Seamlessly Aligning LLMs with Online User Preferences (S.L.O.P.) for Creative Generation

Einar Balan

Abstract

Current alignment methods for large language models (LLMs)—e.g., RLHF and DPO—optimize for broad consensus, but this limits performance on creative tasks where individual taste is essential. I present S.L.O.P., a framework for scalable personalization in creative generation. My main contribution is a systematic study of soft prompts as a middle ground between resource-heavy fine-tuning and weak prompt-engineering baselines. Using Reddit-derived datasets, I show that soft prompts achieve strong adherence to user preferences and greater output diversity, elevating smaller models (Gemma-3-4B) toward parity with much larger models, with as few as 20-100 training examples. Objective metrics (embedding variance, cosine similarity) confirm these findings, while qualitative analysis highlights trade-offs in coherence at scale. A secondary contribution is a proof-of-concept platform integrating the explored LLM alignment methods directly with a seamless preference collection system. Together, these results demonstrate that lightweight, scalable personalization can make LLMs viable tools for subjective creative generation.

1 Introduction

An author wakes up from a terrible nightmare. In his dream he finds himself in a society where all of his work *must* cater to the widest possible audience.

He is no longer permitted to fill *his* niche, to think about things that *he* enjoys, or write about what brings *him* satisfaction. Something like that would never succeed! No, he must take care to inform himself on what is currently popular, adjust his own style accordingly, and never deviate beyond the safety of conformity.

His latest work is a new installment in the DCU (Depersonalized Cinematic Universe), featuring appearances from fan favorites like Captain Uninspired, The Homogenizer, and Generic Joe. It will be met with lukewarm response, but a warm one nonetheless (Lockhart, 2009).

Thankfully, we don't live in this world. After all, what is the point of art if not to be unique, interesting, and thought provoking? Unfortunately, it does represent what is essentially the current state of creative generation research in LLMs. These models are trained on mountains of content in an effort to gather an understanding of language and the world it describes, and it's clear that this approach has been highly effective for objective applications – things that have a correct answer. There are hundreds of benchmarks out there to prove it. However, the second you try a more subjective task, like creative writing, poetry, or comedy, the cracks start to show. It's very clear why: through various training methods like RLHF and DPO, the models learn to cater to everybody. For creative tasks, that just won't work.

In order to address this, I have tried to design a scalable method for aligning an LLM as closely as possible with an *individual's* preferences. To me, it seems that there are two critical aspects of this:

1. **Scalability**: It should be feasible to apply the personalization method at scale.
2. **Seamless Preference Collection**: It should be easy to collect reliable user preferences towards personalization.

In this work, I'll mostly address the first point, but I will touch on the second as well. To address the first, I'll explore and test a few different personalization methods.

2 Related Work

The initial inspiration to investigate creative generation in LLMs came from Jentzsch and Kersting (2023). They showcase the difficulty LLMs have with generating humorous content and some common pitfalls (repeating memorized jokes, nonsensical punchlines, etc).

This project is most heavily inspired by Ning et al. (2024), which aims to efficiently contextualize LLMs through the use of user embeddings. These embeddings are essentially tokens that uniquely identify a user and their conversation history. In effect, the LLM is provided with a space efficient representation of that user's conversation history, allowing for personalization of future output. The paper takes two approaches: one utilizing cross attention and the other utilizing soft prompts (also known as prompt tuning) to integrate the user embedding with the LLM. This same approach is also discussed in Doddapaneni et al. (2024). In the case of these works, the user embedding encodes plaintext preferences, such as restaurant ratings. These ratings can then be accessed via the embedding and used to generate recommendations for new restaurants. Liu et al. (2024) discusses a similar method.

Soft prompts are investigated in more depth in Lester et al. (2021) as well as Hebert et al. (2024). Briefly, a soft prompt is a prefix optimized through gradient descent that is appended to a prompt at inference time. Critically, many different soft prompts can be applied to a single LLM with minimal effort and resources.

A paper from Richardson et al. (2023) discusses using automatic preference summarization techniques for personalization.

3 Approach

I'm proposing a new framework for creative generation that allows for personalization to individual tastes, with an emphasis on scalability and seamless preference collection.

Ultimately, this framework will require an environment similar to those seen in platforms like Reddit, Netflix, TikTok, etc. These platforms are the ultimate preference collection machines, with user preferences being collected as a simple byproduct of using the platform. These preferences provide signficant insight into each user's cultural background and, if utilized correctly, could be integral to generating new content tailored to them.

To that end, I have built a simple proof of concept application that mimics this environment (though that is not the focus of this work). The preference collection mechanism is essential, but if there is not an effective personalization method then all of that will go to waste. Significant effort has gone into designing, implementing, and thoroughly testing several personalization methods.

3.1 Personalization Methods

To be successful in this application, a personalization method needs a few qualities:

1. **Lightweight**: As the method will be applied for many different users, a resource intensive process is not ideal.
2. **Flexible**: It should be easy to add and remove personalization from an LLM.
3. **Effective**: The method should produce content that adheres to the user's preferences and is generally coherent.

The first and second points immediately rule out finetuning as a personalization strategy and suggest an inference time approach is more appropriate. Unfortunately, inference time strategies (particularly prompt based approaches) are known to be far less effective than finetuning. Is there a happy medium between the two?

Soft prompts appear to meet that happy medium. They are lightweight (in comparison to fine tuning), requiring very limited training data, and very flexible in that many different soft prompts can be applied to the same model at inference time. See related work for more information on soft prompts.

In order to determine how effective a soft prompting approach would be, I ran extensive, data driven experiments outlined in the next section.

4 Experiments

Effectiveness and resource efficiency of personalization techniques seem to be in tension with each other. Ostensibily, soft prompts occupy the midpoint between these two, satisfying both. It's clear that they are lightweight, but the effectiveness will be detailed in the coming section. In particular, there are a few concrete qualities required for a personlization method to be considered effective:

1. **Adherence**: A majority of the content generated via the method must adhere to the preferences provided.
2. **Coherence**: The content generated must be intelligible and limit hallucinations.
3. **Variety**: The personalized model should be able to generate a wide range of content, rather than simple variations of the same text repeatedly.

It's useful to have a standard of comparison for the soft prompt approach, so we will also consider the following prompting strategies:

- Self Defined: The model is given a manually created description of topics of interest.
- Auto-summarized: The model is given an LLM generated summary of a list of posts that adhere to preferences.
- Preference History: The model is given direct access to a list of posts that adhere to preferences.

You can see examples of the prompts for these strategies in Appendix A. Each of these satisfy the first two criteria of an effective personalization method: they are very lightweight and flexible. In theory, they are not as effective as the soft prompting approach.

