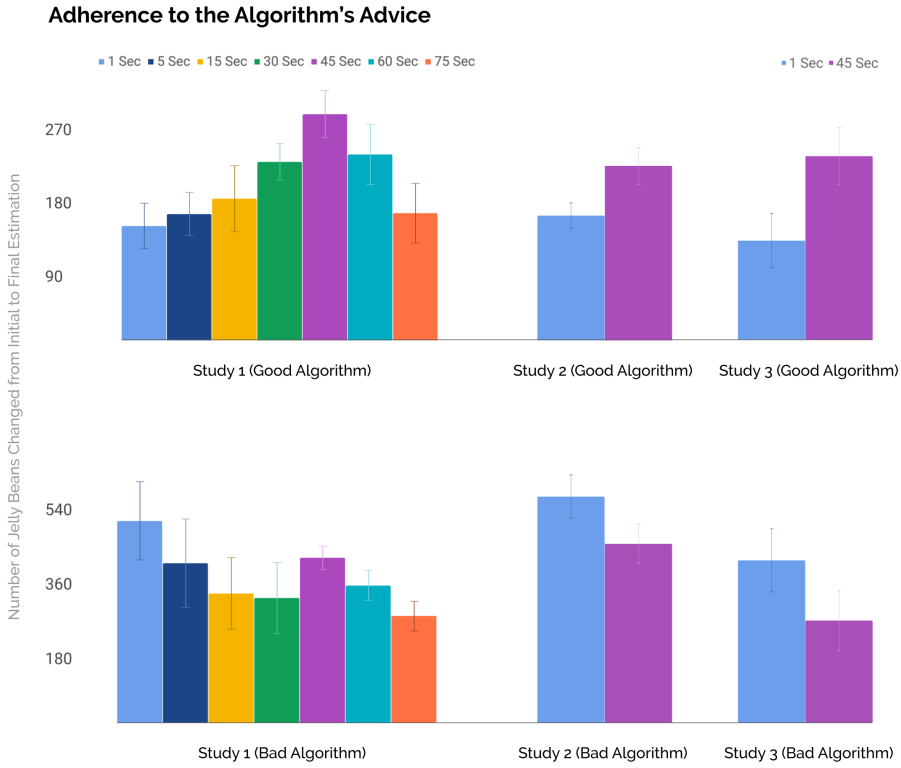

Fig. 2. Figures summarizing the participants' degree of adherence to the advice given by the algorithm. When given an accurate algorithm, the participants changed their initial response more towards the algorithm's suggestion if the algorithm were slower. But when given an inaccurate algorithm, the participants changed their initial response more towards the algorithm's suggestion if the algorithm were faster.

4.1.2 For an inaccurate algorithm, users trust the slower algorithm somewhat less (H2). However, if the accuracy of the algorithm's output is low (100% overestimation), we see the opposite trend. Across all three studies, we find some evidence that weakly supports H2; when given an inaccurate advice from the algorithm, participants who received advice from a slow algorithm with a response time of 45 seconds changed their initial estimation less than the participants who received advice from a fast algorithm with a response time of 1 second (Study 1, Z=43.0, p=0.31; Study 2, Z=1025.0, p=0.06; Study 3, Z=21.0, p=0.13). Similarly, given an inaccurate algorithm, participants in the slow algorithm group were a little less confident in the output of the algorithm than participants in the fast algorithm group.

Given the number of participants we were able to recruit, we do not claim statistical significance for our findings in the bad algorithm's case. However, our results here indicate that a slow algorithm did not cause our participants to blindly trust its suggestion, but rather encouraged our participants to better recognize an accurate algorithm. Thus we answer our overarching research question: in the context of our study, a slow algorithm improves users' assessments of the algorithm's accuracy.

