2019 16th International Conference on Ubiquitous Robots (UR)
Jeju, Korea, June 24-27, 2019

Personalizing the Prediction: Interactive and Interpretable machine learning

Seunghun Koh, Hee Ju Wi, Byung Hyung Kim and Sungho Jo

Abstract— **While many applications with machine learning provide enough utilities for users, they mostly target average of users. Although it might be acceptable in certain domains, there are domains such as health and medical-care where it is crucial to provide personalized service. In such cases, personalization of machine learning model usually does not depend on end users to make change to the system. As machine learning models are black-box, the only information that the users can acquire is input and output of certain decision made by the model. Thus, with no reason behind specific prediction provided by the system, users cannot understand how the system works and make amendments to the system. This shortcoming is directly related to users' credibility in the system. In this paper, we present an interface where the system provides users the reason behind the decision made by the machine learning model and users provide feedback to the model. Moreover, we present the principle behind the suggested interface and prototype that instantiates the suggested interface. Our interface's effectiveness is evaluated through users' surveys regarding two main attributes: (1) how well users understand the system and more importantly, (2) how it influences users to trust in the system.**

I. INTRODUCTION

These days, machine learning is the central concept in many advances in science and technology. With development of machine learning model to make it more accurate in its decision-making process, machine learning is favored by many service providers and used in countless applications [1]. However, machine learning models used in those applications are mostly generic - that is, they focus on providing enough utilities for as many people as possible. Consequentially, they do not provide the best service for individual users. While such tendency can be acceptable in many domains, when it comes to areas such as health, medical-care and recommendation system, where the system should be able to provide suitable services for individual users, failure of the system in making right decision for each user is critical in the system's credibility.

Due to such reasons, along with development of machine learning, personalizing the machine learning model became an important issue. Service providing companies such as Netflix (video-content recommendation service) and Spotify (music recommendation service) are already adopting machine learning to provide personalized service for their users.

This work was supported in part by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT)(No. 2017-0-00432) and in part by the Technology Innovation Program (1 0070171) funded by the Ministry of Trade, Industry & Energy (MI, Korea).

Seunghun Koh is with the School of Computing, KAIST, Republic of Korea. S. Jo is the corresponding author. E-mail: bhyung, shjo@kaist.ac.kr

However, many of such applications do not depend heavily much on users' input to personalize the system. In many cases, systems achieve personalization by acquiring more information from users and developing more complex model to process acquired information to produce more personalized result for each user. In the process of personalizing the system, users can only provide limited inputs to the model. For example, in many recommendation systems, types of input from the user to the system are limited to 1) user's information, 2) user's past record of recommended contents and 3) whether the user likes or dislikes the recommended contents. For systems where personalization is a critical issue, if they fail to provide personalized prediction and users have limited role in personalizing the system, it results in users' loss of credibility in the system.

Despite such shortcoming, many systems do not ask users for their active participation in process of personalization because they do not know what to show to and receive from users. Despite its popularity, machine learning model mostly remains as a black box model [2]. Thus, things that the system can show to users are inputs to the machine learning model and produced output. As it is hard to find out what inputs affects the model's result in what ways, only type of feedback the system can get from the users is whether they think the result from the model is appropriate or not.

In this paper, we propose new type of interface between the machine learning model and users that: 1) analyzes reasons behind the model's behavior based on the features given as inputs and 2) enables users to personalize the model by providing feedback on such reasons. To analyze machine learning model's behavior, explainable AI should be used. Explainable AI, in shorten XAI, is an artificial intelligence whose actions can be trusted and easily understood by humans. Currently, many studies are conducted regarding XAI, mainly to explain machine learning model's decision-making process [3]. In particular, our interface uses LIME, an open source XAI module which analyzes model's behavior to certain input by creating a local linear approximation model around the given input and analyzing linear approximation model's behavior [2]. Based on result provided by LIME, users will be able to change the weight of each input features to personalize the machine learning model.

