Grant Proposal Simulator

Michaël Trazzi
The Inside View

Claude 3.5 Sonnet
Anthropic

With
Jacques Thibodeau & Apart Research

Abstract

We build a VS Code extension to get feedback on AI Alignment research grant proposals by simulating critiques from prominent AI Alignment researchers and grantmakers.

Simulations are performed by passing system prompts to Claude 3.5 Sonnet that correspond to each researcher and grantmaker, based on some new grantmaking and alignment research methodology dataset we created, alongside a prompt corresponding to the grant proposal.

Results suggest that simulated grantmaking critiques are predictive of sentiment expressed by grantmakers on Manifund.

Keywords: Research feedback, grant application, debate

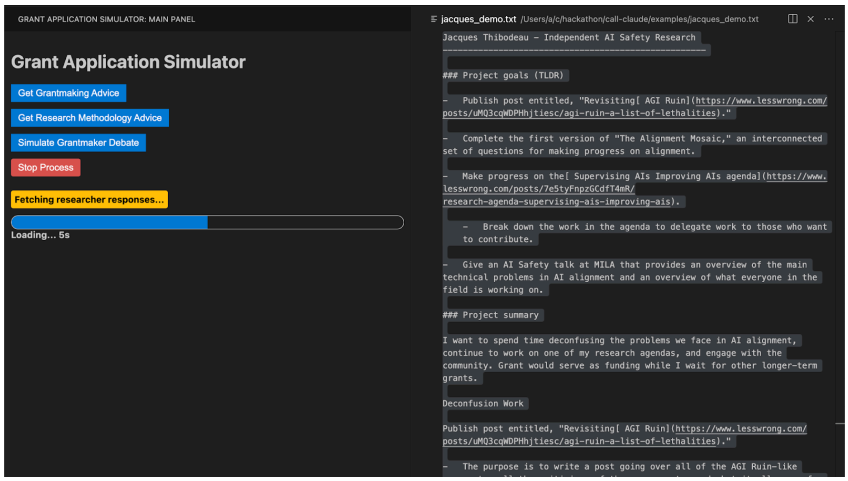


Figure 1: Grant proposal simulator (left), Jacques' grant application (right)

1. Tool overview

Three features:

1. **Get Grantmaking Advice**: Simulates feedback from multiple AI alignment grantmakers (Oliver Habryka, Austin Chen, Evan Hubinger and Adam Gleave) on a grant proposal. The feedbacks are first presented in short summaries, under "Summary", and then the full feedback for each grantmaker is presented in "Full Responses".

2. **Get Research Methodology Advice**: Offers critiques on AI alignment research methodologies from three different AI Alignment researchers who have written about Alignment Research Methodology (Ethan Perez, Paul Christiano and Rohin Shah). Also starts with a Summary.

3. **Grantmaker Debate Simulation**: Facilitates a debate between two selected AI alignment grantmakers (user selects them from the list given in "Grantmaking Advice" at the start), discussing whether they should fund the proposal. Finally, a summary is created of the debate at the beginning, summarizing the decision from the grantmakers.

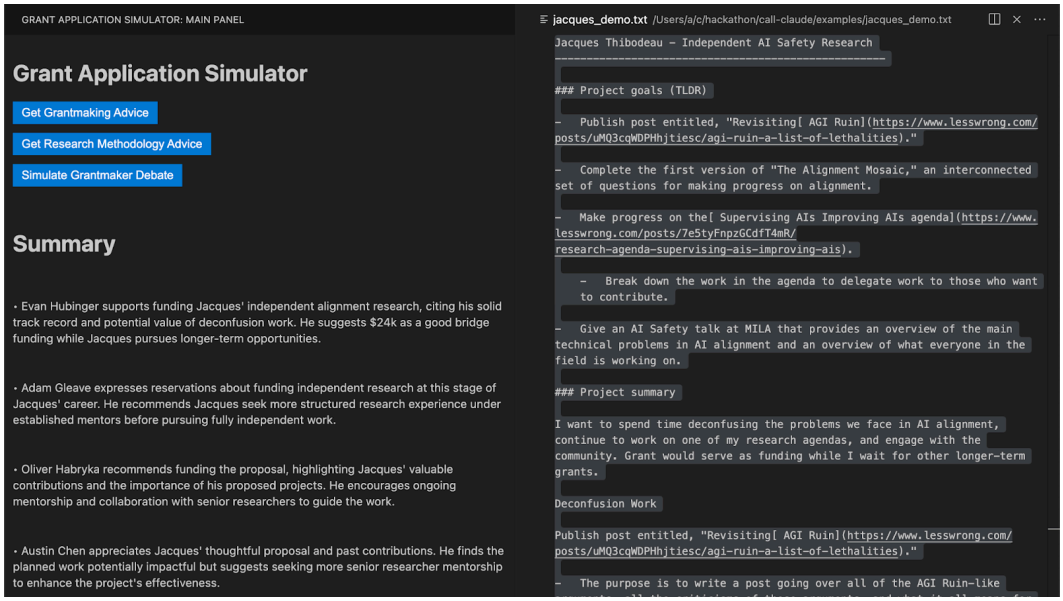


Figure 2: "Get Grantmaking Advice" on Jacques' proposal.

