

Investigating Stability and Consensus in LLM Taxonomy Generation

Alison Yearsley^{1,*}, Anelia Kurteva^{1,*}

¹*School of Computer Science, University of Birmingham, United Kingdom*

Abstract

Taxonomies organise domain knowledge into structured hierarchies, but constructing them by hand is labour-intensive and demands expertise, motivating automation. Large language models (LLMs) are emerging as a candidate, but evaluating their output is an open challenge. Their outputs can be non-deterministic so it is unclear whether any single generation reflects a model's actual judgement, a judgement which is opaque itself. Without grounding text or a reference taxonomy, correctness cannot be measured directly, making *consistency* (whether repeated generations agree with one another) a central consideration. This paper presents a domain-agnostic pipeline, evaluated across LLMs from three providers, that quantifies run-to-run consistency with a label-agnostic stability measure and extracts a *consensus taxonomy*: the groupings that recur across many generations of the same input. Applying it to the Nikoli catalogue of pencil puzzles across nine models and 324 generations, F1 is high throughout (mean F1 ≈ 0.99) while consistency varies sharply across conditions, and that the configurations producing the richest taxonomies are not those producing the most stable ones. The consequences are twofold: a single LLM run cannot be treated as a reliable classification, yet a principled consensus structure can still be recovered from multiple unreliable runs without any reference taxonomy.

Keywords

Large language models, Taxonomy, Output stability, Self-consistency, Knowledge graphs

1. Introduction

LLMs perform well on knowledge-organisation tasks, yet their outputs are usually non-deterministic, their reasoning is opaque, and pre-training knowledge can bias their responses [1]. This leaves the *consistency* of LLM-generated taxonomies an open question. *Consistency* here meaning the degree to which repeated generations from the same model under fixed conditions produce the same groupings, distinct from *accuracy* (whether individual entities are correctly placed).

A growing body of work applies LLMs to organise items into groups and hierarchies [2, 3, 4, 5] via, for example, iterative querying, prompting, cluster consolidation of overlapping groups. The proposed pipeline follows this pattern, focusing on a different input. Rather than extracting relations from a grounding corpus, it operates over a bare entity list, making this taxonomy induction rather than information extraction, and causing the focus to be consistency, rather than correctness. The same prompt yields different outputs across seeds and temperatures [6]; the standard mitigation, self-consistency decoding, retains the most frequent sample [7]. Here instead this variability is treated as the object of study: a provider-agnostic pipeline with prompt formulation, rule provision, model, and temperature as independent variables, measuring stability via sibling co-occurrence and graph edit distance, and extracting the stable core as a partial consensus taxonomy. These label-agnostic alternatives are chosen as Cohen's κ [8], which requires a shared label set not available here.

The pipeline is demonstrated on *Nikoli* logic puzzles (e.g. Sudoku), where the absence of any accepted taxonomy prevents the model from recalling a memorised answer, and the entities are human identifiable. Motivated by this, our primary investigation addresses three research questions:

RQ1. *How do model choice, prompt constraint, temperature, and provided context affect the structural stability of generated taxonomies, measured by sibling co-occurrence and label-agnostic edit distance? When*

Posters, Demos, Blue Sky, and Tutorials at SEMANTiCS 2026, Sep 2026, Ghent, Belgium

*Corresponding author.

✉ alison@yearsley.org (A. Yearsley); a.kurteva@bham.ac.uk (A. Kurteva)

🆔 0009-0009-8374-0760 (A. Yearsley); 0000-0003-4512-5969 (A. Kurteva)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

variables are isolated model choice is shown to dominate, followed by prompt constraint; temperature acts as a multiplier on variance rather than a qualitative change, and context has the smallest seen effect. However when looking at aggregated data temperature appears to have the largest effect, and when relaxing the threshold of sibling co-occurrence context has a greater effect.

RQ2. *Does a domain-grounding document (here, a rules PDF) improve F1, precision and recall measures or consistency, and does it interact with prompt formulation or model capacity?* The PDF reduces hallucination and improves consistency but lowers recall, and interacts with weak prompts or smaller models to trigger the most extreme failures observed. The improvement in consistency is more pronounced when relaxing the binary threshold for sibling co-occurrence.

RQ3. *Can a principled method extract a partial consensus taxonomy from repeated, inconsistent generations without reference to a ground truth?* The stable-subgraph method recovers a conservative but coherent consensus core whose coverage tracks the underlying stability of the source runs.

