Consistent Recursive Augmentation

Ethan Edwards

June 17, 2024

Abstract

Consistent Recursive Augmentation (CRA) is an LLM technique for creating custom databases of written content that informs the background of a desired LLM personality. This database is then used to inform the LLM's behavior via RAG. Fictional characters or specific personalities with consistent and expansive backstories can be created very quickly, with optional human curation giving creators the power to ensure a high quality of authored content. Beginning with some simple guidelines, a fully realized character with a novel-length backstory can be created automatically, with their personality and history growing continuously with user-interaction. This technique has already been used for the creation of several LLM characters and has wide applications in interactive experiences, entertainment, and certain specialized assistant domains.

1 Introduction

Large Language Models offer new possibilities for interaction with consistent conversational entities. Most commercially deployed LLMs have been trained with RLHF in order to encourage conversational structure with one partner, usually known as the "assistant", which has been given a compelling user-facing personality with the option to customize this personality via prompting and fine-tuning. However, much of the factual information informing the LLM's version of the world comes from relatively deep structures which are difficult to change without significant additional training. While often the information a commercial LLM produces by default is very high-quality, the information is based only on the general training data and is still prone to inaccuracy and hallucination. In addition, for those creating experiences which require consistency of character, responses are inconsistent over time as they are generated anew at inference time. A common technique for mitigating this is Retrieval Augmented Generation (RAG) [5] which uses a custom database of desired facts, either from a proprietary database or human-curated trustworthy information, and retrieves any relevant information from the database to the LLM at inference time. RAG has been widely deployed for this purpose, although it is limited by the availability of data and the reliability of retrieval techniques, and

can only answer with the desired consistency and accuracy on topics covered in the database.

RAG is mostly used for LLM applications where a suitable database already exists. A question-answering LLM assistant can replace a normal QA document by ingesting the document as a RAG database for example.

However, LLMs using RAG can also create suitable databases themselves. I introduce Consistent Recursive Augmentation which uses RAG and carefully scaffolded infrastructure to create and improve databases for use with RAG. It augments the database with additional items that are consistent with prior data, leading to an organic expansion, and it can perform this process recursively, using the already accepted data to generate more.

2 Related Work

CRA is a method of synthetically generating RAG data, which draws from previous attempts to use RAG for factual purposes. [5] [8] [9] Using scaffolded LLMs to create fictional content has been probed in some depth. Park et. al created simulated reflective agents, including a full memory of past interactions which can be retrieved during real time interactions. [7] Using LLMs to classify, edit, and improve LLM outputs has notably been used in Anthropic's Constitutional AI system, which is then used as a database for Reinforcement Learning from AI Feedback. [1].

The usefulness of synthetic data for improving LLMs has mostly been discussed in the context of large training-datasets; nevertheless these debates still apply to other applications of synthetic data such as CRA. While some research suggests that using generated data will lead to model collapse, [10] recent work indicates that an accumulation of synthetic data mixed with real data mitigates many of the issues. Policy work on the availability of data in general suggests that synthetic data generation methods are likely to become prominent over time. [12] Using synthetic data in the context of more targeted contexts such as inference-time dialogue generation has been explored in papers such as Bao et al. indicating significant promise applied to factual use-cases. [2]

3 Method Summary

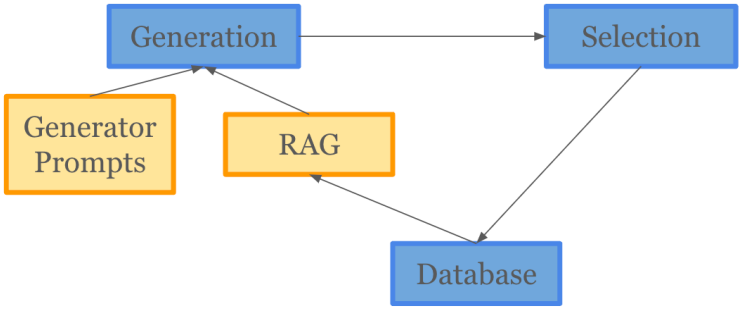

Figure 1: Diagram of the Consistent Recursive Augmentation process

Consistent Recursive Augmentation (CRA) begins with an initial dataset, which may be as small as a simple prompt describing the basic criterion of the desired LLM personality, and proceeds through successive stages of 1) generation and 2) selection before adding new content to the database and repeating the loop. First the initial information is used to prompt an LLM to generate new points of data. For example, an LLM personality might be a young man named Ade from Lagos in 2035, intended to supplement a fictional narrative portraying the city after significant population growth and environmental change. The generator LLM might be prompted to answer questions such as "Where is he from?" "Who are his family members?" "What is his job?"