For soft prompting, I first test the ideal size of the training set to make sure we are training effectively. Then I move on to testing each of the personalization strategies.

4.1 Datasets

All of the preference data was sourced from Reddit. Much of this was gathered from a Kaggle dataset (which required extensive cleaning and filtering) and some of it was gathered directly from Reddit using one of their public endpoints.

I then categorized these posts into various topics of interest such as nerdy posts, posts about parenting, posts about UCLA, in order to then build a "profile" of sorts that could represent the interests of a user. All categorized data can be found here.

The categories of particular interest are: *minecraft, ucla, nostupidquestions, copypasta, nerdy, personal, pop, religion, tech, finance, amitheasshole, okbuddy, food, animals, pregnancy, parenting, baby, boomerhumor*. The contents of each of these is described in more detail in Appendix B.

There are also several "supercategories" consisting of posts from a few categories:

- unalike: posts from very different categories (pop, religion, tech)
- alike: posts from similar categories (tech, nerdy, finance)
- formatspecific: posts from categories that all follow a specific format (copypasta, nostupidquestions, amitheasshole)
- college: posts meant to mimic the interests of the stereotypical college student (ucla, nerdy, okbuddy, copypasta, pop, food, animals)
- newmother: posts meant to mimic the interests of the stereotypical new mother (pregnancy, parenting, baby, food, amitheasshole, pop, boomerhumor)

4.2 Soft Prompt Train Size Ablation

I tested across three axes: model size, dataset, and training set size. I opted for gemma-3-4b-it and gemma-3-27b-it for the models.

For the datasets, I stuck to categories focused on single topics. In particular: minecraft, ucla, nostupidquestions, and copypasta. Each of these datasets are very different from each other and provided useful information on how to train the overall best softprompt. The minecraft dataset is ultraspecific to a single game. It's very obvious if a model has been personalized to that interest. The ucla dataset is a bit more broad, but still quite specific to the interests of someone attending UCLA. The nostupidquestions dataset is quite broad but mainly tests stylistic adherence. Each of the posts in that dataset adhere to a very specific format and style. Finally, the copypasta dataset is more of a wild card. Many of the posts are quite incoherent, putting a lot of strain on the soft prompt to adhere to a very niche writing style and set of interests.

Finally, I tested across training sets of size 10, 20, 50, 100, 250, 500, 1000, and 2000.

Due to a lack of resources, I did not systematically analyze the posts here, instead opting for a qualititative approach. My finding was that, across the board 100 posts was enough to train a soft prompt with good adherence. Even further, for some of the more targeted datasets such as minecraft, 20 posts was all that was needed. This shows the significant benefit of soft prompts over fine tuning: they are significantly less resource intensive.

4.3 Personalization Strategies Ablation

Similarly to above, I tested across three axes here: model size, dataset, and personalization method. For the soft prompt training set, I included 100 posts as determined above.

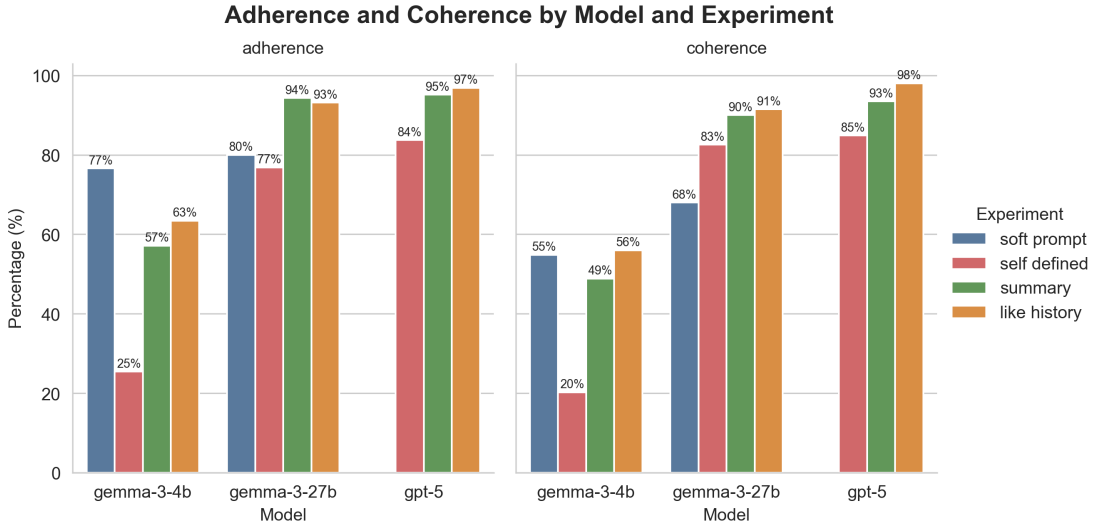

Figure 1: Adherence refers to how closely the generated output adheres to the provided preferences and coherence refers to the intelligibility of the text. Note that there was no soft prompt experiment for gpt-5.

For these experiments, I used gemma-3-4b-it, gemma-3-27b-it, and gpt-5. I could not test the soft prompt approach on gpt-5 as I only have access to the API, but it served as a useful standard of comparison for the prompt based approaches. In particular, I was interested to see how model size affected adherence, coherence, and variety of generated posts across the different datasets and personalization strategies. Note that I used the same hyperparameters (when applicable) including a temperature of 0.7 and top p sampling method with p = 0.9. I also added a repetition penalty of 1.1.

For the datasets, I used nerdy, personal, unalike, alike, format specific, college, and newmother. More information on datasets can be found in Appendix B.

As discussed earlier, the personalization methods include soft prompts, self defined interests, auto-summarized interests, and a list of posts exhibiting interests (which I will refer to as like history from this point).

Soft Prompt Training Process

As discussed, I settled on a training set size of 100 posts for each dataset. Each soft prompt consists of 64 tokens and I trained with a learning rate of 0.2 and a maximum of 1000 training steps. Note the steep learning rate, which is required to training soft prompts effectively.

4.4 Results

Effectiveness of a personalization method comes down to three qualities: adherence, coherence, and variety. Figure 1 shows the adherence and coherence rates grouped by model and personalization strategy. Interestingly, we see the soft prompt approach vastly outperforms the prompt based approaches for the smaller model on adherence, but lags behind for the 27b model. This suggests that the larger a model gets, the better its abiliy to interpet user's intent behind a prompt, leading to better adherence results.