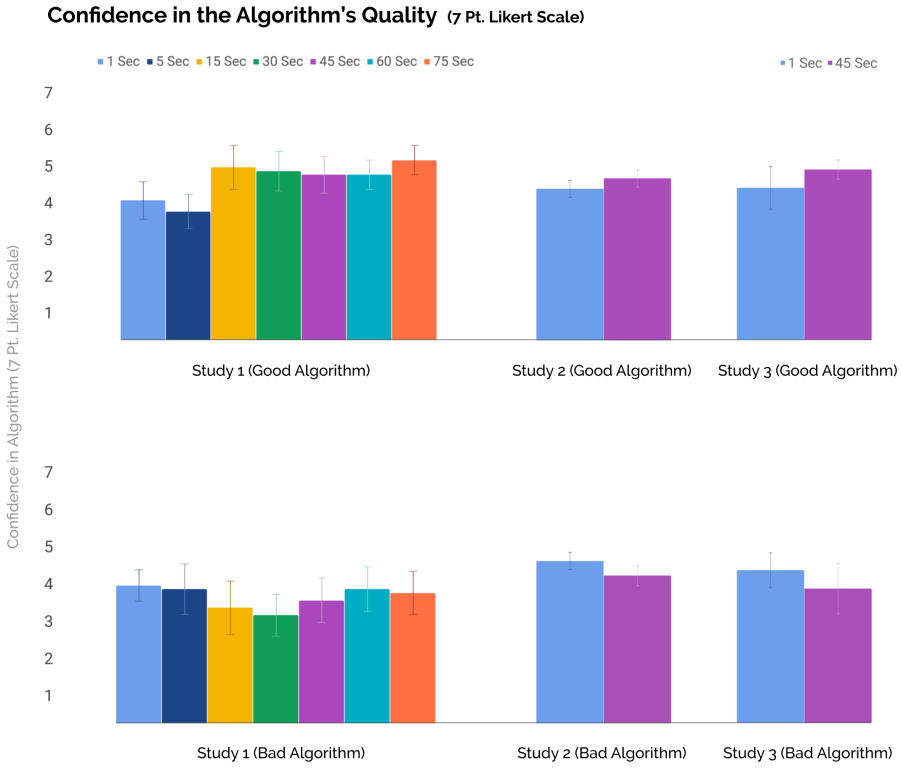

Fig. 3. Figures summarizing the participants' degree of confidence in the algorithm. When given an accurate algorithm, the participants were more confident in the algorithm if it were slower. But when given an inaccurate algorithm, the participants were more confident in the algorithm if it were faster.

4.2 There is an Optimal Response Time for an Algorithm

Despite our findings presented above, it was not the case that the algorithm could be indefinitely slower to be beneficial to the participants. Instead, the results from Study 1 show that participants were most trusting of the good algorithm when its response time was approximately 45 seconds, and least trusting of the bad algorithm when its response time was approximately 30 seconds. Previous literature has shown that in the context of simple tasks involving computers such as data entry, there is an optimal system response time for decreasing the user's error rates, and that the response time should be neither too long nor too short [3, 5, 26]. Our result seems to indicate that this trend extends into more algorithmic tasks like the one presented in the three studies above; there is a better response time for the algorithm that provides cognitive benefits to the participants who are making a decision with this algorithm.

4.3 Waiting Time is a Chance for Reflection

Having observed that participants were better at assessing the accuracy of a slow algorithm in our results presented above, we moved on to explore the mental model of our participants as they were waiting for the algorithm to return its answer by thematically analyzing the exit interview in Study 3. After the participants completed their estimation tasks and the demographics survey, we asked our participants to openly describe what they were thinking about as they were waiting

for the algorithm to return its estimation of the number of jelly beans. A longer waiting time corresponded with a higher likelihood of the participants reflecting about the estimation task at hand. While almost all participants in the slow algorithm group (87.5%) had noted that they actively thought about the task at some point while they were waiting, only some of the participants in the fast algorithm group (37.5%) had noted the same. This is an expected result considering that the participants in the fast algorithm group barely had any time to think at all as illustrated in the following quote: "It wasn't very long to wait. So I didn't have time to think about it a whole lot" (P22).

