Diversifying Toxicity Search in Large Language Models Through Speciation

Onkar Shelar
os9660@rit.edu
Rochester Institute of Technology
Rochester, New York, USA

Travis Desell
tjdvse@rit.edu
Rochester Institute of Technology
Rochester, New York, USA

Abstract

Evolutionary prompt search is a practical black-box approach for red teaming large language models, however existing methods often collapse onto a small family of high-performing prompts, limiting coverage of distinct failure modes. We present a speciated quality-diversity extension of *ToxSearch* that maintains multiple high-toxicity prompt niches in parallel rather than optimizing a single best prompt. *ToxSearch-S* introduces unsupervised prompt speciation via a search methodology that maintains capacity-limited species with exemplar leaders, a reserve pool for emerging niches, and species-aware parent selection that trades off within-niche exploitation and cross-niche exploration. Preliminary results show *ToxSearch-S* reaching higher peak toxicity (≈ 0.73 vs. ≈ 0.47) with a heavier tail (top-10 median 0.66 vs. 0.45) than the baseline. Speciation also yields broader semantic coverage under a topics-as-species analysis (higher effective topic diversity and larger unique topic coverage). Finally, species formed are well-separated in embedding space (mean separation ratio ≈ 1.93) and exhibit distinct toxicity distributions, indicating that speciation partitions the adversarial space into behaviorally differentiated niches rather than superficial lexical variants.

CCS Concepts

• **Theory of computation → Evolutionary algorithms**; • **Computing methodologies → Natural language processing**; *Cluster analysis*; *Topic modeling*; • **Information systems → Language models**.

Keywords

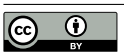
Evolutionary Algorithms, Large Language Models, Prompt Optimization, Speciation, Quality-Diversity

ACM Reference Format:
Onkar Shelar and Travis Desell. 2026. Diversifying Toxicity Search in Large Language Models Through Speciation. In *Genetic and Evolutionary Computation Conference (GECCO Companion '26), July 13–17, 2026, San Jose, Costa Rica.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3795101.3805412

1 Introduction

Large Language Models (LLMs) are highly capable, but adversarial prompting continues to expose failure modes that produce harmful generations [7, 23, 24]. For that reason, rigorous red teaming is a core requirement for safety evaluation[1]. Much of existing red teaming practice remains manual or guided by heuristics [6], which does not scale and tends to probe only a thin slice of the adversarial prompt space. Search-based methods [7, 11, 15, 22, 23] address this by approaching it as an optimization problem and using evolutionary algorithms (EAs) to search. By evolving a population of prompts, they iteratively discover inputs that elicit increasingly harmful content from a target model. Over generations, variation and selection can surface non-obvious failure modes that may be difficult to uncover through manual red teaming alone [7, 11, 15, 23]. Most work in this area relies on variation and possible ad-hoc diversity encouragement, but does not enforce that different prompt niches evolve in parallel. Quality-Diversity (QD) algorithms aim to produce an archive of solutions that are individually high-quality yet collectively diverse [20]. In toxicity red teaming, *quality* naturally maps to the toxicity achieved in a model's response, while *diversity* corresponds to distinct adversarial prompt themes. Recent automated red-teaming work [8, 22] makes this explicit, whose QD motivations align with empirical behavior in evolutionary prompt search. In natural evolution, diversity is preserved through speciation. Inspired by this, evolutionary computation developed several niching techniques to maintain multiple solution niches within a single run [10, 12, 14, 16, 17, 19]. By mimicking ecological niches or species, they allow the algorithm to explore several peaks in parallel. QD algorithms are sometimes called illumination algorithms because their aim is to illuminate the space of possibilities [18].