Coherence showed a similar story, with the soft prompt approach being about on par with like history, but far outperforming the other 2 moethods on the 4b model. It also lags behind for the 27b model. These results are quite encouraging, because they show that a soft prompt can be used to bring the performance of a much smaller model to

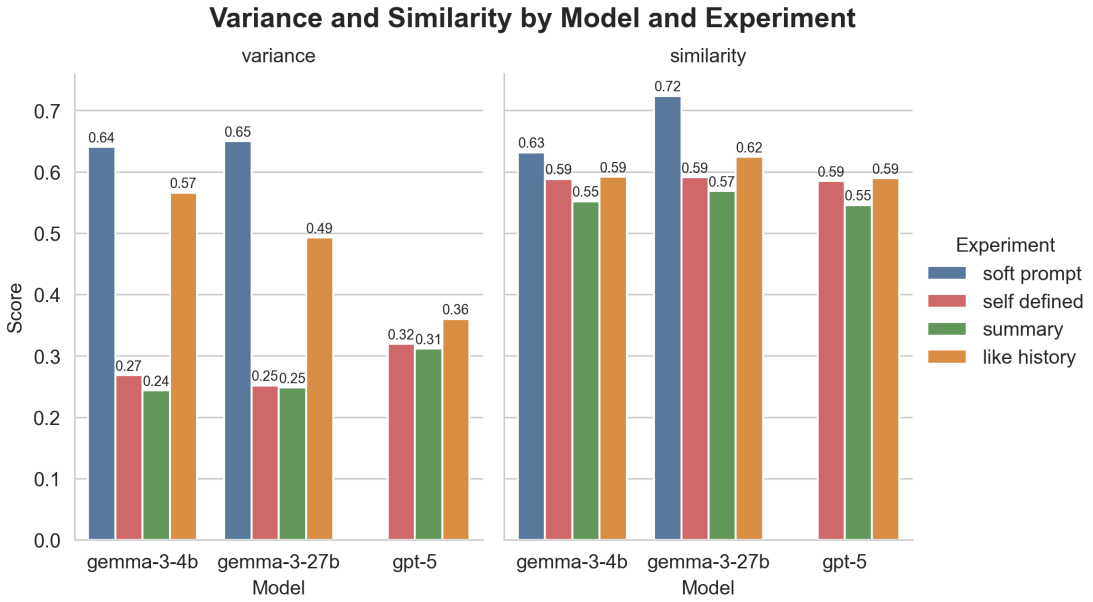

Figure 2: Variance measures the variability of the embeddings of the posts generated through a personalization strategy and similarity measures the cosine similarity of the mean embedding of the original dataset to the mean embedding of the generated posts. For each, higher is better.

around the same level as a larger model. Coherence struggles a bit on the 27b model, but it's likely this is a training issue.

It should be noted that the adherence and coherence data was all manually labeled (approx. 4000 posts). The labeler (me) didn't know which experiment each post belonged to, so no bias was in play there. Clearly, manually labeled data is not ideal, so we also include some objective metrics, namely embedding variance and cosine similarity to the original dataset. The embedding model used is sentence-transformers/all-MiniLM-L6-v2.

Embedding variance is useful to determine how much variety the personalization strategy is able to produce in its dataset. The larger variety of topics and styles we can produce, while still adhering to preferences and staying coherent, the better. Variance is useful to pinpoint a single number, but a visual approach can be helpful. To this end, I also clustered the embeddings for each approach to see how widely spread they are. These can be found in Appendix D. Ideally, we want minimal clustering for this application. A cluster usually indicates that a personalization strategy is producing many variants of the same post with minimal changes. It's possible that all these variants adhere to the preferences and are coherent, but the large quantity of variants produced unfairly inflates the adherence and coherence numbers. Therefore, it is important to take into account the variance and clustering numbers as well. We should note that for every model, the soft prompt approach produces the highest variety and lowest number of clusters. Every other approach suffers from low variety to some degree.

The cosine similarity metric serves a similar function to the adherence metric except that it is entirely objective. Interestingly, we see that yet again, the soft prompt approach outperforms all of the other methods, with a larger lead on the 27b model. This is in contrast to the manually gathered adherence results. Overall, these results seem to suggest that soft prompts provide the best personalization in terms of adherence and variety, with coherence struggling a bit. However, for smaller models it is the undisputed winner.

Additional Discussion

Some minor details of note:

- soft prompt generation was significantly faster compared to any of the other approaches (though I don't have concrete times)

- whenever the models attempted humor, no matter the personalization strategy, the coherence dropped. this is a big reason why the soft prompt approach appears to lack coherence as it attempted humor far more often than the other approaches (when relevant)

- stylistically, the soft prompt posts are vastly more similar compared to other approaches. I didn't account for that while labeling, but that is likely why we see the similarity/adherence discrepancy

There is a lot more to get into here in terms of fine grained analysis, but we'll omit that here.

5 Conclusions & Future Work

We have shown that soft prompting provides the best all around method for scalable personalization towards creative generation. What remains how is to demonstrate the seamless preference collection process. As mentioned, an environment to do so has been fully implemented and just needs to be tested. This environment would also provide valuable data indicating like rates for AI generated content and how well the content blends into to human generated content. Unforunately, to test it requires a lot of time and resources that I do not currently have.

The application mimics reddit with the twist that some of the posts served are LLM generated. These generated posts can be produced with any of the personalization mechanism presented here, or even no personalizatio method. As the user uses the app, we naturally collect their preferences for the topics and styles of content they prefer, as well as measure how well a given personalization mechanism performs. The user can interact with a post in three ways: an upvote, a downvote, or a "mark as AI." We can then compare each of the personalization strategies on these three metrics to determine performance.

Additionally, in retrospect some degree of stylistic adherence would have been useful to measure as well. It would also have been nice to automate the coherence metric with an LLM judge, though in my limited time I was not able to produce one with reliable results. With some finetuning and more time, it could be very doable and improve the scalability of the methodolgy. On that note, comparing the personalization methods to fine tuned models could also have been interesting, but again it was challenging to integrate given the time requirements.

It's clear to me that there is a lot of potential with this work. This is just one small babystep, but I envision a similar approach being utilized for more than just social media post generation. Any creative generation application could benefit: creative writing, video generation, or even music generation. Imagine a Netflix app where you can generate a movie perfectly tailored to your preferences on a whim. That's the potential I see. Clearly these are just baby steps, but it is an exciting direction!

6 Ethical Considerations

There is a lot of debate over AI generated art. Many worry that it could devalue human art and place the livelihoods of many artists at risk. I certainly understand that worry. It's one that could be applied to virtually any field. I would argue that art in particular, though, is safe. Most varieties of it, anyway.

The thing that makes art interesting is the story behind it. The human element. That's something you can't get with a solely AI generated work. There will always be something missing and it won't feel the same. There certainly is a time and a place for AI generated content, but I see it more so as a low stakes time waster, similar to the niche that social media like TikTok currently fills.

