The Effects of Example-Based Explanations in a Machine Learning Interface

Carrie J. Cai
Google, Inc.
Mountain View, CA
cjcai@google.com

Jonas Jongejan
Google, Inc.
Mountain View, CA
jongejan@google.com

Jess Holbrook
Google, Inc.
Mountain View, CA
jessh@google.com

ABSTRACT

The black-box nature of machine learning algorithms can make their predictions difficult to understand and explain to end-users. In this paper, we propose and evaluate two kinds of example-based explanations in the visual domain, normative explanations and comparative explanations (Figure 1), which automatically surface examples from the training set of a deep neural net sketch-recognition algorithm. To investigate their effects, we deployed these explanations to 1150 users on QuickDraw, an online platform where users draw images and see whether a recognizer has correctly guessed the intended drawing. When the algorithm failed to recognize the drawing, those who received normative explanations felt they had a better understanding of the system, and perceived the system to have higher capability. However, comparative explanations did not always improve perceptions of the algorithm, possibly because they sometimes exposed limitations of the algorithm and may have led to surprise. These findings suggest that examples can serve as a vehicle for explaining algorithmic behavior, but point to relative advantages and disadvantages of using different kinds of examples, depending on the goal.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**;

KEYWORDS

Human-AI interaction; explainable AI; machine learning; example-based explanations

ACM Reference Format:
Carrie J. Cai, Jonas Jongejan, and Jess Holbrook. 2019. The Effects of Example-Based Explanations in a Machine Learning Interface. In *24th International Conference on Intelligent User Interfaces (IUI '19), March 17–20, 2019, Marina del Rey, CA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3301275.3302289

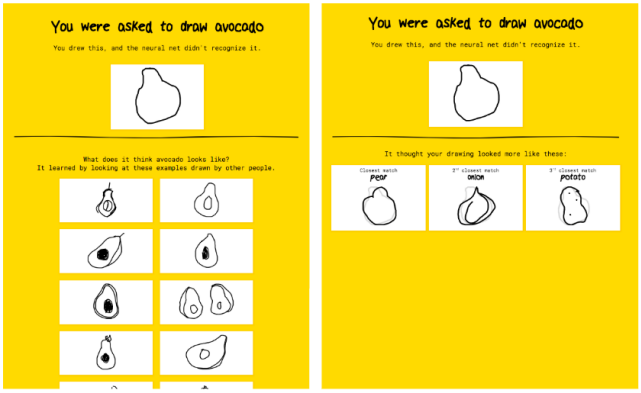

Figure 1: Example-based explanations automatically surface examples from the training set to explain algorithmic behavior. *Normative explanations* (left) establish a norm for the target class by showing training examples from that class (e.g. avocado). *Comparative explanations* (right) show the most similar training example from each of the hypothesized classes (e.g. pear, onion, potato), overlaid on top of the user's input so that users can compare their own drawing to those most-similar examples.

1 INTRODUCTION

Machine learning (ML) is increasingly being used in everyday applications, from social media feeds and educational apps, to games and creativity support tools. Despite the powerful capabilities of machine learning, the growing complexity of these algorithms has made them difficult to explain to end-users. This inscrutability can lead to a sub-optimal user experience, particularly when algorithms make inexplicable errors.

To address these problems, growing work in "Explainable AI" aims to make opaque algorithms more understandable. Algorithmic explanations have been found to improve user understanding of the system, as well as increase user satisfaction and trust [13, 15, 17]. These explanations can range from high-level descriptions of how the algorithm works, to low-level explanations of the specific factors influencing an algorithm's decision. Recently, the use of increasingly complex algorithms (e.g. deep neural networks) has made them more difficult to explain. In many cases, the inner workings of an algorithm may be inexplicable not only to users, but even to the developers of those systems [5]. The size, complexity, and opacity of ML algorithms have led to increasing regulations and demands for more explainable AI [1, 9].

In this paper, we investigate how **example-based explanations** can be used to explain algorithmic results using visual examples. Rather than explicitly explaining how an algorithm produced a given output, this approach instead surfaces examples from the training set, leaning on users' interpretation of those examples.