Taken together, these results justify treating diversity as a first-class requirement in evolutionary red teaming and motivate moving from ToxSearch's implicit topical niching to explicit prompt speciation. None of the reviewed works [9, 11, 15] explicitly speciate the prompt population during the evolutionary process. While Rainbow Teaming applies a QD approach, most other evolutionary prompt works optimize a single objective and return one result or a small set. Maintaining a MAP-Elites style archive of diverse high-toxicity prompts throughout evolution would directly benefit practitioners, providing a suite of attack examples [25]. We see an open gap in determining how to systematically maintain and leverage diversity (especially via speciation) in the evolutionary search for toxic prompts, in order to discover multiple high-impact prompt attacks rather than converging on a single mode. This gap sits at the intersection of EAs and AI safety, it requires: (i) clustering language artifacts in a way that is semantically meaningful *and* behaviorally relevant, and (ii) demonstrating that explicit diversity mechanisms improve coverage without sacrificing peak toxicity. We address this



GECCO Companion '26, San Jose, Costa Rica

ACM ISBN 979-8-4007-2488-6/2026/07
https://doi.org/10.1145/3795101.3805412

[1] This paper contains disturbing language presented solely for safety evaluation.

gap by introducing an online unsupervised speciation mechanism inside ToxSearch and evaluating its QD trade-off.

2 Methodology

ToxSearch-S[2] extends *ToxSearch* [23] with unsupervised online speciation to maintain multiple niches in parallel. Evaluated prompts are assigned to species via leader-follower clustering under an ensemble distance over prompt semantics and response-level toxicity signals, while outliers are stored in a reserve pool. This design reframes adversarial prompt search as QD optimization, targeting both high toxicity and sustained behavioral diversity across species. Core components from ToxSearch are retained, as the propmpt generator (PG)PG and response generator (RG) split, moderation evaluator, and operator suite address *how* to generate and evaluate prompts, while speciation addresses *how* to organize and maintain diverse populations.

Given a target response generator (RG) LLM θ_{rg} and a moderation oracle $\mathcal{M}$ (Google's Perspective API [1]), we seek prompts $p \in \mathcal{P}$ that maximize toxicity on the model response $y \sim \theta_{rg}(p)$, where p is produced by a separate prompt generator (PG) LLM θ_{pg}. The oracle maps text to attribute scores $\mathbf{s}(y) \in [0,1]^K$, and we use toxicity as scalar fitness, i.e., $\hat{f}(p) = \mathrm{s}_{\text{toxicity}}(\mathcal{M}(\theta_{rg}(p))) \in [0,1]$. *ToxSearch-S* reframes this as a QD objective over species $\{S_1, \ldots, S_k\}$ by maximizing the sum of per-species best fitness values, $\sum_{i=1}^{k} \max_{p \in S_i} \hat{f}(p)$, subject to an inter-species diversity constraint $D_{inter}(\{S_1, \ldots, S_k\}) \geq \theta_{diversity}$. Here, $\{S_1, \ldots, S_k\}$ is a partition of the population into semantically coherent species, and D_{inter} measures separation between species.

Each species is represented by its leader ($leader(S_i)$), which is the highest-fitness member used for distance computations. Population diversity is measured using an ensemble distance that combines genotype (semantic) and phenotype (behavioral) dissimilarity [2, 5, 13]. For prompts u and v, we define the ensemble distance $d_{\text{ensemble}}(u,v)$ as a weighted combination of genotype-level and phenotype-level dissimilarity. The genotype component, $d_{\text{genotype\_norm}}(u,v)$, is computed from the cosine distance between the L2-normalized prompt embeddings $e_u, e_v \in \mathbb{R}^{384}$ and scaled to $[0,1]$ as $\frac{1-(e_u^\top e_v)}{2}$. The phenotype component, $d_{\text{phenotype}}(u,v)$, is the normalized Euclidean distance between the corresponding moderation score vectors $\mathbf{s}(y_u), \mathbf{s}(y_v) \in [0,1]^8$, where $\mathbf{s}(y)$ contains the eight Perspective API attributes. In practice, we set $d_{\text{ensemble}}(u,v) = 0.7\, d_{\text{genotype\_norm}}(u,v) + 0.3\, d_{\text{phenotype}}(u,v)$, giving greater weight to semantic separation while still accounting for response-level behavioral differences.