It can be uncomfortable to see someone generate an image in seconds that would have taken a human artist hours to create. On the whole, however, I think this is a good thing! It evens the playing field. Anyone can be an artist. Anyone can express themselves. As always, the good art will be recognized and the bad will be cast aside (in so far as there even is such a thing as good and bad art). Ultimately, I think it will be a net positive.

References

Sumanth Doddapaneni, Krishna Sayana, Ambarish Jash, Sukhdeep Sodhi, and Dima Kuzmin. 2024. User embedding model for personalized language prompting.

Liam Hebert, Krishna Sayana, Ambarish Jash, Alexandros Karatzoglou, Sukhdeep Sodhi, Sumanth Doddapaneni, Yanli Cai, and Dima Kuzmin. 2024. Persoma: Personalized soft prompt adapter architecture for personalized language prompting.

Sophie Jentzsch and Kristian Kersting. 2023. Chatgpt is fun, but it is not funny! humor is still challenging large language models.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning.

Jiongnan Liu, Yutao Zhu, Shuting Wang, Xiaochi Wei, Erxue Min, Yu Lu, Shuaiqiang Wang, Dawei Yin, and Zhicheng Dou. 2024. Llms + persona-plug = personalized llms.

Paul Lockhart. 2009. *A Mathematician's Lament: How School Cheats Us Out of Our Most Fascinating and Imaginative Art Form*. Bellevue Literary Press.

Lin Ning, Luyang Liu, Jiaxing Wu, Neo Wu, Devora Berlowitz, Sushant Prakash, Bradley Green, Shawn O'Banion, and Jun Xie. 2024. User-LLM: Efficient llm contextualization with user embeddings. *arXiv preprint arXiv:2402.13598*.

Chris Richardson, Yao Zhang, Kellen Gillespie, Sudipta Kar, Arshdeep Singh, Zeynab Raeesy, Omar Zia Khan, and Abhinav Sethy. 2023. Integrating summarization and retrieval for enhanced personalization via large language models.

A Appendix: Prompts

Table 1: Personalization Strategy Prompts

Personalization Strategy	**Prompt**
Like History	Please generate one reddit post similar to the following posts: title: I need a Chrons friendly, gluten free, and only hard cheeses recipe. self_text: A buddy of mine has a lot of digestive issues and he's never really had anyone cook for him so I wanted to treat him. I want something fancy as hell and I'm not worried about price of ingredients at all. I want something that's really gonna blow him away and impress him. I know he can eat chicken and any seafood but like selfish do give him a minor reaction. Red meat is completely off the table. subreddit: Cooking [...] It is critical that you stick to this exact format. DO NOT ADD ANYTHING ELSE. \n\ntitle: {title}\n self_text: {self_text}\n subreddit: {subreddit}
Self Defined	Please generate one reddit post. Here are some topics of interest: gaming, pcs, tech, mmos, rpgs, anime. Pick exactly ONE topic per post. It is critical that you stick to this exact format. DO NOT ADD ANYTHING ELSE. \n\ntitle: {title}\n self_text: {self_text}\n subreddit: {subreddit}
Soft Prompt	Please generate one reddit post. It is critical that you stick to this exact format. DO NOT ADD ANYTHING ELSE. \n\ntitle: {title}\n self_text: {self_text}\n subreddit: {subreddit}
Summary	Please generate one reddit post. Here is a summary of the user's interests: Here's a concise profile of interests inferred from the posts they liked: - Home cooking and food science: practical technique tips (stainless steel searing, pan sauces), balancing flavors (sweetness/acidity in Middle Eastern dishes), creamy vs béchamel gratins, browned butter hacks, mashed potatoes, pasta salad takes, food safety, and specialty diets (gluten-free/Crohn's-friendly). Also into kitchen tools (tweezers) and bouillon/seasoning ideas. - Pets and pet care: Dogs: durable beds, hypoallergenic breeds for families, washing donut beds, ESA housing rules, puppy training/over-arousal and separation-anxiety protocols, gear (bowls), adoption fit (Yorkie-poodle). Cats: eye/medical issues, UT prescription diets and treat rules, behavior/being held, leash/harness acclimation. Pick exactly ONE topic per post. It is critical that you stick to this exact format. DO NOT ADD ANYTHING ELSE. title: {title} self_text: {self_text} subreddit: {subreddit}

B Appendix: Datasets

Here's a brief description of each category in the reddit dataset.

- minecraft: posts related to minecraft
- ucla: posts related to UCLA and college life in general
- nostupidquestions: posts from the subreddit r/NoStupidQuestions, ranging a variety of topics in a specific format
- copypasta: posts from the subreddit r/copypasta ranging a wide variety of topics, but often centered around crude humor and repetition
- nerdy: posts related to gaming, pc building, anime, and other nerdy interests
- personal: posts related to personal issues such as relationship advice and mental health
- pop: posts related to pop culture, tv, movies, sports, etc.
- religion: posts related to religion and spirituality
- tech: posts related to consumer technology, hardware, and software
- finance: posts related to finance, stocks, and investing
- amitheasshole: posts related to interpersonal issues, advice
- okbuddy: posts related to crude, gen z humor
- food: posts related to food, cooking, etc.
- animals: posts related to pets, cats, dogs, animals, fish
- pregnancy: posts related to pregnancy
- parenting: posts related to parenting
- baby: posts related to babies
- boomerhumor: posts related to traditional jokes e.g. puns, knock knock, etc.

C Appendix: Metrics Tables

Table 2: Adherence/Coherence/Unique for gemma-3-4b-it

Dataset	Soft Prompt			Self Defined			Summary			Like History		
nerdy	88	70	0	2	0	0	26	64	0	90	78	0
personal	74	64	6	48	38	0	52	40	0	42	32	0
unalike	82	60	6	2	2	0	96	94	0	32	32	0
alike	84	78	44	80	68	0	62	48	0	98	98	14
formatspecific	76	40	2	42	30	0	98	48	0	50	42	0
college	78	66	12	0	0	0	2	2	0	64	58	8
newmother	54	6	2	4	4	2	64	46	0	68	52	4

Table 3: Variance for gemma-3-4b-it

Dataset	Soft Prompt	Self Defined	Summary	Like History
alike	0.666	0.188	0.175	0.463
college	0.700	0.310	0.098	0.703
formatspecific	0.673	0.269	0.250	0.630
nerdy	0.581	0.145	0.377	0.373
newmother	0.574	0.255	0.306	0.632
personal	0.590	0.632	0.391	0.538
unalike	0.702	0.080	0.108	0.623

Table 4: Similarity to Original for gemma-3-4b-it

Dataset	Soft Prompt	Self Defined	Summary	Like History
alike	0.594	0.667	0.526	0.581
college	0.573	0.565	0.361	0.700
formatspecific	0.736	0.538	0.658	0.671
nerdy	0.770	0.618	0.623	0.296
newmother	0.428	0.614	0.528	0.562
personal	0.786	0.772	0.696	0.823
unalike	0.534	0.345	0.472	0.510

