Critiquing-based Modeling of Subjective Preferences

Medlar, Alan

ACM
2022-07

unspecified
publishedVersion

Critiquing-based Modeling of Subjective Preferences

Alan Medlar
University of Helsinki
Finland
alan.j.medlar@helsinki.fi

Jing Li
University of Helsinki
Finland
jing.li@helsinki.fi

Yang Liu
University of Helsinki
Finland
yang.liu@helsinki.fi

Dorota Głowacka
University of Helsinki
Finland
dorota.glowacka@helsinki.fi

ABSTRACT

Applications designed for entertainment and other non-instrumental purposes are challenging to optimize because the relationships between system parameters and user experience can be unclear. Ideally, we would crowdsource these design questions, but existing approaches are geared towards evaluation or ranking discrete choices and not for optimizing over continuous parameter spaces. In addition, users are accustomed to informally expressing opinions about experiences as critiques (e.g. it's too cold, too spicy, too big), rather than giving precise feedback as an optimization algorithm would require. Unfortunately, it can be difficult to analyze qualitative feedback, especially in the context of quantitative modeling.

In this article, we present *collective criticism*, a critiquing-based approach for modeling relationships between system parameters and subjective preferences. We transform critiques, such as "it was too easy/too challenging", into censored intervals and analyze them using interval regression. Collective criticism has several advantages over other approaches: "too much/too little"-style feedback is intuitive for users and allows us to build predictive models for the optimal parameterization of the variables being critiqued. We present two studies where we model: (i) aesthetic preferences for images generated with neural style transfer, and (ii) users' experiences of challenge in the video game Tetris. These studies demonstrate the flexibility of our approach, and show that it produces robust results that are straightforward to interpret and inline with users' stated preferences.

CCS CONCEPTS

• **Human-centered computing** → **User models**; **User studies**.

KEYWORDS

critiquing, subjective preferences, user experience, randomized experiments

---

1 INTRODUCTION

User experience is an increasingly important consideration in systems, services and products, with many applications emphasizing various aesthetic, affective and hedonic properties [22, 23]. Such qualities are particularly important in applications with no instrumental purpose, such as video games, digital toys and interactive artwork [9, 36, 40]. Unfortunately, these applications can be challenging to optimize as they can contain numerous parameters that impact user experience, but for which there are no objective quantities to be maximized and, therefore, need to be set on the basis of subjective preferences. In video games, for example, character movement speed has a significant impact on user experience: too slow and navigating the game world can be tedious, but too fast and it can become difficult to control [40]. The sweet spot that balances these two extremes will be determined by subjective preferences that are influenced by contextual factors (e.g. larger characters tend to move slower) and other human factors (e.g. expert players might prefer a faster, more challenging game) [40].

How should we go about setting such parameters? A single user (e.g. the developer) could simply set a given parameter to whatever feels intuitively correct. Alternatively, multiple users could be recruited to provide feedback via questionnaires or to state their preference for one parameterization over another (i.e. pairwise comparisons). Lastly, we could make assumptions about how user behavior relates to experience, e.g. we could assume that engagement (time spent) correlates with positive user experiences. Unfortunately, all these approaches have downsides to inferring subjective preferences: a single user is unlikely to be representative of the group, questionnaire response scales need to be validated [31] and require large sample sizes per parameterization. Pairwise comparisons do not scale, with the number of comparisons increasing exponentially with the input size [34]. Finally, behavioral data can be misinterpreted; correlating with negative experiences as well as positive ones [43].

In this article, we present a critiquing-based approach for modeling relationships between system parameters and subjective preferences called *collective criticism*. We consider a crowdsourcing

scenario where a group of individuals are presented with a system or product and asked to critique their experiences in terms of a given attribute. These critiques take the form of retrospective summaries, such as "the food was too spicy/bland" after eating a meal, or "the weather was too hot/cold" when describing a vacation [13, 33]. Such feedback is qualitative, but we can extract quantitative information by considering the context within which the feedback was given. Using the video game example from before, if character movement is "too slow" and the speed was set to x, then this implies that the optimal speed lies in the interval from x to $+\infty$ (strictly speaking, the right-censored interval $(x, +\infty]$). Importantly, users do not need to know anything about how the game was implemented to give their critiques: they are based solely on their experiences. This method of transforming critiques into intervals allows us to model relationships between user experience and system parameters using interval regression. We demonstrate the generality of this approach using two studies related to images generated using neural style transfer and users' experience of challenge after playing the video game Tetris.

At present, our approach cannot guard against abuse from bots and disinterested participants on crowdsourcing platforms like Amazon Mechanical Turk. Instead, we focused on groups of users that may not have any particular expertise, but are assumed to provide honest feedback. Examples of such groups include beta testers for a video game that is currently under development or a design community that meets to critique one another's work. To replicate these scenarios, we present studies performed in uncontrolled environments where participants were informally recruited by interrupting their daily lives, e.g. during a conference coffee break or while studying in the university library. The main contributions of this paper are as follows:

- A novel critiquing-based approach for modeling relationships between system parameters and subject preferences called *collective criticism* that combines randomized tasks with summary retrospective feedback.

- A modeling approach that transforms critiques into censored intervals to be analyzed using statistical packages for interval regression.

- We present two case studies using collective criticism to model (i) aesthetic preferences for images generated using neural style transfer, and (ii) user experiences of challenge after playing the video game Tetris.

2 RELATED WORK

In this section, we review related work on crowdsourcing preferences and how critiquing has been used as an interaction mechanism in different types of information systems.

