MIS Quarterly

RESEARCH NOTE

REDUCING RECOMMENDER SYSTEM BIASES: AN INVESTIGATION OF RATING DISPLAY DESIGNS

Gediminas Adomavicius
Information and Decision Sciences, Carlson School of Management, University of Minnesota,
321 19th Avenue South, Minneapolis, MN 55455 U.S.A. {gedas@umn.edu}

Jesse C. Bockstedt
Information Systems and Operations Management, Goizueta Business School, Emory University,
1300 Clifton Road, Atlanta, GA 30322 U.S.A. {bockstedt@emory.edu}

Shawn P. Curley
Information and Decision Sciences, Carlson School of Management, University of Minnesota,
321 19th Avenue South, Minneapolis, MN 55455 U.S.A. {curley@umn.edu}

Jingjing Zhang
Operations and Decision Technologies, Kelley School of Business, Indiana University,
1309 East Tenth Street, Bloomington, IN 47405 U.S.A. {jjzhang@indiana.edu}

Appendix A

Study 1 Experimental Results: Perturbed Recommendations in Various Display Types

As an extension to a more realistic setting and as a robustness check, we examine whether biases generated by *perturbations* in real recommendations from an actual recommender system can be eliminated by the rating display options. Recall that participants received some recommendations that were perturbed either upward (High-Perturbed) or downward (Low- Perturbed) by 1 star from the actual predicted ratings. As a control, each participant also received recommendations without perturbations (Accurate).

For our main analysis of the perturbed recommendations, submitted ratings for the jokes were adjusted for the predicted ratings in order to obtain a response variable on a comparable scale across subjects. Thus, the main response variable is the *rating drift*, which is defined by Adomavicius et al. (2013) as[1]:

$$RatingDrift = UserRating - PredictedRating$$

The *RatingDrift* variable captures the degree to which the user's submitted rating is higher or lower than the true rating predicted by the system (i.e., *before* any experiment-specific perturbations are applied to it), thereby accounting for individual differences in preference for different

[1]Following Adomavicius et al., we use rating drift as the dependent variable for analyzing bias effects with perturbed recommendations. Since *PredictedRating* is a component of rating drift and also included as a control factor in the model, we could alternatively run the model using *UserRating* as the dependent variable. Since this analysis is interested in the effects of perturbations from the predicted rating, we follow Adomavicius et al. in using rating drift (also measuring differences from predicted ratings) as a more natural and intuitive measure in this case. In any event, all conclusions are identical using *UserRating* as the dependent variable (only the *PredictedRating* coefficient estimates change in the model).

items being viewed. Thus, using *RatingDrift* puts our dependent measure on an equal footing (across different users and items) with the perturbations that are the main manipulation for these conditions.

As done with the artificial ratings, we first test for particular differences among the seven different rating display interfaces. Figure A1 is a plot of the aggregate means of rating drift for each treatment group when recommendations were perturbed to be higher or lower or received no perturbation. As can be seen, the negative perturbations (Low, triangle) led to negative rating drifts and positive perturbations (High, dot) led to positive drifts in user ratings, while the accurate recommendations with no perturbation (Accurate, square) led to drifts around zero. For each rating display, we performed pair-wise *t*-tests to compare user-submitted ratings after receiving high and low artificial recommendations. The *t*-test results are presented in Table A1.