Table 5: Adherence/Coherence/Unique for gemma-3-27b-it

Dataset	Soft Prompt			Self Defined			Summary			Like History		
nerdy	86	74	4	98	100	0	78	82	0	94	98	6
personal	86	78	16	100	98	6	98	100	0	100	98	6
unalike	76	62	8	98	88	4	100	94	2	78	74	2
alike	86	84	14	92	94	2	100	100	0	98	98	20
formatspecific	86	80	6	4	0	0	94	68	2	100	96	6
college	82	58	28	46	98	0	100	92	2	84	78	16
newmother	58	40	18	100	100	0	90	94	0	98	98	6

Table 6: Variance for gemma-3-27b-it

Dataset	Soft Prompt	Self Defined	Summary	Like History
alike	0.633	0.269	0.217	0.587
college	0.721	0.185	0.212	0.657
formatspecific	0.739	0.312	0.229	0.355
nerdy	0.633	0.234	0.337	0.432
newmother	0.524	0.143	0.219	0.360
personal	0.575	0.294	0.204	0.401
unalike	0.726	0.325	0.321	0.658

Table 7: Similarity to Original for gemma-3-27b-it

Dataset	Soft Prompt	Self Defined	Summary	Like History
alike	0.763	0.744	0.593	0.680
college	0.708	0.419	0.377	0.481
formatspecific	0.693	0.496	0.623	0.717
nerdy	0.847	0.607	0.629	0.512
newmother	0.478	0.538	0.677	0.715
personal	0.901	0.646	0.582	0.640
unalike	0.675	0.688	0.501	0.624

Table 8: Adherence/Coherence/Unique for gpt-5

Dataset	Self Defined			Summary			Like History		
nerdy	98	100	0	78	82	0	94	98	6
personal	100	98	4	98	100	0	100	98	4
unalike	98	88	4	100	98	0	90	100	0
alike	100	100	4	100	100	0	100	100	10
formatspecific	10	10	0	100	80	0	100	96	0
college	80	98	0	100	100	0	96	96	4
newmother	100	100	0	90	94	0	98	98	10

Table 9: Variance for gpt-5

Dataset	Self Defined	Summary	Like History
alike	0.152	0.093	0.265
college	0.495	0.274	0.466
formatspecific	0.406	0.443	0.504
nerdy	0.294	0.296	0.259
newmother	0.376	0.333	0.362
personal	0.358	0.340	0.368
unalike	0.157	0.402	0.298

Table 10: Similarity to Original for gpt-5

Dataset	Self Defined	Summary	Like History
alike	0.653	0.634	0.692
college	0.561	0.372	0.308
formatspecific	0.660	0.170	0.683
nerdy	0.603	0.607	0.347
newmother	0.304	0.717	0.727
personal	0.698	0.662	0.750
unalike	0.617	0.658	0.620