However, for the 20 participants who reported to have used the waiting time to reflect on the process of estimation, we see a clear thematic pattern arise. Of these participants, 10 participants reported to have reflected on their own process of estimating the number of jelly beans to improve their answers. For example, one participant mentioned: "as I was waiting, I was like still trying to look at the jar to see like maybe if I change my response based on like taking more time to inspect the jar and try to like guess how tall it was" (P14). Another 10 participants reported to have reflected on the process of the algorithm, speculating what and how it would estimate: "I think it would distinguish the jelly beans by, like, the pixel colors, but I'm not sure if it actually does that. I don't know. I was trying to think of ways that the algorithms did it" (P9). Seven participants tried to compare their own process of estimation with that of the algorithm: "... the first time, I was more thinking about the algorithm and how it was [estimating]... but overtime I thought differently. I'm like, okay, now I still want to know what numbers coming up, but how does that fit with the numbers that I'm estimating? Are there patterns in that" (P9)?

4.4 There were Benefits to Waiting and Reflection

4.4.1 Reflection led to new insight about the task, but not about the algorithm. Some participants have noted that they gained more insight about how to estimate the number of jelly beans during the waiting time. One particularly convincing insight that four of the participants who reported to have used the waiting time for reflection mentioned was them noticing that there was a door knob in the background of the photos of the jelly bean that could be used to better speculate how big the container would be in real life: "I kind of consciously realized that the picture, I could get a sense of how tall something was because it showed it in relation to the door with the handle" (P19). However, when asked about whether the waiting time helped them better understand the algorithm's estimation process, the answer was, perhaps unsurprisingly, negative with 4 out of 10 participants who reported to have reflected on the process of the algorithm specifically mentioning that they do not understand how the algorithm works, and the others left guesses they were not confident in: "[the waiting] made me feel like I understood the algorithm a little bit more, but it still is kind of like a black box. I wouldn't know" (P9).

4.4.2 The waiting time gave participants time to reflect before seeing the algorithm's answer. According to some of our participants, however, a convincing benefit to a longer waiting time came from the fact that the participants in the slow group had a chance to think over their estimations before seeing and being influenced by the algorithm's estimation. Here are quotes from two different participants, both of whom received an inaccurate advice that overestimated the correct answer by 100%. Whereas the participant of the first quote received advice from a fast algorithm, the participant of the second quote received advice from a slow algorithm.

> I think once I saw the algorithm's answer, I was more inclined to be like, that's probably right. Whereas if maybe I had more time to think about my own answer, I would have felt more comfortable with mine and less inclined to just blindly adjust my answer

> compared to the algorithm's answer... Because once I, once I made my guess and then I instantly see the algorithm's then it's like, oh, okay. (P20)

Expressing a similar sentiment, a participant who was given an advice from a slow and inaccurate algorithm mentioned:

> While I was waiting for the algorithm's prediction, I kind of just was like thinking over my answer and... I decided like, okay, mine is more accurate before seeing the prediction and then after seeing the prediction, I think that time allowed me to I guess like reaffirm my prediction. (P27)

For P20, seeing the algorithm's estimation right away made the participant much more likely to blindly trust the algorithm when it presented an estimation that was likely too high. On the other hand, for P27, the waiting time gave the participant an opportunity to reassess the accuracy of the participant's own estimation, and helped the participant be less influenced by the algorithm's bad estimation. P20 likened this effect to having an answer sheet right next to you when you are solving a problem set to study for a test; knowing that the answers to the problems are right there, you would exert much less effort to solve the problems and rush into check whether your preliminary answer matches with what is in the answer sheet. This would be harmful to the participants' ability to come up with a better answer on their own, causing them to be less able to judge the accuracy of an algorithm.

4.5 Slowness May Lead to Frustration for Some Users

Within the context of our study, 29 of the participants in both the fast and slow algorithm groups found the respective response time to be acceptable. But it would be misleading to not point out that four of the 16 participants in the slow algorithm group made remarks on the slowness of the algorithm and had hoped for the algorithm to be a little faster: "I think it was like kind of long, but it wasn't like too long. Um, but for me, I feel like I'm relatively impatient and so I just like, wanted to know [the algorithm's estimation] right away" (P12). This response, in part, seems to depend on the participants' existing expectations that stem from previous experience with technology and algorithms in general. A couple of participants in the slow algorithm group suspected that the algorithm might have crashed or was programmed inefficiently. Similarly, a participant in the fast algorithm group was surprised by how fast the algorithm returned the estimation based on the participant's previous experience running programs: "I've been doing a lot of like Matlab homework and I kind of equated that to that... I think analyzing images is harder for computers than it is for us. Especially with all the different colors of the jelly beans and that sort of thing. So that was kind of impressive how well it did and how quickly it went" (P7).

