Interim report: style model comparison, consensus author selection, and LLM cost estimates

Authorship Mimicry Using Generative AI – AAVC pipeline update

1. Background and objectives

The overall goal of this phase was to choose a robust style representation for authors in the Amazon Authorship Verification Corpus (AAVC) and to identify a stable set of authors to carry forward into the generative (LLM) stage of the project.

Concretely, the objectives were:

1. To embed all authors' reviews with six different style models and compute author-level style consistency.
2. To compare the models both numerically and visually, including global 2D t-SNE projections and "worst-case" author plots.
3. To quantify agreement and disagreement between models and define a consensus set of "stylistically stable" authors.
4. To estimate API costs for the next step, where we will train or condition an LLM on selected authors and generate new reviews.

The sections below summarise what has been completed so far and how this leads to a shortlist of 150 authors for the next stage.

2. Data and corpus

The corpus used is the "mixed topics per author" variant of the AAVC. Each author has a directory named by their Amazon author ID (for example, A1A1BM6N28X9J0). Inside each directory are multiple .txt files, each containing one review, with filenames that encode the product category (for example, _Automotive.txt, _Beauty.txt, and so on).

A separate metadata file lists all target author IDs along with the product categories they have written in. For each of these authors, the pipeline:

- Enumerates all .txt review files in that author's directory.

- Reads all non-empty reviews.
- Embeds every available review for every model.

All resulting embeddings, consistency scores, and visualisation artefacts are stored in a structured data/ folder so that downstream experiments can be reproduced.

3. Style models and embedding implementation

Six style models from HuggingFace are currently in use:

Family	Model name	Notes on use
LUAR (original)	rrivera1849/LUAR-CRUD	Custom forward, episode-style encoder
	rrivera1849/LUAR-MUD	As above
LUAR (ST variants)	gabrielloiseau/LUAR-CRUD-sentence-transformers	Standard sentence-transformers interface
	gabrielloiseau/LUAR-MUD-sentence-transformers	As above
Other style models	AnnaWegmann/Style-Embedding	Style-specific encoder, pooled output used
	AIDA-UPM/star	RoBERTa-based STAR encoder, pooled representation

Implementation details:

- For the two original LUAR models, the HuggingFace config and custom forward are respected. The models are instantiated with trust_remote_code=True, and their episode-based forward method is used with a single-comment “episode” per review so that each review yields one style vector.
- For the LUAR sentence-transformer variants and the Style-Embedding model, the standard encode interface is used with batching, and the pooled sentence representation is taken as the style embedding.
- For STAR, the pooled output provided by the model is used rather than raw hidden states, following its documented usage.
- Each model is loaded once per run and reused across all authors, and computation is run on GPU (Apple MPS) where possible, with a small patch for LUAR’s pooling to avoid MPS type issues.

For each model and author, the script produces a compressed file:

data/embeddings/<model_key>/<AUTHOR_ID>.npz

containing the author ID, an (n_reviews × dim) embedding matrix, and the list of source file paths.

4. Consistency scoring and top-K author selection

For each model, author-level style consistency is defined as follows.

For a given author:

1. Let E be the matrix of their review embeddings with shape (n_reviews × d).
2. Compute pairwise cosine distances between all rows of E.
3. For each review, compute its mean cosine distance to all other reviews by the same author.

There are then two cases:

- If the author has fewer than six reviews, a mean style distance is still computed for reference, but the author is excluded from the ranked "most consistent" list.
- If the author has at least six reviews, the six reviews with the smallest mean distance (the tightest internal cluster) are selected. A centroid is computed for these six points, and the author's style consistency score is defined as the mean cosine distance of the six points to this centroid. Lower values indicate higher internal stylistic consistency.

This procedure is applied independently for each model. Initially, the ranking was computed for the top 100 most consistent authors per model. This has now been extended to the top 300 authors per model:

- For each model, authors are sorted in ascending order of mean style distance.
- The top 300 most consistent authors are retained as that model's "top-300" set.
- For each of these authors, the following are recorded per model: rank (1–300), consistency score, number of reviews available, and which six reviews were actually used.

These per-model top-300 lists are the starting point for the cross-model agreement analysis.

5. Cross-model agreement and consensus author selection

To understand where the models agree and disagree, all per-model top-300 lists were combined into a single analysis workbook. For each author that appears in at least one top-300 list, the following quantities were computed:

- In how many models' top-300 lists the author appears (from 1 to 6).
- The author's average, median, and standard deviation of rank across the models where they appear.
- The per-model rank and consistency score.

From this combined view, a consensus criterion was defined:

- An author is considered a "consensus author" if they appear in the top-300 list of at least four out of the six models.

Under this criterion, 157 authors satisfy the consensus ≥4 condition. These are authors whose style is judged internally consistent by a clear majority of the models, not only by a single embedding.

Within this consensus pool, authors are further ordered by the median rank of the authors(ascending).