2.1 Crowdsourced Preferences

Crowdsourcing has eased the collection of subjective preferences by reducing cost and turnaround time, while showing a high degree of agreement with data collected in controlled laboratory environments [8, 14, 42]. In computing, crowdsourcing has been used to collect subjective preferences for algorithm development and evaluation. In particular, it has been used to collect relevance judgments

[1], data for sentiment analysis [6], assessments of toxicity in online discussions [3] and even examples of irony and sarcasm [19]. In these areas, items are scored independently of one another using either binary (e.g. relevant/not relevant) or ordinal (e.g. negative to positive) labels. While these labels are considered subjective, there is assumed to be a consensus within a given culture or community. As this data tends not to be aggregated, but used as training data, each label needs to be correct and, therefore, there is an extensive literature on study design and statistical methods for quality control (for a recent survey, see [26]).

In other domains, crowdsourcing is used to understand the opinions and aesthetic preferences of the general public. It has been used, for example, to study the aesthetics of platforming games [38], 3D models [16] and portrait photography [17]. Crowdsourcing aesthetic preferences has seen extensive use in reconstructive and cosmetic surgery. In reconstructive surgery, it has been used to compare the aesthetics of cleft lip outcomes [42] and the results of different surgical techniques [39]. Whereas, in cosmetic surgery, it has been used to assess buttock augmentation outcomes [46] and to characterize anatomical aesthetic preferences for male [32] and female [20] genitalia. The use of crowdsourcing is considered important as surgical aesthetic outcomes are usually only assessed by an individual, either the patient or surgeon, which can lead to biased assessments [5]. In this article, we include a study comparing the aesthetic preferences of an individual (in our case, a developer) with our approach and find similar disparities. Beyond aesthetics, crowdsourcing has recently been used to understand the public perception of topical subjects, such as AI fairness [45] and moral decision-making in the context of autonomous driving [4]. All of the above examples used either questionnaires or pairwise comparisons to infer subjective preferences, both of which have weaknesses. Questionnaire response scales need to be validated to ensure they are measuring what they purport to measure [31]. Pairwise comparisons do not suffer from this issue, but can require exceptionally large sample sizes (i.e. multiples of $\binom{n}{2}$ pairwise comparisons), practically limiting assessment to a relatively small number of items [34]. Our approach does not require validation like a questionnaire because we only ask a single question that is often directly referencing a given parameter (i.e. we do not use multi-item scales, which would require us to assess construct validity). Furthermore, as users are critiquing individual parameters and not making pairwise comparisons, sample sizes can be much lower.

While not usually considered crowdsourcing, A/B testing uses randomized experiments to compare the effectiveness of two (or more) versions of the same system on the basis of conversion rates, e.g. click-throughs or purchases [28]. This makes the assumption that a given behavior is correlated with positive user experience [29]. However, it can also result in unintended consequences if the measured response corresponds to multiple outcomes, e.g. video watch time is correlated with outrage as well as enjoyment [43]. Our approach is based on stated preferences, avoiding the ambiguity-related issues associated with inferring revealed preferences from behavioral data.

2.2 Critiquing

Critiquing has previously been used as an interaction mechanism in both interactive search and conversational recommender systems [13]. In critiquing-based systems, users provide critiques in relation to item features, e.g. "too expensive", to iteratively navigate a complex information space [13]. The FindMe system was the first critiquing recommender, combining browsing with the critiquing of previously retrieved examples [11]. Later systems utilized the approach to develop interactive retrieval systems for e-commerce [10, 18], decision support [35] and preference-based search [47]. However, the most common application of critiquing is in conversational recommender systems [24], where they have been applied to various domains, including movie [48] and music recommendation [25]. Recently, critiquing-based systems have used language-based attributes, rather than fixed item features, to automatically identify attributes of an item that can be critiqued [49].

In interactive systems, critiques are usually conceptualized as constraints that are applied across items [10, 18, 35, 47]. For example, if a plane ticket is critiqued as too expensive, it does not make sense to recommend tickets with higher prices. In this work, we are concerned with modeling user preferences, so critiques are modeled probabilistically even though they may be constraints from the perspective of individuals.

3 COLLECTIVE CRITICISM

Collective criticism is a critiquing-based approach for modeling the relationships between system parameters and subjective preferences. Each study based on collective criticism requires three elements: (i) critique elicitation, (ii) randomized tasks, and (iii) statistical modeling to analyze user preferences.

3.1 Critique Elicitation

During a study, each participant performs a task where they interact with a system or product for a short period of time. After completing the task, participants are asked to critique the property under study using summary retrospective feedback. In concrete terms, summary retrospective feedback takes the form of judgments, such as "too much" or "too little" of a given property. For example, suppose we wanted to optimize the volume of an audible, but discreet, ringtone for an office environment. We would ask participants to listen to a ringtone and state whether they thought it was "too quiet" to be audible or "too loud" to be discreet.

This kind of critique is qualitative: it does not contain information related to how much something should change, merely the direction of that change. This allows participants to respond with their gut instinct and can be used when an appropriate scale for quantitative feedback does not exist.

3.2 Randomized Tasks

Here, we detail the assumptions of our approach and describe the steps necessary to design and conduct an study.

3.2.1 Assumptions. We assume that investigators have a hypothesis that a given parameter, p, affects a specific property of the system under study. Furthermore, we assume that optimizing p necessitates making a trade-off, i.e. setting p either too high or too

low is detrimental to user experience, but that there is a "sweet spot" where the average user is maximally satisfied. We do not assume that there exists a single optimal parameterization for all users as this depends on the experimental design of a given study. Indeed, we provide a worked example in Section 3.4 that assumes a single optimal parameterization and two further case studies in Sections 4 and 5 where p is modeled in terms of other explanatory variables.

3.2.2 Effective Parameter Ranges. Investigators need to determine the effective range for the parameter, p. This could be the entire range of the parameter, e.g. the decision threshold in a probabilistic classifier is 0 to 1 inclusive, or be limited to a given interval. In this article, we determined the effective range of parameters by trial and error, however, it could also be limited due to physiological reasons, e.g. human hearing is limited to 20 Hz to 20 kHz, or technological reasons, e.g. telephony limits audio frequencies to 300 Hz to 3.4 kHz. Therefore, optimizing the frequency of a tone would have different effective ranges under different circumstances.