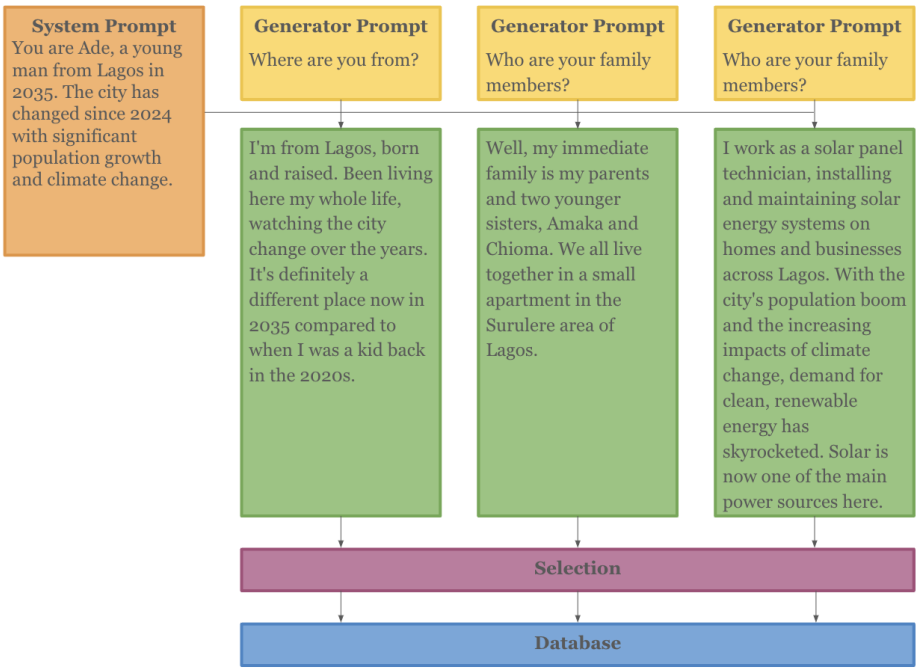

Figure 2: Example of initial generation process.

Possible answers to these questions are generated and then either accepted or rejected based on some selection criterion, the selection stage. Once certain pieces of information are selected, they are ingested into the database and can be used to inform the generation of additional content for the database. So if in prior stages it was determined that Ade was born in Lagos, lives with his family, and he works in the solar industry, the next generation would be informed by these facts and might incorporate them into a description of his routine. This "average day" can then also be admitted into the database and inform future answers. This process can continue recursively and scaling is only restricted by the selection criterion and the creative possibilities of the LLM and prompts.