Our preference is to use 150 authors for the next stage. The working plan is therefore:

- Take the 157 consensus authors (appearing in at least 4 models' top-300).
- Sort them by number of supporting models and median rank, as above.
- Select the top 150 authors from this ordered list for LLM-based training and document generation.

This provides a principled balance between cross-model agreement and individual model rankings, and avoids over-reliance on any single embedding model.

6. Visual analysis: global t-SNE and "worst-case" authors

For each model, a global two-dimensional t-SNE projection has been constructed to visualise the structure of the embedding space:

- For authors with at least six reviews and a defined six-review cluster, only those six selected reviews are included in the t-SNE input.
- For authors with fewer than six reviews, all available reviews are included, so that they still appear in the background structure.
- Before t-SNE, embeddings can optionally be reduced via PCA (for example to 50 dimensions). t-SNE is then run with fixed hyperparameters (two components, fixed perplexity, PCA initialisation, and a fixed random seed) to keep plots comparable across models.

For each model, "worst of the best" plots were generated:

- Within that model's top-100 authors (i.e., the 100 most consistent according to that model's scores), authors ranked 91st to 100th are treated as the worst among the highly consistent group.
- For these authors, individual plots are created on top of the global t-SNE: all points are shown in light grey, and that author's six selected reviews are highlighted in a strong colour.
- This allows visual inspection of how tightly even the less-consistent "top authors" cluster within the embedding space, and how they sit relative to other authors.

These t-SNE plots have been prepared for all six models and will be useful later when interpreting how genuine and generated texts occupy the style space. The next step will be to relate these visual patterns specifically to the 150 selected consensus authors, once that subset is fixed.

7. API cost estimates for LLM training and generation

To anticipate the practical feasibility of the next stage, per-token API cost estimates were computed for several candidate LLM providers under different usage scenarios. The current spreadsheet models the following baseline assumptions:

- 100 authors (this will later be scaled to 150).
- 6 reviews per author.
- Approximately 800 words per review and 800 words per generated continuation.
- Two prompts per author and either one or three generations per prompt.

- One or two “full runs” (e.g., one pass for training data creation, one pass for evaluation).

For each provider, input and output token volumes are estimated under these assumptions and multiplied by published per-million-token prices to yield total cost per scenario. For 100 authors, the resulting cost ranges (in USD) are approximately:

Model	Minimum scenario (1 run, 1 gen)	Recommended scenario (2 runs, 1 gen)
gpt-5.1 (ChatGPT)	≈ 2.90	≈5.8
Gemini 3 Pro	≈ 7.60	≈15.19
Opus 4.1 (Claude)	≈ 50.74	≈101.48
Grok-4.1-fast-reasoning	≈ 0.47	≈0.94
DeepSeek-Chat / DeepSeek-Reasoner	≈ 0.54	≈1.08
Perplexity Sonar	≈ 1.72	≈ 3.44
Perplexity Sonar Pro	≈ 10.15	≈ 20.3
Perplexity Sonar Reasoning	≈ 3.38	≈ 6.77
Perplexity Sonar Reasoning Pro	≈ 5.93	≈ 11.87

All of these figures assume 100 authors. Scaling to 150 authors is straightforward and essentially linear; costs can be multiplied by a factor of 1.5. Even after scaling, the total budget for most models remains in a manageable range, especially for providers such as gpt-5.1, Gemini Pro, Grok, or DeepSeek.

The spreadsheet also notes that enterprise offerings such as Microsoft 365 Copilot Chat have seat-based licensing rather than simple per-token billing, and are therefore less convenient for fine-grained, per-experiment costing.

These estimates indicate that running one or two full passes of prompt–generation experiments for 150 authors should be feasible on typical research budgets, assuming we choose providers from the mid- to lower-cost range.

8. Summary and next steps

To summarise the current status:

- All authors in the selected AAVC variant have been embedded under six style models, with one compressed file per author per model.
- A consistent author-level style consistency metric has been defined and applied to each model, and top-300 lists have been generated.
- Cross-model agreement has been quantified, revealing that while individual models' rankings differ, there is a substantial group of authors that are judged stable by a majority of models.
- Using the criterion "appears in at least four models' top-300 list", 157 consensus authors have been identified. From these, the plan is to select 150 authors based on number of supporting models and average rank, and to use this set for the next phase.
- Global t-SNE projections and "worst-case" plots (authors ranked 91st–100th per model) have been generated, and they will serve as a visual baseline for later comparing genuine versus generated texts.
- Initial API cost modelling suggests that multi-pass generation experiments for 150 authors are affordable with several candidate LLM providers.

Proposed next steps are:

1. Finalise the list of 150 authors from the 157-author consensus pool using the majority-agreement and median-rank ordering described above.
2. For this 150-author set, generate a small number of additional visualisations (for example, highlighting them in the existing global t-SNE plots) to confirm that their clusters look well-behaved in the chosen style space.
3. Begin pilot authorship mimicry experiments on the selected 150 authors using GPT as the baseline.

I am happy to adjust the consensus criterion or the choice of primary embedding model based on your preferences and am ready to move to the LLM training and generation stage once we fix the final author list.