3.2.3 Summary Retrospective Anchors. Study participants give critiques using summary retrospective feedback guided by the investigator. The study design, therefore, needs to include verbal anchors to ensure that users critique the correct property. In general, verbal anchors are words used to indicate the informal meaning of response scales, such as "strongly agree" and "strongly disagree". In our case, anchors are judgments, such as "too hot" and "too cold".

Selecting appropriate verbal anchors is important for two reasons: (i) from the participants' perspective, anchors need to capture their collective understanding of the extremes of the property being assessed, and, (ii) from the investigator's perspective, anchors need to correspond to increasing and decreasing the parameter being optimized.

3.2.4 Procedure. We extract quantitative information from summary retrospective feedback by changing the underlying conditions from which the assessment is made. We achieve this by randomizing the parameter of interest within the parameter's effective range. The procedure is as follows:

(1) We randomly set the parameter, p, to a value selected uniformly at random from the parameter's effective range.

(2) Participants are instructed to perform a study-specific task. Participants could be asked to use a system for a given period of time or to simply look at an image.

(3) After completing the task, participants are asked to critique their experience with respect to a given property using the study's summary retrospective anchors (e.g. too high/too low).

(4) For each observation, we record the random value of p, the participant's critique and any additional metadata or user behavior data that is to be used for modeling (see Section 5 for an example).

If we want to understand the impact that other parameters in the system have on p (i.e. interactions), then these additional parameters also need to be randomized in Step 1 and recorded along with p in Step 4 (see Section 4 for an example).

3.3 Statistical Modeling

After all the tasks have been performed, we model the data set using interval regression. This requires us to transform participants' critiques into censored and/or non-censored intervals. We use left-censored intervals to represent when participants stated a parameter was set too high. That is, if the parameter being optimized was assigned the random value p, then the resulting censored interval is $(-\infty, p]$, i.e. while we do not know the optimal value that would maximize the participant's experience, we assume that it is in the interval up to and including p. If the effective range of this parameter, however, is such that p cannot be negative, then the (non-censored) interval would be $[0, p]$. We use right-censored intervals, $[p, +\infty)$, when the parameter was set too low, following the same logic. Figure 1 shows the intervals from the worked example in Section 3.4, where left-censored intervals are depicted as red arrows and right-censored intervals as blue arrows.

In interval regression, we let $y = \beta\mathbf{X} + \epsilon$, where y is a continuous response variable and errors are assumed to be Gaussian, i.e. $\epsilon \sim \mathcal{N}(0, \sigma^2)$, where σ is the standard deviation. In the general case, the log likelihood is composed of four terms:

$$lnL = -\frac{1}{2} \sum_{j \in \mathcal{C}} \left\{ \left(\frac{y_j - \mathbf{X}\beta}{\sigma} \right)^2 + log(2\pi\sigma^2) \right\}$$
$$+ \sum_{j \in \mathcal{L}} log\ \Phi \left(\frac{y_{\mathcal{L}_j} - \mathbf{X}\beta}{\sigma} \right)$$
$$+ \sum_{j \in \mathcal{R}} log \left\{ 1 - \Phi \left(\frac{y_{\mathcal{R}_j} - \mathbf{X}\beta}{\sigma} \right) \right\}$$
$$+ \sum_{j \in \mathcal{I}} log \left\{ \Phi \left(\frac{y_{2_j} - \mathbf{X}\beta}{\sigma} \right) - \Phi \left(\frac{y_{1_j} - \mathbf{X}\beta}{\sigma} \right) \right\},$$

where $\Phi(\cdot)$ is the cumulative normal function. Each term considers one of the four types of observation: point observations ($\mathcal{C}$), left-censored intervals ($\mathcal{L}$), right-censored intervals ($\mathcal{R}$) and intervals with two end-points ($\mathcal{I}$), respectively [2]. We note that, in our case, $\mathcal{C}$ is empty as we have no point observations, but a task could allow participants to respond that p is optimal. In our experience, however, participants do not insist that parameterizations are optimal due to a lack of experience with the system under investigation. Throughout this article, we used the R Survival package to fit interval regression models [41].

3.4 Worked Example

We present a toy cognitive estimation task to demonstrate the simplest concrete example of collective criticism.

3.4.1 Objective: We placed 568 jelly beans in a jar and used collective criticism to estimate the quantity of jelly beans. We determined empirically that the jar could hold ∼1000 jelly beans, making the effective range from 0–1000, inclusive. We used "greater than" and "less than" as summary retrospective anchors.

3.4.2 Task: For each participant, a number x was sampled uniformly at random from 0-1000, inclusive. Participants were handed the jar of jelly beans and asked the following question:
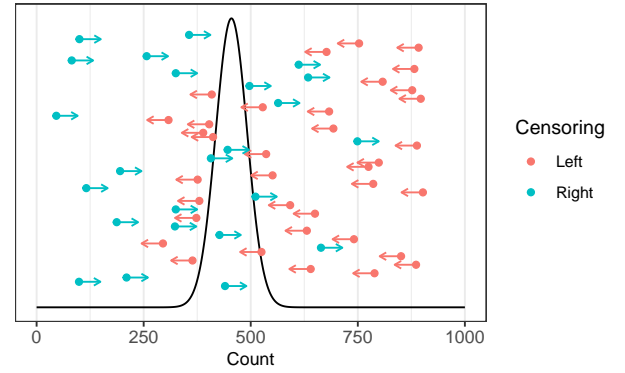
Figure 1: Interval data and final result from worked example in Section 3.4. Left-censored intervals are colored red and right-censored intervals are blue. The y coordinate of each interval is the order tasks were performed in.

> *How many jelly beans are in the jar:* **greater than x**
> *or* **less than x**?

As follow-up questions, we asked participants to freely estimate the number of jelly beans in the jar, and whether they felt critiquing or freely estimating the same quantity was more cognitively demanding.