978-1-7281-3232-7/19/$31.00 ©2019 IEEE

II. RELATED WORKS

A. Explainable and interpretable machine learning

Machine learning models are able to produce more accurate and reliable predictions but with the cost of being more complex and harder to interpret [4]. Such inverse relation between interpretability and delicacy of machine learning models led to new field of research that aims to improve the interpretability of machine learning model. With DARPA's initiative to support XAI, there has been increase in urge to develop interpretable and explainable machine learning model [5]. Various AI and ML communities have been very supportive to such tendencies, leading to various workshops organized by those communities [3] and numerous mathematical algorithms to explain inner working of machine learning model [6]. Response from HCI communities have been similar to that of AI and ML communities. Since late 90s and early 2000s, researchers agreed that users needed to be able to understand what was perceived by the system and what actions the system takes based on that perception [3]. Researchers in field of HCI worked on interfaces that provides textual [7] [8] [9] and visual [10] explanation for underlying context-aware rules and so many other streams of research to make interfaces intelligible.

B. User's role in machine learning

Ever since Fails et al. introduced the term interactive machine learning in their paper and showed that when users can train, classify and correct the classifications, they could quickly fix the errors made by the machine learning model [11], numerous studies were conducted to show that end users interactively controlling the system affects the machine learning model. For instance, Bryan et al. trained instance-based classifiers using end users' interaction with the system [12]. However, most of such studies treated the machine learning system as a black box. Users control the system by providing different inputs and receiving corresponding outputs, but are unable to know what features of different inputs caused such change in outputs.

Recent studies show that if end users better understand how a machine learning model operates, they are better able to interact with it. For example, Fiebrink et al. and Kulesza et al. suggest that with better knowledge of how a machine learning model operates, end users can better personalize the system [13] [14]. Furthermore, Ko et al. and Rugaber et al. suggest that when end users are given with explanations of machine learning model's behavior, they are better able to debug the system [15] [16]. Increased transparency also contributed to users' increased trust in the system's predictions [17] [18]. Our new interface tries to combine these previous results where the interface provides explanations of machine learning model's prediction to users so that users can better personalize the system and also by showing that the system successfully applied users' feedback, the interface gains trust from its users.

III. METHODOLOGY

For prototype of our new interface, we instantiated a movie recommendation system, X-MoRe. In X-MoRe, users can communicate with the system through web-application to provide their feedback to server, which, in turn, will use those feedback to provide more personalized service.

X-MoRe is mainly divided into three parts: (1) movie clusterer that creates movie distributions where similar movies stay close, (2) movie selector that recommends movies based on user's previous records and analyzes reason why the system recommended such movies and (3) graphic user interface (web-application) that the user uses to interact with the system.

A. Preparing dataset

For X-MoRe, we used Movielens dataset containing around 20 million movies from 1880s to 2010s [19]. Initially, Movielens dataset contains each movie's Movielens ID, title of movie, genres and tag information, which is a word or phrase that users who watched the movie used to describe the movie. As X-MoRe tries to receive users' feedback on various fields, we processed the Movielens dataset additionally to make them contain names of director and up to three actors. Additionally, movies' ratings are also crawled from IMDB website. Movies that (1) were not on IMDB website, (2) director's name does not exist in IMDB website, (3) actors' names do not exist in IMDB website or (4) rating doesn't exist in IMDB website were removed from our dataset. Tag data was also processed from movielens dataset, where three most frequently given tags are chosen. If tag data doesn't exist for the movie or there are less than 2 tags available, empty tag fields for such movie are filled with "NoTag". After preprocessing, in total 55846 movies existed in our dataset.

Each time a new user signs up, the user's personal dataset is created by copying entire existing movies. Later, user's feedback are applied to his or her private dataset.

B. Recommendation system of X-More

As shown in Fig. 1, there are two important modules in X-MoRe that are responsible in providing personalized recommendation: movie clusterer and movie selector. Initially, movie clusterer generates two movie distributions using two different features of movies: genre and tag. In both cases, simple model of autoencoder [20] is used, where the autoencoder is trained end-to-end and training uses ADAM learning algorithm [21]. In case of tag-based cluster, each tag is processed into 20-dimensional word vectors using twitter-based Gensim [22].