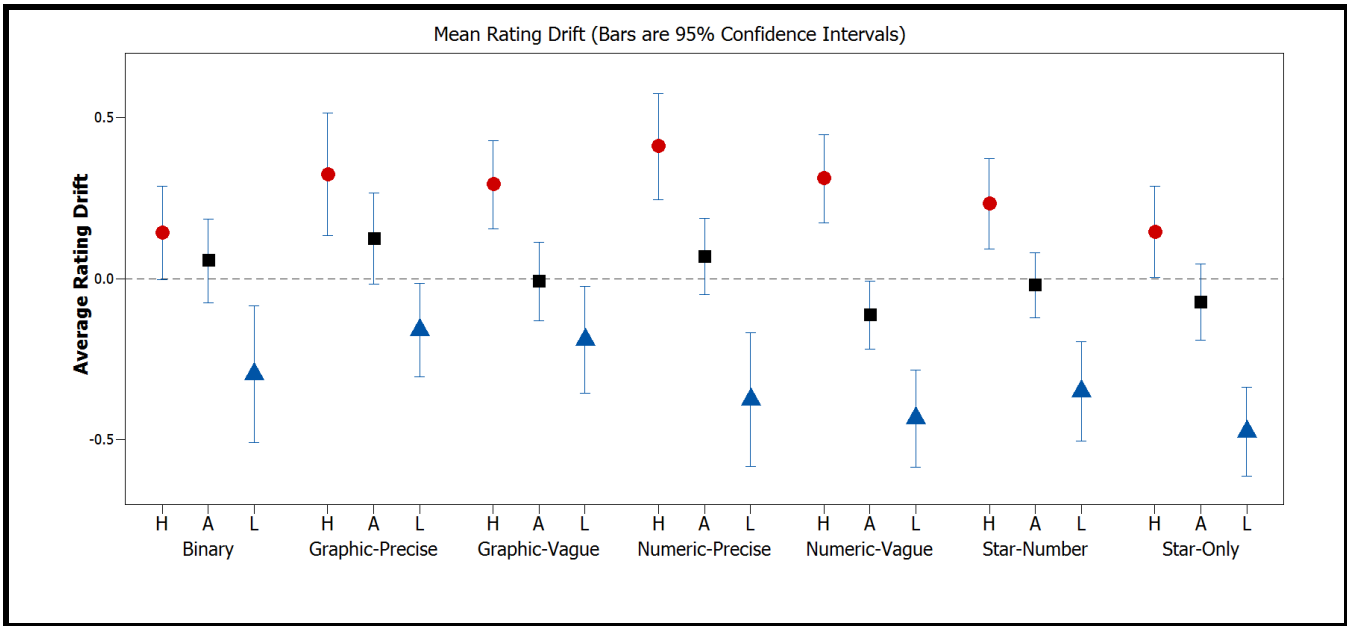


Figure A1. Means and 95% Confidence Intervals of User Rating Drift After Receiving High Perturbed (High: dot), Low Perturbed (Low: triangle), and Non-Perturbed Recommendations (Accurate: square)

Table A1. Pair-Wise Comparisons of Mean Rating Drift Difference for Each Rating Display Option Using *T*-Tests

Rating Display	High – Low	High – Accurate	Low – Accurate
Binary	0.446***	0.104	-0.318**
Graphic-Precise	0.492***	0.292**	-0.187*
Graphic-Vague	0.482***	0.286**	-0.196*
Numeric-Precise	0.799***	0.491***	-0.297**
Numeric-Vague	0.770***	0.315**	-0.420***
Star-Numeric	0.599***	0.196**	-0.391***
Star-Only	0.671***	0.140*	-0.474***

*$p \le 0.05$, **$p \le 0.01$, ***$p \le 0.001$

All mean rating drift comparisons between High and Low perturbed conditions are significant for all rating display options (one-tailed p-value < 0.001 for all High versus Low tests), showing a clear and positive bias of system recommendations on consumers' rating drift. Hence, similarly to the artificial recommendation scenario, we found that none of the seven rating display options could completely remove the biases generated by perturbed real recommendations.

We next performed regression analyses to compare the size of recommendation bias across different rating display options, while controlling for participant-level factors. The random effects GLS model using robust standard errors, clustered by participant, and participant-level controls represents our model for the analysis:

$$RatingDrift_{ij} = b_0 + b_1(High_{ij}) + b_2(Display_i) + b_3(Display_i \times High_{ij}) + b_4(PredictedRating_{ij}) + b_5(AdditionalControls) + u_i + \varepsilon_{ij}$$

We ran the model using the Numeric-Precise as the baseline, paralleling the analysis in Table 5 of the paper. Table A2 summarizes the regression analysis of perturbed recommendations.

Similar to what we found in the artificial recommendation analysis, several of the interaction terms were significant. Although the differences that attain statistical significance at a 5% level are fewer in number, they correspond to those seen in Table 5 with the artificial recommendations. The Numeric-Precise display yielded the greatest system recommendation effect, and the effect was significantly greater than for the Graphic-Vague and Binary displays. In addition, the Binary display showed the least effect, and the effect was significantly lower than for the Numeric-Precise and Numeric-Vague displays.