IUI '19, March 17–20, 2019, Marina del Rey, CA, USA

ACM ISBN 978-1-4503-6272-6/19/03.
https://doi.org/10.1145/3301275.3302289

Because examples are commonly used in explanations between humans, and can be effective for explaining complex concepts [25, 26], example-based AI explanations could potentially help end-users gain some intuition for algorithms that are otherwise difficult to explain through first principles.

We explore two kinds of example-based explanations in the sketch recognition domain. Given an item the user has sketched (e.g. an avocado) and the recognition result (recognized or not), **normative explanations** aim to establish a norm for what drawings look like in the intended class by show training examples from that class (e.g. training examples labeled "avocado", Figure 1 left). **Comparative explanations** show comparisons between the user's sketch and the most similar examples from distractor classes (e.g. pear, onion, etc. as shown in Figure 1 right). By allowing users to see *normative* examples for the intended class and as well as *comparative* examples in alternative classes, example-based explanations could potentially help users infer why the algorithm behaved the way it did and create useful mental models of the system.

To evaluate their effects, we deployed these example-based explanations on QuickDraw[1], an online platform that challenges players to draw pictures of everyday objects, using a deep neural network (DNN) algorithm to guess what the drawings represent. We chose QuickDraw as a platform because it has attracted millions of users worldwide, and thus serves as a natural environment capturing a broad range of perspectives.

Based on results from 1150 users, we found that when the algorithm failed to recognize the user's drawing, those who received normative explanations felt they had a better system understanding, and perceived the system to have higher capability. However, comparative explanations did not always improve perceptions of the algorithm, because they sometimes exposed the limitations of the algorithm and may have led to surprise. Users also spent a longer time viewing explanations when they received comparative explanations, and when their drawings were not recognized. These results suggest that examples are a potentially useful explanatory mechanism, but point to relative advantages and disadvantages of using different kinds of examples, depending on the goal.

2 RELATED WORK

A wide range of research has proposed ways to explain AI systems to users, in domains ranging from context-aware systems [17] and recommenders [10, 22], to social media [23] and advertisements [8]. For example, Lim and Dey explored different types of information that can be presented, such as the model's input features, the decision making process, the reasons underlying a decision, and the agent's certainty [16]. Prior work shows that explanations can help users develop better mental models of intelligent systems [17], and can increase user satisfaction, perception of control, and trust [13, 15]. However, some found that too much explanation can create confusion and degrade trust, and thus argue that AI explanations should only surface the most relevant information [13, 16, 29]. According to psychological research, humans prefer explanations that are simpler, more probable, and more causally relevant, all else being equal [4, 24].

[1]http://quickdraw.withgoogle.com

While simple ML algorithms can be inspected directly if they contain a limited number of components (e.g. decision trees, linear models), others are much more complex, making them challenging to explain (e.g. deep neural networks). For the latter, a variety of explanatory techniques have been proposed in machine learning communities [6, 12, 14, 27, 32], ranging from learning a simpler model on the predictions of the complex model, to perturbing inputs and seeing how the model reacts. Interactive mechanisms can also indirectly help users build mental models of algorithmic systems [3, 7]. Prior work in psychology and education suggests that examples can be an effective way to explain complex concepts [25, 26]. Examples can potentially help users develop intuition for the reasons underlying machine predictions, without directly exposing the internal logic of the algorithm [19, 28, 33]. Our work contributes to this broader literature by investigating the effects of two kinds of examples that are surfaced from the training set.

3 BACKGROUND

Our example-based explanations augment typical sketch recognition interfaces by explaining why a classifier did or did not recognize the user's input. For our implementation, we used the QuickDraw platform because it has a broad user base and is similar to common sketch recognition games like Pictionary.

On QuickDraw, users are given a word to draw, typically a common object. Users have a limited time to draw the object, during which the system attempts to guess what the drawing represents. The round ends when the system successfully guesses the intended word, or when 20 seconds have passed.

QuickDraw classifies drawings using a recurrent deep neural network trained on labeled sketches. The neural net receives strokes from the sketch one sample point at a time, where each sample indicates the change in coordinates since the last point, the time, and whether the point is the start of a new stroke.

