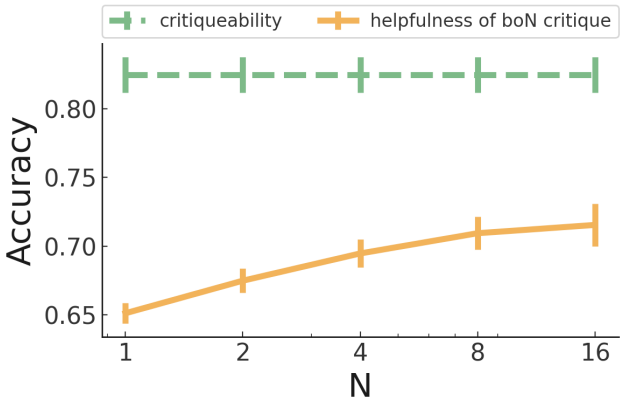


Figure 14: Discrimination between misleading and honest answers. Best-of-N according to a helpfulness model appears to asymptote to worse than a direct critiqueability model. This suggests using helpfulness labels directly as a training signal for debate might not be preferred to training directly against human critique labels, in the topic-based summarization domain.

2. The best-of-N helpfulness score never discriminates as well as the critiqueability score
3. The best-of-N helpfulness score appears to be asymptoting with $N \to \infty$ to worse than the critiqueability score. This suggests naive 2-step debate wouldn't work as well as RLHP for this task.

While these are negative results, they don't imply that debate won't start working with larger models. Some other important caveats:

1. Our helpfulness model may primarily prioritize nit-picky critiques rather than severe ones, so we expect training a severity model (or e.g. a critique-conditional preference-based reward model) would help.
2. Furthermore, we are primarily interested in testing in domains where giving critiqueability labels is even more difficult.
   (a) With human labels, it seems interesting if critiqueability labels come from humans being misled in systematically biased ways. In our case, labels' performance may simply be poor due to noise from occasional carelessness. This doesn't affect accuracy in the limit of infinite training data. Helpfulness-based training may be less biased, and thus debate should help more.
   (b) If studying debate with simple synthetic tasks, it could be particularly interesting with a Σ_2^P/Π_2^P-complete problem such as 2QBF where learning the helpfulness oracle is easy but learning the critiqueability oracle is hard.

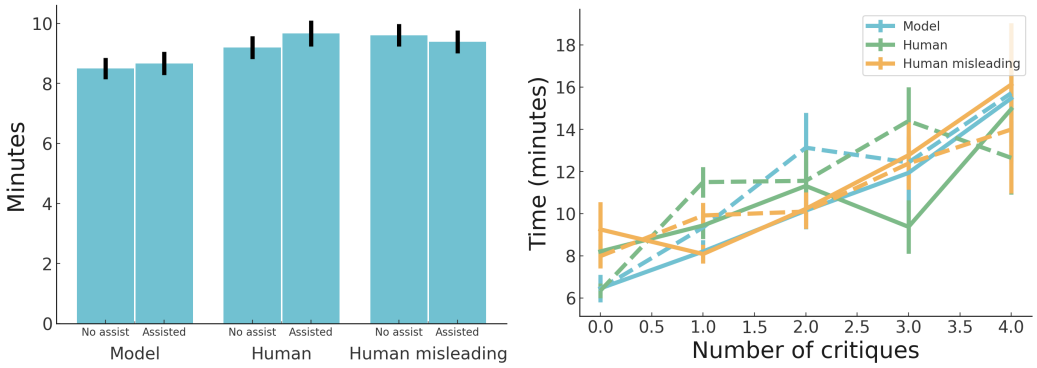
Nevertheless, this could mean that debate is more difficult to implement initially than recursive reward modeling due to the need for a robust helpfulness model. In recursive reward modeling, having a human in the loop to interact with multiple critiques means we can see signs of life without robust critiques.

E Other assistance experiments

E.1 Assistance for comparisons

We initially tried using assistance for the task of doing comparisons. Unlike the critique-writing setting, we were able to observe improvements in speed of comparisons. Our hope was that we could use an ensemble of unassisted humans as "ground truth" to show that critique assistance also helped a single human at accuracy.