Table A2. Regression Analysis on High and Low Perturbed Recommendations (Baseline: Numeric-Precise; Dependent Variable: RatingDrift)

	Coefficient Estimate	Standard Error
High	0.778***	(0.119)
PredictedRating	-0.123*	(0.068)
Display		
Numeric-Vague	-0.083	(0.127)
Star-Numeric	0.008	(0.131)
Star-Only	-0.126	(0.126)
Graphic-Precise	0.198	(0.126)
Graphic-Vague	0.163	(0.132)
Binary	0.065	(0.143)
Interactions		
Numeric-Vague×High	-0.040	(0.153)
Star-Numeric×High	-0.189	(0.157)
Star-Only×High	-0.140	(0.154)
Graphic-Precise×High	-0.285	(0.168)
Graphic-Vague×High	-0.302*	(0.152)
Binary×High	-0.360*	(0.169)
Controls		
jokeFunniness	0.234*	(0.095)
age	0.000	(0.006)
male	0.029	(0.045)
undergrad	-0.051	(0.067)
native	-0.015	(0.056)
IfUsedRecSys	0.049	(0.059)
PredictionAccurate	0.038	(0.031)
PredictionUseful	0.009	(0.025)
Constant	-0.988**	(0.386)
R^2 within-subject	0.149	
R^2 between-subject	0.010	
R^2 overall	0.121	
χ^2	266.15***	

*$p \leq 0.05$, **$p \leq 0.01$, ***$p \leq 0.001$

Paralleling the analysis in Table 6 of the paper, for our perturbed recommendations we next conduct the regression analysis for our baseline 2 × 2 between-subjects design on the two dimensions of display format (numeric versus graphic) and precision of the recommendations (precise versus vague). We created a panel from the data as each participant was exposed to both high and low perturbed recommendations in a random fashion. The regression model used generalized least squares (GLS) estimation, a random effect to control for participant-level heterogeneity, and robust standard errors clustered by participant:

$$\begin{aligned} RatingDrift_{ij} = {} & b_0 + b_1(High_{ij}) + b_2(Presentation_i) + b_3(Precision_i) + \\ & b_4(Presentation_i \times Precision_i) + b_5(Presentation_i \times High_{ij}) + b_6(Precision_i \times High_{ij}) + \\ & b_7(ShownRatingNoise_{ij}) + b_8(PredictedRating_{ij}) + b_9(AdditionalControls) + u_i + \varepsilon_{ij} \end{aligned}$$

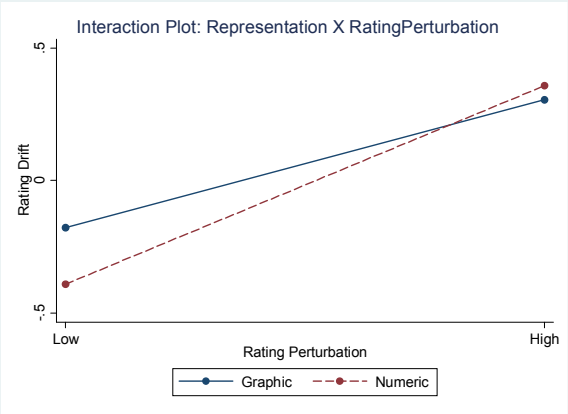
$RatingDrift_{ij}$ is the difference between the submitted rating and the predicted rating for participant i on joke j. The right-hand-side variables are the same as those used in the analysis of the artificial recommendation data in the main body of the paper.

Table A3. Regression Analysis on Perturbed Recommendations, for Numeric/Graphic and Precise/Vague Rating Displays (Dependent Variable: RatingDrift)

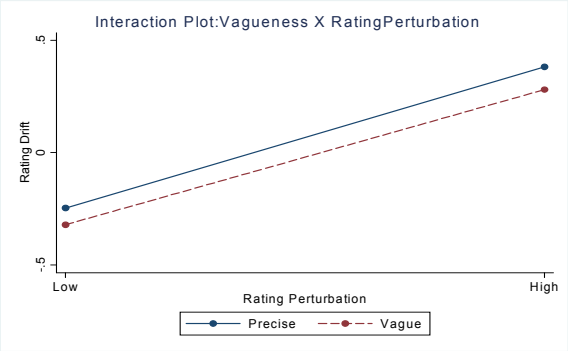
	Coefficient Estimate	Standard Error
High	0.469***	(0.086)
PredictedRating	-0.203**	(0.083)
Presentation (Numeric = 1, Graphic = 0)	-0.254**	(0.098)
Precision (Precise = 1, Vague = 0)	0.034	(0.094)
Presentation×Precision	0.081	(0.118)
Presentation×High	0.267**	(0.108)
Precision×High	0.028	(0.107)
Controls		
jokeFunniness	0.413***	(0.129)
age	0.003	(0.007)
male	0.099	(0.063)
Undergrad	-0.137	(0.089)
native	-0.069	(0.066)
IfUsedRecSys	0.096	(0.075)
PredictionAccurate	0.087*	(0.036)
PredictionUseful	-0.040	(0.027)
Constant	-1.263*	(0.503)
R^2 within-subject	0.174	
R^2between-subject	0.041	
R^2 overall	0.147	
$\chi 2$	184.15***	