3.4.3 Participants: We recruited 60 participants during the coffee breaks of a conference organized in our department (27 female, 33 male). The participants ranged from PhD students to full professors who had a background in theoretical computer science, optimization or a related field.

3.4.4 Results: The 60 participants gave 60 critiques and 60 estimates of the number of jelly beans. No data points were excluded from analysis. The difference between participants' mean estimate of the number of jelly beans (M = 465.52, 95% CI [409.98, 521.05]) and the answer derived from collective criticism (M = 454.96, 95% CI [382.97, 526.95]) was not statistically significant (t(62.56) = -0.260, p = 0.796), showing that our approach is as accurate as allowing users to directly estimate. Furthermore, a majority of participants (47/60, $p = 1.22 \times 10^{-5}$, binomial test) reported that they perceived critiquing to be less cognitively demanding than free estimation.

4 STUDY 1: AESTHETIC PREFERENCES IN NEURAL STYLE TRANSFER

In our first study, we used collective criticism to model users' aesthetic preferences for images generated using neural style transfer [21]. Neural style transfer combines the content from one image (the content image) and the style from another image, usually an artwork (the style image), see Figure 2 for an example. We demonstrate how collective criticism can be used to elicit preferences, model different hypotheses and make practical parameterization decisions.
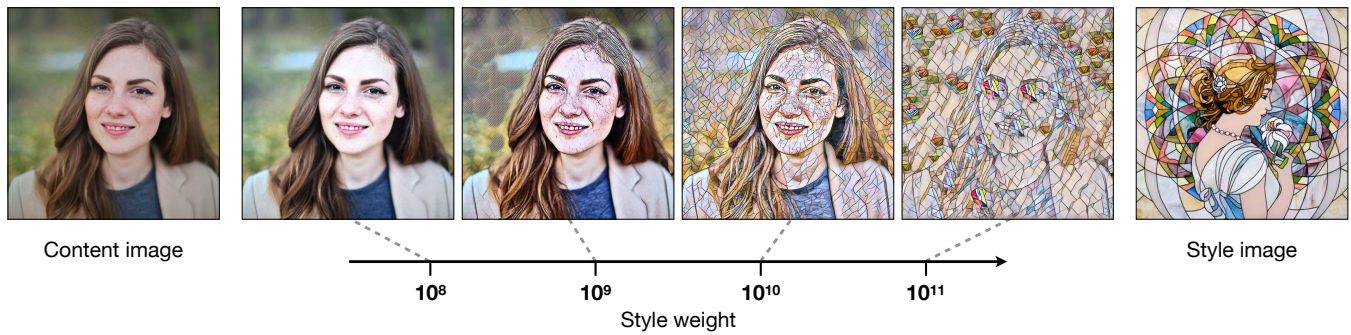
Figure 2: Neural style transfer combines a content image (far left) with a style image (far right). The degree of stylization is controlled by the style weight parameter (middle) which can vary from barely noticeable (10^8) to unrecognizable (10^{11}).

4.1 Objective

Neural style transfer has two hyperparameters: a content weight and a style weight, however, if one parameter is kept constant, there is only one free parameter. We wanted to identify the highest style weight that could be applied to a photo without the subject becoming unrecognizable. Furthermore, we hypothesized that different style weights would be optimal for different kinds of photo, such as head, waist-up and full body shots, i.e. we hypothesized that there is an interaction between style weight and photo type.

4.2 Neural Style Transfer

4.2.1 Implementation: We used the fast neural style transfer implementation included in the PyTorch library[1] that is based on perceptual loss [27] and instance normalization [44].

4.2.2 Model Training: We determined by trial and error that, if the content weight is kept constant at 10^5, the effective range for style weight is 10^8-10^{11}, where higher values result in an output image more heavily influenced by the style image (see Figure 2). We trained 101 neural style transfer models using the COCO 2014 data set [30] and used an image of a mosaic included with PyTorch as the style image (see Figure 2, far right). Each model had a different style weight parameter where the exponent was incremented by 0.03, i.e. 8.0, 8.03, … 10.97, 11.0. This increment was chosen empirically so the difference between images generated by consecutive models was imperceptible. Each model was trained for 2 epochs.

4.2.3 Content Images: We selected photos from a collection of permissively licensed stock photographs[2] to use as content images. We selected three categories of portrait: head shots, waist-up and full body shots. We identified 39 images of similar size, with 13 photos in each of the three categories. All categories included men and women of approximate working age from different ethnicities.

4.3 Task

Participants were told we were creating a new website for our research group and wanted to make our photos look more interesting using neural style transfer. We briefly explained the concept of neural style transfer using example images to illustrate different

levels of stylization and stated that we wanted output images to be as strongly influenced by the style image as possible without the identity of the person in the content image becoming unrecognizable. Each participant was shown 10 randomly sampled images, stylized with randomly sampled style weights. After being shown each image, participants were asked the following question:

> *Do you think the image should look **more realistic** or **more artistic**?*

During the study, we logged the photos shown including the photo category, style weights and user critiques. Each experiment lasted a total of ~3 minutes.

4.4 Participants

We recruited 31 participants from the Faculty of Science by walking up to people in the corridors of the Department of Computer Science and the Department of Mathematics and Statistics (10 female, 21 male). All participants were PhD students or postdoctoral researchers.

4.5 Results

The 31 participants examined a total of 310 photos: 95 head shots, 117 waist-up shots and 98 full body shots.

4.5.1 Baseline: There are no validated questionnaires for rating image stylization and pairwise comparisons between all images using all models would require an impractical sample size. Instead, we used a pre-trained model included in the PyTorch distribution as a baseline to understand whether the developer's original parameterization was sufficient to solve this problem. The pre-trained model used a content weight of 10^5 (the same as our models) and a style weight of 10^{10}.