2. Pipeline Overview

This paper proposes a three-stage pipeline¹: **Generate**, **Ingest**, and **Evaluate**, implemented in Python and communicating through persistent artefacts (JSON files and Neo4j nodes) rather than in-memory state. Figure 1 summarises the inputs and outputs of each stage.

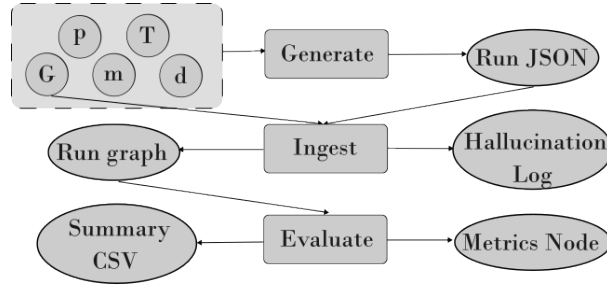


Figure 1: Pipeline Input Output Diagram. *G*: Reference game list, *p*: Prompt template, *m*: model, *T*: Temperature, *d*: Rules document (optional)

In the **Generate** stage, a model *m* is prompted with the reference game list *G* under a prompt template *p*, temperature *T*, and an optional rules document *d*, and instructed to produce a hierarchical taxonomy. Examples of prompts and rules are available in the project repository. Each invocation constitutes a single “run”; multiple runs are performed per configuration to assess output stability. A run uses a two-turn conversation: the first turn generates a free-form taxonomy from *G*; the second reformats it as a structured CSV, separating the classification task from the output-formatting task so that ingestion is independent of how the model originally presented its taxonomy. Both turns, together with the run metadata (*m*, *p*, *T*, *d*, run id), are serialised to a JSON record. The CSV is the contract to the Ingest stage, and has a fixed schema: shown in Figure 2, admitting any taxonomy of depth ≥ 1 .

In the **Ingest** stage, this CSV is parsed, each name is matched against *G* (unrecognised names recorded as hallucinations, treated as false positives and excluded from graphs; multi-category assignments deduplicated and logged), and the result is written to Neo4j, the resulting tree having a root node based on the runID. In the **Evaluate** stage, Cypher queries and Python scripts over these graphs compute: structural statistics, which describe individual taxonomies, and stability measures which assess consistency across runs. Outputs are written both to *Metrics* nodes and to summary CSVs.

A manifest CSV at each stage provides an audit trail, and the artefact-based design allows partial re-execution: failed ingestion does not require re-calling the API, and evaluation can be re-run without repeating either upstream stage.

¹<https://github.com/AlisonYearsley/GALT>; full prompt texts and experimental data available therein.

Source CSV Data Example:

Type, Name, Parent
category, Loop-drawing,
category, Loop on grid edges, Loop-drawing
game, Slitherlink, Loop on grid edges

Each line defines either a Category hierarchy path or maps a Game to its parent classification node.

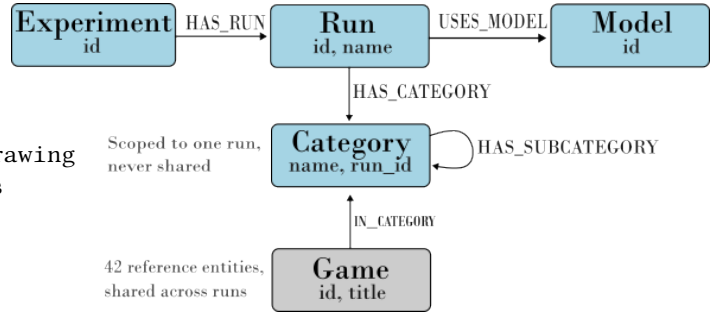


Figure 2: Graph schema produced by the Ingest stage compared against source input rows. Each CSV row becomes either a Category node or an IN_CATEGORY edge from an existing Game. Category nodes carry the run_id of the run that produced them, so two runs never share a category node even when they choose identical labels; Game nodes are the fixed reference set and are shared across runs. HAS_SUBCATEGORY is recursive, admitting any depth ≥ 1 .

3. Evaluation Metrics

With no ground-truth taxonomy, measurements are focused on three things: *coverage* (faithfulness to the input entities), *structure* (the shape of the taxonomy), and *stability* (consistency across runs). Coverage is the only metric with a definite failure mode; it is measured by F1 over the known entity set G , with unmatched names recorded as *hallucinations* and multi-category assignments deduplicated (first assignment kept).