4 EXAMPLE-BASED EXPLANATIONS

To give users insight into why the system recognized or did not recognize their drawing, we created example-based explanations to be shown once users have received the recognition result. Prior work in psychology suggests that humans prefer explanations which compare what's observed to *natural alternatives* [21]. For example, if a drawing looks like an avocado, a convincing explanation would not only explain why the drawing looks like "avocado," but also why it does not look like other similar objects ("pear", "onion", etc.). Likewise, if the drawing does *not* look like an avocado, an explanation might explain how it looks like other objects (e.g. "pear"), as well as how it looks different from a typical avocado. Given this dual nature of explanations, we designed two kinds of example-based explanations, capturing examples from both the target class and from alternative classes.

4.1 Normative Explanations

Normative explanations aim to establish a norm for what drawings look like in the target class, by displaying training examples from that class. For example, if the user was asked to draw "avocado," a normative explanation would display examples of "avocado" from the training set. While a range of methods could be used, we opted

for the simple procedure of displaying a random set of 30 training examples, to give the user a general sense of what the algorithm has seen. Directly above the set of examples, the interface displays this message: *"What does it think [object] looks like? It learned by looking at these examples drawn by other people."*

4.2 Comparative Explanations

Comparative explanations show a comparison between the user's drawing and similar drawings from alternative classes. Given the user's drawing (e.g. avocado), the system identifies the 3-best classification hypotheses for that drawing (e.g. pear, onion, potato). Then, for each hypothesized class, the system displays the most similar training example from that class, overlaid on top of the user's drawing so that the user sees the visual similarities and differences between the two drawings.

Those most-similar drawings are found by pre-computing embeddings of the training examples. In QuickDraw, the embeddings are feature vector representations of the drawings that are thousands of units long [11]. At run time, the system computes the embedding for the user's drawing, then finds the nearest neighbor to the user's drawing via euclidean distance. Above these examples, the interface displays the message *"It thought your drawing looked more like these"* if it was not recognized, and *"It also thought your drawing looked like these"* if the drawing was recognized. In the case where the drawing is recognized, the top hypothesis is the target class.

5 METHOD

To evaluate the effects of example-based explanations on user experience, we deployed these explanations on the QuickDraw platform. After drawing the object, users saw one of the following conditions: In the **no explanation** condition, users simply saw their drawing along with whether or not their drawing was recognized (these were shown to all users regardless of condition). In the **normative** explanation and **comparative** explanation conditions, users additionally saw an explanation of that type, as described in the section above. In the **normative and comparative** condition, users saw both kinds of explanations on the same page. These conditions were randomly assigned for each user (a between-subjects study), and all explanations were automatically generated on-the-fly.

To control for variability due to learning effects, all conditions were deployed in the first round of the game. For external validity, and to preserve a natural user experience, we did not artificially manipulate whether or not the drawing was recognized, but rather allowed the drawings to be naturally recognized or not recognized by the ML. So that these conditions would be balanced, we selected words that historically had a relatively balanced recognition rate (close to 50%). Since drawings and explanations could fluctuate substantially depending on the word, we selected three words with balanced recognition rates, spanning different object categories: food (avocado), household object (paintbrush), and body part (toe).

6 MEASURES

Following the explanation page, users saw a survey of four 7-point Likert scale questions, which they could optionally complete. Given prior evidence that AI explanations can affect understanding and trust, the questions assessed system **understanding** ("I understand what the system is thinking"), as well as **capability** and **benevolence**, key dimensions of trust that are widely used in trust questionnaires [20] ("The system seems capable", "The system seems benevolent"). As explanations can affect how users attribute blame and credit [18, 30], we included a question on **attribution** of the recognition result (1="Totally due to the recognizability of my drawing", 7="Totally due to the system's level of capability"). Finally, since different explanations can take different amounts of time to mentally process, we logged **time** spent on the explanation page.

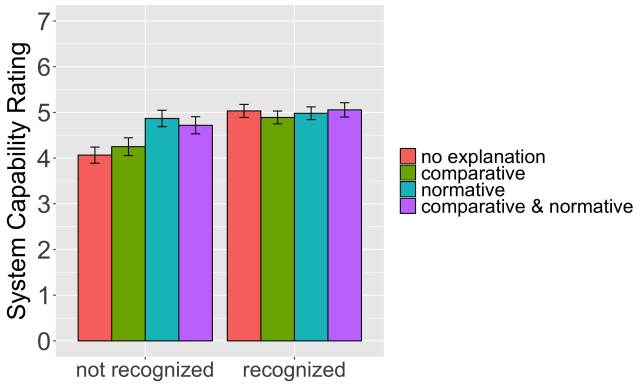

Figure 2: When drawings were not recognized, users who had seen explanations containing a *normative* component (normative-only and normative and comparative conditions) felt the system had higher capability (7-point Likert scale). Error bars show standard error.

