RESEARCH-ARTICLE

Predicting and Presenting Task Difficulty for Crowdsourcing Food Rescue Platforms

ZHEYUAN RYAN SHI, University of Pittsburgh, Pittsburgh, PA, United States

JIAYIN ZHI, Carnegie Mellon University, Pittsburgh, PA, United States

SIQI ZENG, University of Illinois Urbana-Champaign, Urbana, IL, United States

ZHICHENG ZHANG, Carnegie Mellon University, Pittsburgh, PA, United States

AMEESH KAPOOR

SEAN HUDSON

View all

Published: 13 May 2024

WWW '24: The ACM Web Conference 2024
May 13 - 17, 2024
Singapore, Singapore

Conference Sponsors:
SIGWEB

Predicting and Presenting Task Difficulty for Crowdsourcing Food Rescue Platforms

Zheyuan Ryan Shi
ryanshi@pitt.edu
University of Pittsburgh
Pittsburgh, PA, USA

Jiayin Zhi
jzhi@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

Siqi Zeng
siqi6@illinois.edu
University of Illinois Urbana Champaign
Champaign, IL, USA

Zhicheng Zhang
zczhang@cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

Ameesh Kapoor
Sean Hudson
ameesh@412foodrescue.org
sean@412foodrescue.org
412 Food Rescue
Pittsburgh, PA, USA

Hong Shen
Fei Fang
hongs@andrew.cmu.edu
feifang@cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

ABSTRACT

Food waste and food insecurity are two problems that co-exist worldwide. A major force to combat food waste and insecurity, food rescue platforms (FRP) match food donations to low-resource communities. Since they rely on external volunteers to deliver the food, communicating rescue task difficulty to volunteers is very important for volunteer engagement and retention. We develop a hybrid model with tabular and natural language data to predict the difficulty of a given rescue trip, which significantly outperforms baselines in identifying easy and hard rescues. Furthermore, using storyboards, we conducted interviews with different stakeholders to understand their perspectives on how to integrate such predictions into volunteers' workflow. Motivated by our findings, we developed three explanation methods to generate interpretable insights for volunteers to better understand the predictions. The results from this study are in the process of being adopted at Food Rescue Hero, a large FRP serving over 25 cities across the United States.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning by classification**; • **Human-centered computing** → **User studies**.

KEYWORDS

Volunteer retention; Food security; Food waste; Storyboarding

ACM Reference Format:
Zheyuan Ryan Shi, Jiayin Zhi, Siqi Zeng, Zhicheng Zhang, Ameesh Kapoor, Sean Hudson, Hong Shen, and Fei Fang. 2024. Predicting and Presenting Task Difficulty for Crowdsourcing Food Rescue Platforms. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3589334.3648155

WWW '24, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0171-9/24/05.
https://doi.org/10.1145/3589334.3648155

1 INTRODUCTION

Food waste and food insecurity exist in many places around the world. In the US alone, over 25% of the food is wasted, with an average American wasting about one pound of food per day [9]. Meanwhile, 12% of US households struggle to secure enough food [8].

Fortunately, food rescue platforms (FRP) are fighting against food waste and insecurity in over 100 cities around the world. FRPs receive safe, edible food from restaurants, grocery stores, and other businesses with excess food ("donors") and send it to organizations serving low-resource communities ("recipients"). This work is based on an ongoing collaboration between Food Rescue Hero and academic researchers. Food Rescue Hero is a large food rescue platform with operations in over 25 different cities across the US. Since its incorporation in 2015, Food Rescue Hero has delivered over 135 million pounds of food, worthy of over $339 million in retail value to the hundreds of thousands of people served by their over 8000 community partner organizations [30].

What enabled these FRPs to achieve such large-scale impact in the food security ecosystem? That is because FRPs deliver the food with the help of volunteers. Donors call FRPs when they have food to donate. The FRP dispatcher then matches this donation with a recipient. Once a match is found, the dispatcher posts the "food rescue" on the FRP's mobile app. Hereafter, the donation becomes visible to the volunteers who have the FRP's mobile app on their phone. If they choose to claim a "rescue", the app would instruct them where to pick up the donation and where to deliver it. The volunteer then goes out to complete the task.

That said, FRPs need many volunteers to stay afloat, as unclaimed rescues not only lead to immediate food waste, but also discourage the donors and recipients from participating. Yet, volunteers, after all, are not employees. Active volunteers have a high churn rate. One contributing factor is an unfavorable first experience performing a food rescue due to conditions such as a confusing pickup location, long travel time, or difficulty connecting with the point of contact. Such early attrition is a big loss to FRPs and their outreach effort. Thus, FRPs are eager to retain their volunteers by identifying rescues that are easier and recommend them more to new volunteers, and symmetrically, identifying possibly challenging rescues and gear them towards more experienced volunteers.

Centering our study around this challenge, we make the following three contributions. First, we develop a hybrid model to predict the difficulty level of each upcoming rescue trip. We assembled a food rescue difficulty dataset with expert labels from Food Rescue Hero. As manual labels are prohibitively expensive, we developed a BERT-based language model to generate pseudo labels to augment the training dataset. We then use these labels, along with other tabular features, to build the final difficulty prediction model. Our model can identify the easy rescues with 0.710 ROC-AUC, and the hard ones with 0.685 ROC-AUC, significantly outperforming baselines.

Second, we conduct an extensive user study to investigate how to integrate such a prediction model into the volunteers' interaction with the FRP. We conduct focus group sessions with 10 volunteers and staff members of Food Rescue Hero to elicit stakeholders' feedback on different integration designs. Further confirming the need for an AI-based tool for difficulty prediction, the user study also shows that the integration method that presents most information and allows the most volunteer autonomy is most preferred.

Third, the user study also reveals that volunteers want to better understand the rationale behind model's difficulty predictions. Towards this end, we develop three methods to generate explanations tailored to our end-users: natural language explanations, tag-based explanations, and augmented tag-based explanations. We demonstrate the unique advantages of each methods with real examples.

Food Rescue Hero has a network of around 45,000 volunteers and operates 378 rescues on average per day across 25 cities in the United States. Our ML models, the scaffolding findings, and the model explanation methods are in the process of being adopted at Food Rescue Hero. More broadly, our study is also applicable to other volunteer-based platforms beyond food rescue. Volunteer engagement and retention are a challenge on many such platforms. We provide a concrete paradigm for developing ML models for this challenge, and we also offer design implications for how such ML model should be integrated into the volunteer workflow.

2 RELATED WORK

The rapid growth of FRPs around the world has revealed the need for leveraging data and AI to make FRPs more efficient, robust, and socially responsible. A few focus their work on the matching between donors and recipients on the FRP [1, 22, 29], taking the vehicle routing into account [11, 25]. While all these works provide useful insights into the FRP operations, we do not focus on it here, because at most FRPs, donation matching is done by experienced staff who knows every detail about their donors and recipients, and FRPs would not sought after an algorithmic decision-making tool. Meanwhile, a participatory framework that allows for all community stakeholders to express their opinion on the matching could yield additional insights [21]. This work of Lee et al. [21] inspired our user study yet the focus of the paper is orthogonal to ours.