Category and subcategory counts of each taxonomy are recorded (top-level and second-level groups); **game count** (leaf nodes here are game nodes, which may differ from $|G|$ under omission or hallucination); **category size distribution** (min, mean, max, and standard deviation of games per top-level category); and **average path length** (mean edges from root to leaf). Graph edit distance is calculated using the Hungarian algorithm, both run-to-subgraph to visualise dispersion and pairwise run-to-run to measure stability. This measure is implemented to investigate variance for analysis purposes, and is not a part of the subgraph extraction process.

Two games are *siblings* in a run if they share an immediate parent. For g entities there are $\binom{g}{2}$ possible pairs. Across r runs, *total pairs* is the number of distinct pairs that are siblings in at least one run, and *always-together pairs* the number that are siblings in *every* run. Stability is their ratio: $S = \frac{|\text{always-together pairs}|}{|\text{total pairs}|}$. $S = 1$ indicates identical groupings across all runs; $S = 0$ indicates no pair is consistently co-classified. The always-together pairs induce a *stable subgraph*: pairwise always-together entities form cliques extracted as stable groups, yielding a partial consensus taxonomy computed per-model and per-experiment. Some analysis has been done on relaxing this ‘always together’ threshold to instead be a tuneable parameter of percentage of times the pair is together, or instead produce a weighted relationship graph- these experiments are preliminary and introduce new complications beyond the scope of this pipeline, but this may be referenced as ‘co-occurrence threshold’ in later discussion of findings.

4. Experimental Setup

Nine models are chosen across three providers to span a range of capability tiers: Anthropic’s claude-haiku-4-5, claude-sonnet-4-6, and claude-opus-4-6; OpenAI’s gpt-5.4-nano, gpt-5.4-mini, and gpt-5.4; and Google’s gemini-2.5-flash-lite, gemini-2.5-flash, and gemini-2.5-pro.

Three prompt formulations test increasing constraint specificity: **Basic**, a minimal zero-shot instruction; **SpecifyEx**, which adds an explicit instruction to include only the provided entities; and **SpecifyExCats**, which further fixes exactly seven categories with no subcategories. The SpecifyExCats

category constraint is motivated by a pilot study showing the unconstrained models average ≈ 7 top-level categories across all three providers, and allows a narrower point of comparison to analyse cause of variance. Each prompt is run in one of two context conditions: the entity list alone, or the list plus a rules PDF (rule descriptions and images) intended to ground categorisation in game rules rather than name recognition alone.

Two temperature settings are tested: $T = 0$ (greedy decoding) and $T = 1$ (default stochastic sampling). The resulting 12 conditions (3 prompts $\times$ 2 temperatures $\times$ 2 contexts) were each run on all nine models three times, giving 324 generation calls. Three runs is a pragmatic baseline for surfacing patterns; selected conditions were later extended to 10 runs for analysis and investigating outliers.

5. Discussion

Coverage was near-perfect across all 324 runs (mean F1, recall and precision all ≈ 0.99), with ≈ 0.07 hallucinated names per run on average. The consistency measures range from near-perfect consensus to near-zero depending on configuration.

Table 1

Structural and consistency measures by factor. The final two columns describe consensus structure at 0.5 co-occurrence threshold

Factor	Level	n	Cats	Subcats	Depth	Games	F1	Halluc	S	Pairs ≥ 0.5	Games ≥ 0.5
Overall	all runs	324	6.22	7.66	2.60	41.0	0.995	0.068	0.285	40	28
Rules PDF	without	162	6.14	6.18	2.57	41.3	0.996	0.074	0.241	27	26
	with	162	6.29	9.15	2.63	40.7	0.993	0.062	0.328	57	28
Prompt	Basic	108	5.96	12.34	2.94	40.7	0.994	0.065	0.231	13	15
	SpecifyEx	108	5.86	10.65	2.87	41.0	0.993	0.065	0.249	32	28
	SpecifyExCats	108	6.82	0.00	2.00	41.3	0.998	0.074	0.374	79	32
Temperature	T0	162	6.25	7.57	2.59	40.9	0.996	0.062	0.391	42	28
	T1	162	6.18	7.76	2.61	41.0	0.994	0.074	0.178	38	27
Capability tier	high	108	5.62	9.81	2.73	41.2	0.998	0.065	0.281	57	29
	medium	108	6.52	7.80	2.56	40.9	0.996	0.056	0.237	46	30
	low	108	6.51	5.38	2.51	40.8	0.991	0.083	0.336	25	20
Provider	anthropic	108	6.34	10.36	2.69	42.0	0.999	0.009	0.304	36	25
	openai	108	7.10	5.66	2.40	40.4	0.993	0.093	0.208	48	24
	google	108	5.20	6.97	2.71	40.5	0.992	0.102	0.342	35	29