7 HYPOTHESES

Because example-based explanations aim to increase transparency about why system errors occurred, we expect both normative and comparative explanations to improve user perceptions of the algorithm (understanding, capability, benevolence, and attribution of credit) in cases where their drawings are not recognized. In addition, we expect that people will have better perceptions of the algorithm when their drawings are recognized than when they're not. Due to a violation of expectations, we also expect users to view explanations for a longer time when their drawing is not recognized.

8 RESULTS

Overall, 1150 participants completed the study. 1% of results recorded only partial data due to technical issues, and were thus removed. In addition, using standard outlier removal methods (1.5 x interquartile range), we excluded those who had the explanation page open for an extremely long time before completing the survey (6% of participants), indicating that they may have temporarily left the task. All analyses were conducted on the remaining 1070 participants.

When the algorithm failed to recognize the drawing, users who received normative explanations felt they had a better **understanding** of the system. A three-way ANOVA (*Recognized* (drawing was

recognized vs. not) x *Normative* (presence of normative explanation) x *Comparative* (presence of comparative explanation)) found an interaction effect between Recognized and Normative ($p = 0.01$, $F = 6.7$). When the drawing was not recognized, users whose explanations contained a normative component rated their understanding of the system higher ($\mu = 4.6$, $\sigma = 2.0$) than those who did not see normative explanations ($\mu = 4.1$, $\sigma = 2.1$, $p = 0.008$). When the drawing was correctly recognized, no differences were found between explanation conditions. Additionally, there was an overall main effect of Recognized: as expected, those whose drawings were recognized felt they understood the system better ($\mu = 4.9$ vs. $\mu = 4.3$, $p < 0.0001$).

Likewise, when the algorithm did not recognize the drawing, users also rated system **capability** higher if they had seen a normative explanation (Figure 2). A three-way ANOVA, with capability as the outcome variable, found a significant interaction effect between Recognized and Normative ($p = 0.01$, $F = 6.4$). When the drawing was not recognized, users whose explanations contained a normative component rated system capability higher ($\mu = 4.8$, $\sigma = 1.9$ vs. $\mu = 4.2$, $\sigma = 2.0$, $p = 0.0006$). Additionally, there was an overall main effect of Recognized. As expected, when drawings were recognized, users felt the system was more capable ($\mu = 5.0$) than when drawings were not recognized ($\mu = 4.5$, $p < 0.0001$).

In addition, we found a small but significant main effect of comparative explanations on system **benevolence**. Users who saw comparative explanations perceived the system to be more benevolent ($\mu = 5.1$ vs. $\mu = 4.8$, $p = 0.035$). Though comparative explanations did not increase perceptions of system capability, it's possible users may feel that the algorithm is at least attempting to make an effort. However, given the small effect size, the practical significance of this may be limited. Additionally, users found the system more benevolent when drawings were recognized ($\mu = 5.1$ vs. $\mu = 4.7$, $p = 0.0007$), as expected. No significant differences were found in the effect of explanation type on **attribution** of blame or credit, perhaps related to the low risk nature of QuickDraw.

Finally, users spent more **time** on the explanations page when drawings were not recognized ($\mu = 9.9$ sec, $\sigma = 5.4$) than when they were ($\mu = 9.1$, $\sigma = 6.0$, $p = 0.002$). As timing data is not normally distributed, we applied a log-transformation before running significance tests. Explanations containing a comparative component were also viewed for longer than normative-only explanations ($\mu = 9.7$ vs. $\mu = 8.8$, $p = 0.04$).

Contrary to our hypotheses, comparative explanations did not improve perceptions of system capability. Upon further investigation, we found that even though the shown drawings aimed to be visually similar to the user's drawing, they were often semantically unrelated, e.g. word="toe", hypotheses= "hula hoop", "donut" (Figure 3). These surprises may have exposed algorithmic limitations, and may explain why users spent longer viewing comparative explanations even though normative explanations displayed more pictures. Moreover, comparative explanations involved comparison across multiple classes, whereas normative explanations showed pictures all from the same class. Future work could investigate the trade-offs between explanatory power and cognitive workload.