On the volunteer aspect of the FRP, the literature is focused on matching the "right" volunteers to each rescue task. Shi et al. [33] deployed a recommender system for volunteer-rescue pairing, whereas Manshadi and Rodilitz [23] and Shi et al. [34] proposed online learning algorithms for volunteer matching with performance guarantees. All these works aim at maximizing the "claim rate" on the FRP and propose some kind of algorithmic structure to account for volunteer retention. However, none of these works offers any verified evidence for the way volunteer retention is incorporated into the algorithm. For example, in [34], each volunteer is assumed to have an unknown vector which is supposed to characterize their reactions to push notifications for different rescues. Rather than directly go after the claim rate, we strive for understanding the mechanism of volunteer attrition, because only then can we develop robust volunteer engagement algorithms that works in the long run. As a first step, we investigate the role of rescue difficulty in volunteer engagement, and verify it with an extensive user study which has never been done before in this line of literature.

Indeed, recently, recognizing the limitations of merely offering technical perspectives to AI system design and deployment, the HCI and AI communities have started to explore how to elicit impacted stakeholders' perspectives to incorporate their needs, constraints and desires into the AI design and deployment process [32]. For example, Kuo et al. [19] developed AI Lifecycle Comicboarding to explain the entire development life cycle of a housing allocation algorithm to the community, demonstrating the feasibility of making the design of social service AI accessible to a wide range of stakeholders. To probe around social workers' challenges in working with an algorithmic decision support tool, Kawakami et al. [16] developed ten design concepts to understand how to improve the AI interface in workers' day-to-day decision-making process.

Centering around the impacted stakeholders' perspectives, our work is related to the prior work by using design materials as elicitation method. Our work also contributes to this line of research by focusing on the context of a voluntary FRP in the real world. We design and use a set of storyboards [35], each representing different AI integration methods, to probe multiple stakeholder's perspectives on how to integrate a difficulty prediction AI into volunteers' workflow to avoid generating any harm to the community.

3 PREDICTING RESCUE DIFFICULTY

As stated earlier, the challenge of volunteer retention is often associated with a mismatch between the difficulty of the rescue and the volunteer's experience. By predicting the difficulty level of rescue tasks, we hope to get to the root of early volunteer attrition.

3.1 Dataset

We use the database at Food Rescue Hero to develop and evaluate our models. The database contains over 380,000 rescues in the past five years. It also contains a record of all the donor and recipient organizations, the volunteers, and the phone call history to and from Food Rescue Hero. For the purpose of predicting rescue difficulty, each data point comes in the form of (rescue, difficulty), where rescue is the collection of all the tabular features and difficulty is the target label. In what follows, we introduce the feature engineering and label acquisition processes, separately.

3.1.1 Features. Based on our experience at Food Rescue Hero, we identified a set of tabular features most relevant to the prediction tasks. At a high level, these tabular features can be categorized into the following two types.

The first type is the rescue information, which identifies the inherent attributes of the rescue task. For example, this includes

the time of rescue publication, the quantity of the food, and the weather information. For weather information in particular, we use the Climate Data Online service to get the daily summary climate indicators on the day of the food pick-up [26]. We retrieve the data for the weather station closest to the donor location. The indicators include precipitation, snow, high temperature, low temperature, wind speed and movement, and water evaporation.

The second type is the participant information, which involves the three types of participants in the food rescue operation: volunteers, donors, and recipients. This includes the distance between the volunteer and the donor or the recipient, the length of time the participant has been with the FRP, the number of times the participant has participated in a rescue task, and the volunteer average past ratings overall as well as at the particular donor or recipient organization. We will discuss more about the ratings in the label part later in the section.

Finally, aside from these tabular features, we also leverage the text-based comments provided by some volunteers as additional input. These comments are tokenized and then processed with language models for extra feature extraction. We note that these comments are not part of the aforementioned data point (rescue, difficulty), as they are only available after the rescue is completed. We will be using these comments in a different way in Section 3.2.

3.1.2 Label. It can be subtle to define the difficulty level of one food rescue task. There are several proxies for quantification of difficulty. First, the number or length of phone calls at Food Rescue Hero could be useful, as the volunteers tend to call Food Rescue Hero when something goes wrong. One could assign a label to each rescue in this way, but this is not an ideal proxy. Since the content of the dialogue is unknown, it is hard to identify the reasoning for the phone call. Besides, the FRP has no visibility into the direct communication between volunteers and donors as well as recipients, rendering such call data at best an incomplete characterization of the rescue difficulty. The second option is the rating information. Volunteers are requested to provide a rating on a scale of 1 to 4 for the rescue they completed. The intuition is that higher ratings correspond to easy tasks as users are satisfied with the process. Roughly 20% of the rescues have such rating information, thus still a decent size of data. However, people have different standards for ratings and they are not always aligned with rescue difficulty. To illustrate this, as mentioned earlier, volunteers are allowed to provide comments to explain their ratings optionally. We performed topic modeling analysis of user comment corpus using Latent Dirichlet Allocation (LDA). When ratings = 1, we observe one important topic "food wasn't available" which does not necessarily indicate difficulty. With ratings = 2, volunteers appreciate the helpfulness of the staff, while for ratings = 3, volunteers complain about the wrong pickup address. It is hard to find a direct correspondence between ranking and difficulty.

Therefore, we decided to leverage the volunteer comments, and had two domain experts at Food Rescue Hero to label a small subset of the rescues which have these comments, which amounts to 1000 data points. They identify the obviously "Easy" and "Hard" tasks and label the rest as "Undetermined". Filtered by the domain experts, these comments help us get as close as possible to representing task difficulty. Of course, one caveat is that such manual labeling is very

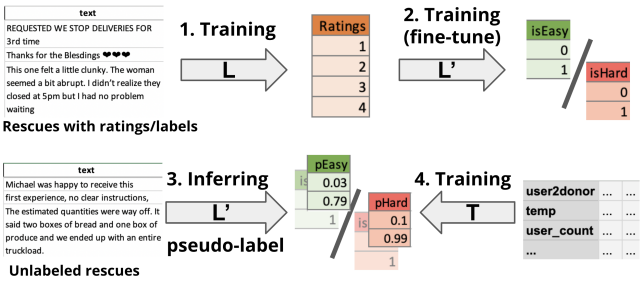

Figure 1: Our proposed algorithm. L stands for the BERT language model. T stands for the tabular prediction model.

expensive and we only create a small labeled dataset. However, as we will show in Section 3.2, we can alleviate this limitation with our algorithm design.

3.2 Modeling

We now introduce our algorithm for predicting difficulty levels for food rescue tasks. We have two symmetrical binary prediction tasks: when predicting whether a rescue is easy, we group the "hard" and "undetermined" rescues to form the negative class; when predicting whether a rescue is hard, we the "easy" and "undetermined" rescues form the negative class. But in either case, the model architecture is identical.

The most straightforward way would be to use any off-the-shelf predictor to predict the difficulty label from the tabular features introduced in Section 3.1. However, by doing so, we would have too few data points as manual labeling the data is extremely expensive.

In order to alleviate the scarcity of labeled data and to make full use of the information we actually have, we expand our dataset with pseudo labels. We first fit a pre-trained BERT language model on the comments from the labeled dataset to predict the difficulty levels. The fitted BERT model can then generate soft difficulty predictions as scores within the range of $[0, 1]$ for all other (unlabeled) rescues with comments (Step 3, Figure 1). We treat these predictions as the pseudo labels. Finally, we combine the tabular features from the ground truth labeled data plus the data points that have pseudo labels to train for the final prediction with a tabular model (Step 4, Figure 1).