D Appendix: Charts

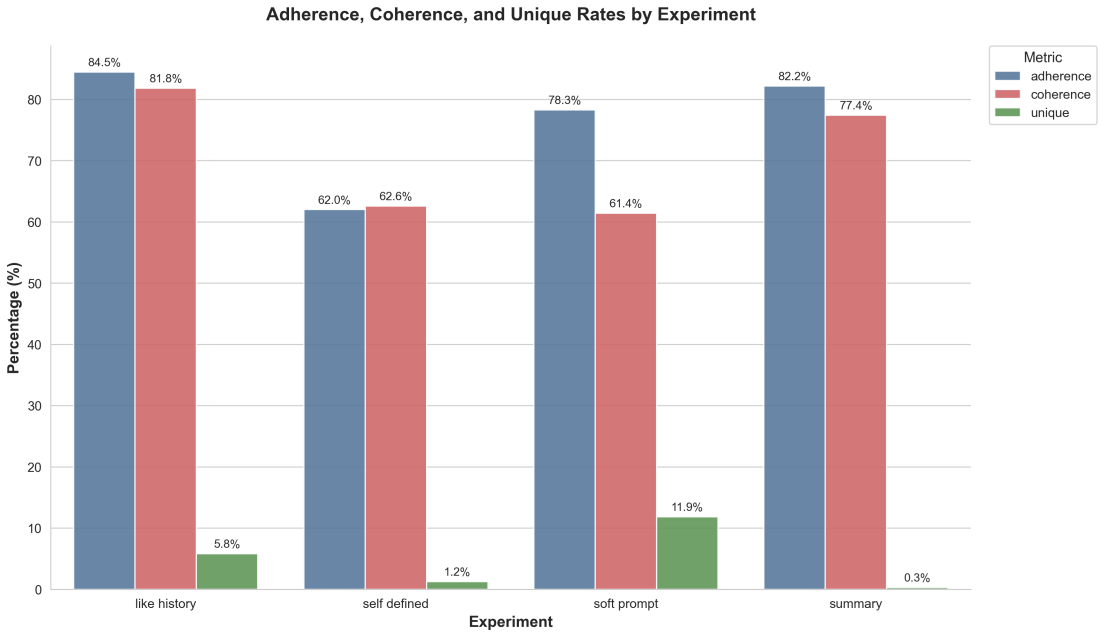

Figure 3

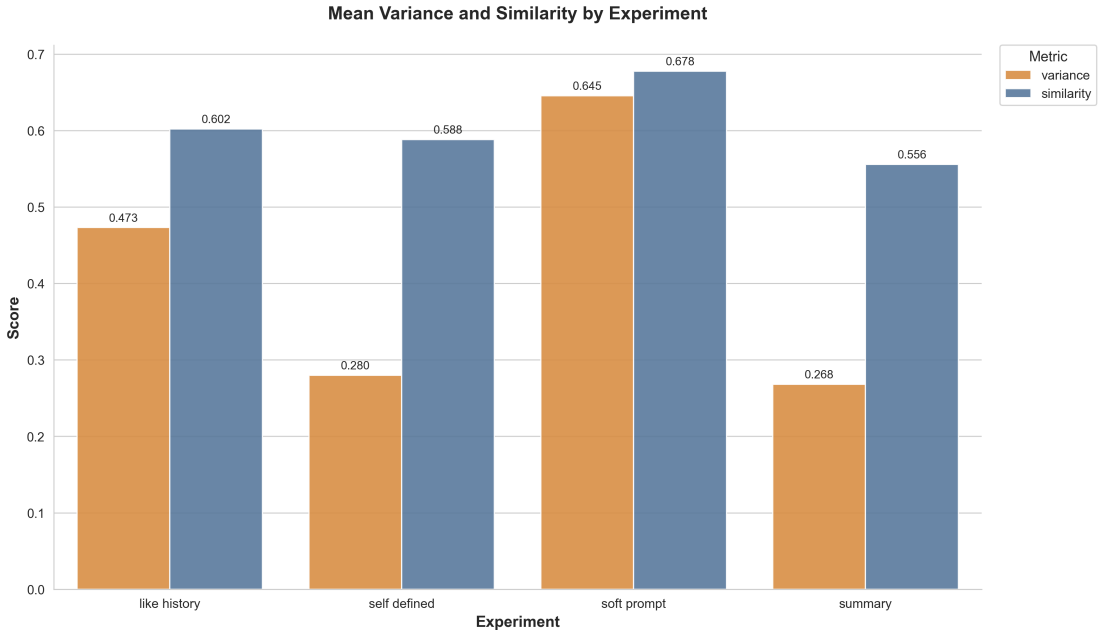

Figure 4

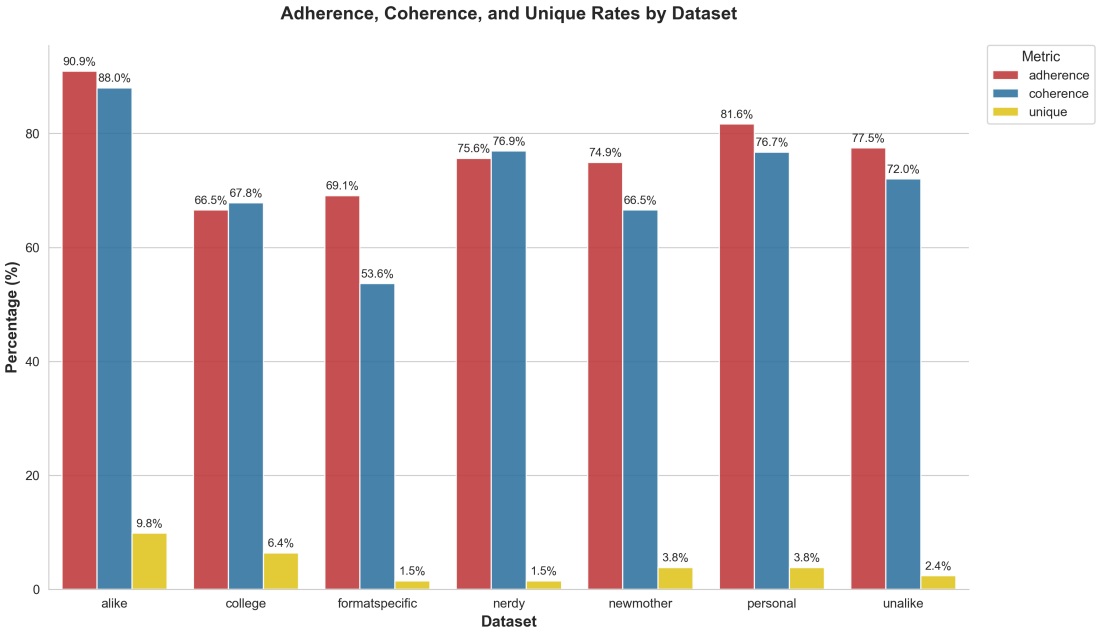

Figure 5

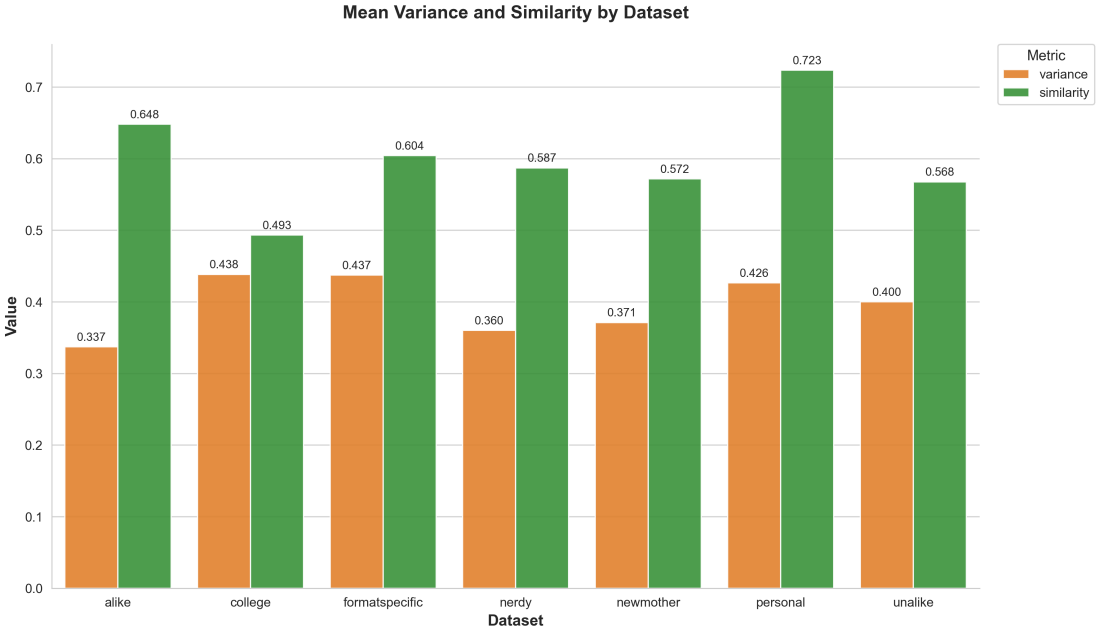

Figure 6

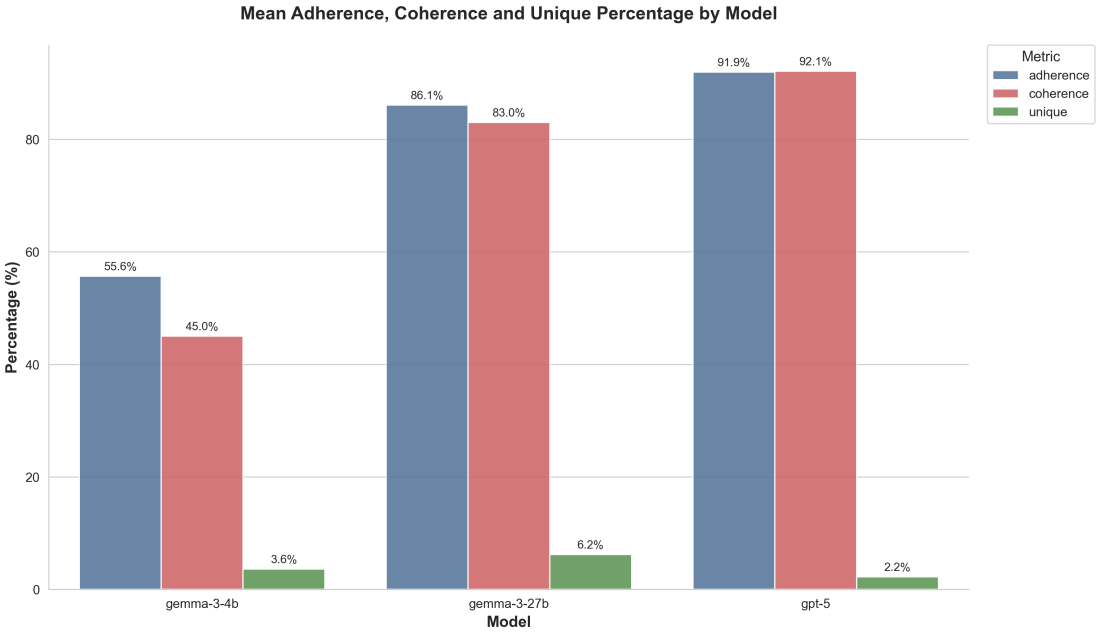

Figure 7

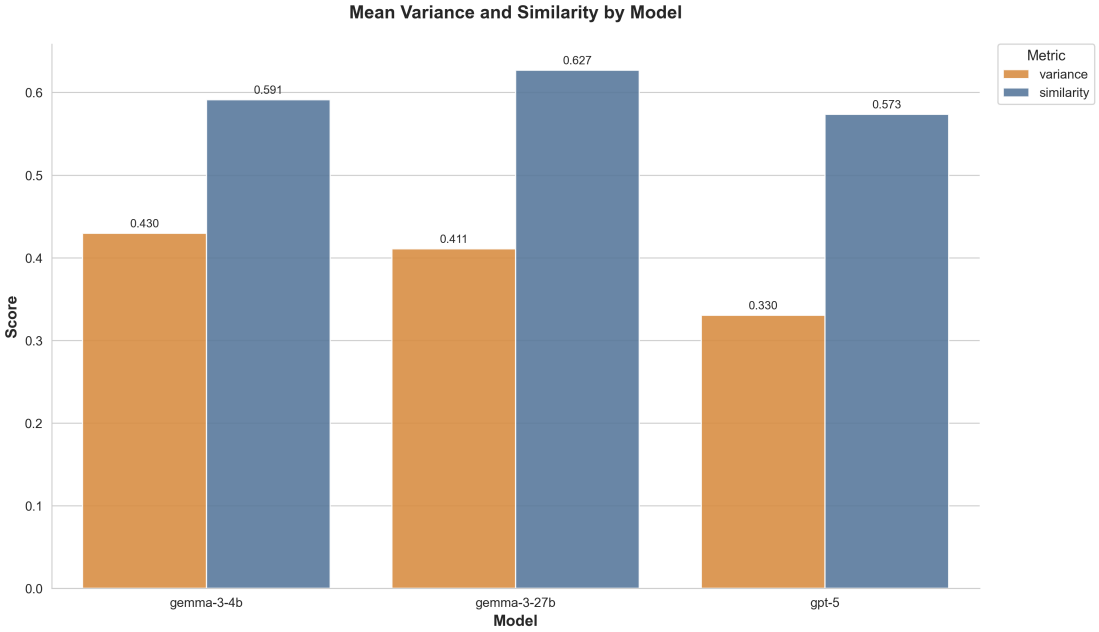

Figure 8