4.5.2 Preference Models: We fitted two preference-based models using collective criticism. In the first model, each study participant was modeled as a random effect (due to repeated measures) and photo type (head, waist-up or full body) was modeled as a fixed effect:

$$y_{ij} = \beta_0 + \beta_1 p_{ij} + u_j,$$

---

[1]https://github.com/pytorch/examples/tree/master/fast_neural_style
[2]https://www.pexels.com/license/

where y are intervals derived from critiques, p is the photo type and u are random intercepts per user, j. The second model was identical to the first, but with the photo type term excluded:

$$y_{ij} = \beta_0 + u_j$$

The style weights from the first model (log_{10}) for head (M = 9.71, 95% CI [9.46, 9.95]), waist-up, (M = 9.53, 95% CI [9.31, 9.75]) and body shots (M = 9.39, 95% CI [9.14, 9.63]) were very similar to one another with overlapping 95% confidence intervals. Indeed, the difference in model fit between the two models was not statistically significant (χ^2 (1.55, N = 310) = 3.14, p = 0.14). This suggests that the style weight from the second model (M = 9.55, 95% CI [9.36, 9.73]) is suitable for all three types of photo, assuming all other conditions, such as photo size and the age range of the subject, remain constant. Finally, the style weight parameter used in the pre-trained baseline was 10^{10}, which was outside of all four confidence intervals and, therefore, the difference was statistically significant.

4.6 Summary

Given these findings, we argue that there is insufficient evidence for using different style weights for each photo type and that the style weight used to train the neural network should be set to $10^{9.55}$ in order to balance style and recognizability. However, if we were to collect more data, then it is likely that the confidence intervals would have been narrower and we could recommend the use of category-specific style weights. This experiment demonstrates how the preferences of an individual (in this case, the original developer) may not match the aesthetic preferences of the group for a specific problem. This case study only looked at group preferences, but could be extended with additional explanatory variables from a user profile to predict personalized style weights.

5 STUDY 2: PREDICTION OF CHALLENGE PERCEPTION IN TETRIS

In our second study, we show how collective criticism can be used to model users' perception of the challenge experienced while playing the video game Tetris. This example demonstrates how to create a model that could be used as the basis for personalization in an adaptive interactive system.

5.1 Objective

Tetris is a tile-matching video game that gets progressively more challenging as the speed of the game increases (see Section 5.2 for a description of gameplay). A game of Tetris starts out slow, not presenting the player with any challenge. During the endgame, however, Tetris can become frustratingly fast, making the player anxious (see Figure 3). We wanted to create a model for an adaptive version of Tetris where the level of challenge is personalized for each player. Namely, we wanted to identify how game speed and other contextual factors could be used to keep players feeling challenged, but not overwhelmed. This could be viewed as modeling a kind of flow state [15]: one of many considerations that go into game design [7] and is actively studied in the development of slot machines [37].
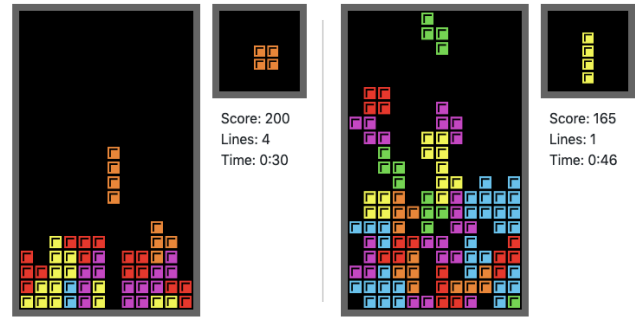
Figure 3: Two screenshots of Tetris: the left image shows the slow (boring) early game, whereas the right image is typical of the faster (stressful) endgame.
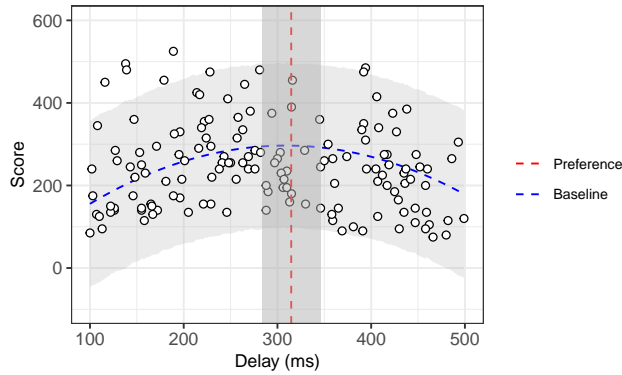
5.2 Tetris

5.2.1 Gameplay: In Tetris, players control falling shapes called tetrominoes to achieve a high score. Players can move tetrominoes left and right, increase their speed of descent (called a "soft drop") or force them to immediately drop to the bottom of the play field (a "hard drop"). When the player completes a line (i.e. a row of blocks without any gaps), it disappears and the player's score increases. After clearing a fixed number of lines, the difficulty level is increased and, along with it, the speed of the game. The game ends when a tetromino overlaps the top of the play field.

5.2.2 Implementation: Figure 3 shows our web-based implementation of Tetris. The play field is 12×20 tiles and the surrounding interface shows the score, the number of lines cleared and a timer indicating how much time is remaining in the task. In most versions of Tetris, the scoring function is proportional to the level and, therefore, the current game speed, i.e. clearing a line is worth more points at level 2 than level 1. In our implementation, however, we used the same scoring function irrespective of the current game speed: 5 points for each placed tetromino and 20 points for each line cleared. The game speed is determined by the delay in milliseconds for a tetromino to move down the play field by one block. We determined by trial and error that the effective range of this delay was 100-600ms, where 100ms gives the fastest speed and 600ms results in the slowest speed.

5.3 Task

Prior to the task, participants were asked to fill out a background questionnaire to capture (i) demographic information, (ii) how often they played video games, (iii) their familiarity with Tetris and (iv) their opinion of Tetris. After completing the questionnaire, participants were allowed to play as many warm-up games of Tetris as they wanted.