We can leverage even more information into our workflow by recognizing the correlation between ratings and difficulty levels. Although ratings are not perfect proxies for the latter, they are available for a lot more rescues. Thus, instead of directly tuning BERT on the binary difficulty, we first train it against the 4 ratings using the bigger dataset (Step 1, Figure 1), and then fine-tune it on the binary difficulty labels (Step 2, Figure 1), hoping the rating information can improve the quality of pseudo labels. This becomes our final algorithm as shown in Figure 1.

4 EXPERIMENT RESULTS

In this section, we report the experiment results of our algorithm and multiple baselines based on historical data.

For all algorithms we set aside the same test set using the ground truth labels provided by the Food Rescue Hero domain experts. For

	Validation Set		Test Set	
Predictor	AUC	Std. Dev.	AUC	Std. Dev.
GBM	**0.686**	**0.118**	0.710	0.023
RF	0.663	0.057	0.703	0.027
LR	0.562	0.055	0.535	0.025
SVM	0.485	0.050	0.470	0.022
MLP	0.495	0.027	0.495	0.031
KNN	0.654	0.022	0.643	0.021

Table 1: Predicting easy rescues using six predictors: LightGBM, random forest, linear regression, support vector machine, multi-layer perceptron, and K nearest neighbors. ROC-AUCs are averaged over 10 trials, with standard deviation shown as well. Decision is made on validation set; test set results are provided just for reference.

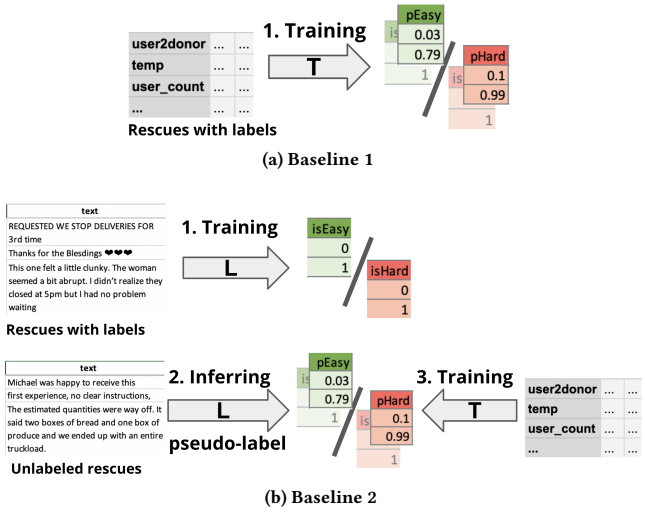

Figure 2: Illustrations of the two baseline algorithms. L stands for the BERT language model. T stands for the tabular prediction model.

algorithms that involve pseudo labels, the training and validation sets contain all rescues that have volunteer comments. Ground truth labels are used when available, otherwise we use the pseudo labels generated by the trained BERT model. For algorithms that do not involve pseudo labels, we use only the ground truth labels for the training and validation sets. All experiments are conducted on a machine with Intel Core i7-7700K CPU, NVIDIA TITAN Xp GPU, and 64GB RAM.

First, we conduct experiments to determine the final-step predictor in our algorithm as described in Section 3.2. As shown in Table 1, we focus on predicting the easy rescues, and try 6 different predictors on the validation set: LightGBM, random forest, linear regression, support vector machine, multi-layer perceptron, and K nearest neighbors. For all tabular predictors, we use the default hyperparameter settings. LightGBM achieves the best AUC 0.686 on the validation set among all these predictors. Thus, for the remainder of the paper, we use LightGBM as the final-step predictor for our algorithm, and for all other baselines where they require such a final-step predictor. For completion, we also show their performance on the test set. Here, we can see that our algorithm with LightGBM achieves 0.710 AUC. In fact, two-sample t-tests show that our algorithm with LightGBM is significantly better than all predictors except for random forest with $p < 10^{-5}$.

We now move on to report the final results of our algorithm against two baselines. The first baseline is simply the LightGBM predictor we converged on earlier. Optimistically, the tabular features already include all the information required for prediction. That is, we ignore the text-based comment corpus and use all the tabular features as input to predict the rescue difficulty with LightGBM. This is illustrated in Figure 2a. The second baseline is similar to our algorithm, except for the BERT training process. Rather than first train BERT on rescue ratings and then fine tune it on rescue difficulty labels, we train BERT directly on rescue difficulty labels, skipping the rescue ratings. The final predictor, as reasoned above, in still LightGBM. This is illustrated in Figure 2b.

As shown in Table 2, our algorithm achieves 0.710 ROC-AUC on predicting easy rescues, and 0.685 ROC-AUC on predicting hard rescues. Baseline 1 shows a significantly lower ROC-AUC on both

	Easy		Hard	
Algorithm	AUC	Std. Dev.	AUC	Std. Dev.
Ours	**0.710**	**0.023**	**0.685**	**0.041**
Baseline 1	0.543	0.024	0.495	0.025
Baseline 2	0.709	0.037	0.563	0.000

Table 2: The performance metrics of our algorithm and two other baselines. For predicting easy rescues (and hard rescues, respectively), we compare their mean ROC-AUC over 10 random seeds, and report the standard deviation across the 10 trials.

prediction tasks than our algorithm, with $p < 10^{-8}$ for both two-sample t-tests. This is probably because the tabular features themselves are not sufficient for predicting rescue difficulty. These results suggest that the comment text corpus and the pseudo-label generation are indeed helpful. Baseline 2 achieves only a slightly lower AUC on easy rescue prediction than our algorithm, with $p = 0.947$ for two-sample t-test, and thus negligible differences. But its performance on hard rescue prediction is significantly worse, with $p < 10^{-5}$ for two-sample t-test. The overall worse performance of baseline 2 compared to our algorithm means that first training BERT on rescue ratings indeed makes sense. Even though rescue ratings are not equivalent to rescue difficulty, they are obviously correlated, and there are far more (about 100x more) data points with ratings than with difficulty labels in the dataset. The sheer volume of data possibly played a role here. Furthermore, why is the gap much bigger on hard rescues than easy ones? There is also an intuitive answer. When people give out ratings, it is natural to default to high ratings unless they feel really urged to make them lower because of their experience. As a result, low ratings

presumably correlate with hard rescues much more than high ratings correlate with easy rescues. Thus, ignoring ratings has a much greater cost when predicting hard rescues.

5 UNDERSTANDING DIFFERENT STAKEHOLDERS' PERSPECTIVES ON DEPLOYING THE AI

The ML model in the previous sections only gets half the job done: it is equally important and challenging, if not more, to design how to best present difficulty information to volunteers since volunteer experience is the ultimate goal of our work. Thus, we conducted a series of focus groups and interview studies with three different stakeholder groups on Food Rescue Hero (newcomers, experienced volunteers and staff members) to understand their perspectives on how to integrate the AI into the existing workflow. The research team collectively generated six different AI-integration methods as design concepts, across different levels of back-end scaffolding and front-end information display, and designed six storyboards to make those concepts accessible to study participants. The study sessions were conducted in July 2023. Following best practices of community engagement in HCI [13, 28], before the user study, we consulted our community partners and researchers in similar domains to ensure that we followed the community norms when recruiting and working with our study participants. During the research, we were transparent about our research goals to our participants and actively built rapport with them. The study protocol was approved by our Institutional Review Board (IRB).