E Appendix: Embedding Clustering

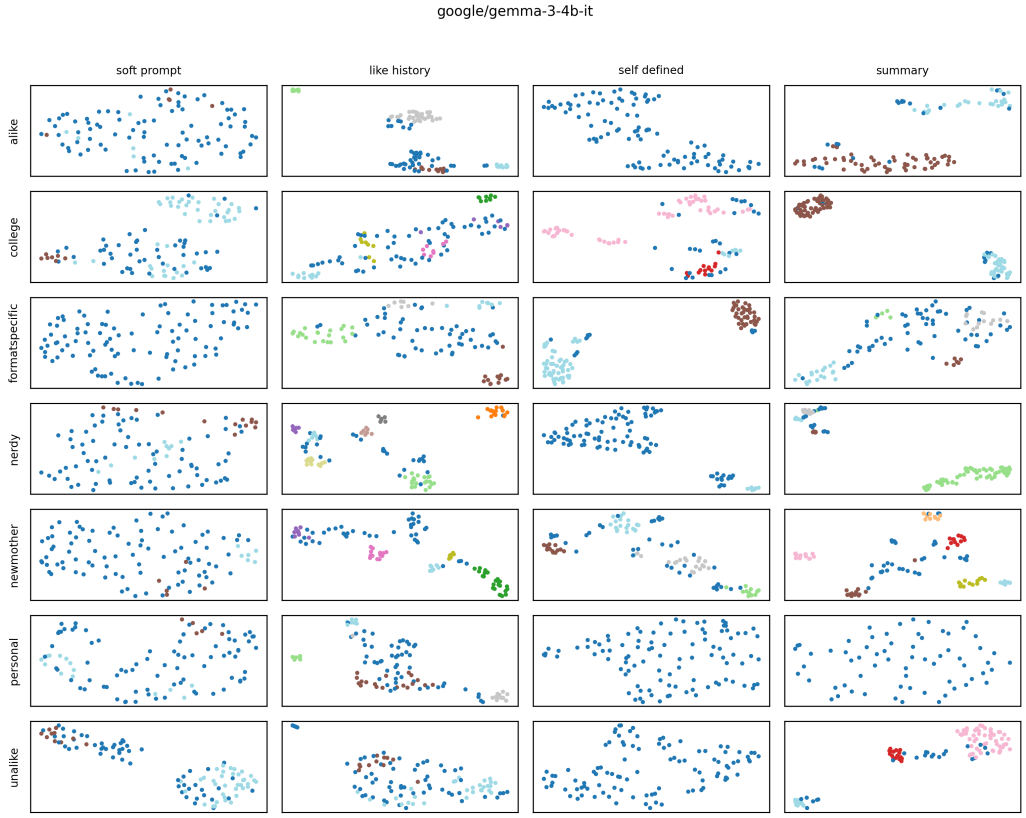

Figure 9: The clustering results show the large variety of posts that the soft prompt approach is able to generate. On the other hand, we see that summary is quite weak in this regard, often only generating small variations of the same 2 or 3 posts.