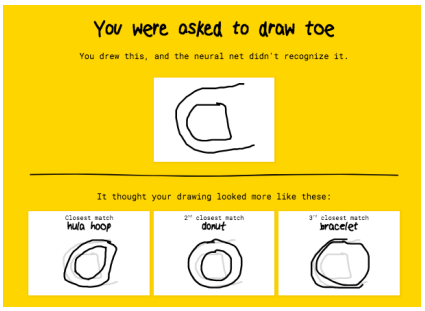

Figure 3: Comparative explanations sometimes showed surprising hypotheses that may have exposed limitations of the algorithm.

9 DISCUSSION

Our findings suggest that examples are a feasible vehicle for explaining algorithmic behavior, but that different kinds of explanations may have relative advantages and disadvantages. On the one hand, normative explanations allowed the system to demonstrate partial capability in cases where it appeared to have failed. Because normative explanations display a range of examples produced by other people, these may be particularly compelling when social norming effects are strong [31], or when there are common representations of the target class that the user had not considered. On the other hand, comparative explanations sometimes revealed algorithmic limitations, and may have contributed to confusion or surprise. These results lend credence to prior indications that convincing explanations should *in turn* be explainable [24].

Hence, different explanations could be leveraged depending on the goal. For instance, if users are known to under-trust a system despite high system capability, normative explanations could help improve user perceptions during system errors. Alternatively, if users tend to over-trust the system, comparative explanations could help establish a more appropriate level of trust. Comparative explanations might also help increase perceived benevolence. Future work could investigate how best to show examples over time, so that users build an accurate mental model and do not over-trust or under-trust the system.

One limitation of this study is that the system knew the user intent a priori, and could show examples related to that intent (the target word). These scenarios are common in contexts where the system provides a prompt for a user activity (e.g. educational exercises, games, etc.) In ML-aided learning contexts [2], for instance, seeing example-based explanations after completing an exercise may help users understand the result, or determine the extent to which to trust an imperfect agent. Although the explanations used in our study can be most readily applied to cases where user intent is known, in cases where the intent is *not* known, explanations could be shown for multiple possible intents, or the intent could be indicated by a user when requesting an explanation. While intent elicitation was outside the scope of this study, we view this as a particularly important direction for future work.

Although we focused on sketch recognition as a test bed for explanations, the general methods used for generating explanations (fetching training examples, finding similar examples) could be applied and tested in other contexts. For example, the relative strength of normative explanations may vary depending on the dynamics of normative influence in particular domains. In some domains, designers may also need to take extra precautions when surfacing training data, such as curating a subset to exclude examples that could be potentially sensitive or offensive. In addition, a limitation of the current work is its focus on self-report outcomes and lack of qualitative interviews. While we chose a natural, online environment for the purpose of collecting organic human-AI interactions, future work could capture deeper user insights and measure behavioral outcomes, such as whether drawings change after users have seen explanations.

Our findings suggest new ways for improving algorithmic transparency, particularly when an algorithm is too complex to explain explicitly, or when the feature set is too large to enumerate. While much work has to date focused on explicitly generating explanations using machine intelligence, example-based explanations implicitly allow humans to play a more active role in interpreting machine behavior, by leveraging their own heuristics, prior experience, and human intelligence to infer why an algorithm produced a given output. Integrating example-based approaches with existing efforts is a promising direction for future research, as it opens up possibilities for leveraging the intelligence of human beings themselves.

ACKNOWLEDGMENTS

We thank Martin Wattenberg, Fernanda Viegas, Emily Reif, Henry Rowley, Dan Russell, D. Sculley, Aaron Sedley, and Michael Terry for their valuable feedback on this work.

REFERENCES

[1] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y Lim, and Mohan Kankanhalli. 2018. 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*. ACM, 582.

[2] Carrie J Cai. 2013. Adapting arcade games for learning. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2665–2670.

[3] Carrie J Cai, Emily Reif, Narayan Hegde, Jason Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda Viegas, Greg S Corrado, Martin C Stumpe, and Michael Terry. 2019. Human-Centered Tools for Coping with Imperfect Algorithms during Medical Decision-Making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM.