With the movie distributions created, movie selector executes a sequence of process to produce final movie recommendation and reasons for such recommendation. Initially, it searches the database to find last 20 movies that the user watched. With those movies, movie selector selects pool of movies containing movies that are close to 20 watched movies in two previously mentioned movie clusters. Moreover, movies with same directors and actors with those

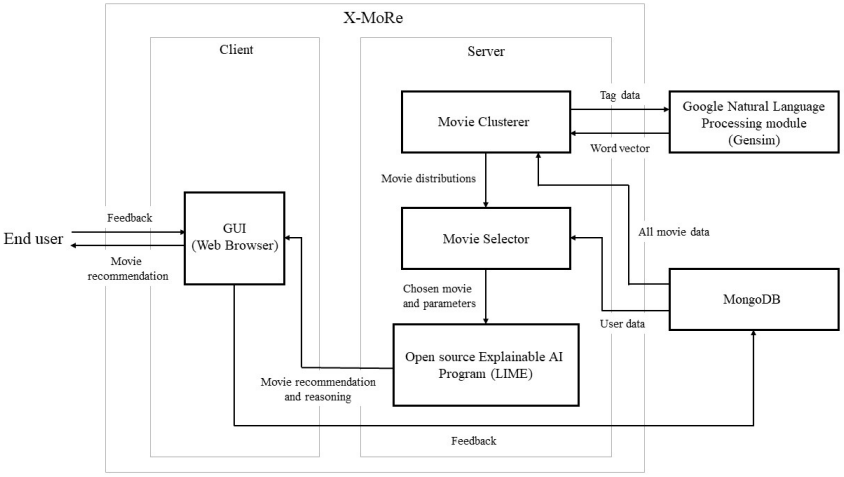


Fig. 1: Component diagram of X-MoRe

of 20 watched movies are also selected to be in the pool. Among the movies in the created pool, repeated movies are removed.

Then, the movie selector uses regression model to choose 20 movies that the user is most likely to enjoy. The regression model is trained with user's personal dataset each time the user wants new recommendation. Training uses information of movies (year of release, names of actors and directors, genres and tag data) as data and movies' ratings as labels for each movie. Due to memory shortage and property of some features (e.g. directors, actors), director, actor and tag information are encoded into categorical feature and genre information is processed as one-hot encoding. Accordingly, to consider both categorical and non-categorical data, decision-tree regression model is used with maximum depth of 40 and at least 3 data is required for a leaf to split. Regression model, with the data of movies in the movie pool, produces expected rating that the user is most likely to give to those movies. Out of them, 20 movies with highest expected ratings are chosen to be shown to the user.

Then, user specific data, along with data of 20 chosen movies are analyzed by LIME [2]. By creating linear approximation of user specific model around each instance of 20 chosen movies, LIME produces weights of each features related to analyzed movies. These weights add up to expected rating produced by the decision-tree regression model. Weights can be both positive and negative, where features with negative weights can be interpreted as features the user doesn't prefer. For user's convenience, weights are presented as percentage using following calculation:

$$p_j = \frac{|w_j|}{\sum_j |w_j|} \tag{1}$$

C. Web-based Graphical User Interface

Fig. 2 describes the composition of overall web application of X-More and the outline of user scenarios. When users log in, X-MoRe's initial webpage is shown to them as