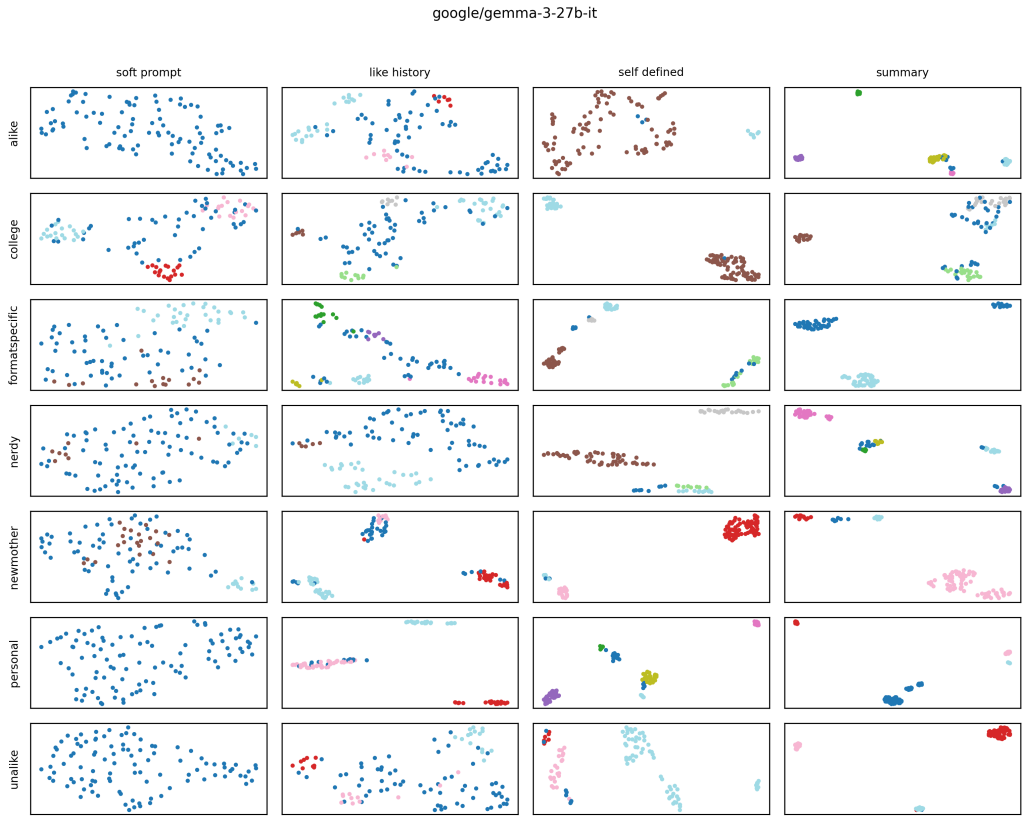

Figure 10: The clustering results show the large variety of posts that the soft prompt approach is able to generate. On the other hand, we see that summary is quite weak in this regard, often only generating small variations of the same 2 or 3 posts. Interestingly, this problem is even more pronounced for the larger model compared to its 4b counterpart.

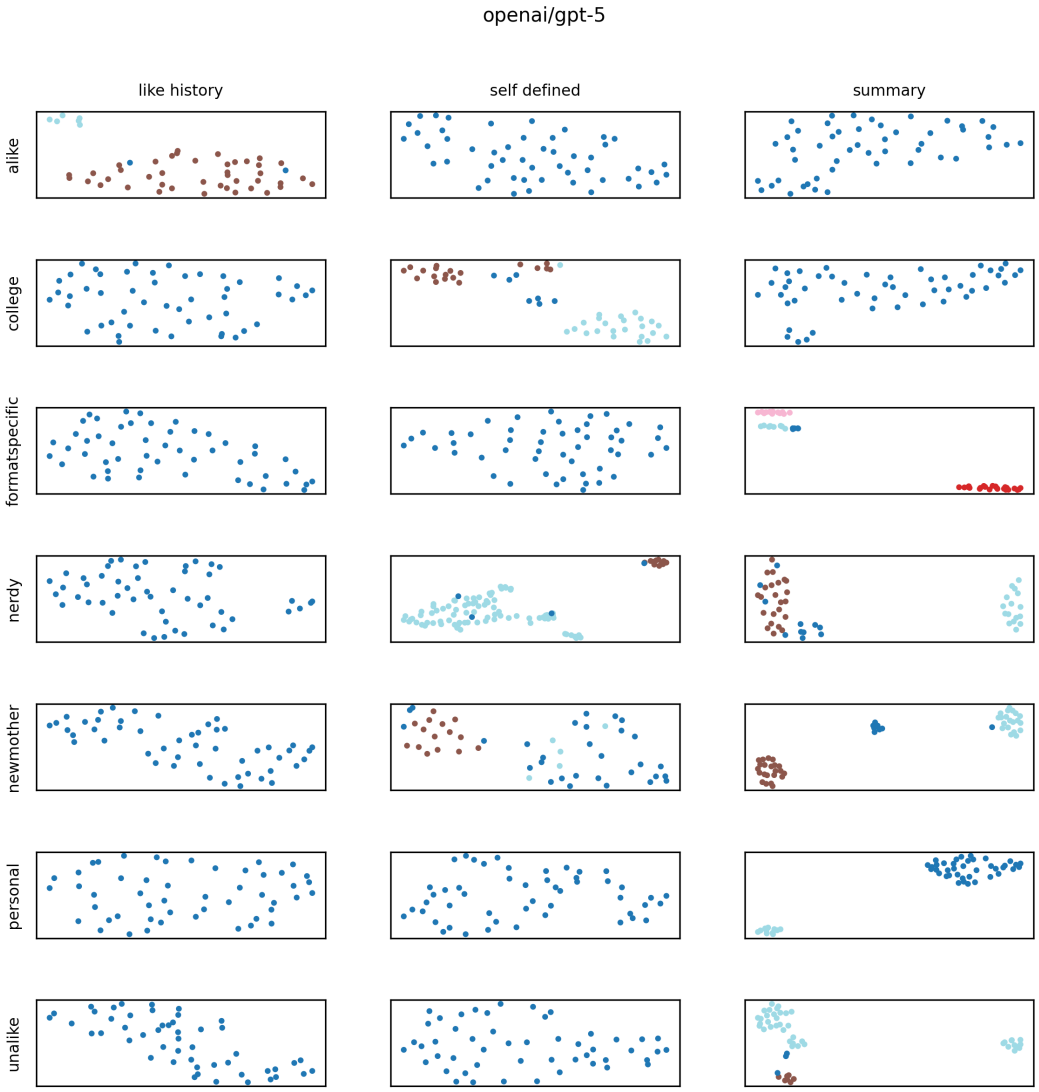

Figure 11: We see similar clustering results here, though not as pronounced as in the gemma models (in most cases).