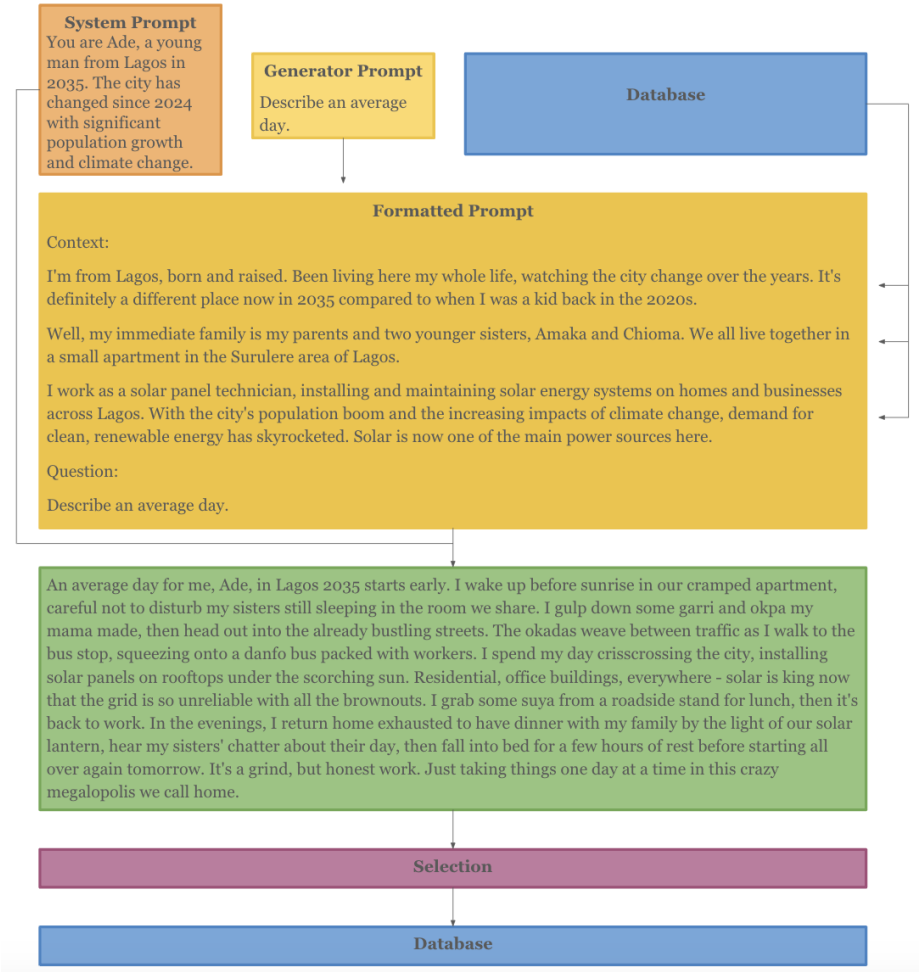

Figure 3: Example of generation process using the RAG database.

The database-informed generation, selection, and database augmentation can be continued indefinitely, ideally done in hierarchical stages of prompting the generator according to the needs of the project. The process can be continued even after deployment, incorporating user-prompted generation as a basis for additional CRA.

4 Generation

CRA can work with any number of text-generation schemes, and need not even use an LLM. Nevertheless, there are certain patterns that work particularly well. The simplest pattern is to use the in-progress LLM personality as the generator

itself and prompt it according to an approximation of eventual user prompts. In the above example, Ade would begin with a simple LLM system prompt and be asked the questions directly, with temperature and other settings adjusted according to desired level of creativity. In later stages, powered by RAG injections, the user prompt would likely be scaffolded with not only the generator prompt, but instructions for how to incorporate the retrieved information i.e. "make sure to be consistent with the information and style" or "answer with colorful personal anecdotes based on your background." If questions are also stored in the RAG database, these can be useful as injections for few-shot prompting [3] for generation, ensuring further consistency of style and content.

4.1 Generator Prompts

The actual prompts used for generation are quite important for the success of CRA. They need to elicit sufficiently creative and informative responses to build the database, to produce completions useful for eventual user interactions, and to be ordered logically so that more core facts are determined first before secondary details.

Needs will vary based on each project, but for character-based CRA the default pattern should be 1) core-characteristics 2) lifestyle and habits 3) major relationships 4) general opinions 5) anecdotes and stories. This ensures that core personality traits or major people are not only present in a single story, and are consistently referenced as they would be in conversation with a consistent and coherent character.

Generator prompts require some human attention, but do not need to be produced entirely manually. LLM generation of prompt lists with careful prompt-engineering and topic selection has proven to be very effective for this stage of CRA and should be the default option for most applications.

4.2 RAG for Generation

Key to the recursive nature of CRA is that the generator stage combines generator prompts with previously generated and selected content. Any sort of RAG will improve the consistency and results of this stage, however there are still considerations that might go into the selection of retrieval method. Retrieval methods are still an active area of research and selection of method will vary based on task. [11]

The most basic is a standard similarity criterion through some combination of keyword search and vector-based similarity [13]. This especially helps with consistency, as relevant information to the present subject-matter will be retrieved. The amount of content retrieved for each generation is another important factor, and will vary based on how much of what has already been written on a subject is necessary for consistency. When CRA is run offline and LLM context window size is not an issue, more is mostly better, although most current use-cases have retrieved 3-5 items.

There may also be reasons to retrieve and inject content which would not immediately score highly on similarity. Additional retrieval processes for specific subjects i.e. other characters, anecdotes, may make the connections between subjects more nuanced and compelling.

4.3 Batching

Adding to the database and beginning an additional cycle can be done individually or with batch patterns based on the specific use case as well as the generation and selection methods chosen. If the generation subject matter is likely to produce contradictory answers (for example a character's family members), then this subject matter is not appropriate to be handled in a single selection batch as each new generation will have to build on the other accepted ones. However if generating anecdotes which all draw from the same data on location, interests, and acquaintances, these can be handled in a batch together. The most effective batching strategy will change according to project specifics and the logic of the generator prompts and associated completions.

5 Selection

Selection methods will vary widely based on the requirements of the specific project and the output quality of the generation method used. Selection may be skipped entirely if the generation method is consistent and trusted enough for the purposes, however it is usually best practice to have an additional layer.