Our results suggest that (1) overall, volunteers value the difficulty prediction AI as a decision-supporting tool to help them navigate the complicated workflow; (2) in terms of integration method, they prefer the least back-end scaffolding and more front-end display to integrate the AI; and (3) they strongly request more explanation to better understand the difficulty prediction AI with a goal to better support their decision-making process.

5.1 Method

The use of storyboards is a common elicitation method in HCI to present visual narratives and rapidly visualize interfaces that communicate the context in which a technology will be used [20]. By using a series of storyboards, researchers probe needs and explore design alternatives with particular use populations [10, 15], instead of merely validating the best narrative. Focus groups facilitate guided discussions for user insights on preliminary ideas, supplying diverse data best enhanced with other research methods, such as storyboards [12]. The dynamic interaction encourages participants to share experiences and needs. Focus groups enable the development of collective insights on shared problems and solutions to the problems [39].

5.1.1 Storyboards. We designed 6 storyboards, each representing a different method of integrating the AI into volunteer's existing workflow.

Existing Workflow: Currently, Food Rescue Hero volunteers open the app to view available tasks on the map, access detailed task information through a floating window, and opt to undertake tasks. Outside apps, they also receive task notifications on their phone.

Possible AI Integration methods: Through iterative discussions, we decided to test six different AI-integration methods as design concepts, across different levels of back-end scaffolding and front-end information display. There are three levels of back-end scaffolding: A) Low: showing all tasks to new volunteers on the map, and sending notifications of all tasks to them; B) Medium: showing all tasks to new volunteers on the map, but customizing notifications by only sending easy tasks to them; C) High: only showing easy tasks to new volunteers on the map, and customizing notifications by only sending easy tasks to them. There are two ways of front-end display: 1) displaying difficulty levels on screen, and 2) not displaying difficulty levels on screen. Combining the three back-end scaffolding levels with the two front-end display method yields six different design concepts in total: A.1, A.2, B.1, B.2, C.1, C.2. We then capture these six design concepts with six storyboards to show to our study participants (see Figure 3 for an example). We summarize these design concepts in Table 4 in the Appendix.

5.1.2 Data Collection & Study Protocol. We recruited 4 new volunteers who had done no more than 5 tasks, and 4 experienced volunteers who had done at least 20 tasks on Food Rescue Hero. Staff members of Food Rescue Hero sent out recruitment messages through the platform. We also conducted two interviews with two other Food Rescue Hero admins. We conducted all sessions over Zoom. The study sessions lasted 36 minutes on average and each participant was compensated with $30 for their participation.

We ran focus groups with one new and one experienced volunteer since newcomers might lack insight-sharing abilities. To ensure volunteers spoke freely without admin presence, we held 1-1 interviews with Food Rescue Hero admins. Each session began with a walkthrough of the volunteer workflow and a brief on the prediction model. Participants then reviewed six design storyboards, thinking aloud about their design preferences and the model's integration. After reviewing, they rated each design on a three-point scale: "mostly positive", "neutral", and "mostly negative". Lastly, we sought suggestions for better model integration.

5.2 Data Analysis

We adopted a reflexive thematic analysis approach [3, 4] across storyboards to understand broader themes in participants' responses, a common approach in HCI storyboarding studies [10]. Two researchers conducted open coding on transcriptions of approximately 218 minutes of audio recording and generated a total of 62 codes. We iteratively refined our codes in a reflexive thematic approach to collaboratively shape themes [24]. In total, we conceptualized 3 third-level themes, 8 second-level themes, and 18 third-level themes. For computation, the ratings "mostly positive", "neither positive nor negative", and "mostly negative" were valued at 1, 0, and -1. We identified the top design concept by averaging the cumulative scores of each storyboard per participant.

5.3 Findings

We organize our findings around three themes identified through our analysis. Quotes from new volunteers (*NV*), experienced volunteers (*EV*), and platform administrators (*A*) are referred to as *NV_Pi*, *EV_Pj*, or *A_Pm*, respectively, where *i* represents the participant index in each stakeholder group. The analysis revealed that new

volunteers and experienced volunteers had converging preferences on the AI integration method. Admins, on the other hand, offered a set of different perspectives and suggested nuanced strategies.

5.3.1 (1) What are user perspectives on the use of AI to predict task difficulty? Difficulty prediction is useful. The integration of an AI model to predict task difficulty has been positively received by volunteers, both novice and experienced, who **acknowledge its usefulness in their experience**. NV_P4 felt the AI model is a pivotal enhancement to the app, saying, *"I am mostly positive I want this feature to be on the app*", as it assists in mentally preparing for upcoming tasks. Furthermore, as EV_P4 put it, *"the AI makes use of text-based feedback about your food rescue experience on the app"*, and this feedback loop is vital for the AI to accurately forecast the difficulty of future rescues. The Food Rescue Hero staff also stressed on the usefulness, by articulating their **concerns regarding volunteers' underestimation of the complexity of tasks**, so **adding the difficulty levels provided by the prediction model are necessary and useful**. They stressed that task difficulty extends beyond the obvious elements of time allocation, drive duration, and location. Subtle, yet crucial, aspects like effective communication, the preparedness of donors, and interpersonal issues further contribute to the challenge of tasks. One staff member pointed out: *"[...] It's also been trouble with expectations, because they think difficulty means just more work. Things that make it difficult are more like interpersonal issues, like, you have to negotiate with the person you're delivering to." (A_P1)*

5.3.2 (2) What AI Integration method is most preferred? The least back-end scaffolding and most front-end display is preferred. Based on calculation of the average rating scores, volunteers – both newcomers and experienced – preferred **the least back-end scaffolding and more front-end display** in integrating the prediction model into their workflow, that is, the integration method that offers most information and allows most autonomy. This corresponds to displaying all the tasks across all difficulty levels on the map (front-end) and sending notifications of all the tasks to volunteers (back-end), as shown in Figure 3. Score breakdown: A.1 (0.4), A.2 (1), B.1 & B.2 (0.4 each), C.1 (0), and C.2 (-0.8).

More front-end display is preferred. Contrary to the notion that too much information can be overwhelming, volunteers expressed a **strong desire for more information** at the early stage. The sooner they can see this information given by the prediction model in their workflow, the better equipped they feel to make decisions. One volunteer encapsulated this view by remarking: *"[...] I advocate for early access to difficulty level information. Restricting visibility isn't the solution. My stance is clear, let us see it all, and let us see it early." (EV_P2)* Similarly, staff members of Food Rescue Hero also supported more information presentations on the front-end that always displays difficulty levels, **considering the learning curve of using a new app**. Notably, for older volunteers, who might be less tech-savvy and might feel intimidated by technology, it is important to **display the information all the time in their workflow**. *"[...] They weren't sure where to look or how to interpret the data. It's particularly significant for our newer volunteers, notably the older segment who might not be as comfortable with technology. They need and deserve an interface that's intuitive and always transparent." (A_P1)*

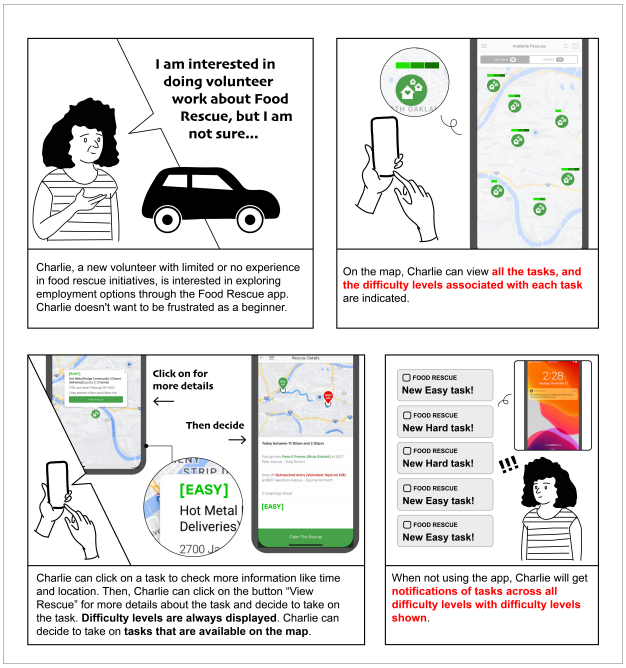

Figure 3: Volunteers prefer design concept A.2: displaying difficulty levels on screen, while showing and sending notifications of tasks of all difficulty levels.