5 DISCUSSION

5.1 Design Implications

There are important design implications that stem from these findings, one of which is to start re-imagining our relationship with technology in contemporary judgment and decision-making scenarios. Ever since Douglas Engelbart presented his 1968 demo of his user interface, the goal of human-computer interaction has been to augment human intelligence rather than to undermine or replace it [15]. Even though intelligent algorithms today make judgments and decisions that are seemingly as good as, or even better than, those of humans, it would be unwise for us to fully delegate *all* decision-making tasks to machines and be subjected to their potential biases and flaws. Rather than blindly accepting or rejecting the decisions made by algorithms, perhaps we can use waiting time as a time to reflect on and assess the algorithm in the process of decision-making.

It is important to note here again, however, that for certain contexts of interaction, slowness might not be the right choice to consider at all. Certain algorithmic systems such as GPS navigators and disaster relief algorithms require immediacy to be useful. The same is true for other algorithms such as matching algorithms that match ridesharing drivers to passengers, which we interact with frequently in low-stakes environments. In these cases, slowness could cause frustration, inefficiency, and harm. Additionally, it must be recognized that slow interaction with computing systems can lower the efficiency and productivity of the users, and cause frustration for all stakeholders. This could affect not only the users of the systems, but also the business interests of the companies that are creating and maintaining these algorithms in use today, potentially hindering more widespread consideration for slower interaction.

It is clear that we need to further study ways in which slowness can add value to the users, not simply by encouraging them to reflect, but by actively enriching and transforming the waiting time to benefit users. To this end, previous work in slow technology has included work on slowing the response time of a search engine to return better quality results to users [32], or designing interaction with serendipity in mind [1, 7]. But by showing that *simply* slowing down the pace of interaction could provide benefits to the user, we add to previous efforts that have explored slowness in conjunction with other human-computer interaction elements.

5.2 Limitations and Future Work

It is important not to over-generalize the findings presented here. We do not conclude from our work that slowing down all algorithms would result in the outcomes and benefits presented in this paper. Earlier scholars in human-computer interaction have shown us that finding the optimal speed for interaction is complicated and context-dependent in traditional computing systems [21]. What we have tried to show is that this logic could apply to our interaction with algorithmic systems. To this end, we have studied how users interact with algorithms of varying response times in an experimental algorithmic setup that estimates the number of jelly beans in a jar. In doing so, we have shown that the slow response time of an algorithm can bring cognitive benefits to the users in certain contexts.

Future work needs to verify whether our findings hold under different and more ecologically valid contexts. Three intuitive axes to pursue further include time sensitivity, algorithm opaqueness, and high-stakes contexts. In the time domain, we suspect future work will focus on human and algorithm interaction contexts in which the task being performed is not time sensitive. For example, slowness may be useful in the process of complex exploratory search and analysis of information where time is not of the essence, and the user may need to actively reflect on what to search for and how to analyze the information. Also, when an algorithmic system is opaque and the users are not fully aware of how and why the algorithm makes judgments as it is the case for many commercial algorithmic systems, slowness might be a particularly relevant element in the interaction that could help users better assess the algorithm's impact and accuracy. Finally, some of the motivating examples of algorithmic system usage we presented in this paper included cases in which algorithms help us decide whom to send to jail or whom to date in which the algorithms are known to make inappropriate decisions partly due to a flawed training dataset or biased metrics. These are often high-stakes scenarios in which wrong decisions seriously impact the well-being of stakeholders of the decisions. Future work should investigate whether slower interaction with such algorithmic systems would result in similar benefits.

6 CONCLUSION

Our work presents empirical evidence that there can be benefits for users in slowing down the response time of algorithms. In the context of our study, the waiting time was often used by the

participants to reflect on the task and the estimation process of themselves and the algorithm, and to compare and reevaluate the two processes. This process of reflection often produced benefits to the participants' ability to make decisions with an algorithm by helping them better understand the task, and by encouraging them to more carefully evaluate their answers. This resulted in a statistically significant difference between those who interacted with a fast algorithm and those who interacted with a slow algorithm in the measures we used to determine how well the participants evaluated the accuracy of the algorithm's output.