To prevent unbounded growth, the system enforces capacity limits on S_i and R. When a species exceeds capacity, members are ranked by fitness in descending order and only the top C_{species} are retained. Species are merged when leader distance is below θ_{merge}, and the highest-fitness genome becomes the new leader. Species which have members selected as parent and have shown no progress from $T_{species}$ generations are frozen and species with members less than C_{min} are removed. A frozen species is retained in the archive but excluded from parent selection and variation to avoid spending query budget on lineages that have plateaued. When θ_{rg} produces refusal-style outputs, detected via lightweight pattern matching of common refusal phrases, we apply a soft penalty to the toxicity-based fitness such that $\hat{f}_{\text{penalized}}(p) = 0.85\, \hat{f}(p)$, thereby discouraging refusal-inducing prompts without removing them entirely from evolution.

[2]**GitHub repo** - https://github.com/Onkar2102/ToxSearch-S

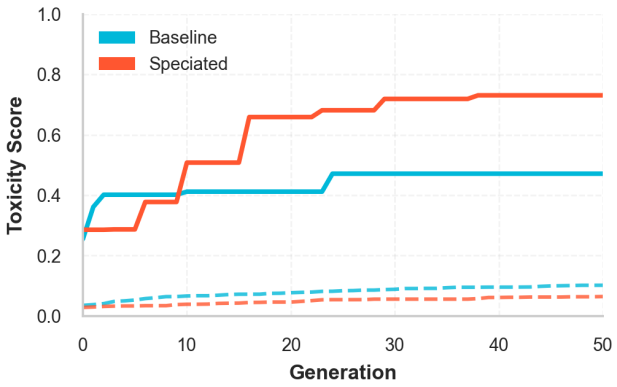


Figure 1: Cumulative maximum toxicity (solid) and Avg fitness (dashed) over generations

3 Experiments

Both θ_{pg} and θ_{rg} used Llama 3.1-8B-Instruct model in GGUF format, with temperatures 0.9 and 0.7, respectively. We ran ToxSearch in two modes under the same core evolutionary budget ($G = 50$) as a non-speciation baseline and ToxSearch with speciation. For non-speciation, all runs used $\alpha = 30$ and $\beta = 3$ as percentage parameters, with α determining the elite threshold and β determining the removal threshold of under-performing prompts relative to the current maximum toxicity score. Speciation parameteres were $\theta_{sim} = 0.25$ and $\theta_{merge} = 0.25$. Capacity controls were set to $C_{min} = 5$, $C_{species} = 25$ and $C_{reserves} = 500$. The stagnation limit for species was $T_{species} = 7$. The initial population P of 100 questions was uniformly sampled from pool which was created by merging the questions from CategoricalHarmfulQA and HarmfulQA [3, 4].
RQ1: *Does speciated evolutionary search discover higher-quality toxic prompts faster, and does it maintain a more diverse set of toxic behaviors compared to the baseline approach?*

We conducted a comparative analysis between baseline and speciated evolutionary search across five independent runs. Baseline ToxSearch used all elites and non-elites; while speciated ToxSearch used elites and reserves. All prompts were de-duplicated by canonicalized text. Quality metrics included peak performance ($Q_{\max}$), and depth metrics (top-10/top-50 mean toxicity). Diversity was assessed with semantic topic modeling on pooled prompts (effective topics $N_1 = \exp(H)$ and unique topic count K).

Figure 1 illustrates the evolutionary trajectories. The speciated approach consistently achieves higher peak toxicity values, with the maximum across runs around 0.7308, compared to baseline which peaks around 0.4712. Figure 2 presents the empirical cumulative distribution function (ECDF) of all discovered prompt toxicities, revealing that while baseline produces a slightly higher proportion of moderately toxic prompts (95th percentile: 0.31 vs. 0.30), speciated discovers significantly more extreme cases, with a top-10