Less back-end scaffolding and more volunteer autonomy is preferred. Volunteers believed **the prediction model should play an assistive role in helping them make decisions on taking tasks**. They wanted to **have full control of what tasks they could view, and what notifications they could receive**, instead of the other integration methods that make decisions for them. One volunteer said: *"Being restricted by the app in terms of when I should be notified or what I'm supposedly capable of is constricting, I find that a little offensive as a volunteer. Yeah, let me decide what we want to do. While notifications are appreciated, I strongly oppose any restrictions on what I can view." (EV_P1)*

5.3.3 (3) With the least scaffolding, how can we improve users' experience with respect to this prediction model? Explanations of the model are needed. On the one hand, Food Rescue Hero staff pointed out that **the difficulty prediction model is useful as it considers multiple facets of the complexity of task execution**. On the other hand, **explanations are strongly requested by volunteers to accept, understand, and use the results from the prediction model** to make their own decisions.

When considering decisions based on difficulty levels, volunteers expressed a strong desire to **understand the underlying mechanisms by which the prediction model gauges difficulty**. It's crucial for the prediction model, along with the integration design, to **provide explanations to better support human logic and reasoning**. One volunteer stressed on the importance of explainability for newcomers: *"Particularly for newcomers, there's always that underlying query: how do we differentiate between an easy task and a hard one? The absence of any standardized definition makes*

it even more challenging." (NV_P3) On a similar note, another volunteer highlighted the significance of clarity in terms of assisting volunteers, remarking *"[...] Because it's not offering me any insight about the difficulty, such vagueness doesn't serve to assist me in my role as a volunteer." (EV_P3)* They further elaborated on the necessity for clear categorization to make volunteers understand the difficulty levels for themselves, suggesting *"[...] If there's an understanding of the parameters used to classify tasks as easy or difficult, it would be beneficial to state them. [...] So what would a user consider that a difficulty? [...] So what are you going to give the difficulty level based upon?" (EV_P3)*

6 EXPLANATIONS FOR DIFFICULTY PREDICTION

Our qualitative findings in Section 5.3 confirm the usefulness of difficulty prediction and also reveal a crucial need for interpretable explanations. This revealed need echoes prior studies showing that explanations play a pivotal role in enhancing user trust and understanding of machine learning model predictions, since users often find raw model outputs arcane and untrustworthy without further explanation [17, 37].

While the need for explanations is clear, there is no single way of generating explanations. Researchers studied the effect of different types of explanations on user trust in AI systems, such as input attribution, rule-based explanations, output attribution, and textual explanations [2, 14, 18, 38]. The effect of explanations varied depending on the user's prior knowledge, task complexity, and model accuracy.

In response to this need for diverse types of explanations, we provide three explanation-generating methods tailored to our volunteers on the food rescue platform, aligned with Explainable AI needs identified in previous research [18]. We first develop natural language explanations due to their application in recommendation systems by generating personalized recommendations [6, 7]. We also provide two tag-based explanations, motivated by Vig et al.'s method of explaining recommendations using tags [36]. The two tag-based explanations offer respectively an easily digestible format and a data-rich contextualization. These three types of explanation collectively aim to address the spectrum of user preferences for details and context.

6.1 Explanation Methods

6.1.1 Natural Language Explanations. We first employ Local Interpretable Model-agnostic Explanations (LIME), a popular technique for generating model-agnostic prediction explanations to extract insights into our model [31]. With LIME, we extract the top 10 features that influence the most model's prediction. These features are then used as inputs to a large language model (LLM) [5], which is prompted to construct a coherent and concise natural language sentence that explains the contributing factors for the user in non-technical and simple terms. Here we use GPT-4 [27] as the LLM for generating the explanations. The structure of the prompt used is shown in Listing 1, which is composed of the general instruction of the explanation task and description of LIME, the meaning of the features, and the top 10 feature importance values generated by LIME.

```
% [Instruction for the explanation]
You are tasked with explaining how different
↪ features influence the difficulty level of
↪ food rescue tasks to an audience with no
↪ expertise in AI...
In the context of LIME, or Local
↪ Interpretable Model-agnostic Explanations,
↪ interpreting the outputs...

% [Feature Meanings]
PRCP means precipitation
...
user_counts means how many rescues has the
↪ user completed previously, higher means
↪ more experience

% [Top 10 Features from LIME]
Feature user_counts <= 5.00: 0.69
Feature total_quantity > 10.00: 0.15
...

Complete this: this task is {HARD/EASY}
↪ because
```

Listing 1: Prompt provided to the LLM to generate natural language explanations.

6.1.2 Tag-Based Explanations. Building on the natural language explanations, we further refine the information into a tag-based format. Here, we utilize the LLM again to distill the sentence into a set of tags, which typically consist of an adjective and noun pairing, thereby providing a more snapshot overview of the features' implications. We choose to use LLMs for tag generation instead of a rule-based method from the LIME outputs, because LLMs offer more diversity in the generated tags, mitigating the repetition and redundancies of the rule-based method. To make the tags more user-friendly, we additionally impose some constraints on the tags in the form of templates. Specifically, we add a `[templates]` section in the prompt to tell the LLM which constitutes a good tag for a set of user-related features, like the `user_counts` feature shown in Listing 2. The full prompt can be found in Appendix A.

```
% [Instruction for tag generation]
Now create clear, distinct tags by combining
↪ an adjective and noun phrase to explain
↪ why a task is easy or hard...

% [templates]
For 'user_counts' related features, use:
↪ "{hard or easy} for
↪ {less/moderate/more/the most} experienced"
...
```

Listing 2: Prompt provided to the LLM to generate tags

6.1.3 Augmented Tag-Based Explanations. To enhance the descriptive power of the tags, each feature is supplemented with additional contextual information. Specifically, we incorporate key data such as its percentile within the training dataset, as well as the actual

feature value formatted with relevant units. The augmented information is tailored for each individual feature, considering factors like the usefulness of providing comparative metrics and units. For instance, for the feature `total number of rescues`, we present its percentile based on the training set distribution. Conversely, for features where precise values matter, like time-related features, we provide the actual figures. Moreover, for features that benefit from multifaceted information, like `food_quantity`, we provide a list of information, including both percentile and the actual figure with the unit.

6.2 Qualitative Analysis of Explanations

We examine the advantages of each explanation type through a qualitative analysis of two example cases shown in Table 3.