*$p \leq 0.05$, **$p \leq 0.01$, ***$p \leq 0.001$

Our results (Table A3) corroborate the findings from the analyses using artificial recommendations. Specifically, the results confirm that the precision of the recommendations does not have a significant main effect ($p = 0.372$) or interaction with rating perturbation (i.e., *Precision* × *High*, $p = 0.390$) in affecting rating shifts. As shown in Figure A2b, recommendations displayed in precise versus vague forms have a similar level of effects in influencing rating drift (slopes are not detectably different).



(a) Information Representation (Numeric or Graphic) and Perturbed Rating (High or Low)



(b) Precision of Recommendation (Precise or Vague) and Perturbed Rating (High or Low)

Note: The dotted reference line denotes the mean user pre-treatment ratings.

Figure A2. Interaction Plots of Rating Displays Forms and Rating Values

However, presenting recommendations in numeric format compared to a graphical format exhibits a significant main effect ($p = 0.005$) and interaction (i.e., *Presentation* × *High*, $p = 0.0075$) upon rating drift. The interaction plot in Figure A2a illustrates that the combined effect—the difference between ratings for High versus Low conditions (slope of the line)—is greater in the numeric conditions compared to the graphical conditions.

Appendix B

Study 2 Experimental Results: Perturbed Recommendations in Various Display and Response Types

Following the analyses of Study 1, we compared the mean rating drift of the four experimental groups after receiving perturbed recommendations. Table B1 summarizes the aggregate means for each of the experimental groups and within-subjects conditions. All comparisons between High and Low conditions are significant across the four experimental conditions, showing that a significant bias exists for all display-response combinations. As with Study 1, none of the interface designs were able to remove the bias completely.

Table B1. Mean Rating Drift for Each Rating Response and Display Interface Option, After Receiving Perturbed Recommendations

Experimental Conditions	High Perturbed	Low Perturbed	Accurate
GraphicDisplay, GraphicResponse	0.443	-0.260***	0.119
GraphicDisplay, NumericResponse	0.331	-0.160***	0.125
NumericDisplay, GraphicResponse	0.282	-0.271***	-0.035
NumericDisplay, NumericResponse	0.412	-0.375***	0.070

*** High versus Low, $p < .001$

For the system recommendations that are perturbed to be either higher or lower, we again use rating drift as a measure of bias and a model paralleling that for Study 1:

$$RatingDrift_{ij} = b_0 + b_1(High_{ij}) + b_2(InterfaceMatch_i) + b_3(NumericalDisplay_i) + b_4(InterfaceMatch_i \times High_{ij}) + b_5(NumericalDisplay_i \times High_{ij}) + b_6(PredictedRating_{ij}) + b_7(AdditionalControls) + u_i + \varepsilon_{ij}$$

All variable definitions are consistent with those described in the paper. Table B2 shows the regression results. The results somewhat differ for the perturbed recommendations. Only the matching display-response formats variable shows a higher bias for perturbed recommendations. The numerical versus graphical display manipulation did not influence participants' bias.

Table B2. Regression Analysis on Perturbed Recommendations for Numeric/Graphic Displays and Responses (Dependent Variable: RatingDrift)

	Coefficient Estimate	Standard Error
High	0.497***	(0.100)
PredictedRating	-0.046	(0.085)
InterfaceMatch	-0.079	(0.084)
InterfaceMatch×High	0.220*	(0.111)
NumericalDisplay	-0.094	(0.082)
NumericalDisplay×High	0.059	(0.110)
Controls		
jokeFunniness	0.367***	(0.120)
Age	-0.002	(0.006)
Male	0.071	(0.065)
Undergrad	-0.073	(0.083)
Native	-0.029	(0.062)
IfUsedRecSys	0.115†	(0.078)
PredictionAccurate	0.013	(0.037)
PredictionUseful	0.014	(0.032)
Constant	-1.378	(0.477)
R^2 within-subject	0.154	
R^2 between-subject	0.035	
R^2 overall	0.129	
χ^2	175.20***	

†$p \leq 0.1$, *$p \leq 0.05$, **$p \leq 0.01$, ***$p \leq 0.001$

Overall, Study 2 shows consistently significant support for the effect of scale compatibility upon bias. However, the graphical/numerical processing differences for incoming information describe a less stable effect.