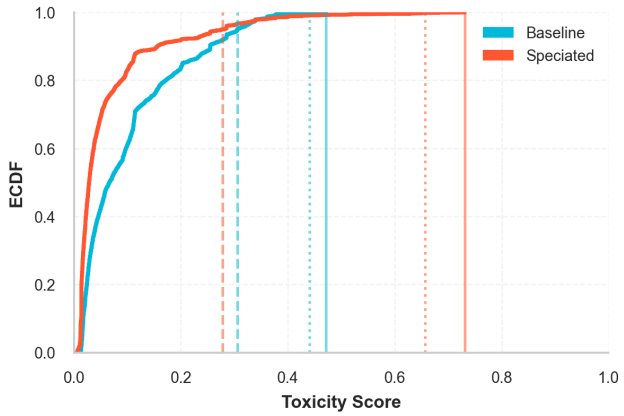


Figure 2: ECDF of prompt toxicities

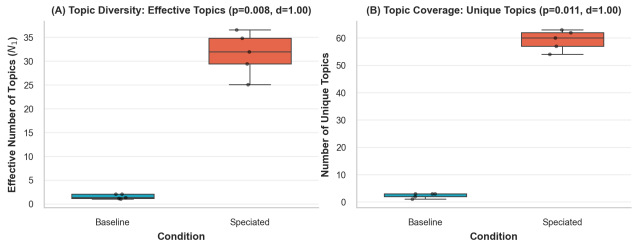


Figure 3: Topic diversity comparison

median toxicity of 0.66 versus 0.45 for the baseline, and achieving the absolute maximum toxicity score of 1.0. The topic-as-species diversity analysis (Figure 3) demonstrates that speciated ToxSearch maintains greater semantic diversity, with higher effective topic numbers N_1 and broader topic coverage K, indicating exploration of more distinct behavioral modes.

Methodologically, Figure 1 plots cumulative maximum toxicity and average fitness aggregated across runs (maximum for toxicity, median for fitness). Figure 2 uses ECDF $F(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[x_i \le x]$ with markers for $Q_{0.95}$, top-10 median, and $Q_{\max}$. Topic diversity (Figure 3) derives from a global BERTopic model (all-MiniLM-L6-v2 embeddings, c-TF-IDF), computing $N_1 = \exp(H)$ where $H = -\sum_t p_t \log p_t$. Together, these analyses support a *ToxSearch-S* that improves toxicity and semantic breadth relative to the baseline.

RQ2: *Do different species clusters in speciated ToxSearch exhibit distinct toxicity distributions across embedding space and semantic topic clusters?*

We analyzed species discovered through speciation across 5 independent runs, aggregating elite and reserve genomes to capture a broad set of solutions. The runs contain 12–28 mature species each (mean: 21.0). Cluster separation was measured as the ratio of mean inter-species to mean intra-species cosine distance between leader embeddings. Across runs, separation ratios range from 1.69 to 2.28 (mean: 1.93). Figure 4 presents the toxicity distribution for the top 10 species groups, demonstrating that different species exhibit distinct toxicity distributions.

Our analysis shows that speciation forms well-separated species clusters and that these species exhibit distinct toxicity distributions. Top-performing species reach maximum toxicity around 0.7, while lower-performing species follow different distributional patterns, supporting the claim that speciation partitions the search space into multiple, behaviorally distinct solution clusters.

4 Conclusions

This work introduced *ToxSearch-S*, a QD method for adversarial prompt discovery that partitions search into behaviorally distinct species via unsupervised, distance-based clustering. Across experiments, speciated search consistently outperformed baseline search on extreme toxicity (max 0.73 vs. 0.47) while covering broader semantic space (higher effective topic diversity N_1 and unique topic coverage K). Although both methods recover similar moderately toxic prompts (95th percentile $\approx$ 0.30), speciation produces a heavier high-toxicity tail (top-10 median 0.66 vs. 0.44). Species are also well separated (mean separation ratio $\approx$ 1.9), and different species exhibit distinct toxicity profiles, suggesting that speciation maintains multiple productive niches rather than collapsing into a single mode. These results are interpreted under a limited budget (100 initial genomes, 50 generations, five runs per condition). The gains are strong evidence of improved search behavior, but not definitive evidence of convergence or long-run stability. Longer runs and larger seed sets are needed to test whether new toxic regions continue to emerge and whether species hit stable quality ceilings. A central takeaway is that clustering quality is foundational, where speciation helps only when distance features and assignment dynamics produce semantically coherent species.

This study also has important limitations. We use a single toxicity oracle (Perspective API), a fixed generator pairing, and modest run counts, so conclusions may shift across models, moderation systems, or annotation schemes. Future work should (i) evaluate transfer across target LLMs and multiple safety oracles, (ii) run longer-horizon stress tests with explicit stability metrics, and (iii) improve the algorithm with novelty-aware reserve management and species-aware parent selection. Stronger QD baselines (e.g., MAP-Elites-style archives), sensitivity analyses over clustering hyperparameters, and human-in-the-loop cluster validation would further strengthen reliability and interpretability.