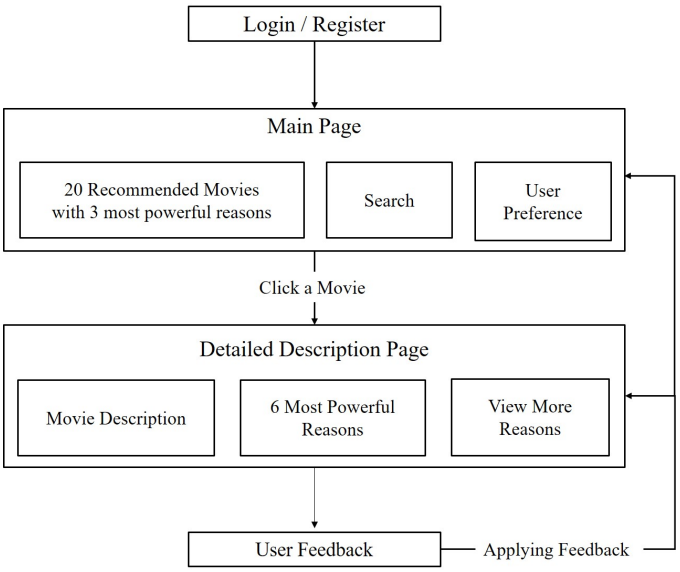
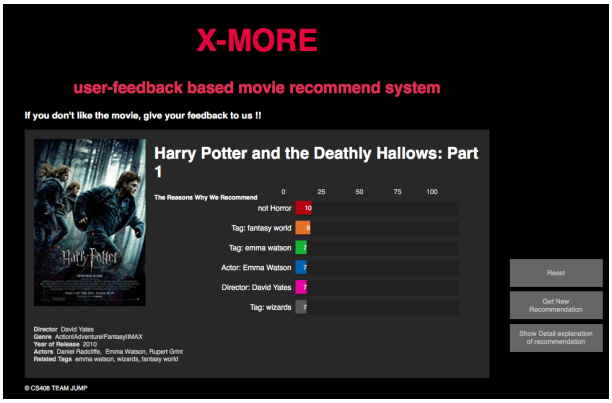


Fig. 2: Simple flow diagram of web-application in X-MoRe

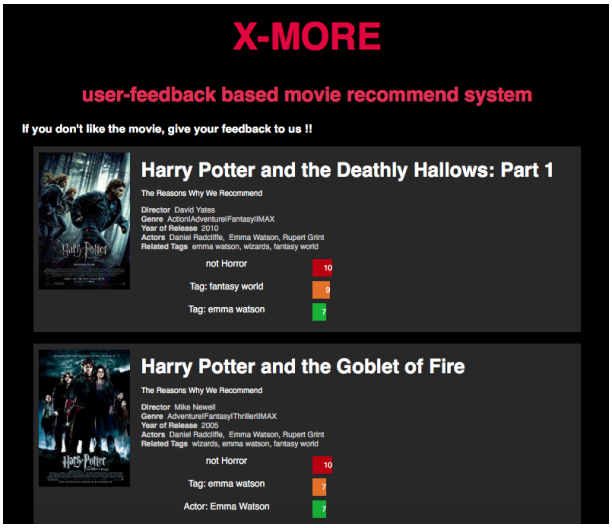
shown in Fig. 3b. In X-MoRe's initial webpage containing 20 recommended movies, we provide big posters and simple information of each movie so that users can receive movie information easily. Moreover, users can also see 3 most significant reasons for each movie with bar graphs to show their respective weights. This makes users to understand intuitively why our system recommended the movie.

When users click on the movie that they are interested in, X-MoRe provides 6 most critical reasons why certain movie is recommended with bar graphs and its figures on the right side of the name of each element as shown in Fig. 4. This is placed at the center of the page to make users understand physically and cognitively easily.

If users want to see a detailed explanation of the recommendation, X-MoRe provides weights of every input feature of the movie with the bar graphs and the figures indicating the percentage of each element occupies. Users can view both



(a) Detailed description of a movie. Users can give feedback in this webpage.



(b) Home page of web-application. Users can click on title of each movies to see detailed reasoning why X-MoRe recommended the movie.

Fig. 3: Graphical User Interface (web-application) of X-MoRe

positive and negative factors which gave positive and negative effects on the movie to be recommended, respectively. The elements are sorted in the descending order. The weights are rounded to nearest integers to make users understand the figures visually more easily.

After users provide the feedback (ways of giving feedback is explained in next section), they can simply click "Get New Recommendation" button to get newly feedback adapted recommendation list. The system saves users' feedback to each user's personal database, applies the feedback to each user's personal model and provides new recommendations. Then, users can view the main page that shows 20 new recommended movies and its reasons of recommendation, again.

Just like any other recommendation platform, X-MoRe has searching function. Moreover, users can see their overall preferences in features by clicking "USER PREFERENCE" button and reset their preferences by clicking "RESET" button.