Without a doubt, slowing down our technology comes at a cost. Prior work discussed in this paper has shown that a slow response time of computing systems, whether it's the traditional computers explored by the early scholars in human-computer interaction [26] or the newer algorithms [32], could lead to lower satisfaction, productivity, and engagement. Even in our results, a few of the participants who interacted with the slow algorithm expressed some degree of frustration and hoped for a faster interaction. However, decisions in an ever growing number of areas such as the justice system, the employment market, and the medical field are being made by algorithms. These are deeply consequential decisions that could have profound impact on individuals and society. Perhaps then, users' satisfaction, productivity, and engagement – some of the most widely used dimensions to evaluate our technology – might not be the right measures to optimize for. If slowing down our technology offers us an opportunity to make better, and more conscious decisions with algorithms, we need to consider new evaluation metrics to explore and experiment with the potential of slow algorithms.

ACKNOWLEDGMENTS

We would like to thank the members of the Social Spaces Group for their feedback throughout the designing and writing process of this work. We also appreciate the valuable comments from the anonymous reviewers who helped greatly to improve this paper.

REFERENCES

[1] Paul André, m.c. schraefel, Jaime Teevan, and Susan T. Dumais. 2009. Discovery is never by chance: designing for (un)serendipity. *Proceedings of the seventh ACM conference on Creativity and cognition* (October 2009), 305–314.

[2] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias. *ProPublica* (May 2016). https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[3] Raymond E. Barber and Henry C. Lucas Jr. 1983. System response time, operator productivity and job satisfaction. *Commun. ACM* 26, 11 (November 1983), 972–986.

[4] Frank Bentley, Katie Quehl, Jordan Wirfs-Brock, and Melissa Bica. 2019. Understanding Online News Behaviors. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (April 2019), 11.

[5] Nigel Bevan. 1981. Is there an optimum speed for presenting text on a VDU? *International Journal Man-Machine Studies* 14, 1 (1981), 59–76.

[6] Berkeley J. Dietvorst, Joseph P. Simmons, and Cade Massey. 2015. Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General* 144, 1 (February 2015), 114.

[7] Marian Dörk, Peter Bennett, and Rosamund Davies. 2013. Taking our sweet time to search. *Proceedings of CHI 2013 Workshop on Changing Perspectives of Time in HCI* (April 2013).

[8] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *CoRR* (2017).

[9] Yael Eisenstat. 2019. The real reason tech struggles with algorithmic bias. *Wired* (February 2019). https://www.wired.com/story/the-real-reason-tech-struggles-with-algorithmic-bias/

[10] Anthony M Grant. 2003. The impact of life coaching on goal attainment, metacognition and mental health. *Social Behavior and Personality: an international journal* 31, 3 (2003), 253–263.

[11] Michael Guta. 2017. Facebook will put faster loading sites top of news feeds. *Small Business Trends* (August 2017). https://smallbiztrends.com/2017/08/facebook-news-feed-will-favor-faster-sites.html

[12] Lars Hallnäs and Johan Redström. 2001. Slow technology–designing for reflection. *Personal and ubiquitous computing* 5, 3 (2001), 201–212.

[13] Arthur Jenness. 1932. The role of discussion in changing opinion regarding a matter of fact. *The Journal of Abnormal and Social Psychology* (1932).