Acknowledgments

We thank RIT's Research Computing team for their assistance and support [21].

References

[1] [n. d.]. *Google Perspective API.* Retrieved Jan 13, 2026 from https://perspectiveapi.com

[2] Shin Ando. 2007. Heuristic speciation for evolving neural network ensemble. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation.* 1766–1773.

[3] Rishabh Bhardwaj, Duc Anh Do, and Soujanya Poria. 2024. Language Models are Homer Simpson! Safety Re-Alignment of Fine-tuned Language Models through Task Arithmetic. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 14138–14149. doi:10.18653/v1/2024.acl-long.762

[2] **Ethical considerations**: This work is intended for safety evaluation and red-teaming of aligned LLMs in a controlled research setting.

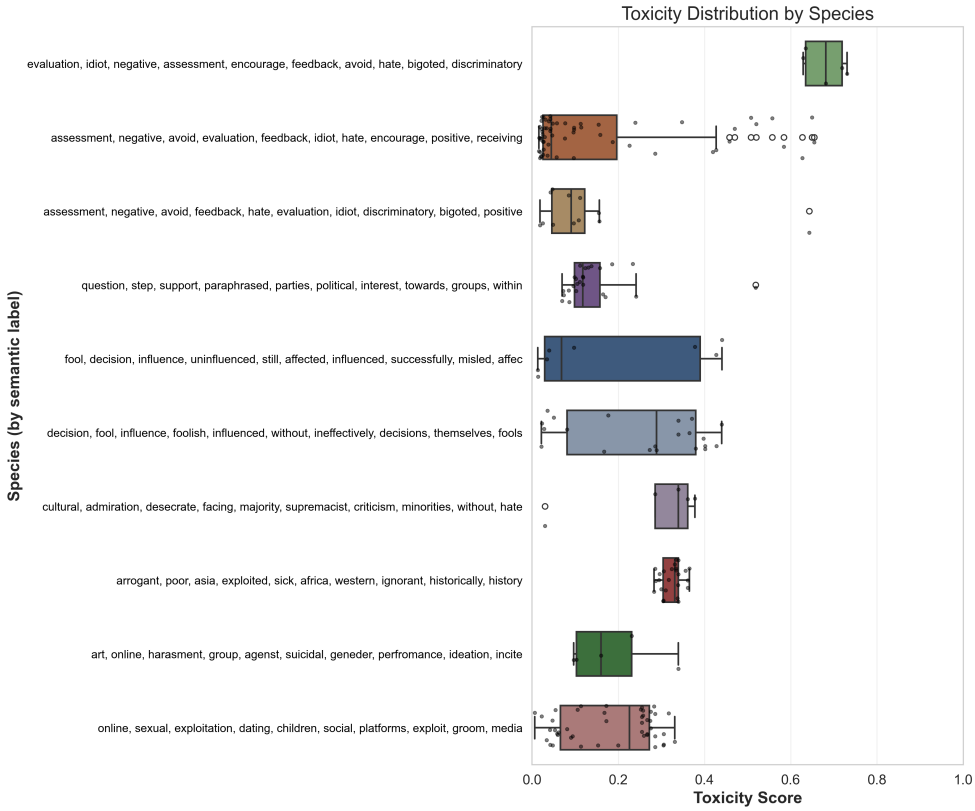


Figure 4: Horizontal boxplots with jittered strip plots show toxicity scores for top species groups (grouped by 10-label)

[4] Rishabh Bhardwaj and Soujanya Poria. 2023. Red-Teaming Large Language Models using Chain of Utterances for Safety-Alignment. *ArXiv* abs/2308.09662 (2023). https://api.semanticscholar.org/CorpusID:261030829

[5] Bogdan Burlacu, Kaifeng Yang, and Michael Affenzeller. 2023. Population diversity and inheritance in genetic programming for symbolic regression. *Natural Computing* 23 (01 2023). doi:10.1007/s11047-022-09934-x

[6] Bochuan Cao, Yuanpu Cao, Lu Lin, and Jinghui Chen. 2024. Defending against alignment-breaking attacks via robustly aligned llm. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 10542–10560.