Natural language explanations appear intuitive and understandable, making them suitable for volunteers who seek clarity without any technical detail. This approach offers easy-to-understand reasons like "the recipient has many completed rescue" (Instance 2) and "a mixed satisfaction from previous rescues" (Instance 1), which can be easily understood without requiring any further context.

The tag-based explanations provide a much more succinct summary that strips the explanation down to its core components. This is ideal for volunteers already familiar with the system who want to prioritize a speedy rescue but might be viewed as confusing for new volunteers unfamiliar with some of the jargon like "frequent recipient rescues" (Instance 2).

The augmented tag-based explanation combines the advantages of the previous two approaches by having both the core features and the detailed information. They provide qualitative information like "frequent recipient rescues" (Instance 2), but also quantitative metrics like "recipient's past rescue counts higher than 92%" (Instance 2). These can be useful for volunteers seeking more in-depth reasoning so that they can more readily rely on the model's prediction results. But for users with less experience or seeking less rationale behind the prediction, these tags can be perceived as too verbose.

Type	Explanation
	Instance 1
Natural Language	This task is HARD for you because you have less experience, a mixed satisfaction from previous rescues, and the recipient location is far.
Tag-based	Hard for less experienced • Prior mixed satisfaction • Far recipient location
Augmented Tag-based	Hard for less experienced (your past rescue counts lower than 26%) • Prior mixed satisfaction (your average rating higher than 28%) • Far recipient location (higher than 94%)
	Instance 2
Natural Language	This task is EASY for you because you have plenty of experience, the recipient has many completed rescues, and the food quantity is small.
Tag-based	Easy for more experienced • Frequent recipient rescues • Small food quantity
Augmented Tag-based	Easy for more experienced (your past rescue counts higher than 85%) • Frequent recipient rescues (recipient's past rescue counts higher than 92%) • Small food quantity (lower than 72%, 2 items)

Table 3: Two explanation examples for task difficulty.

7 DISCUSSION AND CONCLUSION

Volunteer attrition is a major pain point for food rescue platforms. The key to address this challenge is to match volunteers with tasks of difficulty commensurate with their experience. We develop a hybrid ML model with tabular and natural language data to identify easy and hard rescues. We address the label scarcity issue by generating pseudo labels which significantly improved the prediction performance. We believe that merely developing such an ML model is far from addressing the issue in the real world. Thus, we conduct an extensive user study with diverse stakeholders to investigate how to best integrate such difficulty information into volunteer's workflow. In addition to confirming the need for an ML-based tool for rescue difficulty prediction, the user study revealed that volunteers prefer less back-end scaffolding and more front-end display. The study reveals more nuances in deploying the model to the volunteers, as well as pathways towards more balanced and creative integration mechanisms of such a prediction model.

In fact, the user study also shed light on the ML model development itself. As discovered in Section 5.3, volunteers want to better understand the mechanisms by which the ML model gauges difficulty. Thus, we developed three LLM-based methods to generate explanations of the predictions. The three types of explanation can accommodate a variety of users with different preferences. However, limitations arise from employing the model-agnostic LIME and the possibility of hallucinations in LLM. We control hallucination by carefully crafting the prompt and by controlling the inference temperature. A future direction would be to continually refine the trustworthiness and robustness of the explanation.

This work was conducted in partnership with Food Rescue Hero. The ML model and the three model explanation methods are in the process of being deployed at Food Rescue Hero. Volunteers will be able to choose the explanation they prefer. All these models will be presented to the volunteers according to the scaffolding design findings from this study.

ACKNOWLEDGMENTS

We thank Zhiyu Chen for the constructive discussions that led to this work. This work was supported in part by NSF grant IIS-2046640 (CAREER), CMU CyLab Seed Grant, and CMU Block Center for Technology and Society Seed Fund Award. Co-author Fang is supported in part by Sloan Research Fellowship. Co-author Shen is also supported by a PIT-UN award.

REFERENCES

[1] Martin Aleksandrov, Haris Aziz, Serge Gaspers, and Toby Walsh. 2015. Online fair division: analysing a food bank problem. In *Proceedings of the 24th International Conference on Artificial Intelligence*. 2540–2546.

[2] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information fusion* 58 (2020), 82–115.

[3] Virginia Braun and Victoria Clarke. 2012. Thematic analysis. In *APA handbook of research methods in psychology, Vol. 2. Research designs: Quantitative, qualitative, neuropsychological, and biological*, Harris Cooper, Paul M Camic, Debra L Long, A T Panter, David Rindskopf, and Kenneth J Sher (Eds.). American Psychological Association, 57–71.

[4] Virginia Braun and Victoria Clarke. 2019. Reflecting on reflexive thematic analysis. *Qualitative research in sport, exercise and health* 11, 4 (2019), 589–597.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[6] Shuo Chang, F Maxwell Harper, and Loren Gilbert Terveen. 2016. Crowd-based personalized natural language explanations for recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 175–182.

[7] Hanxiong Chen, Xu Chen, Shaoyun Shi, and Yongfeng Zhang. 2021. Generate natural language explanations for recommendation. *arXiv preprint arXiv:2101.03392* (2021).

[8] Alisha Coleman-Jensen, Matthew P Rabbitt, Christian A Gregory, and Anita Singh. 2018. Household Food Security in the United States in 2017. *USDA-ERS Economic Research Report* (2018).

[9] Zach Conrad, Meredith T Niles, Deborah A Neher, Eric D Roy, Nicole E Tichenor, and Lisa Jahns. 2018. Relationship between food waste, diet quality, and environmental sustainability. *PloS one* 13, 4 (2018), e0195405.

[10] Scott Davidoff, Min Kyung Lee, Anind K Dey, and John Zimmerman. 2007. Rapidly exploring application design through speed dating. In *Proceedings of the 9th international conference on Ubiquitous computing*. Springer, 429–446.

[11] Canan Gunes, Willem-Jan van Hoeve, and Sridhar Tayur. 2010. Vehicle routing for food rescue programs: A comparison of different approaches. In *CPAIOR*.

[12] Bruce Hanington and Bella Martin. 2012. *Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions*. Quarto Publishing Group USA.

[13] Christina Harrington, Sheena Erete, and Anne Marie Piper. 2019. Deconstructing community-based collaborative design: Towards more equitable participatory design engagements. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–25.

[14] Robert R Hoffman, Shane T Mueller, Gary Klein, and Jordan Litman. 2023. Measures for explainable AI: Explanation goodness, user satisfaction, mental models, curiosity, trust, and human-AI performance. *Frontiers in Computer Science* 5 (2023), 1096257.

[15] Kenneth Holstein, Bruce M. McLaren, and Vincent Aleven. 2019. Designing for Complementarity: Teacher and Student Needs for Orchestration Support in AI-enhanced Classrooms. In *Proceedings of the 20th International Conference on Artificial Intelligence and Education*. Springer, 257–268.

[16] Anna Kawakami, Venkatesh Sivaraman, Logan Stapleton, Hao-Fei Cheng, Adam Perer, Zhiwei Steven Wu, Haiyi Zhu, and Kenneth Holstein. 2022. Why Do I Care What's Similar? Probing Challenges in AI-Assisted Child Welfare Decision-Making through Worker-AI Interface Design Concepts. In *DIS Conference on Designing Interactive Systems (DIS '22)* (June 13-June 17, 2022). ACM, New York, NY, USA, 1–17.