5.1 LLM-based selection

LLMs or other NLP-based classifiers can be used for selection, approving responses based on appropriateness, consistency, and alignment with the general project goal. Generator programs, even with appropriately instructed, can often produce content that is not appropriate, and an additional layer of LLM checking with different instructions is often extremely helpful. LLMs can also be instructed to score or rank the various generated answers and return the top answer. [4]

5.2 Human-in-the-loop selection

CRA works extremely well with human-the-loop processes of selection. As selection ultimately comes down to a binary choice, it can still be efficient for a human to approve the various responses. Large amounts of disapproval are also a useful signal that the generator program or prompts should be modified. It may also make sense for a human supervisor to approve several answers in particular topic domains, or focus their selection on early stages of CRA before there is a great amount of content to ensure consistency.

5.3 Deployed selection

If selection is trusted enough, CRA can also be used in live-deployed systems to ensure a consistent and evolving user experience. Previous conversational threads can be stored, and even be used as the basis for further CRA-augmentation between sessions, giving the user experience of a character actively learning and changing. The major difficulty is in developing selection processes that can be run with a sufficiently high degree of confidence, as user interactions will have the most unpredictable content and poor selection has the potential to degrade future CRA content. Multiple layers of automated classifiers are the best option for this case.

6 Applications

CRA is useful in any situation where additional generated content is a desirable background for an LLM personality. While this precludes purely factual or informational assistants, even small details of background information or preferences can benefit from CRA. The many easter-eggs common in voice assistants such as Amazon's Alexa or Apple's Siri [6] suggest that there is demand for creative content even in systems which are primarily factual in use.

Any sort of independent character can be significantly improved via CRA. This can include personalities intended to illustrate factual situations i.e. a representative customer based on demographic and trend research, or a political constituent whose personality has been constructed according to careful issue polling. It can also include purely fictional characters, providing narrative content through conversational interaction.

Another application of CRA is to create facts about a fictional world as background which can be used by LLM programs or multiple individual characters. Using a slightly different generator program a shared background of history, events, and narratives can be generated which several characters can consistently share in common, rather than focusing entirely on a single character.

Further applications of CRA will be determined by the increased real world use of LLMs. While the applications are currently best for areas where consistent creative output is desired, if LLMs become more reliable in deducing information or making decisions based on data, CRA and related techniques might also be used for research augmentation and general predictive processes.

7 Conclusion

CRA can be an extremely useful tool for any creator of scaffolded LLM systems. The flexibility of generation and selection methods allows for a high degree of control for automatic content generation. It is likely that new LLM techniques i.e. new forms of RAG, enhanced generation or selection programs, will be able

to be incorporated into the CRA framework for greater performance and new creative possibilities.

References

[1] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, and J. Kaplan. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[2] Jianzhu Bao, Rui Wang, Yasheng Wang, Aixin Sun, Yitong Li, Fei Mi, and Ruifeng Xu. A synthetic data generation framework for grounded dialogues. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10866–10882. Association for Computational Linguistics, 2023.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, and Dario Amodei. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[4] Michael Desmond, Zahra Ashktorab, Qian Pan, Casey Dugan, and James M. Johnson. Evalullm: Llm assisted evaluation of generative outputs. In *Companion Proceedings of the 29th International Conference on Intelligent User Interfaces*, IUI '24 Companion, page 30–32, New York, NY, USA, 2024. Association for Computing Machinery.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[6] Carrie Marshall. The best siri easter eggs: how apple's voice assistant can entertain you. *TechRadar*, September 25 2021. `https://www.techradar.com/news/the-best-siri-easter-eggs-how-apples-voice-assistant-can-entertain-you`.

[7] J. S. Park, J. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22, October 2023.

[8] M. R. Parvez, W. U. Ahmad, S. Chakraborty, B. Ray, and K. W. Chang. Retrieval augmented code generation and summarization. *arXiv preprint arXiv:2108.11601*, 2021.

[9] S. Sharma, D. S. Yoon, F. Dernoncourt, D. Sultania, K. Bagga, M. Zhang, and V. Kotte. Retrieval augmented generation for domain-specific question answering. *arXiv preprint arXiv:2404.14760*, 2024.

[10] I. Shumailov, Z. Shumaylov, Y. Zhao, Y. Gal, N. Papernot, and R. Anderson. The curse of recursion: Training on generated data makes models forget. *arXiv preprint arXiv:2305.17493*, 2023.

[11] Daniel Thulke, Niklas Daheim, Clémentine Dugast, and Hermann Ney. Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog. *arXiv preprint arXiv:2102.04643*, 2021.

[12] Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Position: Will we run out of data? limits of LLM scaling based on human-generated data. In *Forty-first International Conference on Machine Learning*, 2024.

[13] Alan J. Zitzelberger, David W. Embley, Stephen W. Liddle, et al. Hykss: Hybrid keyword and semantic search. *Journal on Data Semantics*, 4(4):213–229, 2015.