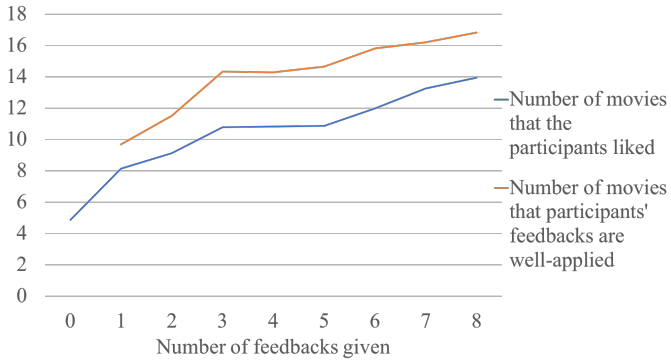


Fig. 4: Number of movies that the participants liked (blue) and number of movies that participants' feedback are well-applied (orange) after each time they provided feedback to the system.

D. Applying user feedback

X-MoRe has several types of feedback available for users to give and it reacts differently for each type of feedback. One type of feedback is when a user changes the weight of certain features. We use C as the change in user specific rating of the movie, B as weight in percentage of chosen feature before the user changes it, A as weight in percentage of chosen feature after the user changes it, r as user specific rating of the movie and c as the constant to regulate the system's behavior. C is calculated as below:

$$C = \frac{c \times r \times (A - B)}{100} \tag{2}$$

Value of c depends on which feature's weight the user changed. If the feature is related to genre, then c is 0.15 and otherwise, it is 0.3. Because genre occupies 20 out of 28 dimensions of input data, change in weight of genre changes the model much more significantly than same change in weight of any other features does. To regulate such behavior, value of c when feature is related to genre, is reduced to 0.15.

Then, value of C is added to user's personal rating of every movie with same feature. When the regression model is trained with updated user's personal data, the regression model identifies that the user changed the weight of specific feature and recommend accordingly with changed tendency. Second type of feedback is when the user notifies that he or she doesn't like specific feature of the movie. This type of feedback is treated similar to the first type of feedback, with value of A being 0 to lead to drastic effect of change in model.

Last type of feedback available is when the user notifies that he or she considers that a tag does not match with the movie or gives a new tag for the movie. In this case, the chosen tag is removed from user's private dataset. When the user gives a new tag, if there's already 3 tags related to the movie, it randomly removes one and add the given tag and if not, new tag is just added to the dataset.

IV. EXPERIMENTAL SETUP

A. Participants

We recruited 23 participants with various backgrounds and experiences in machine learning. As participants should compare existing recommendation system and that of X-MoRe in various dimensions, we only chose participants who had experience with recommendation platform to ensure detailed comparison is possible. Moreover, all participants had no prior knowledge of XAI or manipulation of machine learning model.

B. Experimental Procedure

At the beginning, participants were asked to choose one of the movies that they like, preferably one that they watched before, as they have to provide feedback on it and we cannot show them actual movies. With the chosen movie, each participant was asked to give any kind of feedback to the movie. Types of feedback one could provide included: (1) changing the weight of certain features, (2) declaring that one doesn't like certain features, (3) adding a new tag to the movie. With the given feedback, participants are given 20 new movie recommendations after their feedback are applied.

Every time participants were given with a list of movie recommendations, including the initial stage, they were asked to count number of movies that they like or are interested in and number of movies that reflects their previous feedback. Participants were to give at least 10 feedback to the system and answer those questions.

After 10 feedback were given by the participant, they were asked to fill in survey. Three questions are asked in survey, each related to usability, credibility and usefulness of new interface.

V. EXPERIMENTAL RESULT

A. User experiment

X-MoRe has three attributes to evaluate. Along with usability and credibility that our interface mainly focuses on, X-MoRe also has to achieve accuracy in its output because it is basically a recommendation system. There are two kinds of accuracies related to X-MoRe. First type is how accurate X-MoRe's recommendations are and second type is how accurately X-MoRe applied user's feedback. As two types of accuracy depends on user's perception, we asked participants to evaluate X-MoRe's accuracy by themselves.