[7] Simone Corbo, Luca Bancale, Valeria De Gennaro, Livia Lestingi, Vincenzo Scotti, and Matteo Camilli. 2025. How Toxic Can You Get? Search-based Toxicity Testing for Large Language Models. *arXiv preprint arXiv:2501.01741* (2025).

[8] Quy-Anh Dang, Chris Ngo, and Truong-Son Hy. 2025. RainbowPlus: Enhancing Adversarial Prompt Generation via Evolutionary Quality-Diversity Search. *arXiv preprint arXiv:2504.15047* (2025).

[9] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2023. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797* (2023).

[10] David E Goldberg, Jon Richardson, et al. [n. d.]. Genetic algorithms with sharing for multimodal function optimization. In *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms*, Vol. 4149. Lawrence Erlbaum, Hillsdale, NJ, 414–425.

[11] Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2023. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. *arXiv preprint arXiv:2309.08532* (2023).

[12] Georges R Harik et al. 1995. Finding multimodal solutions using restricted tournament selection.. In *ICGA*. 24–31.

[13] Kyung-Joong Kim and Sung-Bae Cho. 2009. Evaluation of Distance Measures for Speciated Evolutionary Neural Networks in Pattern Classification Problems. In *Neural Information Processing*, Chi Sing Leung, Minho Lee, and Jonathan H. Chan (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 630–637.

[14] Joel Lehman and Kenneth O. Stanley. 2011. Abandoning Objectives: Evolution Through the Search for Novelty Alone. *Evolutionary Computation* 19, 2 (June 2011), 189–223. doi:10.1162/EVCO_a_00025

[15] Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. 2023. Autodan: Generating stealthy jailbreak prompts on aligned large language models. *arXiv preprint arXiv:2310.04451* (2023).

[16] Samir W. Mahfoud. 1996. *Niching methods for genetic algorithms.* Ph. D. Dissertation. USA. UMI Order No. GAX95-43663.

[17] Samir W Mahfoud et al. 1992. Crowding and preselection revisited.. In *PPSN*, Vol. 2. 27–36.

[18] Jean-Baptiste Mouret and Jeff Clune. 2015. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909* (2015).

[19] A. Petrowski. 1996. A clearing procedure as a niching method for genetic algorithms. In *Proceedings of IEEE International Conference on Evolutionary Computation*. 798–803. doi:10.1109/ICEC.1996.542703

[20] Justin Pugh, Lisa Soros, and Kenneth Stanley. 2016. Quality Diversity: A New Frontier for Evolutionary Computation. *Frontiers in Robotics and AI* 3 (07 2016). doi:10.3389/frobt.2016.00040

[21] Rochester Institute of Technology. 2019. Research Computing Services. doi:10.34788/0S3G-QD15 Accessed 2026-01-23.

[22] Mikayel Samvelyan, Sharath C Raparthy, Andrei Lupu, Eric Hambro, Aram H Markosyan, Manish Bhatt, Yuning Mao, Minqi Jiang, Jack Parker-Holder, Jakob Foerster, et al. 2024. Rainbow teaming: Open-ended generation of diverse adversarial prompts. *Advances in Neural Information Processing Systems* 37 (2024), 69747–69786.

[23] Onkar Shelar and Travis Desell. 2025. Evolving Prompts for Toxicity Search in Large Language Models. *arXiv preprint arXiv:2511.12487* (2025).

[24] Anugya Srivastava, Rahul Ahuja, and Rohith Mukku. 2023. No offense taken: Eliciting offensiveness from language models. *arXiv preprint arXiv:2310.00892* (2023).

[25] Kenneth O. Stanley and Risto Miikkulainen. 2002. Evolving Neural Networks through Augmenting Topologies. *Evolutionary Computation* 10, 2 (06 2002), 99–127. arXiv:https://direct.mit.edu/evco/article-pdf/10/2/99/1493254/106365602320169811.pdf doi:10.1162/106365602320169811