Appendix C

Performance of Recommendation Algorithms

We used the well-known item-based collaborative filtering (CF) technique (Sarwar et al. 2001) to implement a recommender system that estimated users' preference ratings for the jokes. We chose to use this technique in our experiment for several reasons. First, item-based CF is one of the most popular techniques used in real-world applications because of its efficiency and accuracy (e.g., Adomavicius and Tuzhilin 2005; Deshpande and Karypis 2004; Linden et al. 2003; Sarwar et al. 2001). Second, this technique allows us to precompute the main portion of our recommendation model (i.e., the similarity scores between items based on their rating patterns) *in advance* based on the extensive Jester rating dataset. Thus, we could compute rating predictions for our experimental participants as soon as they submitted their preferences. We did not have to perform extensive recomputations for each new participant on the fly (as needed for some other recommendation techniques), which was important due to the real-time nature of our experiment. Third, while an item-based approach was very computationally efficient, its accuracy performance was also either better or not significantly different than a number of other techniques that we explored. The details of this selection process are provided below.

Table C1. Comparison of Recommendation Algorithms on Joke Rating Dataset

Algorithm	Description	Predictive Accuracy (MAE)
User Average	Predict each unknown user-item rating as an average of all ratings of that user (item)	0.7458
Item Average		0.7686
Baseline Estimate	Computes each unknown user-item rating with the baseline estimate which is a combination of the global mean of ratings in the dataset, the average rating deviation of corresponding user and the average rating deviation of the corresponding item.	0.6897
Weighted Slope One	Estimates average rating difference between all item pairs. For a given unknown user-item rating, finds all the items that were co-rated with this item and computes predictions based on each of these items. The final prediction is a weighted sum of predictions from individual items (Lemire and Maclachlan 2005).	0.6897
User-based CF	For each unknown rating, finds the most similar items that have been rated by the same user (or the most similar users who have rated the same item) and predicts the rating as a weighted sum of neighbors' ratings (Breese et al. 1998; Sarwar et al. 2001).	0.6841
Item-based CF		**0.6795**
Matrix Factorization	Decomposes the rating matrix into two matrices so that each user and each item is associated with a user vector and an item vector of latent variables. Prediction is done by taking inner product of user and item vectors (Funk 2006; Koren et al. 2009).	0.6817

Based on the joke rating data collected in Task 1, we built a recommender system to predict user's preference ratings on jokes. We compared seven popular recommendation techniques (Table C1) to find the best-performing technique for our data set. The techniques included simple user- and item-based rating average methods, user-item baseline method (Bell and Koren 2007), weighted Slope One approach (Lemire and Maclachlan 2005), user- and item-based collaborative filtering (CF) approaches (Breese et al. 1998; Sarwar et al. 2001), as well as a model-based matrix factorization algorithm (Funk 2006; Koren et al. 2009). Each recommendation algorithm was evaluated using five-fold cross validation based on the standard Mean Absolute Error (MAE) metric. As we can see from the Table C1, the item-based CF algorithm had the best MAE performance (0.6795), followed by matrix factorization (0.6817), then by user-based CF (0.6841), weighted slope one (0.6897) and baseline (0. 6897), and lastly by the simple user average (0.7458) and item average (0.7686) approaches. Based on these results, we selected the item-based CF approach as our recommendation technique to predict users' preference ratings on jokes.

To further validate the performance of the selected item-based CF algorithm, we applied the algorithm on the jokes that were included in the control conditions. These control jokes were rated by users as part of the experiment but did not have any recommendation displayed. We then compared the predicted rating against the actual user ratings, and the mean difference, calculated as an average of pair-wise differences between corresponding actual and predicted ratings, is 0.005. Our pair-wise *t*-test results show that the differences between the system-predicted and actual ratings are not significant (two-tailed p-value = 0.8202). We further conducted the comparison test separately for each treatment group and found that none of the groups had significant differences between system-predicted ratings and actual user ratings. Table C2 summarizes our results.