Using ensembles of 5 humans as ground truth, we observed statistically significant improvements when using *human-written critiques* as assistance. With model-written critiques, we observe small



(a) Assistance does not appreciably slow down labelers. Any effect goes away when controlling for number of critiques found.

(b) Each additional found critique is correlated with about an additional minute of time. Here, the dotted line represents the no assist condition.

Figure 15: Amount of time labelers spend writing critiques, with and without assistance. We filter out outlier times of greater than an hour, as they are likely due to labelers taking breaks or timer malfunctions.

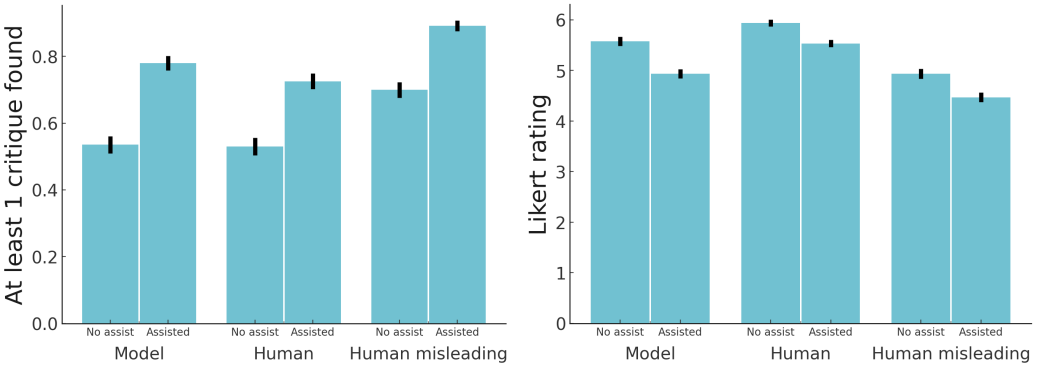


Figure 16: Assistance increases the fraction of answers with critiques found and decreases Likert score (1-7) judgements.

improvements that were within noise. Overall, this set up required a lot more effort and labeling to observe effects, so we discontinued it.

E.2 Quotes as assistance

We experimented with quotes as a form of assistance. We retrained the model to also generate supporting quotes for the critiques, from the response and/or text. Quotes were highlighted when the labeler clicked on the critique, and buttons let labelers scroll to the corresponding location in the text.

We found that:

- Quotes had no effect on number of critiques found
- Quotes save labelers a little under a minute of time.
- However, a baseline of highlighting longest common substrings between the critiques and text saved almost the same amount of time

E.3 Ablation of number of critiques

Earlier on in the project, we tried both 4 and 8 model-generated critiques as assistance. With only 4 critiques, finding critiques was possibly faster than the unassisted setting. However, it resulted in less

critiques found than the 8 critiques setting. The effect on number of critiques from 8 critiques was a little less than twice as large. Results are shown below.

	Time (minutes)	Number of critiques
No assistance	6.27 ± 0.31	0.74 ± 0.05
4 critiques	5.82 ± 0.27	1.06 ± 0.07
8 critiques	6.26 ± 0.27	1.31 ± 0.08

F Samples

In this section, all samples are uniformly randomly chosen, subject to the constraints described. We also always omit samples we deemed to have triggering content (this happened just once).

F.1 Self-critique and helpfulness samples

Here we provide random samples of model-generated answers, critiqueability scores, self-critiques, and self-assessed helpfulness scores. Finally, we provide whether each critique was marked helpful by a human.

All samples come from our largest model. We use a random non-zero temperature for the answer, and use the same temperature for critiques. For each answer, we draw 8 critiques, but deduplicate, similarly to in our assistance experiments.

The samples are shown in Tables 6-15.

F.2 Assistance samples

Here we choose random samples from the experiment in Section 3.4 such that each assistance condition had at least one critique used.

The samples are shown in Tables 16-25. We release the full dataset of samples in our assistance dataset release.

F.3 Refinement samples

Here we provide random samples of self-critiques and refinements from our largest model. We show three refinements: a conditional refinement using a random critique, a conditional refinement using a best-of-8 critique, and a direct refinement.

We filter for cases with all three refinements ranked higher than the original answer, according to a human labeler.

The samples are shown in Tables 26-35.