During the study, each participant played 3 games of Tetris. In the first game, the fall delay was sampled uniformly at random from the full effective range, 100-600 inclusive. In the second and third games, we altered the upper or lower bounds of the delay range to reflect user feedback. For example, if in the first round a delay of 300 was too slow, then in the second round the delay would be sampled from the range 100-300 (lower delays mean higher speeds). Each

Figure 4: Using player preferences of tetromino fall delay comes to similar conclusions as a quadratic model based on scores. Gray shaded region is the 95% confidence interval.

Figure 5: Predicted delays from adaptive preference model. Higher number of lines cleared, Tetris opinion and familiarity predict lower delays are necessary to provide a challenge.

game of Tetris lasted a maximum of 2 minutes. After 2 minutes had expired or the game was lost, participants were asked the following question:

> We are collecting data to create a game of Tetris that helps players improve their skill level. It should be just fast enough to feel like a challenge. For a player of your skill level, do you think the game you played should have been **faster** or **slower**?

During the study, we logged user critiques, game data (fall delay, time spent playing, score, number of lines cleared) and interaction data (keyboard events and their associated timestamps). Each task took a total of ~10 minutes.
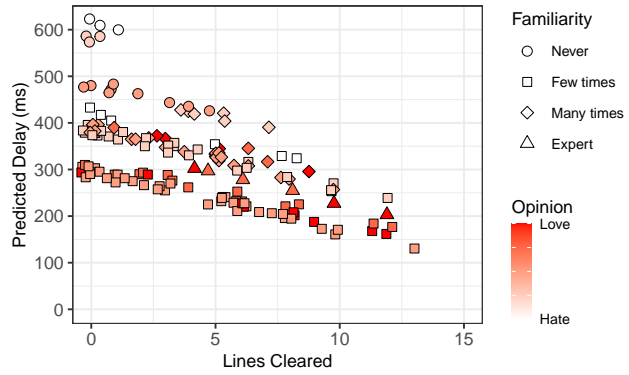
5.4 Participants

We recruited 50 participants who were studying at the university library (24 female, 26 male). Participants were aged between 20-49 with a median age of 28. According to the background questionnaire, over 2/3 of participants played video games at least occasionally (never (14), occasionally (23), every week (9), every day (4)), and a majority of participants, 45/50, had at least some prior experience of playing Tetris (never (5), a few times (31), many times (12), experienced (2)). Overall, participants were neutral in their opinion of Tetris (mean = 2.98 on a 5 point scale where 1 = hate and 5 = love). One participant stated that they hated Tetris, despite having never played the game.

5.5 Results

The 50 participants played a total of 150 games of Tetris. We compared a baseline model that was fit using behavioral data with two models created using collective criticism.

5.5.1 Baseline: We assumed that the fall delay that maximizes the average score would also maximize players' perception that the game was just challenging enough (we are really maximizing the rate of scoring as each experiment has a fixed duration). We fitted a linear mixed model with fall delay as linear and quadratic fixed

effects and participant as a random effect:

$$y_{ij} = \beta_0 + \beta_1 d_{ij}^2 + \beta_2 d_{ij} + u_j,$$

where y is the score, d is delay and u are random intercepts per user, j. The baseline found that the average score was maximized when the delay was 308.68 ms (see Figure 4, dashed blue curve). The baseline does not provide any uncertainty estimates for the optimal delay, only standard errors for the coefficients of the delay terms in the model. Furthermore, this analysis is based on assumptions and we do not know whether this delay is truly inline with user preferences.

5.5.2 Static Preference Model: We used collective criticism to fit a model for users' speed preferences with no explanatory variables other than participant as a random effect:

$$y_{ij} = \beta_0 + u_j,$$

where y are intervals derived from critiques and u are random intercepts per user, j. The preference model found that the optimal delay is 314.70 ms (95% CI [283.34, 346.05]). This is very similar to the point estimate from the baseline model, despite using different response variables and very different assumptions. Unlike the baseline, this model allows us to estimate the uncertainty in the mean delay (see Figure 4, dashed red line). As the baseline result (308.68 ms) falls into the 95% confidence interval for the preference model, there is no statistically significant difference between the two results. However, the preference model has the advantage of having evidence that the optimal delay is inline with user preferences.

5.5.3 Adaptive Preference Model: Predicting the optimal delay for the average participant masks the variability between players and we assume novice and expert players will have different ideas of what constitutes a challenge. Unfortunately, this is not possible to model with the baseline because the score–a key measure of player ability–is already being used as the response variable. However, the preference model can incorporate the score as an additional explanatory variable. We tried numerous different models and assessed model fit using AIC.

Figure 5 shows in-sample predictions from the best model we found based on AIC score (AIC = 163.65). This model used Tetris familiarity and Tetris opinion from the background questionnaire, as well as the number of lines cleared during a game as explanatory variables (using the game score had a slightly higher AIC, but the two variables were strongly correlated):

$$y_j = \beta_0 + \beta_1 s_j + \beta_2 f_j + \beta_3 o_j + u_j,$$

where y are intervals derived from critiques, s is the number of lines cleared, f is a factor for familiarity, o is a factor for opinion and u are random intercepts per user, j. Figure 5 shows that as player performance (number of lines cleared) increases, the predicted fall delay decreases to offer a greater challenge. Similarly, higher Tetris opinions and familiarity tended to correlate with lower predicted fall delay. We investigated the use of many additional variables in the model, e.g. average number of hard drops per tetromino and average time between key presses were highly predictive, however, including them in a model together with number of lines cleared resulted in higher AIC.

5.6 Summary

The difference between the baseline and the two models created using collective criticism was the incorporation of preference information, making us confident that we are modeling user experience and not potentially misinterpreting behavioral data. While the baseline model would only satisfy the average player, the additional flexibility of freeing up the score variable allows us to create a model for personalization based on player performance (as determined by the prior two minutes of playtime). Furthermore, the preference models directly predict the optimal delay parameter allowing it to be easily integrated into the game itself, whereas the baseline required us to maximize a quadratic function to calculate the optimal result.