[14] Eric Johnson. 2018. Is it time for a 'slow food' movement for the internet? Retrieved April 3, 2019 from https://www.recode.net/2018/9/12/17848368/nicole-wong-cto-google-twitter-slow-food-tech-internet-congress-regulation-kara-swisher-podcast
[15] James Landay. 2019. Smart Interfaces for Human-Centered AI. *HAI* (2019), March.
[16] Min Kyung Lee. 2018. Understanding perception of algorithmic decisions: Fairness, trust, and emotion in response to algorithmic management. *Big Data & Society* 5, 1 (March 2018).
[17] Siân Lindley, Robert Corish, Elsa Kosmack Vaara, Pedro Ferreira, and Vygandas Simbelis. 2013. Changing perspectives of time in HCI. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. ACM, 3211–3214.
[18] Zachary C. Lipton. 2016. The mythos of model interpretability. *CoRR* (2016).
[19] Ally Marotti. 2018. Algorithms behind Tinder, Hinge and other dating apps control your love life. Here's how to navigate them. *Chicago Tribune* (December 2018). https://www.chicagotribune.com/business/ct-biz-app-dating-algorithms-20181202-story.html
[20] Tafari Mbadiwe. 2017. The Potential Pitfalls of Machine Learning Algorithms in Medicine. *Pulmonology Advisor* (December 2017). https://www.pulmonologyadvisor.com/home/topics/practice-management/the-potential-pitfalls-of-machine-learning-algorithms-in-medicine/
[21] Robert B. Miller. 1968. Response time in man-computer conversational transactions. *Proceedings Spring Joint Computer Conference* (1968), 267–277.
[22] William Odom, Richard Banks, Abigail Durrant, David Kirk, and James Pierce. 2012. Slow technology: critical reflection and future directions. In *Proceedings of the Designing Interactive Systems Conference*. ACM, 816–817.
[23] William T. Odom, Abigail J. Sellen, Richard Banks, David S. Kirk, Tim Regan, Mark Selby, Jodi L. Forlizzi, and John Zimmerman. 2014. Designing for slowness, anticipation and re-visitation: a long term field study of the photobox. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (April 2014), 1961–1970.
[24] Forough Poursabzi-Sangdeh, Daniel G. Goldstein, Jake Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. 2018. Manipulating and Measuring Model Interpretability. *CoRR* (2018).
[25] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should I trust you?: Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (February 2016).
[26] Ben Shneiderman. 1984. Response time and display rate in human performance with computers. *ACM Computing Surveys (CSUR)* 16, 3 (1984), 265–285.
[27] Eric Shurman and Jake Brutlag. 2009. Performance related changes and their user impact. *Velocity 2009* (2009).
[28] Daniel Stein and Anthony M. Grant. 2014. Disentangling the relationships among self-reflection, insight, and subjective well-being: The role of dysfunctional attitudes and core self-evaluations. *The Journal of psychology* 148, 5 (September 2014), 505–522.
[29] Bijan Stephen. 2018. Time is different now ... and so are we. *The Verge* (November 2018). https://www.theverge.com/2018/11/25/18111179/facebook-twitter-global-news-engagement-perception-of-time
[30] James Surowiecki. 2004. The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations. *Doubleday; Anchor* (2004).
[31] Goodman T. and Spence R. 1978. The effects of system response time on users of interactive computer systems. *SIGGRAPH '78 Proceedings of the 5th annual conference on Computer graphics and interactive techniques* (August 1978), 100–104.
[32] Jaime Teevan, Kevyn Collins-Thompson, Ryen W White, Susan T Dumais, and Yubin Kim. 2013. Slow search: Information retrieval without time constraints. In *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*. ACM, 1.
[33] Nitasha Tiku. 2019. The soothing promise of our own artisinal internet. *Wired* (February 2019). https://www.wired.com/story/soothing-promise-our-own-artisanal-internet/
[34] W. C. Tsai, A. Y. S. Chen, S. Y. Hsu, and R. H. Liang. 2015. CrescendoMessage: interacting with slow messaging. *Proceedings of the International Association of Societies of Design Research Conference* (2015), 2078–2095.
[35] Mark E. Whiting, Grant Hugh, and Michael S. Bernstein. 2019. Fair Work: Crowd Work Minimum Wage with One Line of Code. In *Seventh AAAI Conference on Human Computation and Crowdsourcing*.
[36] Wikipedia. 2019. Slow movement (culture). Retrieved April 3, 2019 from https://en.wikipedia.org/wiki/Slow_movement_(culture)
[37] C. M. Williams. 1973. System response time: A study of users' tolerance. *IBM Advanced Systems Development Division Tech. Rep. 17-272* (July 1973).
[38] Ming Yin, Jennifer Wortman Vaughan, and Hanna Wallach. 2019. Understanding the Effect of Accuracy on Trust in Machine Learning Models. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2019).

Received April 2019; revised June 2019; accepted August 2019