[17] Eoin M Kenny, Courtney Ford, Molly Quinn, and Mark T Keane. 2021. Explaining black-box classifiers using post-hoc explanations-by-example: The effect of explanations and error-rates in XAI user studies. *Artificial Intelligence* 294 (2021), 103459.

[18] Sunnie SY Kim, Elizabeth Anne Watkins, Olga Russakovsky, Ruth Fong, and Andrés Monroy-Hernández. 2023. " Help Me Help the AI": Understanding How Explainability Can Support Human-AI Interaction. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–17.

[19] Tzu-Sheng Kuo, Hong Shen, Jisoo Geum, Nev Jones, Jason I Hong, Haiyi Zhu, and Kenneth Holstein. 2023. Understanding Frontline Workers' and Unhoused Individuals' Perspectives on AI Used in Homeless Services. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–17.

[20] James A. Landay and Brad A. Myers. 1996. Interactive Sketching for the Early Stages of User Interface Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 43–50.

[21] Min Kyung Lee, Daniel Kusbit, Anson Kahng, Ji Tae Kim, Xinran Yuan, Allissa Chan, Daniel See, Ritesh Noothigattu, Siheon Lee, Alexandros Psomas, and Ariel D. Procaccia. 2019. WeBuildAI: Participatory Framework for Algorithmic Governance. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 181 (Nov. 2019), 35 pages. https://doi.org/10.1145/3359283

[22] Taylor Lundy, Alexander Wei, Hu Fu, Scott Duke Kominers, and Kevin Leyton-Brown. 2019. Allocation for social good: auditing mechanisms for utility maximization. In *ACM EC*.

[23] Vahideh Manshadi and Scott Rodilitz. 2020. Online Policies for Efficient Volunteer Crowdsourcing. *arXiv preprint arXiv:2002.08474* (2020).

[24] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and inter-rater reliability in qualitative research: Norms and guidelines for CSCW and HCI practice. *Proceedings of the ACM on human-computer interaction* 3, CSCW (2019), 1–23.

[25] Divya Jayakumar Nair, Hanna Grzybowska, David Rey, and Vinayak Dixit. 2016. Food rescue and delivery: Heuristic algorithm for periodic unpaired pickup and delivery vehicle routing problem. *Transportation Research Record* 2548, 1 (2016), 81–89.

[26] Climate Data Online. 2024. https://www.ncei.noaa.gov/cdo-web/.

[27] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[28] Jennifer Pierre, Roderic Crooks, Morgan Currie, Britt Paris, and Irene Pasquetto. 2021. Getting Ourselves Together: Data-centered participatory design research & epistemic burden. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–11.

[29] Canice Prendergast. 2016. The Allocation of Food to Food Banks. *EAI Endorsed Trans. Serious Games* 3, 10 (2016), e4.

[30] 412 Food Rescue. 2023. 2022 Impact Report. https://412foodrescue.org/wp-content/uploads/2023/08/412-Food-Rescue_2022-Impact-Report_Final.pdf.

[31] 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*. 1135–1144.

[32] Hong Shen, Leijie Wang, Wesley H Deng, Ciell Brusse, Ronald Velgersdijk, and Haiyi Zhu. 2022. The model card authoring toolkit: Toward community-centered, deliberation-driven AI design. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 440–451.

[33] Zheyuan Ryan Shi, Leah Lizarondo, and Fei Fang. 2021. A recommender system for crowdsourcing food rescue platforms. In *Proceedings of the Web Conference 2021*. 857–865.

[34] Zheyuan Ryan Shi, Zhiwei Steven Wu, Rayid Ghani, and Fei Fang. 2022. Bandit Data-Driven Optimization for Crowdsourcing Food Rescue Platforms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 12154–12162.

[35] Khai N Truong, Gillian R Hayes, and Gregory D Abowd. 2006. Storyboarding: an empirical determination of best practices and effective guidelines. In *Proceedings of the 6th conference on Designing Interactive systems*. 12–21.

[36] Jesse Vig, Shilad Sen, and John Riedl. 2009. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent user interfaces*. 47–56.

[37] Chao Wang and Pengcheng An. 2021. Explainability via Interactivity? Supporting Nonexperts' Sensemaking of pre-trained CNN by Interacting with Their Daily Surroundings. In *Extended Abstracts of the 2021 Annual Symposium on Computer-Human Interaction in Play*. 274–279.

[38] Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y Lim. 2019. Designing theory-driven user-centric explainable AI. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–15.

[39] Sue Wilkinson. 1998. Focus group methodology: a review. *International journal of social research methodology* 1, 3 (1998), 181–203.