6 DISCUSSION AND LIMITATIONS

In this paper, we introduced a critiquing-based approach for modeling the relationships between system parameters and subjective preferences called collective criticism. Collective criticism combines randomized tasks with critiques that are transformed into intervals and modeled using interval regression. We demonstrated collective criticism using two studies: (i) aesthetic preferences in neural style transfer, and (ii) modeling users' perceptions of challenge in Tetris. In neural style transfer, we showed that the optimal parameterization was different from a baseline pre-trained model, but found insufficient evidence in the data collected to support there being an interaction between style weight and photo type. In Tetris, we demonstrated that modeling subjective preferences using collective criticism allowed us to create more complex models than with behavioral data alone. Furthermore, we could do so without making assumptions about how behavior maps to preferences.

There are several limitations to our approach. First, collective criticism requires an understanding of regression modeling and how to express experimental design through model specification in order to produce meaningful results. Collective criticism is, therefore, very general, but difficult to use compared to other crowdsourcing approaches where the goal is to generate annotations or a ranking.

Second, in its present form, participants can only give feedback with respect to a single variable. While this is still very general, there may be situations where two or more variables interact to produce a given user experience, e.g. how a video game feels to play might depend on a combination of character speed and projectile speed. We are currently investigating multiple output regression methods to remedy this limitation. Third, as we stated earlier, we could not test our approach on crowdsourcing platforms, such as Amazon Mechanical Turk, where studies tend to include trick questions to guard against abuse from bots and disinterested participants. Given the subjective nature of the experiences we are aiming to model and the limited responses used in task design, it is unclear how investigators would ensure data integrity other than simply increasing the sample size and assuming a majority of participants are well-intentioned. This too, will be a topic for future work. Lastly, as we can see in Figure 1, asking users to critique experiences that are either very high or very low is wasteful after a certain amount of evidence has already accumulated. We plan to investigate combining collective criticism with Thompson sampling [12], where the next data point is sampled from the posterior distribution, so the effective parameter range gets narrower as the experiment proceeds and participants will only be asked to critique the areas of greatest uncertainty. Thompson sampling has the disadvantage that you need to specify the model a priori, which could lead to sample inefficiency if the model is unnecessarily complex. For well-defined problems, however, this should not be an issue.

ACKNOWLEDGMENTS

REFERENCES

[1] Omar Alonso, Daniel E Rose, and Benjamin Stewart. 2008. Crowdsourcing for relevance evaluation. In *ACM SigIR Forum*, Vol. 42. ACM New York, NY, USA, 9–15.
[2] Takeshi Amemiya. 1973. Regression analysis when the dependent variable is truncated normal. *Econometrica: Journal of the Econometric Society* (1973), 997–1016.
[3] Lora Aroyo, Lucas Dixon, Nithum Thain, Olivia Redfield, and Rachel Rosen. 2019. Crowdsourcing subjective tasks: the case study of understanding toxicity in online discussions. In *Companion proceedings of the 2019 world wide web conference*. 1100–1105.
[4] Edmond Awad, Sohan Dsouza, Richard Kim, Jonathan Schulz, Joseph Henrich, Azim Shariff, Jean-François Bonnefon, and Iyad Rahwan. 2018. The moral machine experiment. *Nature* 563, 7729 (2018), 59–64.
[5] Beina Azadgoli, Daniel J Gould, Emma Vartanian, and Ketan M Patel. 2019. The public's perception on breast and nipple reconstruction: a crowdsourcing-based assessment. *Aesthetic surgery journal* 39, 9 (2019), NP370–NP376.
[6] Rushlene Kaur Bakshi, Navneet Kaur, Ravneet Kaur, and Gurpreet Kaur. 2016. Opinion mining and sentiment analysis. In *2016 3rd international conference on computing for sustainable global development (INDIACom)*. IEEE, 452–455.
[7] Sean Baron. 2012. Cognitive flow: the psychology of great game design. *Gamastura* (2012).
[8] Tara S Behrend, David J Sharek, Adam W Meade, and Eric N Wiebe. 2011. The viability of crowdsourcing for survey research. *Behavior research methods* 43, 3 (2011), 800–813.
[9] Regina Bernhaupt. 2010. User experience evaluation in entertainment. In *Evaluating user experience in games*. Springer, 3–7.
[10] Robin Burke. 2002. Interactive critiquing forcatalog navigation in e-commerce. *Artificial Intelligence Review* 18, 3 (2002), 245–267.
[11] Robin D Burke, Kristian J Hammond, and BC Yound. 1997. The FindMe approach to assisted browsing. *IEEE Expert* 12, 4 (1997), 32–40.
[12] Olivier Chapelle and Lihong Li. 2011. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*. 2249–2257.

[13] Li Chen and Pearl Pu. 2012. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction* 22, 1 (2012), 125–150.

[14] Matthew JC Crump, John V McDonnell, and Todd M Gureckis. 2013. Evaluating Amazon's Mechanical Turk as a tool for experimental behavioral research. *PloS one* 8, 3 (2013), e57410.

[15] Mihaly Csikzentmihaly. 1990. *Flow: The psychology of optimal experience.* Vol. 1990. Harper & Row New York.

[16] Kapil Dev, Nicolas Villar, and Manfred Lau. 2017. Polygons, points, or voxels? stimuli selection for crowdsourcing aesthetics preferences of 3D shape pairs. In *Proceedings of the symposium on Computational Aesthetics.* 1–7.

[17] Most Attractive Expressions. [n. d.]. Mirror Mirror: Crowdsourcing Better Portraits. *To appear in ACM TOG* 33 ([n. d.]), 6.

[18] Boi Faltings, Pearl Pu, Marc Torrens, and Paolo Viappiani. 2004. Designing example-critiquing interaction. In *Proceedings of the 9th international conference on Intelligent user interfaces.* 22–29.