RQ1. The four independent variables (model, prompt constraint, temperature, and context) have different influence. Temperature has the highest observable effect on sibling co-occurrence doubling at $T=0$ compared to $T=1$. However when we relax that binary threshold this difference is much less pronounced and when other factors are held constant this difference is seen to be a multiplier on other factors impacts rather than an isolated driver. Lower temperature is the safer default where reproducibility matters, the marginally higher F1 at $T = 1$ may be a statistical artefact of error patterns averaging out across more variable outputs rather than a genuine advantage of stochastic decoding; this was not tested directly. For structural effects SpecifyExCats shows the greatest disparity but this largely reflects the imposed flat seven-category shape- showing convergence on a constraint rather than spontaneous taxonomic agreement. Looking at individual runs model choice can be seen to dominate: setting the kind of taxonomy a provider tends to produce before any prompt takes effect, with anthropic, and higher capacity models having notably more subcategories but only marginally less categories- an indication that the categories are more nuanced and less bloated. And one instance of an anthropic model instead outputting a meta-taxonomy of games organised according to different factors. Holding prompt, temperature, and context fixed (SpecifyExCats, list, $T = 0$), mean run-to-run graph edit distance still ranges from 0.0 for Gemini-Flash-Lite to 12.4 for Gemini-Pro.

RQ2. Attaching the rules PDF reduces hallucination and improves consistency, along with increasing the subcategory depth, confirming that rules expose mechanical distinctions that names alone do not. But the benefit is conditional on sufficient prompt structure: every extreme failure in the dataset (structural collapse into a single category, recall loss in the smallest model, near-star graphs) occurs only under the PDF paired with an underspecified prompt. The PDF exposes a model’s inability to reconcile a long document with a weak instruction; a structured prompt retains the benefits without triggering these failures.

RQ3. The stable-subgraph method successfully recovers meaningful partial taxonomies with no ground truth across multiple invocations. The groupings that survive are interpretable even without defining category names: a loop-drawing core, a connectivity-constrained shading cluster, and a small number of tightly paired games recur across models and conditions, while a few games remain persistently unstable. The strict unanimity criterion is conservative by design and doesn’t surface any pairs when comparing all the combinations together. However relaxing the co-occurrence threshold suggests substantial agreement lies below the binary threshold and warrants further investigation. As the model’s reasoning chain is not directly interpretable, validating semantic accuracy would require an external reference taxonomy- which, if available, would complement the stability metrics by grounding a notion of correctness.

Findings. Increasing constraint does not consistently improve compliance: the explicit-exclusion prompt achieved *lower* F1 than the unconstrained baseline in smaller Gemini and GPT models (behaviour absent in the Claude family) suggesting boundary instructions can misdirect attention in models with weaker instruction-following. There is a stability-depth trade-off. Tables 3a and 3b illustrate this: Haiku’s constrained output is complete with all 42 games, identical across runs. Sonnet’s is deep and rule-grounded but only 5 of 47 observed pairs (stability = 0.106) survive unanimously. Only game pairs consistently co-assigned across all runs are shown. Subcategory labels are drawn from the modal run. The choice of configuration should therefore follow the intended use: applications needing a single taxonomy favour heavy constraint. Applications where a human curator will prune and merge favour the richer, less stable output. All applications should consider outputs as guidance rather than authoritative without further analysis of correctness. Single-run output cannot be treated as a reliable classification; downstream use must either sample-and-consolidate or report stability alongside results.

Category	Games
Number Placement	Sudoku, Ripple Effect, Juosan, Nansuke
Path & Connection	Hashiwokakero, Numberlink, Slitherlink, Masyu, Nagareru, Dotchi-Loop, Mid-loop
Region Division	Shikaku, Fillomino, LITS, Evolomino, Five Cells, Sashigane
Cell Colouring	Nurikabe, Hitori, Kurodoko, Kurotto, Norinori, Heyawake, Satogaeri, Yajilin, Usowan
Block Placement	Chained Block, Choco Banana, Double Choco, Nuritwin, Tilepaint, Shakashaka, Sto-stone
Line Drawing	Akari, Hebi, Makaro, Moon-or-Sun, Nurimisaki, Pencils, Suraromu
Hybrid & Complex	Nurimeizu, Yosenabe

Top-level category	Stable subcategory	Games
Loop-Drawing	Unconstrained Path	Masyu, Slitherlink
Cell-Shading	Connectivity-Constrained	Kurodoko, Nurikabe
Cell-Shading	Neighbour-Count Shading	Kurotto, Usowan
Number-Placement	Region-Based Number Fill	Makaro, Ripple Effect
Path / Movement	Circle/Token Movement	Satogaeri, Yosenabe

(a) Proposed taxonomy by Claude Haiku, *SpecifyExCats*, list context, $T = 0$.