Table C2. Mean Difference between Predicted and Actual Ratings on Control Jokes

Group	Mean Predicted Rating	Mean User Rating	Mean Difference	*P*-value of pair-wise t-test
Binary	2.8495	2.8900	0.0405	0.5162
Graphic-Precise	3.0175	3.0450	0.0275	0.6917
Graphic-Vague	2.8575	2.8650	0.0075	0.9179
Numeric-Precise	2.9790	2.9000	-0.0790	0.2313
Numeric-Vague	2.8144	2.8103	-0.0041	0.9551
Star-Number	2.8947	2.9311	0.0364	0.5156
Star-Only	2.8772	2.8837	0.0065	0.9171

Appendix D

Examples of Graphical Rating Bar Displays

Below are screenshots of different graphical rating bar examples (e.g., accompanied by numeric values or not, shaped like a progress bar versus a slider bar, etc.) taken from different websites. Red borders were added for emphasis.

Figure D1. Orbitz (travel/hotel)

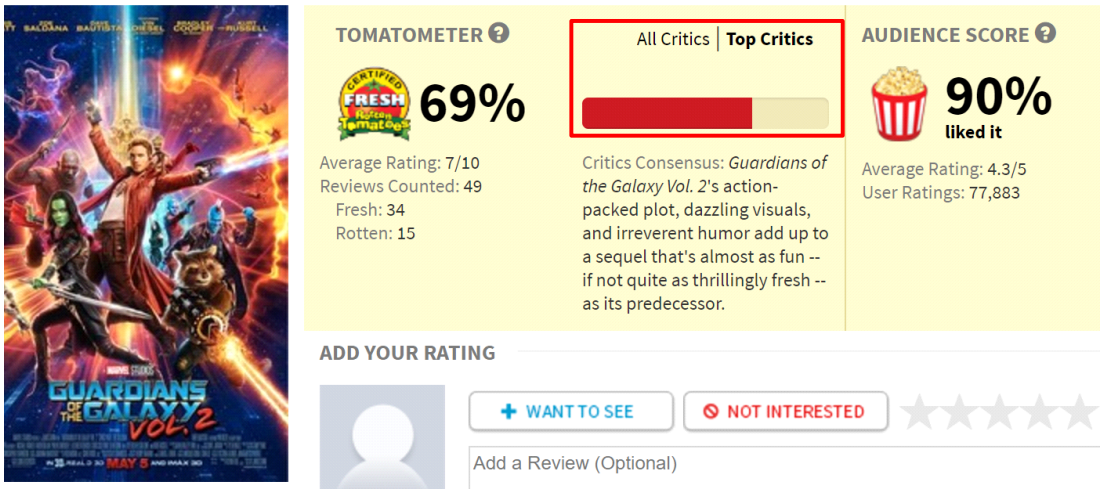


Figure D2. Rotten Tomatoes (movies)

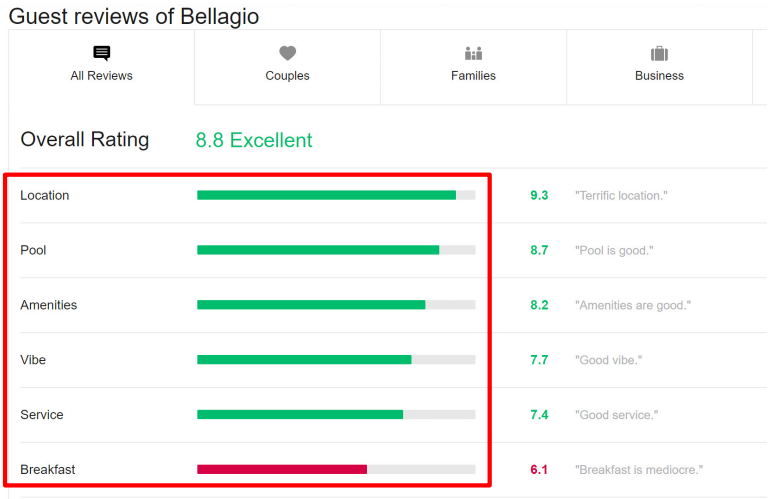


Figure D3. Kayak (travel/hotel

Figure D4. Eigentaste (joke recommendation website)