[19] Elena Filatova. 2012. Irony and Sarcasm: Corpus Generation and Analysis Using Crowdsourcing.. In *Lrec.* Citeseer, 392–398.

[20] Gianfranco Frojo, Aurora M Kareh, Kenneth X Probst, Jeffrey D Rector, Christina M Plikaitis, Herluf G Lund, and Alexander Y Lin. 2021. Defining Ideal External Female Genital Anatomy Via Crowdsourcing Analysis. *Aesthetic surgery journal* (2021).

[21] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576* (2015).

[22] Marc Hassenzahl. 2008. User experience (UX) towards an experiential perspective on product quality. In *Proceedings of the 20th Conference on l'Interaction Homme-Machine.* 11–15.

[23] Marc Hassenzahl and Noam Tractinsky. 2006. User experience-a research agenda. *Behaviour & information technology* 25, 2 (2006), 91–97.

[24] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2020. A survey on conversational recommender systems. *arXiv preprint arXiv:2004.00646* (2020).

[25] Yucheng Jin, Wanling Cai, Li Chen, Nyi Nyi Htun, and Katrien Verbert. 2019. MusicBot: Evaluating critiquing-based music recommenders with conversational interaction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management.* 951–960.

[26] Yuan Jin, Mark Carman, Ye Zhu, and Yong Xiang. 2020. A technical survey on statistical modelling and design methods for crowdsourcing quality control. *Artificial Intelligence* (2020), 103351.

[27] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision.* Springer, 694–711.

[28] Ron Kohavi and Roger Longbotham. 2017. Online Controlled Experiments and A/B Testing. *Encyclopedia of machine learning and data mining* 7, 8 (2017), 922–929.

[29] Ron Kohavi and Stefan Thomke. 2017. The surprising power of online experiments. *Harvard Business Review* (2017).

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision.* Springer, 740–755.

[31] Gitte Lindgaard and Jurek Kirakowski. 2013. The tricky landscape of developing rating scales in HCI. *Interacting with Computers* 25, 4 (2013), 271–277.

[32] Jonathan P Massie, Rachita Sood, Ian T Nolan, Daniel C Sasson, Marco Swanson, Shane D Morrison, and Otto Placik. 2021. Defining aesthetic preferences for the penis: a photogrammetric and crowdsourcing analysis. *Aesthetic surgery journal* (2021).

[33] Alan Medlar, Joel Pyykkö, and Dorota Glowacka. 2017. Towards fine-grained adaptation of exploration/exploitation in information retrieval. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces.* 623–627.

[34] Maria Perez-Ortiz and Rafal K Mantiuk. 2017. A practical guide and software for analysing pairwise comparison experiments. *arXiv preprint arXiv:1712.03686* (2017).

[35] Pearl Pu and Boi Faltings. 2004. Decision tradeoff using example-critiquing and constraint programming. *Constraints* 9, 4 (2004), 289–310.

[36] José Luis González Sánchez, Francisco Luis Gutiérrez Vela, Francisco Montero Simarro, and Natalia Padilla-Zea. 2012. Playability: analysing user experience in video games. *Behaviour & Information Technology* 31, 10 (2012), 1033–1054.

[37] Natasha Dow Schull. 2005. Digital gambling: The coincidence of desire and design. *The Annals of the American Academy of Political and Social Science* 597, 1 (2005), 65–81.

[38] Noor Shaker, Georgios N Yannakakis, and Julian Togelius. 2012. Crowdsourcing the aesthetics of platform games. *IEEE Transactions on Computational Intelligence and AI in Games* 5, 3 (2012), 276–290.

[39] Marissa Suchyta, Amee Azad, Ashraf A Patel, Rohit K Khosla, H Peter Lorenz, and Rahim S Nazerali. 2020. Applied online crowdsourcing in plastic and reconstructive surgery: a comparison of aesthetic outcomes in unilateral cleft lip repair techniques. *Annals of plastic surgery* 84, 5S (2020), S307–S310.

[40] Steve Swink. 2008. *Game feel: a game designer's guide to virtual sensation.* CRC Press.

[41] Terry M Therneau and Thomas Lumley. 2014. Package 'survival'. *Survival analysis Published on CRAN* 2 (2014), 3.

[42] Raymond W Tse, Eugene Oh, Joseph S Gruss, Richard A Hopper, and Craig B Birgfeld. 2016. Crowdsourcing as a novel method to evaluate aesthetic outcomes of treatment for unilateral cleft lip. *Plastic and reconstructive surgery* 138, 4 (2016), 864–874.

[43] Zeynep Tufekci. 2018. YouTube, the great radicalizer. *The New York Times* 10 (2018), 2018.

[44] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022* (2016).

[45] Niels Van Berkel, Jorge Goncalves, Danula Hettiachchi, Senuri Wijenayake, Ryan M Kelly, and Vassilis Kostakos. 2019. Crowdsourcing perceptions of fair predictors for machine learning: a recidivism case study. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–21.

[46] Emma Vartanian, Daniel J Gould, Ziyad S Hammoudeh, Beina Azadgoli, W Grant Stevens, and Luis H Macias. 2018. The ideal thigh: a crowdsourcing-based assessment of ideal thigh aesthetic and implications for gluteal fat grafting. *Aesthetic surgery journal* 38, 8 (2018), 861–869.

[47] Paolo Viappiani, Boi Faltings, and Pearl Pu. 2006. Preference-based search using example-critiquing with suggestions. *Journal of artificial intelligence Research* 27 (2006), 465–503.

[48] Jesse Vig, Shilad Sen, and John Riedl. 2011. Navigating the tag genome. In *Proceedings of the 16th international conference on Intelligent user interfaces.* 93–102.

[49] Ga Wu, Kai Luo, Scott Sanner, and Harold Soh. 2019. Deep language-based critiquing for recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems.* 137–145.