2. Why this tool should exist

Some criteria (like having a mentor) are crucial in doing impactful Alignment research, and are therefore things grantmakers look for in successful applications.

But not all criteria are made transparent. Jacques (alignment researcher) confirmed this was problematic.

This tool helps researchers identify holes in their projects by predicting what reservations grantmakers would have.

3. Results and next steps

Results: Tool was predictive on real grantmaker behavior on four different Manifund proposals (first and only ones I tried). Simulated grantmakers were enthusiastic about Apart's proposals, fitting Manifund's mostly positive comments (not taking into account very positive comments because our simulated grantmakers don't have access to private information). Mentorship concerns (eg. Evan here) fit some advice that Jacques had heard in the past. For Michaël's proposal, reservations regarding "limited reach" (see Adam here) did appear in actual Manifund comments.

All Experiments (each grant proposal with each feature):

Get Grantmaking advice:

- Jacques (grant link): output
- Michaël (grant link): output
- Esben (grant link): output
- Apart (grant link): output

Get Research Methodology Advice

- Jacques (grant link): output
- Michaël (grant link): output
- Esben (grant link): output
- Apart (grant link): output

Simulate Grantmaker Debate:

- Jacques (grant link): (Austin Chen and.Adam Gleave)
- Michaël (grant link): (Austin Chen and Adam Gleave)
- Esben (grant link): (Evan Hubinger and Oliver Habryka)
- Apart (grant link): (Evan Hubinger and Oliver Habryka)

Limitations:

- streaming only works on the debate feature, not all
- the loading time is more than 10 seconds and could be sped up by compressing data or simply using the minimum amount of data that we need to have good feedback)
- answers are sometimes truncated, which is from the token limit in the requests. I could try to change the system prompt to force it to be shorter, or simply extend the token limit
- passing data through prompts has a limit of 200k tokens

Next steps: compare grantmaker finetunes to claude 3.5 with sonnet (current), compress more data to context window, use agenda critiques data to improve

feedback, incorporate at different research project stages, add grant description to Manifund data (not just comments). (more)

Next Steps:

- try out different system prompts in debate to make it more advesarial / two people disagreeing on something, where one takes the camp of saying good things about the grant, and the other one bad things, instead of this where they sometimes agree and nothing much is said
- add grant description to Manifund data (not just comments)
    - create a jsonl dataset of "grant description" / "grant comment" pairs, not just comments for Manifund
    - try to do the same with the LTFF based on the writeups
- compare grantmaker finetunes to claude 3.5 with sonnet (current)
    - using this data, use some finetuning API like gpt-4o mini and see what kind of results do we get, and if they are similar to claude 3.5 sonnet
    - if not, try some other model, possibly LLAMA 3.1 base, or other base models
- compress more data to context window
    - if still using claude 3.5 sonnet, try to see if you could compress a lot of the context by asking models to reduce the token size of what's in the context directly
    - this could help speeding up the tool and also pack more data
- use more grantmaking data
    - could use more people from LTFF I haven't used
    - could look at other grant reports, like open phil
- use agenda critiques data to improve feedback
    - ideally, we'd want to use other sources of data, not just grantmaking data
    - for this, we could use critiques of agendas published on lesswrong
    - we could also use critiques from ai-plans
- incorporate at different research project stages
    - let's say a researcher wants feedback after having an idea for 2 hours, we should be able to give feedback on that too
    - if the researcher had an idea that was not ready for a grant proposal, we'd also want to provide feedback
    - for that, we'd need data about giving feedback on more raw ideas
    - I could also just see if the current grantmaking critiques would work on this more raw data (my current experiment on simple cases in meta_demo.txt "What if we could talk to Alignment researchers like Paul Christiano through a VSCode Extension? Could the workflow of AI Alignment researchers become two times faster? I think I would be done with this project in only 2 days because Claude is basically superhuman at coding." tells me this works)
    - we could also imagine versions of this that takes as input some research logs or experiment logs, maybe using google docs comments that are public (from alignment researcher) or even github issues / github PR comments

4. **References**
5. **Appendix**

Method. The prompts I use to imitate the grantmakers and researchers, depend on both a system prompt (telling the model how to imitate) and the prompt (being the grant being evaluated, or more generally the text being selected in the VS code extension that the user wants input on).

The full prompt is then sent to Claude 3.5 Sonnet using the API.

For system prompts, I have *intermediary system prompts* that depend on the kind of person we are imitating:

- **For grantmaking**: "You are evaluating a grant proposal for AI alignment research. Provide concise, constructive feedback focusing on the proposal's potential impact, feasibility, and alignment with current priorities in the field. Be specific and actionable in your critique."
- **For research methodology advice**: "You are providing feedback on an AI alignment research methodology. Focus on how to improve the approach to make significant progress in alignment research. Consider aspects such as experiment design, theoretical foundations, and potential impact on the field. Please give actual critiques about the feasibility of the approach, and how it would actually lead to more impactful AI Alignment research being produced down the line, say papers published at conferences, or progress being made to reduce existential risk."