Accuracy of X-MoRe's recommendation is evaluated as number of movies that each participant liked or is interested in out of 20 recommended movies. Although there was discrepancy between participants, in general, number of movies that the participants liked increased as more feedback were given and it reached around 14 by the end of experiment.

How accurately X-MoRe applied user's feedback is evaluated by counting number of movies that the participants thought their feedback is well-applied. Again, in general, number of such movies tended to increase as more feedback were given and it reached around 17 by the end of experiment. These two results suggest that X-MoRe successfully analyzed participants' tendency in movie with feedback given by the participants and recommended movies well following analyzed tendency.

Evaluation factor	**Response (number of participants)**				
	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
Usability	0	0	6	15	2
Credibility	0	1	9	11	2
Usefulness	0	0	5	14	4

TABLE I: Result of survey. Usability, credibility and usefulness each corresponds to question (1), (2) and (3) stated above, respectively.

Although there has been improvement in accuracy, accuracy measured by some participants either did not improve or improved only little, compared to initial accuracy. Such results could have been caused by nature of X-MoRe. When recommending movies from 55846 movies in its database, X-MoRe, rather than placing priority on recent movies, recommends movies with higher expected ratings. Because initial rating is collected from IMDB website, movies that are not famous but with high rating can be recommended to users. Moreover, in current situation where X-MoRe cannot show actual movies to participants, it is possible that users might like the recommended movie but only with given information about the recommended movie, they can think in other way.

B. Survey

Similar with accuracy, evaluation of credibility depended on users' survey as it is very subjective criterion. Credibility of X-MoRe was evaluated through three questions: (1) Was X-MoRe's interface intuitive and easy to use, (2) Compared to other recommendation platforms, was X-MoRe more credible that it will provide more personalized recommendation, (3) Would it be useful if existing recommendation platforms adapt similar interface to that of X-MoRe?

Overall, participants showed satisfaction towards X-MoRe and furthermore, towards new type of interface provided by X-MoRe. As shown in Table I, 17 out of 23 participants replied that X-MoRe's weight changing system is intuitive. This suggests that participants well understood X-MoRe's system and machine learning system of X-MoRe, which provides a baseline of easy personalization.

As for credibility of the system, X-MoRe's method of personalization by receiving user's input was somewhat successful. 13 participants replied that X-MoRe gave more credibility than existing recommendation platforms. Most of these participants agreed that providing users with enough information and enabling them to actively personalize the system played a big part in gaining credibility. Some participants focused on other factors. One participant stated that X-More was more credible because of its high accuracy in

recommendation and another participant stated that X-MoRe was credible because he could see that X-MoRe well-applied his feedback about his tendency in movies.

It should also be noted that 10 participants replied that X-MoRe was not more credible than existing recommendation platforms. Their reasons were quite diverse. Most frequent reason was lack of information. One participant stated that "Reason why X-MoRe recommended certain movies did not include actual reason why I like those movies. For example, I like certain movie because it is touching but such reason did not show up at all." Such drawback of X-MoRe is mostly because X-MoRe uses pre-chosen information of movies (year, genres, directors and etc) to recommend the movie and LIME only analyzes reason of recommendation based on inputs of recommendation model.

It is also notable that most participants agreed that interface provided by X-MoRe is useful. As shown in Table I, 18 participants replied that if existing recommendation platforms adapt this new type of interface, it would be useful. Considering the fact that we had both participants who knew well about the machine learning and those who had no knowledge regarding the machine learning, this results shows that new interface can be accepted wide variety of users, enabling them to personalize the system while acquiring their credibility.

VI. CONCLUSION

In this paper, we demonstrated how our new interface, where the system provides users the reason behind its decision and users provide feedback to the system, can help users to personalize the machine learning model effectively and more importantly, how it can successfully acquire users' credibility in the system. As an example of such interface, we proposed X-MoRe, a movie recommendation system, as recommendation system is one of the domains where personalization and users' credibility are critical. X-MoRe enables users to personalize the system by creating each user's database and providing users with variety of ways of giving feedback to change their own database to meet their taste.