(b) Stable core: Claude Sonnet, *SpecifyEx*, rules context, $T = 0$.

Figure 3: Example subgraph taxonomies formatted as tables- showing stability-richness trade-off

6. Limitations and Future Work

Three runs for most configurations means conclusions are directional rather than definitive, and reported statistics are indicative rather than precise, as no significance testing was performed. All experiments use a single domain, and generalisation requires further exploration. The *SpecifyExCats* prompt’s fixed seven-category constraint may disadvantage models with different natural categorisations. Its stability gains partly reflect convergence on the imposed shape rather than spontaneous agreement; this was not accounted for in aggregate statistical measures. The three **prompt formulations** are simple specificity progressions. Chain-of-thought and iterative refinement were not investigated. The **metrics** are deliberately structural and do not assess semantic meaning. **Reproducibility** is constrained by dependence on availability of provider-hosted model versions; combined with residual nondeterminism at $T = 0$, exact numerical results may not reproduce in future testing, though qualitative patterns should persist. The reformatting step, of text to CSV, could in principle merge, split, or drop groupings.

However, 2 runs from each provider were checked and were found to be consistent, and this kind of text formatting is a known strength of LLMs.

Future work Preliminary experiments surfaced out-of-scope behaviours (meta-taxonomies, model self-evaluation, facet-graph generation, UI vs. API differences). These were out of scope of this paper so experiments used were designed to exclude these considerations which each merit dedicated study. Clear extensions include applying the pipeline to other taxonomic domains to test generalisation to other data sources and sizes, exploring alternative PDF presentation formats to retain precision benefits without recall cost, and properly exploring subgraph extraction methods that operate over weighted rather than binary co-occurrence, potentially recovering larger consensus structures from moderate-stability data- this will be particularly beneficial when calculating over a higher number of runs. To further analyse slight divergences in consensus it could also be worth exploring the distribution of the next token prediction when considering open weight models.

Declaration on Generative AI

The authors used Claude for minor paraphrasing and draft preparation following validation. After using Claude, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

7. Project Files

<https://github.com/AlisonYearsley/GALT>

References

- [1] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu, A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, *ACM Transactions on Information Systems* 43 (2025) 1–44. doi:10.1145/3703155.
- [2] V. Viswanathan, K. Gashteovski, C. Lawrence, T. Wu, G. Neubig, Large language models enable few-shot clustering, *Transactions of the Association for Computational Linguistics* 12 (2024) 321–350. doi:10.1162/tacl_a_00648.
- [3] C. M. Pham, A. Hoyle, S. Sun, P. Resnik, M. Iyyer, TopicGPT: A prompt-based topic modeling framework, in: *Proceedings of NAACL*, 2024, pp. 2956–2984.
- [4] Q. Zeng, Y. Bai, Z. Tan, S. Feng, Z. Liang, Z. Zhang, M. Jiang, Chain-of-layer: Iteratively prompting large language models for taxonomy induction from limited examples, in: *Proceedings of CIKM*, 2024, pp. 3093–3102.
- [5] P. Kargupta, N. Zhang, Y. Zhang, R. Zhang, P. Mitra, J. Han, TaxoAdapt: Aligning LLM-based multidimensional taxonomy construction to evolving research corpora, in: *Proceedings of ACL*, 2025, pp. 29834–29850. doi:10.18653/v1/2025.acl-long.1442.
- [6] M. Renze, E. Guven, The effect of sampling temperature on problem solving in large language models, in: *Findings of the Association for Computational Linguistics: EMNLP*, 2024.
- [7] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, D. Zhou, Self-consistency improves chain of thought reasoning in language models, in: *Proceedings of ICLR*, 2023.
- [8] J. Cohen, A coefficient of agreement for nominal scales, *Educational and Psychological Measurement* 20 (1960) 37–46. doi:10.1177/001316446002000104.