[4] Matteo Colombo, Leandra Bucher, and Jan Sprenger. 2017. Determinants of judgments of explanatory power: Credibility, Generality, and Statistical Relevance. *Frontiers in psychology* 8 (2017), 1430.

[5] Malin Eiband, Hanna Schneider, Mark Bilandzic, Julian Fazekas-Con, Mareike Haug, and Heinrich Hussmann. 2018. Bringing Transparency Design into Practice. In *23rd International Conference on Intelligent User Interfaces*. ACM, 211–223.

[6] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2009. Visualizing higher-layer features of a deep network. *University of Montreal* 1341, 3 (2009), 1.

[7] Motahhare Eslami, Karrie Karahalios, Christian Sandvig, Kristen Vaccaro, Aimee Rickman, Kevin Hamilton, and Alex Kirlik. 2016. 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*. ACM, 2371–2382.

[8] Motahhare Eslami, Sneha R Krishna Kumaran, Christian Sandvig, and Karrie Karahalios. 2018. Communicating Algorithmic Process in Online Behavioral Advertising. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 432.

[9] Bryce Goodman and Seth Flaxman. 2016. European Union regulations on algorithmic decision-making and a" right to explanation". *arXiv preprint arXiv:1606.08813* (2016).

[10] Jonathan L Herlocker, Joseph A Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. ACM, 241–250.

[11] Daniel Keysers, Thomas Deselaers, Henry A Rowley, Li-Lun Wang, and Victor Carbune. 2017. Multi-Language Online Handwriting Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 6 (2017), 1180–1194.

[12] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*. 2673–2682.

[13] René F Kizilcec. 2016. How much information?: Effects of transparency on trust in an algorithmic interface. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 2390–2395.

[14] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730* (2017).

[15] Todd Kulesza, Simone Stumpf, Margaret Burnett, and Irwin Kwan. 2012. Tell me more?: the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1–10.

[16] Brian Y Lim and Anind K Dey. 2009. Assessing demand for intelligibility in context-aware applications. In *Proceedings of the 11th international conference on Ubiquitous computing*. ACM, 195–204.

[17] Brian Y Lim, Anind K Dey, and Daniel Avrahami. 2009. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2119–2128.

[18] Bertram F Malle. 2011. Attribution theories: How people make sense of behavior. *Theories in social psychology* 23 (2011), 72–95.

[19] David Martens and Foster Provost. 2013. Explaining data-driven document classifications. (2013).

[20] Roger C Mayer, James H Davis, and F David Schoorman. 1995. An integrative model of organizational trust. *Academy of management review* 20, 3 (1995), 709–734.

[21] Tim Miller. 2017. Explanation in artificial intelligence: insights from the social sciences. *arXiv preprint arXiv:1706.07269* (2017).

[22] Pearl Pu and Li Chen. 2006. Trust building with explanation interfaces. In *Proceedings of the 11th international conference on Intelligent user interfaces*. ACM, 93–100.

[23] Emilee Rader, Kelley Cotter, and Janghee Cho. 2018. Explanations as Mechanisms for Supporting Algorithmic Transparency. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 103.

[24] Stephen J Read and Amy Marcus-Newhall. 1993. Explanatory coherence in social explanations: A parallel distributed processing account. *Journal of Personality and Social Psychology* 65, 3 (1993), 429.

[25] Alexander Renkl. 2014. Toward an instructionally oriented theory of example-based learning. *Cognitive science* 38, 1 (2014), 1–37.

[26] Alexander Renkl, Tatjana Hilbert, and Silke Schworm. 2009. Example-based learning in heuristic domains: A cognitive load theory account. *Educational Psychology Review* 21, 1 (2009), 67–78.

[27] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 1135–1144.

[28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-precision model-agnostic explanations. In *AAAI Conference on Artificial Intelligence*.

[29] James Schaffer, Prasanna Giridhar, Debra Jones, Tobias Höllerer, Tarek Abdelzaher, and John O'Donovan. 2015. Getting the message?: A study of explanation interfaces for microblog data analysis. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM, 345–356.

[30] Kelly G Shaver. 2012. *The attribution of blame: Causality, responsibility, and blameworthiness*. Springer Science & Business Media.

[31] Muzafer Sherif. 1936. The psychology of social norms. (1936).

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

[33] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. (2017).