From user study conducted, X-MoRe successfully achieved its goals, which are: (1) to provide accurate movie recommendation service, (2) to enable active personalization by users, and (3) to acquire their credibility in the system. Our result shows that focusing on user's role in personalizing the system is important in making the system more sustainable and personalized.

REFERENCES

[1] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[2] M. T. Ribeiro, S. Singh, and C. 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*, KDD '16, (New York, NY, USA), pp. 1135–1144, ACM, 2016.

[3] A. Abdul, J. Vermeulen, D. Wang, B. Y. Lim, and M. 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, (New York, NY, USA), pp. 582:1–582:18, ACM, 2018.

[4] L. Breiman, "Statistical modeling: The two cultures (with comments and a rejoinder by the author)," *Statist. Sci.*, vol. 16, pp. 199–231, 08 2001.

[5] D. Gunning, "Explainable artificial intelligence (xai)," *Defense Advanced Research Projects Agency (DARPA), nd Web*, 2017.

[6] O. Biran and C. Cotton, "Explanation and justification in machine learning: A survey," in *IJCAI-17 workshop on explainable AI (XAI)*, vol. 8, 2017.

[7] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Building interpretable classifiers with rules using bayesian analysis," *Department of Statistics Technical Report tr609, University of Washington*, 2012.

[8] B. Y. Lim and A. K. Dey, "Toolkit to support intelligibility in context-aware applications," in *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, UbiComp '10, (New York, NY, USA), pp. 13–22, ACM, 2010.

[9] J. Vermeulen, G. Vanderhulst, K. Luyten, and K. Coninx, "Answering why and why not questions in ubiquitous computing," 2009.

[10] J. Vermeulen, J. Slenders, K. Luyten, and K. Coninx, "I bet you look good on the wall: Making the invisible computer visible," in *Ambient Intelligence* (M. Tscheligi, B. de Ruyter, P. Markopoulus, R. Wichert, T. Mirlacher, A. Meschterjakov, and W. Reitberger, eds.), (Berlin, Heidelberg), pp. 196–205, Springer Berlin Heidelberg, 2009.

[11] J. A. Fails and D. R. Olsen, Jr., "Interactive machine learning," in *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, (New York, NY, USA), pp. 39–45, ACM, 2003.

[12] N. J. Bryan, G. J. Mysore, and G. Wang, "Isse: An interactive source separation editor," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, (New York, NY, USA), pp. 257–266, ACM, 2014.

[13] R. Fiebrink, P. R. Cook, and D. Trueman, "Human model evaluation in interactive supervised learning," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, (New York, NY, USA), pp. 147–156, ACM, 2011.

[14] T. Kulesza, S. Stumpf, M. Burnett, and I. Kwan, "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*, CHI '12, (New York, NY, USA), pp. 1–10, ACM, 2012.

[15] A. Ko and B. Myers, "Debugging reinvented," in *2008 ACM/IEEE 30th International Conference on Software Engineering*, pp. 301–310, May 2008.

[16] S. Rugaber, A. K. Goel, and L. Martie, "Gaia: A cad environment for model-based adaptation of game-playing software agents," *Procedia Computer Science*, vol. 16, pp. 29–38, 2013.

[17] M. T. Dzindolet, S. A. Peterson, R. A. Pomranky, L. G. Pierce, and H. P. Beck, "The role of trust in automation reliance," *International journal of human-computer studies*, vol. 58, no. 6, pp. 697–718, 2003.

[18] B. H. Kim and S. Jo, "Deep physiological affect network for the recognition of human emotions," *IEEE Transactions on Affective Computing*, 2018.

[19] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, pp. 19:1–19:19, Dec. 2015.

[20] P. Baldi, "Autoencoders, unsupervised learning and deep architectures," in *Proceedings of the 2011 International Conference on Unsupervised and Transfer Learning Workshop - Volume 27*, UTLW'11, pp. 37–50, JMLR.org, 2011.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *IN PROCEEDINGS OF THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS*, pp. 45–50, 2010.