A PROMPT FOR LARGE LANGUAGE MODEL TO GENERATE EXPLANATIONS

Here we list in full the prompt used as input to the GPT-4, in Listing 3 and 4, to generate natural language explanation and tags.

```
You are tasked with explaining how different features influence the
↪ difficulty level of food rescue tasks to an audience with no expertise
↪ in AI.

I will provide you with a list showing how significant each aspect is
↪ according to the LIME analysis. Each item in the list is composed of:
↪ [feature] [inequality sign] [threshold]: [feature importance].

In the context of LIME, or Local Interpretable Model-agnostic
↪ Explanations, interpreting the outputs is a nuanced process that
↪ requires careful attention to detail. Each item in the LIME output
↪ list, represented in the format [feature] [inequality sign]
↪ [threshold]: [feature importance], holds significant information about
↪ how the model makes its predictions. The feature importance indicates
↪ the strength and direction of the relationship between the feature and
↪ the prediction; a positive feature importance suggests that as the
↪ feature value increases, so does the model's prediction or the
↪ probability of the predicted class, and vice versa for a negative
↪ importance.

The inequality sign (> or <) specifies the direction of the threshold
↪ that the feature value must cross to impact the prediction in the
↪ manner indicated by the feature importance. For instance, Age > 30:
↪ +1.5 suggests that being over 30 years old has a positive influence on
↪ the model's prediction with a magnitude of 1.5. The threshold is the
↪ boundary value that the feature must exceed or fall below to affect
↪ the prediction as determined by the sign of the feature importance.

When interpreting this list, it is crucial to understand that the
↪ features are ranked from the most impactful to the least based on the
↪ absolute values of the feature importance. This ranking directs
↪ analysts to prioritize their focus on the features at the top of the
↪ list, which have the largest absolute values, as these are the ones
↪ that most strongly drive the model's predictions. The magnitude of
↪ these values signifies the strength of each feature's influence,
↪ irrespective of whether this influence is positive or negative.

It is also essential to note that the [feature] [inequality sign]
↪ [threshold] component of the output conveys the actual condition or
↪ state of the feature in the instance being explained. This aspect
↪ provides the specific context in which the feature contributes to the
↪ model's prediction, detailing the precise nature of its impact. It is
↪ the combination of the feature's condition, the inequality sign, and
↪ the magnitude of the feature importance that offers a comprehensive
↪ view of how the model arrives at its predictions.

In essence, the LIME output should be carefully examined by taking into
↪ account both the direction indicated by the feature importance sign
↪ and the feature's state as described by its relationship with the
↪ threshold. This close examination ensures accurate interpretations,
↪ which is imperative for model transparency and for stakeholders who
↪ rely on these interpretations for decision-making. Features with
↪ larger absolute importance values, particularly those at the top of
↪ the list, merit a deeper examination due to their substantial role in
↪ influencing the model's output.

In essence, the LIME output should be carefully examined by taking into
↪ account both the direction indicated by the feature importance sign
↪ and the feature's state as described by its relationship with the
↪ threshold. This close examination ensures accurate interpretations,
↪ which is imperative for model transparency and for stakeholders who
↪ rely on these interpretations for decision-making. Features with
↪ larger absolute importance values, particularly those at the top of
↪ the list, merit a deeper examination due to their substantial role in
↪ influencing the model's output.

The interplay of these elements-feature importance, threshold values, and
↪ inequality signs-paints a detailed picture of the predictive landscape
↪ for a particular instance. Understanding this interplay is vital for
↪ extracting meaningful and actionable insights from LIME, ensuring that
↪ the focus is placed on the most relevant features that have the most
↪ significant impact on the model's decisions.

Your job now is to describe why the task is easy or hard in simple terms
↪ in one SHORT sentence based on the given list of features.

You need to mention around at least three features, preferrably from both
↪ sides. You should NOT omit the ones with the most influence. In
↪ interpreting the raw features, you should look very very closely at
↪ the meaning of each feature provided to you. Don't make up the meaning
↪ of the features, always consult the table. Also, if the feature
↪ meaning is hard to understand for users, you should find a better way
↪ for explaining it.

Remember to keep the explanations straightforward and avoid technical
↪ jargon, including the raw feature itself and any numerical values.
↪ This is meant to be shown directly to the users in the interface, so
↪ you have to be very very concise and have no redundancy in the output
↪ sentence. There's no need to emphasize again why the task is easy or
↪ hard. Before outputting the sentence, you need to think about whether
↪ the features actually make the task harder or easier. Don't output any
↪ contradicting features. Don't add anything else, such as unnecessary
↪ adjectives. Don't speculate anything, such as the user being busy,
↪ etc. Avoid using comparative words, such as "fewer", because it is not
↪ grounded.

For your information, here's what each feature means:
PRCP means precipitation
SNOW means snowfall
SNWD means snowdepth
TMAX means max temperature
TMIN means min temperature
AWND means average wind
EVAP means evaporation
WDMV means wind movement
recipient_lon means recipient longitude
recipient_lat means recipient latitude
donor_lon means donor longitude
donor_lat means donor latitude
total_quantity means the quantity of food in this donation
user_lon means user (volunteer) longitude
user_lat means user latitude
user2donor means straight-line distance between user and donor
user2recipient means straight-line distance between user and recipient
donor_exp means how long has it been since the donor signed up on the
↪ platform (probably in seconds)
recipient_exp means how long has it been since the recipient signed up on
↪ the platform (probably in seconds)
user_exp means how long has it been since the user signed up on the
↪ platform (probably in seconds)
user_rating means average rating of past rescues **given** to the rescues
↪ by the user, indicating how good the rescue experience was for the
↪ user (usually a low rating means the user experienced frustation
↪ previously); describe this as frustration or satisfaction, or perhaps
↪ mixed, for the previous rescues
recipient_rating means average rating provided by other users for rescue
↪ trips with this recipient, where the rating indicates how good the
↪ rescue experience was for the user when delivering to this recipient
donor_rating means average rating provided by other users for rescue
↪ trips with this donor, indicating how good the experience was for the
↪ user when picking up from this donor
pub_Y means year of rescue publication
pub_M means month of rescue publication
pub_D means day of rescue publication
pub_H means hour of rescue publication
donor_counts means how many rescues has the donor completed previously
recipient_counts means how many rescues has the recipient completed
↪ previously
user_counts means how many rescues has the user completed previously,
↪ higher means more experience

[Top 10 Features from LIME]

Replace "user" with "you" in second person since this sentence will be
↪ directly displayed to the user. Remember, in total, you have to
↪ mention around three to four most influential features! Use very very
↪ simple words and sentences so that the user can understand it quickly
↪ with a glimpse. Don't forget to make the sentence more fluent. Also
↪ avoid using vague words, like "certain", "some". Avoid ANY redundancy
↪ to keep the sentence short.

Complete this: this task is {result} for you because [MASK]
Optionally, you can add: but it is hard/easy because [MASK], but should
↪ not contain any redundancy with the first part.
```

Listing 3: Full Prompt provided to the LLM to generate natural language explanation

```
Now create clear, distinct tags by combining an adjective and noun phrase
↪ to explain why a task is easy or hard, given a specific context. Your
↪ response should consist of tags separated by commas. Ensure that each
↪ tag is unambiguous and conveys the required information without
↪ duplication. Additionally, follow the template suggestions provided
↪ below for specific feature types. If the tag's raw feature is listed,
↪ adhere to the template suggestion. If the feature is not listed,
↪ create a suitable tag.

For 'user_counts' related features, use: "{hard or easy} for
↪ {less/moderate/more/the most} experienced"
For 'user_rating' related features, use: "prior frustration or prior
↪ great experience"
For 'user_exp' related features, use: "{hard or easy} for users who've
↪ been on the platform for {less/moderate/more/the} most time"
```

Listing 4: Full Prompt provided to the LLM to generate tags

B DESIGN CONCEPTS OF POSSIBLE INTEGRATION METHOD

We test six different AI-integration methods as design concepts, across different levels of backend scaffolding and frontend information display. There are three levels of backend scaffolding: A) Low: showing all tasks to new volunteers on the map, and sending notifications of all tasks to them; B) Medium: showing all tasks to new volunteers on the map, but customizing notifications by only

Demographic Information	Participant Counts or Statistics
Race	African American (4), White (4), Asian descent (1), Middle Eastern descent (1)
Age	Mean: 39.6, Maximum: 65, Minimum: 24
Gender	Female (4), Male (6)
Number of tasks done on Food Rescue Hero (volunteers only)	Mean: 34.5, Maximum: 200, Minimum: 1
Months in Food Rescue Hero Volunteer Tenure (volunteers only)	Mean: 11.3, Maximum: 60, Minimum: 1
Primary job (volunteers only)	Software engineer (3), Graphic/UI designer (2), Retired (3)
Education level (volunteers only)	Master's degree (8)
Years in Food Rescue Hero Administration Tenure (admins only)	4(1), 1.5(1)

Table 5: Aggregated participants' self-reported demographics

sending easy tasks to them; C) High: only showing easy tasks to new volunteers on the map, and customizing notifications by only sending easy tasks to them. There are two ways of frontend display: 1) displaying difficulty levels on screen, and 2) not displaying difficulty levels on screen. Combining the three backend scaffolding levels with the two frontend display method yields six different design concepts in total: A.1, A.2, B.1, B.2, C.1, C.2. We then capture these six design concepts with six storyboards to make them accessible to our study participants, as shown in Table 4.

Front-end	Back-end Scaffolding	Design Concept of Each Storyboard
No display	Low-level	A.1: No display, showing all tasks on the map, and sending notifications of tasks of all difficulty levels
Display	Low-level	A.2: With display, showing all tasks on the map, and sending notifications of tasks of all difficulty levels
No display	Medium-level	B.1: No display, showing all tasks on the map, and customizing notifications by only sending easy tasks
Display	Medium-level	B.2: With display, showing all tasks on the map, and customizing notifications by only sending easy tasks
No display	High-level	C.1: No display, only showing easy tasks on the map, and customizing notifications by only sending easy tasks
Display	High-level	C.2: With display, only showing easy tasks on the map, and customizing notifications by only sending easy tasks

Table 4: Six design concepts represented in the storyboards

C PARTICIPANT DEMOGRAPHICS

We present aggregated information about participant self-reported demographics in Table 5.