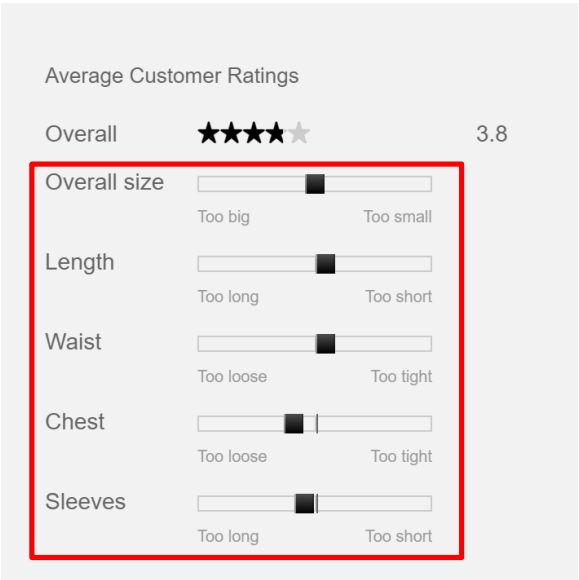


Figure D5. Banana Republic (retail/apparel)

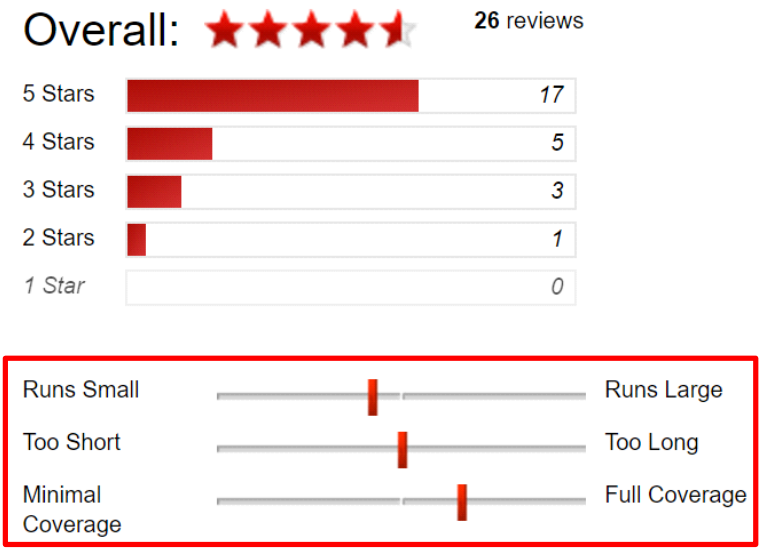


Figure D6. Macy's (retail/apparel)

Examples of System Ratings and User Rating Interfaces Displayed Together

Below are screenshots of several online services that commonly display system ratings next to the interface for users to submit their personal ratings for items.