We then combine these intermediary system prompts with all of the writing from this person (called researcher prompt here) to get our final system prompt:

- **Final System Prompt:** "${systemPrompt}\n\nYou are ${researcher}. Provide a response that reflects your unique viewpoints and critiques. Focus on your specific background and expertise. Avoid general ethical considerations or points that other researchers could make. Be concise and direct. What specific, unique critique or insight would you give to this code or text snippet, as a way to make progress in AI Alignment? Do not mention your name in your answer, or speak in the third person. Really speak like if you were that person. Here is some text that represents your unique perspective: [BEGIN TEXT]${researcherPrompt}[END TEXT]. Speak like you were that person. \n\n Again, DO NOT USE ANY BULLET POINTS AND ONLY SAY ONE PARAGRAPH`"

A lot of this has been some work to make sure that the answers were one pagraph long without using bullet points. I also wanted to make sure they do not say platitudes that everyone could say, and really pulled information from what they believed this person would say.

Similarly, for debate I use a system prompt that depends on the grantmaker name and some other variable.

- **For debate**: "You are ${grantmaker}, a grantmaker in AI alignment. You're discussing with your colleague whether to fund the following grant

proposal. ${isFirstMessage ? "Provide your initial thoughts on the proposal." : "Respond to your colleague's points, then add your own perspective. Ensure you're addressing the most recent points made by your colleague."} Consider the proposal's potential impact, feasibility, and alignment with current priorities in the field. Be concise, direct, and focus on your unique expertise and viewpoint. Avoid general statements that any grantmaker could make."

-

Note that for the debate system prompt we use a ternary to have a different prompt for the first message and for responses.

Again, these system prompts are preprended to the grant proposal that we want to study to get the full prompt.

Dataset. To imitate grantmakers and researchers, we use data from multiple sources.

For grantmaking, we started with comments from Manifund on grants. We did not prepend the grant text, because we wanted to simply imitate comments to begin with, and because we thought that adding all of the proposals would make the context window become higher than the 200k token limit. However, after more thoughts there might be some ways to still make it work under the token limit by only choosing a few manifund projects.

Most of the Manifund grantmaker data was used for Austin Chen who has basically commented on most Manifund projects, and we wanted to use him as a more optimistic grant reviewer than other grantmakers, since Austin basically approved all of the grants on Manifund (that we have seen so far). We also wanted to try using different data sources.

The second data source that we used is LTFF reports, which you can find a breakdown here, with clickable links.

For the research methodology feedback, we use data written by Rohin Shah, Paul Christiano and Ethan Perez, specifically about research methodology or advice for AI Alignment researchers.

Here is the breakdown of the data by data size and tokens (note: the goal was to stay under 200k tokens, not have enough data for finetuning for now).

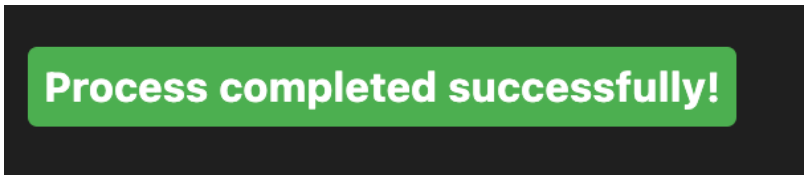
Filename	Dataset Size (KB)	Approx. Tokens
oliver_habryka.txt	238	47,600
austin_chen.txt	106	21,200
evan_hubinger.txt	31	6,200
ethan_perez.txt	44	8,800
adam_gleave.txt	48	9,600
rohin_shah.txt	61	12,200
paul_christiano.txt	30	6,000

Here's a summary of each file:

```
|adam_gleave.txt                    #Adam Gleave Manifund comments + LTFF writeups
|austin_chen.txt                    #Austin's Manifund comments (formatted)
|ethan_perez.txt                    #Ethan's post on advice for alignment research
|evan_hubinger.txt                  #Evan's Manifund commetns + LTFF writeups
|oliver_habryka.txt                 #Oliver's LTFF writeups
|paul_christiano.txt                #One post by Paul on research methodology
|rohin_shah.txt                     #One post by Rohin about AI Alignment Research
```

Details on the interface & process.

For how to run the extension, please check the [github repository](#).



The interface has three main buttons, but also has a:

- stop button that enables to stop the process
- some feedback (below the loading bar) that passes information like if the processes have been stopped, or if we need to select some text before startin g the tool
- some “update” colored indicator that tells you what the process is currently doing (doing fetches, doing summaries, and it changes to green when done)
- a loading bar (that would ideally fill to 100% but in practice we don’t know how long it takes to finish so it defaults to 10 seconds which is approximately how long the two first ones take)

-