Figure D7. IMDB

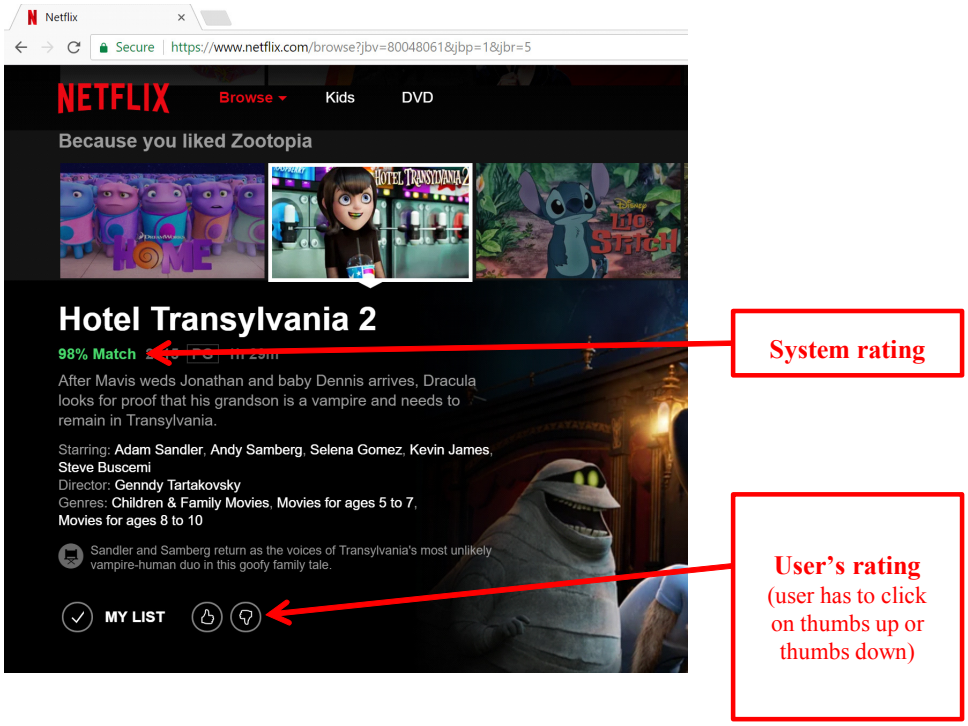


Figure D8. Netflix

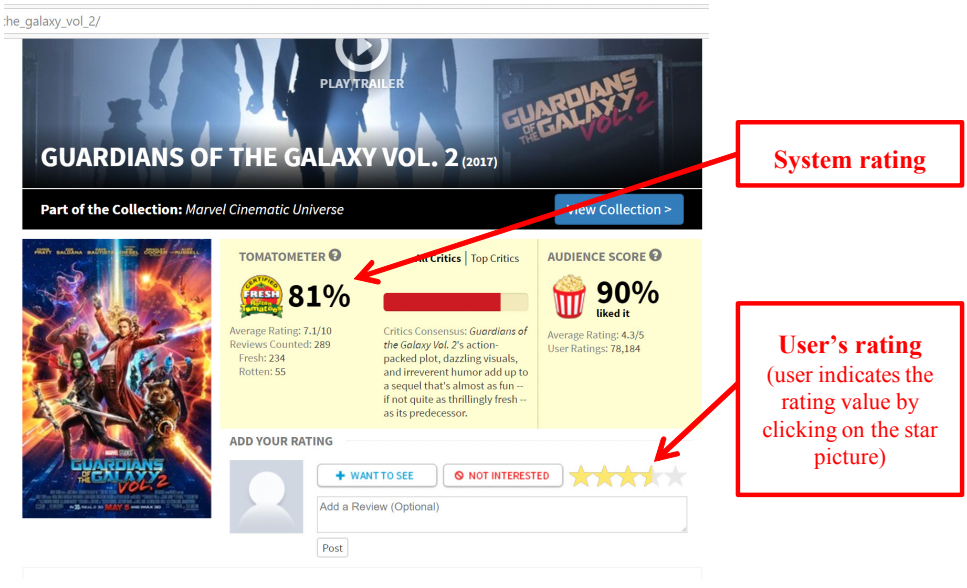


Figure D9. Rotten Tomatoes

References

Adomavicius, G., Bockstedt, J., Curley, S., and Zhang, J. 2013. "Do Recommender Systems Manipulate Consumer Preferences? A Study of Anchoring Effects," *Information Systems Research* (24:4), pp. 956-975.

Adomavicius, G., and Tuzhilin, A. 2005. "Toward the Next Generation of Recommendation System: A Survey of the State-of-the-Art and Possible Extensions," *IEEE Transactions on Knowledge and Data Engineering* (17:6), pp. 734-749.

Bell, R. M., and Koren, Y. 2007. "Improved Neighborhood-Based Collaborative Filtering," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge (KDD Cup'07*), New York: ACM, pp. 7-14.

Breese, J. S., Heckerman, D., and Kadie, C. 1998. "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, San Francisco: Morgan Kaufmann, pp. 43-52.

Deshpande, M., and Karypis, G. 2004. "Item-Based Top-N Recommendation Algorithms," *ACM Transactions on Information Systems* (22:1), pp. 143-177.

Funk, S. 2006. "Netflix Update: Try This at Home" (http://sifter.org/~simon/journal/20061211.html).

Koren, Y., Bell, R., and Volinsky, C. 2009. "Matrix Factorization Techniques for Recommender Systems," *IEEE Computer Society* (42), pp. 30-37.

Lemire, D., and Maclachlan, A. 2005. "Slope One Predictors for Online Rating-Based Collaborative Filtering," in *Proceedings of the 2005 SIAM International Conference on Data Mining,* H. Kargupta, J. Srivastava, C. Kamath and A. Goodman (eds.), Newport Beach, California, pp. 471-475.

Linden, G., Smith, B., and York, J. 2003. "Amazon.Com Recommendations: Item-to-Item Collaborative Filtering," *IEEE Internet Computing* (7:1), pp. 76-80.

Sarwar, B., Karypis, G., Konstan, J. A., and Riedl, J. 2001. "Item-Based Collaborative Filtering Recommendation Algorithms," in *Proceedings of the 10th International World Wide Web Conference*, New York: ACM, pp. 285-295.